



UNIVERSIDAD
NACIONAL
DE LA PLATA

FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

TÍTULO: Accesibilidad en videojuegos: su aporte a la inclusión y al proceso de creación de videojuegos inclusivos en la industria independiente

AUTORES: Federico Pacheco

DIRECTOR/A: Claudia Queiruga, Ivana Harari

CODIRECTOR/A: -

ASESOR/A PROFESIONAL: -

CARRERA: Licenciatura en Sistemas

Resumen

El presente trabajo de tesina de grado abarca la investigación, el estudio y análisis de la accesibilidad en videojuegos independientes, el desarrollo del plugin "Godot Subtitles" para el motor de videojuegos Godot Engine y su publicación en el repositorio de software GitHub, y la elaboración de un "Catálogo de buenas prácticas en el diseño de accesibilidad de videojuegos". En este trabajo se realiza un estudio sobre el surgimiento de la accesibilidad desde sus inicios hasta el presente, sobre su impacto en la era digital, haciendo foco en el desarrollo de videojuegos. Los conceptos y temáticas asociadas a la accesibilidad se analizan desde el punto de vista del jugador y del desarrollador, con especial atención en las discapacidades auditivas e intentando comprender la relación que existe entre estas dos perspectivas. Se analizan varios casos de estudio de desarrollos independientes para entender las limitaciones y dificultades en el desarrollo que pueden impactar en la accesibilidad.

Palabras Clave

Accesibilidad, Inclusión, Diversidad, Videojuegos, Industria Independiente, Indies, Godot Engine, Plugin, Subtítulos, Diseño Accesible.

Conclusiones

Del presente trabajo es posible concluir que para el sector independiente de desarrollo de videojuegos y en particular la comunidad de desarrolladores que utilizan Godot Engine, la implementación de estrategias de accesibilidad está desvalorizada frente a otro tipo de tareas dentro del proceso de desarrollo. Por ello disponer de herramientas como un plugin que agrega soporte para subtítulos en Godot Engine, o un catálogo de buenas prácticas en el diseño de la accesibilidad, facilitan la implementación de dichas estrategias y representan un avance en la incorporación de accesibilidad en los productos realizados dentro de este sector de la industria.

Trabajos Realizados

Este trabajo incluye el desarrollo de un catálogo de buenas prácticas en el diseño de accesibilidad en videojuegos. Adicionalmente se realiza un estudio de accesibilidad del motor para desarrollo de videojuegos Godot Engine, y se desarrolla un plugin que agrega soporte para subtítulos, publicado en la web oficial de dicho motor. Finalmente se realiza el desarrollo y testeado de un prototipo de videojuego accesible, mediante una metodología iterativa con foco en la mejora de la accesibilidad, publicado en un repositorio público de Github.

Trabajos Futuros

Continuar con el análisis del impacto de la accesibilidad en equipos de desarrollo independiente e implementarlas en el prototipo, haciendo foco en otros tipos de discapacidades además de la auditiva. Extender el catálogo de buenas prácticas en el diseño de accesibilidad con foco en otras discapacidades. Incorporar mejoras de personalización en el plugin "Godot Subtitles".

Índice General

Índice General	2
Capítulo 1: Introducción	8
1.1 Motivación: la problemática de la accesibilidad en la industria independiente de videojuegos	8
1.2 Objetivos	10
Objetivo general	10
Objetivos específicos	10
1.3 Estructura del documento	10
Capítulo 2: Accesibilidad	12
2.1 El concepto de accesibilidad. Sus orígenes y enfoques.	12
Los enfoques de integración e inclusión y su relación con la accesibilidad	12
Surgimiento y evolución del concepto de accesibilidad	14
2.2 Accesibilidad digital: ¿Qué es? ¿Qué implica?	16
Pautas de Accesibilidad al contenido web: WCAG	19
Las WCAG y su aplicación a la construcción de videojuegos accesibles	20
2.3 Accesibilidad en Argentina. Marcos normativos	20
El enfoque asistencialista de la ley de discapacidad	20
El camino hacia un enfoque de derecho humano en la ley de discapacidad	21
2.4 Accesibilidad en videojuegos	22
Evolución del concepto de accesibilidad aplicada en videojuegos	25
2.5 Diversidad de jugadores	29
2.6 Diversidad funcional en los videojuegos	31
Tecnologías de asistencia	31
Dificultades comunes en la interacción con videojuegos	32
Dificultades en la visión	34
Dificultades en la audición	37
Dificultades en la movilidad	39
Dificultades cognitivas	41
Relación entre usabilidad y accesibilidad en videojuegos	43
2.7 Tecnologías de asistencia para jugadores	45
Teclados adaptados	46
Dispositivos de entrada alternativos	47
Puntero cefálico	47
Dispositivo de botón o interruptor simple	47
Interruptores “sip-and-puff”	49
Software de seguimiento de la mirada y la cabeza	49
Herramientas de comunicación alternativa	51
Línea braille	52
Software de dictado	54
Magnificador o lupa digital	54

Software de ampliación de pantalla	55
Software de texto a voz	55
Software lector de pantalla	56
Software de reconocimiento óptico de caracteres	57
2.8 Accesibilidad en sistemas operativos	57
Capítulo 3: Industria independiente y accesibilidad	61
3.1 Estado actual de la industria del videojuego	61
Presente y futuro	62
Tendencias del mercado global	64
El estado actual de la industria en Argentina y Latinoamérica	68
Características del sector independiente de videojuegos	72
Independencia financiera	72
Independencia creativa	73
Independencia editorial	73
Un acercamiento empírico a la definición de “juego independiente”	74
Caracterización del sector de desarrollo de videojuegos en Argentina	75
3.2 Accesibilidad desde la perspectiva de la industria	78
Obstáculos en la implementación de accesibilidad en la industria independiente	80
Ejemplos de accesibilidad en juegos independientes	82
Skye Tales	83
ScourgeBringer	85
Conclusiones	91
Iniciativas de concientización sobre accesibilidad	91
Jams de accesibilidad	92
Entidades y Organizaciones	93
Premios a la accesibilidad	94
Charlas y talleres sobre accesibilidad	94
¿Qué dicen los desarrolladores independientes sobre la accesibilidad?	95
Sobre los desarrolladores	95
Sobre la forma de trabajo	96
Sobre la Accesibilidad en Videojuegos	97
Sobre los juegos independientes y la Accesibilidad	99
Sobre las pruebas y los jugadores con discapacidad	100
Sobre las guías de accesibilidad	101
Conclusiones	102
Capítulo 4: Diseño de accesibilidad en videojuegos	104
4.1 La importancia del diseño de la accesibilidad	104
¿De qué forma podemos diseñar accesibilidad?	105
Efectos de la implementación de la accesibilidad en el desarrollo de proyectos de videojuegos	107
¿Cuándo se debería pensar en la accesibilidad?	108
4.2 Guías de accesibilidad en videojuegos	110
WGAC	110

Game Accessibility Guidelines	110
APX Design Patterns	111
XBox Accessibility Guidelines	111
4.3 Catálogo de buenas prácticas en el diseño de accesibilidad en videojuegos	111
Motivaciones	111
Estructura del catálogo	112
Catálogo principal	114
Subtitular los diálogos críticos	114
Implementar subtítulos cerrados (“closed caption”)	117
Facilitar la identificación del hablante	121
Diseñar correctamente los subtítulos	126
Personalización de los subtítulos	136
Proveer controles de volumen exclusivos para cada fuente de sonido	142
Permitir ajustar el tipo de canal de salida (mono/stereo)	146
Evitar la información monocanal	151
Catálogo secundario	156
Activar subtítulos automáticamente	156
No mostrar demasiado texto en los subtítulos	157
Evitar información monocanal: visuales	157
Compatibilidad con lectores de pantalla	157
Permitir la configuración de los controles	157
Controles amigables	158
Usar el mismo control para el juego y la UI	158
Permitir la utilización del modo ventana	159
Asistencias in-game	159
Permitir pausar el juego	159
Incluir tutoriales interactivos	159
Permitir ocultar elementos de violencia o sangre	159
Ofrecer múltiples modos de dificultad	160
Capítulo 5: Accesibilidad en Godot Engine	161
5.1 Sobre Godot Engine	161
5.2 Sobre la accesibilidad en Godot Engine	165
Plugin con soporte para lectores de pantalla	166
¿Qué tan accesible es Godot Engine?	167
Futuras mejoras de accesibilidad de Godot	171
5.3 Soporte de subtítulos en Godot	172
Motivaciones	172
Requerimientos	172
Sobre el formato SRT	174
¿Por qué un plugin?	175
Desarrollo del plugin “Godot Subtitles Plugin”	178
Nodo SRTImporter	178
Nodo LiveCaption	180
Soporte para BBCode	182

Publicación del plugin	183
5.4 Conclusiones	183
Capítulo 6: Prototipo de juego accesible	185
6.1 Metodología	185
6.2 Sobre el proceso iterativo del desarrollo del prototipo	185
6.3 Sobre el diseño de juego	187
6.4 Sobre la implementación del prototipo	188
Arquitectura	188
Escenas principales	189
Escena World	189
Escenas Player y Enemy	190
Interfaz de usuario	191
Funciones de accesibilidad	193
Ocultar fondo	194
Escalar pantalla	194
Remapeo de controles	195
Ajustes de audio	196
Asistencias dentro del juego	198
Diálogos y subtítulos	199
6.5 Sobre el proceso de testing	200
Participantes	201
Observador	201
Playtesting	202
Toma de datos y feedback	203
6.6 Publicación del prototipo	206
6.7 Conclusiones	206
7. Conclusiones finales	210
7.1 Sobre los objetivos propuestos en la tesina	210
7.2 Sobre la accesibilidad en la industria independiente	210
7.3 Sobre el desarrollo en Godot Engine: el plugin “Godot Subtitles” y la construcción y testeo de un prototipo accesible	211
7.4 Líneas futuras de trabajo	212
Bibliografía	212
Anexo 1: Guía para la conducción de las sesiones de playtesting	224
Roles	224
Observador	224
Participante	224
Selección de participantes	224
Preparación	225
Inicio de la sesión	225
Registro de observaciones	225
Recomendaciones	226

Finalización	227
Anexo 2: Documento de diseño de juego - Mahur: el último Runalún	228
Introducción	228
Concepto	228
Nombre	228
Descripción	228
Género	228
Trasfondo	230
Plataformas de lanzamiento	230
Implementación técnica	230
Historia	230
Capítulos	231
Background	231
Capítulo 1	231
Diálogos	231
1er encuentro: bienvenida (aparece)	232
2do encuentro: antes del primer platforming	232
3er encuentro: en la entrada del templo	232
4to encuentro: antes de las ruinas (aparece)	233
5to encuentro: en las ruinas del templo (aparece)	233
Personajes	233
Mahur	233
Tráucos	233
Rumi	234
Diseño de niveles y ambiente	236
Ambiente	236
Secciones	236
Flujo de Pantallas	237
Menú Principal	237
Menú Opciones	238
Menú de Pausa	239
Gameplay	239
Platforming	239
Interacciones	239
Enemigos	240
Piedras luna	240
Hogueras	240
Puertas pilares	240
Arte	240
Capítulo 1: Entrada al bosque	241
Capítulo 2: Bosque Largo	242
Capítulo 3: Las cuevas	243
Capítulo 3: Calabozos	244
Sonido y música	244

Controles	244
Accesibilidad	245
1. Posibilidad de cambiar el modo de juego entre pantalla completa y modo ventana	
245	
2. Posibilidad de ocultar el fondo de juego	246
3. Escalado de elementos de interfaz de usuario	246
4. Remapeo de controles	247
5. Ajustar el volumen de la música y efectos de sonido	248
6. Modo de audio mono-canal	248
7. Subtítulos descriptivos	248
8. Soporte Text-To-Speech	249
9. Asistencia: tutorial de controles	249
10. Asistencia: mostrar lista de objetivos	250

Capítulo 1: Introducción

1.1 Motivación: la problemática de la accesibilidad en la industria independiente de videojuegos

Si bien la accesibilidad dentro del campo de la construcción de productos de software no es algo nuevo, es verdad que para la industria de los videojuegos ha sido uno de los aspectos más relegados, o incluso olvidados por mucho tiempo. El tratamiento de la accesibilidad en los videojuegos fue evolucionando impulsado por políticas dentro los estudios de desarrolladores, por cuestiones legales o simplemente por razones personales de los desarrolladores (Westin, T. et al, 2011). Ciertamente es que la discapacidad está muy presente y afecta de muchas maneras a los jugadores de videojuegos y también a los desarrolladores en la forma de construir sus productos, políticas y tomas de decisiones (Yuan, B. et al, 2011) (John R. Porter et al, 2013).

Recientemente, han surgido nuevas propuestas de inclusión de aspectos de accesibilidad dentro del desarrollo de videojuegos. La importancia de la persona detrás del “jugador” fue cobrando relevancia dentro de la industria de los videojuegos, ejemplo de ello es el evento anual de premiaciones a lo mejor de los videojuegos “The Game Awards”, que en su edición 2020 por primera vez incluyó una nueva categoría: “Innovación en accesibilidad” (The Game Awards, 2020). Esto pone en evidencia que la industria en general está ajustando prioridades y está prestando más cuidado a la implementación de la accesibilidad dentro de sus productos, no solo para cumplir estándares, sino que forma parte, incluso, desde las primeras etapas del desarrollo (Josh Straub, 2020).

Dentro de los grandes estudios de desarrollo de videojuegos es sencillo invertir recursos, tiempo y dinero en accesibilidad. Los “juegos AAA” por lo general cuentan con equipos grandes de desarrollo, marketing y diseño, donde pueden tomarse el tiempo necesario y realizar juegos con una escala muy grande (Mercado O, 2020). El problema de la accesibilidad se evidencia cuando miramos el otro costado de la industria: el de los juegos independientes. A diferencia de los grandes estudios de producción de videojuegos, la industria de juegos independientes se caracteriza por trabajar con recursos reducidos, contando con tiempos limitados para el desarrollo, equipos acotados de desarrolladores, y un presupuesto pequeño (What is an Indie Game and Why Should You Care?, IndieGameMag, 2020). Como resultado, en general, en los desarrollos independientes se pasa por alto la accesibilidad obteniendo productos que no son accesibles y que por lo tanto

no son inclusivos, dejando de lado un grupo importante de personas, jóvenes y adultos sin posibilidades de participar (Matthew Forde, 2019).

La motivación principal de este trabajo de tesina de grado es dar visibilidad a la problemática de la falta de accesibilidad en el ámbito independiente de la industria de los videojuegos, y ofrecer un conjunto de herramientas teóricas y prácticas que apunten a aliviar la dificultad de los desarrolladores independientes a la hora de hacer sus juegos accesibles. Personalmente, aunque no me considero una persona con discapacidad, es cierto que estoy casi ciego del ojo derecho, y tengo sordera del 100 por ciento en el oído izquierdo, con lo cual me resulta sencillo notar malos diseños de accesibilidad, y es con bastante frecuencia que encuentro este tipo de problemas cuando juego. Creo que es necesario cerrar esta brecha y hacer sencilla la inclusión de opciones de accesibilidad en cualquier tipo de juegos, sin importar el presupuesto, tiempo de desarrollo o tamaño del equipo.

Godot Engine es un motor de desarrollo de videojuegos argentino que ha crecido tremendamente desde que se volvió de código fuente abierto en el año 2014, contando actualmente con una comunidad global sumamente amplia, con presencia en más de 20 países. El trabajo técnico y las mejoras que se fueron incluyendo con el tiempo lo convirtieron en uno de los motores más utilizados para la creación de videojuegos, ubicándolo a la par de gigantes como Unity o Unreal Engine. Si bien cuenta con juegos publicados que han sido exitosos y conocidos, una gran parte de la comunidad que utiliza Godot Engine pertenece al sector del desarrollo independiente. Si miramos los resultados de la encuesta de la comunidad de Godot del año 2020, notamos ciertas características comunes entre las y los usuarios de este motor: trabajan solos, o con un equipo de desarrollo de pocas personas; utilizan el motor como un hobby, o dentro de un estudio independiente, y además en su mayoría no sacan un provecho económico de los productos creados (Godot Engine Community poll results, 2020).

Observando el crecimiento del motor, podemos concluir que el foco del desarrollo está puesto en mejorar sus prestaciones en cuanto a capacidad gráfica, facilidad de desarrollo, y performance. Esto hace que cuestiones que otros motores ya tienen resueltas, como la accesibilidad, queden relegadas constantemente frente a otras características consideradas más importantes. Tecnologías como TTS (text-to-speech), input remapping, subtítulos, closed-caption, entre otras, no están incluidas oficialmente dentro del motor, siendo la comunidad la que provee estas funcionalidades en forma de plugins extra oficiales (Godot Engine: Accessibility Proposals). Todos estos factores hacen de Godot Engine un candidato ideal para llevar adelante este trabajo: una comunidad vinculada a la industria independiente de videojuegos, que por lo general es propensa a pasar por alto la accesibilidad, sumado a la escasa disponibilidad de herramientas vuelve a este motor una

oportunidad perfecta para contribuir a la mejora de la industria en el ámbito de la accesibilidad e inclusión de jugadores.

1.2 Objetivos

A continuación se detallan los objetivos planteados para este trabajo.

Objetivo general

Contribuir en la creación de videojuegos inclusivos en el sector de la industria independiente, y específicamente colaborar con la comunidad de “Godot Engine”, haciendo foco en la discapacidad auditiva.

Objetivos específicos

- Investigar sobre el estado del arte de la accesibilidad en la industria de desarrollo de videojuegos, con especial atención en el sector independiente, mediante el planteo de problemáticas inherentes a la accesibilidad en el desarrollo de proyectos de software en estas industrias.
- Desarrollar una estrategia metodológica orientada al diseño de la accesibilidad, y una herramienta orientada al desarrollo de videojuegos en Godot Engine, que permitan mitigar y/o solucionar problemas de accesibilidad, atendiendo las necesidades de las y los usuarios con discapacidad auditiva.

1.3 Estructura del documento

El documento actual se organiza de la siguiente manera:

- Capítulo 1: Introducción. Se describen los temas que se trabajarán en la tesina indicando la motivación y los objetivos a trabajar.
- Capítulo 2: Accesibilidad. Se analizan los conceptos de accesibilidad digital y accesibilidad en videojuegos, y se examina la diversidad de jugadores y su manera de interactuar con el medio digital.
- Capítulo 3: Industria Independiente y Accesibilidad. Se describen las características que identifican a un estudio independiente de producción de videojuegos, y las dificultades a las que puede enfrentarse al realizar juegos accesibles.
- Capítulo 4: Diseño de Accesibilidad en Videojuegos. Se realiza un estudio del diseño de la accesibilidad y cómo afecta al desarrollo de un videojuego, se propone un catálogo de buenas prácticas para la implementación de estrategias de accesibilidad en videojuegos.

- Capítulo 5: Accesibilidad en Godot Engine. Se analiza el motor Godot Engine desde el punto de vista de la accesibilidad, y se desarrolla un plugin para brindar soporte de subtítulos a dicho motor.
- Capítulo 6: Prototipo de juego accesible. Se realiza el desarrollo de un prototipo de videojuego con diseño accesible, realizando pruebas iterativas con diversidad de jugadores.
- Capítulo 7: Conclusiones finales. Se describen las conclusiones finales para este trabajo de tesina de grado, y se proponen líneas futuras de trabajo.
- Anexo 1: Guía para la conducción de las sesiones de playtesting. Documento que describe la metodología utilizada para la realización de las pruebas con jugadores.
- Anexo 2: Documento de diseño de juego - Mahur: el último Runalún. Documento que describe el diseño del juego desarrollado para este trabajo.

Capítulo 2: Accesibilidad

2.1 El concepto de accesibilidad. Sus orígenes y enfoques.

Según la Organización Mundial de la Salud (OMS) alrededor de 15% de la población mundial (más de mil millones de personas) experimentan algún tipo de discapacidad (OMS, 2022), y hasta 190 millones de personas con más de 15 años de edad tienen dificultades importantes para funcionar, lo que a menudo exige servicios de atención de salud.

Además de enfrentar impedimentos físicos, cognitivos o sensoriales, el mismo informe señala que las personas con discapacidad enfrentan barreras para la inclusión en diversos aspectos de la vida: suelen tener menos oportunidades socioeconómicas, menor acceso a la educación y tasas de pobreza más altas. En ocasiones, el estigma y la discriminación son el principal freno para la participación plena e igualitaria dentro de la sociedad.

La accesibilidad es la respuesta clave para revertir esta situación. La definición puede variar, pero en un nivel general se trata de la posibilidad que tiene una persona, con o sin problemas de movilidad o percepción sensorial, de entender un espacio, integrarse en él e interactuar con sus contenidos (Coterón et al, 2018).

Según el libro “Manual de accesibilidad universal” (Simonetti et al, 2010), refiriéndose a accesibilidad en arquitectura, una buena accesibilidad es aquella que pasa desapercibida a los usuarios. Esta “accesibilidad desapercibida” implica algo más que ofrecer una alternativa a las escaleras como acceso: busca un diseño equivalente para todos, cómodo, estético y seguro. Es importante resaltar este último punto: la accesibilidad no trata únicamente de resolver las dificultades de personas con discapacidad, sino que trata de lograr la inclusión en el diseño para todas las personas ya sea que posean algún grado de discapacidad o no. De acuerdo a este mismo manual, la accesibilidad es “el conjunto de características que debe disponer un entorno urbano, edificación, medio de comunicación, producto o servicio para ser utilizado en condiciones de comodidad, seguridad, igualdad y autonomía por todas las personas, incluso por aquellas con capacidades motrices o sensoriales diferentes” (Simonetti et al, 2010, p. 12). Entonces podemos utilizar el concepto de accesibilidad para referirnos al grado o nivel en el que cualquier persona, más allá de su condición física o cognitivas, puede utilizar una cosa, disfrutar de un servicio o hacer uso de una infraestructura.

Los enfoques de integración e inclusión y su relación con la accesibilidad

El concepto de accesibilidad se encuentra muy relacionado con los conceptos de “integración” e “inclusión”, aunque estos tengan acepciones muy diferentes. La inclusión y la

integración representan dos posiciones filosóficas totalmente diferentes, aún cuando a primera vista tienen objetivos y funciones aparentemente iguales. En las propuestas para realizar un entorno accesible basados en un enfoque integrador, la guía temática sobre “Accesibilidad e inclusión de personas con discapacidad” (Biblioteca CEDIB-INADI, 2018) establece que “las transformaciones son superficiales y se hacen las mínimas adecuaciones, destinadas únicamente a las personas con discapacidad”. Por otro lado, la misma guía señala que en los diseños de entornos accesibles pensados desde el enfoque de la inclusión, “se transforman los sistemas y el entorno completo para que sean de calidad para todas las personas, partiendo de la base de la diversidad como característica de toda sociedad” (Biblioteca CEDIB-INADI, 2018, p. 8). En la “integración”, una comunidad o una institución se “acomoda” para recibir al otro, al distinto. En la “inclusión”, todos gozan de los mismos derechos, obligaciones, oportunidades y todo está diseñado para contemplar las fortalezas y debilidades de cada uno de sus miembros. Nadie se adapta o se acomoda a nada ni a nadie, todos son partícipes en grados semejantes (Casal et al, 2012). En la figura 2.1 se describe una interpretación gráfica de estos conceptos.

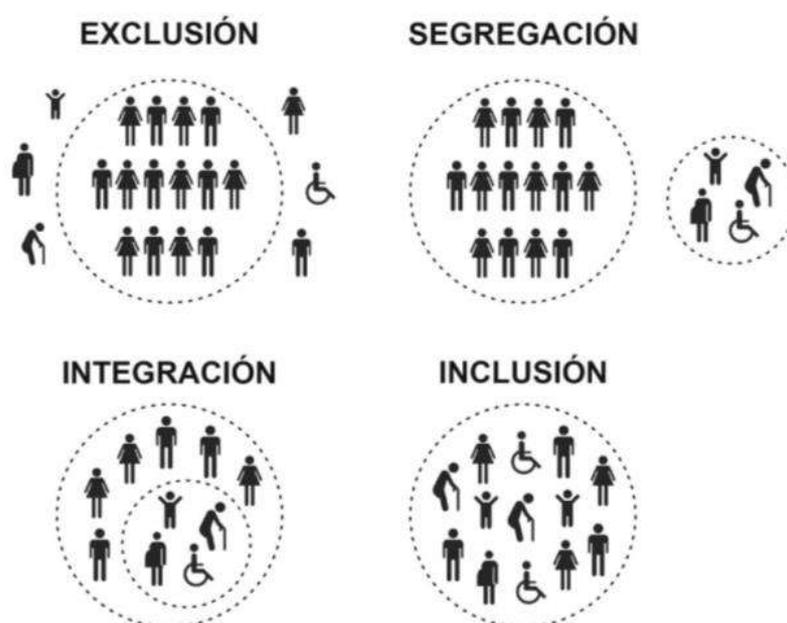


Figura 2.1 Relación entre los conceptos de exclusión, segregación, integración e inclusión. Gráfico extraído de Claves en la formación del discurso de Inclusividad en la Arquitectura Chilena: 1960 - 2016. Universidad de Chile (2019)

Existen diversas ayudas técnicas para impulsar la accesibilidad y equiparar las posibilidades de todas las personas. Esto supone que un espacio, servicio o programa informático que presenta buenas condiciones de accesibilidad puede recibir y ser utilizado por toda clase de personas sin que exista un perjuicio o una complicación para nadie. Las rampas para

discapacitados, las sillas de ruedas, el alfabeto Braille y las señales auditivas son algunas de las ayudas técnicas más comunes que solemos conocer.

Surgimiento y evolución del concepto de accesibilidad

La accesibilidad tal como la conocemos actualmente es un concepto que apareció a finales del siglo XX y que surgió tras las presiones de diversos movimientos sociales relacionados con la discapacidad, pero que anteriormente estuvo presente a lo largo de la historia bajo diversas formas.

En la segunda mitad del siglo XX es cuando surge la necesidad de adaptar el medio físico a las personas, por medio de ayudas técnicas para que las personas puedan integrarse con el resto de la sociedad sin problemas. Con esto emerge el primer entendimiento de accesibilidad como “eliminación de barreras”, definido como el procedimiento por el que se intentan suprimir todo aquello que bloquee, frene, limite o aleje de los objetivos planteados por la persona previamente, o restrinja sus oportunidades y/o capacidades de expresión o acción (Sampedro-Palacios, 2019). Podemos decir que esta definición quedó grande en su momento, ya que en la realidad se tradujo sólo en la construcción de rampas y ascensores, acceso para sillas de ruedas y elementos del sistema braille.

En el año 1974 se celebró en Nueva York la “Reunión del Grupo de Expertos sobre el Diseño Libre de Barreras”, donde se establecieron los primeros antecedentes sobre la necesidad de la eliminación de barreras físicas que dificulten a las personas con discapacidad participar plenamente de la sociedad en igualdad de condiciones (Arjona, 2014). Adicionalmente se estableció como requisito en la formación de arquitectos, ingenieros, urbanistas y paisajistas el estudio de la inclusión. Es en esta época cuando empiezan a circular los primeros documentos sobre la formación de los profesionales para la eliminación de las barreras físicas.

Con el correr de los años se comenzó a hablar del “Diseño para Todos” o “Diseño Universal” nacido principalmente de los movimientos sociales y políticos que se venían cultivando en las décadas de los años 50 y 60: con el movimiento arquitectónico del “funcionalismo orgánico escandinavo” de los años 50 y las actualizaciones en el campo de la ergonomía de los años 60. A finales de la década de los 70 aparece dentro de la política social sueca el concepto de “Una sociedad para Todos” referido principalmente a la accesibilidad (Arjona, 2014). Esta filosofía que incorpora el nuevo modelo social de intervención con personas con discapacidad, en el que los derechos humanos están presentes, dando pie a una nueva forma de plantear el problema: ya no se trata de derribar barreras, sino de diseñar sin ellas, dando comienzo a difundir el concepto de accesibilidad en el ámbito de la edificación, el urbanismo, el transporte y posteriormente la comunicación, como quedó plasmado en el “Programa de Acción Mundial para las Personas con Discapacidad” de las Naciones Unidas

en 1982. Se instala el concepto de “Accesibilidad Física Integral” como condicionante para la inclusión y normalización de las personas con movilidad y comunicación reducida y que sería el germen que ha ido evolucionando hasta el concepto de “Accesibilidad Universal” como concepto globalizador e integrador. Paralelamente en Estados Unidos se incorpora el “Universal Design” con la “Americans with Disability Act” y en el Reino Unido el “Inclusive Designs”.

Esta idea siguió evolucionando y en 1989, el arquitecto Ronald L. Mace hace una valoración crítica de los conceptos de “Accesibilidad Física” y propone como etapa de evolución la del “Diseño Universal”, que define como la *“creación de productos y entornos diseñados de modo que sean utilizables por todas las personas en la mayor medida posible, sin necesidad de adaptaciones o diseños especializados”*, concepto éste con rápida difusión en ámbitos académicos y de investigación (Gonzalo Arjona Jiménez, 2012).

Mace fue el arquitecto que más influyó en el pensamiento internacional respecto al diseño, fundó el “Center for Universal Design” de la Universidad de Carolina del Norte (The Universal Design File, Molly Follette, 1998). En los años noventa creció el interés por el concepto de diseño universal, sobre todo en el área del diseño industrial, por lo que Ron Mace encarga a un grupo de diseñadores y abogados que crearan una serie de principios que sintetizan esta filosofía del diseño. A estos principios se les conoce como los “siete principios del diseño universal”, que ofrecen a los diseñadores una guía para incluir mejor las características que resuelven las necesidades de tantos usuarios como sea posible (Principios del Diseño Universal, COPIDIS, web buenosaires.gob.ar). De este modo el diseño universal significa que al diseñarse un producto se tenga en cuenta los principios que se describen en la tabla 1.

Principio	Descripción
1. Igualdad de uso	El diseño es útil y alcanzable a personas con diversas capacidades.
2. Uso flexible	El diseño se acomoda a un amplio rango de preferencias y capacidades individuales.
3. Uso simple y funcional	El diseño es fácil de entender independientemente de la experiencia, conocimientos, habilidades o nivel de concentración del usuario. El diseño es simple en instrucciones e intuitivo en el uso.
4. Información comprensible	El diseño debe ser capaz de comunicar al usuario de manera eficaz, independientemente de las condiciones ambientales o las capacidades sensoriales

	del mismo.
5. Tolerancia al error	El diseño debe minimizar los riesgos y las consecuencias adversas de acciones involuntarias o accidentales.
6. Mínimo esfuerzo físico	El diseño debe poder ser usado eficazmente y con el mínimo esfuerzo posible.
7. Adecuado tamaño de aproximación y uso	Los tamaños y espacios deben ser apropiados para el alcance, manipulación y, independientemente de su tamaño, posición o movilidad.

Figura 2.2 Principios del Diseño Universal (Principios del Diseño Universal, COPIDIS, web buenosaires.gov.ar)

El objetivo que pretende el diseño universal es simplificar la vida de todas las personas, haciendo que los productos, las comunicaciones y el entorno construido sean más utilizables por la mayor cantidad posible de usuarios con un costo nulo o mínimo (The Universal Design File, Molly Follette, 1998). En 1993 se constituye el European Institute for Design and Disability (EIDD) con el objetivo de “mejorar la calidad de vida mediante el Diseño para Todos”. En 1994, con ocasión de la sesión pública del “Seminario Iberoamericano de Accesibilidad al Medio Físico”, en Río de Janeiro, se planteó la superioridad del “Diseño Universal” sobre la “Accesibilidad Universal” llegándose a exigir entre otras cosas la eliminación del uso del símbolo internacional de accesibilidad por considerarlo discriminatorio al establecer recorridos, espacios e instalaciones específicos para personas con discapacidad y por consiguiente la exclusión de los otros, utilizados por la mayoría de las personas. En 2004, en la junta anual celebrada en Estocolmo, se aprobó la “Declaración de Estocolmo para el Diseño para Todos” en la que se afirma que el “Diseño para Todos es un planteamiento holístico e innovador, que se constituye un reto ético y creativo para todos los diseñadores, empresarios, administradores y dirigentes políticos”.

2.2 Accesibilidad digital: ¿Qué es? ¿Qué implica?

Los conceptos de accesibilidad y la inclusión siguieron evolucionando dentro de los ámbitos donde se originaron inicialmente: la arquitectura, el urbanismo, el transporte y los servicios públicos en general. Al mismo tiempo la tecnología digital avanzó a pasos agigantados a partir de fines del siglo XX, y el cambio tecnológico producido en esta época se tradujo, entre otras cosas, en la aparición de dispositivos digitales, herramientas y formas de comunicación mediadas por las tecnologías digitales (Gonzalo Arjona Jiménez, 2012). El surgimiento de Internet, a finales del siglo XX, la popularización de los teléfonos celulares, las nuevas formas de comunicación a través de computadoras y otros medios digitales,

cobró suma relevancia e impactó en la vida cotidiana de una gran parte de la población a nivel personal, incluyendo aquellas con algún grado de discapacidad. Esto permitió que la accesibilidad cruzara la frontera del mundo físico y apareciera también en el mundo digital. Hoy en día entendemos que la accesibilidad es un concepto que debe estar presente en los contenidos digitales que frecuentamos, sin embargo esto no se concibió así desde el principio de la irrupción digital, sino que fue un camino gradual en donde la legislación y las organizaciones que luchan por la vigencia de los derechos humanos cumplieron un papel fundamental.

Para entender el surgimiento del concepto de accesibilidad digital tenemos que remontarnos al año 1973, cuando en Estados Unidos se aprobó una ley que tuvo un efecto muy grande en millones de personas con discapacidad, la “Ley de Rehabilitación”. Antes de la aprobación de esta ley, las agencias federales no tenían ninguna obligación de prestar servicios especiales a personas con discapacidad (Rehabilitation Act of 1973, U.S. Access Board). Esto significa que no existía ninguna regulación y cada agencia administraba la implementación de soluciones de accesibilidad de forma particular o ninguna. La intención de la ley fue regular a las agencias federales, y a todos los organismos y agencias relacionadas, de forma que se aseguraba que no existía discriminación basada en discapacidades ya sea en la designación de empleos, la asistencia financiera, y lo más importante, el acceso a tecnologías digitales. Esto significa que cualquier tipo de tecnología creada por cualquier agencia gubernamental y sus empresas relacionadas debían ser accesibles para personas con discapacidad. En el momento de la sanción de la ley no existía Internet como la conocemos hoy en día, y la expansión de los sitios web todavía no había ocurrido. La “Ley para Estadounidenses con Discapacidades” de 1990, también conocida como “Americans with Disabilities Act” o ADA, es una ley que prohíbe la discriminación a individuos con discapacidades, en todas las áreas de la vida pública, incluyendo trabajos, escuelas, transporte, y todos los espacios públicos y privados que están abiertos al público general. El propósito de esta ley es asegurar que las personas con discapacidad tengan las mismas oportunidades (ADA National Network, “What is the Americans with Disabilities Act”). Tal vez la sección más interesante es el Artículo 3, el cual prohíbe la discriminación en base a la discapacidad, en todas las actividades de los lugares considerados “de acceso público”. Es importante para la historia de la accesibilidad digital dado que la definición de lugares de acceso público hasta ese momento se referían exclusivamente a espacios físicos, sin embargo esta definición cambió, al incluir al espacio virtual.

La expansión masiva de internet, con la creación de miles de sitios web y con la ampliación del acceso a la red desde los hogares, las personas con discapacidad se enfrentaron con un problema grave: los sitios web, en general, no estaban diseñados de forma que fueran

fáciles de navegar e interactuar por personas con discapacidad. Legalmente no estaba contemplado, dado que el Artículo 3 del ADA no incluía a los contenidos digitales en la definición de lugares de acceso público. En 1996 esto cambió, el Departamento de Justicia de Estados Unidos (DOJ) dictaminó que los sitios web también debían ser considerados espacios de acceso público, y que por lo tanto eran alcanzados por el ADA. Este dictamen no fue una solución definitiva para el problema de la inclusión en el espacio digital, dado que a pesar estar incluidos los sitios webs dentro del ADA, no existía una definición clara sobre otros tipos de contenidos digitales, ni una especificación sobre la forma en las que se debía proveer accesibilidad. El año 1998 es un año fundamental para la accesibilidad digital, dado que se amplió la Ley de Rehabilitación de 1973, la cual no incluía ninguna mención sobre el mundo digital, y requirió a las agencias federales y a las organizaciones vinculadas, que todo el contenido electrónico y digital generado fuera accesible para personas con discapacidad. Esta modificación significó una protección dentro del marco legal brindando la posibilidad de presentar una denuncia legal ante casos donde una agencia gubernamental no provea la accesibilidad necesaria a su contenido digital. El avance de la tecnología de telefonía celular durante la primera década de los años 2000, es otro hito a señalar en relación a la accesibilidad digital. La compañía Apple lanzó en el año 2005 el producto "VoiceOver", un lector de pantalla que permitía a los usuarios con discapacidades visuales utilizar el sistema operativo Mac OS de una forma mucho más natural y sencilla que en otros sistemas operativos como Windows en ese momento. Esto representó un hito en la historia de la accesibilidad digital, dado que con la introducción de VoiceOver, Apple se convirtió en el primer creador de sistemas operativos con un lector de pantallas incorporado, completamente funcional, sin instalaciones extras. En años posteriores Apple lanzó VoiceOver en otros productos como el iPhone, iPad y un tiempo después el Apple Watch.

En 2010, también en Estados Unidos, se legisló una ley conocida como CVAA por sus siglas en inglés de "Communications and Video Accessibility Act". Aunque esta ley está centrada principalmente en asegurar la accesibilidad de todas las telecomunicaciones "modernas" para las personas con discapacidad (Pike, 2010), es relevante mencionarla en esta tesina dado que tiene una gran implicancia en las comunicaciones que se realizan dentro de los videojuegos, como pueden ser los sistemas de chat accedidos por los jugadores dentro de los juegos multijugador. Esta ley requiere que *"todas las funcionalidades que implican comunicación, como por ejemplo los chats dentro del juego, y las interfaces de usuario utilizadas para navegar y operar estos elementos deben ser accesibles para las personas"* con discapacidades visuales, auditivas, motoras, cognitivas o del habla. Esto significa que todos los videojuegos que provean un sistema de comunicación dentro del juego, deben cumplir con esta reglamentación de accesibilidad. Es

importante aclarar que esta reglamentación se debe cumplir solo para los componentes de comunicación, y no alcanza al resto del juego. De no cumplir con estos requerimientos, los usuarios quedan habilitados de presentar una denuncia ante la Comisión Federal de Comunicaciones, a partir de la cual esta agencia trabajará con la compañía afectada para solucionar el problema.

Pautas de Accesibilidad al contenido web: WCAG

Tim Berners-Lee, considerado el padre de la internet, fundó en 1994 el consorcio WWW (World Wide Web Consortium, W3C) con el objetivo de desarrollar protocolos y directrices que garanticen el crecimiento a largo plazo de la Web. El consorcio integrado por expertos investigadores lanzó en 1999 la Web Content Accessibility Guidelines (WCAG) versión 1.0.

La WCAG 1.0 es un compendio de pautas y checkpoints necesarios para hacer una web “accesible”. Cada una de estas pautas cubren un elemento específico de la accesibilidad web, y está destinada a todos los desarrolladores, autores y diseñadores de sitios web. Con el objetivo de ayudarlos a crear contenido web accesible, y al mismo tiempo ayudar a los usuarios con algún grado de discapacidad al momento de utilizar la Web (W3C, “Web Content Accessibility Guidelines 1.0”). La creación de este consorcio y el impulso dado a la construcción de sitios web accesibles mediante el desarrollo de las guías de accesibilidad representaron un hito en el camino hacia una web accesible. En el año 2008 se lanzaron las “Pautas de Accesibilidad de Contenido Web 2.0”, que continúan la premisa de definir maneras correctas de crear contenido web accesible para personas con discapacidad. A pesar que estas pautas abarcan un amplio espectro de cuestiones relacionadas a la discapacidad, no cubren los diferentes tipos, grados y combinaciones de discapacidades. Con el fin de contemplar necesidades tan variadas, las pautas incluyen una serie de niveles: principios fundamentales, pautas generales, criterios de éxito verificables y una amplia colección de técnicas suficientes, técnicas aconsejadas y fallos comunes documentados con ejemplos, vínculos a recursos y código (W3C, “How WCAG 2.0 Differs from WCAG 1.0”).

En 2018 se difunde la versión WCAG 2.1 que se apoya en la versión anterior y extiende sus pautas para alcanzar un mayor rango de personas con discapacidades, incluyendo la pérdida parcial y total de la visión, pérdida parcial y total de la audición, trastornos del movimiento, dificultades del habla, fotosensibilidad, dificultades en el aprendizaje y limitaciones cognitivas (Kirkpatrick & Cooper, 2018). Para los usuarios de dispositivos móviles, provee una orientación actualizada incluyendo soporte para las interacciones basadas en el tacto, introduciendo gestos más complejos, y soporte para evitar activaciones de interfaces no intencionadas. Para los usuarios con baja visión, se extienden los requerimientos de contraste a los gráficos, y se introduce un nuevo requerimiento para el

texto y la customización del layout para soportar una mejor percepción visual del contenido y los controles (W3C, “What’s New in WCAG 2.1”). Para los usuarios con discapacidades cognitivas, del lenguaje y del aprendizaje, las mejoras incluyen un requerimiento para proveer información acerca del propósito específico de los controles de entrada, así como también requerimientos adicionales para soportar timeouts debido a inactividad. Esto puede ayudar a los usuarios a entender mejor el contenido web y a interactuar correctamente con él.

Las Pautas de Accesibilidad WCAG 3.0 serán la próxima versión de las pautas de accesibilidad del contenido en la web del consorcio W3C. El estado actual de esta versión es la de “borrador de trabajo”, y se espera que esté finalizado para finales del 2022. Las diferencias con la versión anterior incluyen cambios en la semántica y forma de nombrar algunas propiedades para incluir a otros contenidos digitales y no solo a la web. Además se mejoran algunas descripciones utilizando lenguaje coloquial, más sencillo de entender por personas que no son expertos en tecnología.

Las WCAG y su aplicación a la construcción de videojuegos accesibles

Si bien las Pautas de Accesibilidad WCAG no fueron diseñadas específicamente para su aplicación en videojuegos, pueden servir como base para tomar algunas decisiones en la implementación de accesibilidad en videojuegos (Wilson & Crabb, 2018). Las WCAG ofrecen una serie de lineamientos que pueden ponerse en práctica para la implementación de ciertos aspectos e interacciones con las interfaces de usuario en juegos móviles y en juegos web (basados en el uso de Canvas y WebGL), como por ejemplo en el tamaño y color de los elementos visuales, y de los gestos utilizados para interactuar con estos elementos (Barlet et al, 2019).

2.3 Accesibilidad en Argentina. Marcos normativos

A continuación se describirán brevemente la evolución de los marcos normativos, en nuestro país, sobre discapacidad y accesibilidad, su evolución y cambios de enfoque.

El enfoque asistencialista de la ley de discapacidad

En Argentina, la Ley 22.431, sancionada en 1981 como ley marco en discapacidad, instituyó el denominado “sistema de protección integral de las personas discapacitadas”, una normativa que da cuenta de una concepción de la discapacidad de anclaje eminentemente médico, proteccionista, asistencialista, que entiende a la persona con discapacidad sólo como sujeto de cuidado y no como sujeto de derecho; que anula el reconocimiento de la persona como actor social y la limita a ser receptora de acciones de contención

diferenciales y segregacionistas (Sitio argentina.gob.ar, “Agencia Nacional de Discapacidad”).

En el año 1994 se sanciona la Ley Nacional 24.314 de “Accesibilidad de personas con movilidad reducida”, extendiendo la definición de accesibilidad del espacio y barreras físicas dentro de los ámbitos urbanos, arquitectónicos y del transporte (Sitio web argentina.gob.ar, “Ley 24314”).

En el año 2003 la Legislatura de la Ciudad Autónoma de la Ciudad de Buenos Aires sanciona la Ley 962 de “Accesibilidad para todos”, que introduce modificaciones al código de edificación de la Ciudad de Buenos Aires (Comisión para la Plena Participación e Inclusión de las Personas con Discapacidad (COPIDIS), “LEY 962. Accesibilidad física para todos”).

El camino hacia un enfoque de derecho humano en la ley de discapacidad

Podemos señalar como hito trascendente, en nuestro país, en relación a los derechos de las personas con discapacidad la sanción, en el año 2008, de la Ley Nacional 26.378 (Sitio web argentina.gob.ar, “Ley 26378”) que aprueba la Convención sobre los Derechos de las Personas con Discapacidad de la ONU (CDPD) y su protocolo facultativo, que en el Artículo 9 se refiere a la accesibilidad de este modo: “[...] A fin de que las personas con discapacidad puedan vivir en forma independiente y participar plenamente en todos los aspectos de la vida, los Estados Partes adoptarán medidas pertinentes para asegurar el acceso de las personas con discapacidad, en igualdad de condiciones con las demás, al entorno físico, el transporte, la información y las comunicaciones, incluidos los sistemas y las tecnologías de la información y las comunicaciones, y a otros servicios e instalaciones abiertos al público o de uso público, tanto en zonas urbanas como rurales [...]”.

Según el sitio web oficial del Estado Argentino de la Agencia Nacional de Discapacidad, la Nueva Ley de Discapacidad será una ley de Derechos Humanos, con perspectiva de género, interseccional e intercultural, que se ampare en los lineamientos del Modelo Social que entiende a la persona primero como persona en el reconocimiento de sus derechos humanos, civiles, políticos y de libertades fundamentales (Sitio web argentina.gob.ar, “Sobre la Agencia Nacional de Discapacidad”). Se tratará de una ley que promueve diseños universales de todos los entornos y que entienda a la persona con discapacidad como sujeto activo de la vida en sociedad en todas sus esferas. Esta nueva ley de discapacidad representa un paso fundamental en el camino de la armonización legislativa, cuyo objetivo es cambiar el paradigma con el cual se aborda la discapacidad actualmente. Intenta consolidar un marco legal de reconocimiento de derechos humanos, civiles y políticos.

La ley vigente, sancionada en 1981, considera a la persona con discapacidad desde una mirada rehabilitadora, asistencialista, mucho más cercana al anticuado modelo médico y

lejana del modelo social de la discapacidad, donde la persona toma un rol social activo. La nueva ley busca un cambio hacia el modelo social, donde se identifica a las personas con discapacidad primero como personas y como sujeto con participación activa de la vida sociedad, en el estudio, el trabajo, la salud, la participación política y protección social.

La ley de accesibilidad de las páginas web. La adopción de las WCAG Una ley fundamental en relación a la accesibilidad de páginas web es la Ley 26.653 de “Accesibilidad de la Información Pública en las Páginas Web”, sancionada en 2010, que tiene por objetivo facilitar el acceso a los contenidos de las páginas Web a las personas con discapacidad, buscando evitar todo tipo de discriminación a través de la igualdad real de oportunidades y trato. De esta manera el acceso a la información es reconocido como un elemento esencial para garantizar el derecho a la libertad de expresión y opinión, que incluye, la libertad de recabar, recibir y facilitar información e ideas.

Según lo establecido por la Oficina Nacional de Tecnologías de Información (ONTI) en la “Guía de Accesibilidad para Sitios Web del Sector Público Nacional” (Resolución 69/2011), los sitios web alcanzados por la Ley 26.653 deberán cumplir con las pautas internacionales WCAG 1.0 definidas en 1999 por la W3C. Bajo esta normativa, deberán respetar en los diseños de sus páginas web las normas y requisitos sobre accesibilidad de la información que faciliten el acceso a sus contenidos a todas las personas con algún tipo de discapacidad. En el 2014, la ONTI definió bajo una nueva disposición (2/2014) que la versión de la WCAG 1.0 quedaba reemplazada por la WCAG 2.0.

Una novedad interesante de señalar es la sanción del Decreto 656/2019 que introduce cambios en la Ley 26.653, como por ejemplo, se establece a la ONTI como autoridad de aplicación de la ley. Si bien la ONTI mantiene a su cargo la asistencia y asesoramiento de aquellas personas físicas y jurídicas alcanzadas por la Ley 26.653, ahora también podrá solicitar a las entidades del Poder Ejecutivo Nacional información referente a páginas web de las instituciones u organizaciones de la sociedad civil, con el fin de controlar que cumplan con las disposiciones referidas a la accesibilidad de la información en sus páginas web. De igual manera, la ONTI deberá informar a la Agencia Nacional de Discapacidad (ANDIS) los incumplimientos a la Ley 26.653 para que esta última pueda aplicar sanciones.

2.4 Accesibilidad en videojuegos

Durante las últimas décadas la industria del software de entretenimiento digital, popularmente conocida como la industria de los videojuegos se ha convertido en una industria global multimillonaria, con un consumo equivalente a 184,4 mil millones de dólares a nivel global para el año 2022, según la consultora Newzoo (Wijman, 2022, “The Games Market in 2022: The Year in Numbers”). La competencia constante presente en esta

industria ha generado un estímulo muy grande en cuanto al impulso del desarrollo tecnológico, logrando que los videojuegos sean cada vez más avanzados en su poder gráfico y de procesamiento, creando experiencias mucho más realistas con el correr de los años. Algunos títulos AAA (videojuegos con muchísimo éxito comercial), son comparables a las películas de Hollywood de la industria del cine, prácticamente son películas interactivas en las que el jugador se sumerge en el mundo del juego. El aumento en la calidad de las producciones de esta industria ha sido acompañado por el incremento en los costos de realización (Banco Interamericano de Desarrollo, 2019, “Los videojuegos no son un juego”). Se estima que los costos de desarrollo de producción medios para un juego de este estilo oscilan entre los 60 y 100 millones de dólares (RocketBrush Studio, 2022), siendo necesario realizar una venta tan grande de copias que justifique el nivel de inversión en el desarrollo. Según el artículo publicado por la ESRB (Entertainment Software Rating Board), en el año 2021 más de la mitad de los juegos clasificados fueron incluidos en la categoría “E” (juegos para todos), un número que 2018 llegaba era del 42%. La figura 2.3 describe la distribución de juegos rankeados por la ESRB según su tipo (Juegos para todos, Para mayores de 10 años, Para adolescentes, Para adultos) de un total de 4839 de juegos clasificados.

Categorización 2021 por ESRB

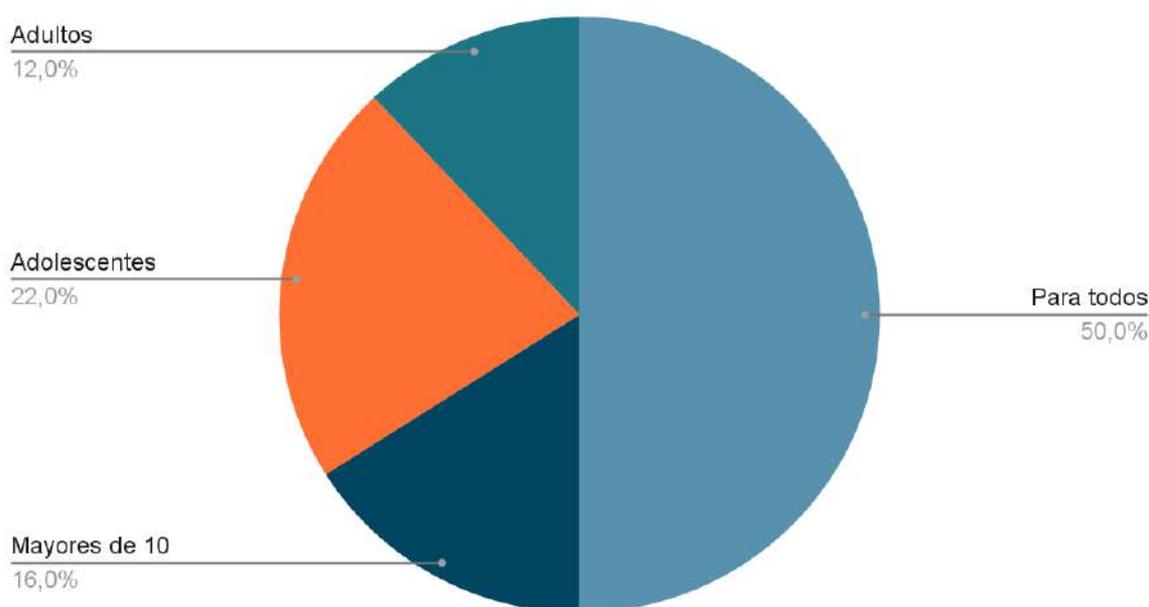


Figura 2.3 Clasificación de la ESRB para juegos según su categoría (ESRB Blog, “Half of All ESRB Ratings Assigned in 2021 Were E for Everyone”, 2022)

Al igual que en la industria del cine, las compañías desarrolladoras de videojuegos trabajan duro para asegurarse que sus productos tengan un alcance global, y que lleguen al mayor número posible de jugadores, una intención que queda bastante clara mirando la

clasificación a la que apuntaron la mayoría de los juegos en los últimos años. Además de la clasificación, uno de los factores más importantes para lograr alcanzar una gran cantidad de jugadores es la localización en varios idiomas. La localización permite que un juego lanzado en Estados Unidos, por ejemplo, pueda llegar a los jugadores de todo el mundo. Muchos juegos obtienen hasta el 50% de sus ganancias de las ventas en el mercado internacional (Chandler, 2006), lo que hace cada vez más conscientes a los creadores de videojuegos de la necesidad de incluir la localización en el proceso de desarrollo del juego, para no tener que trabajar en esto luego de terminado el desarrollo, siendo generalmente un proceso más costoso.

La discapacidad, como se describió en las secciones previas, puede ser considerada como un “desajuste o discordancia” entre las necesidades del individuo y del entorno, producto o servicio que utiliza. En ese sentido, cuando una persona con discapacidad interactúa con dichos productos o servicios, se generan “barreras” que impiden a la persona utilizar el servicio o producto correctamente, o incluso impedir su uso. En el contexto de los videojuegos, la accesibilidad significa romper esas barreras para asegurar que todas las personas tengan una experiencia disfrutable como jugador cuando interactúan con el juego (Baker, 2021).

Tradicionalmente el término “accesibilidad” ha sido asociado con personas con discapacidades físicas, sensoriales o mentales. En el contexto de esta tesina el término “accesibilidad en videojuegos” se utiliza para describir una situación en la que una persona es capaz de seguir jugando a un juego incluso mientras juega bajo “condiciones limitantes”, las cuales pueden incluir discapacidades mentales, sensoriales o físicas permanentes o temporales (Mangiron et al, 2014).

Una de las diferencias clave entre la accesibilidad de los medios físicos y la accesibilidad en los medios digitales es la interacción del usuario con el medio. En el caso particular de los videojuegos, donde los usuarios son llamados “jugadores”, el reto de la accesibilidad es mucho mayor, y toma una importancia superior, dado que la función principal del videojuego es entretener. En los videojuegos, el jugador debe superar diversos obstáculos, cumplir determinados objetivos, llegar a la meta según las reglas del juego para poder ganar, al mismo tiempo que disfrutar del proceso (Barlet & Spohn, 2012). Para las personas con algún tipo de discapacidad un videojuego con accesibilidad pobre o mal implementada podría afectar su progreso durante el proceso de juego, causando frustración y excluyéndolos de una de las formas de entretenimiento más populares en la actualidad (Ellis et al, 2020).

La accesibilidad en videojuegos no solo se trata de crear un producto que pueda ser utilizado por la mayor cantidad de jugadores posibles, va más allá del proceso de evitar y eliminar barreras innecesarias en la interacción de los jugadores con el juego. Para ello, es

importante reconocer y entender el contexto social en el que los videojuegos de hoy en día son diseñados, desarrollados y publicados.

Evolución del concepto de accesibilidad aplicada en videojuegos

Las primeras máquinas de juegos arcade surgieron en la década de 1970, con juegos como Pong (1972), y tuvo su época dorada durante los últimos años de esa época y los primeros años de la década de 1980. Juegos como Space Invaders (1978), Galaxian (1979), Pac-Man (1980), Donkey Kong (1981) eran distribuidos en bares y otros espacios de recreación, mediante máquinas que incluían el microchip con la programación de la lógica interna, una pantalla en donde se mostraba el juego, y un soporte con botones y palancas que se utilizaban para controlar el videojuego.

Con las primeras máquinas arcade y consolas caseras también aparecieron las primeras formas de accesibilidad. En esta época la máquinas y juegos eran casi sinónimos, dado que la idea de la consola moderna en donde se podían jugar múltiples juegos todavía no existía. Por lo general, la máquina era diseñada y fabricada con una plaqueta que permitía ejecutar un único juego. Sin embargo algunos videojuegos de esta época estaban diseñados para incluir múltiples niveles de dificultad, opciones para simplificar los controles, e incluso existían dispositivos que seguían la mirada del jugador y permitían mover al personaje sin utilizar un joystick.

Estos avances eran raramente impulsados por las grandes compañías de videojuegos, por lo que fueron mayormente impulsados por desarrolladores y aficionados de la computación, quienes utilizaron sus conocimientos para ayudar a jugar de una forma más sencilla a sí mismos, o alguien cercano. La gran mayoría de los productos que estaban disponibles comercialmente eran muy caros para el consumidor promedio, o simplemente no eran muy conocidos. Muchos jugadores decidieron que si los desarrolladores no hacían sus productos más accesibles, *“eran los usuarios quienes debían tomar la responsabilidad de hacerlo, una forma de pensar que continuaría hasta el día de hoy”* (Wilds, 2020).

Con la aparición de la computadora personal, y su masificación, surgieron los primeros juegos para computadoras. En su mayoría eran juegos de aventura basados en texto, es decir que el desarrollo de la historia no se veía mediante imágenes o sonido, sino que se leía en palabras en pantalla. La interacción del jugador también era a través de las palabras, ingresando comandos por teclado para realizar acciones dentro del videojuego. Con el correr de los años y la mejora en las capacidades de procesamiento de las computadoras personales y de las placa de video y sonido, aparecieron videojuegos con interfaces visuales, gráficos más realistas, sonido y soporte para otros tipos de controladores además del teclado, como mouse, joysticks y gamepads. La creación de mods (modificaciones de juegos hechas por jugadores), y la facilidad para customizar el uso de teclado y mouse

dieron a la comunidad de jugadores de PC una ventaja importante al momento de hacer a los juegos más fáciles de jugar. Esta fue la plataforma de juego más elegida por personas con diversidad funcional por mucho tiempo. Existe una diferencia entre las barreras que se producen en la utilización del software y del hardware, y por desgracia algunos problemas no pueden ser solucionados completamente por los usuarios, dado que requieren de cambios que solo el desarrollador podría realizar. Un ejemplo de esto puede ser una opción para ajustar la dificultad del juego, esto es algo que requiere desarrollo y soporte por parte del equipo de desarrollo del juego, y no es algo que se pueda ajustar con mods o parches extra-oficiales.

Con la aparición de consolas más modernas, como PlayStation y Xbox, se produjo un salto muy grande en la capacidad técnica con respecto a la generación de consolas anteriores, y el nivel de procesamiento de las mismas permitió la creación de experiencias de juego mucho más inmersivas para los jugadores. Sin embargo los juegos de esta generación no incluían ningún tipo de opciones de accesibilidad, por lo menos no como las conocemos actualmente. Muchas de las funcionalidades que sirvieron para asistir a jugadores con diversidad funcional fueron accidentales o habían sido pensadas para algo diferente por el desarrollador, haciendo que la mejora en la jugabilidad sea más una coincidencia que una característica diseñada adrede. Ejemplos de esto pueden ser la utilización de macros¹ de teclado y mouse originalmente usadas para controlar el sistema operativo, y que algunos juegos no estaban al tanto ni bloqueaban su utilización, y la implementación de interfaces lo suficientemente sencillas como para que una persona ciega pueda navegarlos, aunque no fueran diseñados con ese propósito.

Esto no significa que no existieran opciones de asistencia en juegos anteriores a estas consolas. Por ejemplo, en el juego "Bertie the Brain" de 1950 se incluían niveles de dificultad ajustables, pudiendo configurar el nivel de dificultad de la inteligencia artificial de algunos minijuegos para hacer más accesible el juego para todos (Bateman, 2014). La consola Atari 2600 ofrecía para varios de sus juegos un "modo especial" (indicado con un sticker de un oso teddy en el cartucho), diseñado especialmente para los niños más pequeños, ralentizando la velocidad del juego o alterando la jugabilidad para hacerla más sencilla para niños de edad temprana (Sjardijn & Alyson, 2019).

Hacia finales de la década de 1980 Nintendo crea un dispositivo para controlar los juegos sin utilizar las manos, el Hands Free Controller. Este dispositivo del tipo "sip-and-puff", adaptado para ser compatible con la Nintendo Entertainment System (NES), se ataba al pecho del jugador y permitía aspirar y soplar dentro de un tubo que se podía acomodar con la lengua. La dirección se controlaba utilizando el mentón, y la presión de aire a través del

¹ Secuencia de eventos simulados de mouse o teclado que se utilizan para realizar tareas repetitivas.

tubo se traducía como pulsaciones de los botones (Gadia et al, 2017). Sin embargo, no muchos jugadores conocieron la existencia de este dispositivo de asistencia, dado que no se vendía en tiendas y únicamente estaba disponible para la venta vía línea telefónica. La figura 2.4 muestra la publicidad y el packaging de este dispositivo de Nintendo.



Figura 2.4 Información promocional y empaquetado del dispositivo “Nintendo Hands Free Controller” creado por Nintendo (Wilds, 2020).

Unos años después, a mitad de la década de 1990, la consola SEGA Saturn tomó una decisión que significó una ayuda importante para muchos jugadores con diversidad funcional. Todos los juegos lanzados para esa consola permitían reconfigurar completamente los botones, incluso se lanzó el juego “Real Sound: Kaze no Regret”, específicamente creado para jugadores con deterioro total de la visión, aunque solo estuvo disponible en Japón. A pesar de la buena intención de SEGA de permitir reconfiguración de los controles en sus juegos, estrategia que se mantuvo fuerte al momento de lanzamiento de la consola, con el tiempo se dejó de llevar adelante esta práctica, resultando en que pocos juegos permitieron a los jugadores aprovechar la ventaja de configurar el control del juego con el objetivo de simplificar su uso.

Ian Hamilton, especialista y consultor en accesibilidad y uno de los fundadores de “Game Accessibility Guidelines”, señala que hacia finales de la década del 2000 fue cuando se comenzaron a tomar acciones más contundentes relacionadas a la accesibilidad en la industria de los videojuegos (Wilds, 2020). En 2008 Ubisoft, una compañía de videojuegos

muy reconocida, cambió su política de construcción de software y comenzó a incluir nuevos requerimientos mandatorios en la creación de sus juegos. Estos requerimientos estaban enfocados al uso de los subtítulos y a la intención de evitar disparadores de epilepsia en los juegos. El videojuego “World of Warcraft” uno de los videojuegos multijugador masivo en línea (de su sigla en inglés Massively Multiplayer Online, MMO) más grandes del mundo, agregó un modo con soporte para daltonismo en 2015, y en 2016 Electronic Arts estableció su división de accesibilidad, cuyo trabajo mejoró muchísimo la experiencia de los jugadores en sus juegos de deporte.

En 2017 XBox lanzó su “modo copiloto”, con el cual se puede controlar un único personaje utilizando dos controladores, permitiendo que un segundo jugador asista a un jugador con discapacidad. Microsoft dió un gran paso adelante un año después con la creación del “Adaptive Controller”, un dispositivo específicamente pensado para ayudar a los jugadores con discapacidad.

Aunque muchos de los juegos de la generación actual han mejorado notablemente la accesibilidad con respecto a juegos de generaciones previas, la accesibilidad en general se encuentra en un terreno complicado. A diferencia de otros aspectos de los videojuegos que ya están más maduros y estandarizados dentro de la industria, cómo por ejemplo el soporte para gráficos en 3D, el soporte para sonido envolvente 7.1, avances en inteligencia artificial o networking, la accesibilidad es todavía una de las caras del desarrollo de videojuegos que no está completamente definida. Por ejemplo, las opciones de accesibilidad difieren enormemente según el juego, en algunos se permite configurar el tamaño de los subtítulos y en otros no, en algunos se incluyen opciones de remapeo de botones y en otros no. El problema es que no existe una única forma de realizar accesibilidad, no hay una guía que sirva para todos los juegos, y por el contrario es algo que hay que pensar como un proceso de optimización que no tiene un objetivo fijo a alcanzar (Ian Hamilton, 2021).

Podemos señalar como un aspecto positivo que “la industria realizó un cambio de 180° respecto de cómo estaba hace 3 años. Accesibilidad ya no es una mala palabra” (Barlet et al, 2019). Por ejemplo EA Sports lanzó un portal sobre accesibilidad con el objetivo de facilitar a los jugadores encontrar información sobre features de accesibilidad en sus juegos. La serie de juegos “Uncharted” realizada por Naughty Dog, mejoró muchísimo para su cuarta entrega, ofreciendo una sinfín de opciones de accesibilidad surgidos de una conversación con Josh Straub, un jugador que también ofició de consultor de accesibilidad para Ubisoft y Epic Games (Klepek, 2016). Según Josh, no pudo terminar de jugar “Uncharted 2” debido a un error de accesibilidad hacia el final del juego, lo cual fue una experiencia frustrante y fue uno de los factores que lo decidió a fundar DAGERS (<https://dagersistem.com/>), una web de periodismo para jugadores con diversidad funcional enfocado en la accesibilidad en videojuegos. En el juego “Borderlands 3” (2019) se permite

al jugador escalar la interfaz de usuario completa, y en el juego “Marvel’s Spider-Man” (2018) se introducen muchísimas opciones pensadas para asistir las discapacidades visuales, incluyendo subtítulos de tamaño grandes y el uso de marcas de sonido.

2.5 Diversidad de jugadores

Según el Informe sobre discapacidad realizado por la Organización Mundial de la Salud en 2011, más de 1 billón de personas viven con discapacidad (OMS, Informe mundial sobre discapacidad, 2011). Esto significa que alrededor del 15% de la población mundial vive su día a día con algún tipo de diversidad funcional.

En el caso particular de Argentina, según un estudio realizado por el INDEC, en 2018, sobre el perfil de personas con discapacidad en todo el país, se halló que alrededor de un 10,2% de la población mayor a 6 años presenta algún tipo de discapacidad, esto se corresponde con 3,5 millones de personas (INDEC. Agencia Nacional de Discapacidad, 2018). Haciendo una estimación para el año 2022 este número podría alcanzar los 4,6 millones de personas. La figura 2.5 muestra la cantidad de personas con discapacidad agrupada por rangos etarios. El estudio indica que los rangos etarios más altos son los que reúnen la mayor cantidad de personas con discapacidad.

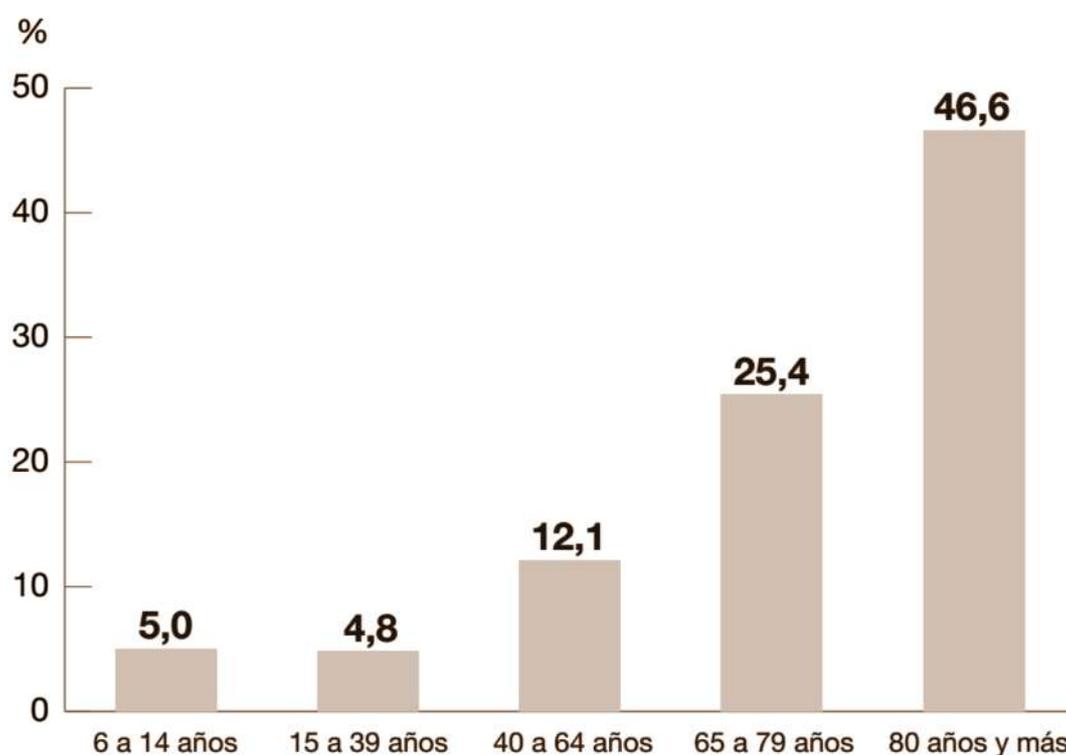


Figura 2.5 Porcentaje de personas de 6 años y más, con dificultad, por grupos de edad (INDEC, 2018)

Mirando los datos del informe podemos realizar un análisis agrupando las discapacidades por rango de edad, con el fin de entender qué tipo de discapacidades afecta a estos grupos:

- En la población de 6 a 14 años de edad predominan las dificultades mental-cognitivas (48.3%).
- En el grupo de 15 a 65 años predominan las personas con deterioro motriz y visual.
- En el rango de edad mayor a los 65 años, predominan las personas con discapacidades motrices.

Por otro lado, los datos discriminados por tipo de discapacidad indican que, la dificultad motriz es la principal causa de discapacidad con un 4,9%, seguido de la discapacidad visual con un 2,5%, la discapacidad auditiva con un 2,1%, dificultades para entender, aprender, recordar o concentrarse con 1,4% y por último las dificultades en el habla y la comunicación con 0,8%.

En este punto, intentaremos responder las siguientes preguntas: ¿quiénes juegan con videojuegos?, ¿es posible caracterizar la población que usa videojuegos?. Para ello, apelamos al informe realizado por DFC Intelligence en el 2020, en donde se señala que casi un 40% de la población mundial juega a los videojuegos, lo que equivale a más de 3,1 mil millones de personas alrededor de todo el mundo (DFC Intelligence, 2020). Hasta hace algunos años, existía una idea bastante generalizada que afirmaba que los videojuegos eran cosa de chicos, que solo tenían la finalidad de entretener a los más chicos y a los adolescentes, y que no tenían mucho valor más allá de la de pasar un rato. Hoy en día vemos que la industria del videojuego tiene un crecimiento acelerado, no solo en los mercados tradicionales de Japón y América del Norte, sino que con el avance de las plataformas de distribución de videojuegos como Steam en PC, Google Play Store en dispositivos móviles, Itch.io en la web, es cada vez más sencillo para los desarrolladores de videojuegos difundir sus juegos alrededor de todo el mundo. Por otro lado, se puede observar un avance significativo del videojuego también en el ámbito social, dado que no solo no se concibe al videojuego como solo una forma de entretenimiento, sino que es aceptado también como una forma de arte (Ellis, 2019). La masividad de los juegos para dispositivos móviles, propició un aumento significativo del promedio de edad de los jugadores a nivel mundial, llegando a un promedio de 36.2 años (Kaplan, 2019).

Newzoo, una consultora internacional que realiza reportes sobre la industria del videojuego en todo el mundo, indicó que durante los últimos tres años, la región de América Latina creció un 11%, convirtiéndose en una de las regiones de más rápido crecimiento. En el caso particular de Argentina, se puede observar que en 2017 la cantidad de personas que juegan videojuegos alcanzan los 18,5 millones de personas (Newzoo, 2017).

Apoyándonos en los datos arrojados por estos reportes, podemos afirmar que la cantidad de argentinos que juegan videojuegos es significativa, e incluso se evidencia un crecimiento en la cantidad de jugadores año a año (Gala, 2019). A su vez, se observa un aumento en la edad promedio de los jugadores, una métrica impulsada por la llegada de adultos y adultos

mayores al sector de juegos móviles en los últimos años (Newzoo, 2022). Teniendo en cuenta la diversidad funcional presente en la población argentina, descrita previamente, podemos concluir que un grupo importante de los jugadores de videojuegos en Argentina se ve afectado en sus posibilidades de juego y esto afecta no solo al grupo etario más joven, sino que al contrario de lo que se podría esperar, afecta a jugadores de todas las edades.

El presupuesto generalizado de que las personas con diversidad funcional representan un pequeño grupo dentro del mercado de los videojuegos puede indicar que la falta de accesibilidad en los videojuegos es por lo general resultado de la omisión y de la falta de atención, más que de un diseño deliberado (Ellis, 2019). En el caso de jugadores con “diversidad funcional” hacemos referencia no solo a personas con algún tipo de discapacidad, sino que también incluimos a personas con limitaciones físicas y mentales, permanentes y temporales. En este sentido, podemos afirmar que las personas con diversidad funcional juegan videojuegos, y como cualquier otra persona tienen derecho de acceder a este tipo de entretenimiento. Comprender esto y darle la importancia que realmente posee es necesario para que la industria del videojuego impulse procesos de desarrollo, lineamientos, guías y estándares en cuanto a la accesibilidad en los juegos que se lanzan al mercado.

2.6 Diversidad funcional en los videojuegos

Tecnologías de asistencia

Como se describió previamente, a partir de informes nacionales e internacionales, actualmente es muy vasta la cantidad de personas con algún tipo de diversidad funcional que accede diariamente a dispositivos digitales como computadoras, teléfonos celulares y consolas de videojuegos. La interacción de estos usuarios con el medio digital puede representar una dificultad grande dependiendo del tipo y grado de diversidad funcional. Para resolver estas dificultades, las personas con discapacidad utilizan algún tipo de tecnología que les permite interactuar con el dispositivo. Estas denominadas tecnología de asistencia, dado que asisten al usuario en la interacción con el medio digital, facilitan su manejo y la utilización del software. La tecnología de asistencia comprende a todos los dispositivos, equipos de hardware y software que ayudan a los usuarios con diversidad funcional a vivir una vida más independiente. De acuerdo a la OMS, la tecnología de asistencia permite a las personas llevar una vida sana, productiva, independiente y digna, y formar parte de los sistemas educativos, del mercado laboral y participar de la vida social (OMS, 2011). Según la OMS en la actualidad solo 1 de cada 10 personas que la necesitan, tiene acceso a tecnología de asistencia, debido a su costo elevado, a la escasa sensibilización sobre el tema, y a la falta de personal formado, de políticas públicas sobre el tema y de financiación

(OMS, 2018). Algunos ejemplos de tecnología de asistencia incluyen: pantallas braille, software de ampliación de pantalla, y la tecnología text-to-speech (traduce de texto a voz). Otro tipo tipos de tecnología de asistencia son las sillas de ruedas, los caminadores y otros productos de apoyo al deterioro de la movilidad.

Dificultades comunes en la interacción con videojuegos

Para comprender dónde pueden aparecer problemas con la accesibilidad en entornos de videojuegos digitales, es útil tener un entendimiento básico del rango de discapacidades y diversidades funcionales que encontramos en el contenido digital y las barreras relacionadas a estos.

Yuan y Folmer (2008) construyeron un modelo de interacción de juego, generalizado, que ofrece una manera de comprender cómo los jugadores interactúan con los juegos. Este modelo está basado en el establecimiento de “puntos en común entre los pasos que se ejecutan cuando se juega a un juego”, a los que los autores arribaron mediante el análisis de cómo los jugadores interactúan con los juegos de diferentes géneros. El modelo de interacción de juego está compuesto de tres pasos: recibir el estímulo, determinar la respuesta y proveer una entrada, como se describe en la figura 2.6.

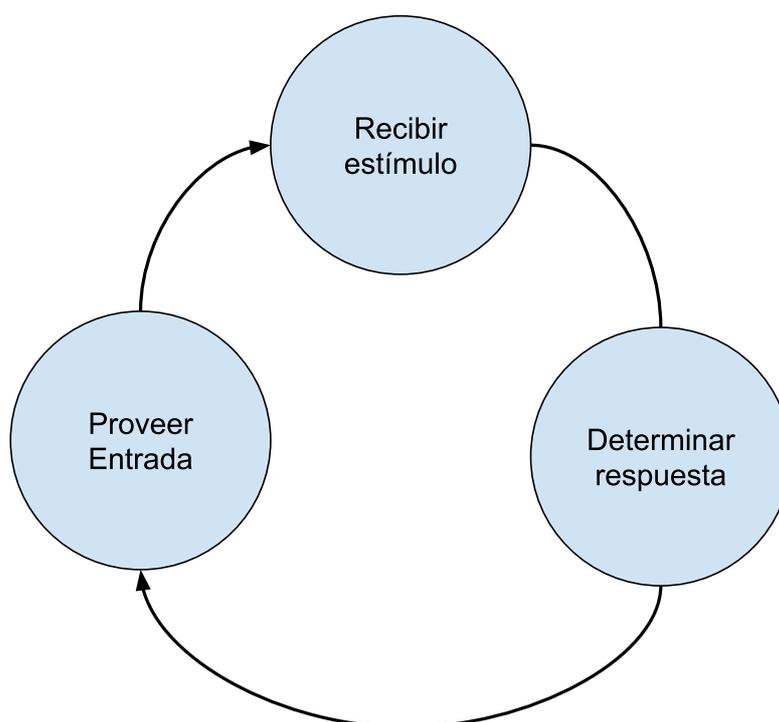


Figura 2.6 El modelo de interacción de juego. Adaptado de Yuan y Folmer (2008)

Para Yuan y Folmer (2008) los videojuegos proporcionan estímulos visuales, auditivos y hápticos, los cuáles pueden ser del tipo primario o secundario. A su vez, los autores afirman que aunque se provean estímulos auditivos y hápticos primarios, éstos por sí solos “no

proveen la suficiente información para permitir jugar un videojuego” y en este sentido señalan que los elementos visuales constituyen la fuente principal de retroalimentación para el jugador. Por otra parte, los estímulos secundarios son suplementarios y ofrecen feedbacks o retroalimentaciones para progresar en el juego, pero por lo general son esenciales para incrementar la inmersión y enriquecer la experiencia de juego. Al recibir un estímulo, el jugador “debe determinar cognitivamente qué respuesta debe utilizar dentro del juego”, de las opciones disponibles. En muchos casos el jugador puede elegir una sola o una combinación de acciones como respuesta. Una vez que el jugador decidió una respuesta, debe confirmar físicamente a través de la interacción con un dispositivo de entrada, los cuales pueden variar dependiendo del juego, como por ejemplo un teclado, joystick o gamepad. Yuan y Folmer (2008) clasifican los dispositivos en dos tipos: discretos y analógicos; los primeros permiten dar respuestas binarias, como por ej. prendido o apagado, por ejemplo el teclado y los segundos miden la entrada continuamente, como el mouse de una computadora. Proveer la entrada se vuelve más difícil y complejo cuando se combinan las entradas discretas y analógicas, como es el caso con los controladores de juegos como el gamepad, donde tenemos botones (entradas discretas) y palanca (analógica) (Yuan y Folder, 2008; Hu, 2017).

A partir del modelo de interacción en videojuegos propuesto por Yuan y Folder, es posible clasificar en tres categorías a las dificultades relacionadas a la accesibilidad, de acuerdo a la interactividad intrínseca de los juegos:

- dificultades al recibir estímulos, ya sean visuales, auditivos, táctiles.
- dificultades al momento de procesar la información y determinar la respuesta adecuada para poder avanzar en el juego.
- dificultades al proporcionar un input al juego.

Dado que la intención de esta tesina es acercar problemática de la accesibilidad en videojuegos desde la perspectiva humana y de las personas, se analizarán las dificultades en la interacción con videojuegos desde el punto de vista físico y de salud, teniendo en cuenta discapacidades, trastornos y limitaciones que pueden presentar los jugadores y cómo éstos afectan en el juego. Es por ello, que en este punto abandonamos la clasificación propuesta por Yuan y Folder (2008).

Dicho esto, las dificultades claves que afectan a la accesibilidad en videojuegos se pueden separar en cuatro grandes categorías principales (Yuan & Folmer, 2010):

- Visuales: como la ceguera, la baja visión y el daltonismo.
- Auditivas: que pueden oscilar desde la pérdida leve hasta la pérdida total de la audición.
- Motrices: como la ausencia de extremidades, dedos, parálisis, pérdida de la motricidad fina, inestabilidad o dolor al usar los dedos, manos, muñecas o brazos.

- Cognitivas: incluye dificultades en el desempeño de tareas mentales, oscilando entre problemas leves que afectan una función cognitiva específica, hasta casos severos donde los jugadores no pueden llevar adelante las actividades de su día a día

Algunos ejemplos de condiciones limitantes al momento de jugar incluyen jugar:

- en un idioma extranjero
- en un ambiente muy oscuro o muy brillante
- en un ambiente muy silencioso o muy ruidoso
- en movimiento, jugar cansado
- siendo un jugador novato o casual
- siendo muy joven o muy anciano
- siendo zurdo
- usando un dispositivo de entrada alternativo, como puede ser un teclado virtual
- usando dispositivos con limitaciones o problemas técnicos, por ejemplo un teclado con teclas rotas, o en una pantalla que está muy lejos.

Las condiciones descritas suponen complicaciones al momento de interactuar con el videojuego. Algunas son circunstanciales, como por ejemplo jugar cansado, algo que puede generar problemas para identificar elementos en la pantalla, disminuir la velocidad de reacción y bajar la calidad de procesamiento de información compleja. Otras pueden ser condiciones permanentes, como por ejemplo jugar siendo zurdo; se calcula que aproximadamente un 12% de las personas del mundo son zurdas, aunque varía según la región geográfica y otros factores y cuando se interactúa con contenido digital, y especialmente con los videojuegos, los usuarios pueden encontrarse con contenido diseñado para personas no zurdas, como por ejemplo en el uso del teclado, mouse y joystick (Mangiron et al, 2014). A continuación se describirán algunas de las dificultades claves que afectan la accesibilidad de los videojuegos.

Dificultades en la visión

Los jugadores con pérdida total de la visión son los que tienden a encontrar la mayor cantidad de barreras al acceder al contenido digital en general, dada la naturaleza visual del contenido. En la interacción con los videojuegos por lo general se produce una barrera tan grande que puede llegar a impedir el acceso a los jugadores. Estos jugadores utilizan por lo general un lector de pantalla para acceder a la computadora o dispositivo móvil, y opcionalmente una pantalla braille para entender el texto que aparece en pantalla.

Las barreras más comunes para este grupo incluyen:

- Interfaces de usuario demasiadas complejas o con contenido excesivo
- Elementos funcionales que no pueden ser accedidos vía teclado o joystick
- Contenido visual que no tiene alternativa en texto

- Falta de habilidad para navegar por el contenido
- Contenido que no está correctamente estructurado, dificultando la interpretación de lectores de pantalla
- Navegación inconsistente de la interfaz
- Límites de tiempo, tiempo insuficiente al completar tareas
- Acciones o flujos inesperados, por ejemplo redirigir a otro punto de la navegación cuando un elemento recibe el foco
- Elementos multimedia sin descripción por audio

Existen algunos juegos diseñados específicamente para jugadores con pérdida total de la visión, denominados “audio-juegos” o “juegos de audio”, que se apoyan en el sonido para crear una experiencia interactiva que pueda ser utilizada sin problemas por un jugador con pérdida total de la visión. Algunos ejemplos de audio-juegos son:

- A Blind Legend, Dowino (2016)
- Dark Echo, RAC7 Games (2015)
- Paladin of the Sky, VGStorm (2013)

En este punto es importante diferenciar entre los juegos accesibles y los juegos inclusivos. Como se señaló en secciones previas, la inclusividad se basa en brindar la misma posibilidad de interacción y participación a todos los individuos, en este caso jugadores, sin presentar una estrategia diferenciada según alguna condición propia como puede ser una discapacidad. Los videojuegos con un diseño inclusivo permiten que cualquier persona pueda jugarlos de la mejor manera posible, sin importar sus condiciones físicas o mentales. Bajo esta distinción, podemos decir que los juegos mencionados anteriormente son juegos accesibles, pero no así inclusivos dado que permiten ser accedidos de una forma excelente por jugadores con pérdida de la visión, sin embargo presentan grandes dificultades en la interacción para otros tipos de jugadores. *La accesibilidad en videojuegos a menudo resulta en el llamado “efecto de corte de acera” o “Curb Cut Effect”. Esta analogía se ilustra con la rampa en las esquinas que fue inicialmente diseñada para beneficiar a las personas en sillas de rueda, pero que terminó beneficiando a muchos.* (GamesHub, 2022).

Algunos jugadores con discapacidad visual pueden jugar a videojuegos que no fueron diseñados específicamente para personas con este tipo de discapacidad. Este es el caso de Michael Espinoza, un jugador ciego nacido en Texas, Estados Unidos, quien juega Mortal Kombat guiándose sólo mediante el sonido del juego, como se puede observar en la figura 2.7 (Sánchez, 2019). Esto es posible gracias a que los personajes emiten sonidos característicos con cada acción que realizan, permitiendo a Michael identificar cuándo un personaje está saltando, golpeando, o bloqueando un ataque.



Figura 2.7 Michael Espinoza, un jugador ciego utiliza su capacidad auditiva para jugar Mortal Kombat

Los jugadores con baja visión pueden, por lo general, acceder al contenido digital si está amplificado. Es posible utilizar un software amplificador de pantalla para agrandar el tamaño y cambiar el contraste del contenido para hacerlo más visible. Es menos probable que estos jugadores utilicen un lector de pantalla, cosa que sí es común en los jugadores con pérdida total de la visión. Los jugadores con baja visión dependen de las opciones de amplificación y personalización del texto presente en los videojuegos y sistemas operativos, o de herramientas de software instaladas específicamente para este propósito.

Algunas barreras comunes para las personas con dificultades en la visión, incluyen:

- Contenido diseñado para ser utilizado en un solo tamaño, por ejemplo diseñado en píxeles en lugar de unidades de longitud relativas.
- Navegación inconsistente de la interfaz.
- Imágenes o texto que pierden calidad cuando están amplificadas.
- Texto con tipografías difíciles de leer.
- Bajo contraste, por ejemplo dificultad para distinguir entre el texto y el fondo.
- Límites de tiempo.
- Acciones inesperadas.

Entre las personas con dificultades en la visión también están incluidos los jugadores que conviven con algún tipo de alteración de la visión, como puede ser daltonismo o glaucoma. Estos jugadores se ven muy favorecidos por la presencia de opciones de personalización para los colores de texto, menús, y otros elementos del HUD (Head-Up Display), así como también opciones para cambiar el tamaño de las fuentes y otros componentes de la interfaz.

Adicionalmente un buen diseño en la utilización de colores en los niveles, personajes, y entorno permiten a estos jugadores jugar con mayor facilidad. En la figura 2.8 se puede observar el modo de accesibilidad implementado en el juego “The Last of Us: Part 2”, donde se pintan con diferentes colores a los personajes, enemigos y elementos interactivos. Adicionalmente se permite realizar un zoom sobre cualquier sección de la pantalla utilizando el joystick. Estas mejoras son muy útiles para los jugadores con deterioro visual y también cognitivo.



Figura 2.8 Modo de accesibilidad que resalta los enemigos y elementos interactivos con diferentes colores, útil para jugadores con dificultades visuales y cognitivas. The Last of Us: Part 2 (2020)

Dificultades en la audición

La mayoría de los jugadores que poseen pérdida de la audición se enfrentan a barreras relacionadas a la falta de alternativas para la información comunicada a través del sonido. Por lo general este es el único tipo de barrera que se encuentran para acceder al contenido digital y a los videojuegos. Para aquellos jugadores que se comunican mediante lenguaje de señas, el lenguaje escrito en un contenido digital, como puede ser un videojuego puede producir barreras similares a aquellas que se producen al leer un lenguaje extranjero. Algunas de las barreras que se enfrentan las personas con dificultades de audición, incluyen:

- Audio sin transcripción alternativa en texto.
- Multimedia sin subtítulos o transcripciones.
- Falta de utilización de lenguajes alternativos, como el lenguaje de señas.

- Problemas en el diseño del juego cuando se emite información vital para el jugador únicamente mediante el sonido.

La mayoría de los elementos esenciales de un juego suceden a través del sonido, por ejemplo, el sonido de los pasos de los jugadores dentro de los videojuegos del género Battle Royale², dado que permiten saber si hay un enemigo cerca. Es por esto que los jugadores con deterioro parcial o completo de la audición necesitan de subtítulos consistentes al jugar videojuegos.

Un caso importante en donde se evidencian significativamente las dificultades relacionadas a la audición es en los juegos del género multijugador online. Estos juegos son por lo general muy competitivos, los jugadores compiten entre ellos, solos o en equipos, con lo cual es necesario una comunicación fluida para ponerse de acuerdo y generar una estrategia de juego y hacer observaciones durante la partida. Desafortunadamente en estas situaciones, los jugadores con pérdida de la audición suelen quedar excluidos porque algunos juegos no ofrecen las herramientas necesarias para tomar parte de estas conversaciones.

En el capítulo 4, sobre diseño de accesibilidad, se presentarán posibles soluciones para estos tipos de problemas. En este punto, resulta relevante mencionar que los problemas a los que se enfrentan actualmente las personas con dificultades auditivas al intentar jugar videojuegos multiplayer online va en aumento, dado que la cantidad de jugadores que se suman a este tipo de juegos se ha incrementado significativamente o en los últimos años (Gough, 2022; Observatorio de la Industria del Videojuego, UNRaf, 2022), y son pocos los videojuegos que atienden el problema y ofrecen soluciones adecuadas. Una de las excepciones es el juego “Back 4 Blood” de Turtle Rock Studios (2021), en donde se incluyó entre las opciones de accesibilidad una opción “Speech To Text”, como se muestra en la figura 2.9). Esta opción procesa automáticamente la conversación en audio que están teniendo los jugadores, y la muestra en pantalla en forma de subtítulos, permitiendo de esta manera que los jugadores con pérdida de la audición puedan participar de la conversación.

² Battle Royal es un género de videojuegos multijugador que combina elementos de supervivencia con batallas multijugador



Figura 2.9 Menú de accesibilidad de “Back 4 Blood” (2021) donde se puede ver la opción “Speech To Text”, la cual permite traducir el chat de voz a texto en pantalla.

Dificultades en la movilidad

Las dificultades relacionadas a la movilidad son muy variadas. Aquellas personas con uso limitado de las manos, o con deterioro de la motricidad fina pueden encontrar limitaciones en su habilidad para alcanzar o clickear elementos del contenido digital mediante el uso del mouse o pantalla táctil. Incluso puede que no utilicen un mouse o pantalla táctil, y en su lugar dependan de otros dispositivos como el teclado, control por voz, o dispositivos tipo switch o interruptores para poder acceder al contenido digital. Por otro lado, un jugador podría utilizar sillas de ruedas pero no encontrar ningún tipo de barrera para acceder al contenido digital.

Algunas barreras comunes a las que se enfrentan las personas con dificultades en la movilidad, incluyen:

- Áreas clickeables muy pequeñas.
- Elementos funcionales que no pueden ser accedidos por teclado.
- Límites de tiempo en la interacción con elementos funcionales.
- Falta de soporte para realizar input remapping (configuración del control).
- Controles complejos.
- Control complejo de la cámara.

Este tipo de dificultades están íntimamente relacionadas a los controladores típicos de las consolas de videojuegos como joysticks y gamepads, o en el caso de los juegos de computadora, con el teclado. Los jugadores con deterioro motriz pueden presentar dificultades para utilizar correctamente los controles, ya sea por características que limitan el movimiento independiente o el control de una o más partes del cuerpo, o por limitaciones

en la coordinación, fuerza o agilidad. Algunos ejemplos de dificultades en la movilidad que pueden generar barreras en la interacción con videojuegos incluyen: artritis, tendinitis, Parkinson, lesiones musculares, amputaciones, atrofia muscular, etc.

En este sentido, podemos afirmar que los problemas en la interacción con videojuegos a los que se enfrentan los jugadores con dificultades en la movilidad, se relacionan fuertemente al diseño de los controles. Por ejemplo, un juego que fuerza al jugador a realizar una combinación de botones excesivamente compleja puede generar problemas para el jugador con discapacidad motora. Un ejemplo de esto lo podemos observar en el juego “Batman: Arkham Knight” (2015), donde al complejo control del combate de los juegos anteriores de la misma serie, se le suma la complejidad del control del batimóvil que requiere en algunos momentos combinaciones complejas de hasta 3 dedos diferentes, lo que sumado a su aparición durante todo el juego, genera importantes dificultades por ejemplo para los jugadores con deterioros en la motricidad fina (Josh Straub, 2015). Otro ejemplo al que se enfrentan los jugadores con dificultad motriz es el caso del jugador Rocky Stoutenburg, también conocido como “RockyNoHands”, un jugador y streamer que a los 19 años sufrió un accidente que lo dejó cuadripléjico. A pesar de no poder mover sus brazos, rompió varios Guinness World Records en el juego Fortnite en el año 2020, y además se transformó en el primer jugador cuadripléjico que ingresó a una organización de E-Sports. Para jugar Rocky utiliza un dispositivo llamado QuadStick, que es un controlador que consiste de un brazo articulado flexible con un dispositivo montado en la punta que permite controlar la computadora o consola mediante la boca, como se puede observar en la figura 2.10.

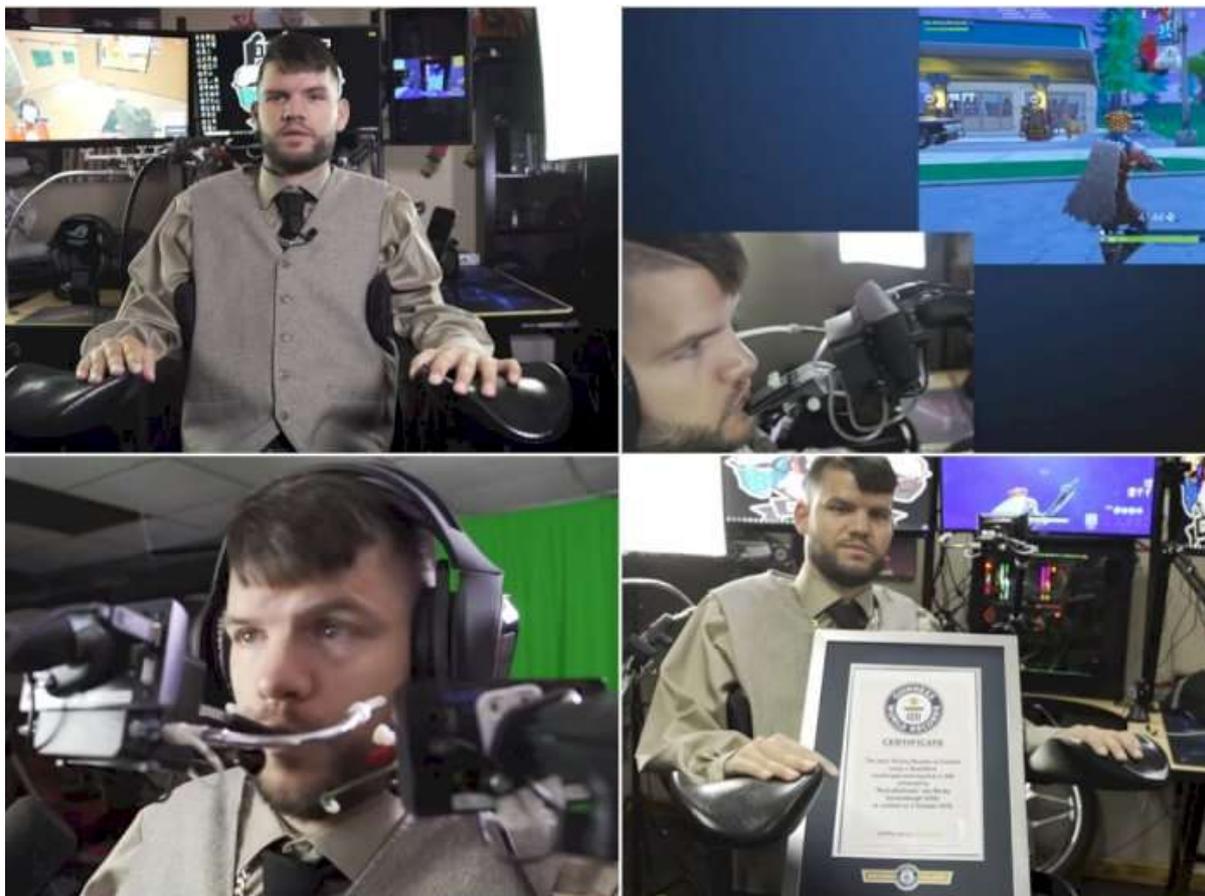


Figura 2.10 El jugador Rocky Stoutenburg utiliza el dispositivo QuadStick para participar en juegos de E-Sports.

Dificultades cognitivas

Las dificultades cognitivas o las relacionadas al aprendizaje pueden ser tan variadas como las relacionadas a la movilidad, desde una leve dificultad relacionada a la lectura, hasta un deterioro fuerte de las capacidades cognitivas que resultan en un uso limitado del lenguaje y dificultad en el procesamiento de información compleja.

Cuando hablamos de discapacidad cognitiva nos referimos al término que funciona como paraguas para un subconjunto de discapacidades de la salud mental que principalmente afectan a la función cognitiva. La mayoría de las discapacidades cognitivas impactan las habilidades para aprender, para resolver problemas, la memoria y la habilidad para acceder y procesar la información (Gauci, 2021). Algunas de las discapacidades cognitivas que pueden dificultar la interacción con los videojuegos son: trastorno del desarrollo intelectual, trastorno del espectro autista, trastorno del aprendizaje, trastorno por déficit de atención o hiperactividad, y trisomía 21 también conocido como Síndrome de Down.

Las manifestaciones de estos trastornos son variadas y pueden impactar en las habilidades del jugador en múltiples formas. Consecuentemente, se vuelve muy difícil identificar qué barreras enfrentan los jugadores.

Algunas barreras comunes relacionadas a dificultades cognitivas, incluyen:

- Utilización de lenguaje excesivamente complejo o avanzado.
- Navegación inconsistente de la interfaz.
- Excesiva cantidad de contenido, o contenido excesivamente complejo.
- Límite de tiempos en la interacción.
- Falta de estructura en los contenidos.
- Acciones inesperadas.

Aunque los jugadores con dificultades cognitivas pueden no experimentar barreras físicas para recibir los estímulos durante el juego, existen algunos aspectos sobre el feedback dentro del juego que puede seguir provocando inaccesibilidad. Debido al déficit de atención, algunos jugadores pueden distraerse por elementos visuales que no son directamente relevantes para el juego. La distracción también puede ocurrir cuando el juego es visualmente estático, ya que el jugador puede perder la concentración y quitar la atención del juego. Por otra parte, si un juego posee sobreabundancia de estímulos visuales, se puede inducir ansiedad, dado que el jugador puede no distinguir los diferentes elementos. Lo mismo se puede aplicar a los estímulos auditivos, que pueden ser percibidos como ruido (Bartoli et al, 2014). Adicionalmente un contenido visual parpadeante o “un contenido fuertemente emocional” puede no ser adecuado para jugadores con dificultades cognitivas, dado que pueden inducir ansiedad, o en algunos casos más severos, ataques de epilepsia (“Accessible Player Experiences (APX)”, 2020).

Los jugadores con dificultades cognitivas pueden encontrar múltiples retos para determinar la respuesta a un estímulo durante el juego. Por ejemplo, los jugadores con trastornos en el aprendizaje pueden tener problemas para leer y entender las instrucciones durante el juego, y por consecuencia no lograr determinar cómo responder. Los trastornos en el razonamiento y en la resolución de problemas también puede impedir a los jugadores con dificultades cognitivas determinar qué respuesta proveer al estímulo. Adicionalmente a los desafíos que presenta el juego, los jugadores con dificultades cognitivas pueden tener problemas al determinar respuestas cuando el juego requiere realizar múltiples acciones.

Dependiendo del grado del trastorno, los jugadores pueden tener problemas con las configuraciones de control más complejas. Por consecuencia, algunos jugadores con dificultades cognitivas pueden utilizar dispositivos de entrada alternativos a los controles estándar como el gamepad o teclado. En estos casos el jugador no puede determinar qué acción dentro del juego realizar o cuál botón en el controlador se corresponde a esa acción (Yuan y Folmer, 2008).

Un ejemplo de dificultad cognitiva es el de la dislexia, un trastorno del aprendizaje por el cual se vuelve difícil leer, escribir y procesar el texto escrito. En el juego “Overland” (2019) se incluye una opción de accesibilidad mediante la cual se permite utilizar una fuente más amigable con los jugadores que poseen dislexia, un cambio en el diseño que se realizó luego de que un jugador hiciera notar a los desarrolladores de este problema particular. En la figura 2.11 se puede observar la utilización de esta tipografía.

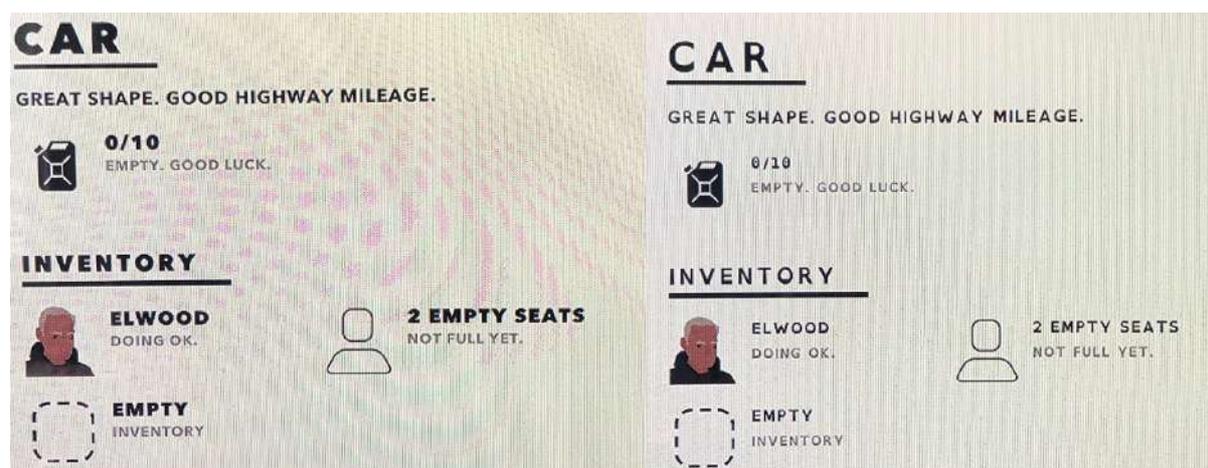


Figura 2.11 Captura de pantalla del juego “Overland” (2019). En la izquierda se ve la tipografía usada por defecto, en la derecha la tipografía amigable con jugadores disléxicos.

Relación entre usabilidad y accesibilidad en videojuegos

Aunque por lo general pensamos en las barreras en términos de accesibilidad para personas con discapacidad, existen muchas barreras que afectan a todos los tipos de jugadores. Estas son por lo general consideradas en términos de usabilidad. La usabilidad y la accesibilidad son conceptos que están íntimamente relacionados. Agregar funciones de accesibilidad para un grupo específico de jugadores puede mejorar la usabilidad para el resto de los jugadores (recordemos el “efecto corte de acera”). Muchos jugadores, incluso aquellos que consideran no poseer ningún tipo de discapacidad, por ejemplo personas adultas mayores a los 50 años, pueden experimentar efectos propios de la edad como la pérdida de la visión, problemas de audición, o habilidad cognitiva.

Algunos problemas de usabilidad comunes incluyen:

- Botones y otros elementos funcionales con texto que no describen correctamente el funcionamiento esperado.
- Contenido excesivamente complejo.
- Navegación inconsistente.
- Bajo contraste en el diseño de la interfaz.

- Bajo contraste o dificultad para diferenciar el fondo de los personajes o de los elementos interactivos.

Aunque no son consecuencia directa de alguna discapacidad o trastorno presente en el jugador, las barreras en la usabilidad pueden llegar a generar dificultad en la interacción entre el jugador y el videojuego. En la figura 2.12 se puede observar la ocurrencia de los 12 problemas de usabilidad presentes en múltiples géneros de videojuegos (Livingston, Mandryk, Stanley, 2010).

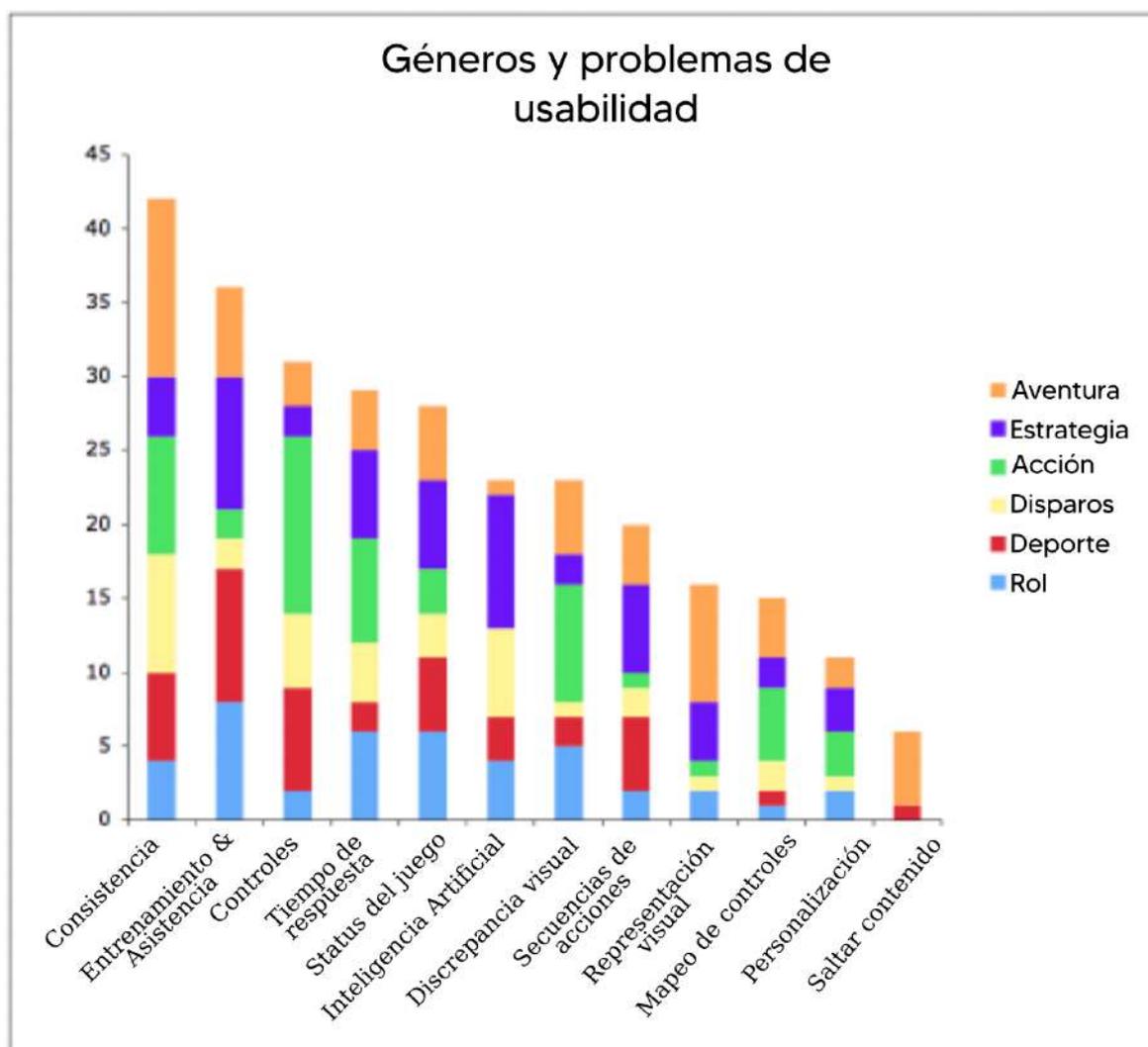


Figura 2.12 Ocurrencia de los 12 problemas de usabilidad más comunes agrupados por género (adaptado de Livingston, Mandryk, Stanley, 2010)

Según Livingston, Mandryk y Stanley (2010), los 12 problemas de accesibilidad más comunes encontrados en los videojuegos incluyen:

1. Consistencia: detección de colisiones pobre, simulación de físicas pobres, respuestas inconsistentes al input.

2. Entrenamiento y asistencia: no se proveen elecciones por defecto o recomendadas, no se provee sugerencias o ayuda, no se provee la documentación, instrucciones, tutoriales o misiones de entrenamiento adecuadas.
3. Controles: controles con mucha sensibilidad, controles no naturales o no responsivos.
4. Tiempos de respuesta: tiempos de respuesta lentos interfiere con la habilidad del jugador de interactuar correctamente con el juego.
5. Estado del juego: no se provee información adecuada del personaje, del entorno, o de los enemigos. Indicaciones visuales, íconos, y mapas inapropiados.
6. Inteligencia artificial: problemas con el pathfinding (técnica para encontrar el camino entre dos puntos utilizada para la navegación), problemas con los NPC (personajes no jugables, controlados por la computadora).
7. Discrepancias visuales: malos ángulos de cámara, obstrucciones en la vista, la vista no se ajusta lo suficientemente rápido a las acciones del usuario.
8. Secuencias de acciones: la curva de aprendizaje es muy empinada, se requiere mucha microgestión, las secuencias de acciones a tomar son muy complejas, largas o incómodas.
9. Representaciones visuales: mala visualización de la información, mucho desorden en la pantalla, muchos personajes o elementos de juego en la pantalla al mismo tiempo, dificultad para distinguir contenido interactivo del no interactivo.
10. Mapeo de controles: mala implementación del input mapping, soporte de dispositivos limitado, personalización de los controles limitado.
11. Personalización: no se permite al usuario cambiar las configuraciones de audio, dificultad o velocidad del juego.
12. Saltar contenido: no se pueden saltar clips de vídeo o sonido, secuencias que se repiten con demasiada frecuencia.

2.7 Tecnologías de asistencia para jugadores

Estar informado sobre los avances en las tecnologías de asistencia y cómo funcionan es un aspecto relevante a tener en cuenta tanto para los creadores de contenido digital (sitios web, aplicaciones de escritorio, apps para dispositivos móviles o de escritorio) como para los creadores de videojuegos. Entender esto es imprescindible para poder tomar decisiones en el diseño e implementación del videojuego, eliminando barreras en la interacción con el videojuego para los jugadores con algún tipo de discapacidad y mejorando la usabilidad del juego en general. A continuación se describirán las tecnologías de asistencia más comunes en la actualidad.

Teclados adaptados

Este tipo de tecnología de asistencia incluye una variedad de teclados diseñados especialmente para mejorar la interacción de jugadores que presentan determinadas dificultades. Existen una amplia variedad de diseños que se ajustan a discapacidades y problemas de accesibilidad específicos. Los jugadores con destreza o control motriz limitado, por ejemplo suelen utilizar teclados con teclas hundidas en lugar de levantadas. Esto permite a la persona encontrar una tecla en particular y posicionar sus manos de una manera más sencilla. Presionar teclas en un teclado como estos requiere una acción más deliberada por parte del jugador.

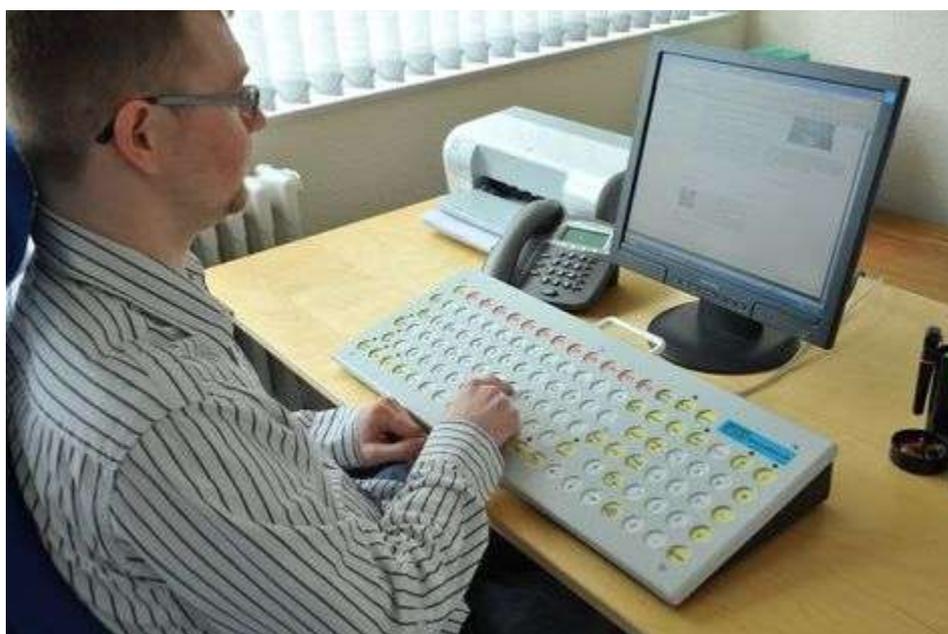


Figura 2.13 Teclado expandido Maltron

El teclado expandido Maltron, como se muestra en la figura 2.13 es un ejemplo de teclado adaptado, su amplia superficie hace más fácil su utilización para personas con dificultades en el control muscular, como la parálisis cerebral. Este teclado posee un protector de teclas, el cual ayuda al usuario a evitar pulsaciones accidentales. Por otro lado, los teclados de alto contraste ayudan a los jugadores con pérdida parcial o deterioro de la visión a diferenciar las teclas más fácilmente. A menudo los teclados adaptados vienen con un software que incluye auto completado para palabras, lo que acelera el proceso de tipeo y reduce la cantidad de pulsaciones accidentales.

Algunos jugadores prefieren la utilización de un cobertor para el teclado, una lámina de plástico que permite marcar con diferentes colores teclas específicas. Algunos cobertores incluyen símbolos o imágenes para hacer el teclado aún más accesible.

Dispositivos de entrada alternativos

Los dispositivos de entrada más comunes para interactuar con una computadora son el teclado y el mouse, aunque no son los únicos. Existen dispositivos de entrada alternativos que brindan a los usuarios con movilidad reducida nuevas opciones de interacción, permitiendo por ejemplo a los jugadores de computadoras encontrar una forma alternativa de interacción con los juegos. Estos dispositivos son activados generalmente por tecnologías de rastreo de movimiento, aunque también se pueden utilizar otros tipos de activaciones como control por señales eléctricas en los músculos, rastreo óptico e incluso actividad cerebral. La mayoría trabajan con un teclado en pantalla, un tablero de palabras o una interfaz de pictogramas. A continuación se describen algunos de estos tipos de dispositivos.

Puntero cefálico

Un lápiz táctil u otro tipo de dispositivo se monta en la cabeza, permitiendo al usuario interactuar con la computadora utilizando los movimientos de su cabeza. Este es uno de los dispositivos más simples, dado que no necesita una configuración complicada y es sencillo de aprender a utilizar. La figura 2.14 muestra un ejemplo de puntero cefálico. Existe una gran variedad de tipos de punteros, que permiten al usuario interactuar con pantallas táctiles, con el teclado, o con otros periféricos de entrada.



Figura 2.14 Un niño utiliza un puntero cefálico para interactuar con un teclado adaptado

Dispositivo de botón o interruptor simple

Este tipo de dispositivo consiste en un botón único que envía una señal a la computadora cuando es presionado, realizando una acción, generalmente su apariencia es de un botón

redondeado y de color llamativo. Este botón se puede customizar para realizar cualquier acción requerida, y a veces es utilizado en conjunto con otros botones para realizar múltiples acciones, siendo muy aprovechado en la comunidad de jugadores de videojuegos, dado que por lo general los juegos necesitan realizar una serie de acciones con el teclado o mouse que pueden llegar a ser emulados por estos tipos de interruptores. En la figura 2.15 se pueden observar diferentes tipos de interruptores simples y controles que permiten enviar la señal del interruptor a la computadora. En la figura 2.16 vemos el mismo tipo de interruptores pero en forma de pedal, dado que se utilizan presionando el botón con los pies.



Figura 2.15 Diferentes tipos de interruptores y controladores adaptativos



Figura 2.16 Interruptores de pie, o pedales.

Interruptores “sip-and-puff”

Estos dispositivos permiten al jugador controlar una computadora usando la inhalación y exhalación de una pequeña cantidad de aire a través de un tubo. El dispositivo se conecta a una computadora, donde se puede utilizar en conjunto con un software que muestra una interfaz con una variedad de áreas seleccionables que representan acciones a realizar. Cuando una de estas áreas toma el foco, el usuario puede “hacer clic” sobre la misma soplando o aspirando aire, dependiendo de cómo esté configurado el dispositivo.



Figura 2.17 Dispositivo Sip & Puff, de la marca Permobil

Software de seguimiento de la mirada y la cabeza

Es un software especializado que realiza un seguimiento de los ojos del usuario, de manera que éste puede interactuar con la interfaz de un sitio web, teclado en pantalla, o aplicación utilizando su mirada. La tecnología se basa en la utilización de una cámara que observa la posición de los ojos del jugador, o incluso la posición de su cabeza y transforma esa

información en comandos que puede interpretar una computadora. Por lo general, se controla el cursor en pantalla, pero también se puede utilizar en combinación con un teclado en pantalla para ingresar texto. Por supuesto estas señales también se pueden traducir a acciones dentro de un videojuego. La idea de utilizar la posición y rotación de los ojos y la cabeza para controlar la computadora no es algo nuevo. El ejemplo más conocido es el de Stephen Hawking, quien utilizaba un software de seguimiento del movimiento de su mejilla, desarrollado por Intel, para comunicarse usando un software de predicción de palabras en conjunto con un software Text-To-Speech. Sin embargo, para jugar videojuegos, la interacción puede ser un poco más complicada. Muchas veces es necesario tener un nivel de precisión muy avanzada, y una velocidad de reacción mucho más rápida si se quiere ganar en juegos de simulación de carreras, juegos de disparos en primera persona y otros tipos de géneros de juegos en PC.

En los últimos años este tipo de tecnología avanzó al punto de ser tan precisa y rápida que puede ser utilizada para jugar juegos complejos. Esto es posible si se combina con otros tipos de tecnología de asistencia como los interruptores para lograr un control más preciso. Para esto es importante el soporte y la ayuda de los desarrolladores de videojuegos, al permitir mapear los controles y acciones dentro de los juegos a pulsaciones de teclado o de interruptores que facilitan el control. Esta tecnología también la podemos encontrar integrada en teléfonos celulares, ampliando su alcance, y permitiendo abaratar los costos de controles basados en el seguimiento de los ojos. Las compañías desarrolladoras del sistema operativo de los teléfonos celulares habilitan el uso de las APIs de seguimiento integrada a la cámara, lo cual abre la posibilidad a otros desarrolladores a crear nuevas aplicaciones que utilizan el celular como una herramienta más a la hora de controlar la computadora.

En la figura 2.18 vemos cómo se utiliza el software Eyewear Beam, basado en el software de seguimiento de la mirada de los teléfonos iPhone, para controlar un juego complejo como es "League of Legends".



Figura 2.18 Demostración de utilización del software Eyeware Beam para jugar League of Legends usando la mirada.

Herramientas de comunicación alternativa

También conocida como AAC, por las siglas en inglés de Augmentative and Alternative Communication (comunicación aumentativa y alternativa). El foco de este tipo de dispositivos es resolver barreras en la comunicación, ofreciendo un rango de herramientas y métodos para personas que no se comunican verbalmente o tienen problemas para hablar. Se puede considerar al lenguaje de signos y señas como uno de los métodos de comunicación alternativa. Dentro de las herramientas digitales podemos incluir la utilización de tableros de palabras y pictogramas, como por ejemplo la aplicación ElineSpeaks como se puede observar en la figura 2.19.

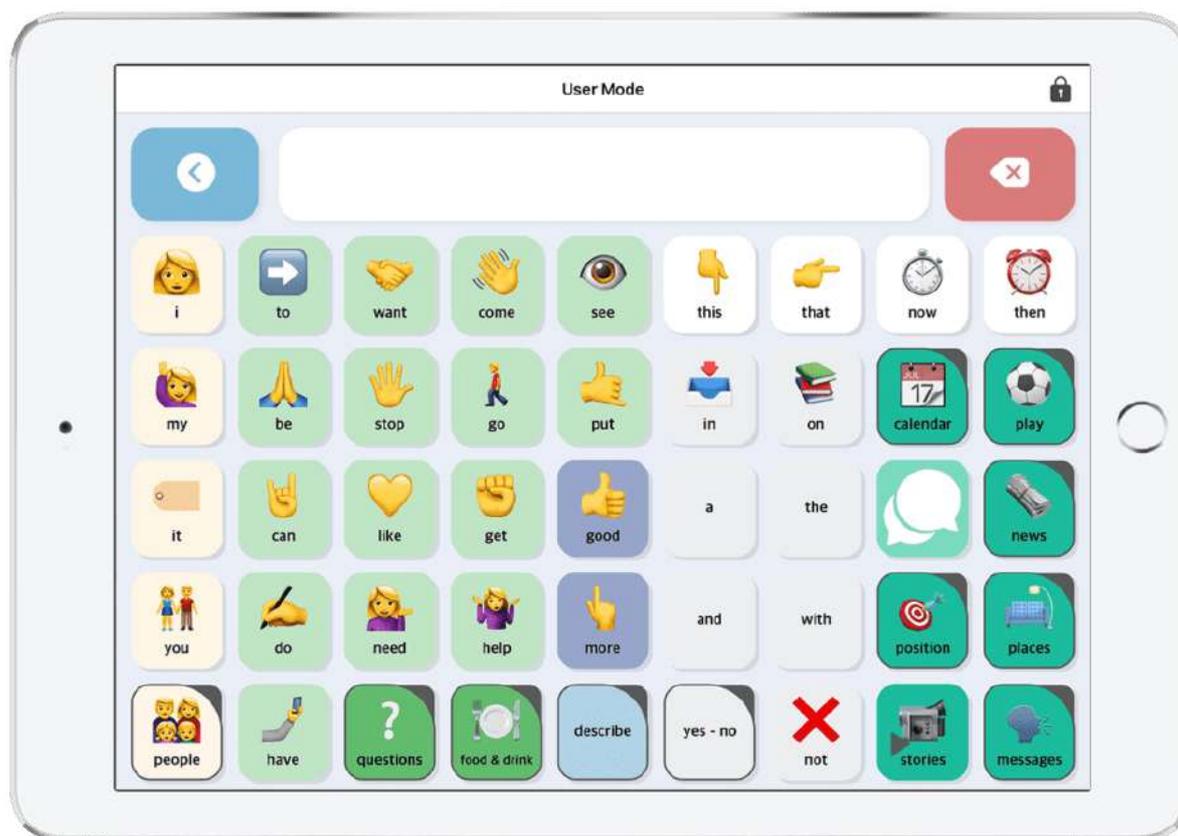


Figura 2.19 La aplicación ElineSpeaks es un ejemplo de comunicación alternativa mediante pictogramas.

Uno de los sistemas de AAC más conocidos es el que utilizaba Stephen Hawking para comunicarse, el cual incluía un dispositivo de salida de voz. Estos dispositivos sintetizan la voz humana mediante la utilización de un software especial, permitiendo a los usuarios comunicarse mediante el habla.

Aunque la intención en esta tesina no es la de profundizar en el funcionamiento de la comunicación alternativa, sí es importante comprender que es un instrumento que principalmente se enfoca en resolver el problema de la comunicación. Si combinamos esto con los videojuegos podemos obtener una herramienta muy potente para contribuir en la mejora de la comunicación, y fomentar la interacción, el aprendizaje, el pensamiento y resolución de tareas, y el entretenimiento (Harmann, 2018).

Línea braille

Está formado por un display táctil y dinámico que convierte el texto digital que aparece en la pantalla de la computadora en texto braille, el cual puede ser leído por un jugador que entiende este sistema de escritura. Se pueden conectar a una computadora, smartphone o tablet mediante bluetooth o cable USB. Algunas versiones de este dispositivo incluye un teclado QWERTY, quedando disponible los dos para que los jugadores los puedan utilizar.

La figura 2.20 muestra un ejemplo de dispositivo braille que permite tanto entrada de caracteres mediante un teclado convencional como la lectura de información mediante un display braille.



Figura 2.20 Línea Braille con teclado QWERTY

Además existe el anotador braille, un dispositivo electrónico pequeño y portátil que se utiliza para escribir y leer información en el sistema braille (figura 2.21). Este anotador incluye tanto voz sintetizada como una línea braille. Los jugadores con pérdida parcial o total de la visión y audición lo utilizan para escribir, leer y tomar notas. A pesar del nombre, los anotadores braille hacen mucho más que solo convertir texto a braille, ofrecen acceso a Internet, correo electrónico, reproducen archivos multimedia y cuentan con aplicaciones como calendario, reloj y lector de PDF.



Figura 2.21 Anotador braille de la marca Humanware

Software de dictado

El software de dictado convierte la palabra hablada en texto. También permite ejecutar comandos por voz del estilo “Pulsar enter”, o “Borra eso”. El software de dictado también se denomina “Speech-To-Text” o software de reconocimiento de voz, y está disponible tanto en computadoras de escritorio como en teléfonos inteligentes y tabletas.

Este tipo de software es ampliamente utilizado por jugadores con discapacidades motrices, aunque los jugadores con dificultades en el aprendizaje como la dislexia también se ven beneficiados.

Magnificador o lupa digital

Este es un dispositivo muy simple, ayuda a ampliar el texto o imagen de una pantalla. Existen tipos de magnificadores que funcionan como lupas comunes, aumentando el tamaño de la pantalla sobre la que se coloca. Pero también existen otros tipos de magnificadores digitales que permiten utilizar una pantalla digital en conjunto con una cámara para ampliar el texto o imagen que se desea ver en detalle.

Los amplificadores permiten al jugador ajustar el color y otros modos de pantalla como el brillo, contraste y zoom, siendo sumamente útil para jugadores con daltonismo o sensibilidad a la luz. Los magnificadores actuales son portátiles, lo que permite llevarlos con uno y utilizarlos también para leer o realizar tareas cotidianas.

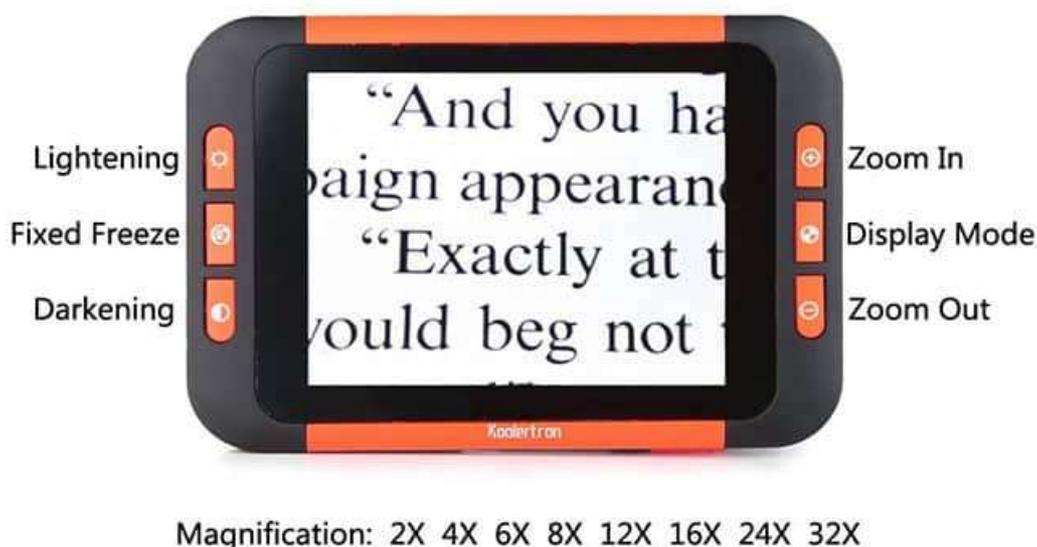


Figura 2.22 Magnificador digital

Software de ampliación de pantalla

El software de ampliación aumenta el tamaño del texto, las imágenes o los gráficos de la pantalla. Adicionalmente incluye otras funcionalidades útiles como aumentar el tamaño del cursor, mostrar la posición del cursor en la pantalla mediante guías o líneas, y otras opciones visuales como inversión de colores, ajuste de brillo, contraste y saturación.

Este software permite a los jugadores con dificultades visuales hacer un zoom sobre una parte de la pantalla y ver cosas pequeñas que de otro modo no podrían ver.

Los sistemas operativos de computadoras personales cuentan con una lupa integrada, además de otras opciones de accesibilidad, aunque también es posible instalar nuevas aplicaciones y software adicional.

Es importante notar que algunos videojuegos y consolas integran su propia versión del magnificador de pantalla. Esto facilita a los jugadores la utilización de este software, dado que está integrado naturalmente con el juego y no es necesario realizar combinaciones extras de acciones para activarla. Por ejemplo, el juego "The Last of Us: Part 2" incluye opciones de accesibilidad que permiten realizar un zoom a cualquier parte de la pantalla durante la partida.

Software de texto a voz

Este tipo de software denominado TTS (por sus siglas en inglés "Text-To-Speech"), o tecnología de "lectura en voz alta", está disponible en la mayoría de las computadoras y dispositivos móviles. Convierte el texto que se encuentra en documentos, páginas web y correos electrónicos en voz sintetizada, haciéndolos más accesibles para las personas ciegas, con dificultades visuales o disléxicas.

Este tipo de tecnologías es ampliamente usada en los videojuegos. Se puede utilizar para algo tan simple como agregar voz a las opciones de los menús, lo que permitiría a los usuarios con dificultades en la visión navegar los menús de una forma más simple. En los juegos multijugador online donde existe un chat para los jugadores, es importante también contar con esta tecnología para transformar en audio los mensajes enviados por los usuarios. Adicionalmente se puede utilizar para crear audio descriptivo de los eventos que se desarrollan durante el juego en pantalla. El costo de integrar TTS en un videojuego para crear accesibilidad es relativamente pequeño, sobre todo cuando se considera la cantidad de jugadores que se beneficiarían de esta herramienta.

Software lector de pantalla

Un lector de pantalla es un software que transforma el texto digital en audio, permitiendo a los usuarios con dificultades para leer el texto escrito escucharlo mediante la generación de una voz sintetizada. Este tipo de software hace mucho más que solo leer texto en voz alta, además de integrar un software TTS, permite al usuario navegar por varios tipos de interfaces, incluyendo páginas web y aplicaciones, siempre y cuando éstas implementen un soporte adecuado. Adicionalmente, en lugar de convertir texto a voz, también integran drivers que permiten comunicarse con dispositivos como una línea braille para mostrar el texto en este dispositivo de salida.

En una interfaz visual compatible, este tipo de software puede identificar todo tipo de estructuras de navegación como menús, encabezados, tablas, ventanas y alertas, elementos interactivos como menú desplegable, botones, enlaces, y otros elementos visuales como texto e imágenes, agregando la posibilidad de leer en voz alta estos elementos, así como también navegar entre ellos o interactuar mediante teclado y mouse.

Gran variedad de mecánicas dentro de los juegos no son apropiadas para ser accesibles mediante lectores de pantalla, pero cualquier elemento basado en texto es un candidato ideal, como los menús y HUD dentro del juego. Los juegos basados fuertemente en interfaces, como por ejemplo Football Manager o Hearthstone, son muy apropiados para ser accedidos mediante lectores de pantalla.

Por lo general los juegos diseñados para ser accesibles para jugadores con deterioro visual, se pueden lograr mediante una combinación de mecánicas de juego amigables con los jugadores ciegos, y además integrar un buen soporte para los lectores de pantalla en los menús. Skullgirls (2012) es un buen ejemplo de esto. Como es un juego de luchas, el gameplay en sí es naturalmente amigable con los jugadores ciegos, dado que es posible jugarlo enteramente sin ayudas visuales implementando un buen sistema de sonido. Además los menús son accesibles mediante lectores de pantalla.

Existen librerías que simplifican la tarea de interactuar con las APIs de los sistemas operativos de computadoras de escritorio, consolas y dispositivos móviles, y que facilitan la integración con lectores de pantalla.

Software de reconocimiento óptico de caracteres

El software de reconocimiento óptico de caracteres (OCR) convierte los documentos impresos en texto digital. Se trata de una tecnología basada en detectar letras, números y caracteres especiales individuales, para convertirlos en texto digital. Esta tecnología puede funcionar con imágenes y documentos escaneados o fotocopias. También puede convertir archivos PDF en documentos editables por un software procesador de texto.

El fundamento de esta tecnología de asistencia se basa en que el texto digital es un formato mucho más accesible que el texto impreso o una nota manuscrita, dado que permite al usuario personalizar el texto de la forma que necesite para leerlo más cómodamente.

En el caso de los videojuegos, los OCR se aplican por ejemplo en juegos que no ofrecen un soporte adecuado para lectores de pantalla. El software detecta el texto en pantalla, y agrega el soporte para poder traducir el texto, y hasta leerlo en voz alta usando la tecnología TTS. En la figura 2.23 se puede observar el software “Visual Novel Translator” utilizado en juegos del género de novelas visuales para detectar y traducir el texto del idioma original al inglés.

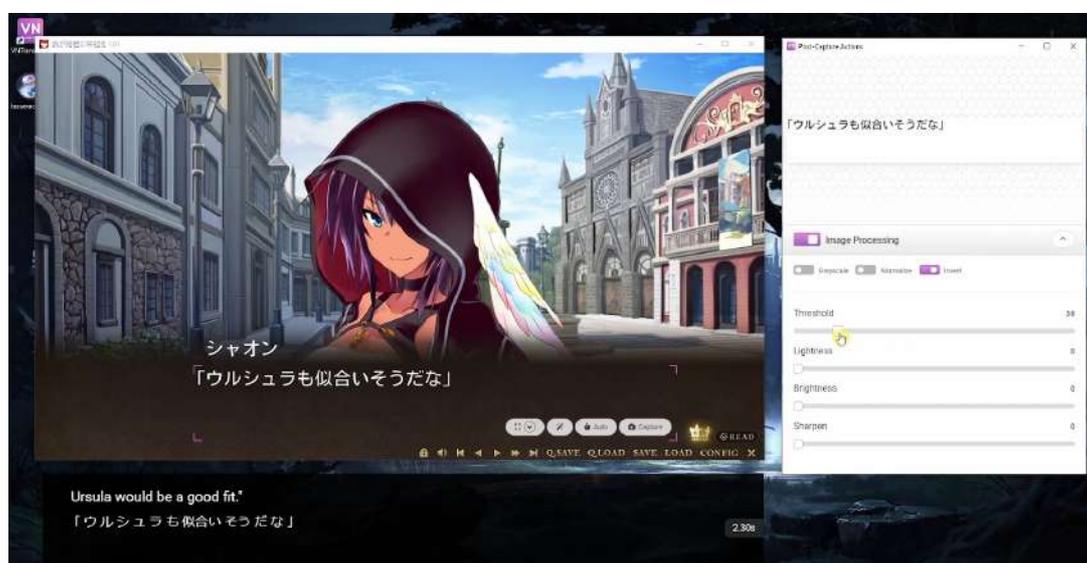


Figura 2.23 El software Visual Novel Translator permite detectar y traducir texto en juegos que no implementan soporte para lectores de pantalla.

2.8 Accesibilidad en sistemas operativos

Al analizar la accesibilidad en los sistemas operativos, podemos observar que los primeros sistemas operativos, como el DOS (Disk Operating System) de Microsoft y UNIX,

presentaban escasas barreras de accesibilidad para los usuarios de tecnologías de asistencia porque estaban basados en texto. Sin embargo, a medida que los sistemas operativos evolucionaron, y las interfaces gráficas ganaron popularidad y fueron adoptadas por todos los sistemas operativos, las barreras de accesibilidad aumentaron para las personas que no podían ver los gráficos o usar un mouse. Hoy en día los sistemas operativos más utilizados son Windows de Microsoft y Mac OS de Apple, y distribuciones de Linux como Ubuntu, Arch, Fedora, RedHat. En el caso de Linux, existe una sólida tradición de uso de la consola de comandos, y las tecnologías de asistencia que utilizan estas interfaces de líneas de comando han estado disponibles durante mucho tiempo. Sin embargo, se ha desarrollado una gran variedad de entornos gráficos de escritorio, como los mencionados, y en consecuencia, se han levantado barreras para los usuarios que no usan el mouse y/o que tienen algún tipo de discapacidad visual.

Una función de las interfaces gráficas de los sistemas operativos es proporcionar una API a los programadores para que puedan escribir aplicaciones compatibles con el entorno operativo. Es importante que dicha API proporcione soporte para la accesibilidad. Por ejemplo, todos los menús y controles en una interfaz gráfica de usuario deben ser accesibles a través del teclado, no solo del mouse, y deben mostrarse con una tipografía y esquema de color que el usuario pueda personalizar fácilmente. Siempre que la API proporcione los medios para ofrecer estas y otras características de accesibilidad, los desarrolladores de aplicaciones de software pueden construir aplicaciones para dicho sistema operativo que sean accesibles.

Actualmente el soporte de accesibilidad de las APIs ofrecidas por los sistemas operativos es significativamente dispar. Por ejemplo, Microsoft abordó los problemas de accesibilidad de la API de Windows desde el principio y proporcionó a los desarrolladores las herramientas para desarrollar aplicaciones que fueran accesibles. La mayoría de las aplicaciones para Windows, se pueden operar completamente a través del teclado, es decir no requieren el uso del mouse. Microsoft también desarrolló un estándar mediante el cual las aplicaciones pueden comunicarse de manera efectiva con las tecnologías de asistencia. Este estándar denominado Microsoft Active Accessibility (MSAA), está disponible desde Windows 95. Estos primeros esfuerzos para respaldar la accesibilidad, combinados con el dominio del mercado de Windows, llevaron a que se desarrollara una cantidad desproporcionada de tecnologías de asistencia para Windows.

Con el lanzamiento de Mac OS X, Apple mejoró considerablemente la accesibilidad de su sistema operativo. Por ejemplo, en la actualidad es posible acceder operar la interfaz gráfica a través del teclado. Adicionalmente Apple integró la accesibilidad en su API Carbon, favoreciendo que las aplicaciones de Mac OS X se comuniquen de manera efectiva con las tecnologías de asistencia.

A pesar que varias tecnologías de asistencia y otras funciones de accesibilidad están incorporadas en los sistemas operativos, por lo general solo ofrecen un nivel mínimo de accesibilidad, y de esta manera no se puede asegurar un acceso equitativo al sistema operativo y sus aplicaciones.

Las siguientes son funciones comunes de accesibilidad integradas en todos los sistemas operativos:

- Personalización del teclado para que los usuarios puedan ajustar el comportamiento del teclado, permitiendo:
 - presionar una única tecla en lugar de realizar una combinación de teclas.
 - usar el teclado para controlar el movimiento del mouse.
 - cambiar el tiempo que toma registrar una pulsación de tecla.
 - escuchar la pulsación de teclas.
- Personalización del mouse, para ajustar:
 - el tiempo que toma registrar un doble click.
 - la velocidad del cursor en pantalla.
 - la configuración del botón principal del mouse.
- Personalización de la pantalla, para que los usuarios puedan controlar el contraste de la pantalla, el estilo y tamaño de las fuentes, el tamaño de los íconos, y otras características de la pantalla.
- Alertas alternativas en forma de destellos de pantalla para usuarios que no pueden escuchar otro tipo de alertas.

Además de estas funciones básicas de accesibilidad, tanto Windows como Mac OS incluyen un software básico de ampliación de pantalla (Magnifier y Zoom respectivamente). Además Windows proporciona un screen-reader básico llamado Narrator, y Mac OS incluye capacidades limitadas de texto a voz a través de su función de reconocimiento de voz, "PlainTalk". Cada uno de estos productos proporciona un nivel básico de acceso, pero estas aplicaciones están muy por debajo de las aplicaciones de ampliación de pantalla y lectura de pantalla más completas que están disponibles para estos sistemas operativos a través de desarrolladores externos.

Linux a diferencia de Windows y Mac OS es un sistema operativo de código fuente abierto que cuenta con el apoyo de una comunidad global dedicada al desarrollo del sistema operativo, distribuciones y tecnologías. Hasta la fecha, la comunidad de desarrolladores de Linux ha producido un conjunto básico de funciones de accesibilidad, así como una aplicación combinada de lector y ampliador de pantalla, software para salida de texto en Braille, y un teclado en pantalla. Cada uno de estos productos fue desarrollado para el popular escritorio GNOME, un entorno de interfaz gráfica que se ejecuta tanto en Linux como Unix ("Ubuntu Desktop Guide - Accessibility", 2022).

En cuanto a las funcionalidades de accesibilidad en los sistemas operativos de escritorio, no se encuentra una diferencia notable entre las capacidades de los sistemas operativos Windows y Apple. Las tecnologías de asistencia están integradas desde la base en estos sistemas operativos y en todos los dispositivos que los utilizan, y las empresas creadoras de software compiten para hacerlos más útiles y flexibles (Szopa, 2021). En la mayoría de los casos la accesibilidad es una consideración primordial de las empresas desde las primeras etapas del diseño del producto hasta su lanzamiento. Ambos sistemas operativos ofrecen soluciones que incluyen: reconocimiento de voz, alertas sensoriales, audio monocanal, subtítulos personalizables, entre otras. Linux por otra parte, si bien puede ser considerado un sistema operativo complicado de utilizar, puede cumplir las expectativas y necesidades de las personas gracias a sus capacidades técnicas. Por un lado Linux tiene una gran capacidad de personalización, es estable y robusto, y puede ejecutarse en hardware antiguo con más facilidad que otros sistemas operativos (Bhalla, 2021). Además de esto, existen múltiples distribuciones de Linux disponibles a los usuarios, incluyendo algunas orientadas a mejorar la accesibilidad del sistema como por ejemplo la distribución "Accessible-Coconut", que está diseñada para ser utilizada por personas con dificultades visuales. Adicionalmente, existen múltiples soluciones de accesibilidad disponibles de forma gratuita para sistemas operativos linux compatibles, como por ejemplo lectores de pantalla, magnificadores de pantalla, creadores de macros y atajos de teclado, teclado en pantalla, entre otros.

Capítulo 3: Industria independiente y accesibilidad

En el capítulo anterior se trató el concepto de la accesibilidad en general y también en el ámbito de los videojuegos. Se describen sintéticamente datos referidos a discapacidad a nivel global y local, poniendo en evidencia la existencia de una población importante de jugadores alrededor del mundo y dentro de Argentina que vive con algún tipo de diversidad funcional. A su vez, se identificaron las discapacidades más comunes que afectan a la población de jugadores y las tecnologías disponibles que utilizan para poder interactuar con el mundo digital. Este capítulo se propone dar a conocer el estado actual de la industria del videojuegos a nivel global y local. Para poder comprender el futuro a corto plazo de la industria del videojuego, es útil conocer el estado actual de la misma, y analizar los cambios y tendencias más fuertes que se están llevando a cabo en la actualidad. Para ello, se describirán las características que identifican a un estudio independiente de producción de videojuegos, y las dificultades a las que puede enfrentarse al realizar juegos accesibles.

3.1 Estado actual de la industria del videojuego

Dentro de la industria del cine, la película “Avengers: Endgame” (2019) ha sido la que más ha recaudado en su fin de semana de estreno, alcanzando una cifra de 357 millones de dólares (Box Office Mojo, 2022). Para realizar una comparación con la industria del videojuego podemos analizar el caso del juego “GTA V” (2013), un juego de acción de mundo abierto lanzado por Rockstar Games, el cual generó una recaudación de 815 millones de dólares en el primer día de venta (Makuch, 2013). Además, “GTA V” alcanzó actualmente una recaudación total estimada de 6 mil millones de dólares (Donnelly, 2018), mientras que la película de ciencia ficción “Avatar” (2009), la que más ha recaudado a lo largo del tiempo, logró alcanzar únicamente 2,8 mil millones de dólares de recaudación (Box Office Mojo, 2022). Esto significa que un videojuego exitoso tiene igual o más poder de recaudación que una película taquillera, y no solo en su estreno sino que de forma permanente. El juego “GTA V” introdujo un modo online “GTA: Online” el cual funciona bajo la modalidad de suscripción, y permite a los usuarios jugar a través de Internet con otros jugadores alrededor del mundo, al mismo tiempo que reciben nuevo contenido de manera periódica, mientras se mantenga la suscripción. Para el estudio Rockstar esto significa un modelo de negocio que permite el ingreso de dinero no solo con la venta del juego, sino con las suscripciones mensuales, lo que permitió alcanzar la cifra de 6 mil millones de dólares de ganancia. Esto prueba no solo la importancia que tiene la industria del videojuego en la

actualidad, sino también el alcance y el impacto que pueden tener un único juego AAA³ sobre los jugadores como fenómeno cultural a nivel global.

Presente y futuro

Como se describió en el capítulo anterior, la cantidad de videojugadores a nivel mundial crece año a año. Una estimación realizada por NewZoo (Newzoo Global Games Market Report, 2022) indica que en el año 2022 se superará la marca de 3,1 mil millones de jugadores en todo el planeta, un número impulsado por la cantidad de nuevos jugadores dentro del mercado de dispositivos móviles, y por el crecimiento de algunos mercados como el de Latinoamérica y del sudeste asiático (Wijman, 2020).

La industria del videojuego a nivel global también creció enormemente en los últimos años. NewZoo estima que en el año 2019 el valor de la industria del videojuego, medida en ganancias generadas, fue de 151,55 mil millones de dólares. Para el año 2020 el valor de la industria fue de 179,1 mil millones, en el año 2021 fue de 192,7 mil millones, y para el año 2022 se estima que el valor de la industria será el valor más grande hasta el momento, con 203,1 mil millones de dólares de ganancias (Mason, 2022). De acuerdo a pronósticos realizados por expertos de “Mordor Intelligence”, se estima que para el 2025 el valor de la industria de los videojuegos alcanzará los 256,97 mil millones de dólares, con un crecimiento estimado del 9,17% con respecto al año 2020 (Mordor Intelligence, 2020).

Como ejemplo de esta tendencia, se puede señalar el fuerte crecimiento de la industria del videojuego durante la pandemia generada por el coronavirus Covid-19, a diferencia de otros sectores productivos afectados negativamente,, estimando que seguirá creciendo en los próximos años. De acuerdo a Statista (Von Kameke, 2022) y NewZoo (Wijman, 2019), la región de Asia y el Pacífico es la que genera mayores ganancias a nivel mundial. Se estima que en el 2022, las ganancias generadas solamente por esta región fueron de 96,3 mil millones de dólares. Otro dato importante salido de la misma estimación es la fluctuación del mercado en la región de norteamérica, la cual se espera que aumente un 6,6% con respecto al año anterior, llegando a un valor de ganancias de 54,3 mil millones de dólares.

³ AAA (triple A) es una clasificación informal utilizada para referirse a los videojuegos producidos y distribuidos por un editor o distribuidora importante, típicamente teniendo un nivel de marketing y desarrollo de alto presupuesto.

Mercado Global de Videojuegos (2022) En miles de millones

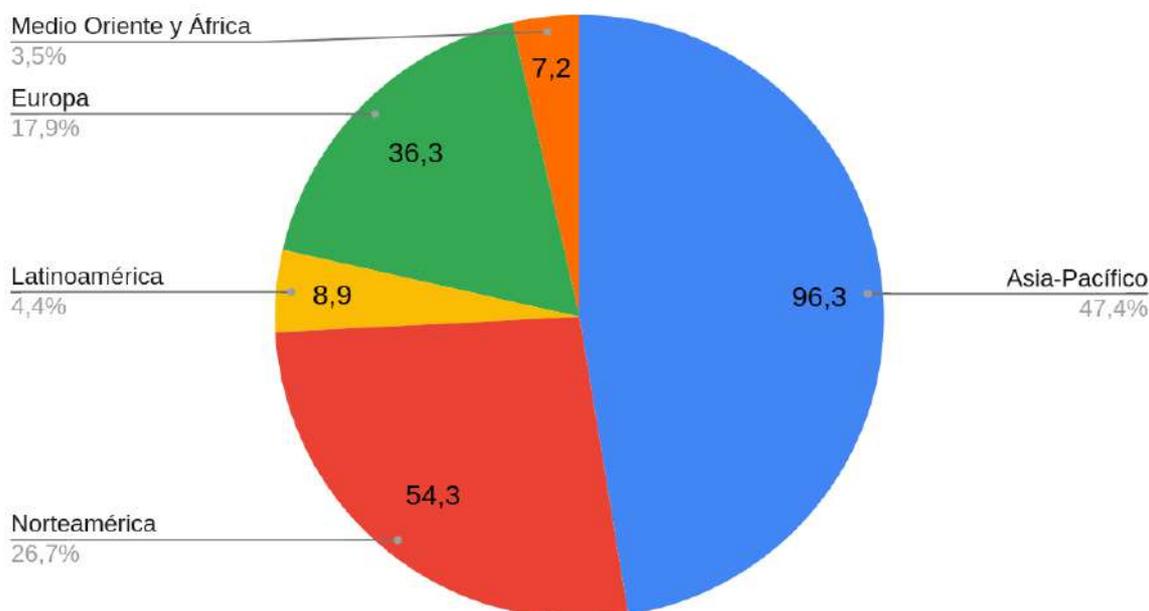


Figura 3.1 Mercado Global de Juegos del año 2022 por región. Adaptado del informe de NewZoo.

En la figura 3.1 se muestra un estimado de la distribución de ganancias generadas por las diferentes regiones del mundo en el año 2022. La región con más mayor peso es la de Asia y el Pacífico, seguida por la de Norteamérica. El informe para el año 2022 indica que la región con mayor crecimiento durante este año fue la de Medio Oriente y África, seguida en segundo lugar por la región Latinoamericana, con un aumento del 10% con respecto al año 2021. Según el informe, las ganancias generadas por el mercado de los videojuegos en dispositivos móviles se espera llegue a un total de 103.5 mil millones de dólares en el 2022. Aunque el mercado de los dispositivos móviles muestra un crecimiento en 2022, es notablemente más bajo que el crecimiento con respecto a los años 2020-2021, dado que es difícil competir con el crecimiento de los años anteriores provocado por la pandemia.

A pesar de los problemas con respecto a las restricciones en las cadenas de suministros que sufrieron las consolas durante el último tiempo (Boutighmass, 2022), se espera que los juegos de consola generarán 58,6 mil millones de dólares de ganancia en el 2022, mostrando un crecimiento del 8,4% con respecto al año anterior. Según el reporte de mercado de NewZoo, los juegos de navegador van a declinar su presencia, a medida que los jugadores cambian sus hábitos de juego y gastos hacia el mercado de dispositivos móviles, y a medida que los desarrolladores priorizan los desarrollos para móviles. En cuanto al mercado de los juegos en PC en China, se espera que descienda su presencia con el correr de los años, acelerándose hacia el 2024, como resultado directo de la represión del gobierno chino a los videojuegos (Wijman, 2022).

Para el futuro se espera que el desarrollo de la industria de los videojuegos siga creciendo, proyectando una ganancia de 222,6 mil millones de dólares hacia el 2024. Este fuerte crecimiento está impulsado por la expansión de los servicios de suscripción, una modalidad que están eligiendo cada vez más los jugadores de consolas y PC. Las compañías que antes priorizaron el desarrollo de sus juegos para el mercado de PC y consolas, comenzaron a llevar sus productos al mercado de los móviles, como en el caso de “Apex Legends Mobile” y “Diablo Immortal”. Adicionalmente cada vez son más los usuarios de dispositivos móviles que se introducen al mercado, a través de juegos progresivamente más complejos e inmersivos (Baker, 2021). En la figura 3.2 se puede observar cómo el mercado de videojuegos de dispositivos móviles (juegos en celulares y juegos tablets) en conjunto ocupan un 51% del total del mercado, más que los juegos en PC y consolas.

Mercado Global de Videojuegos (2022) En miles de millones

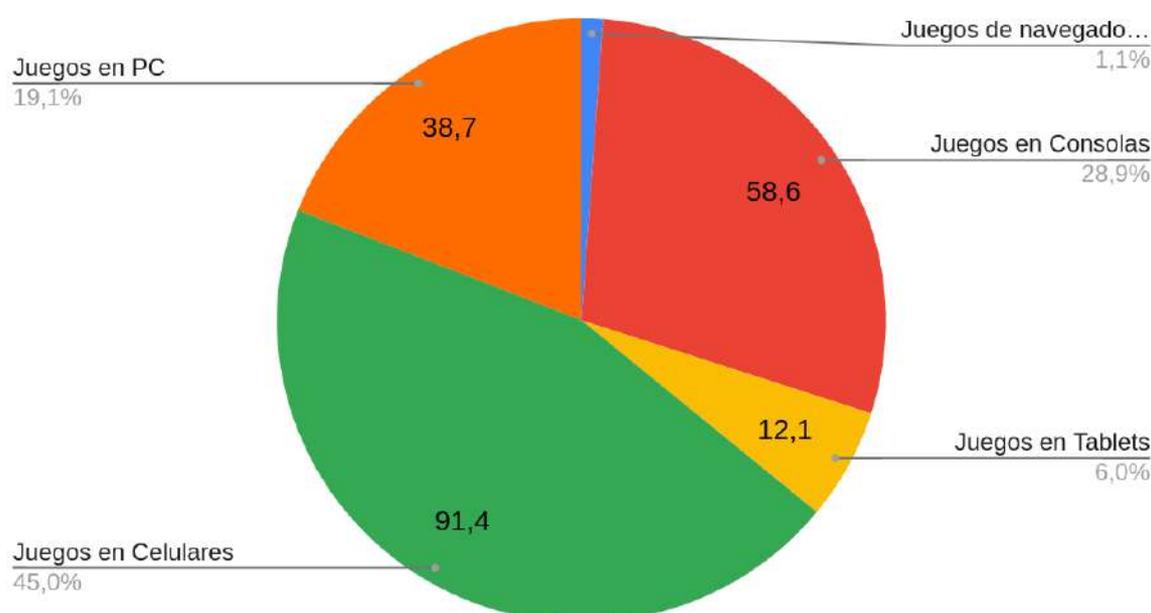


Figura 3.2 Mercado Global de Juegos del año 2022, por segmento. Adaptado del informe de NewZoo.

Tendencias del mercado global

Cuando se cancelaron los eventos deportivos que se transmitían en vivo debido a la pandemia provocada por el COVID-19, muchas personas recurrieron a los deportes electrónicos, también conocidos como E-Sports. Estos son videojuegos competitivos, que pueden estar relacionados a un deporte o no, y en donde las competencias son por un premio en efectivo. Los jugadores compiten entre sí en línea o en eventos para espectadores en grandes estadios y arenas y se ha convertido en una industria de miles de millones de dólares con importante crecimiento año a año. Según Statista (eSports market

revenue worldwide from 2020 to 2025, 2022), los ingresos globales de los E-Sports están valorados en 1,13 mil millones de dólares para el año 2021 y se prevé un crecimiento a 1,86 mil millones para 2025. Para convertirse en un atleta, ya sea en un deporte real o en un E-Sport es importante la dedicación, el entrenamiento, y la superación de obstáculos. Ya sea como jugadores o simplemente como espectadores, muchos jugadores con diversidad funcional se acercaron a este tipo de juegos (Disability Horizons, 2019). Si bien puede parecer que los jugadores con discapacidad tienen más desventajas que la mayoría de jugadores, los E-Sports les han brindado una forma de competir contra personas de todo el mundo, y medirse competitivamente de igual a igual, dado que durante las partidas las discapacidades no importan, solo la habilidad para jugar. Muchos de los juegos de E-Sport más grandes se ocupan de brindar opciones de accesibilidad en sus juegos, algo que jugadores como Rocky Stoutenburg⁴ agradece dado que le permite jugar Fortnite mediante su dispositivo QuadStick.

Durante el año 2020, un número cada vez mayor de videojuegos se volvieron más inclusivos con diferentes culturas y estilos de vida, y es probable que esta tendencia continúe. “Marvel’s Spider-Man: Miles Morales” (2020), desarrollado por Insomniac Games, presenta como personaje principal un adolescente afrolatino, y “Hades” desarrollado por Supergiant Games, tiene un elenco de dioses griegos étnicamente diversos. Además, videojuegos como “The Last Of Us: Part 2” y “Tell Me Why” tienen representación LGBTQ+ en sus personajes e historia. Adicionalmente la diversidad de jugadores se ve cada vez más representada dentro de los juegos. Un ejemplo de esto es la inclusión del personaje Hailey Cooper en el juego “Spider-Man: Miles Morales”, quien es sorda y además se comunica con Spider-Man mediante un lenguaje de señas realista, como se puede observar en la figura 3.3.

⁴ Jugador parapléjico y streamer de 35 años ganador de varios Record Guinness



Figura 3.3 En el juego “Spider-Man: Miles Morales” (2020), existe un personaje sordo, y se comunica mediante lenguaje de señas.

Según Empirical Software Engineering (Bezemer et al, 2018) alrededor del 15% de los juegos en Steam, son de “acceso anticipado” y esta tendencia continuará a lo largo del 2021 y más allá. La modalidad de acceso anticipado o “early access” significa que los jugadores pueden probar juegos antes de su lanzamiento oficial, y los desarrolladores de los juegos pueden utilizar los comentarios y feedback de los jugadores para modificar y mejorar el juego. “Rimworld” (2018) se convirtió en uno de los juegos mejor calificados en Steam en gran parte gracias a una estrategia de acceso anticipado. El modelo de acceso anticipado es por lo general mucho más atractivo para los desarrolladores individuales o para estudios independientes pequeños. Según Empirical Software Engineering, 88% de los juegos de acceso anticipado corresponden a juegos del tipo independientes (Bezemer et al, 2018).

Los juegos en la nube son una nueva solución para los jugadores frustrados por las horas de espera al descargar un nuevo videojuego, la falta de hardware que cumpla los requerimientos mínimos para poder jugar, o por problemas de almacenamiento en su PC o consola. Los juegos en la nube o “cloud gaming” evitan estos problemas al transmitir el juego en tiempo real a través de la red, desde un equipo en donde se ejecuta realmente el juego, hasta el dispositivo en donde lo termina utilizando el jugador. Los servicios de streaming como “PS Now” brindan a Sony y a las compañías de videojuegos un flujo de ingresos constante y una forma de monetizar videojuegos antiguos. Esta tendencia puede suponer una alternativa y una mejora en la cantidad de juegos disponibles para todos los jugadores, incluyendo los jugadores con algún tipo de discapacidad. La compañía “Blacknut” (<https://www.blacknut.com/>), por ejemplo, trabaja en conjunto con expertos en

accesibilidad para asegurarse de que todas las interfaces de sus aplicaciones de cloud gaming sean accesibles para todos los jugadores (Video games and handicap: Blacknut and HitClic (Handigamers) are committed to the accessibility of cloud gaming, Blacknut Website, 2021). Para la industria de videojuegos independientes, el cloud gaming también parece ser una tendencia que promete mejorar el alcance y las ganancias a corto y medio plazo. Por ejemplo, Microsoft está lanzando una solución de cloud gaming, y apunta a que más desarrolladores de juegos independientes comiencen a utilizar y lanzar sus juegos a través de esta plataforma (Wilde, 2022). Algunos estudios no están de acuerdo con este nuevo lugar que ocuparían los juegos independientes dentro del cloud gaming, fundamentalmente por cuestiones contractuales que atarían a los estudios y juegos a la plataforma de lanzamiento, limitando las libertades de poder lanzar el juego a través de otras plataformas de distribución como por ejemplo en Steam. Microsoft, Google y Amazon, los dueños de la infraestructura que soporta estos servicios, son los que cobran a los jugadores por el acceso al servicio, y no por la compra del juego en sí, por lo que las ganancias se van centralizar todavía más en las manos de las plataformas en lugar de los estudios independientes (Gordon, 2021).

Es bastante común el estereotipo entre los jugadores de videojuegos, en donde se los ve siempre como jugadores jóvenes y varones. Si bien es cierto que muchos jóvenes juegan videojuegos, la realidad es que actualmente las mujeres representan casi la mitad de los jugadores en todo el mundo (Distribution of video gamers in the United States from 2006 to 2022, by gender, Statista, 2022), y la cantidad de jugadores adultos mayores también está creciendo. La diversidad de tipos de juegos disponibles y la pandemia del COVID-19 han contribuido a esta creciente y cambiante demografía de los jugadores (Nielsen, 2020). Aunque los jóvenes juegan más que nunca, los millennials (18-34), la generación X (35-54) y los adultos mayores también están jugando. En 2022, la "Entertainment Software Association" encuestó a la población de Estados Unidos en relación a los videojuegos, e informó que la edad promedio de un jugador es de 33 años, y que el jugador promedio estadounidense ha jugado por al menos 14 años (Nielsen, 2020). Según los datos de la encuesta compilados por Statista (2021), el 20% de los jugadores de videojuegos en todo el mundo son hombres entre 21 y 35 años. Además indica que en los Estados Unidos, las mujeres representan el 45% de los jugadores y que los juegos son una fuente de entretenimiento cada vez más popular para los adultos mayores. AARP, una organización estadounidense sin fines de lucro que atiende las necesidades e intereses de las personas mayores de 50 años, realizó una encuesta en línea en 2019 entre aproximadamente 1600 adultos mayores de 50 años que juegan videojuegos al menos una vez al mes (Brittne & Nelson-Kakulla, 2019). Estos jugadores mayores informaron que juegan por diversión, para reducir el estrés y para mantenerse mentalmente activos. El 73% de estos jugadores

mayores usan dispositivos móviles, en lugar de computadores, y la mayoría prefiere los juegos de puzzles en lugar de juegos de disparos o de acción. De los encuestados, el 55% dijo que los juegos tenían un impacto positivo en su bienestar, el 40% dijo que no tuvo ningún impacto, mientras que el 5% informó un impacto negativo. Esta encuesta, aunque no es representativa de la población en general de los adultos mayores, ofrece una pista sobre algunas de las tendencias de los jugadores mayores que merecen atención. Los videojuegos pueden ser una herramienta poderosa para mejorar la conectividad cerebral, mejorar el tiempo de respuesta, la atención, la navegación espacial e incluso la memoria de corto plazo en adultos mayores (Anguera y Gazzaley, 2015).

Tanto por la expansión del mercado de videojuegos hacia los dispositivos móviles, con pantallas mucho más pequeñas de lo que estamos acostumbrados para un videojuego, como por el cambio demográfico y etario de la población de jugadores, es necesario reconocer estas tendencias y abordarlas en relación a la accesibilidad. En el Capítulo 4 se analizará cómo se puede mejorar la accesibilidad en este tipo de juegos y para este tipo de jugadores.

El estado actual de la industria en Argentina y Latinoamérica

En América Latina y el Caribe existen 397 millones de jugadores y el 80% de ellos se concentra en México, Brasil, Argentina, Colombia y Venezuela (Banco Interamericano de Desarrollo, 2019). Nuestra región se ha convertido en la segunda con mayor incremento en la industria, con un índice de crecimiento anual esperado de 10% (Newzoo, 2020). En la actualidad esta industria goza de uno de los más altos crecimientos en la historia del entretenimiento, y representa una oportunidad para poner el talento de América Latina en el mapa global.

Desde el 2018, la Universidad Nacional de Rafaela a través del Observatorio de la Industria Argentina de Videojuegos realiza un informe que reúne información acerca de la industria del videojuego en Argentina (Informe General del Observatorio de la Industria Argentina de Desarrollo de Videojuegos, UNRaf 2021). En el último informe se indica que la industria argentina de videojuegos recaudó 86,7 millones de dólares durante el año 2021, unos 10 millones más que en el informe del año 2020. Adicionalmente, de las 84 empresas que participaron del informe, 71% considera que su recaudación final será mayor que en el año anterior.

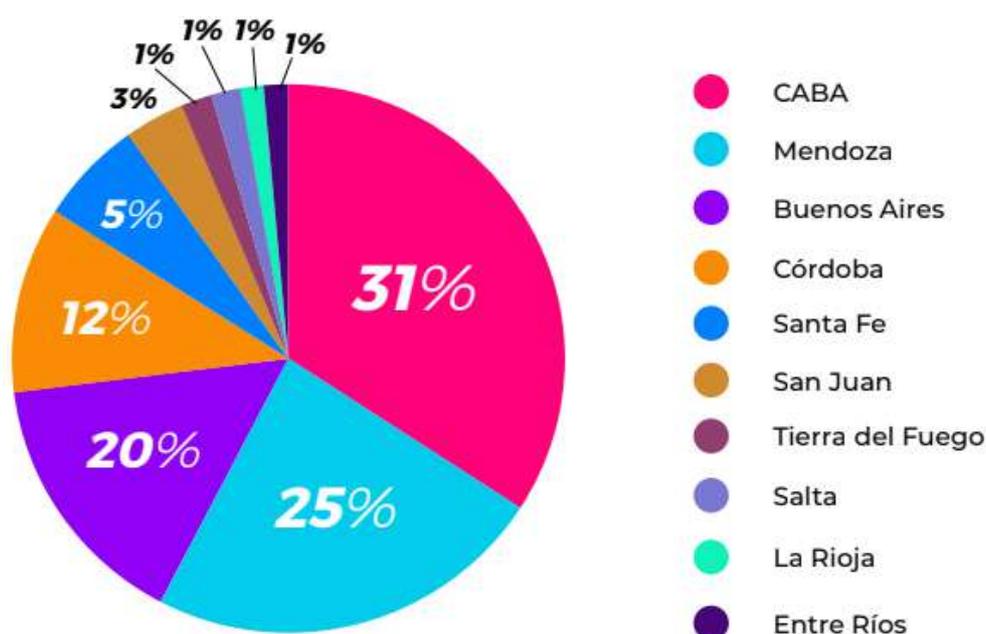


Figura 3.4 Distribución de los encuestados por localización. Informe del año 2021 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

En la figura 3.4 se puede observar la distribución de los encuestados según la locación del estudio de desarrollo. La mayor parte del desarrollo nacional se lleva a cabo dentro de la provincia de Buenos Aires con 51% (31% en la Ciudad Autónoma de Buenos Aires y 20% en el resto de la provincia), seguido por la provincia de Mendoza (25%), y la provincia de Córdoba (12%).

Alejandro Iparraguirre, coordinador de videojuegos del Ministerio de Cultura de la Nación, indica que el 90% de la facturación, dentro de la industria argentina, se concentra en cuatro empresas (Early Access, 2022). El resto de la industria está formada por estudios medianos de hasta 50 personas y miles de desarrolladores independientes o estudios de hasta 5 personas que son monotributistas (Delfino, 2022).

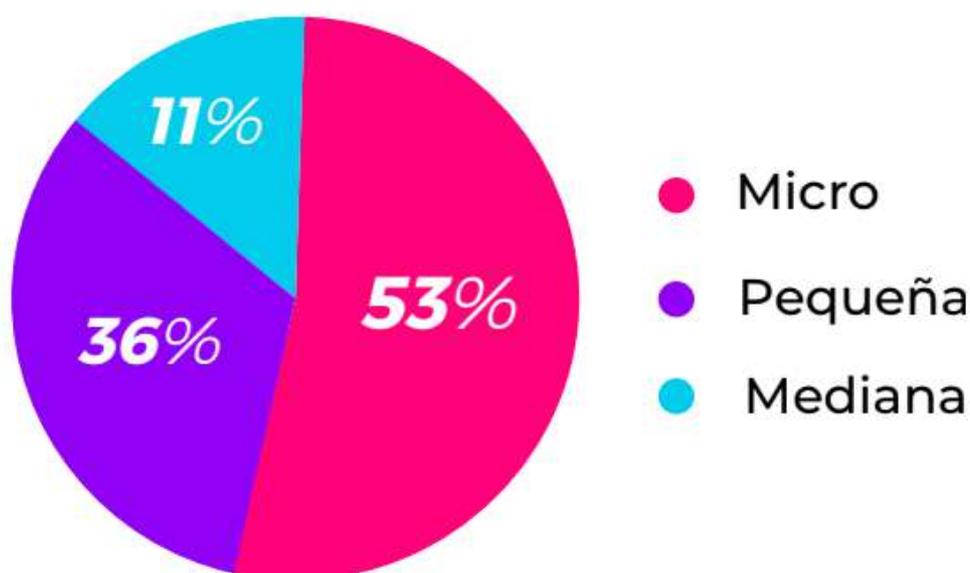


Figura 3.5 Tamaño de las empresas según la clasificación PYME de la AFIP en categoría Servicios por número de empleados. Informe del año 2021 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

En la figura 3.5 se describe la distribución de las empresas que participaron del mencionado informe, de acuerdo al tamaño categorizado por la AFIP en el rubro Servicios de las Pequeñas y Medianas Empresas (PyMEs). El informe contabiliza empleados en relación de dependencia y freelancers en modalidad full-time. Más de la mitad de las empresas dentro de la industria (53%) pertenecen a la categoría “micro”, es decir que cuentan con una cantidad de empleados menor a 7 personas. En segundo lugar (36%) tenemos a las empresas “pequeñas” con hasta 30 empleados, y en tercer lugar (11%) empresas medianas de hasta 165 personas (¿Qué es una MiPyME?, Argentina.gob.ar, 2022).

Las cuatro empresas con mayor presencia dentro de la industria del videojuego en Argentina son: Globant, Nimble Giant, Etermax y Wildlife. Globant desarrolla videojuegos para múltiples plataformas, y trabaja junto a empresas muy grandes dentro de la industria del entretenimiento a nivel global como Electronic Arts y Disney. Nimble Giant creó videojuegos con mucho impacto como “Quantum League” (2020) y “Hellbound” (2020). Etermax tiene múltiples productos lanzados, siendo el más exitoso el videojuego “Preguntados” (2013). El último estudio es el de Wildlife, fundado originalmente en Brasil, pero ingresó al mercado argentino durante el año 2020.

Cuando analizamos los datos ofrecidos por el informe del Observatorio de la Industria Argentina de Videojuegos, se observa que la tendencia es similar a la que se desarrolla a nivel global. Como ejemplo de ello, a nivel global la industria está comenzando a ser impulsada por nuevos jugadores atraídos por juegos móviles, por lo general del género

“casual” (Gu, 2022). En la figura 3.6 se observa que más de la mitad de los videojuegos desarrollados en Argentina en el 2021 (53%) pertenecen a la categoría de juegos “casuales”.

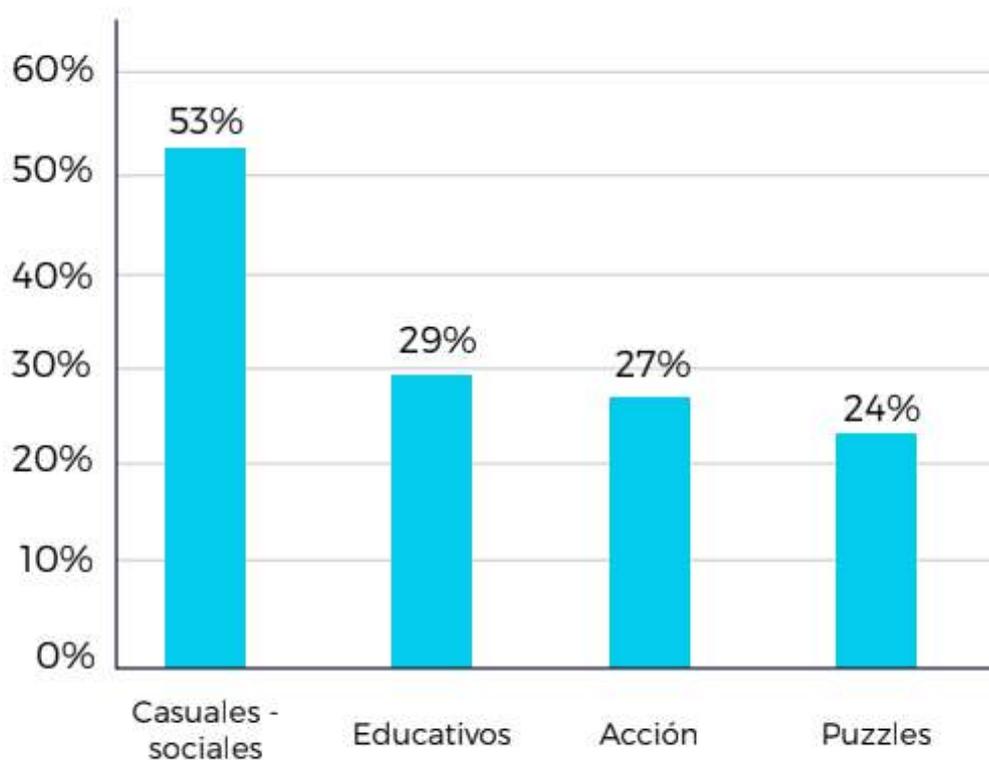


Figura 3.6 Tipos de videojuegos desarrollados. Informe del año 2021 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

Por otro lado, en el informe se refleja la fuerza que poseen los desarrollos para dispositivos móviles: en Argentina, 74% de los videojuegos son desarrollados para dispositivos móviles y en segundo lugar, el 71%, son para computadoras de escritorio (PC y Mac). En la figura 3.7 se puede observar una comparativa entre las plataformas elegidas para el desarrollo de los videojuegos en los años 2020 y 2021: los juegos para móviles adquieren mayor importancia en el año 2021, incluso sobrepasando a los juegos para computadoras.

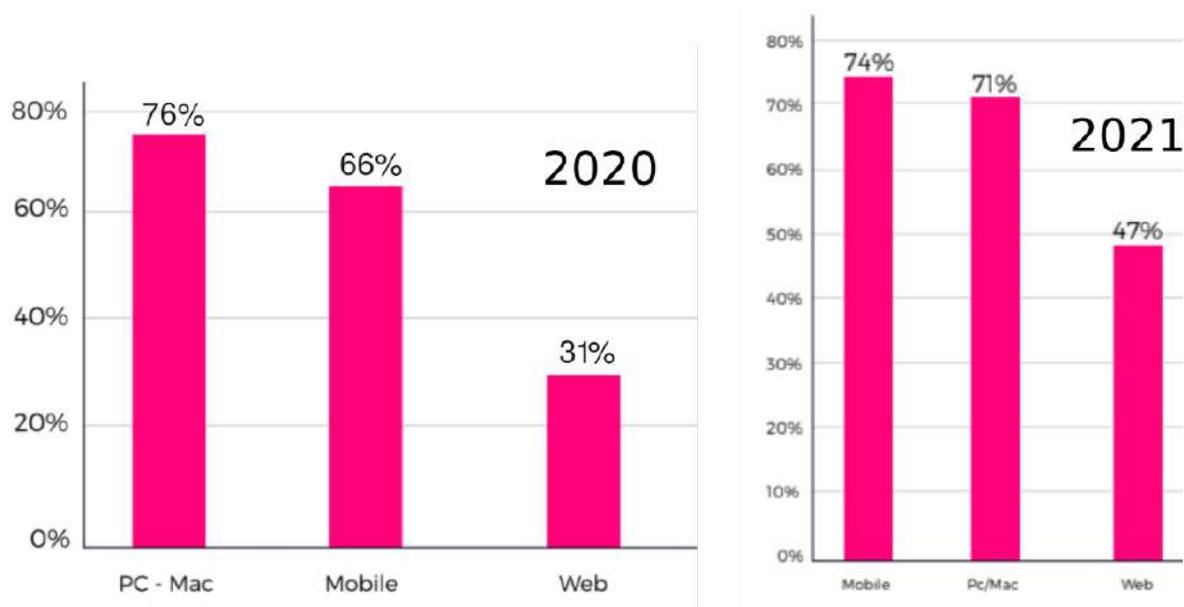


Figura 3.7 Comparativa de las plataformas para las cuales se desarrolla (2020-2021). Informes del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

En síntesis, se puede concluir que la industria argentina de videojuegos crece todos los años de manera constante. Según el informe del Observatorio de la Industria Argentina de Desarrollo de Videojuegos, la cantidad de empresas y nuevos emprendimientos se duplicó en los últimos cinco años. Esto se puede visualizar en varios indicadores de la industria que vienen evolucionando desde hace años: la cantidad de personal que trabaja dentro del sector, la recaudación generada, y la presencia de la industria a nivel mundial, sobre todo dentro de los mercados norteamericanos y asiáticos.

Características del sector independiente de videojuegos

El término “juego indie”, generalmente utilizado de forma intercambiable con “juego independiente”, es probablemente uno de los conceptos relacionados a videojuegos más difíciles de definir formalmente. Aunque las malas interpretaciones constantes del término “juego indie” ha llevado a algunos autores a evitar el término por completo (Waren, 2014), desde hace años se observan discusiones con respecto a la utilización del término, en foros, sitios web y revistas. Garda y Grabarczyk (2016) proponen una definición que involucra tres tipos de independencias: financiera, creativa y editorial.

Independencia financiera

La independencia financiera está basada en la relación entre el desarrollador y el inversor. El inversor no solo puede referirse a compañías privadas o individuos que brindan su capital a los desarrolladores con el objetivo de obtener ganancias a futuro, sino que también incluye a organizaciones gubernamentales y otras instituciones que financian proyectos no

comerciales. Es importante notar que cuando nos referimos a la independencia financiera, nos referimos a la independencia por parte del desarrollador, y no de la compañía de la cual el desarrollador forma parte. Esto significa que incluso los juegos desarrollados por grandes compañías que no reciben ningún tipo de ayuda financiera externa, pueden ser financieramente dependientes, ya que el equipo que desarrolla el juego depende de los equipos financieros y de marketing que coexisten dentro de la misma compañía.

Independencia creativa

Por lo general la independencia creativa está dada por la presencia de la independencia financiera, pero no se debe confundir estos dos tipos. La independencia creativa hace referencia a la capacidad del equipo desarrollador de diseñar los juegos de manera que complazca los deseos propios. También se puede expresar como la idea de que el juego es independiente creativamente cuando el desarrollador es la audiencia objetivo. En otras palabras se puede hablar de control creativo cuando el jugador implícito es sinónimo del autor mismo (Aarseth, 2007). En este aspecto no hay una metodología que podamos aplicar para identificar cuando un juego es creativamente independiente, sino que debemos guiarnos por las declaraciones de los desarrolladores en el material promocional, entrevistas o descripciones del producto, en donde muestran su conformidad, o no, con el diseño creado. Podemos identificar casos en donde existe dependencia creativa por ejemplo, cuando la audiencia objetivo está marcada por el inversor. Por supuesto los juegos pueden también perder su independencia debido a decisiones autónomas del desarrollador, cada vez que se decida ajustar el juego a la demanda del mercado existente. Un buen ejemplo de esto es la práctica popular de crear clones de juegos con mucho éxito, especialmente en plataformas móviles (Martin and Deuze, 2009). Por otro lado, incluso si el juego cuenta con respaldo institucional, pero se originó a partir de decisiones autónomas del desarrollador, aún puede llamarse creativamente independiente. Un ejemplo de esto es el juego "Fez" (Polytron, 2011), el cual fue inicialmente apoyado por instituciones gubernamentales (Parkin, 2011).

Independencia editorial

Un editor es una compañía que se encarga de publicar (y en algunos casos distribuir) videojuegos desarrollados por un equipo de desarrollo de videojuegos. En el caso de la independencia editorial, el juego es independiente cuando el desarrollador es también el editor. A menudo si el juego es exitoso luego de su lanzamiento inicial, puede ser publicado por un editor externo, por ejemplo para llevar el juego a otra plataforma o para introducirlo al

proceso de distribución tradicional. En muchos casos la dependencia editorial y financiera se superponen, dado que el editor también toma el rol del inversor externo.

Este tercer tipo de independencia es especialmente importante, dado que el crecimiento reciente del número de juegos editorialmente independientes puede atribuirse al aumento de múltiples canales que fomentan y facilitan la auto-publicación. Desde la expansión de la conexión de Internet, se crearon nuevas formas que facilitan la publicidad y distribución de videojuegos sin la intervención de compañías especializadas. Adicionalmente, la tarea del testing y QA⁵ de los juegos también se facilitó al dejarlas a cargo de los jugadores, mediante plataformas como Patreon o “Early Access” en Steam (Clark & Wang, 2020).

En resumen, la noción de “juego independiente” puede explicarse mediante la disyunción de los tres tipos distintos de independencia mencionadas. Un juego puede pertenecer a uno, a dos, o a los tres tipos de independencia, pero para poder llamarse independiente tiene que pertenecer por lo menos a uno de los tipos (figura 3.8).

Independencia Financiera	Independencia Creativa	Independencia Editorial	Ejemplo
No	No	No	Assassin's Creed: Unity (Ubisoft Montréal, 2014)
No	No	Sí	Broken Age: Act 1 (Double Fine, 2014)
No	Sí	No	La serie de juegos Metal Gear Solid
No	Sí	Sí	Antichamber (Demruth, 2013)
Sí	No	No	Robbo (Pelc, 1989)
Sí	No	Sí	Angry Birds Rio (Rovio, 2011)
Sí	Sí	No	Braid (Number None, 2008)
Sí	Sí	Sí	Super Meat Boy (Team Meat, 2010)

Figura 3.8 Ejemplos de posibles combinaciones de independencia en un juego (Garda y Grabarczyk, 2016)

Un acercamiento empírico a la definición de “juego independiente”

De un estudio realizado sobre jugadores y desarrolladores se identificaron múltiples rasgos o características que, idealmente, debe poseer un juego para ser llamado “independiente” (Pereira, 2021). Los cinco rasgos más importantes que debe poseer un juego independiente son:

1. Debe ser autofinanciado.
2. No debe estar asociado con un editor.

⁵ Por las siglas en inglés de “Quality Assurance”.

3. Debe proveer al desarrollador control creativo total.
4. Debe ser desarrollado por un estudio “pequeño”.
5. Debe contar con un presupuesto “bajo”.

El primer rasgo o característica hace alusión a que los desarrolladores puedan financiar el proyecto completo sin intervención externa. El segundo hace referencia a la percepción negativa respecto de los lazos contractuales que se presentan en contratos con los editores, y cómo pueden afectar o restringir el desarrollo del juego. La tercera característica está relacionada con las anteriores, y se refiere a que el desarrollador no debe ser controlado creativamente por ningún tipo de influencia externa. Con respecto al rasgo de “estudio pequeño”, se puede referir a múltiples aspectos de una compañía: el número de empleados, tamaño físico de la estructura o lugar de trabajo, tamaño en el mercado, etc. Por último, el bajo presupuesto sigue una lógica similar, estableciendo un contraste entre el grado de inversión económica designada a los juegos independientes, comparada con el gran costo de los juegos desarrollados tradicionalmente en la industria.

Caracterización del sector de desarrollo de videojuegos en Argentina

Existen múltiples características dentro del sector del desarrollo de videojuegos en Argentina que lo acercan a la definición de “independiente”. Se puede advertir que la mayoría de los desarrollos dentro de la industria argentina actualmente poseen rasgos que, según la definición previamente enunciada, parecen agruparlos bajo el concepto de “juegos independientes”.

Con respecto a la financiación, según el informe del Observatorio de la Industria Argentina de Desarrollo de Videojuegos, el 66% del financiamiento utilizado por las empresas pertenecen a aportes del emprendedor, dueño o socio del estudio (figura 3.9). En este sentido, la mayor parte de la industria se financia a través de medios propios, es decir que los estudios de desarrollo son en su mayoría financieramente independientes. Recién en el segundo y tercer lugar se observan otros tipos de financiamiento como financiación mediante editores (9%) y subsidios del gobierno (6%).



Figura 3.9 Financiación de proyectos. Informe del año 2021 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

Por otro lado, del informe también se puede observar cuál es la fuente de financiamiento deseada por los estudios. La mayor parte de las empresas dentro de la industria (38%) desean lograr un financiamiento mediante un editor externo (figura 3.10). Esto nos puede indicar que una gran parte de la industria desea el apoyo de un editor, pero por el momento no lo consigue. Esto puede deberse a diversas razones, pero lo más probable es que se relacione con el deseo de lograr una estabilidad económica, algo que se puede alcanzar a través del trabajo en conjunto con un editor, quien puede solucionar varias de las problemáticas relacionadas a la publicación y distribución del juego, como así también facilitar la portabilidad del juego a nuevas plataformas o el ingreso del juego a la venta en tiendas físicas (Deuze & Martin, 2009). En segundo lugar, se puede observar que las empresas también desean poder financiar sus juegos de forma independiente. El porcentaje de empresas que desean lograr financiamiento propio es el 25%, un porcentaje importante del total y no tan lejano del porcentaje que desea trabajar junto a un editor. Estos números son importantes si comparamos esta métrica con la del informe del año 2020 (figura 3.11), en se puede observar que estos dos números están mucho más separados (48% y 21% respectivamente), lo que indica una tendencia en la industria en donde se busca lograr financiamiento propio.

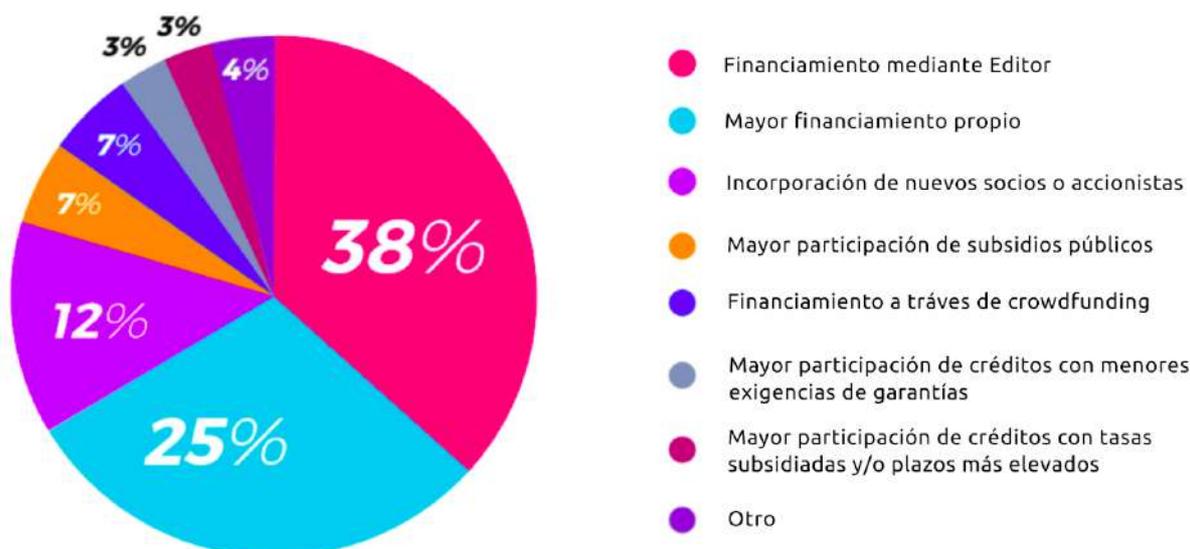


Figura 3.10 Financiación de proyectos deseada. Informe del año 2021 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

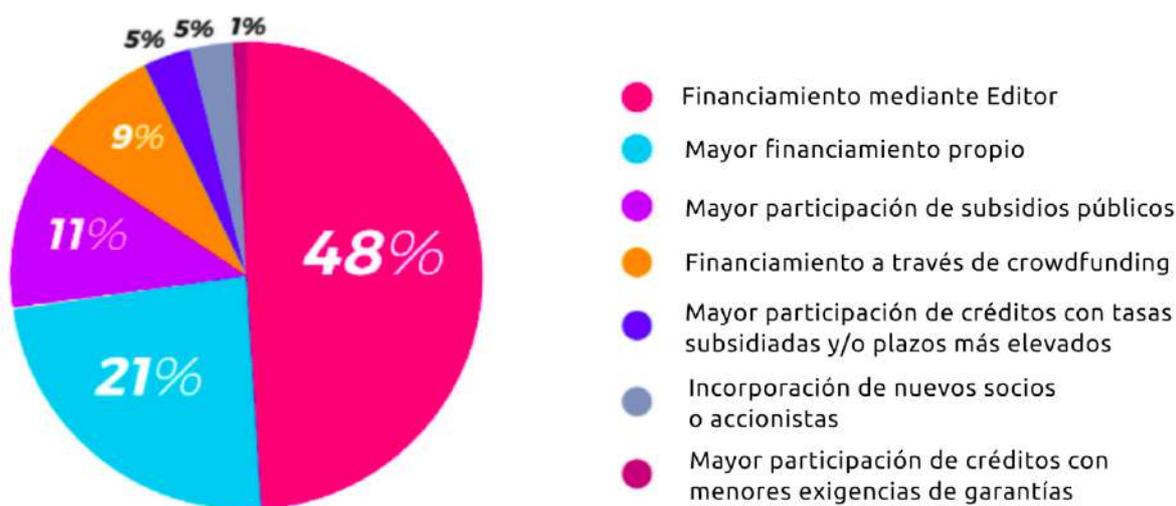


Figura 3.11 Financiación de proyectos deseada. Informe del año 2020 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

Acompañando a esta tendencia, el informe también menciona las actividades realizadas por los estudios que fueron encuestados (figura 3.12), de los cuales el 42% se dedica al desarrollo y venta de juegos propios, y comparado con el segundo grupo parece demostrar un nivel de independencia creativa mayor. En el segundo grupo se observa que las empresas proveen servicios de desarrollo de videojuegos a otras empresas (36%). De forma similar a los estudios de cine y música, los editores de juegos subcontratan cada vez

más la producción de sus juegos a los llamados estudios “third-party”, que la mayoría de las veces son los verdaderos “indies” operando fuera del sistema corporativo (Deuze, 2007).



Figura 3.12 Actividades realizadas por los estudios. Informe del año 2020 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

Por último se puede observar que la industria de videojuegos argentina está compuesta en su mayor parte por empresas de muy pocos empleados. Como observamos en la figura 3.5, el 53% de las empresas son del tipo “micro”, lo cual significa que están compuestas por menos de 7 personas en su totalidad (Sitio argentina.gob.ar).

En conclusión, una parte significativa de la industria actual de videojuegos en Argentina cumple con características asociadas al sector independiente de videojuegos. Muchas empresas del sector apuestan por la autofinanciación, y en este aspecto apuntan a lograr mayor control económico de sus proyectos (Early Access, 2022). Adicionalmente, aunque muchas empresas buscan trabajar en conjunto con un editor externo, un sector importante de hecho prefiere no estar asociado con un editor. Como última característica asociada a los estudios independientes, se observa que en la industria Argentina abundan las empresas que poseen muy pocos recursos humanos empleados en el desarrollo de sus productos. Apoyándonos en esta caracterización de la industria Argentina como “independiente”, resulta relevante para esta tesina analizar cómo se trabaja la accesibilidad en este tipo de industria, y conocer qué tipos de problemáticas se dan en este tipo de desarrollos que obstaculizan la creación de juegos accesibles.

3.2 Accesibilidad desde la perspectiva de la industria

Cuando los juegos imponen limitaciones sobre los jugadores con discapacidad, al no ofrecerles una manera de interactuar adecuadamente con el mismo, obstaculizan la capacidad del jugador para sumergirse en la experiencia completa del “juego” como

actividad libre y absorbente (Huizinga, 1955). Es más, el trabajo extra que deben realizar los jugadores con discapacidad al tratar de romper esos obstáculos que encuentran al interactuar con el juego, representa una barrera que los jugadores deben sortear y que por lo general les demanda un esfuerzo y costo adicional.

Por lo general los jugadores encuentran su propia solución alternativa para sortear estas barreras en los juegos. Por ejemplo, los jugadores con dificultades visuales se apoyan en los elementos de sonido del juego, en la ayuda externa de un jugador mediante un chat de voz, o ajustan los colores y tamaños de los elementos visuales desde las opciones de configuración del juego para hacerlos más visibles. Dependiendo de la naturaleza del juego y de las necesidades del jugador, algunas de estas soluciones alternativas pueden llegar a ser tan complejas como el remapeo del control del juego para aceptar ser controlado con los pies, o la utilización de extensiones de terceros para poder controlar el juego usando la mirada del jugador.

Generalmente se considera a los videojuegos como una mera forma de entretenimiento, sin embargo para un jugador con discapacidad el videojuego puede cobrar otra relevancia, permitiéndole conectar con otras personas, relajarse y divertirse con amigos interactuando en un ambiente seguro. Adicionalmente, los videojuegos pueden ser utilizados como herramientas con fines educativos y terapéuticos. Francesca Gauci (2021), por ejemplo, presenta en su tesis (*Game Accessibility for Children with Cognitive Disabilities*, 2021) una serie de soluciones de accesibilidad para mejorar la interacción entre los videojuegos y los jugadores con dificultades cognitivas como el trastorno de aprendizaje, trastorno de la coordinación motora y el trastorno del déficit de la atención. Aunque la intención de Gauci es mejorar la accesibilidad en cualquier tipo de juegos, remarca la importancia de los videojuegos como entretenimiento y diversión, por sobre la de los juegos serios. Los “juegos serios” se pueden definir como los juegos que fueron diseñados con un objetivo diferente al de la diversión del jugador. Existen múltiples propósitos en los juegos serios: además de la educación se pueden realizar juegos para publicitar una marca o producto, juegos para la salud, en donde el juego está diseñado como terapia psicológica o rehabilitación física, o juegos de simulación como por ejemplo los simuladores de gestión de compañías o negocios (Dörner, 2016).

Uno de los objetivos principales de los desarrolladores de videojuegos entonces debería ser crear videojuegos accesibles, permitiendo a los jugadores con discapacidad reducir la mayor cantidad posible de esfuerzo adicional requerido para interactuar con el juego. Esto significa analizar qué tipo de dificultades pueden surgir para personas con discapacidad de acuerdo al tipo de juego que estamos creando, y diseñar estrategias que reduzcan estas dificultades para el jugador. Estas estrategias puede que nunca lleguen a alcanzar a la gran mayoría de los jugadores, y podemos caer en el error de pensar que por eso no son

importantes, pero para la minoría que realmente las necesita tiene un impacto verdaderamente tangible. El hecho de que un juego soporte o no ciertas opciones de accesibilidad puede hacer que un jugador opte por no jugarlo, que pueda jugarlo pero de una manera incómoda, o puede que el juego sea lo suficientemente inaccesible para excluir al jugador por completo.

Uno de los objetivos de esta tesina es evidenciar la importancia que tiene la accesibilidad en gran parte de la comunidad de jugadores. Ya se analizaron cuáles son las barreras que enfrentan los jugadores con diversidad funcional, y también se examinaron las características que poseen los estudios independientes dentro de la industria de desarrollo de videojuegos. A continuación se describirán las particularidades del desarrollo independiente de videojuegos que pueden obstaculizar la creación de accesibilidad.

Obstáculos en la implementación de accesibilidad en la industria independiente

De acuerdo a Kovanto (2013), en cada uno de los aspectos en el proceso de desarrollo de un juego independiente, como el diseño, la gestión del proyecto, el marketing y el desarrollo, pueden ocurrir problemas que influyen negativamente en el producto final construido. Luego de estudiar postmortems algunos juegos independientes, el autor identificó los 10 problemas más comunes:

1. Mal diseño del juego.
2. Planificación pobre.
3. Malas decisiones durante la producción.
4. Problemas técnicos.
5. No hay suficiente contenido.
6. Problemas en el lanzamiento y post-lanzamiento.
7. Múltiples roles.
8. Mala promoción y marketing.
9. Malas decisiones de diseño.
10. Falta de experiencia de los desarrolladores.

Aunque estas problemáticas están relacionadas con las etapas de todo el proceso de desarrollo de software, se pondrá especial atención al análisis de cómo pueden afectar a la accesibilidad. Para esto se analizarán, la problemática del mal diseño del juego, la planificación pobre, y el problema de los múltiples roles.

Las problemáticas del mal diseño del juego están relacionadas con la etapa de diseño, y no solo involucra a el diseño de arte conceptual, de las interfaces de juego o el diseño de niveles. Está relacionado también con el diseño del gameplay. En este grupo se identifican problemáticas que se dan en torno a la jugabilidad, los controles, las mecánicas de juego, el diseño del modo multijugador, los tutoriales, la interfaz del usuario, y errores en el balance

del juego. Obviamente los problemas en el diseño del juego se dan en las etapas tempranas del desarrollo, y se acarrearán durante todo el proceso de creación. Además de impactar en el tiempo de desarrollo y calidad del resultado final, estos problemas afectan a la creación de la accesibilidad. Dentro de cada uno de los aspectos mencionados anteriormente es posible analizar la dimensión de la accesibilidad. Por ejemplo, podríamos tener una mecánica muy divertida para el juego, pero que impacta negativamente en la inclusión de algunos jugadores: si la mecánica se basa en la utilización de colores y no se piensa en la accesibilidad, es posible generar barreras en los jugadores con una deficiencia en la percepción del color. Si se diseña un sistema de combate que se basa en la utilización de 5 diferentes botones para realizar un combo, vamos a encontrar durante la etapa de testing que algunos jugadores tienen problemas para controlar al personaje. Estos son ejemplos de cómo la omisión del análisis de la accesibilidad en la etapa de diseño puede generar problemas de inclusión a futuro.

La problemática de la planificación pobre está relacionada a la gestión del proyecto, y contiene problemas vinculados a la mala sincronización de tareas, el crunching⁶, la falta de experiencia en planificación, cargas de trabajo subestimadas, errores de estimación y features (características) que aparecen a último momento. Los datos recopilados de los postmortems evidencian que las malas estimaciones de tiempo, un alcance del proyecto poco realista y la decisión de agregar nuevas features y contenido a último momento fueron las causas más comunes de problemas en proyectos de juegos independientes. También, debido a una planificación de tareas poco precisa obliga a recurrir al crunching. La mala planificación del proyecto afecta a todas las tareas de desarrollo, incluyendo a las vinculadas con la accesibilidad. Un error usual que se puede cometer con las tareas de accesibilidad es pensar que debido al alcance que tiene sobre la cantidad de jugadores, su prioridad no es alta, y que se puede dejar para último momento. La implementación de la accesibilidad a último momento puede traer algunas consecuencias negativas como por ejemplo, el alargamiento del tiempo de desarrollo, la pérdida de calidad con respecto al diseño inicial, o directamente la eliminación de la feature de accesibilidad (Watterton, 2022). Cuando el equipo de desarrollo es pequeño y hay menos recursos humanos disponibles, muchos miembros del equipo tienen que cumplir múltiples roles. Un equipo indie perfecto es un grupo de personas con múltiples talentos, y que pueden ser productivos en tantas áreas de desarrollo como sea posible. Equilibrar la cantidad de roles a cubrir puede ser problemático. En muchos de los proyectos independientes, un problema típico es cumplir demasiados roles, lo que significa que las tareas de trabajo se fragmentan demasiado, y por lo tanto demasiadas áreas diferentes de las que ocuparse. Este es un escenario bastante

⁶ En la industria de los videojuegos, el crunch es un tiempo extra de trabajo durante el desarrollo de un juego que obliga a los desarrolladores a trabajar fuera de hora, comúnmente para cumplir con fechas límites.

común y que debe evitarse. Según Baker (2020), en estos casos se recomienda contratar un especialista en accesibilidad a tiempo completo, quién será el encargado de mantener una línea de decisiones y dirigir el esfuerzo de todo el equipo de desarrollo para lograr llevar a cabo las soluciones de accesibilidad. En caso de no poder contar con la posibilidad de contratar a un especialista, es recomendable que solo uno de los desarrolladores tome este rol, en lugar de repartir la responsabilidad entre varios.

Para los estudios pequeños de juegos independientes es realmente importante el proceso de QA. El juego debe probarse tanto como sea posible antes del lanzamiento. El control de calidad debe incluir un plan de prueba y las herramientas adecuadas, además de contar con los casos de prueba necesarios. Es importante organizar sesiones de testing para obtener feedback de los jugadores. Por lo general, en la etapa de alpha testing se cuenta con voluntarios que participan de estas pruebas, y en la etapa de beta testing se organizan pruebas con grupos de jugadores más específicos, elegidos por los desarrolladores. Estas pruebas brindan información valiosa sobre la mecánica, el “game feel”⁷, los controles, gráficos, implementación de la inteligencia artificial, y si al grupo objetivo le gusta o no el juego. Con respecto a la accesibilidad, esta etapa es esencial para poder detectar barreras en los jugadores. Así como se aplica QA para detectar problemas en el juego y generar una lista de bugs a corregir, se debe utilizar el proceso de QA para identificar problemas con el diseño, balance o mecánicas de juego que afectan a la accesibilidad y luego crear una lista de posibles soluciones a los problemas identificados. Es importante incorporar al proceso de QA a jugadores con diversidad funcional, con el objetivo de obtener un feedback de primera mano. Adicionalmente es importante volver a realizar una revalidación sobre la implementación de las soluciones de accesibilidad (Watterton, 2022). Es fundamental prestar atención al feedback y críticas constructivas que brindan los jugadores para los cuales se implementaron las soluciones de accesibilidad, para validar que realmente funcionan de manera correcta.

Ejemplos de accesibilidad en juegos independientes

Para entender mejor cómo trabajar con la accesibilidad es útil conocer sobre los problemas y soluciones con los que se encontraron otros desarrolladores. Se puede aprender mucho de la experiencia de otros proyectos independientes, y por eso a continuación se analizarán algunos postmortems de juegos independientes, y se examinará cómo afecta la accesibilidad en estos juegos.

⁷ En videojuegos el game feel o “sensación de juego” hace referencia a la sensación intangible que transmite el juego al jugador.

Skye Tales

En primer lugar tenemos el caso del juego “Skye Tales” para Nintendo Switch, del estudio Punny Astronaut. Cari Watterton, diseñadora de accesibilidad, cuenta en la charla en el marco de la GAConf, una conferencia impulsada por IGDA-GASIG (International Game Developers Association, Game Accessibility Special Interest Group) con el objetivo de compartir las experiencias de desarrollo de accesibilidad y ayudar a la industria a hacer juegos accesibles para todos (Watterton, 2022). Cari cuenta sobre el desarrollo del juego Skye Tales, y cómo se logró la implementación de la accesibilidad. Con cuatro meses para terminar el juego y lanzarlo al público, se comenzaron tareas de mejoras en todos los apartados: los gráficos, el gameplay, UX y agregar algunas nuevas features al juego. Fue en esta etapa tardía del desarrollo que también se comenzó a trabajar la accesibilidad.

Ya que Cari, encargada de llevar adelante las tareas de accesibilidad para Skye Tales, no contaba con experiencia trabajando con esa temática, necesitaba primero entender sobre accesibilidad. Una de las primeras fuentes que encontró útil fue la “Family Video Game Database”, una recopilación de varios análisis de videojuegos en donde se analizan diversos aspectos de los juegos, entre los cuales se encuentra la accesibilidad. Estos análisis están escritos de una forma coloquial, sin términos técnicos, de forma que son sumamente sencillos de entender. La segunda fuente de información sobre accesibilidad fueron los recursos de accesibilidad disponibles en la web, entre las que se encuentran la “XBox Accessibility Guidelines”⁸, la “Game Accessibility Guidelines”⁹, y las guías APX de AbleGamers¹⁰. Otra fuente importante de información fueron otros desarrolladores de videojuegos, de los cuales pudo conocer qué problemas encontraron y cómo los solucionaron. Con respecto al diseño de la accesibilidad de Skye Tales, se pudieron implementar más de 40 features de accesibilidad en 4 meses.

El equipo de arte desarrolló una herramienta de debug que permite aplicar filtros al juego que simulan diferentes tipos de alteraciones en la percepción del color (figura 3.13). Gracias a esta herramienta se pudieron detectar problemas con algunos de los puzzles y diseños originales.

⁸ <https://docs.microsoft.com/en-us/gaming/accessibility/guidelines>

⁹ <https://gameaccessibilityguidelines.com/>

¹⁰ <https://accessible.games/accessible-player-experiences/>



Figura 3.13 Herramienta de debug que permite simular diferentes tipos de daltonismo.

Otra feature que implementaron fue una configuración que permite al jugador utilizar un modo de alto contraste, elegir colores diferentes para el jugador y los objetos interactivos, y hasta controlar el contraste con el fondo (figura 3.14).

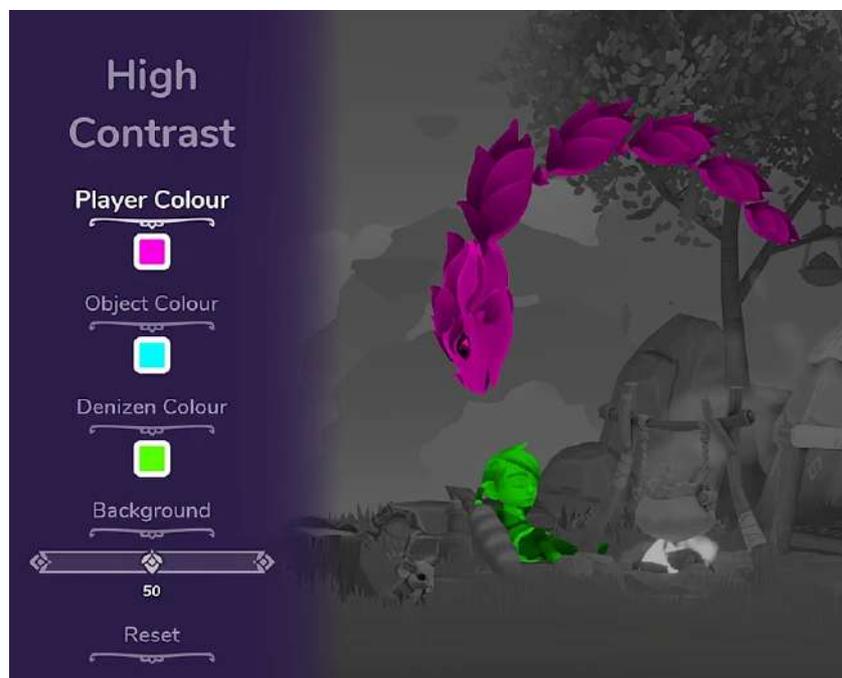


Figura 3.14 Configuración de contraste de elementos implementada para el juego Skye Tales.

Uno de los problemas que encontraron con la plataforma de lanzamiento es que Switch no provee soporte integrado para lectores de pantalla. Debido a que el juego posee narración

que necesita ser compatible con lectores de pantalla, se consultó con el equipo de Xbox, quienes utilizaron una solución propietaria para Minecraft en Nintendo Switch, herramienta que no estaba disponible para otros desarrolladores.

Otra solución de accesibilidad importante que no pudieron implementar fue el input remapping¹¹. Debido a que esta feature de accesibilidad no fue diseñada desde el principio, y a que su implementación requería demasiados cambios en el código, no pudo ser desarrollada para este juego. En lugar de eso se eligió dar soporte a la mayor cantidad de controles que se pudiera. Algo parecido sucedió con el soporte para escalado del texto. Esta feature no estaba presente en los diseños originales del juego, con lo cual toda la UI había sido diseñada de una forma incompatible con esta opción de accesibilidad. Realizar cambios en el diseño de las interfaces tan tarde en el desarrollo fue imposible. Uno de los puntos que remarca Cari en su presentación es la importancia de trabajar con la comunidad de jugadores para la cual se desarrollan estas soluciones de accesibilidad. No solo para detectar barreras presentes en el juego, sino también para validar que las soluciones implementadas realmente funcionan. Uno de los feedbacks que recibieron de los jugadores fue que se debería poder mover al personaje utilizando no solo el stick, sino también el D-Pad. Para algunos jugadores, utilizar solo el stick para mover el personaje puede llegar a ser cansador, por lo que es valioso contar con una alternativa.

De la experiencia narrada por Cari podemos resaltar:

- Cómo iniciar a trabajar la accesibilidad, y qué recursos hay disponibles.
- Ejemplos prácticos de por qué es importante tener en cuenta la accesibilidad en el planeamiento del proyecto.
- Cómo y por qué se deben validar las features de accesibilidad.

ScourgeBringer

El segundo ejemplo es una historia contraria a la anterior. En lugar de agregar accesibilidad sobre un juego previamente diseñado, en este caso se pensó la accesibilidad desde el inicio del diseño. Thomas Altenburger, de Flying Oak Games, cuenta cómo fue el diseño del juego “ScourgeBringer” (2020), y cuáles fueron los desafíos relacionados a la implementación de opciones de accesibilidad.

Al contrario que Cari, Thomas contaba con experiencia trabajando en la accesibilidad para un juego anterior “Hoy!” (figura 3.15). Este juego, que estaba diseñado para utilizar el sistema Kinect¹² de Xbox 360, era un juego de baile en donde los jugadores debían

¹¹ Opción de accesibilidad que permite al jugador reconfigurar los controles necesarios para ejecutar acciones dentro del juego

¹² Un controlador para Xbox 360 que permite a los usuarios controlar e interactuar con la consola sin necesidad del contacto físico, mediante la utilización de gestos y comandos de voz.

posicionar su cuerpo de acuerdo a las instrucciones que aparecían en pantalla, y mover las piernas y brazos de acuerdo a las figuras que aparecían durante la partida.



Figura 3.15 Persona jugando al videojuego “Hoy!” usando el sistema de Kinect para Xbox 360.

Luego que un jugador con movilidad reducida se acercara a los desarrolladores, indicando que el juego era muy divertido pero que no lo podía jugar, Thomas agregó un “modo sentado” que permitía jugar a “Hoy!” sin la necesidad de mover las piernas, y utilizando solo la movilidad de los brazos. A partir del encuentro con este jugador, Thomas comenzó a aprender más sobre accesibilidad en videojuegos. Según Thomas, por lo general los desarrolladores independientes no toman conciencia sobre la accesibilidad en los videojuegos hasta que no lo experimentan de primera mano, con algún familiar o, como en su caso, un jugador se lo hace notar.

“ScourgeBringer” (2020) es un juego rápido platformer roguelite¹³ de movimiento libre, bastante demandante en la parte visual, y con dificultad muy elevada. Desde el principio, el objetivo de Thomas fue diseñar el juego de forma que pudiera minimizar la mayor cantidad de barreras posible para todos los jugadores. Con el objetivo de crear un juego “justo” con el jugador, tomó varias decisiones de diseño que afectaron enormemente la accesibilidad del

¹³ Un juego roguelite es un juego que cuenta con algunas o pocas características de un juego roguelike, es decir un juego de mazmorras.

resultado final. En principio, no hay scrolling lateral ni horizontal, de modo que la escena de juego permanece estática y no hace falta controlar la cámara. En segundo lugar, el gameplay está diseñado para tomar lugar en el centro de la pantalla, de modo que en pantallas grandes, el jugador no tenga que mover mucho la cabeza. La UI no interrumpe en ningún momento la acción, y se mantiene siempre en los costados de la pantalla (figura 3.16).



Figura 3.16 Pantalla de juego de uno de los niveles de “ScourgeBringer” (2020).

Adicionalmente al diseño del nivel y la cámara, fue necesario tomar el alto contraste como una parte principal del diseño. Para poder identificar y diferenciar fácilmente los elementos de juego en un ambiente caótico, con muchos enemigos y proyectiles en pantalla, fue imprescindible contar con un fondo simple y de tonalidades oscuras. Los personajes en pantalla también cuentan con un diseño pensado en el contraste, por ejemplo los enemigos poseen una gama de colores que contrastan con el color del ambiente del nivel en donde aparecen, y el personaje principal siempre tiene colores tendiendo al blanco, para poder identificarlo fácilmente en la pantalla llena de enemigos, proyectiles y otro tipo de ruido visual. Estas decisiones de diseño hacen que siempre se pueda identificar dónde está el jugador, dónde están los enemigos, y qué es lo que está pasando en pantalla.

Otra decisión de diseño que se implementó teniendo en cuenta la accesibilidad fue la

utilización de “sound cues”¹⁴ para todas las acciones importantes que realiza el personaje principal, y los enemigos. De esta forma se puede escuchar a un enemigo y entender cuándo está por disparar un proyectil. Adicionalmente, una decisión que se tomó para bajar la dificultad, y simplificar el control del juego, fue la de no permitir al jugador controlar la dirección de disparo de los proyectiles, de forma que no fuera necesario utilizar un segundo stick para controlar esto.

Una herramienta muy útil que se utilizó durante el desarrollo del juego fue un filtro que se podía activar durante el juego, y permite agregar un efecto de difuminado sobre la imagen (figura 3.17). Utilizando este filtro durante las partidas, los desarrolladores pueden validar, de una forma sencilla, si aún con ese nivel de ruido en la pantalla es posible distinguir el personaje, los enemigos y los elementos de la pantalla. Según Thomas esta fue una herramienta muy útil, sencilla de usar y rápida para implementar.



Figura 3.17 En la izquierda, la pantalla del juego como se ve normalmente. En la derecha la pantalla de juego luego de aplicar el filtro de difuminado.

En cuanto al gameplay, y teniendo el mismo Thomas dificultades musculares en las manos al probar el juego, se implementaron opciones que permiten desactivar la vibración del joystick. El juego permite ajustar el volumen de la música y los efectos de sonido por separado, además de permitir cambiar el modo de la música, pudiendo elegir entre una música dinámica¹⁵ o un modo de música estándar, mucho más predecible y sencilla de escuchar.

En cuanto a los controles, se permite cambiar el modo de interacción preferido para algunas acciones. Por ejemplo para mostrar el mapa del nivel, se puede configurar para poder presionar una sola vez el botón, en lugar de mantenerlo presionado. Además se implementó remapping para todos los controles, de manera que hacerlos totalmente configurables. Para los jugadores que prefieren utilizar otro estilo de controles, se permite usar el mouse y teclado para controlar el juego.

¹⁴ Efectos de sonido u otro tipo de audio que tiene un significado específico dentro del juego.

¹⁵ En videojuegos, la música dinámica es un tipo de música que se genera en vivo en base a las acciones que están pasando dentro del juego.

Este juego fue lanzado en Early Access mediante Steam a principios del año 2020. Aprovechando esta nueva base de jugadores se integraron analíticas¹⁶ dentro del juego, las cuales incluyen estudios demográficos de los perfiles de los jugadores. Se identificaron tres “niveles” de jugadores:

- Un 10% de los jugadores estaban jugando de una forma muy buena, completaron los niveles de una forma rápida, sin reiniciar muchas partidas, y sin recibir mucho daño. Para este tipo de jugadores “expertos”, el contenido del juego era escaso, se quedaban sin nuevos niveles muy rápidamente, y la dificultad del juego era baja.
- El 80% de los jugadores estaban en un balance óptimo, ya que terminaban el juego en el tiempo esperado y los enemigos presentaban una dificultad adecuada.
- Para el 10% de los jugadores restantes el juego resultó extremadamente difícil de jugar, con muchos jugadores ni siquiera pudiendo terminar el primer nivel.

El objetivo propuesto para balancear la población de jugadores fue llevar a los jugadores a esa zona media en donde el juego representa cierta dificultad, pero sin ser completamente excluyente con algunos jugadores. La solución planteada para lograr este objetivo fue introducir opciones que mejoren la accesibilidad y la dificultad directamente dentro de la narrativa del juego, a través de elementos diegéticos¹⁷. Esto se realizó a través de la introducción de un personaje NPC¹⁸, que aparecía en determinados momentos para ayudar al jugador si estaba con dificultades para progresar, o para premiar a los jugadores que avanzan muy bien en el juego. Esto se logró en base a un sistema que monitorea las acciones del jugador (salud, cantidad de enemigos encontrados, rapidez para progresar, etc) y analiza qué tipo de ayuda o dificultad debe utilizar para llevar al jugador a un balance de juego óptimo.

Otros tipos de opciones de accesibilidad que se agregaron se relacionan con la velocidad del juego. Se permite ajustar tanto la velocidad del juego en general, como la velocidad de los proyectiles de los enemigos. Thomas expresa que esta fue una opción de la que todos se vieron beneficiados. Por un lado los jugadores que tienen dificultad para poder avanzar en el juego pueden bajar la velocidad, y de esta forma avanzar más fácilmente. Por el otro, los jugadores expertos pueden aumentar la velocidad del juego, y agregar un desafío extra a la partida. Según Thomas esta feature también fue muy rápida de implementar, y probó ser muy exitosa.

Aunque la tipografía elegida para los elementos de la UI encaja con el diseño y estética del juego, cercana al pixel art¹⁹, hace difícil de ver y entender el texto. Debido a esto se introdujo una opción que permite cambiar la tipografía utilizada en las interfaces por una

¹⁶ Forma de análisis de comportamiento de los jugadores que se basa en el tracking y análisis de datos.

¹⁷ Que forman parte del universo del juego.

¹⁸ Non Playable Character, un personaje dentro del juego que no es controlado por el jugador.

¹⁹ Es una técnica de arte que se basa en dibujar imágenes digitales a nivel de píxeles.

fuentes en alta definición, que en general se ven mucho mejor, y son más fáciles de leer en pantallas grandes (figura 3.18 y figura 3.19).

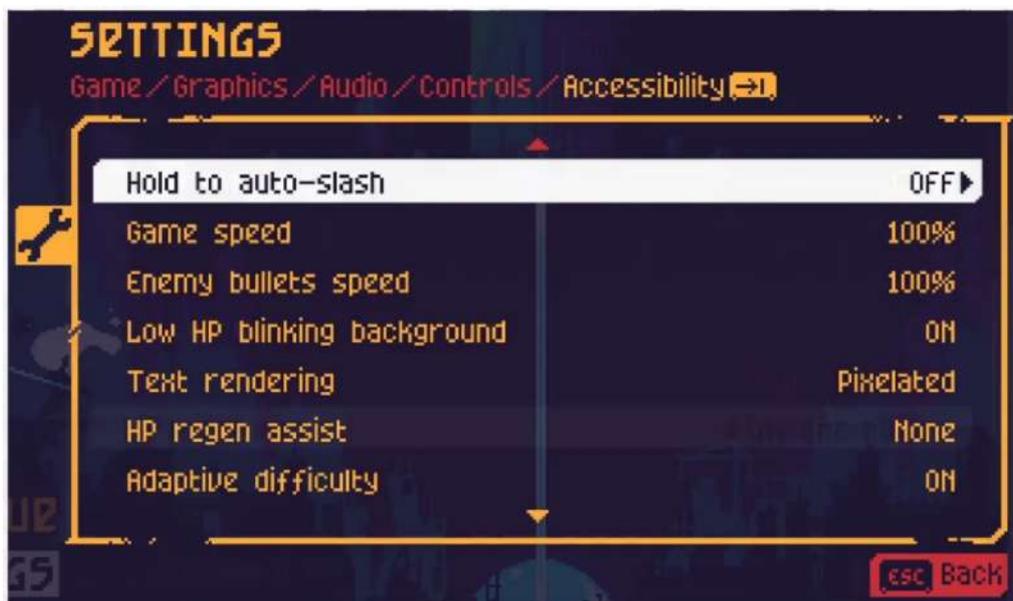


Figura 3.18 Pantalla de configuración del juego, en donde se utiliza una tipografía con efecto pixelado. ScourgeBringer (2020).

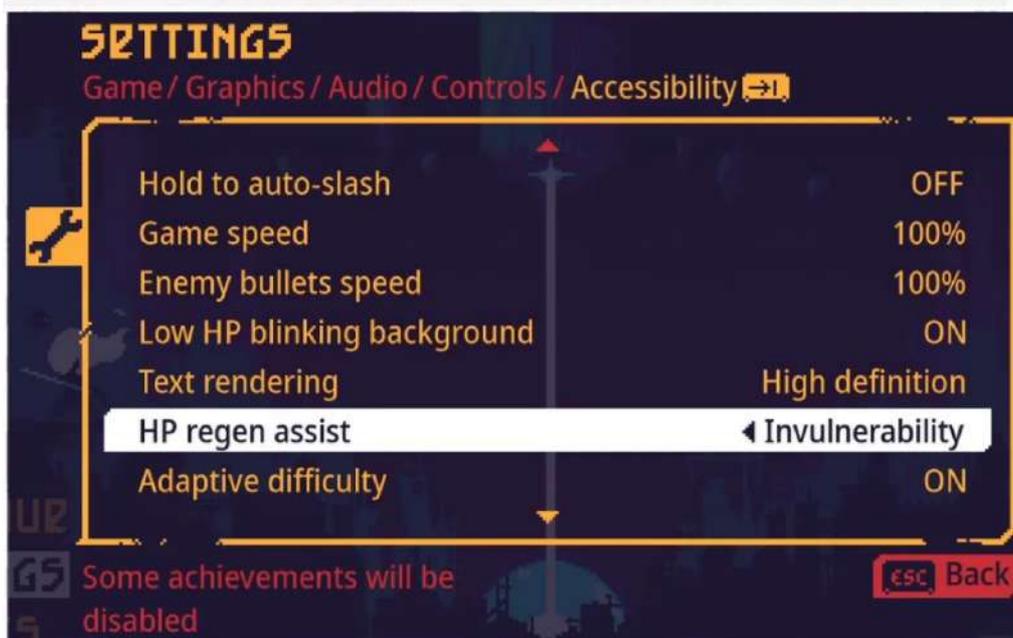


Figura 3.19 Pantalla de configuración del juego, donde se utiliza la tipografía en alta definición. ScourgeBringer (2020).

Por último, una opción de accesibilidad que probó ser extremadamente útil para algunos jugadores fue la inclusión del modo “invencible”. En este modo, el nivel de regeneración de la salud del personaje es mucho más alta, con lo cual no se puede morir frente a proyectiles

enemigos. Esto permite que muchos jugadores que tienen grandes dificultades para progresar, puedan seguir avanzando en el juego.

Luego de implementadas estas mejoras de accesibilidad, se pudo comprobar que la población de jugadores quedó balanceada en 5% de jugadores “expertos”, 5% de jugadores con dificultades para avanzar, y 90% de jugadores en el nivel óptimo de dificultad. Aunque se identificaron otras mejoras de accesibilidad, Thomas cuenta que no hubo el tiempo suficiente para implementarlas. Estas mejoras involucran un modo de juego diseñado para los jugadores con dificultades auditivas, y una simplificación de los controles para realizar combos en los ataques. Aunque estas mejoras si fueron planeadas, el tiempo de desarrollo necesario para poder llevarlas a cabo sobrepasaba el de tiempo con el que contaban.

De la experiencia implementando accesibilidad en “ScourgeBringer” se pueden extraer algunos puntos valiosos:

- Es importante encontrarse con los jugadores, y entender qué dificultades tienen al jugar.
- Las opciones de accesibilidad muchas veces son fáciles de implementar, no llevan mucho tiempo de desarrollo y benefician a muchos jugadores.
- Por lo general las opciones de accesibilidad pueden resultar beneficiosas para todos los jugadores, sin importar si tienen algún tipo de discapacidad o no.

Conclusiones

De la experiencia contada a través de estos postmortems de juegos independientes es posible rescatar algunas conclusiones.

- **Pensar la accesibilidad desde un principio:** diseñar el juego teniendo la accesibilidad en mente desde las primeras etapas permite al desarrollador lograr una planificación más organizada del proyecto.
- **Las opciones de accesibilidad brindan algún beneficio a todos los jugadores, no solo a los jugadores con discapacidades:** la usabilidad, legibilidad y flexibilidad se benefician de la accesibilidad.
- **Trabajar en mejoras de accesibilidad junto a los jugadores para quienes están destinadas estas mejoras:** siempre es mejor aprender de la experiencia directa de los jugadores e identificar cuáles son las dificultades o barreras que encuentran en el juego, y además es importante revalidar con estos jugadores que las soluciones implementadas realmente funcionan.

Iniciativas de concientización sobre accesibilidad

Dentro de la comunidad de desarrolladores existen varias iniciativas que apuntan a concientizar, tanto a jugadores como a otros desarrolladores, acerca de la accesibilidad, la

diversidad de los jugadores e incluso plantear guías o soluciones de accesibilidad para videojuegos. Como se mencionó en el Capítulo 2, durante los últimos años se comenzó a dar más visibilidad a la accesibilidad en videojuegos, no solo mediante la inclusión de opciones de accesibilidad en juegos AAA como “The Last of Us: Part 2” (2020) y en juegos independientes como “Celeste” (2018), sino también a nivel de toda la industria mediante premios y reconocimientos a los juegos accesibles. Una de las iniciativas más comunes para dar a conocer y generar conciencia sobre la accesibilidad son las denominadas “Game jams”. Este tipo de eventos no solo son una oportunidad para desarrollar y poner a prueba las habilidades como desarrollador de videojuegos, sino también para aprender de otros desarrolladores (Scott, 2013).

Jams de accesibilidad

La “Game jam” es un evento, también llamado hackatón, que tiene como propósito la creación de uno o más videojuegos en un corto período de tiempo, normalmente entre 24 y 48 horas (Global Game Jam Website, 2022). Los participantes de estas jams suelen ser de diferentes disciplinas: programadores, diseñadores de juegos, artistas, actores de voz, etc. Una de las jams más conocidas es la “Global Game Jam” que se celebra todos los años y de la cual participan desarrolladores de todo el mundo.

Existen varias jams orientadas a crear conciencia sobre la accesibilidad, y una de ellas es la “Access Jam” impulsada por “Accessibility Unlocked”, un grupo conformado por desarrolladores de videojuegos de Australia y Nueva Zelanda cuyo objetivo es conectar a los desarrolladores y jugadores con algún tipo de discapacidad con la finalidad de crear redes, compartir recursos y lograr crear juegos cada vez más accesibles (Accessibility Unlocked Website, 2022). La Access Jam dura todo un mes, y está abierta a participantes de todo el mundo. Se alienta a los desarrolladores a hacer juegos sobre su experiencia con la discapacidad, a implementar juegos con mecánicas accesibles, a implementar opciones de accesibilidad en los videojuegos, o a crear herramientas que faciliten el trabajo de la accesibilidad, como generadores de paletas de colores, editores de subtítulos, etc.

En Argentina también se realizó en el 2021 la jam “Crear juegos con accesibilidad”, un evento impulsado por la Agencia Nacional de Discapacidad (ANDIS) y ayudado por la Fundación Argentina de Videojuegos (FUNDAV), en celebración del 3 de diciembre, el Día Internacional de las Personas con Discapacidad. Este evento incluyó capacitaciones en materia de creación de videojuegos y de accesibilidad. La idea general de la jam fue crear juegos accesibles, respetando una guía de accesibilidad provista para el evento, y basada en la “Game Accessibility Guidelines” (<https://gameaccessibilityguidelines.com/>).

Entidades y Organizaciones

Existen varias entidades y organizaciones alrededor del mundo que trabajan con el objetivo de lograr mejorar la accesibilidad y la inclusión en los videojuegos. Por lo general estos grupos están conformados por desarrolladores promotores de la accesibilidad, jugadores con discapacidades, y especialistas en accesibilidad.

La fundación “IGDA” (International Game Developers Association) creó en 2003 un subgrupo especial de voluntarios interesados en promover la accesibilidad en videojuegos, o “GASIG” (Game Accessibility Special Interest Group, 2022). Este grupo se creó para apoyar a los desarrolladores de videojuegos alrededor del mundo en crear lazos entre desarrolladores, detectar necesidades y barreras relacionadas a discapacidades, impulsar la tecnología en videojuegos desde el punto de vista de la accesibilidad, y compartir el conocimiento entre los miembros de la comunidad (IGDA-GASIG Website, 2022).

La fundación “AbleGamers”, fundada en 2004, es una organización sin fines de lucro, dedicada a mejorar la accesibilidad en los videojuegos, permitiendo a más personas con discapacidades jugar videojuegos (“Get help”, Able Gamers Website, 2022). Esta fundación trabaja directamente con jugadores para evaluar sus necesidades y desafíos y determinar el equipamiento necesario para que puedan jugar videojuegos. También trabaja para crear recursos y material enfocado directamente en los desarrolladores, con el objetivo de facilitar la creación de videojuegos accesibles.

“DAGERSystem” es un portal web de periodismo de videojuegos, creado por y para personas con discapacidades. Presenta una variedad de reseñas sobre videojuegos, siempre desde el punto de vista de la accesibilidad, y artículos que brindan perspectivas de accesibilidad en los videojuegos (DAGERSystem Website, 2022).

“Can I Play That?” es un sitio de reseñas de videojuegos desde la perspectiva de la accesibilidad. Se creó en 2018 como un sitio que reúne noticias y reseñas sobre videojuegos, y creció hasta convertirse en un sitio consultado tanto por jugadores como por estudios profesionales alrededor del mundo. Brinda información, historias y puntos de vista sobre accesibilidad en videojuegos para los jugadores y desarrolladores, además de diseñar y llevar adelante talleres para estudios profesionales sobre las temáticas de diversidad, igualdad, inclusión y accesibilidad en los videojuegos (Can I Play That? Website, 2022).

La FUNDAV (Fundación Argentina de Videojuegos), creada en 2016 con el objetivo de promover e impulsar el desarrollo de videojuegos en Argentina, está conformada por un grupo de profesionales de la industria que impulsó la creación de varios proyectos, entre ellos la “Game Work Jam” (FUNDAV Website, 2022), y varios proyectos en conjunto con el Ministerio de Cultura de la Nación enfocados en la promoción del trabajo y la reinserción laboral. El sello de Accesibilidad de Videojuegos es una iniciativa de FUNDAV que tiene

como objetivo impulsar la empatía y la concientización en el diseño de videojuegos para lograr alcanzar un grado de accesibilidad cada vez mayor, y a su vez proveer información al jugador acerca del grado de accesibilidad y limitaciones que poseen los videojuegos.

La Asociación de Desarrolladores de Videojuegos Argentinos (ADVA), es una organización sin fines de lucro compuesta por estudios y desarrolladores de videojuegos que trabajan para promover el crecimiento del entretenimiento digital e interactivo producido en Argentina (ADVA Website, 2022). Es una institución intermedia que articula acciones entre los estudios y emprendedores dedicados al desarrollo de videojuegos y el sector público, el sector privado y la academia, impulsando y potenciando el desarrollo de la industria local de videojuegos.

Premios a la accesibilidad

Desde hace años y como una forma de dar visibilidad a la accesibilidad en videojuegos, se comenzaron a premiar a los equipos que implementan soluciones de accesibilidad en sus juegos. En estos premios participan todo tipo de juegos de la industria, desde juegos AAA hasta independientes. “The Game Awards”, una entrega de premios a los mejores juegos de la industria que se realiza desde el 2014, agregó en el 2020 una categoría exclusivamente para premiar a la accesibilidad: “Innovación en accesibilidad” (The Game Awards Website, 2020).

En el año 2020 también se comenzaron a realizar los “Video Game Accessibility Awards”, una premiación anual creada por la fundación AbleGamers para honrar a los juegos que realizan un esfuerzo consciente en mejorar e incorporar opciones de accesibilidad. El sitio “Can I Play That?” también realiza desde el 2019 su propia entrega de premios a los mejores juegos accesibles de la industria.

Charlas y talleres sobre accesibilidad

La fundación IGDA como parte de su estrategia para impulsar la diversidad y la inclusión en los videojuegos, realiza anualmente una serie de conferencias relacionadas a la accesibilidad, las “Game Accessibility Conference” (GA Website, 2022), mediante las cuales los desarrolladores cuentan con un espacio para encontrarse y dar a conocer sus experiencias en la implementación de accesibilidad en videojuegos.

La conferencia “Game Developers Conference” reúne talentos profesionales, tanto de la industria AAA como independiente, en una serie de conferencias y foros que duran 5 días con la finalidad de intercambiar ideas, plantear soluciones y compartir experiencias en el desarrollo de videojuegos. Aunque no tratan exclusivamente sobre accesibilidad, ocasionalmente se presenta esta temática en las charlas presentadas, por lo que estas

conferencias son un excelente recurso para aprender de la experiencia de otros profesionales en relación a la implementación de accesibilidad (Dupire & Westin, 2016).

La serie de análisis de videojuegos “Game Maker's Toolkit”, creada por el periodista de videojuegos Mark Brown, examina varios aspectos del diseño de videojuegos, y trata de ayudar a los desarrolladores a tomar decisiones conscientes y a mejorar el diseño de los videojuegos creados (Game Maker's Toolkit YouTube Channel, 2022). A través de esta serie se tocan varias temáticas relacionadas a la accesibilidad y la inclusión de jugadores en los videojuegos. La serie “Designing for Disability” fue elegida por Polygon como uno de los mejores ensayos en video del 2018 (Schindel, 2018), y trata específicamente sobre el diseño de la accesibilidad en los videojuegos. A través de esta serie Brown también impulsa la jam “GMTK Game Jam”, la cual se lleva a cabo anualmente desde el 2017, y ocasionalmente incluye temáticas relacionadas a la inclusión y diversidad de jugadores y se impulsa la incorporación de opciones de accesibilidad.

¿Qué dicen los desarrolladores independientes sobre la accesibilidad?

Una de las formas más sencillas y completas en las que podemos aprender sobre accesibilidad, es a través de la experiencia de los mismos desarrolladores. Para poder conocer el estado actual de la industria del videojuego independiente es preciso escuchar a los mismos desarrolladores. A continuación se describen tres entrevistas realizadas a desarrolladores independientes de videojuegos de la escena local. El documento con la guía original de las preguntas de la entrevista está disponible como documento anexo a esta tesina.

Sobre los desarrolladores

La primera entrevista fue con Nicolás Castez, un desarrollador de 37 años de la ciudad de La Plata, Buenos Aires, egresado de la carrera de Diseño y Comunicación Visual. Nicolás trabaja como diseñador de videojuegos y también está a cargo de su propio estudio “Avix Games”, un estudio independiente que desde hace 8 años realiza tanto outsourcing de videojuegos para empresas externas, como proyectos propios. Algunos de los juegos realizados por este estudio incluyen “Thumb Fighter” un juego de lucha de pulgares para dispositivos móviles, que tiene la particularidad de que se juega con un solo botón, y “Cartoon Network: Golf Stars”, un juego de golf para dispositivos móviles que mezcla varios personajes de los shows de la cadena de televisión Cartoon Network.

La segunda entrevista se realizó con Francisco Delledonne, un desarrollador de 28 años entusiasta de los videojuegos, y con varios proyectos desarrollados en su haber. Francisco trabaja actualmente en el estudio de videojuegos “Game Ever”, un estudio de videojuegos de Buenos Aires enfocado en la creación de videojuegos y animaciones 2D y 3D. Dentro del

estudio, Francisco trabaja como programador en videojuegos con tecnología HTML5, principalmente utilizando Phaser, un framework para crear juegos para navegadores web usando Canvas y WebGL. Además de su trabajo profesional, como hobby creó varios videojuegos desde el 2015, entre ellos “Infinite Surfer” y “Vuelteros”.

La última entrevista fue con Bruno Martínez, un entusiasta de los videojuegos de 30 años de la provincia de Tierra del Fuego. Bruno comenzó su camino en los videojuegos después de comenzar sus estudios en Comunicación Social, haciendo entrevistas sobre videojuegos. Bruno administra y es productor en su estudio “Puerto Global”, un estudio con base en Tierra del Fuego que realiza outsourcing de videojuegos, diseño multimedia de sitios webs y brinda servicios relacionados al diseño visual y multimedia. Algunos proyectos en los que participó fue “Cuando sea Grande” un juego desarrollado para la tercera edición del concurso nacional “Crear”, y un juego con temáticas relacionadas a la disminución de la capacidad auditiva, “In-Sonido” ganador del concurso nacional “Crear” en su cuarta edición (<https://www.argentina.gob.ar/jefatura/mediosycomunicacion/crear/crear-n3/proyectos-ganadores>, <https://www.argentina.gob.ar/jefatura/mediosycomunicacion/crear/proyectos-ganadores-de-la-beca-crear-n4-videojuegos-para-la-ninez>).

Sobre la forma de trabajo

De los desarrolladores entrevistados, nos interesa conocer la forma de trabajo de su equipo, su composición y tamaño. Además nos interesa saber qué herramientas utilizan para el desarrollo de videojuegos con especial atención en la accesibilidad.

En cuanto al equipo de desarrollo de Avix, Nicolás lo definió como un equipo chico, ya que por lo general cuentan con siete desarrolladores en total. Considera que el equipo de desarrollo es ágil, permitiéndole al estudio lanzar varios juegos en un corto período de tiempo. Destacó también que por lo general los desarrolladores no tienen un solo rol, sino que Avix es un equipo multidisciplinario en donde cada integrante conoce de varias disciplinas, y en donde comúnmente los desarrolladores participan de varios proyectos a la vez. El equipo trata de no quedarse sin trabajo para realizar, y aprovechan los tiempos con baja carga de desarrollo para investigar nuevas tecnologías o para trabajar en proyectos propios. Como tecnología de desarrollo utilizan Construct 3, para realizar juegos en HTML5, y Unity Engine para juegos en 3D. Además hace poco comenzaron a utilizar el motor PlayCanvas, primero poniéndolo a prueba en desarrollos más chicos como en game jams y proyectos propios, y luego usándolo en proyectos para clientes externos.

La forma de trabajo de Francisco es por lo general en solitario, es decir que no trabaja en equipo. Investiga por su cuenta sobre el desarrollo de videojuegos, aprende nuevas

herramientas y crea prototipos de videojuegos. Por lo general primero trabaja solo en la programación y diseño de juego, y luego, de ser necesario, involucra a otra persona que pueda aportar en el apartado artístico. En su último proyecto personal “Era en un bosque”, una aventura gráfica, trabajan 5 desarrolladores en total. Francisco utiliza principalmente Godot Engine y Unity, pero también suele utilizar Construct. En su ambiente laboral utiliza Phaser para realizar videojuegos para navegadores web.

Bruno Martínez comenta que en Puerto Global el equipo de trabajo es chico, ya que nunca son más de 4 personas. Como micro pyme, los proyectos y clientes con los que se relacionan son chicos, por lo que la “espalda financiera” con la que cuentan Puerto Global es más bien mínima. Bruno comenta que en Puerto Global además de desarrollar videojuegos hacen diseño multimedia, por lo que las herramientas que utilizan no sólo están destinadas al desarrollo de videojuegos, como Unity y Construct, sino que también utilizan Blender por ejemplo para realizar modelado 3D y animación y las herramientas comunes para el desarrollo web. Bruno además participó de varias game jams, eventos en los que trabaja por lo general solo, utilizando Construct o Flash para realizar el desarrollo del juego. Además utiliza Blender para las animaciones y el modelado de los personajes.

Sobre la Accesibilidad en Videojuegos

Se les preguntó a los entrevistados acerca del concepto de la accesibilidad en videojuegos, de la inclusividad en los videojuegos, y si consideraban que todos los juegos deberían ser inclusivos con el jugador. Por último se les preguntó a los tres desarrolladores ejemplos de accesibilidad que hayan implementado en sus juegos, y si conocen facilidades de accesibilidad que brindan las herramientas que utilizan para el desarrollo de videojuegos.

Con respecto al concepto de accesibilidad, Nicolás entiende que se refiere a las barreras u obstáculos que se pueden producir al jugar un videojuego, y a las soluciones que se implementan para que la audiencia objetivo pueda jugar sin problemas. Menciona que la accesibilidad no solo tiene que ver con la discapacidad en los jugadores, sino que pueden haber obstáculos para todo tipo de jugadores pudiendo convertir al juego en inaccesible. Un ejemplo que menciona es el de la realidad virtual. Es poco accesible, dado que solo los jugadores que poseen todas sus capacidades sensoriales y motrices activas pueden utilizarlo, y además solo está disponible para los jugadores poseen la capacidad económica mínima para acceder al equipo necesario. Nicolás menciona que no todos los juegos deben ser inclusivos porque sí, sino que existen algunos casos donde el juego puede ser limitado a cierto público de manera intencionada, por ejemplo por una cuestión artística. Desde su punto de vista, es imposible realizar un juego que sea accesible para absolutamente todas las personas, ya que existen infinitos jugadores e infinitas posibles barreras.

Algo importante que menciona Nicolás es el hecho de que como los productos que se crean dentro de su estudio son más bien juegos “chicos”, en comparación con juegos de empresas más grandes, no se puede destinar recursos de tiempo y fuerza de desarrollo a implementar opciones de accesibilidad, ya que comercialmente no tienen mucho impacto. Es decir, no se mejoran las ganancias del juego por agregar opciones de accesibilidad. Dentro del estudio, el enfoque tomado para crear accesibilidad es a través del diseño mismo del juego. Una manera de mejorar la accesibilidad, es mediante la introducción de tutoriales en los juegos, o mediante el diseño de mecánicas de juego accesibles para el público familiar y casual (su audiencia objetivo), con controles sencillos e intuitivos. Nicolás menciona el caso de diseño de un juego de carreras, en donde se optó por utilizar un tipo de cámara en particular para evitar mareos en los jugadores, y un tamaño de auto balanceado en relación con el tamaño de la pantalla, para que tanto los jugadores con baja visión como los jugadores sin problemas visuales pueda jugar de una forma cómoda.

Con respecto a las opciones de accesibilidad disponibles en los motores y frameworks utilizados en el desarrollo de los juegos, Nicolás menciona que Construct posee algunas facilidades para comunicarse con el sistema operativo, e incluir opciones como por ejemplo Text-To-Speech y control de la vibración en los controles.

Según Francisco, un juego accesible es un juego que puede ser jugado por cualquier persona, sin importar las limitaciones que posea. Además menciona que la inclusividad en videojuegos está presente en la narrativa, la historia y las temáticas que se incluyen en el juego. Hay una diferencia entre los términos “accesible” e “inclusivo”, dado que el concepto de inclusividad está muy ligado a problemáticas sociales, temas relacionados a minorías como la diversidad de género. Desde ese punto de vista, Francisco expresa que le interesa que sus juegos sean inclusivos, para ampliar la representación, y lograr que los temas que se traten dentro de los videojuegos sean cercanos a la realidad. Con respecto a la implementación de opciones de accesibilidad Francisco menciona que si bien le interesaría brindar estas opciones en sus juegos, por lo general para los desarrolladores independientes que se manejan con presupuesto bajo y con poco tiempo disponible, esto es algo muy difícil de lograr. Comparado con los juegos AAA, los recursos de tiempo, dinero, y cantidad de desarrolladores, son cosas que los equipos “indie” no tienen. Con respecto a opciones de accesibilidad implementadas en sus juegos, no recuerda haber implementado ninguna. Según Francisco esto se debe a que nunca se le ocurrió agregar estas opciones porque nunca había necesitado ninguna. Por lo general uno siempre ve las cosas desde su óptica, y le cuesta ponerse en la postura de otro tipo de jugadores. En cuanto a las herramientas de accesibilidad disponibles en los motores que utiliza, Francisco menciona que Godot Engine, por ejemplo, permite realizar input remapping de una manera bastante

sencilla. Es una funcionalidad que considera que debería estar disponible para cualquier juego, aunque por lo general se ven mucho más en juegos de PC.

Bruno define a la accesibilidad como la característica de un producto, relacionado a las facilidades que posee para permitir el uso por personas que no están en la misma línea de capacidad. También define a un juego inclusivo como aquel que en el criterio de realización tiene en cuenta a las personas que no tienen las mismas capacidades que el desarrollador, y por lo tanto permite que el juego sea disfrutado también por personas con discapacidades. También menciona que desde la construcción de la narrativa, un juego inclusivo no empuja hacia afuera a los jugadores que pertenecen a minorías, ya que la inclusividad es también conocer la realidad de los jugadores. Por ejemplo en el juego “Life Is Strange” los desarrolladores incluyeron un link que provee recursos de ayuda para jugadores con padecimientos de salud mental. En cuanto al interés por crear juegos accesibles, Bruno menciona que la realidad vivida por muchos estudios independientes argentinos, obliga a ser un poco más racional en cuanto a la implementación de opciones de accesibilidad. Desde un punto de vista ideal, implementaría opciones de accesibilidad en todos los juegos. Desde el punto de vista realista, es mucho trabajo extra que representa una inversión de tiempo, dinero y esfuerzo que no se justifica, ya que no traen ningún beneficio económico. Como ejemplo de opciones de accesibilidad que implementó en sus juegos, Bruno menciona que siempre trata de soportar múltiples tipos de controles, como teclado, mouse y joystick. Además comenta más allá de las opciones de accesibilidad, una forma que utiliza para lograr que los juegos sean accesibles, es realizar un diseño de juego que sea naturalmente accesible. De acuerdo al tipo de juego que se quiere construir, se tomarán las decisiones adecuadas para asegurarse de que sea accesible con un determinado grupo de jugadores. Para Bruno es importante pensar en la accesibilidad antes de que el juego sea desarrollado, ya que es difícil agregar accesibilidad encima de un juego que ya está terminado. En cuanto algunas soluciones de accesibilidad pensadas desde el diseño, pone como ejemplo a “Clash Royale”, un juego para dispositivos móviles en donde miles de personas juegan partidas online. En este juego la comunicación entre jugadores se realiza utilizando solo emojis, un ejemplo de cómo se puede diseñar un sistema de comunicación entre jugadores que sea sencillo de entender y además que rompa la barrera del idioma.

Sobre los juegos independientes y la Accesibilidad

En cuanto a la falta de herramientas de accesibilidad en los motores, Nicolás está de acuerdo en que de haber más herramientas disponibles, esto ayudaría a crear juegos más accesibles. Pero aún así, las dificultades en crear juegos accesibles tienen otras causas más allá de la falta de herramientas. A muchos estudios actualmente les cuesta crecer

económicamente y que sus juegos sean rentables, y en este sentido lograr que un juego sea accesible no aporta ningún beneficio económico, sino al contrario. Se vuelve más difícil prestar atención a la accesibilidad cuando se sabe que esto no aportará nada para volver más rentable el estudio. Según Nicolás, es importante que los juegos “mainstream” como “Counter-Strike”, “Fortnite” o “Minecraft” sean accesibles, ya que son juegos muy populares y con un valor cultural muy grande. Incluir a jugadores con discapacidad en estos tipos de juegos es algo importante y en donde hay que poner esfuerzo.

Francisco comenta que si bien se debería implementar accesibilidad en todos los juegos, es verdad que los equipos indies muchas veces no cuentan con los recursos disponibles para llevar a cabo esta tarea. Pero esto no significa que no se puedan construir juegos más accesibles, por ejemplo, muchas de las opciones de accesibilidad que parecen complejas son en realidad sencillas de implementar, o por lo menos, no tan costosas como parecen. Tal vez por un desconocimiento sobre el tema muchas veces los indies no implementan accesibilidad en sus juegos. Las empresas más grandes tienen presupuesto disponible para realizar investigación en accesibilidad, y recursos para poder realizar testing con jugadores, cosas que los indies no se pueden dar el lujo de tener.

Bruno también está de acuerdo con esta visión, y comenta su propuesta. Los desarrolladores deberían estar atentos a la accesibilidad que puede requerir un videojuego en particular, de acuerdo a las características del mismo. Es decir, dependiendo del tipo de juego, del género, de la cantidad de jugadores, pueden surgir unos u otros problemas de accesibilidad. Bruno está convencido que no es necesario que todos los juegos sean accesibles. Dependiendo del juego, algunas mecánicas pueden no ser naturalmente accesibles, por lo cual hay que encontrar un balance. *“Por un lado es muy bueno que el juego sea lo más accesible posible, pero no hay que perder de vista los límites planteados por las mecánicas del juego y los recursos disponibles”*. Y agrega: *“Todos los juegos deberían ser accesibles, pero sin limitar la capacidad creativa”*.

Sobre las pruebas y los jugadores con discapacidad

En cuanto a la etapa de testing de los juegos, se les preguntó a los desarrolladores sobre las metodologías que utilizan para probar sus juegos y qué tipo de jugadores están involucrados durante esta etapa.

Tanto Nicolás como Bruno comentan que no realizan pruebas en sus juegos a gran escala. Como son juegos “chicos”, el testing se realiza principalmente por los mismos desarrolladores. En ocasiones también suelen mostrar el juego a amigos y conocidos, para obtener un feedback más objetivo. Bruno agrega que por lo general las primeras pruebas que se llevan a cabo son sesiones de playtesting, para saber si el prototipo es divertido, si

las mecánicas funcionan de acuerdo a lo esperado, detectar secciones complicadas, y para balancear la dificultad. Más adelante las pruebas están más que nada dirigidas a detectar bugs y otros problemas técnicos que puedan aparecer en el juego y que hayan pasado desapercibidas por el desarrollador. Bruno agrega que a veces le pide feedback a un amigo daltónico, para asegurarse que los colores del juego sean fáciles de identificar para este tipo de jugadores. Con respecto a si encuentran útil el hecho de involucrar a jugadores con algún tipo de discapacidad en la etapa de testing, Bruno menciona que sí, pero que cree que es más importante ser consciente de las decisiones que se toman durante el diseño del juego, para conocer las consecuencias en la accesibilidad.

Francisco cuenta que en su ambiente laboral, el QA de los juegos que desarrollan queda a cargo del cliente, es decir que no llevan adelante el proceso de testing dentro del estudio. Los proyectos personales suele mostrarlos primero a amigos y conocidos, pero cuenta que es importantísimo mostrarlos también en eventos públicos. De esta manera cualquier persona puede acercarse y probar el juego, y el feedback que se consigue de estos jugadores suele ser muy útil. Francisco dice que nunca incluyó a jugadores con discapacidad durante las pruebas, aunque sí cree que es muy valioso hacer esto para poder detectar problemas que el desarrollador no ve en su propio juego.

Sobre las guías de accesibilidad

Como parte final de la entrevista, enumero algunos recursos online que sirven como guía para crear accesibilidad en videojuegos. Nicolás argumenta que el problema con este tipo de guías o recetas es que muchas veces te dan la solución final, te dicen lo que tenes que hacer, en lugar de explicar cuál es el problema y hacer hincapié en cómo afecta a los jugadores que se encuentran con ese problema. Según Nicolás, ayudaría mucho más conocer sobre el problema, comprender sus causas, y saber a qué tipo de jugadores pueden afectar, para poder tener una mejor base de toma de decisiones al momento de elegir si se debe o no implementar una opción de accesibilidad. Francisco tiene una opinión similar en cuanto a las guías de accesibilidad: *“Estaría bueno que hayan más recursos, pero hace falta empujar para que estos recursos expliquen también el por qué”*. Bruno también se expresó con respecto a las guías de accesibilidad: *“Si seguís las guías al pie de la letra el juego va a ser super accesible, pero va a limitar totalmente el diseño del juego”*. Por esto dice que es importante conocer sobre las problemáticas de accesibilidad primero, para poder tener más conocimientos al momento de elegir qué cosas implementar de todas las opciones que muestran las guías, o por lo menos para entender cómo van a afectar al jugador.

Conclusiones

En las entrevistas con desarrolladores independientes se encontraron algunas opiniones y puntos en común en cuanto al tratamiento de la accesibilidad, que se describen a continuación:

- **La accesibilidad es considerada, por lo menos en el ámbito de los desarrollos independientes locales, un balance entre el diseño accesible, los recursos disponibles y las limitaciones por las mecánicas del juego.** Bruno lo argumenta diciendo que si se cumple con muchas recomendaciones de las guías de accesibilidad, se podría limitar el diseño original del juego. También menciona que desde un punto de vista realista, las opciones de accesibilidad representan un peso extra en el desarrollo que muchos estudios independientes no están listos para soportar.
- **La implementación de opciones de accesibilidad, por una razón u otra, siempre queda relegada.** Una de las causas puede ser la falta de recursos, ya sea falta de recursos económicos, falta de recursos humanos que realicen la implementación o investigación sobre accesibilidad, o la falta de tiempo de desarrollo.
- **Implementar accesibilidad en algunos proyectos simplemente “no vale la pena”.** Ya sea porque el impacto de las opciones de accesibilidad sería mínimo entre la audiencia de jugadores, o porque no se justifica la inversión de recursos en implementar features que no generan mayor recaudación.

Para terminar me gustaría extraer tres conclusiones importantes de las entrevistas. En primer lugar tanto Bruno como Nicolás mencionaron que es importante trabajar la accesibilidad desde el diseño. Muchas veces, si no es posible implementar opciones de accesibilidad, se puede diseñar un juego que sea naturalmente accesible para los jugadores. Para esto es importante que el desarrollador cuente con el conocimiento adecuado durante la etapa de diseño del juego. En segundo lugar, Francisco y Bruno están de acuerdo en que muchas veces las soluciones de accesibilidad son sencillas y rápidas de implementar. Si los desarrolladores contaran con el conocimiento necesario para identificar los problemas de accesibilidad que se podrían presentar en sus juegos, y si entendieran la dificultad de implementar las opciones de accesibilidad, sería más común encontrar opciones de accesibilidad en los juegos. Por último, Francisco y Bruno se refieren a las guías de accesibilidad, comentando que muchas guías solo listan las soluciones. Es decir que simplemente mencionan qué es lo que hay que hacer para hacer accesible al juego, y no explican cuál es el problema original que se intenta solucionar, ni el impacto que tienen las barreras en el jugador.

Teniendo en cuenta estas conclusiones, en el siguiente capítulo se profundizará en la importancia del diseño de la accesibilidad, se analizarán algunas de las guías de accesibilidad disponibles, y se examinarán buenas prácticas de diseño de accesibilidad.

Capítulo 4: Diseño de accesibilidad en videojuegos

4.1 La importancia del diseño de la accesibilidad

Cuando hablamos de crear o diseñar accesibilidad e inclusión nos referimos a crear oportunidades y posibilidades para todos los jugadores y no solo para unos pocos, con el objetivo de promover la igualdad en las tareas que todos realizamos en el día a día (Cairns et al, 2019). Como se describió en el Capítulo 2, esto es un fenómeno dentro de la industria de los videojuegos que se viene desarrollando desde hace décadas y que, por suerte, sigue en constante desarrollo y transformación. De acuerdo a lo analizado, los videojuegos tienen su propia historia de desarrollo de la accesibilidad y, aunque hubo muchos avances, todavía no existe algo parecido a un estándar, y todavía hay mucho camino para recorrer (Shin, 2021).

Los videojuegos ocupan una parte muy importante en la sociedad actual. Además de ser una forma de entretenimiento, y una herramienta con grandes potenciales terapéuticos (Brooks et al, 2017), los videojuegos pueden construir comunidades y acercar personas. Las personas que viven con diversidad funcional suelen tener una vida más complicada, en algunos casos relacionadas al acercamiento entre pares y que pueden derivar en dificultades para la sociabilización (Conception, 2017). Para estos jugadores, los videojuegos ofrecen un sentido de pertenencia, y proveen un nivel de interacción, trabajo en equipo, comunicación y diversión, que no es sencillo de conseguir de otra manera y desde un entorno seguro para el jugador. Los beneficios que se pueden conseguir de la utilización de videojuegos como herramientas terapéuticas son variados y se pueden aplicar para tratar diversas patologías como el trastorno del espectro autista, trastorno del desarrollo intelectual, del aprendizaje, déficit de atención o hiperactividad y el Síndrome de Down (Gauci, 2021). Esta es una de las razones más importantes por la cual es necesario crear juegos accesibles, un juego con un diseño mal ejecutado podría dejar fuera a muchísimos jugadores que los necesitan por otras razones además del entretenimiento (Cairns, 2019).

Muchas veces las decisiones de los desarrolladores, aunque se tomen inconscientemente, llegan a influir en todas estas cuestiones que afectan al jugador. El objetivo de este capítulo es brindar información y ejemplos de algunos de los problemas de diseño que se presentan comúnmente en los videojuegos, y que generan barreras en los jugadores, con la intención de crear conciencia sobre el impacto y el alcance que pueden tener las decisiones de diseño del juego sobre cierto grupo de jugadores. Si bien existen varios lineamientos y guías para el desarrollo de la accesibilidad en videojuegos, son por lo general muy acotados

y solo se limitan a contar la solución al problema. En el Capítulo 3 varios desarrolladores mencionan que las guías de accesibilidad son útiles si se explican los razonamientos detrás de las soluciones propuestas y cómo éstas afectan a los jugadores. Yuan & Folmer (2008) sostienen que uno de los problemas más graves con las guías de accesibilidad es que no proveen una explicación acerca de qué problema de accesibilidad se supone que están intentando resolver, haciendo difícil para los desarrolladores entender cuándo y porqué se debería aplicar la guía o directriz (Yuan et al, 2010). Puesto que el objetivo de esta tesina es visibilizar la problemática de la accesibilidad, y lograr acercar soluciones a los desarrolladores independientes, es esencial brindar una explicación más extensa sobre las decisiones de diseño relacionadas a accesibilidad en los videojuegos provista por algunas de estas guías.

En el modelo social de la discapacidad, tener una discapacidad significa quedar excluido de las actividades de las que otros pueden ser parte libremente, debido a un desajuste entre el individuo y sus habilidades y el mundo diseñado (Shakespeare, 2013). Los juegos accesibles pueden ser enriquecedores para algunos jugadores, porque eliminan las barreras de discapacidad que los limita en otros aspectos de sus vidas. Esto lo experimentan tanto en la capacidad de poder realizar lo que el juego requiere de ellos, como en la habilidad de poder realizar también lo que otros jugadores hacen dentro del juego. Dejar el mundo físico y moverse al mundo virtual de los videojuegos les da a los jugadores con diversidad funcional los mismos desafíos que tienen todos los demás. Por supuesto, sin cuidado en el diseño de los juegos, es posible que los videojuegos trasladen las limitaciones del mundo real al virtual (Carr, 2010), o incluso que se lleguen a crear nuevas barreras inesperadas.

¿De qué forma podemos diseñar accesibilidad?

En cuanto al diseño e implementación de las estrategias de accesibilidad, existen dos grandes maneras de lograrlas:

- **Diseño inherentemente accesible**, aplicando los principios del diseño universal (Yuan et al, 2010). Durante la etapa de diseño del juego, se plantean posibles problemas de accesibilidad que podrían surgir en el futuro. Se trata de encontrar una manera en la cual un cambio realizado sobre el diseño original del juego elimine o reduzca dichos problemas. En el Capítulo 3 esta estrategia es la adoptada en la realización del juego “ScourgeBringer”. Esta es, en mi opinión, la mejor manera de resolver los problemas de accesibilidad, dado que permite producir un juego accesible naturalmente, posibilitando la inclusión de todos los tipos de jugadores sin necesidad de utilizar herramientas o configuraciones extras para poder jugar. Como analizamos en el Capítulo 2, la inclusión y la accesibilidad son conceptos relacionados pero que se refieren a ideas diferentes. En un juego inclusivo todos los

jugadores gozan de las mismas oportunidades al momento de jugar, dado que todo el sistema está diseñado contemplando las fortalezas y debilidades de cada jugador.

- **Agregar “accesibilidad” al juego**, transformando el diseño original y adaptándolo para facilitar la interacción a ciertos jugadores. Esta estrategia realiza modificaciones sobre el diseño original una vez que ya está implementado. Ejemplo de ello es la realización del juego “Skye Tales”, analizado en el Capítulo 3. A mi juicio, esta es una forma de implementar accesibilidad que, si bien tiene sus ventajas, parte de la presunción que nuestro juego no estará naturalmente disponible para todos los jugadores. La forma de lograr accesibilidad con este segundo enfoque, es ofrecer personalizaciones y configuraciones que se activan opcionalmente para modificar diversos aspectos del juego, de manera que facilite la interacción para algunos jugadores.

La elección de una u otra estrategia dependerá de varios factores. En principio no podemos limitar el diseño del juego en favor de la accesibilidad, y tampoco podemos limitar la accesibilidad por el diseño del juego. Mantener un balance entre las dos estrategias, nos ofrece la posibilidad de contar con las ventajas de ambas. Una de las conclusiones arribadas a partir de las entrevistas con desarrolladores independientes, fue que la accesibilidad muchas veces exige cambios que limitan la libertad creativa o artística original, algo que puede verse de forma negativa. Durante la entrevista con Bruno Martínez se menciona que una de las posturas que se pueden tomar es identificar las mejores opciones de accesibilidad que se ajusten de acuerdo al género o tipo de juego. Es decir, no existe un balance que se ajuste perfectamente a todos los tipos de juegos, sino que depende del juego en particular que estemos creando. Por ejemplo, algunos tipos de juego son más flexibles que otros, y soportan cambios en favor de la accesibilidad con mayor facilidad. Esto aplica principalmente a los juegos casuales (Yuan et al, 2010), o juegos de puzzles, en donde existe una única mecánica principal, suficientemente simple como para poder alterarse sin perder el sentido original del juego. Imaginemos por ejemplo un juego de ajedrez, donde la mecánica principal del juego es mover piezas de acuerdo a un conjunto de reglas determinadas. Los elementos involucrados pueden ser alterados de cualquier forma, cambiar el color de las piezas, cambiar el color del tablero, cambiar las piezas por símbolos o íconos, o incluso utilizar una representación no-visual para el juego (por ejemplo mediante sonidos). Estos cambios se pueden realizar sin perder el sentido original del juego. Por el contrario existen otros juegos en donde el diseño original puede ser menos flexible o directamente no aceptar alteraciones sin perder el sentido original de las reglas del juego. Un ejemplo de esto se da en el juego “The Witness” (2016), en donde uno de los puzzles necesarios para avanzar en el juego se basa exclusivamente en el sonido. Para poder resolver el puzzle es necesario que el jugador cuente con la habilidad de escuchar correctamente el sonido del puzzle. En este caso se puede hacer más accesible este

puzzle, pero seguramente la solución implique cambiar el diseño original del puzzle, como por ejemplo utilizar ayudas visuales para poder resolverlo. Si la intención de los desarrolladores era utilizar el sonido para resolver el puzzle, esta alteración de la intención original puede verse como una limitación creativa o artística.

La implementación de un diseño inclusivo trae consigo algunas ventajas para el desarrollador, entre ellas reducir el costo y el tiempo de desarrollo que se debe invertir a futuro para lograr accesibilidad. Durante la primera etapa de diseño, el proyecto es bastante flexible, y un cambio en alguno de los aspectos del juego tiene un impacto mucho más bajo que si se hubiera hecho en etapas posteriores. Adicionalmente, el jugador también puede obtener beneficios del diseño inclusivo: el juego será equitativo, flexible e intuitivo sin importar las condiciones o habilidades del jugador. Por otro lado, implementar opciones de accesibilidad a posteriori permite al desarrollador flexibilizar el desarrollo del proyecto, permite poder ajustar prioridades y dejar para después las tareas de implementación de accesibilidad. La personalización y customización del juego siempre es algo que pueden aprovechar los jugadores, tanto los que portan algún grado de discapacidad, como por el resto de los jugadores.

Efectos de la implementación de la accesibilidad en el desarrollo de proyectos de videojuegos

Según Ian Hamilton, especialista en accesibilidad en videojuegos, los diferentes extremos en la industria experimentan diferentes tipos de presiones y barreras para implementar accesibilidad en los juegos (Hamilton, 2018). En el extremo de los juegos independientes existen ideas equivocadas sobre el costo y tiempo que conlleva la implementación de soluciones accesibles, y también problemas derivados de la implementación tardía de la accesibilidad. En el otro extremo, el de los juegos AAA, se dan conceptos erróneos acerca del tamaño del mercado interesado en juegos accesibles, ideas equivocadas sobre las ganancias generadas a partir de la implementación de juegos accesibles, y falta de conocimientos acerca de cómo darle prioridad a la accesibilidad dentro de un backlog (pendientes) de tareas muy grande.

Diane Landais de "Accidental Queens", un estudio francés de desarrollo independiente de videojuegos, explica en una entrevista al portal web PocketGamer.biz que pensar en la accesibilidad no solo significa atender a un grupo determinado o con necesidades específicas de jugadores. Por el contrario, se trata de asegurarse de que la experiencia que se desea crear, pueda ser disfrutada por igual por tantas personas como sea posible, y en tantas situaciones como sea posible. La autora señala que *"Una de las cosas de las que me di cuenta mientras trabajaba en Alt-Frequencies, un juego narrativo de misterio basado en audio que ha sido diseñado para ser jugado por igual por jugadores con o sin dificultades"*

visuales, es que hacer un juego más accesible no significa comprometer o recortar features” (How do indie developers address accessibility in video games?, 2019). Hacer un juego accesible es difícil y costoso de reconciliar con el diseño si se trata de agregar funciones de accesibilidad encima de un juego que está originalmente diseñado para una audiencia específica.

Una manera importante en la que los juegos difieren significativamente de otras industrias, es que algunas de las barreras que se generan al jugar son deseables y necesarias para darle al juego reglas y desafíos, imprescindible para que un juego sea un juego (Ellis & Kao, 2019). Este tipo de barreras deseables son propias y específicas para cada juego, lo que hace que la accesibilidad del videojuego sea un proceso de optimización y no simplemente seguir una lista predeterminada de guías o lineamientos. Se optimiza la accesibilidad identificando, evitando, o eliminando tantas barreras innecesarias como sea posible, mientras se mantiene una experiencia agradable y desafiante para el jugador (Ellis et al, 2020).

Ser conscientes de las de las alternativas de diseño y saber cómo aplicarlas, permite a los desarrolladores de videojuegos tomar las decisiones necesarias para alcanzar un diseño inclusivo del juego y accesibilidad desde una etapa temprana del proceso. El reacondicionamiento en una etapa tardía del desarrollo puede ser algo mucho más caro y difícil, reduciendo el alcance de lo que puede realizarse y la efectividad de las soluciones (Ellis et al, 2020). Implementar accesibilidad puede ser un desafío intimidante. A veces los equipos de desarrollos en juegos AAA mencionan la cantidad de funciones de accesibilidad que tienen, o el número de desarrolladores que están trabajando en accesibilidad o investigando con el fin de demostrar el “nivel” de accesibilidad del juego (Baker, 2021). Para los equipos más pequeños puede parecer que hay muchas cosas que hacer para cumplir con algún tipo de estándar, pero como mencionamos anteriormente, no existe un estándar a cumplir. Según Ian Hamilton , no existe una barrera a alcanzar u objetivos específicos a cumplir. En lugar de eso, se debe ver la accesibilidad como un proceso de optimización del juego (Hamilton, 2021).

¿Cuándo se debería pensar en la accesibilidad?

La accesibilidad debe tenerse en cuenta en las primeras etapas de desarrollo (M. Baker, 2020). La razón detrás de esto es que, por lo general, una vez asentadas las bases fundamentales del diseño se complejiza la aplicación de la accesibilidad. Para entender esto mejor podemos examinar cuáles son las etapas de desarrollo en un videojuego. Dentro del pipeline de desarrollo de videojuegos, el producto pasa generalmente por cinco etapas: concepto, pre-producción, producción, lanzamiento, y post-lanzamiento.

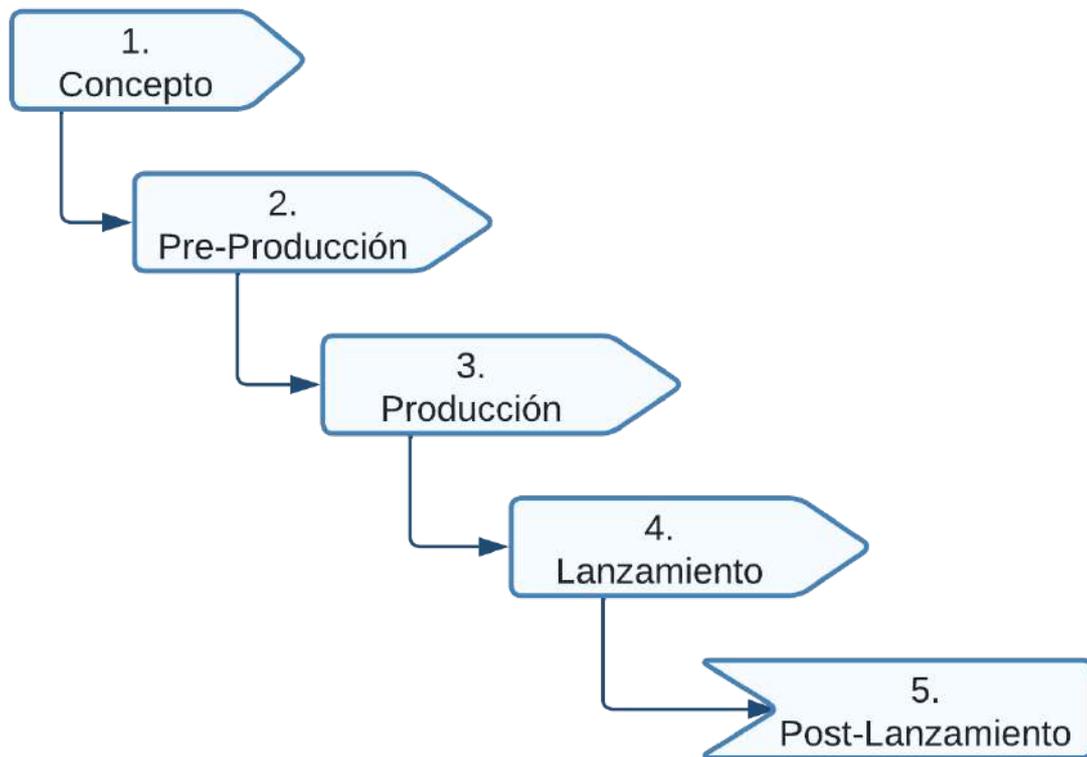


Figura 4.1 Etapas de desarrollo de un videojuego. (M. Baker, 2020)

En la primera etapa el juego es solo un concepto, se imagina la historia del juego, los personajes, etc. La etapa de pre-producción se centra en organizar los tiempos de desarrollo, delimitar la historia, completar el diseño de los personajes, las mecánicas de juego y el diseño de los niveles. En esta etapa el juego comienza a tomar forma y los detalles empiezan a quedar más definidos, dejando todo listo para la etapa de producción, en donde todos los recursos se terminan de mezclar para dar forma al videojuego (M. Baker, 2020).

Las herramientas de accesibilidad deben integrarse durante la etapa de pre-producción. Una vez que el juego ingresa en la etapa de producción, es muy difícil volver atrás si nos damos cuenta, por ejemplo, que todos los sonidos que diseñamos y creamos no son accesibles para las personas con dificultades auditivas. Ante un problema de este tipo podríamos pensar en agregar subtítulos, pero esto solo cubrirá los diálogos y efectos de sonido más comunes, dejando fuera otro tipo de sonidos clave²⁰. Las “Game Accessibility Guidelines” aconsejan realizar el trabajo de análisis y planeamiento de *features* de accesibilidad durante la etapa de diseño (Game Accessibility Guidelines, 2022). En los

²⁰ Las claves o señales auditivas son piezas de sonido que sirven para transmitir información crucial o para generar inmersión al jugador, por ejemplo el sonido de alerta de salud de un personaje.

estudios de desarrollo grandes, es común que se contrate un gerente o especialista en accesibilidad, encargado de crear y mantener un único punto de vista a través de toda la compañía, tomar decisiones y direccionar las soluciones implementadas hacia versiones más accesibles. Si el estudio es chico, como en el caso de los proyectos independientes, entonces puede ser difícil contar con un consultor a tiempo completo. En este caso, uno de los desarrolladores puede asumir la responsabilidad de realizar investigación activa sobre el tema y estar a cargo de conseguir que las soluciones se lleven a cabo.

4.2 Guías de accesibilidad en videojuegos

Las guías de accesibilidad en videojuegos fueron mencionadas varias veces durante el desarrollo de esta tesina como fuentes de información sobre la temática y como recursos útiles para la implementación de soluciones de accesibilidad. Muchas de las organizaciones y entidades que impulsan la concientización sobre la accesibilidad en los videojuegos incluyen entre sus iniciativas a las guías de accesibilidad. A continuación veremos las características y objetivos de algunas de las guías más importantes disponibles en la web.

WGAC

Las Web Content Accessibility Guidelines (Guías de Accesibilidad para el Contenido Web, en inglés) son un conjunto de guías y lineamientos que especifican cómo se debe crear el contenido web para hacerlo más accesible para todas las personas, especialmente para las personas con discapacidades. Esta guía tuvo varias actualizaciones a lo largo de los años, agregando nuevos lineamientos y criterios de cumplimiento. Aunque las guías están creadas específicamente para el contenido web, algunas de las técnicas y lineamientos que detalla se pueden aplicar a otro tipo de aplicaciones, incluyendo a los videojuegos, por ejemplo en juegos para dispositivos móviles (Barlet et al, 2019). Esto es así dado que incluye lineamientos y guías detalladas cómo desarrollar la UI de las aplicaciones, tanto desde el aspecto gráfico como en la forma de interacción de los usuarios, algo que se puede aprovechar para realizar interfaces y menús para videojuegos (Wilson & Crabb, 2018).

Game Accessibility Guidelines

Las Guías de Accesibilidad de Videojuegos (<https://gameaccessibilityguidelines.com/>) fueron creadas inicialmente en el año 2012, fruto del esfuerzo de un grupo conformado por múltiples desarrolladores de estudios de videojuegos, especialistas en accesibilidad y académicos, con el objetivo de facilitar la inclusión de opciones de accesibilidad en videojuegos. A diferencia de las WCAG, estas guías están específicamente orientadas a la

accesibilidad en videojuegos, y fueron creadas para proveer a los desarrolladores referencias escritas en un lenguaje familiar, fáciles de comprender y de utilizar.

APX Design Patterns

En 2019, la fundación AbleGamers publicó su guía de accesibilidad llamada “APX Design Patterns” (<https://accessible.games/accessible-player-experiences/design-patterns/>), un conjunto de recomendaciones enfocadas en conseguir que los estudios de desarrollo comiencen a pensar de manera integral sobre la discapacidad y las barreras que impiden a los jugadores interactuar con los videojuegos. Estas recomendaciones están implementadas en forma de una colección de patrones de diseño, categorizadas en dos grandes grupos: patrones de acceso y patrones de desafíos. Según AbleGamers estos dos niveles de patrones se pueden combinar para poder lograr construir experiencias realmente accesibles para los jugadores.

XBox Accessibility Guidelines

Estas guías (<https://docs.microsoft.com/en-us/gaming/accessibility/guidelines>) fueron creadas por Microsoft en conjunto con especialistas en accesibilidad, desarrolladores y miembros de la comunidad de jugadores con discapacidades. Fueron creadas de manera que sirven como guía de diseño, implementación, y validación para diseñadores, desarrolladores y testers por igual.

Estas son algunas de las guías de accesibilidad que se pueden encontrar en la web. Aunque existen muchas otras, éstas son las que personalmente encuentro más útiles, por la calidad del contenido y por las entidades y organismos que las crearon y mantuvieron actualizadas a través de los años.

4.3 Catálogo de buenas prácticas en el diseño de accesibilidad en videojuegos

Motivaciones

Para esta tesina se propone desarrollar una estrategia metodológica orientada al diseño de la accesibilidad, que permita a los desarrolladores independientes mitigar y/o solucionar problemas de accesibilidad en sus videojuegos. Con este propósito se dispone a crear un catálogo, compuesto de una serie de buenas prácticas y soluciones a problemas comunes que se pueden encontrar durante el diseño de videojuegos y que atentan contra la accesibilidad. Considero que los recursos disponibles que brindan soporte sobre la construcción de soluciones de accesibilidad se apoyan sobre todo en barreras de interacción con el juego ligadas a diferentes discapacidades además de la auditiva. Por esta

razón, para esta tesina se propuso trabajar principalmente sobre los problemas de diseño que constituyen barreras para los jugadores con discapacidades auditivas. Aunque se considerarán otros tipos de discapacidades para la creación de este catálogo, sólo se detallarán las buenas prácticas y soluciones a las barreras de accesibilidad relacionadas con la discapacidad auditiva.

Es necesario aclarar que esta es una lista creada estrictamente para esta tesina, en base a mi experiencia previa en el desarrollo de videojuegos, en los conceptos incorporados durante la investigación para el desarrollo de esta tesina, y en mi experiencia como jugador con discapacidad auditiva. Dicho esto, algunas de las buenas prácticas listadas en este catálogo también pueden encontrarse en otras guías de accesibilidad. La intención de este catálogo no es evidenciar nuevas prácticas que no hayan sido mencionadas anteriormente en otros catálogos o guías de accesibilidad, sino sumar información acerca del impacto que tienen las decisiones de diseño sobre el jugador, y cómo pueden afectar al desarrollo del proyecto. La intención última de este catálogo es que sea de utilidad para los desarrolladores de videojuegos independientes para poder identificar problemas, tomar decisiones conscientes, y poseer el conocimiento necesario para poder considerar el punto de vista de la accesibilidad desde las primeras etapas de diseño del videojuego.

Estructura del catálogo

La estructura propuesta para el catálogo de buenas prácticas de diseño consta de ocho secciones:

1. **Título:** Nombre descriptivo de la pauta.
2. **Resumen:** Breve descripción que explica el problema de diseño.
3. **Problema:** Detalle del problema. Responde a las preguntas: ¿Cómo afecta a los jugadores? y ¿Cuáles son las causas más comunes del problema?.
4. **Solución:** Detalle de la solución. Responde a las preguntas: ¿Cómo se puede implementar? y ¿Qué otras variantes de la solución existen?
5. **Beneficios:** Ventajas extras que se consiguen con la aplicación de la solución. ¿A qué otro tipo de jugador puede beneficiar? ¿Cómo afecta al desarrollo del proyecto?
6. **Clasificación:** Clasificación de la pauta en las cuatro categorías propuestas y que se describen en este mismo apartado.
7. **Ejemplos:** En videojuegos reales en donde se puede experimentar el problema. Videojuegos en donde se aplicó la solución.
8. **Herramientas:** Guías, lineamientos y material extra que agrega más información sobre el problema. Permite conocer y ayudar a identificar el problema con mayor facilidad.

La organización de este catálogo se realizará tomando como base algunas de las clasificaciones propuestas por las guías de accesibilidad anteriormente mencionadas.

La clasificación de las pautas presentará cuatro diferentes categorías:

- A. **Grupo de discapacidad afectada:** Visual, Auditivo, Motriz o Cognitivo
- B. **Momento de interacción que se ve afectado:** En el modelo de interacción entre el jugador y el juego propuesto por Yuan y Folmer (2008), momento del ciclo que se ve afectado por el problema:
 - 1. Al recibir el estímulo
 - 2. Al determinar la respuesta
 - 3. Al ingresar la entrada
- C. **Principio de Diseño Universal:** A cuál de los principios del Diseño Universal apunta la solución.
- D. **Dificultad de implementación:** Categorizado en básico, intermedio y avanzado, según las GAG (Game Accessibility Guidelines, 2012).

Podemos clasificar las pautas de este catálogo de acuerdo a qué tipo de evento corresponde dentro del modelo de interacción propuesto por Yuan-Folmer (2008):

- 1. **Recibir estímulo:** la pauta apunta a solucionar dificultades que el jugador tiene para recibir la información que el juego intenta transmitir.
- 2. **Determinar respuesta:** la pauta trata de apoyar al jugador a procesar la información más fácilmente, con el objetivo de facilitar la identificación de la respuesta que el juego requiere.
- 3. **Proveer input:** la pauta intenta simplificar la cantidad o cualidad de las acciones requeridas por el juego para poder avanzar.

Adicionalmente vamos a clasificar las pautas de acuerdo a los principios del Diseño Universal:

- 1. **Igualdad de uso:** la solución ayuda a que la interacción con el juego sea la misma para todos los jugadores.
- 2. **Uso flexible:** la solución apunta a crear juegos flexibles, por lo general utilizando configuraciones y opciones de accesibilidad para cambiar la forma de interacción con el juego.
- 3. **Uso simple y funcional:** el objetivo es conseguir una interacción sencilla con el juego, enfocándose en el funcionamiento en lugar del diseño.
- 4. **Información comprensible:** se apunta a simplificar el procesamiento de la información para el jugador de manera que la pueda comprender más fácilmente.
- 5. **Tolerancia al error:** la finalidad es mejorar la entrada del jugador o asistirlo de cierta manera que permita controlar el juego con más precisión.

6. **Bajo esfuerzo:** se apunta a disminuir el esfuerzo que debe realizar el jugador para poder interactuar con el juego.
7. **Dimensiones apropiadas:** para que el jugador pueda interactuar con el juego a través de la interfaz de usuario (botones, menús, navegación).

En cuanto a la dificultad de implementación, se tomaron como guía los lineamientos propuestos en las Game Accessibility Guidelines. La clasificación propuesta para estas pautas se corresponden con tres categorías: **básico**, **intermedio** y **avanzado**. Estos niveles están basados en el balance entre tres aspectos diferentes:

- **Alcance:** número de personas a quién beneficia.
- **Impacto:** efecto que produce en dichas personas.
- **Valor:** costo de implementación.

Catálogo principal

Subtitular los diálogos críticos

A. Resumen

En el caso de los diálogos en un videojuego, es importante que la totalidad de los mismos, o al menos de los diálogos más importantes, estén subtitulados. De lo contrario, los jugadores con dificultades auditivas podrían experimentar barreras para interactuar con el videojuego.

B. Problema

Para entender mejor el problema, imaginemos un juego en donde el jugador debe atravesar una puerta custodiada por un guardia. Cuando el jugador se acerca a la puerta, el guardia tiene una línea de diálogo: “No puedes pasar por aquí, primero tienes que recoger tu equipo”. Como el equipo de desarrollo del juego es muy pequeño y con muchas tareas en su backlog de desarrollo, se decidieron a implementar subtítulos pero solo para los personajes principales del juego. Como consecuencia, el diálogo de este personaje secundario (guardia NPC²¹), no está subtitulado. Un jugador con dificultades para escuchar podría tener problemas para interpretar lo que el juego requiere en este momento, generando una barrera que podría potencialmente impedir que el jugador avance sin la ayuda de un tercero.

Este problema se presenta, por lo general, en juegos que deciden subtitular solo una parte del juego, o en juegos que subtitulan solo las cinemáticas. Es importante remarcar que además de ser una barrera para los jugadores con dificultades auditivas, puede generar problemas y llegar a causar frustración en jugadores que prefieren jugar sin sonido, o que se encuentran en alguna circunstancia en donde no pueden usar el volumen del juego muy fuerte, por ejemplo cuando se está viajando en un medio de transporte público.

²¹ NPC por las siglas en inglés de “Non-Playable Character”, un personaje controlado por la computadora.

C. Solución

La solución a este problema es obvia, implementar subtítulos para todos los diálogos críticos del juego. Con “diálogos críticos” se hace referencia a diálogos que contienen información crítica o necesaria para poder avanzar en el juego. Esto puede incluir información necesaria para el gameplay, como el diálogo del guardia NPC, o información sobre la historia o el lore²² del juego.

Este problema puede resolverse de varias maneras y en varios momentos del desarrollo. Siempre es preferible considerar la implementación de los subtítulos desde las primeras etapas de desarrollo del juego, aunque puede suceder que se olvide esta *feature*, o se opte por dejarlo para una etapa tardía del desarrollo. Si se piensa en los subtítulos en una etapa temprana, es más sencillo diseñar los recursos de diálogo del juego como si fueran un par audio-texto, donde cada archivo de audio del diálogo está acompañado por su correspondiente transcripción a texto. Si esto lo realizamos para todos los diálogos posibles desde el momento en el que son creados, no vamos a llegar a una situación problemática como la presentada en el ejemplo anterior. Por otra parte, si se decide realizar la implementación de los subtítulos en una etapa tardía, se debería realizar la transcripción de todos los archivos de audio de diálogo del juego o, para simplificar la tarea, realizarlo solo con algunos de los audios. En cualquier caso, es recomendable llevar adelante una ronda de playtesting para asegurarnos que la información transmitida al jugador mediante los subtítulos no está incompleta o es confusa. Aunque es muy recomendable realizar pruebas con jugadores con dificultades auditivas y visuales, también es sencillo detectar este problema simplemente apagando el sonido o bajando el volumen del juego.

D. Beneficios

Aunque no se trate de un videojuego centrado en los diálogos, como por ejemplo una aventura gráfica, es una buena práctica pensar en cómo se implementarán los subtítulos para todo el juego: diálogos de personajes, narraciones y cinemáticas, dado que presenta varias ventajas. Por otra parte, los desarrolladores cuentan desde el principio del desarrollo con la lista completa de diálogos, permitiendo planificar mucho mejor el desarrollo. Si bien esta lista puede cambiar con el tiempo, es una ventaja poder contar en las primeras etapas del desarrollo con un mecanismo para mostrar subtítulos dentro del juego, y ahorrarse el costo extra y los dolores de cabeza que se pueden dar si se tratan de encajar los subtítulos en un juego que se encuentra en las últimas etapas de desarrollo.

Por lo general los juegos se lanzan con soporte para varios idiomas, por ejemplo, se puede jugar en inglés o español. Realizar el doblaje de voces para los diálogos de todos los personajes puede ser costoso en cuanto a recursos como el tiempo y dinero, por lo cual en

²² El lore del juego es el conjunto de narrativas secundarias que apoyan y enriquecen a la historia principal del juego.

la mayoría de los casos se elige únicamente proveer soporte del idioma a través de los subtítulos. Una ventaja al momento de realizar la traducción del juego a otros idiomas es contar con todas las líneas de diálogo transcritas, dado que facilita la organización y la rapidez de las traducciones de los subtítulos. Para los jugadores que no hablan el idioma original de nuestro juego esto también representa un beneficio, asegurándonos que todo lo necesario para poder avanzar en el juego lo pueden obtener a través de los subtítulos.

E. Clasificación

- A. **Grupo de discapacidad:** Auditiva
- B. **Momento de interacción:** 1. Al recibir el estímulo
- C. **Principio de Diseño Universal:** 1. Igualdad de uso, 4. Información comprensible
- D. **Dificultad de implementación:** Intermedio

F. Ejemplos

En “Assassin 's Creed: Valhalla” (2020) algunos diálogos durante el gameplay no están subtitulados (figura 4.2). Por ejemplo, cuando el personaje principal se acerca a una zona con enemigos, dice en una voz susurrante: “Necesito ser silencioso... no quiero alertar a estas personas”. Este diálogo no está subtulado, y por lo tanto algunos jugadores con dificultades auditivas podrían tener problemas para recibir esta información. Esto produce que ingresen a las zonas con enemigos de una forma ruidosa, cuando la intención de los desarrolladores era que se utilice el sigilo.



Figura 4.2 El personaje principal se acerca a una zona inexplorada y dice “Necesito ser silencioso”, pero no existen subtítulos para este diálogo. Assassin's Creed: Valhalla (2020)

Por el contrario, el juego “Night in the Woods” (2017) (figura 4.3) presenta los diálogos de todos los personajes en forma subtitulada. Al ser un juego que se enfoca mucho en la

historia y en los personajes, es imprescindible lograr que el jugador entienda la trama completa a través de los diálogos de los personajes.



Figura 4.3 Todos los diálogos de “Night in the Woods” (2017) están subtítulos mediante burbujas de diálogo.

G. Herramientas

- Dialogic, un plugin para Godot Engine que facilita el diseño de diálogos
<https://github.com/coppolaemilio/dialogic>
- Xbox Accessibility Guidelines, la sección de subtítulos incluye guías y recursos útiles para implementar subtítulos en los juegos
<https://docs.microsoft.com/en-us/gaming/accessibility/xbox-accessibility-guidelines/104>

Implementar subtítulos cerrados (“closed caption”)

A. Resumen

Adicionalmente a los diálogos subtítulos, es una buena práctica ofrecer subtítulos descriptivos, también llamados “closed-caption”, para describir eventos de índole auditiva que son importantes para el jugador. Los jugadores con deterioro auditivo parcial o total pueden necesitar una alternativa para mejorar la percepción de la información visual o auditiva presentada por el juego.

B. Problema

Este problema no está dado por la falta de subtítulos sobre los diálogos de los personajes, sino que se enfoca en la falta de subtítulos sobre cualquier tipo de evento exclusivamente auditivo que, aunque no es esencial para el jugador, sí es importante para procesar y entender el contexto dentro del juego. En este grupo podemos incluir cualquier tipo de sonido que se utiliza para enfatizar o comunicar información importante para el jugador:

disparos, sonidos de alarmas, alertas, sonidos del ambiente, sonidos out (fuera de campo de visión), golpes, etc. Estos tipos de sonido sirven para terminar de completar el contexto del juego, y la falta de subtítulos para los mismos puede constituirse en una barrera para los jugadores con deterioro auditivo.

Por ejemplo, imaginemos un juego donde tenemos una palanca que abre una puerta, la cual el jugador debe atravesar para avanzar. Al utilizar la palanca, vemos que la puerta se abre, y escuchamos el sonido “tic-tac” de un reloj. Un jugador que escucha ese sonido entiende entonces que la puerta se va a mantener abierta solo por un período de tiempo corto, y que luego se cerrará, por lo que deberá acelerar el paso para poder cruzarla. Un jugador con deterioro auditivo no escuchará este sonido y por lo tanto no comprenderá el cambio de contexto del juego, por lo que no acelerará el paso y encontrará la puerta cerrada. Esto puede ser todavía peor si la puerta se cierra fuera de pantalla, lo que aumentará la confusión (y hasta frustración) del jugador cuando la encuentre cerrada.

C. Solución

El término “closed captions” generalmente se utiliza para referirse a los subtítulos, aunque son dos cosas diferentes. Los subtítulos cerrados, o “closed captions” están pensados no solo para el diálogo sino también para otros efectos de sonido. A menudo los subtítulos cerrados incluyen indicadores de música o sonidos en segundo plano (por ejemplo explosiones, o sonidos de vidrio rompiéndose). Como herramienta, los subtítulos cerrados están diseñados bajo el supuesto que el espectador es sordo o tiene dificultades para escuchar, por lo que su intención es la de proveer igualdad de acceso a la información auditiva.

En el ejemplo de la palanca y la puerta, si mostramos en pantalla un subtítulo indicando que “se escucha el tic-toc de un reloj” cuando se activa la palanca, todos los jugadores, hayan escuchado el sonido o no, entenderán que deben apurarse. Los subtítulos “closed caption” o subtítulos cerrados se utilizan en este caso para describir esta información contextual de forma visual mediante un texto o descripción que acompaña al sonido.

Los elementos sonoros que son importantes para la narrativa se incluyen en los subtítulos cerrados y se escriben entre corchetes “[]”. La información contextual de cada línea de diálogo la podemos escribir entre paréntesis“()”, como por ejemplo “(AL TELÉFONO)”, o “(CON ACENTO FRANCÉS)”.

En cuanto al formato, los efectos de sonido se deben indicar con una etiqueta completamente en mayúscula, en una nueva línea, y alineado a la izquierda. Además se espera que las etiquetas tengan la estructura “sujeto + verbo”, por ejemplo “[HOMBRE GRITA]”.

Existen algunos lineamientos y estándares a tener en cuenta cuando se crean subtítulos cerrados. Por ejemplo, según la BBC (BBC Subtitle Guidelines, 2022) solo se deben

subtitular los efectos de sonido críticos para que el espectador pueda comprender los eventos en la pantalla, o para entender la ambientación o atmósfera de la escena. Muchas veces se ven transcripciones innecesarias, ya que en la imagen visual se deja clara la información. Por ejemplo, en algunos casos el sonido de un perro ladrando puede ser esencial para terminar de entender la escena, y en otros casos no lo es. Otro lineamiento de la BBC para crear subtítulos cerrados es describir los sonidos, no las acciones. Por ejemplo, si en la pantalla podemos ver una escena de disparos, la etiqueta de subtítulo cerrado debería decir: “DISPAROS”, en lugar de “SE DISPARAN ENTRE ELLOS”.

D. Beneficios

Este tipo de subtítulos es muy útil para brindar información contextual al jugador, creando una inmersión mucho más profunda, y evitando barreras en el juego. Adicionalmente al utilizarlos estamos creando un diseño de juego más robusto, dado que permite tanto a jugadores con deterioro auditivo como a jugadores que simplemente les gusta jugar con el sonido apagado, tener una experiencia de juego similar.

Una ventaja de los subtítulos descriptivos es que nos permite solucionar, de una forma sencilla (y en cierto sentido no de la manera más correcta) la información monocanal. Está claro que el problema de la palanca y la puerta es un problema de información monocanal, dado que se está informando el cambio de contexto únicamente mediante el audio. La mejor manera de resolverlo sería indicar de forma visual, por ejemplo mediante un ícono, o mediante un contador que muestre el tiempo restante. Dependiendo del caso particular que se quiera resolver puede que resulte complicado el desarrollo, ya sea por falta de tiempo, por falta de recursos (por ejemplo no contamos con las imágenes o íconos necesarios), o por dificultad de implementación. En este caso, utilizar subtítulos descriptivos puede resultar una solución más sencilla de implementar.

E. Clasificación

- A. **Grupo de discapacidad:** Auditiva
- B. **Momento de interacción:** 1. Al recibir el estímulo
- C. **Principio de Diseño Universal:** 1. Igualdad de uso, 4. Información comprensible
- D. **Dificultad de implementación:** Intermedio

F. Ejemplos

The Quarry (2022) es un videojuego de terror y drama interactivo, fuertemente basado en la narración y en las decisiones del jugador para contar la historia. Es básicamente una película interactiva de terror, y como tal las emociones en la voz de los personajes son necesarias para lograr una mejor inmersión del jugador. Es por esto que la utilización de subtítulos descriptivos es esencial para transmitir la emoción a los jugadores con dificultades auditivas. En la figura 4.4 se puede observar el uso de la etiqueta “(heavy breathing)” para transmitir la sensación de los personajes en una de las escenas del juego.



Figura 4.4 Subtítulo cerrado que indica “heavy breathing” en el juego The Quarry (2022), esencial para transmitir las sensaciones de la voz de los personajes a los jugadores con dificultades auditivas.

En varios de los juegos de Valve, como los de la saga “Half-Life”, “Portal” y “Left 4 Dead”, se incluyen subtítulos descriptivos para todos los efectos de sonido del juego. “Left 4 Dead”, por ejemplo, es un FPS²³ en donde cuatro jugadores tienen que sobrevivir a un apocalipsis zombie. En este juego los jugadores cubren un terreno muy grande de juego, por lo que suele suceder que un jugador experimente eventos que el resto no, por estar alejado. Por ejemplo cuando aparece un enemigo, algunos jugadores lo verán y el resto no. Debido a la naturaleza cooperativa del juego, es necesario que los jugadores se ayuden mutuamente para lograr sobrevivir y ganar la partida. Para que los jugadores estén al tanto de la aparición de enemigos se utilizan efectos de sonidos característicos de cada tipo de enemigo, los cuales los jugadores pueden escuchar e interpretar. Estos sonidos representan información crítica para el jugador, por lo que tienen su equivalente en subtítulo descriptivo. En la figura 4.5 se puede observar la implementación de subtítulos descriptivos para los efectos de sonido del enemigo “Spitter” y de su ataque con ácido, información sumamente importante que el jugador necesita conocer.

²³ First Person Shooter: un género de videojuegos que se caracteriza por basarse en disparar usando una cámara en primera persona.

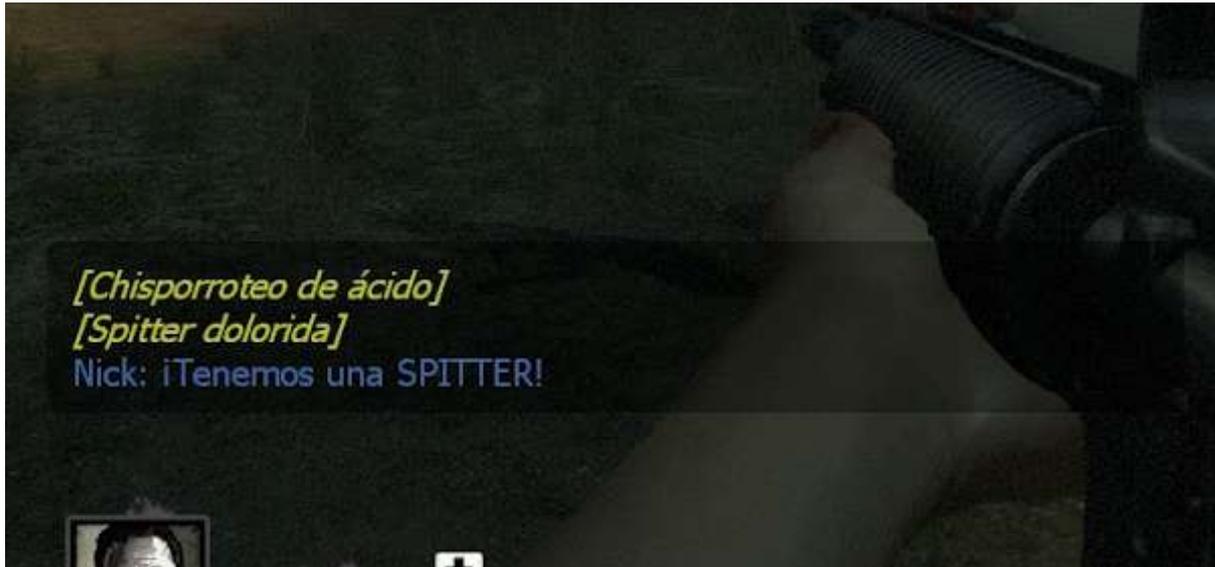


Figura 4.5 En “Left 4 Dead 2” (2009) se indican los efectos de sonido mediante subtítulos descriptivos.

G. Herramientas

- **Godot Subtitles Plugin**, un plugin desarrollado para esta tesina que permite la creación de subtítulos descriptivos en juegos hechos en Godot Engine (<https://github.com/FEDE0D/godot-subtitles>)
- **Guía de la BBC** para subtítular efectos de sonido y crear subtítulos descriptivos (<https://www.bbc.co.uk/accessibility/forproducts/guides/subtitles/#Sound-effects>)

Facilitar la identificación del hablante

A. Resumen

El jugador encuentra problemas para identificar el personaje que está hablando. Los jugadores con dificultades visuales o cognitivas pueden necesitar un refuerzo adicional para poder determinar el origen de las líneas de diálogo.

B. Problema

En los casos donde el jugador tiene problemas para escuchar el doblaje de los diálogos, no entiende el idioma o los acentos de los personajes, se puede crear una barrera en donde el jugador no identifica el origen de las líneas de diálogo. Adicionalmente en algunos tipos de juegos, como los RPGs, aparecen muchos personajes en pantalla a la vez, esto también puede ser un problema para identificar cuáles de los personajes es el que está hablando.

C. Solución

De acuerdo a especificaciones técnicas sobre la implementación de subtítulos creada por Microsoft para los juegos en Xbox (Xbox Accessibility Guidelines), los subtítulos deben incluir el nombre del personaje que está hablando al inicio de la línea de diálogo. Esto debe mostrarse solo la primera vez que habla el personaje, pudiendo removerse en las líneas

subsecuentes. También se debe incluir el nombre en casos donde el diálogo se retoma después de ser interrumpido por el diálogo de otro personaje.

Adicionalmente se puede utilizar el color para representar el vínculo con el personaje, en donde cada personaje tiene su color característico para los subtítulos. Otras opciones de vínculo con el personaje que se pueden utilizar son por ejemplo un indicador en la dirección donde está el personaje que habla mediante flechas o líneas, o la utilización de burbujas de diálogo como en las historietas.

Algunos juegos muestran una imagen del personaje que está hablando junto a los subtítulos, de manera que se entiende claramente qué personaje es el que está relacionado con las líneas de diálogo mostradas en ese momento.

D. Beneficios

Para los jugadores con deterioro auditivo completo puede ser crucial tener la posibilidad de identificar el personaje por el nombre o el color, sobre todo en juegos fuertemente basados en diálogos, como en las aventuras gráficas. Adicionalmente ofrecer a los jugadores un control más fino sobre el tipo de colores elegidos para los personajes, el tamaño y posición de los subtítulos puede ayudar a jugadores con algún tipo de deterioro visual, sobre todo los que poseen baja visión y trastornos de la percepción del color a utilizar más cómodamente el juego. Algunos juegos muestran los subtítulos cerca de los personajes cuando hablan, una práctica que puede ayudar en la identificación del personaje, pero puede ocasionar otro tipo de problemas en jugadores con baja visión, sobre todo cuando esperan que el diálogo aparezca como lo hace normalmente en el área central baja de la pantalla.

Además de utilizar nombres, burbujas de diálogo o colores para identificar al hablante en los subtítulos, se pueden utilizar herramientas de otro tipo para facilitar la identificación de los personajes. Algunos ejemplos incluyen:

- Utilizar bordes para destacar al personaje, muy útil cuando el personaje es muy pequeño en pantalla, o hay múltiples personajes.
- Mostrar al personaje en primer plano, o aumentar el tamaño en pantalla.
- Realizar una animación para ayudar a identificar al personaje hablante, este es un recurso muy simple y fácil de realizar, pero muy efectivo.

E. Clasificación

- A. **Grupo de discapacidad:** Auditiva
- B. **Momento de interacción:** 1. Al recibir el estímulo
- C. **Principio de Diseño Universal:** 1. Igualdad de uso, 4. Información comprensible
- D. **Dificultad de implementación:** Intermedio

F. Ejemplos

Uno de los ejemplos más conocidos es el del videojuego "Minecraft" (figura 4.6) en el cual a pesar de no haber diálogos entre personajes, existen efectos de sonido que son críticos

para el jugador. Los subtítulos en este juego, que en realidad son subtítulos descriptivos, incluyen indicadores de dirección, que apuntan al lugar en el que se producen los sonidos.



Figura 4.6 En "Minecraft" se utilizan indicadores de dirección con los subtítulos descriptivos.

"Nine Witches: Family Disruption" (2020), es una aventura gráfica con diseño pixel art creada por el estudio bonaerense "Indiesruption". En este juego es muy sencillo entender qué personaje es el que está hablando, dado que se utilizan burbujas de diálogo que apuntan al personaje (figura 4.7).



Figura 4.7 "Nine Witches: Family Disruption" muestra todo el diálogo mediante burbujas de diálogo.

Otra manera de indicar el personaje que está hablando es mediante la utilización de colores. Un ejemplo de esto se puede ver en el juego "Grim Fandango" (1998) de

LucasArts, en donde el diálogo se pinta con un color diferente para cada personaje. Algo que podemos mencionar sobre los subtítulos en este juego es que no respetan la posición recomendada para los subtítulos (zona central inferior de la pantalla), sino que se muestran al lado del personaje. Mientras que esto puede parecer una buena práctica para identificar al personaje que está hablando, en realidad puede generar confusión al jugador, dado que no puede predecir en qué lugar de la pantalla aparecerá el subtítulo. Para la versión remasterizada del juego (Grim Fandango, 2015, Double Fine) se eliminó esta práctica y se optó por mostrar los subtítulos siempre en la parte baja central de la pantalla (figura 4.8). Además en la versión remasterizada también se mejora el ajuste del texto, dado que no se generan líneas de diálogo innecesarias.



Figura 4.8 Comparación de subtítulos en “Grim Fandango”. Arriba la versión original de 1998 con problemas en la ubicación del texto y problemas en el corte de línea. Debajo el remaster del año 2015, con ambos problemas solucionados.

Este problema lo podemos encontrar en juegos más recientes, como por ejemplo en el videojuego independiente “Darkestville Castle” (2017) desarrollado por el estudio cordobés

“Epic Llama”. En este juego de género aventura gráfica se utiliza un estilo para los subtítulos similar al de “Grim Fandango” original, en donde los subtítulos no respetan la posición recomendada y en su lugar se muestran sobre la cabeza de los personajes cuando hablan. Algo que se puede remarcar de este ejemplo es también el mal diseño de los subtítulos en cuanto al color del texto y a la tipografía. Ambas cosas dificultan su lectura, aunque esto lo revisaremos en el siguiente punto del catálogo.



Figura 4.9 Ejemplo del diálogo implementado en “Darkestville Castle” (2017). Vemos los mismos problemas que en Grim Fandango (1998).

Por último, una de las prácticas más comunes en videojuegos, es mostrar una imagen de la cara del personaje junto con las líneas de diálogo. Un ejemplo de esto lo podemos encontrar en el videojuego argentino “Evan’s Remains” (2020) de Matías Schmied, en donde los diálogos se muestran junto con una imagen del personaje, de manera que no hay confusión con respecto a la identificación del origen de las líneas de diálogo.



Figura 4.10 Ejemplo de diálogo en “Evan’s Remains” (2020)

Este recurso también se utilizó en “Juanito Arcade Mayhem” (2017), un videojuego argentino de acción y arcade del estudio “Game Ever Studio”.



Figura 4.11 Ejemplo de diálogos de los personajes en “Juanito Arcade Mayhem” (2017)

G. Herramientas

- **Godot Subtitles Plugin**, un plugin desarrollado para esta tesina que permite la inclusión del nombre del personaje antes de su línea de diálogo, además de utilizar colores únicos para cada personaje. (<https://github.com/FEDE0D/godot-subtitles>)
- **Guía de la BBC** para identificar a los hablantes
<https://www.bbc.co.uk/accessibility/forproducts/guides/subtitles/#identifying-speakers>

Diseñar correctamente los subtítulos

A. Resumen

El subtítulo debe ser diseñado correctamente, de forma que pueda cumplir su objetivo sin generar nuevas barreras para el jugador. Existen errores en la implementación de los subtítulos que no son técnicos, sino que se producen por no respetar el estándar o lineamiento sugerido para los subtítulos.

B. Problema

Los subtítulos son una herramienta que permite a los jugadores con dificultad auditiva poder acceder a la información sonora en forma visual. En algunas ocasiones los desarrolladores diseñan los subtítulos sin respetar este objetivo de accesibilidad, lo que produce barreras innecesarias e inesperadas entre el jugador y el juego.

Uno de los problemas más comunes es el problema del tamaño. Algunos juegos incluyen subtítulos que son muy pequeños para poder ser leídos con facilidad. El tamaño es un problema para los jugadores que tienen dificultades visuales, o que están jugando en una pantalla pequeña como la de un dispositivo móvil, o una pantalla que está muy lejos como cuando se juega con un mando inalámbrico alejado de la televisión.

Otro problema bastante común en los subtítulos es la falta de contraste. Es un problema que se puede dar en otros elementos de la interfaz de usuario, por ejemplo en el menú del juego. En este problema el color del texto no tiene una relación de contraste adecuada cuando se compara con el color del fondo, dificultando la lectura del texto.

Estos problemas pueden aparecer en otros medios audiovisuales, pero se acentúan todavía más en el caso de los videojuegos, dado que al tratarse de un medio interactivo, el jugador es quien toma muchas veces el control de la cámara, por lo que no se puede conocer con anterioridad el fondo sobre el que se va a mostrar el texto. El texto puede confundirse fácilmente cuando comparte colores con el fondo, o cuando el fondo contiene patrones o imágenes intrincadas.

Por último, la fuente o tipografía elegida para el texto puede ser subestimada durante el diseño del subtítulo. Algunos juegos seleccionan una tipografía para los subtítulos que acompañan muy bien al diseño artístico del resto del juego, pero que dificulta mucho su legibilidad. Los jugadores con dificultad visual y con dislexia pueden confundir o perderse los detalles en los trazos que identifican a los caracteres en tipografías complejas o sobre-estilizadas.

Es necesario mencionar también que tanto la duración en pantalla como el contenido del texto pueden dificultar la comprensión y legibilidad de los subtítulos. Cuando el texto de los subtítulos tiene una duración muy corta en pantalla se fuerza una aceleración en la lectura y comprensión del texto que algunos jugadores no pueden realizar, por lo que se pierde la información que se está intentando transmitir. Con respecto al contenido de los subtítulos, una oración que está escrita de manera compleja, con faltas de ortografía, fallas en la semántica o la sintaxis, también pueden aumentar innecesariamente la dificultad de la lectura.

C. Solución

No existe una única solución que se pueda aplicar, dado que pueden haber múltiples problemas con los subtítulos. La solución propuesta es designar un espacio específico en el

desarrollo del proyecto para poder pensar y diseñar los subtítulos con mayor consciencia. Todo el esfuerzo que se realiza en la producción y transcripción del contenido para los subtítulos puede ser en vano si los jugadores no pueden ver, leer o comprender el resultado final. Existen varios recursos útiles que sirven como apoyo en el diseño de subtítulos para un juego, listados en la sección Herramientas. Estos recursos parten de prácticas estándar en otras industrias, establecidas por empresas como la BBC, Amazon y Netflix. A modo de resumen se analizarán algunos puntos importantes a tener en cuenta durante el diseño de los subtítulos.

En relación al contenido de los subtítulos, deben ser precisos, es decir deben transmitir, al jugador, la información de una forma correcta, completa y sin errores de ortografía. Las líneas de diálogo se deben corresponder lo más posible con el diálogo en el audio, y con los hechos que están ocurriendo en pantalla. Hay algunos casos en donde no se puede traducir literalmente el audio a texto, o no se puede realizar sin perder información, como por ejemplo en la traducción a otro idioma. En estos casos se puede realizar un ajuste al contenido, teniendo en cuenta la coherencia con el resto del contenido creado. Por ejemplo, se debe decidir si se mantienen los nombres originales de los personajes y lugares mencionados, o si se traducen. La opción elegida se debe mantener para todo el contenido creado para el juego. Otro ejemplo es el uso de los subtítulos cerrados para ciertos efectos de sonido. Si se decide subtítular un efecto de sonido, se debe mantener ese subtítulo cada vez que aparece el efecto de sonido en el juego, y se debe utilizar siempre el mismo texto para representarlo.

Además del contenido se puede analizar la posición del subtítulo en pantalla. La recomendación es que los subtítulos en videojuegos se comporten de la misma manera que el resto del contenido multimedia. Por lo general la audiencia espera que los subtítulos aparezcan en la parte inferior central de la pantalla, por lo que es ahí donde los subtítulos deben aparecer en los videojuegos. El resto de los elementos de la UI, como por ejemplo el HUD²⁴ debe ser diseñado teniendo en cuenta que este espacio está reservado para los subtítulos. Si es necesario mostrar un elemento de UI en dicho espacio, por ejemplo un aviso o sugerencias al jugador, se deben posicionar los subtítulos un poco más arriba para evitar superponer los elementos.

A diferencia de otros tipos de contenidos subtítulados, en los videojuegos es impredecible el momento en que aparecerán algunas líneas de subtítulo en pantalla. Por ejemplo, los subtítulos descriptivos pueden aparecer en cualquier momento dependiendo de cuándo se producen los efectos de sonido en el juego. En estos casos, las líneas de texto nuevas deben desplazar a las anteriores, pero no eliminarlas de la pantalla. La recomendación es

²⁴ Head-Up Display: una componente transparente, generalmente en la parte inferior de la pantalla, en donde se muestra información importante de la partida al jugador, como la salud, ítems u otro tipo de status.

que las nuevas líneas de texto aparezcan en la parte inferior de la pantalla, debajo de las líneas anteriores. Una solución común es hacer que los subtítulos anteriores se muevan hacia arriba, dando lugar a las nuevas líneas, y desapareciendo después de un tiempo determinado.

Con respecto al tiempo que aparecen los subtítulos en pantalla, se debe respetar el margen recomendado, para evitar que desaparezcan antes de que puedan ser leídos por el jugador. Por lo general se estima una velocidad de lectura de los jugadores de 160 a 180 palabras por minuto (BBC Subtitle Guidelines, 2016). La duración recomendada para las líneas con una sola palabra es de un segundo como mínimo. Suponiendo una línea de 38 caracteres como máximo, se recomienda de 2 a 2,5 segundos por cada línea del subtítulo. Para jugadores más jóvenes, la BBC recomienda de 3 a 3,5 segundos por línea para jugadores menores de 13 años, y hasta 4 a 4,4 segundos para jugadores menores de 7 años. Una solución que se puede aplicar cuando se trata de un diálogo entre personajes, es dejar que el jugador maneje el ritmo de los diálogos. Esto se logra por ejemplo dejando que el jugador realice una acción para avanzar a la siguiente línea de diálogo.

Cuando la atención del jugador se divide entre el subtítulo y el gameplay, por ejemplo durante un combate, puede pasar que se necesite más tiempo para poder leer el texto completo. Se recomienda prestar atención a este detalle durante las rondas de playtesting para poder ajustar la duración de los subtítulos. La recomendación es no utilizar líneas más largas que 38 caracteres, y como máximo ajustar el texto a dos líneas.

Con respecto a la tipografía utilizada para el subtítulo, es importante mencionar que se puede elegir un tipo de fuente que acompañe el diseño general del juego, pero siempre se debe tener como prioridad el diseño accesible por sobre el diseño artístico. Si consideramos al subtítulo como una herramienta de accesibilidad, es necesario que sea lo más accesible posible para la mayor cantidad de jugadores. Las comunicaciones vía texto diegético, por ejemplo texto escrito en una carta que aparece dentro del juego, sí pueden utilizar una tipografía que apoye al diseño artístico, pero se recomienda que este tipo de contenido también esté subtítuloado.

Las tipografías Serif se caracterizan por el extremo de los caracteres, llamados serifas, que se extienden un poco más allá del espacio vertical o horizontal de cada letra, lo que les da una apariencia tradicional y elegante. Este tipo de tipografía se utiliza a menudo en los textos con gran cantidad de texto como los libros impresos dado que las serifas crean la ilusión de una línea horizontal, facilitando la lectura. Por el contrario las tipografías Sans-Serif no tienen estos extremos, y se caracterizan por su aspecto más moderno y sencillo. Se usan a menudo para los textos en pantallas, dado que son más legibles a pequeña escala.

En los textos digitales se utiliza con mayor frecuencia la tipografía sin serifa, debido a que se ven mejor en pantallas digitales que las tipografías serif, por lo que son las recomendadas para su utilización en textos dentro de videojuegos. En una pantalla con menor cantidad de píxeles o con menor resolución, las serifas de un texto pueden llegar a verse mal, e incluso a confundir un carácter con otro. Algunas opciones de tipografías Sans-Serif son: Arial, Helvética y Verdana.

Los jugadores disléxicos pueden tener problemas para leer ciertos tipos de fuentes, sobre todo cuando diferentes caracteres tienen una representación visual muy parecida, por ejemplo los caracteres “D” y “P” minúscula suelen verse muy parecidos, o “L” e “I” mayúscula. Este parecido entre los caracteres puede dificultar aún más la lectura de texto a los jugadores con dislexia. En la figura 4.12 podemos ver una muestra de diferentes tipografías, comparando los caracteres que más comúnmente causan dificultad en la lectura. La fuente “OpenDyslexic”, creada específicamente para facilitar la lectura a los usuarios con dislexia, es mucho más sencilla de leer que otras fuentes por sus caracteres únicos y con “peso” agregado a los trazos para indicar la dirección de las letras.

Gill Sans	rn m	MW	dpqb	lIlij
Verdana	rn m	MW	dpqb	l1IijJ
OpenDyslexic	rn m	MW	dpqb	l1IijJ
Times	rn m	MW	dpqb	l1IijJ
Helvetica	rn m	MW	dpqb	l1IijJ

Figura 4.12 Comparación de OpenDyslexic con otros tipos de tipografías. Tomado de opendyslexic.org

Por último, el tamaño del subtítulo es también algo importante para tener en cuenta. La guía de accesibilidad de XBox propone un tamaño mínimo para los textos de acuerdo a la plataforma. Considerando una resolución de 1080p, se espera que el texto tenga un tamaño igual o superior a 26px en consolas y 18px en PC. En dispositivos móviles se debe tener en cuenta los DPI²⁵ del dispositivo, y se recomienda un tamaño de 18px a 100 DPI, 36px a 200 DPI y 72px a 400 DPI. La guía alienta a los desarrolladores a utilizar, siempre que sea posible, tamaños aún más grandes que los mínimos propuestos.

²⁵ Del inglés “Dots per inch” puntos por pulgadas.

D. Beneficios

Apoyarse en lineamientos y guías que son estándares en otras industrias ayuda a mejorar de una forma efectiva la accesibilidad en los videojuegos. En particular, los subtítulos son una de las herramientas más documentadas y probadas por organizaciones que trabajaron para mejorarlos. Los desarrolladores de videojuegos pueden verse beneficiados por aprender a utilizar correctamente recursos de este estilo.

Designar un tiempo adicional para diseñar correctamente los subtítulos asegura que esta solución de accesibilidad no solo permita a los jugadores con dificultades auditivas poder jugar con mayor facilidad, sino también asegura que no será una barrera para el resto de los jugadores.

E. Clasificación

- A. Dificultad sensorial relacionada: Auditiva, Visual
- B. Dificultad en la interacción: Al recibir el estímulo
- C. Principio de diseño universal: (1) Igualdad de uso, (4) Información comprensible.
- D. Dificultad de implementación: Media

F. Ejemplos

El videojuego “Champions of Chaos 2” (2015) del estudio argentino Epic Llama incluye diálogos entre personajes y subtítulos para cinemáticas. En la figura 4.13 se puede observar una captura de una cinemática del juego en donde se utilizan subtítulos. El tamaño del subtítulo es correcto, y es posible verlo fácilmente en pantalla. También incluye un borde y sombra, logrando un buen contraste contra el fondo. La tipografía sin embargo no es la adecuada para subtítulos dado que es del tipo Serif, la cual puede verse mal en pantallas. También podemos ver que no se respeta la recomendación de 2 líneas de subtítulo como máximo, haciendo más difícil de comprender el contenido de esta captura de pantalla.

Si podemos agregar algo más sobre los subtítulos en esta escena, es que parecen estar flotando dentro y fuera del juego, es decir que no tienen un espacio diseñado específicamente para ellos.



Figura 4.13 Cinemática con subtítulos en “Champions of Chaos 2”

El videojuego “Dorfs: Hammers for Hire” (2022), del estudio cordobés Ravegan incluye subtítulos en su cinemática introductoria al juego. Como se puede ver en la figura 4.14, el subtítulo tiene un tamaño y color adecuado por lo que se puede leer fácilmente. Este es un ejemplo de cómo se puede diseñar un subtítulo para que forme parte del diseño artístico del juego, sin dejar de ser accesible a los jugadores. La tipografía es sans serif, el contraste con el fondo es correcto, y los subtítulos tienen la duración y cantidad de texto adecuada.

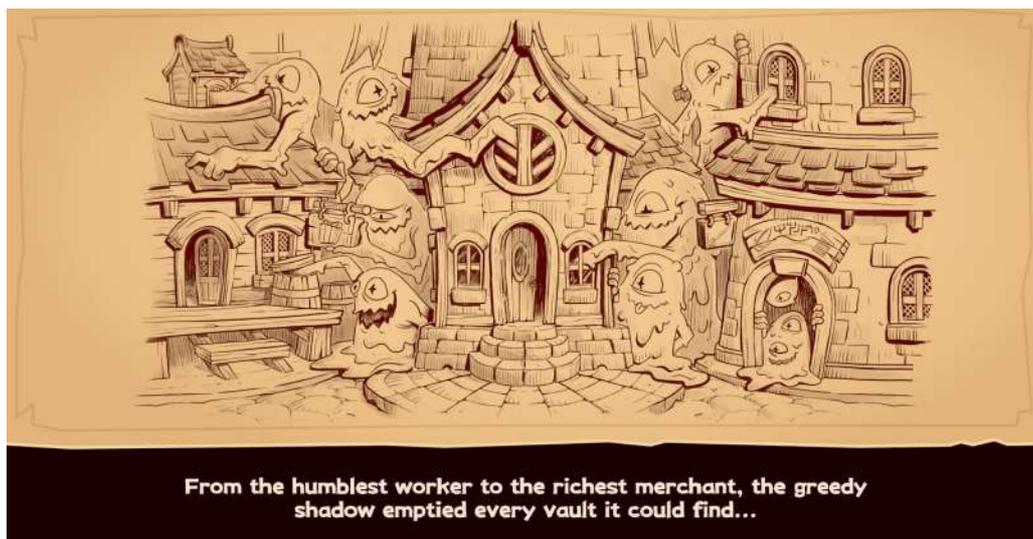


Figura 4.14 Subtítulos en el juego “DORFS Hammers for Hire” (2022)

“Blue Fire” (2021) del equipo cordobés Robi Studios es un juego de plataformas 3D de acción y aventura enfocado en la exploración, con múltiples personajes y diálogos durante

todo el juego. En la figura 4.15 se puede observar un ejemplo del subtítulo diseñado para este juego. Se está utilizando una tipografía Serif, por lo que en pantallas pequeñas el texto puede ser mucho más difícil de leer. El subtítulo se encuentra dentro de un recuadro semitransparente que se utiliza para asegurar el contraste entre los subtítulos y el fondo, algo que funciona muy bien en todo el juego. En la misma figura también podemos observar un ejemplo de cómo se puede diseñar el espacio para que los subtítulos puedan interactuar con otros elementos de la UI.



Figura 4.15 Ejemplo de subtítulos en “Blue Fire” (2021) interactuando con otros elementos de UI.

Una de las recomendaciones más fuertes es mostrar los subtítulos siempre en la sección baja central de la pantalla. En la figura 4.16 se puede observar una captura del juego “The Night of the Scissors” (2022) del desarrollador argentino Tomás Esconjaureguy. En este juego, todos los diálogos e instrucciones del juego aparecen alineados a la izquierda de la pantalla, por lo que no respetan la ubicación recomendada para los subtítulos. Además de esto, otro problema que encontramos en “The Night of the Scissors” es que tanto los textos narrativos, como los diálogos entre personajes, como las instrucciones del juego, se ven todas de una forma muy similar, dificultando la comprensión por parte del jugador. Todos los textos se ven con una fuente, color y posición muy similares, por lo que no resulta sencillo diferenciar entre este tipo de contenido. En la figura 4.16 se puede observar un ejemplo de un subtítulo del juego con el texto alineado a la izquierda y el nombre del personaje antes de la línea de diálogo, algo muy útil para poder identificar cuál de los dos personajes está hablando.



Figura 4.16 Los subtítulos en “The Night of the Scissors” (2022) no respetan la ubicación recomendada en pantalla.

En la figura 4.17, se puede observar un ejemplo de otro diálogo dentro del mismo juego, pero esta vez no se ve el nombre del personaje antes de la línea de diálogo. En esta escena, además, se debe esperar una acción del jugador (presionar “continuar”) para poder terminar el diálogo. En ambos casos se usa una estrategia diferente a la elegida para el diálogo inicial entre los dos personajes, por lo que se pierde la consistencia de los subtítulos dentro del mismo juego.



Figura 4.17 Otro subtítulo de diálogo en “The Night of the Scissors” (2022), inconsistente con respecto al diálogo en la Figura 4.16.

Por último se puede observar cómo se muestran instrucciones al jugador en “The Night of the Scissors” de una forma muy similar a la que se muestran los diálogos en el resto del juego. No construir un lenguaje visual, o construir un lenguaje visual pobre, puede generar

confusión y dificultar innecesariamente la comprensión de información que debería ser muy sencilla de procesar por el jugador. Como ejemplo en la figura 4.18, se puede observar una instrucción del juego hacia el jugador, visualmente muy similar a un subtítulo dentro del juego. En la figura 4.19, se puede observar un texto narrativo que ofrece información circunstancial sobre el entorno de juego, algo que también podemos llegar a confundir con una instrucción o con un diálogo de personaje.



Figura 4.18 En "The Night of the Scissors" las instrucciones son visualmente similares a los subtítulos de diálogos entre personajes.



Figura 4.19 En "The Night of the Scissors" los textos narrativos son visualmente similares a los subtítulos de diálogos entre personajes.

"The Night of the Scissors" es un ejemplo concreto de cómo un juego independiente puede mejorar sustancialmente al aplicar reglas que no tienen que ver con soluciones

técnicamente complejas, sino con conceptos teóricos y estándares sencillos de implementar, y de la importancia de la etapa del diseño de los subtítulos en el desarrollo de un videojuego.

G. Herramientas

- **Conferencia GDC Ian Hamilton (2019): “Subtitles are changing, don’t be left behind”** Experto en accesibilidad en la industria, explica cuales son los problemas más comunes con respecto al diseño de los subtítulos
<https://www.youtube.com/watch?v=fVFgSkZtCio>
- **Guía de la BBC** para equiparar y armonizar el subtítulo al diálogo original en traducciones
<https://www.bbc.co.uk/accessibility/forproducts/guides/subtitles/#Editing>
- **Principios Generales** de Amazon para el diseño de la UI
<https://developer.amazon.com/docs/fire-tv/design-and-user-experience-guidelines.html#general-principles>

Personalización de los subtítulos

A. Resumen

Es útil permitir a los jugadores cambiar la forma en la que se ven los subtítulos en pantalla. No solo los jugadores con dificultades auditivas utilizan los subtítulos, los jugadores con dificultad visual suelen necesitar una ayuda extra para poder leer los subtítulos de forma cómoda.

B. Problema

Siempre que se diseñan las opciones de accesibilidad en un videojuego, no solo se piensa en qué tipos de barreras se tratan de solucionar, sino también se consideran qué tipos de barreras nuevas se están creando. Cuando se implementan soluciones de accesibilidad con subtítulos, se están solucionando las barreras derivadas de las dificultades auditivas para ciertos jugadores, pero se pueden generar nuevas barreras para jugadores con dificultades visuales y cognitivas. Puede que estas barreras se produzcan incluso para jugadores sin ningún tipo de discapacidad.

Así como se consideran las discapacidades visuales durante el diseño de cualquier elemento que aparecerá en pantalla (íconos, personajes, interfaces de usuario, etc), se deben considerar también durante el diseño de los subtítulos. Existen múltiples barreras que se pueden crear involuntariamente con la introducción de los subtítulos en un videojuego, analizadas en detalle en los puntos anteriores de este catálogo. Incluso con un excelente trabajo de diseño, puede que las características finales elegidas para el subtítulo no se ajusten a todos los jugadores, por lo que es necesario ofrecer una manera de configurar y ajustar estas características de acuerdo a cada jugador.

Los jugadores con buena audición y visión, pero con algún tipo de discapacidad cognitiva, por ejemplo, pueden experimentar dificultades para leer los subtítulos. Más allá del tamaño

de la tipografía, pueden tener problemas para comprender los subtítulos cuando la cantidad de texto en pantalla es demasiada, o cuando está escrito de una manera difícil de entender. Otros jugadores que podrían presentar dificultad para leer son aquellos que todavía no tienen la suficiente práctica en la lectura, por ejemplo jugadores que están aprendiendo a leer. Incluso puede suceder que el tamaño del texto sea el correcto, pero que la tipografía o fuente utilizada para el subtítulo sea difícil de interpretar, por ejemplo, para los jugadores con dislexia. Algunos jugadores pueden experimentar problemas para entender los subtítulos cuando están jugando muy alejados de la pantalla, o cuando la pantalla es muy pequeña, como es el caso de los juegos en dispositivos móviles. La edad también puede ser un factor influyente. Como se mencionó, los jugadores que están aprendiendo a leer, como por ejemplo los jugadores menores de edad, pueden presentar dificultades para leer un subtítulo cuando avanza muy rápido, o cuando incluye textos muy largos o con palabras complejas. Por otra parte, los jugadores de edad avanzada también suelen tener problemas para ver el texto de los subtítulos cuando es muy pequeño, o cuando tiene bajo contraste en relación al fondo.

Por último algunos jugadores simplemente pueden elegir no utilizar los subtítulos. Los subtítulos son una herramienta que puede no ajustarse a las necesidades de todos los jugadores. Por ejemplo, algunos jugadores eligen utilizar los subtítulos sólo en las cinemáticas, por considerarlos elementos que distraen durante el gameplay. Los subtítulos descriptivos también pueden ser considerados elementos que distraen del gameplay, por lo que algunos jugadores pueden elegir no utilizarlos.

C. Solución

Cada jugador es único, con necesidades únicas, estilos de juego y circunstancias diferentes al momento de jugar. Es por esto que los subtítulos en los juegos, al igual que otras herramientas de accesibilidad, deben ser opcionales y flexibles. Deben permitir cierto grado de personalización para que cada jugador pueda ajustar solo lo que necesita para cubrir estas necesidades únicas.

Las opciones de subtítulos es una de las configuraciones que más varían su ubicación dentro del menú de configuración del juego. En algunos juegos se encuentra dentro de las opciones de pantalla, en otras en opciones de juego, en opciones de audio, o hasta en un menú propio de accesibilidad. Las opciones de subtítulos deben estar bajo la categoría "Audio", dado que es una representación tanto de los diálogos como de los efectos de sonido que forman parte del audio del videojuego.

Una buena práctica es permitir al jugador elegir qué tipo de subtítulos quiere utilizar. Algunas de las opciones que generalmente se utilizan son:

- Apagar subtítulos: los subtítulos están completamente desactivados.

- Subtítulos para diálogos: los subtítulos se activan solo para el diálogo hablado de los personajes.
- Subtítulos descriptivos: adicionalmente a los subtítulos en diálogos, se utiliza closed-caption para subtítular todos los efectos de sonido.

Algunos juegos permiten activar los subtítulos en diferentes categorías, como por ejemplo subtítulos de NPCs, subtítulos del personaje principal, subtítulos de tutoriales, etc. Este tipo de categorización tiene sentido dependiendo de cada juego.

Además de permitir apagar o prender los subtítulos, la personalización hace referencia a la configuración de las características del texto que aparece en pantalla. La categoría más importante a considerar es el tamaño del subtítulo. Aunque es una de las características más sencillas y obvias, suele suceder que se subestime y se abandone hasta etapas tardías del desarrollo. Desde el inicio del diseño de la interfaz del juego se debe tener en cuenta esta característica dado que, dependiendo del estilo de la UI y el lugar de la pantalla donde aparecen los subtítulos, se puede generar algún problema de superposición de elementos cuando los subtítulos son muy grandes. Por ejemplo, si el jugador selecciona un tamaño muy grande de subtítulos y el diseño de la UI no lo tuvo en cuenta o no es lo suficientemente flexible, puede hacer que el subtítulo no se vea correctamente y se pierda información. Por este motivo, en algunos juegos solo se permite seleccionar el tamaño entre una serie de valores predeterminados. La recomendación es soportar un escalado del texto de un 50% hasta un 200% del tamaño original. Los motores de videojuegos actuales tienen soluciones para construir interfaces flexibles que se ajustan a diferentes tamaños de pantalla. En este caso es recomendable presentar al jugador un slider como mecanismo para ajustar el tamaño del texto. En caso de que el juego no utilice un motor con soporte para interfaces flexibles, es recomendable la configuración utilizando valores predeterminados para el tamaño de los subtítulos.

En segundo lugar, pero aún así con igual importancia, encontramos la configuración de los colores. Los jugadores con dificultades para identificar los colores pueden tener problemas para visualizar correctamente los subtítulos, por lo que es necesario proveer una manera para cambiar el color elegido en el diseño de los subtítulos. Algunos juegos permiten seleccionar entre una lista de colores predeterminados, pero la mejor solución es dejar el control completo al jugador, y utilizar algún tipo de selector de colores. En caso que los subtítulos implementen la identificación del personaje hablante, el juego debe permitir también personalizar esta característica. Por ejemplo, si se utilizan colores para identificar cada personaje, se puede permitir al jugador cambiar los colores preseleccionados de cada personaje. Dependiendo de la cantidad de personajes que existan en el juego, se puede permitir al jugador seleccionar un color para cada personaje, o seleccionar entre una serie de paletas con colores predeterminados.

Lo mismo sucede con los colores del fondo de los subtítulos. Por lo general el fondo suele ser transparente, dado que el contraste generado por los bordes de los caracteres es suficiente para separar el texto del fondo. Para algunos jugadores puede no ser suficiente este contraste, por lo que se debe brindar una opción para cambiar el color de fondo de los subtítulos. Por lo general se suele dividir esta característica en dos configuraciones, una que controla el color, y otra que controla la opacidad del elemento.

Aunque no suele ser una opción muy utilizada, algunos juegos permiten cambiar la tipografía utilizada en los subtítulos. Por lo general, durante la etapa de diseño se selecciona la tipografía que más se ajuste a las necesidades del juego y de los jugadores. Aún así, dependiendo de las decisiones de diseño que se tomaron en la construcción del juego, se puede brindar una opción para cambiar la tipografía por defecto de los subtítulos por una que permita ser leída de una forma más sencilla.

De existir otras características implementadas en los subtítulos, se debe decidir si es necesario proveer una manera de desactivarlas, o de personalizarlas. Por ejemplo, si se muestra el nombre del personaje hablante al inicio de la línea de diálogo, se puede ofrecer una opción para cambiar esto y ocultar el nombre de los personajes. Otro ejemplo es la velocidad con la que avanzan los subtítulos en los diálogos entre personajes, si bien se puede brindar una opción para controlar la velocidad o el tiempo que duran los subtítulos en pantalla, siempre es mejor dejar esto bajo el control del jugador, por ejemplo al requerir la pulsación de un botón para avanzar a la siguiente línea de diálogo.

Una opción que algunos juegos suelen pasar por alto, es la configuración del idioma de los subtítulos. La localización de un videojuego es un tema en sí mismo, con sus propias dificultades y decisiones de diseño. En relación con el contenido de los subtítulos, podemos incluir opciones para configurar el idioma. De ser posible, el idioma de los subtítulos debería ser configurable independientemente del idioma del resto del juego. Debería ser posible, por ejemplo, jugar escuchando el diálogo en el idioma original, mientras que los subtítulos están en otro idioma diferente. Esto parece algo muy lógico, pero aún así algunos juegos no permiten hacerlo.

Por último, una opción de configuración de subtítulos que no es requerida pero si es recomendable, es mostrar una vista previa de los cambios en los subtítulos en la pantalla de configuración, facilitando la personalización de los subtítulos por parte de los jugadores.

D. Beneficios

Los subtítulos son herramientas que permiten solucionar barreras en la interacción con los videojuegos. Por esta razón, es importante que sean lo suficientemente flexibles como para que se ajusten a todos los tipos de jugadores. La personalización de subtítulos beneficia no solo a los jugadores con discapacidad auditiva, sino también a los jugadores que tienen dificultades visuales como baja visión o daltonismo. Los jugadores que poseen dificultades

para leer, dislexia o dificultades cognitivas también se ven beneficiados al poder ajustar los subtítulos de acuerdo a sus necesidades personales. En los juegos que soportan múltiples doblajes de audio también permiten al jugador por ejemplo leer los diálogos de los personajes en un idioma accesible, mientras que escuchan el doblaje original del juego.

Algunos juegos brindan una personalización de los subtítulos poco granular, por ejemplo brindando plantillas o presets con un conjunto predeterminado de ajustes. Esto puede no ser útil para algunos jugadores, dado que no permite ajustar la característica en particular que el jugador necesita.

E. Clasificación

- A. Dificultad sensorial relacionada: Auditiva, Visual, Cognitiva
- B. Dificultad en la interacción: Al recibir el estímulo,
- C. Principio de diseño universal: (1) Igualdad de uso, (2) Flexibilidad, (4) Información comprensible.
- D. Dificultad de implementación: Media

F. Ejemplos

El juego “Grim Fandango” (2015) permite cambiar dentro de su menú de configuración algunas opciones con respecto a los subtítulos. Por ejemplo permite ajustar mediante un slider la velocidad del texto en los diálogos (figura 4.20).

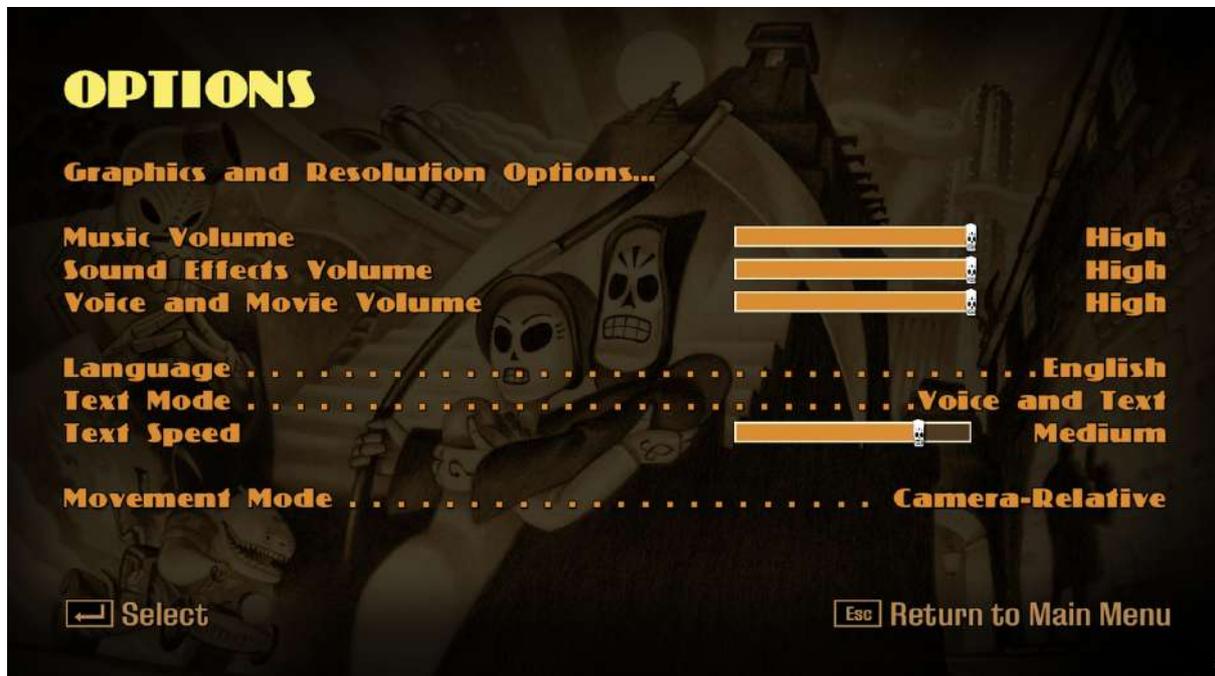


Figura 4.20 Pantalla de configuración de subtítulos en “Grim Fandango Remastered” (2015)

Aunque se permite cambiar el idioma de los subtítulos, no se permite ajustar la configuración para por ejemplo escuchar el doblaje original con las voces en inglés, y leer

los subtítulos en otro idioma. Para poder hacer esto, los jugadores tienen que editar manualmente los archivos del juego dado que no se permite desde las opciones de configuración (Grim Fandango Remastered Steam Formus, 2015).

Durante el diseño de “ScourgeBringer” se decidió utilizar una tipografía que encajaba con el diseño del resto del juego, pero que era poco legible (figura 4.21). Más adelante en el desarrollo se agregó una opción de configuración que permite utilizar una tipografía más sencilla de leer.



Figura 4.21 Pantalla de configuración de “ScourgeBringer” con tipografía poco legible.

El juego “The Last of Us: Part 2” incluye opciones de personalización para los subtítulos (figura 4.22). Por ejemplo permite configurar los subtítulos de los combates y de la historia en forma separada. Se pueden ajustar el tamaño y el color del fondo del texto. Además de ser necesario se pueden activar o desactivar los nombres de los personajes en el diálogo, así como también el color de los mismos. Este juego también incluye una vista previa de edición del subtítulo en la parte inferior de la pantalla de configuración.

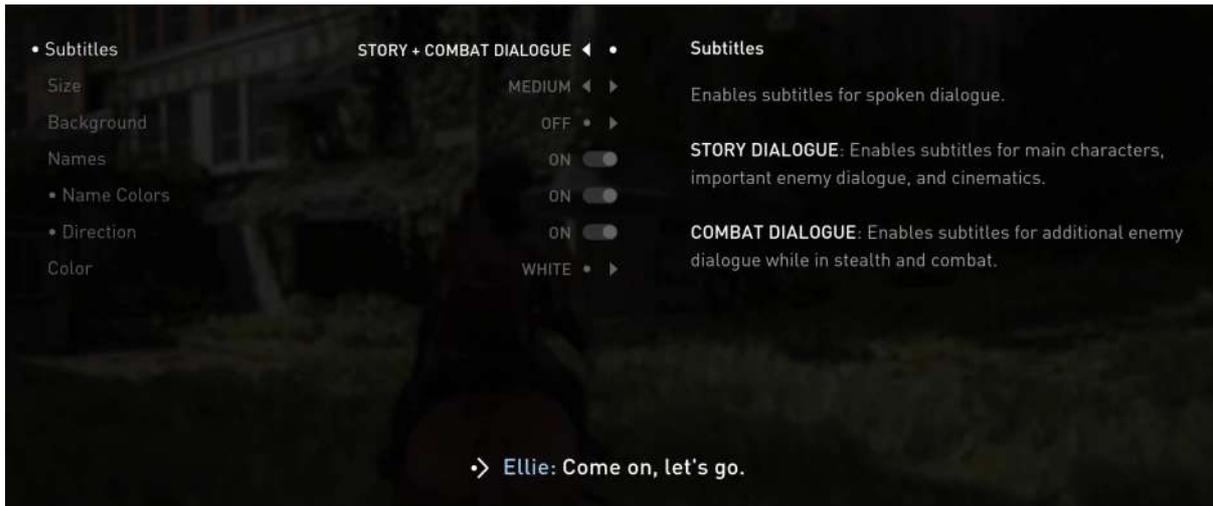


Figura 4.22 “The Last Of Us: Part 2” incluye muchas opciones de personalización para los subtítulos

G. Herramientas

- **XBOX Accesibility Guidelines:** implementación del diálogo y subtítulo
<https://learn.microsoft.com/en-us/gaming/accessibility/xbox-accessibility-guidelines/104#implementation-guidelines>
- **Where to Begin: Games Accessibility, Morgan Baker:**
https://www.youtube.com/watch?v=NDD_vSfkrVI

Proveer controles de volumen exclusivos para cada fuente de sonido

A. Resumen

Brindar al jugador la posibilidad de controlar de forma separada el volumen de las diferentes fuentes de sonido del juego: música, diálogos y efectos de sonido.

B. Problema

En los videojuegos, como cualquier otro medio audiovisual, la mayor parte de la información es visual. Esto no quita la importancia que tiene la información transmitida mediante el sonido. El diseño de sonido en videojuegos es una disciplina en sí misma, y se trata de una rama del diseño de sonido que se centra en crear y agregar elementos auditivos a un videojuego. Involucra la creación de librerías de audio, grabación de efectos de sonido y diálogo, introducción de música, y masterización de las diferentes pistas de audio.

En un videojuego, la información auditiva se apoya en la información visual, y a su vez también la acompaña y consolida los mensajes que trata de transmitir al jugador, y viceversa. Cuando se genera alguna barrera entre el jugador y el juego que afecta una de estas partes, la sinergia generada entre las partes se corta y el jugador puede perder información esencial. Por lo general cuando se toca este tema, se analizan las barreras que se generan con la pérdida de la información visual, y pocas veces se examinan los problemas que afectan a la parte auditiva.

Los jugadores que más se ven afectados por problemas en el diseño de sonido en videojuegos son obviamente los jugadores con dificultades auditivas. En estos casos, la pérdida total de la audición crea una barrera completa ante la información auditiva que intenta transmitir el juego, por lo que cualquier información exclusivamente transmitida mediante el sonido se perderá. Tomemos como ejemplo un juego de lucha, en donde los jugadores pueden efectuar golpes, y estos golpes pueden conectar con el oponente o fallar. Si la única manera de saber cuándo se conecta o falla un golpe es mediante el sonido, un jugador completamente sordo no contará con la información necesaria para poder jugar correctamente. Lo mismo sucede cuando los jugadores con pérdida parcial de la audición, como por ejemplo los jugadores con hipoacusia unilateral o bilateral.

Los jugadores con dificultades visuales también pueden verse afectados por problemas de diseño de sonido. Si el jugador no puede detectar de forma sencilla las pistas visuales, dependerá de la información auditiva para poder jugar correctamente. Algunos ejemplos de señales o pistas visuales en los videojuegos incluyen representaciones gráficas de la salud o del daño recibido, indicadores relacionados a ítems o munición de armas, efectos visuales que muestran cuando un personaje ha sido golpeado por un ataque, etc.

Las discapacidades cognitivas también pueden alterar la interacción con el medio auditivo en el videojuego. Cuando existen múltiples fuentes sonoras que suenan al mismo tiempo, compiten por la atención del jugador y pueden ser más difíciles de procesar para algunos jugadores. El trastorno del procesamiento auditivo, por ejemplo, es una condición que suele darse con más frecuencia en niños, en donde no existe necesariamente un problema para escuchar los sonidos, sino para procesarlos y entenderlos. Los ambientes ruidosos o con distracciones suelen ser un problema que dificulta la identificación de los diferentes sonidos, algo que puede obstaculizar incluso la capacidad de aprender del niño, o hasta de interactuar con otras personas. En Estados Unidos se estima que este trastorno puede aparecer en el 2% a 3% de los niños, mientras que en adultos mayores a 60 años este porcentaje asciende hasta 10% o 20%. También se estima que un 50% de las personas con dislexia también padecen de este trastorno auditivo (Díaz et al, 2022).

C. Solución

Los problemas descritos anteriormente pueden disminuir sus efectos si se permite a los jugadores modificar el audio del juego, de manera que los sonidos que causan distracción o ruido se puedan apagar o reducir en intensidad, y los sonidos que interesan al jugador se puedan resaltar del resto.

Esto se logra ofreciendo opciones de configuración para la intensidad de los diferentes tipos de sonidos dentro del juego. Las configuraciones más utilizadas incluyen: volumen de la música, volumen de los diálogos, y volumen de los efectos de sonido. En otros juegos se

permite configurar el volumen de los sonidos de la UI, volumen de chat de voz, o volumen del sonido ambiental.

Este tipo de configuraciones permiten a los usuarios personalizar la mezcla de sonidos, sobrescribiendo el diseño de sonido original. Esto es importante porque los jugadores pueden customizar el sonido del juego de acuerdo a sus necesidades personales. Por ejemplo, los jugadores que necesitan ayuda para escuchar los diálogos entre personajes pueden bajar el volumen de la música y subir el volumen de los diálogos. Los jugadores con dificultades visuales pueden mejorar la navegación de los menús y la interacción con los elementos de las interfaces si pueden escuchar más fácilmente los sonidos característicos de la UI.

D. Beneficios

Brindar flexibilidad al jugador siempre es una buena opción, porque permite preparar al juego para situaciones para las que no fue diseñado o para circunstancias no imaginadas previamente. La implementación de esta opción de accesibilidad no es complicada, y es mejor que brindar un único control para todo el audio del juego. Imaginemos por ejemplo un jugador que está jugando en un dispositivo móvil mientras intenta escuchar música. Si el juego no tuviera ninguna opción de configuración para disminuir el volumen de la música, es probable que encuentre incómodo escuchar música en una aplicación externa al mismo tiempo que juega. Si bien se puede apagar el volumen de todo el juego, se perderían los diálogos y efectos de sonido que tal vez sean importantes para poder jugar correctamente.

En la actualidad es común compartir contenido a través de Internet. En redes sociales, plataformas web, aplicaciones de mensajería instantánea, se suelen compartir todo tipo de contenido incluyendo el relacionado a videojuegos. Una de las formas de marketing de videojuegos más extendidas en la actualidad se realiza a través del streaming de gameplays por jugadores a través de sitios como Twitch o YouTube (Caby et al, 2023). Por la naturaleza de este tipo de contenidos, se suele configurar el audio del juego, por ejemplo apagando la música para agregar comentarios en vivo durante un stream, o para reemplazar por otra música en un video de YouTube. Para los desarrollos independientes es importante ser flexible en este aspecto, y así favorecer el marketing y a que el juego sea compartido más fácilmente. Incluso puede beneficiar al desarrollador al momento de crear un trailer para el juego dado que es posible, al momento de grabar contenido para el trailer, silenciar la música y otros sonidos que no son necesarios, y grabar un gameplay limpio y fácil de editar.

E. Clasificación

- A. Dificultad sensorial relacionada: Auditiva, Visual, Cognitiva
- B. Dificultad en la interacción: Al recibir el estímulo, Al procesar el estímulo
- C. Principio de diseño universal: (2) Flexibilidad, (4) Información comprensible

D. Dificultad de implementación: Baja

F. Ejemplos

El juego “Cult of the Lamb” (2022) de “Massive Monster”, incluye las opciones de configuración usuales de volumen general, volumen de música y volumen de efectos de sonido (figura 4.23). Este juego incluye escenas narradas, por lo cual se incluye una opción para poder controlar exclusivamente el volumen de la narración.



Figura 4.23 Pantalla de configuración de sonido del juego “Cult of the Lamb” (2022).

En la figura 4.24 se puede observar una captura del menú de opciones del juego “Guacamelee!” (2013), el cual incluye entre sus opciones de configuración dos opciones para controlar por separado el volumen de la música y el resto del sonido del juego.



Figura 4.24 Captura de pantalla del menú de opciones del juego “Guacamelee!” (2013)

G. Herramientas

- **Basics of Sound Design, Michael Cullen**
<https://frost.ics.uci.edu/ics62/BasicsofSoundDesignforVideoGames-MichaelCullen.pdf>
- **XBox Accessibility Guide:** accesibilidad en el audio
<https://learn.microsoft.com/en-us/gaming/accessibility/xbox-accessibility-guidelines/105>

Permitir ajustar el tipo de canal de salida (mono/stereo)

A. Resumen

Permitir al jugador configurar el modo de salida de los canales de audio del juego. Es decir utilizar un solo canal de salida (mono), dos canales (stereo), o incluso más canales de salida como 5.1 o 7.1

B. Problema

Los efectos de sonido, la música y el audio de un videojuego en general son esenciales para lograr una buena inmersión del jugador en el mundo propuesto por el videojuego (Brereton et al, 2015). Sin embargo, al igual que otros aspectos del diseño de un videojuego, no está exento de provocar barreras de accesibilidad. Además de los problemas mencionados en el punto anterior, se pueden identificar otro tipo de barreras relacionadas a la cantidad de canales de audio del juego. Una de estas barreras se presenta cuando un dato crítico o información importante para el jugador se transmite por uno solo de estos canales de audio. En la mayoría de los casos este es el comportamiento

deseado, por ejemplo en el caso del audio posicional, en donde dependiendo donde esté el origen del estímulo en el mundo virtual, escucharemos el sonido en un canal o en otro.

Para identificar el problema imaginemos un juego en donde hay una batalla entre dos equipos rivales. Para que el juego sea más inmersivo, se decidió que el sonido de las explosiones durante la batalla utilice audio posicional. Por esto, el sonido de las explosiones a la izquierda de la pantalla se escuchará por el altavoz izquierdo, y las explosiones a la derecha por el altavoz derecho. Un jugador que utiliza auriculares para jugar, se sentirá naturalmente más integrado y sentirá que forma parte de la escena. La figura 4.25 ejemplifica la situación con un diagrama en donde se ven dos explosiones que llegan a los oídos del jugador por diferentes posiciones.



Figura 4.25 Diagrama de jugador y juego con audio posicional.

Sin embargo este tipo de diseño puede generar una barrera para los jugadores con disminución total o parcial de su capacidad auditiva. Siguiendo el ejemplo del juego con audio posicional, si el jugador posee hipoacusia unilateral, es decir que solo puede escuchar por uno de sus oídos, entonces se perderá por completo la información auditiva que llega por uno de sus altavoces. En la figura 4.26 podemos ver un diagrama que ejemplifica la situación. Esta pérdida de información puede llegar a confundir al jugador, o impedir jugar con normalidad, o incluso hacer imposible poder avanzar en el juego. Es necesario remarcar la importancia que puede tener este tipo de tecnología auditiva en videojuegos al momento de mejorar la localización y percepción de objetos virtuales en el espacio virtual por parte del jugador (Bonada et al, 2011), pero sin olvidar a los jugadores con discapacidad auditiva, ya que por lo general este tipo de jugadores utiliza con mayor frecuencia auriculares u otro tipo de dispositivos para poder asistir su audición, con lo cual este problema se acrecienta todavía más.



Figura 4.26 Diagrama de jugador con hipoacusia unilateral y juego con audio posicional.

C. Solución

Si bien la mayoría de los sistemas operativos incluyen entre sus opciones de accesibilidad la posibilidad de configurar la cantidad de canales para la salida de audio, es mejor ofrecer esta posibilidad desde la configuración del juego. Por ejemplo, el sistema operativo Ubuntu 20.04 no ofrece a los usuarios una manera sencilla de poder cambiar esta configuración (Ubuntu Desktop Guide, Accessibility, 2022).

Los motores para la creación de videojuegos modernos como “Unity”, “Unreal Engine”, “GameMaker” y “Godot Engine” incluyen soporte por defecto para audio posicional en 2D y 3D. Si bien este nivel de abstracción en las herramientas de creación de videojuegos ayuda a los desarrolladores a crear experiencias más inmersivas, también puede generar efectos indeseados bajo algunas situaciones o condiciones no previstas, como la explicada en el ejemplo anterior.

Es necesario estudiar y utilizar correctamente la API de audio de estos motores, y exponer al jugador una serie de configuraciones de sonido, de manera que se permita adaptar el juego a situaciones no previstas. En general los motores de videojuegos permiten cambiar la cantidad de canales de salida, incluyendo la posibilidad de convertir el audio a mono. La solución es exponer en las opciones de configuración de sonido, una opción para interactuar con el motor de audio del juego, y permitir cambiar la cantidad de canales de salida de audio.

D. Beneficios

De la misma manera que ocurre con los últimos dos puntos del catálogo, la personalización de los canales de salida de audio es una manera de adaptar el juego a situaciones imprevistas, para las cuales no fue diseñado inicialmente. La posibilidad de forzar la salida de audio a un solo canal permite a los jugadores con hipoacusia una mejor interacción con el videojuego. Incluso puede ser útil y proveer una mejor claridad de los sonidos cuando la

calidad del dispositivo de audio de salida es pobre, por ejemplo por tener una mala calidad del rango dinámico del audio²⁶, o por poseer un solo parlante de salida.

El audio espacial es fundamental para los jugadores con pérdida total de la visión, ya que les permite conocer la posición de elementos virtuales que no pueden percibir de forma visual. Sin embargo, esto no quita que exista otra necesidad para otro tipo de jugadores, por ejemplo los que tienen hipoacusia unilateral, por lo que el juego debe ser flexible y adaptarse a todo tipo de jugadores.

Aunque algunos sistemas operativos de escritorio y móviles incluyen este tipo de opción de accesibilidad, es mucho mejor ofrecer una propia dentro del juego. De esta forma no es necesario cambiar la configuración de todo el dispositivo, lo que puede ser tedioso si es un dispositivo que se comparte entre varios jugadores.

E. Clasificación

- E. Dificultad sensorial relacionada: Auditiva
- F. Dificultad en la interacción: Al recibir el estímulo
- G. Principio de diseño universal: (1) Igualdad de uso, (2) Flexibilidad, (4) Información comprensible, (6) Bajo esfuerzo
- H. Dificultad de implementación: Fácil

F. Ejemplos

La consola de juegos “Playstation 4” fue lanzada en el año 2013 y siendo todavía una de las consolas más utilizadas en todo el mundo (Chen H., (2022). *The Impact of the Covid-19 on the Video Game Market*, p. 73). Aun así no cuenta con opciones básicas de accesibilidad, como por ejemplo una opción para poder utilizar una salida monoaural de audio. Solo permite cambiar el formato del audio por PCM lineal, Dolby y DTS. Estas tres alternativas ofrecen audio espacial y la utilización de múltiples canales de salida, pero ninguno permite la utilización de un solo canal para la salida del audio. Esto obligó a los jugadores a implementar soluciones como rotar la cámara dentro del juego para poder escuchar el audio correctamente, dejar de utilizar auriculares y posicionar estratégicamente los altavoces, o incluso hasta utilizar una solución de hardware que mezcla los canales de audio en uno solo.

Esto cambió con la salida de la consola PlayStation 5, la cual incluye entre las opciones de accesibilidad, la posibilidad de utilizar una salida monoaural. Esto benefició a muchos jugadores, como se puede ver en el foro de PlayStation 5 del sitio Reddit. En la Figura 4.27 podemos leer un comentario del usuario “yuridrums”: “Estoy sordo de un oído y finalmente no tendré que rotar mi personaje o la cámara al lado izquierdo solo para poder escuchar una conversación”, extraído del sitio web Reddit:

²⁶ Se llama así a la diferencia en decibeles entre el sonido más débil y el más fuerte.

(www.reddit.com/r/PS5/comments/wfx3da/ps5_at_some_point_added_mono_audio_as_an/)

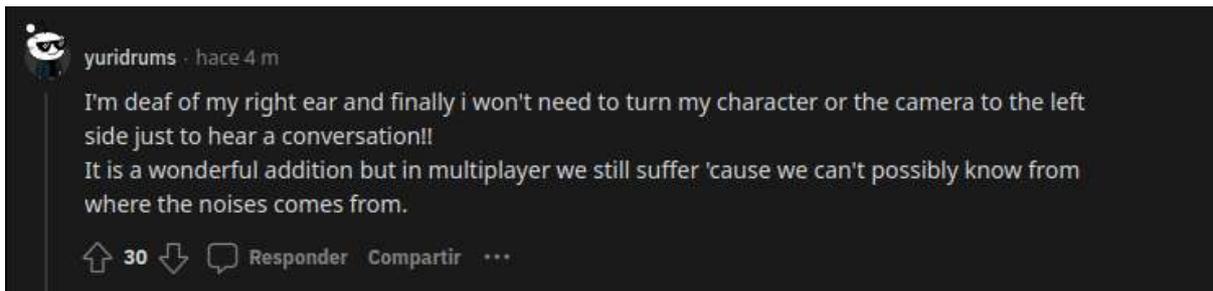


Figura 4.27 Un usuario del foro de Playstation 5 en el sitio Reddit, explica cómo se beneficia de la opción de accesibilidad de la salida monoaural de audio.

“Valorant” (2020) es un videojuego de disparos multijugador, y uno de los juegos de eSports más jugados actualmente. El juego incluye entre las opciones de audio una opción para cambiar la configuración del altavoz (mono o stereo). En un juego multijugador como “Valorant” es imprescindible poder localizar el origen de los sonidos (disparos, voces, explosiones) con facilidad para poder entender el contexto de cada partida. Sin embargo, el juego se puede adaptar a otro tipo de jugadores y circunstancias eligiendo una salida monoaural para el altavoz.

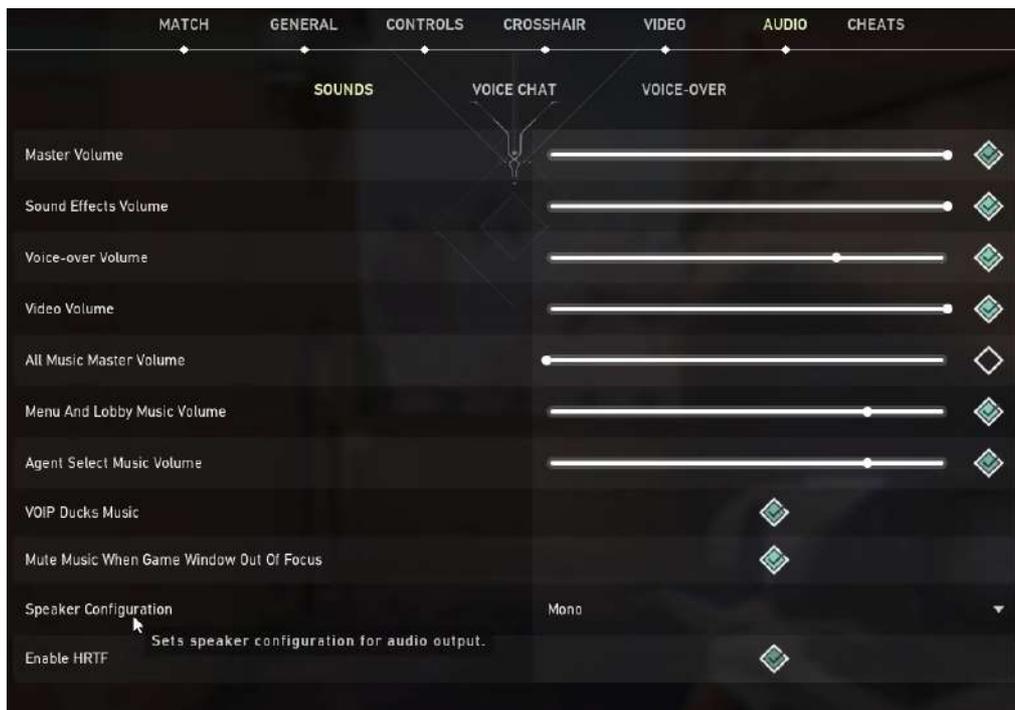


Figura 4.28 Pantalla de configuración de audio del juego “Valorant”, donde se permite cambiar la salida de audio “stereo” por “mono”.

G. Herramientas

- **Documentación sobre salida de audio en consolas:**
 - <https://support.xbox.com/es-AR/help/account-profile/accessibility/listen-in-mono>

- <https://www.playstation.com/es-ar/support/hardware/ps5-accessibility-settings/#mono>
- **Documentación de AudioSource en Unity:**
 - <https://docs.unity3d.com/Manual/class-AudioSource.html>
- **Documentación del motor de audio en Unreal Engine:**
 - <https://docs.unrealengine.com/5.1/en-US/audio-engine-overview-in-unreal-engine/>
- **Documentación de AudioStreamPlayer en Godot Engine:**
 - https://docs.godotengine.org/es/stable/classes/class_audiostreamplayer.html

Evitar la información monocanal

A. Resumen

Evitar utilizar un único canal de comunicación para transmitir información crítica sobre el juego al jugador. En el caso del audio, es una mala práctica utilizar sonidos, diálogos, música o cualquier otro tipo de sonido para brindar información al jugador sin que esta se vea reflejada o reforzada en otro canal de comunicación.

B. Problema

En el diseño de juegos, los “canales de información” se refieren a las formas de comunicación de la información del estado interno del juego a través de los sentidos fisiológicos del jugador como la visión, la audición y el tacto (Schell, 2019). También podemos describir a los canales de comunicación como los conjuntos de medios o formas en las que el jugador puede interactuar con el juego. Algunos ejemplos de estos son: comunicación visual (mediante los gráficos y el texto que el jugador puede ver en la pantalla), comunicación auditiva (mediante los sonidos, diálogos y música), o la comunicación háptica (por ejemplo mediante la vibración del joystick).

Con el término “información monocanal” nos referimos al tipo de información o datos que el juego transmite al jugador mediante un único canal de comunicación. Como explican Yuan y Folmer (2008), en todo juego se da un ciclo constante de eventos entre el juego y el jugador. Estos eventos pueden darse utilizando uno o más de los canales de comunicación mencionados. Por ejemplo, cuando el jugador realiza un salto en un juego de plataformas, está ingresando un comando al juego, por ejemplo presionando un botón. El juego toma esta entrada y la transforma en otro tipo de datos, información que el jugador va a recibir por múltiples canales. Por ejemplo, puede ver el resultado de la acción realizada mediante una animación del personaje en pantalla y un efecto de sonido que acompaña a esta animación. En este caso tenemos el canal visual y auditivo transmitiendo en simultáneo estímulos que en conjunto comunican la idea del salto al jugador.

La expresión “información monocanal en el audio” refiere a la situación en la cual la información que el juego transmite se representa exclusivamente mediante el canal auditivo. Desde el punto de vista de la accesibilidad auditiva, esto puede traer algunos problemas.

Por lo general este tipo de accesibilidad se centra en la utilización de subtítulos para asegurarse que la información transmitida mediante los diálogos pueda ser entendida por todos los jugadores. Sin embargo, los juegos no solo se basan en los diálogos, sino que existen otros tipos de recursos auditivos que también son utilizados para transmitir información al jugador. Muchos de los recursos del apartado auditivo pueden ser incluidos en los juegos por razones “estéticas”, pero algunos otros son esenciales para el gameplay. Los jugadores con dificultades auditivas que tienen problemas para recibir, entender o procesar este tipo de estímulos pueden perder la información crucial que se intenta transmitir, y en consecuencia no poder interactuar correctamente con el juego.

Imaginemos un juego en donde existen llaves y cofres, en donde cada cofre se abre con una llave en particular. El objetivo del juego es conseguir llaves y abrir todos los cofres. Supongamos que el jugador posee una llave y múltiples cofres, y quiere intentar abrir alguno de ellos con la llave. Cuando el jugador utiliza la llave sobre el primer cofre, se escucha un sonido de alerta que indica que la llave no puede utilizarse, pero no se observa ningún otro tipo de estímulo, vibratorio o visual. En cambio, cuando prueba con el segundo cofre, este se abre y el jugador obtiene una recompensa. En este ejemplo simple, la idea de que “el cofre no puede abrirse porque se está usando la llave incorrecta” se comunica al jugador únicamente mediante el canal auditivo, utilizando el efecto de sonido de una alerta. Un jugador que posee deteriorada su facultad auditiva, no escuchará este sonido y por lo tanto no comprenderá fácilmente la idea que el juego intenta transmitir: el cofre no se abrió porque la llave es incorrecta.

La “sobrecarga sensorial” es un fenómeno que se da en el cerebro y se caracteriza por la dificultad de procesar la información cuando se está expuesto a una gran cantidad de estímulos de variada intensidad (Mahmood, 2011). En videojuegos, la sobrecarga sensorial puede intensificar el problema de la información monocanal. Por ejemplo, en un FPS²⁷, pueden darse situaciones en donde hay múltiples personajes en pantalla, disparos, explosiones y comunicación entre jugadores. Esto puede generar una sobrecarga sensorial en el jugador, lo que dificultará captar y procesar un estímulo auditivo en particular y separarlo del resto del “ruido”, como por ejemplo el sonido de aviso de una granada cercana.

C. Solución

Trabajar de forma sinérgica y lograr la transmisión de un dato o información de forma confiable se logra mediante la utilización de múltiples canales de comunicación. La solución a este problema es reforzar la información auditiva mediante un canal de comunicación adicional. De esta forma, los estímulos son recibidos en conjunto por el jugador, y procesados de una manera más sencilla y confiable (Mahmood, 2011). Ya incluimos

²⁷ First Person Shooter, juego de disparos en primera persona

algunos ejemplos de estas soluciones en este mismo catálogo: implementar subtítulos para todos los diálogos, incluir subtítulos cerrados y facilitar la identificación del hablante también son soluciones al problema de la información monocanal. Cada uno de estos puntos del catálogo tratan de incrementar la cantidad de canales de información en la que se transmiten los datos críticos.

Una forma de reforzar los eventos en el canal auditivo es utilizar una representación gráfica para los mismos. Por ejemplo se pueden utilizar indicadores visuales para eventos como disparos, golpes, ataques y otros tipos de interacciones de los personajes con el ambiente virtual.

D. Beneficios

Este punto del catálogo apunta a solucionar un problema que afecta principalmente a los jugadores con discapacidad auditiva, ya que un evento con información transmitida exclusivamente mediante el sonido será parcial o completamente descartada por los jugadores con este tipo de discapacidad. Sin embargo, no solo apunta a solucionar un problema desde el punto de vista de la accesibilidad, sino que también mejora el diseño del juego. La utilización de múltiples canales para transmitir una misma idea permite que la información sea procesada más fácilmente por el jugador (Mahmood, 2011). Un buen diseño es consciente de los canales de comunicación disponibles al transmitir información al jugador, y entiende la importancia de manejar correctamente el caudal de información que se transmite con el objetivo de enfatizar elementos claves del juego y disminuir la sobrecarga sensorial (Baker, 2021).

Incluso cuando el jugador no presenta una discapacidad que anule o disminuya su facultad para procesar información en alguno de los canales de información, se puede ver beneficiado por la implementación de esta solución. Por ejemplo cuando se juega en condiciones inesperadas la utilización de eventos representados en múltiples canales mejoran la experiencia de juego. Algunas de estas condiciones pueden incluir: ambientes ruidosos, falta de iluminación, exceso de iluminación, jugar con volumen bajo o nulo, etc.

E. Clasificación

- A. Dificultad sensorial relacionada: Auditiva
- B. Dificultad en la interacción: Al recibir el estímulo, Al determinar la respuesta
- C. Principio de diseño universal: (1) Igualdad de uso, (2) Flexibilidad, (4) Información comprensible, (5) Tolerancia al error
- D. Dificultad de implementación: Medio

F. Ejemplos

En los juegos clásicos de la saga de Tomb Raider, desde “Tomb Raider” (1996) hasta “Tomb Raider: Chronicles” (2000), existe un ejemplo claro de uso de información monocanal: las

cerraduras. El gameplay asociado a la utilización de cerraduras de las puertas en los juegos clásicos se da de la siguiente manera:

1. Encontrar el artefacto o llave de la puerta.
2. Acercarse a la cerradura y presionar el botón de acción.
3. Se muestra el inventario al jugador. El jugador selecciona un artefacto para utilizar en la cerradura.
 - a. Si el artefacto es el correcto se abre la puerta mediante una animación
 - b. Si el artefacto no es el correcto, la puerta permanece cerrada y se reproduce un efecto de sonido que indica que el artefacto es incorrecto.

En el camino exitoso, el feedback que recibe el jugador es claro y conciso, ya que se utiliza una animación y un efecto de sonido para dar a entender que el artefacto es el correcto y que se abrió la puerta. Sin embargo, cuando el intento no es exitoso, el feedback que recibe el jugador es confuso. Para transmitir la idea de que el artefacto es incorrecto, se utiliza únicamente una línea de diálogo del personaje (“No”). No existe un indicador visual, o un subtítulo. Un jugador con dificultad auditiva pierde este único feedback y por lo tanto se pierde el mensaje que el juego intenta transmitir. En la figura 4.29 podemos observar el paso 3 del gameplay de las cerraduras, en donde no existe ningún feedback visual que acompañe a la línea de diálogo “No”. Algunas soluciones a este problema pueden ser: subtítular la línea de diálogo, agregar un sonido o alerta acompañado por su respectivo subtítulo cerrado, o mostrar un ícono en pantalla.



Figura 4.29 Ejemplo de información monocanal en el audio en “Tomb Raider” (1996)

En “Resident Evil” (1996) existe una mecánica similar involucrando puertas y llaves. Sin embargo en este juego se muestra un feedback tanto en el canal visual como en el auditivo cuando no es posible abrir una puerta. En la figura 4.30 podemos ver una captura de

pantalla del momento en el que el jugador intenta abrir una puerta cerrada con llave. Al igual que en “Tomb Raider”, se utiliza un efecto de sonido para indicar que la puerta no se abrió, pero adicionalmente se indica con un subtítulo la frase “La puerta está bloqueada”. A diferencia de “Tomb Raider”, se puede entender correctamente el resultado de la interacción con las puertas incluso apagando el sonido del juego.



Figura 4.30 Captura de pantalla del juego “Resident Evil” (1996) en donde se indica mediante un subtítulo que la puerta no se puede abrir.

Un problema común en los juegos FPS, en donde el jugador tiene el control de la cámara y se mueve libremente por un nivel, es transmitir información de eventos o alertar al jugador de cosas que están ocurriendo fuera de pantalla. Por lo general estos eventos se notifican al jugador únicamente a través del canal auditivo, pero para hacer accesible el combate se pueden realizar otras mejoras como agregar un subtítulo cerrado para el evento, agregar una ayuda visual, o ambas. Por ejemplo, el juego “Fortnite” (2017) incluye una representación de los sonidos en forma gráfica: un aro en el centro de la pantalla. De esta manera los jugadores con dificultad para entender y procesar los sonidos pueden utilizar los indicadores en pantalla para “escuchar visualmente” dentro del juego. En la figura 4.31 podemos ver un ejemplo de los indicadores visuales durante una partida de Fortnite, en donde se indica de forma gráfica la dirección, intensidad y tipos de sonidos dentro del mundo virtual.



Figura 4.31 Captura de pantalla de “Fortnite” (2017) en donde se implementan indicadores visuales para asistir a los jugadores con dificultades auditivas.

G. Herramientas

- **XBox Accessibility Guidelines**
 - <https://learn.microsoft.com/en-us/gaming/accessibility/xbox-accessibility-guidelines/103>
- **Game Accessibility Guidelines**
 - <https://gameaccessibilityguidelines.com/ensure-that-all-important-supplementary-information-eg-the-direction-you-are-being-shot-from-conveyed-by-audio-is-replicated-in-text-visually/>

Catálogo secundario

Existen otros tipos de pautas que también se pueden aplicar para mejorar la accesibilidad en videojuegos. Las pautas indicadas a continuación no forman parte del catálogo principal ya que no están relacionadas ni aportan únicamente a la accesibilidad auditiva.

Activar subtítulos automáticamente

Durante los últimos años se comenzó a considerar como buena práctica la activación por defecto de los subtítulos en los juegos. Esto permite que los subtítulos estén activados

automáticamente desde el inicio del juego, o que se permitan activar fácilmente antes de iniciar el juego. Si el juego comienza con una cinemática, o un diálogo entre personajes, es posible que algunos jugadores con dificultad auditiva se pierdan información importante del juego.

No mostrar demasiado texto en los subtítulos

La cantidad excesiva de texto también es un problema importante que suele dificultar la lectura. Más allá de los jugadores con discapacidades visuales, la cantidad de texto puede afectar negativamente en los jugadores con dificultades para leer o procesar la información visual.

Evitar información monocanal: visuales

De la misma manera que sucede con la información monocanal auditiva, se puede incurrir en un problema para los jugadores con dificultades visuales si se utiliza únicamente la comunicación visual para brindar feedback sobre el estado interno del juego. Los jugadores con dificultades visuales como ceguera parcial o total pueden perder información crucial. La solución implica reforzar la comunicación visual con eventos auditivos o hápticos. Por ejemplo, algunos juegos de lucha como Street Fighter (2010) permiten a los jugadores ciegos jugar escuchando los sonidos de los diferentes movimientos y ataques realizados por los personajes.

Compatibilidad con lectores de pantalla

No muchos juegos poseen esta característica esencial para algunos jugadores. La compatibilidad con lectores de pantalla permite a diversos tipos de jugadores poder interactuar mejor con el juego. Desde poder navegar por los menús del juego mediante el sonido, hasta poder escuchar diálogos y elementos interactivos de manera auditiva.

Permitir la configuración de los controles

Como vimos en el segundo capítulo, existe una diversidad muy grande de jugadores, cada uno con sus propias necesidades y dificultades específicas. Es importante brindar al jugador la posibilidad de personalizar cualquier aspecto del juego, incluyendo los controles. Algunos jugadores utilizan software especial para interactuar con la computadora, no solo el mouse y el teclado. Por ejemplo el jugador Rocky Stoutenburg “RockyNoHands”, utiliza un dispositivo llamado QuadStick que le permite controlar la computadora únicamente utilizando la boca. Por lo general los juegos traen preconfigurado múltiples plantillas de configuración del control: joystick, mouse y teclado, solo teclado, etc. Las APIs de los sistemas operativos

modernos permiten escuchar cualquier tipo de evento, incluyendo los eventos producidos por los controladores como el QuadStick. Aunque pueda parecer algo difícil de implementar, los motores de videojuegos más comunes se integran con estas APIs, con lo cual se facilita su utilización.

Controles amigables

En esta pauta podemos incluir varios puntos importantes relacionados al control del juego. En principio, se intenta que el juego sea fácil de controlar. Esto quiere decir que por ejemplo, se intenta que todas las acciones del juego se realicen utilizando los controles de forma intuitiva y sin demasiadas combinaciones de botones. Por ejemplo, se puede considerar excesivo tener que presionar más de 2 botones al mismo tiempo para realizar una acción. Esto incluye también evitar la utilización de “Quick Time Events”²⁸, ya que por lo general estos tipos de gameplays pueden representar una dificultad muy elevada para cierto tipo de jugadores. De la misma manera, se intenta evitar el “button mashing”²⁹ ya que los jugadores con dificultad motriz, muchas veces, encuentran imposible avanzar en este tipo de gameplay. La solución a estos problemas es no incluir gameplay de este estilo en el juego, o permitir apagarlo desde un menú de configuración.

En segundo lugar podemos mencionar la capacidad del juego para poder utilizar múltiples formas de control al mismo tiempo. Más allá de permitir controlar el juego con distintos tipos de controladores (mouse, teclado, joystick, etc), es una buena práctica permitir utilizar estos controladores de forma indistinta. Por ejemplo, un jugador puede empezar utilizando mouse y teclado, y a mitad de la partida conectar un joystick y continuar jugando con este dispositivo. Esto permite a algunos jugadores con necesidades especiales contar con la asistencia de otra persona en momentos clave del juego, por ejemplo durante la navegación del menú.

Usar el mismo control para el juego y la UI

Esta pauta se basa en permitir al jugador utilizar el mismo tipo de controlador (mouse, teclado, joystick) para controlar el juego y la UI. Esto quiere decir que si un jugador está utilizando el teclado durante el gameplay, también se debe permitir la navegación de los menús con el teclado. De esta manera el jugador no tiene que cambiar de dispositivo durante toda la partida.

²⁸ Evento de tiempo rápido: se refiere a un momento del juego en el que el jugador debe responder de manera sorpresiva y rápida a un evento del juego.

²⁹ Presionar repetidamente un botón para realizar una acción dentro del juego.

Permitir la utilización del modo ventana

Algunos juegos modernos no permiten ejecutar el juego en modo ventana. Esto implica que el jugador pierde la posibilidad de utilizar herramientas o complementos ajenos al juego, ya que se toma el control completo de la pantalla. Si el jugador utiliza software especial para interactuar con la computadora, la ejecución del juego a pantalla completa podría obstaculizar la utilización de estas herramientas.

Asistencias in-game

Dentro del juego, se pueden brindar diversos métodos de asistencia a los jugadores. Desde posicionamiento automático de cámaras, apuntado automático de la mira, modos de juego con dificultades más sencillas, modos de juego en donde el jugador no recibe daño, ajuste de la velocidad del juego, implementación de checkpoints, listado de objetivos, etc.

Es muy importante que estas asistencias se puedan habilitar de manera separada. Es decir, que cada tipo de asistencia se pueda apagar o prender de acuerdo a las necesidades del jugador. Esto permite que los jugadores customicen la experiencia de juego de acuerdo a sus necesidades personales, sin penalizar a los jugadores que solo necesitan un único tipo de asistencia.

Permitir pausar el juego

Algunos jugadores pueden necesitar un tiempo extra para observar la escena y tomar una decisión. Para estos jugadores la pantalla de pausa puede ser una buena herramienta para tomar un respiro y analizar el siguiente movimiento. De acuerdo al tipo de juego, es importante analizar cómo será el diseño de la pantalla de pausa, pensando por ejemplo en no tapar la escena con componentes de UI o con un fondo opaco.

Incluir tutoriales interactivos

El tutorial sirve como punto de entrada para cualquier tipo de jugador. Debe enseñar las reglas básicas del juego dentro de un ambiente seguro y sin penalizaciones para el jugador. Esto permite al jugador acomodarse al juego mientras se acostumbra a los controles y al contexto del juego. Además permite a los jugadores identificar qué tipos de asistencias u opciones de accesibilidad necesitará, de manera de poder activarlas antes de comenzar el juego principal.

Permitir ocultar elementos de violencia o sangre

Algunos jugadores pueden verse afectados por este tipo de elementos visuales o auditivos, por lo que es una buena idea permitir personalizar este tipo de contenido.

Ofrecer múltiples modos de dificultad

Siempre apuntando a la adaptación de la experiencia del juego a las capacidades y necesidades de cada jugador, es una buena idea implementar varios modos de dificultad. De acuerdo al tipo de juego, se pueden realizar modos en donde el daño sea menor, donde los enemigos aparezcan en menor frecuencia o donde no existan, modos donde el jugador no pueda morir, etc. Es importante prestar atención al nombre de estos modos de juego, ya que durante mucho tiempo se utilizaron frases despectivas o con connotaciones negativas para este tipo de modos, por ejemplo llamándolos “Modo bebé”, “Modo super fácil”. En lugar de esto se podría utilizar nombres como “Modo explorador”, o “Modo historia”, y además explicar cuáles son las diferencias entre cada tipo de modo de dificultad.

Capítulo 5: Accesibilidad en Godot Engine

En este capítulo se profundiza en el motor para desarrollo de videojuegos Godot Engine, y su relación con la industria del desarrollo independiente de videojuegos. Además se examinarán las propuestas de accesibilidad que ofrece este motor, así como también las carencias y mejoras pendientes que tiene el mismo. Por último se realiza una propuesta y desarrollo de un plugin que agrega soporte para subtítulos a este motor.

5.1 Sobre Godot Engine

Godot Engine es un motor de videojuegos de uso general, de código abierto y publicado bajo la licencia MIT. Permite crear todo tipo de aplicaciones y juegos 2D y 3D, y está disponible para los sistemas operativos Windows, OS X y Linux. Este software nació y comenzó a utilizarse inicialmente dentro de la empresa argentina de videojuegos OKAM Studios, pero en febrero del 2014 se abrió al mundo de los proyectos de código abierto y se publicó en Github. Esto le permitió crecer de una forma acelerada a lo largo de los últimos años, recibiendo contribuciones y mejoras por parte de desarrolladores de todo el mundo. Desde el principio se logró construir una comunidad de colaboradores que se mantiene fuerte hasta la actualidad. Hoy es uno de los motores para la creación de videojuegos más utilizados por desarrolladores independientes de todo el mundo, y uno de los proyectos FOSS (Free and Open Source Software) más exitosos en Github. Desde hace años que las contribuciones al proyecto (código principal, documentación y proyectos relacionados) se mantienen constantes, comparables con otros proyectos grandes de software libre como Blender (un software especializado en la edición y creación de gráficos computarizados) o GIMP (un editor de imágenes). En la figura 5.1 se compara el nivel de actividad de los proyectos de software libre Blender, GIMP y Godot, en términos de contribuciones. Se puede observar que Godot mantiene un buen nivel de actividad con una importante cantidad constante de contribuciones en los últimos años.

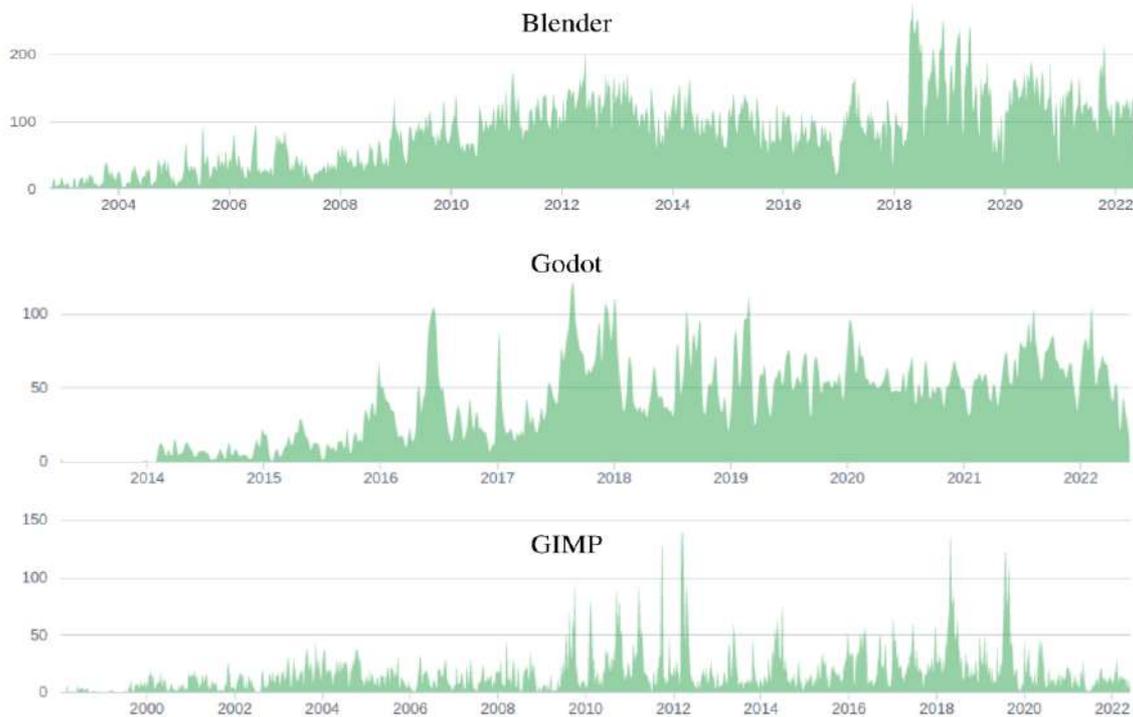


Figura 5.1 Cantidad de contribuciones (en número de commits) a lo largo del tiempo en los proyectos Blender, Godot y GIMP, tomado de los repositorios correspondientes en GitHub.

Godot permite crear videojuegos, aplicaciones de escritorio y para dispositivos móviles de una forma sencilla, a través de un IDE desde el cual se accede a todas las herramientas que ofrece. Las aplicaciones creadas con Godot se pueden exportar a diversas plataformas como Windows, Linux, Mac OS, dispositivos Android y iOS, y consolas de juegos como Xbox y PlayStation a través de empresas de terceros especializadas en llevar los proyectos hechos en Godot a estas consolas. El IDE utilizado para crear juegos en Godot es muy liviano, por ejemplo la última versión (3.4.5) lanzada el 2 de agosto del 2022 pesa tan solo 38MB en su versión para plataformas Linux. Este editor engloba todas las herramientas necesarias para trabajar con el motor, navegador de archivos, editor de animaciones, shaders, editor de escenas 2D y 3D, soporte para masterización del audio, y editor de scripts (figura 5.2).

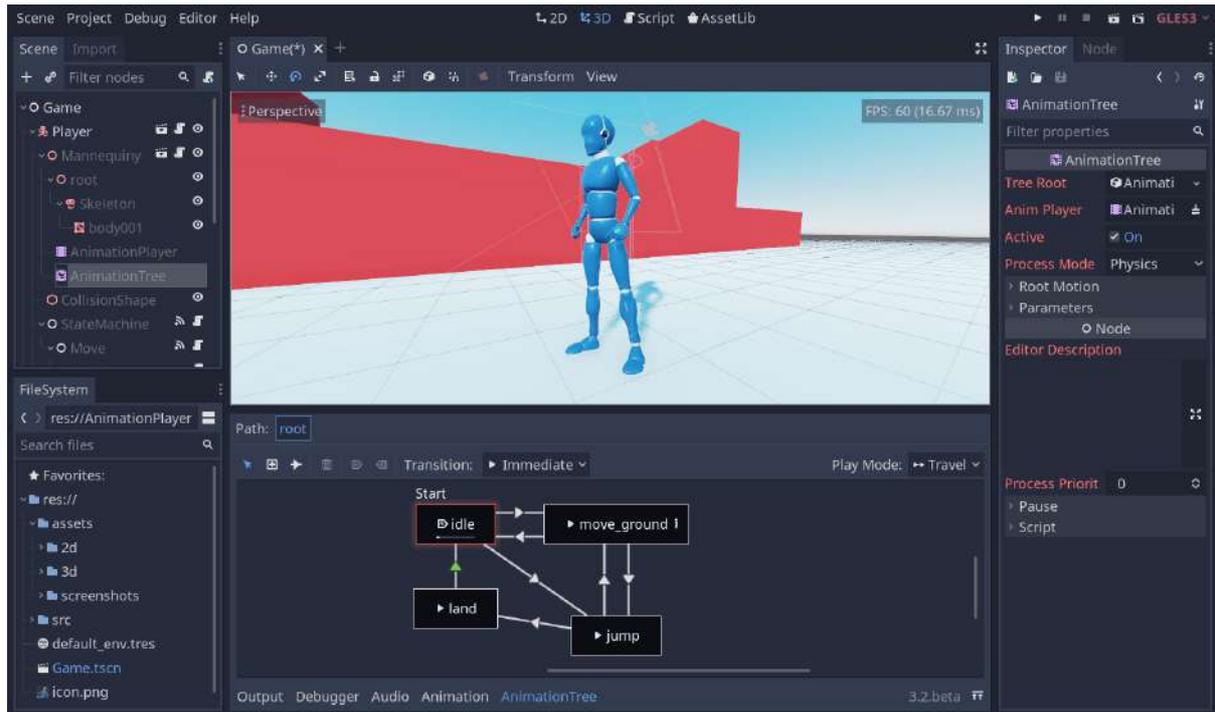


Figura 5.2 Vista general del IDE de Godot, en donde se está editando la animación para un personaje 3D.

Uno de los pilares de Godot es el diseño simple, extensible y accesible. Esto se puede observar en varios de los aspectos que rodean a este proyecto: la colaboración, la arquitectura del motor, el diseño de la UI, la manera de utilizar nodos y escenas como componentes reutilizables, y en GDScript, el lenguaje de scripting creado especialmente para ser utilizado dentro del motor. La facilidad para trabajar con el motor, la simpleza de su IDE, y la cantidad de recursos disponibles para aprender a utilizarlo impulsó fuertemente la evolución de este proyecto, tomando cada vez más relevancia dentro del mundo del desarrollo de videojuegos. Algunas aplicaciones y herramientas creadas con este motor incluyen: “Material Maker”, “RPG in a Box” y “Pixelorama”. Algunos juegos creados con Godot Engine son: “Primal Light” (2020), “Haiki” (2021), “Ex-Zodiac” (2022) y “Sonic Colors Ultimate” (2021).

Desde el punto de vista de la colaboración podemos decir que Godot es un proyecto acogedor y con buena predisposición a recibir aportes de nuevos colaboradores. Cualquier desarrollador que esté interesado en aportar su tiempo y esfuerzo puede contribuir al proyecto cuenta con una amplia y variada documentación y material oficial a su disposición. Existen listas públicas y backlogs de tareas pendientes, reportes de bugs y propuestas de mejoras, todo administrado y mantenido por la comunidad. Estos puntos son importantes para lograr que un nuevo colaborador que se suma al proyecto lo haga de una manera óptima.

Un equipo de desarrollo de videojuegos por lo general cuenta con trabajadores de distintas disciplinas: artistas 2D, diseñadores de niveles, programadores, animadores, etc. En cuanto

a los roles de los usuarios podemos decir que el editor de Godot es también inclusivo, dado que puede ser utilizado por usuarios con diferentes niveles de necesidades, expectativas y conocimientos técnicos indistintamente. El diseño del motor basado en escenas y nodos permite encapsular responsabilidades y por lo tanto ocultar complejidades que podrían ser una barrera para el usuario. Por ejemplo, Godot permite al diseñador de niveles crear un nivel completo sin tener conocimientos de programación o de GDScript, a la vez que permite a un animador 2D trabajar en las animaciones del personaje principal sin necesidad de entender cómo está implementado internamente.

Según el Informe General del Observatorio de la Industria Argentina de Desarrollo de Videojuegos del año 2021 (Universidad Nacional de Rafaela, 2021), Godot Engine fue uno de los motores más utilizados durante el 2021, ya que un 10% de los proyectos realizados durante ese año lo utilizan, un porcentaje que solo queda por detrás de otros motores más populares como Construct, Unreal Engine y Unity³⁰. Cabe destacar que Godot es el motor de videojuegos de tipo open-source más utilizado en Argentina, por delante de Phaser y Cocos2D³¹.

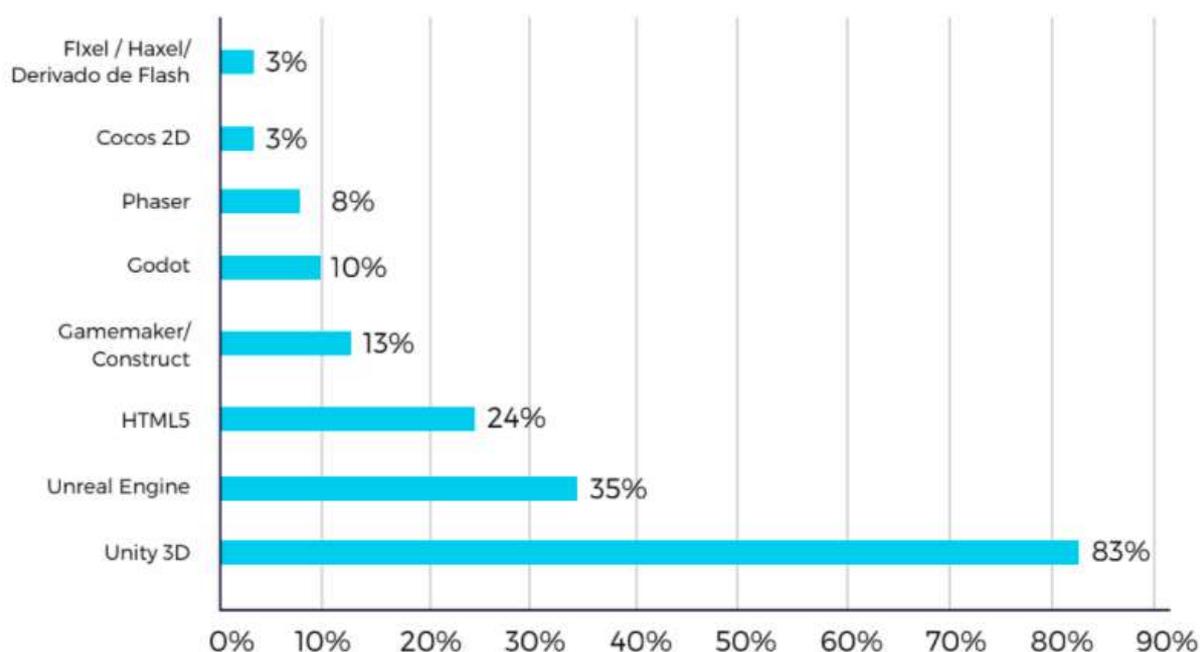


Figura 5.3 Frameworks y Engines. Informe del año 2021 del Observatorio de la Industria Argentina de Desarrollo de Videojuegos.

La serie de análisis de videojuegos conocida como “Game Maker’s Toolkit”, además de realizar análisis profundos sobre varios aspectos de diseño de videojuegos, impulsa anualmente desde el 2017 una “game jam” en donde los desarrolladores tienen 48 horas para diseñar y crear un videojuego ajustándose a un tema relacionado al diseño de

³⁰ <https://www.construct.net/>, <https://www.unrealengine.com/>, <https://unity.com/>,

³¹ <https://phaser.io/>, <https://www.cocos.com/en/cocos2dx>

videojuegos, revelado al inicio de la jam. Godot Engine está creciendo su popularidad en la comunidad independiente, esto se puede observar a partir de las estadísticas generadas en las últimas ediciones de las game jams impulsadas por GMTK (figura 5.4). De la figura se puede observar que entre la comunidad de desarrolladores que participan de estas jams (GMTK Game Jam, 2021), los motores más utilizados son: Unity, GameMaker, Construct y Godot Engine. En cuanto a la utilización de Godot, vemos que crece año a año: 7.7% en 2019, 12.2% en 2020, 13.1% en 2021 y 15.6% en 2022. Con respecto a los otros motores, vemos que el único que se mantiene estable es Unity, a diferencia de GameMaker y Construct, quienes están bajando en popularidad.

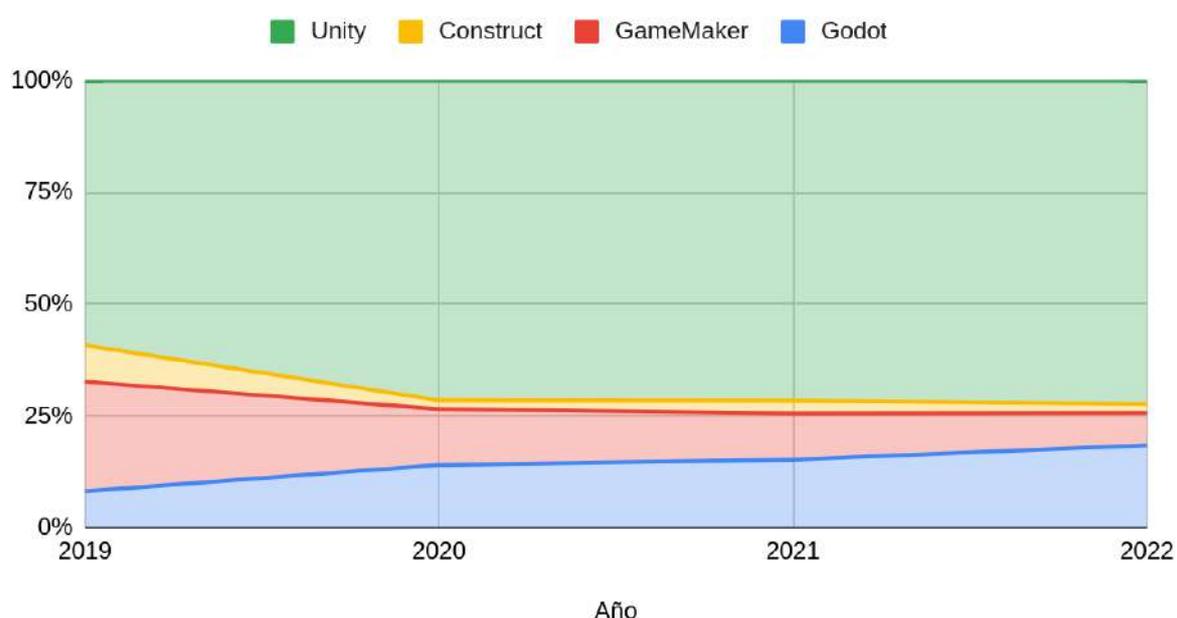


Figura 5.4 Comparativa de motores utilizados en GMTK Game Jam en los años 2019, 2020, 2021 y 2022 (Game Maker's Toolkit Twitter account, 2022).

Estos datos estadísticos muestran que Godot es uno de los motores más utilizados por los desarrolladores independientes, no solo en Argentina, sino a nivel mundial. Además podemos ver que el motor crece en popularidad, cantidad de contribuidores y calidad del proyecto desde su inicio. Esto lo vuelve un punto importante en el camino hacia una mejor consideración de la accesibilidad entre los creadores de la comunidad independiente de desarrolladores de videojuegos.

5.2 Sobre la accesibilidad en Godot Engine

Godot Engine es un motor que posee en la actualidad un conjunto muy importante de herramientas que le permiten a los desarrolladores trabajar cómodamente en aspectos técnicos del desarrollo de videojuegos, como pueden ser el arte en 2D o 3D, animaciones, shaders, audio, y otras más. En el apartado de la accesibilidad Godot todavía no posee las herramientas necesarias para que un desarrollador pueda construir accesibilidad con la

misma facilidad que trabaja en alguno de los aspectos técnicos mencionados anteriormente. Para entender el nivel de accesibilidad que provee Godot, primero es necesario entender cómo funciona su editor o IDE y el comportamiento de los componentes de UI dentro del motor.

Quizás una de las características más valiosas que tiene Godot es que su IDE está creado utilizando Godot mismo. Es decir que para crear el editor se utilizó la misma librería de componentes que utilizan los desarrolladores de videojuegos para crear sus aplicaciones dentro de Godot. Al contrario que en otros motores de videojuegos, donde la GUI del editor está construida sobre una librería totalmente diferente, en Godot se utilizan los componentes de la UI tanto para armar juegos como para el editor mismo. Podemos destacar que las mejoras y los avances que se realicen sobre los componentes y widgets del editor también quedan disponibles para los desarrolladores que utilicen estos mismos componentes en sus juegos. Por ejemplo, si se soluciona un bug en un componente de input de texto del editor, este bug estará solucionado también en el input de texto dentro del juego.

Esto trae ciertas ventajas dado que permite utilizar el mismo código y componentes tanto para crear un juego como para extender el editor. Además significa que los componentes de la interfaz del editor están disponibles para todas las plataformas en las que Godot está disponible, facilitando la portabilidad del editor a otras plataformas, incluso existe una versión del editor que funciona dentro del navegador web.

También el hecho de implementar una librería de GUI trae consigo desventajas en Godot. La más notoria es que la librería es responsable de implementar todos los comportamientos más comunes que podemos esperar de los componentes y widgets, por ejemplo la navegación y el uso de atajos de teclado, soporte para pantallas táctiles, y por supuesto las opciones de accesibilidad. En un proyecto donde la GUI es la parte principal, como es el caso de la librería GTK, es común que se dé importancia a la implementación de un buen soporte de accesibilidad. Si bien en Godot la GUI es robusta y está bastante desarrollada, no es el aspecto que genera más interés dentro de un motor de videojuegos en crecimiento, por lo que el soporte de accesibilidad queda relegado ante otros desarrollos más grandes. Con el tiempo algunos colaboradores comenzaron a recopilar problemas y se crearon algunas propuestas para mejorar Godot en este aspecto. Algunos desarrolladores incluso crearon plugins no oficiales que mejoran el soporte de accesibilidad de los componentes.

Plugin con soporte para lectores de pantalla

Un ejemplo de mejoras de accesibilidad en Godot Engine es el plugin “Godot Accessibility” creado por Nolan Darilek, el cual incorpora mejoras de accesibilidad al editor de Godot. Este

plugin agrega capacidad TTS³² a Godot, y la implementación de un lector de pantalla integrado. La lectura de pantalla está implementada mediante un nodo llamado “ScreenReader”, el cuál contiene la lógica necesaria para transformar las interfaces visuales de Godot en algo accesible y navegable de una forma similar a la que se hace con un lector de pantalla tradicional.

El plugin incluye mejoras para que la navegación de la UI sea accesible mediante el teclado y mediante gestos comunes utilizados en lectores tradicionales. Además intenta solucionar varios problemas relacionados al manejo de eventos presentes en Godot, por ejemplo, hace que los elementos de UI tomen foco automáticamente cuando una escena es inicializada. El plugin también intercepta automáticamente todos los eventos de entrada de las pantallas táctiles para emular el funcionamiento de la navegación en lectores de pantalla al estilo Android e iOS, donde se navega utilizando los gestos “swipe” y “double tap”.

El plugin también expone al desarrollador una interfaz para interactuar con el sistema operativo y utilizar sus capacidades de texto a voz. Esto es utilizado por el plugin para poder leer en voz alta el contenido de varios de los componentes de UI más comunes.

¿Qué tan accesible es Godot Engine?

Pensar la accesibilidad al momento de diseñar nuestro software es una pieza fundamental, ya sea un videojuego u otro producto de software. En el caso de Godot, además de pensar en crear videojuegos accesibles, también se analiza en este apartado su accesibilidad para con los desarrolladores que lo utilizan para crear videojuegos.

Una de las falencias más grandes de Godot es que la interfaz del editor no es compatible con lectores de pantalla, siendo una barrera para muchos desarrolladores que utilizan este tipo de tecnología. El plugin de Nolan soluciona este problema, inyectando el nodo ScreenReader dentro del editor. Esto habilita, en teoría, la navegación mediante lector de pantalla en todo el editor, utilizando los controles comunes “Tab” y “Shift + Tab” en conjunto con las teclas de dirección del teclado para navegar entre las diferentes secciones del editor, y además pronunciando en voz alta el nombre y contenido del componente cada vez que toma el foco. Es un avance importante en soporte de la accesibilidad de la GUI y vuelve al IDE mucho más amigable a los usuarios con capacidades visuales disminuídas. Sin embargo, siguen existiendo algunos problemas en la interfaz del editor que son difíciles de solucionar mediante el uso de plugins. Estos problemas están ligados al funcionamiento de los componentes mismos y del diseño del editor, por lo que deben ser solucionados por la implementación del motor en sí y no mediante plugins.

Uno de los problemas más importantes es que el editor es demasiado complejo como para desarrollar un juego completo sin la utilización del mouse. Existen gran cantidad de

³² Text-To-Speech: texto a voz

secciones dentro del editor donde la navegación mediante el teclado es complicada, o simplemente imposible. La vista del sistema de archivos y el editor de scripts son dos de las áreas particularmente complicadas de controlar.

Algunos de los controles son muy complejos para manipular (incluso utilizando el mouse) como por ejemplo el editor de gradientes, el editor de árbol de animaciones y el editor de polígonos (figura 5.5). Para algunos de estos controles se permite la edición mediante el ingreso de valores por teclado, como por ejemplo para la edición de gradientes. La alternativa para estos controles altamente visuales, es una edición manual de los archivos de configuración en un software externo, algo que se podría evitar ofreciendo un diseño más inclusivo.

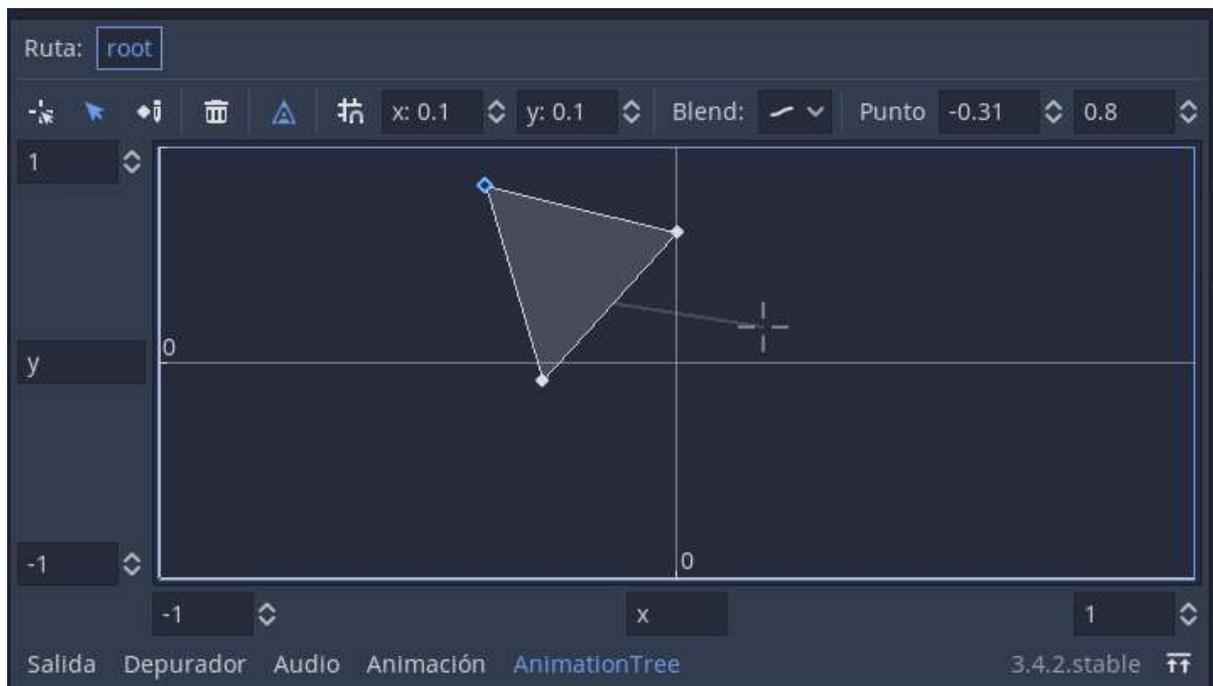


Figura 5.5 El editor de árboles de animaciones es demasiado complejo para ser controlado vía teclado.

Uno de los aspectos más difíciles de la navegación vía teclado es la pérdida del foco de los componentes. Esto es un problema que se fue mejorando en las nuevas versiones, pero que no está completamente eliminado. La navegación de la interfaz vía teclado depende de que los componentes de la UI no pierdan el foco, algo que sucede con frecuencia dentro del editor. Un ejemplo de este problema se da cuando se lanza la preview del juego. Cuando se abre la ventana del juego, el editor pierde el foco, y no lo obtiene incluso cuando la ventana del juego se cierra. Otro ejemplo podemos verlo cuando navegamos por el inspector de propiedades. Una vez que navegamos hasta una propiedad de ingreso de texto en modo multilínea (por ejemplo editando un componente Label), las combinaciones de teclado “Tab” y “Shift + Tab” que se usan para navegar entre componentes dejan de tener efecto, y comienzan a usarse para ingresar el carácter de control “Tab” dentro del texto. Para poder

ganar el control de la navegación vía teclado es necesario utilizar el mouse para clickear fuera del componente.

No todo es malo en cuanto a la accesibilidad en Godot. Como aspecto positivo podemos mencionar que a medida que fue creciendo la adopción de este motor dentro de la comunidad de desarrolladores, se comenzó a prestar más atención a algunos puntos básicos sobre la accesibilidad. Las nuevas versiones del editor permiten personalizar la configuración del tema que utilizan todos los componentes del editor, pudiendo elegir entre temas claros, oscuros y de alto contraste, lo cual permite a los usuarios con dificultades visuales cambiar los colores del editor por unos que se adecuen a sus necesidades (figura 5.6). Adicionalmente se permite ajustar los colores de los íconos y las fuentes (claros u oscuros), ajustar el tamaño de los bordes, agregar espaciado extra a los componentes de UI, o incluso ajustar el contraste de los colores mediante un slider.

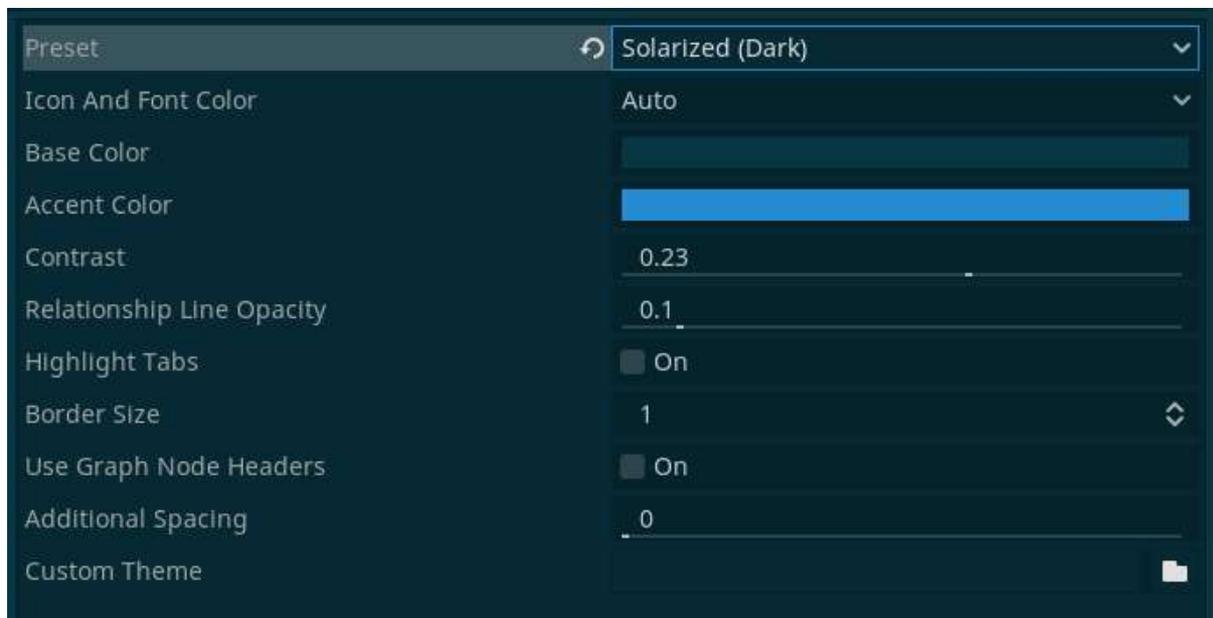


Figura 5.6 Opciones de configuración para la UI del editor principal.

En cuanto a las fuentes utilizadas, se permite cambiar la tipografía, color y tamaño de las fuentes presentes en todo el IDE y en el editor de código.

Las personas con dificultad visual más grave pueden aplicar un factor de escala en la UI que afecta todos los componentes del IDE, generando un efecto de ampliación sobre el editor y facilitando la identificación de los elementos en pantalla. Adicionalmente se permite cambiar el tamaño de espacios, barras de división y otros elementos que facilitan la separación e identificación de las secciones y componentes del editor.

Una de las mejoras más útiles incorporadas al editor es la posibilidad de crear atajos de teclado para las acciones más comunes (figura 5.7). Además de mejorar la experiencia de uso y la rapidez para el desarrollador, esto permitiría por ejemplo a un usuario con problemas motrices crear atajos para las acciones más complicadas de realizar dentro del

editor, eliminando barreras en la utilización de algunas de las herramientas. Por el momento solo se permiten atajos de teclado, pero en un futuro se podrían utilizar eventos provenientes de joysticks u otros tipos de tecnología de asistencia para controlar el IDE.

Nombre	Binding
Animation Editor	
Aplicar Reset	None
Eliminar Selección	Delete
Duplicar Selección	Control+D
Duplicar Transpuesto	Control+Shift+D
Ir a Paso Próximo	Control+Right
Ir a Paso Previo	Control+Left
Animation Player Editor	

Figura 5.7 Godot permite mapear acciones comunes dentro del editor a atajos de teclado.

Teniendo en cuenta que los componentes de GUI con los que interactúan los desarrolladores dentro de Godot son los mismos que utilizan los jugadores dentro del juego, los conceptos, ventajas y desventajas descritos anteriormente también aplican para las interfaces dentro de los juegos. Por ejemplo, en el caso del plugin “Godot Accessibility” las capacidades de lector de pantalla y TTS están también disponibles para los juegos creados. Actualmente se está trabajando en el lanzamiento de la versión 4 de Godot Engine. Esta nueva versión traerá mejoras de todo tipo: refactorings en el código principal del motor para hacerlo más sencillo de extender, nuevas tecnologías de renderizado, mejoras en el editor y mejoras de performance en muchas de sus funcionalidades (Verschelde, 2022). El editor es uno de los aspectos del motor que recibió más mejoras, apuntando a la usabilidad y accesibilidad de las herramientas. Por ejemplo, se re-implementó el editor de mapas para mejorar la integración con el editor de escenas 2D, facilitando agregar animaciones y trabajar con metadata. En relación a la accesibilidad, una de las mejoras más importantes es la integración de la función “Command Palette” (Verschelde, 2021), una forma de acceder mediante atajos de teclado a comandos y acciones que usualmente no tienen atajos o están escondidas en algún menú. Algunas de las features que incluye son: un atajo simple para abrir el buscador de comandos, una forma de filtrar y encontrar los comandos, y una forma de ver los atajos de teclado asociados a los comandos. La segunda mejora importante en cuanto a accesibilidad, es la incorporación de un tema visual que aplica nuevos colores y fuentes al editor. Este nuevo tema provee una sensación más moderna al IDE, y además utiliza un nuevo esquema de colores, mejorando la accesibilidad de todo el editor. Adicionalmente, el editor también se beneficia del nuevo renderizado de texto y el soporte para lenguajes con escritura de derecha a izquierda, lo que abre la puerta a desarrolladores y jugadores de otras regiones del planeta (figura 5.8).

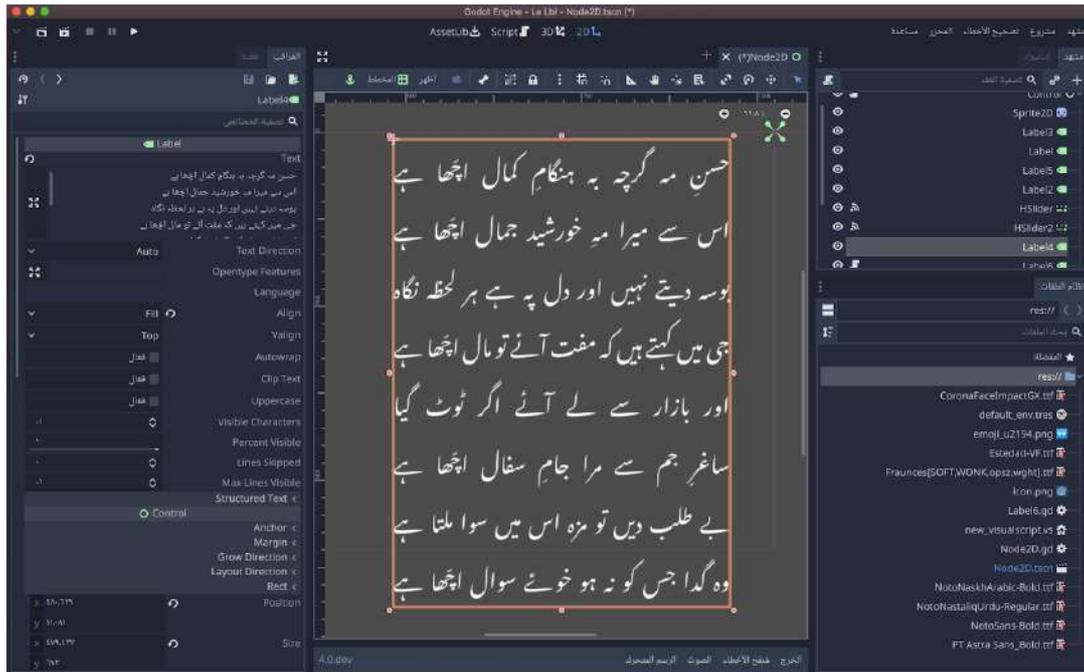


Figura 5.8 El editor de Godot 4 soportará lenguajes con otros sistemas de escritura.

Futuras mejoras de accesibilidad de Godot

Aunque la próxima versión de Godot (4.0) brindará mejoras en la usabilidad y permitirán a los desarrolladores y jugadores lograr una mejor interacción con el motor y con los juegos creados, todavía quedan algunos aspectos sobre accesibilidad que no fueron cubiertos. El usuario George Marques (nombre de usuario vnen) en Github redactó una propuesta (Improve accessibility features for keyboard navigation, UI scaling, TTS, voice recognition, colorblindness and subtitles, 2020) de mejoras para el motor, centrada en mejorar la accesibilidad de Godot. Las propuestas mencionan: mejoras en el input re-mapping, solucionar problemas con la navegación mediante el teclado, agregar soporte para gestos táctiles de navegación, escalado de fuentes y de UI, soporte para text-to-speech y speech-to-text, filtros y herramientas para trabajar el daltonismo, y el soporte para subtítulos. Algunas de las problemáticas enunciadas en esta propuesta fueron, parcialmente, resueltas en nuevas contribuciones al motor. Por ejemplo, la versión 4 de Godot ofrecerá soporte text-to-speech, UI scaling y mejoras en el renderizado de fuentes de texto. Pero otras siguen sin recibir algún tipo de soporte o solución dentro del motor, como por ejemplo soporte para trabajar el problema de la percepción del color, o los subtítulos.

5.3 Soporte de subtítulos en Godot

Motivaciones

La falta de subtítulos en los juegos independientes es bastante común. Por un lado, algunos desarrolladores de videojuegos suelen no molestarse en incluir opciones de subtítulos en sus juegos cuando ya incluyen el diálogo en forma hablada (Mangiron et al, 2014). El trabajo que conlleva transcribir todo el diálogo del juego a archivos de subtítulos puede ser muy grande para un equipo pequeño de desarrolladores, por lo que se priorizan otras tareas. Por el lado técnico, puede suceder que los recursos multimedia se reproduzcan mediante librerías demasiado simples, que dificultan la sincronización del audio o video con los subtítulos, o que directamente no soportan la reproducción de subtítulos.

Para los jugadores con dificultades auditivas, o para aquellos jugadores que no comprenden el lenguaje utilizado en el doblaje, la falta de subtítulos puede resultar en una barrera común en los videojuegos (Mangiron et al, 2014). En el caso de Godot no existe un soporte oficial para la creación de subtítulos. Según algunos colaboradores, las herramientas que provee el motor para mostrar texto en pantalla deberían ser suficientes para poder implementar subtítulos dentro del juego (Marques, 2020). Sin embargo, no deja de ser cierto que la falta de soporte oficial para la creación de subtítulos influye tremendamente en la ausencia de subtítulos en los juegos independientes creados en Godot.

Como se describe en el Capítulo 2, una gran cantidad de jugadores en Argentina experimentan dificultades relacionadas a la audición (2,1%), solo por debajo de otras dificultades con más presencia en la población como las motrices y visuales (Agencia Nacional de Discapacidad, Estudio Nacional sobre el Perfil de las Personas con Discapacidad, 2018). Dado que la falta de subtítulos es un problema muy común en videojuegos, y que Godot no provee un soporte oficial para la creación de subtítulos, es una buena oportunidad para trabajar en el objetivo propuesto para esta tesina: disminuir las barreras relacionadas a las discapacidades auditivas en videojuegos. Por estas razones considero necesario tomar la responsabilidad de crear un soporte para poder incluir de forma sencilla subtítulos en las aplicaciones y juegos creados con Godot 3.

Requerimientos

Para poder brindar soporte de subtítulos dentro de Godot, es necesario plantear primero cuáles son las problemáticas que se desean resolver. Esto nos ayudará a delimitar los requerimientos, y lograr diseñar una solución satisfactoria. Se espera que el soporte de subtítulos en Godot facilite las siguientes dos funcionalidades:

Mostrar los subtítulos dentro de juegos y aplicaciones creadas con Godot Engine:

significa brindar un soporte íntegro para los subtítulos, de forma que se puedan utilizar en cualquier contexto de juego y en cualquier plataforma soportada por Godot. El subtítulo debe poder mostrarse tanto en cinemáticas, como durante el gameplay, debe verse exactamente igual en Windows como en Android, sin tener que utilizar configuraciones adicionales o setups complejos.

Editar los subtítulos desde dentro del editor de Godot: sería una dificultad para el desarrollador trabajar los subtítulos en una herramienta externa, en este sentido la edición de los subtítulos (contenido, tipografía, tamaño, etc) se ofrecerá desde dentro del IDE de Godot.

Estos dos objetivos de diseño se pueden dividir en los siguientes requerimientos concretos:

Trabajar con los subtítulos de una forma sencilla e intuitiva.

Este requerimiento de diseño apunta a que el desarrollador utilice la herramienta de la misma forma sin importar el contexto: en subtítulos de diálogos entre personajes, en subtítulos de cinemáticas, en caso de implementar closed caption, etc. Adicionalmente la edición del contenido de los subtítulos debe poder realizarse de forma sencilla, sin necesidad de utilizar un editor externo para cambiar alguna de las características del mismo, dado que el subtítulo se verá igual en todas las plataformas en donde se ejecute el juego. Por último este objetivo también hace referencia a la facilidad para poder integrar la solución dentro del proyecto, es decir la solución no requerirá una instalación o configuración complicada para el desarrollador.

Los subtítulos deben estar integrados con el resto de la UI.

En cuanto a la parte visual, es importante que los subtítulos estén integrados naturalmente con el sistema de interfaces de usuario dentro del motor. El plan es no utilizar técnicas de renderizado especiales no soportadas por el UI de Godot, como renderizar los subtítulos mediante shaders, o mediante texturas 2D. Godot permite controlar múltiples aspectos de los componentes de la UI, como son la posición, tamaño, y rotación de los elementos. Además brinda soporte para escuchar y procesar eventos del usuario, y controlar el orden de renderizado en pantalla (z-index). Una de las funciones más fuertes que posee Godot es la posibilidad de animar cualquier propiedad de un nodo, ya sean nativas del nodo como la posición, rotación o tamaño, o agregadas por el desarrollador. Si integramos la solución de subtítulos a la jerarquía de nodos de Godot, podemos utilizar animaciones sobre cualquiera de sus propiedades, lo cual es un beneficio importante que se desea mantener en la solución propuesta.

Los subtítulos deben poder personalizarse.

El desarrollador debe ser capaz de cambiar el color o tamaño del texto, la posición en pantalla, utilizar cualquier tipografía para la fuente, cambiar el color de fondo del texto, etc.

Es necesario brindar una solución personalizable para poder adaptarla a cualquier tipo de juego, y también para poder ser adaptable a cualquier tipo de jugador.

Los subtítulos deben poder utilizarse de forma flexible.

La solución propuesta debe ser lo suficientemente flexible como para poder ser utilizada en múltiples contextos. En un videojuego normalmente se utilizan subtítulos para los diálogos de los personajes, pero también existen otros momentos en donde es útil incorporarlos. La herramienta debe ser flexible como para poder ser utilizada en cinemáticas implementadas mediante videos o animaciones in-game. Es importante brindar la flexibilidad para crear subtítulos dinámicos, es decir subtítulos para eventos que ocurren dinámicamente dentro del juego, como por ejemplo el sonido de una puerta abriéndose, o el sonido de un arma recargándose. Este tipo de subtítulos no se pueden crear con anticipación, dado que dependen de cómo se desarrollan los eventos del juego. Por otro lado, como se describió en el Capítulo 4, una característica importante que agrega accesibilidad a los subtítulos es la posibilidad de incluir el nombre del personaje en la línea del subtítulo. La solución para el soporte de subtítulos debe ser lo suficientemente flexible como para permitir incluir el nombre del personaje hablante en la línea de subtítulo.

Importar subtítulos en formato SRT.

Uno de los formatos más populares para almacenar subtítulos es el formato SRT (SRT File Format Website, 2022). La solución propuesta debe brindar soporte para permitir trabajar con este formato. Por ejemplo, deberíamos poder importar a nuestro proyecto subtítulos en este formato y poder reproducirlos dentro del editor y dentro del juego.

Sobre el formato SRT

El formato de archivos SRT es uno de los formatos más comunes utilizados en el proceso de subtitulado. SRT hace referencia a “SubRip Subtitle”, un tipo de archivo que se originó del software SubRip. Este software se creó en el año 2000 para el sistema Windows y tiene la finalidad de extraer subtítulos de películas, videos y DVDs. Hoy en día se volvió el formato más extendido en el uso de subtítulos en reproductores multimedia y software de video.

El archivo contiene una secuencia numerada de subtítulos, indicadores de inicio y fin temporales y el texto del subtítulo. El formato especifica cuatro partes para cada secuencia (The SRT or Subrip subtitle format, 2010):

1. Un contador numérico indicando el número o posición del subtítulo.
2. Tiempo de inicio y fin separados por los caracteres “-->”.
3. Texto del subtítulo en una o más líneas.
4. Una línea en blanco indicando el fin del subtítulo.

El tiempo de inicio y fin deben respetar el formato HH:MM:SS,MIL (horas, minutos, segundos y milisegundos).

Efecto	Etiquetas
Negrita	<code>...</code> o <code>{b}...{/b}</code>
Cursiva	<code><i>...</i></code> o <code>{i}...{/i}</code>
Subrayar	<code><u>...</u></code> o <code>{u}...{/u}</code>
Color del texto	<code>...</code>

Figura 5.9 Tabla de etiquetas permitidas en el formato SRT

El formato SRT además toma prestadas algunas de las etiquetas del lenguaje HTML, lo cual le permite extender todavía más las opciones de personalización y formateo del texto. Las etiquetas se integran directamente en el texto del subtítulo, y de la misma manera que funcionan en el formato HTML, permiten cambiar la forma en la que se ve el texto. En la figura 5.9 se describe la lista de etiquetas permitidas oficialmente en el formato SRT. El color del texto se puede ingresar usando el nombre del color o el valor en hexadecimal usando el formato #RRGGBB (por ejemplo #FF0000 para el color rojo).

En la figura 5.10 se puede observar un ejemplo de archivo de subtítulos, con las cuatro secciones correspondientes al formato de subtítulos SRT.

```

1
00:01:00,400 --> 00:05:15,300
Este es un ejemplo de un subtítulo.

2
00:05:16,400 --> 00:05:20,300
Este es el <u>segundo subtítulo</u>.
Con una segunda línea y un <b>texto en negrita</b>
```

Figura 5.10 Ejemplo de archivo con formato SRT

¿Por qué un plugin?

Los motores de videojuegos están pensados para poder funcionar en una variedad de contextos, diseños y géneros de videojuegos. Pero suele suceder que en algunos casos el

motor que se está utilizando no tiene la solución o herramienta que necesitamos. En esos casos, es necesario alterar el motor original del juego para poder conseguir la solución que el juego necesita. Esto suele suceder en varios escenarios, como por ejemplo porque el motor original no cuenta con la herramienta que necesitamos, o porque el juego no es lo suficientemente performante.

Godot Engine está diseñado de manera que las funcionalidades y herramientas sean fácilmente extensibles. Al ser un proyecto de software libre, siempre se puede cambiar el código fuente original y agregar el soporte que el juego necesita. En particular Godot provee tres maneras diferentes de agregar nueva funcionalidad al motor: mediante un nuevo módulo en el core del motor, utilizando GDNative, o mediante la creación de plugins.

Por lo general, toda la lógica de los juegos creados en Godot se encuentra en archivos de script escritos en el lenguaje GDScript. Este lenguaje está basado en Python, y está específicamente diseñado para integrarse con el motor y facilitar su escritura. Al ser un script, se interpreta en tiempo de ejecución, al contrario de la lógica del motor que ya se encuentra pre-compilada (Documentación de GDScript, 2022).

En la extensión mediante módulos nativos se trata de agregar directamente en el código fuente del motor un nuevo módulo con la nueva lógica (Documentación de módulos, Godot Website, 2022). Por el contrario, GDNative es una tecnología específica de Godot que permite al motor interactuar con librerías compartidas en tiempo de ejecución (Documentación de GDNative, Godot Website, 2022). Se pueden utilizar para ejecutar código nativo sin necesidad de ser encapsulado junto con el ejecutable del motor. La creación de plugins hace referencia a la utilización de GDScript, pero con acceso extra a funciones específicas del editor que permiten customizar o agregar nuevas funcionalidades al mismo (Documentación de plugins, Godot Website, 2022).

La extensión del motor mediante módulos se utiliza en casos donde es necesario que la lógica específica de nuestro juego sea pre-compilada con el resto del motor, en lugar de ser interpretada en tiempo de ejecución, como ocurre normalmente al utilizar GDScript. Las ventajas de utilizar módulos es que proveen una integración más profunda con el motor, dado que se tiene a disposición la API completa de Godot. Adicionalmente no dependen de librerías externas como en GDNative, dado que toda la lógica se encuentra disponible con el motor. Otra ventaja es que la performance puede ser mejor que la de GDNative, si es que el código escrito requiere una comunicación intensa a través de la API de scripting. Por último, un módulo es compatible con todas las plataformas en donde Godot está disponible, al contrario de GDNative en donde se necesita una librería compilada para cada plataforma. Como ventajas de GDNative se encuentran, que al contrario que los módulos nativos, no es necesario compilar el código fuente del motor completo, por lo que es más sencillo de crear y distribuir el trabajo. Adicionalmente GDScript se puede utilizar cuando existe una lógica

específica que necesita de alta performance y no se desea re-compile todo el motor para poder utilizarla. Dado que se basa en librerías externas, las soluciones construidas con GDNative permiten ser distribuidas a través de la librería de assets de Godot, una herramienta integrada en el editor que facilita la descarga de Internet de diferentes tipos de recursos como sprites, sonidos, plugins y hasta proyectos completos.

Un plugin de Godot es una manera de ampliar las herramientas disponibles en el editor. Una facilidad de los plugin es la posibilidad de crearse mediante escenas y archivos GDScript estándar, sin necesidad de recargar el editor. A diferencia de los módulos no es necesario escribir código nativo ni recompilar el motor. Aunque esto puede hacer que agregar plugins sean menos potentes que agregar un módulo o un GDNative, son útiles cuando se requiere agregar funcionalidad nueva al editor. Al igual que una librería GDNative, los plugins se pueden distribuir mediante la librería de assets de Godot.

Para poder brindar soporte de subtítulos al motor, en este trabajo de tesis se optó por una solución basada en plugins. Las razones de su elección son las siguientes: a) se puede compartir mediante la librerías de assets de Godot. Esto es importante para mejorar la visibilidad del proyecto, dado que aparecerá en la lista de plugins disponibles para ser utilizados por los desarrolladores y además el setup o configuración es mínima dado que todo lo que el desarrollador debe hacer es instalarla mediante la librería de assets. b) no resulta una buena idea integrar el soporte de subtítulos como un módulo nativo del motor, dado que esta línea de implementación obligaría, a los desarrolladores que quieran utilizarlos, a usar una versión extraoficial de Godot (alterada para brindar dicho soporte). Se podría pensar en integrar el módulo desarrollado al proyecto oficial, pero como se mencionó anteriormente, algunos colaboradores no ven con buenos ojos la integración de los subtítulos al motor (Marques, 2020), por lo que un “pull request” al repositorio corre el riesgo de no mergearse nunca a la rama master. c) desarrollar un plugin resulta más adecuado que una librería GDNative. Si bien las dos permiten su distribución mediante la librería de assets dentro del editor, una librería GDNative requiere algo más de configuración para poder ser utilizada en un proyecto. Uno de los requerimientos de diseño que se planteó para esta solución es que sea sencilla de instalar y configurar, lo que pone al plugin por delante de GDNative en este aspecto. Una ventaja extra de la implementación mediante un plugin en lugar de utilizar GDNative es que no es necesario compilar el código para poder probar el desarrollo. Adicionalmente, en lo personal, me gustaría que la solución sea sencilla de adaptar a nuevas versiones de Godot, dado que las versiones nuevas del motor se lanzan con bastante regularidad. Un plugin escrito en GDScript solo depende de la API de scripting, y de la sintaxis del lenguaje, por lo que adaptarlo a nuevas versiones es sencillo cuando se documentan los cambios entre versiones.

Desarrollo del plugin “Godot Subtitles Plugin”

El plugin desarrollado en esta tesina se denomina “Godot Subtitles Plugin”, está creado como un plugin de editor, íntegramente desarrollado con GDScript, lo que facilita su extensión a futuro y compatibilidad con nuevas versiones de Godot. No necesita soporte nativo de la plataforma donde se ejecuta el juego, y se puede instalar de forma sencilla en cualquier proyecto. Introduce dos nuevos nodos que brindan soporte para subtítulos en Godot: “SRTImporter” y “LiveCaption”.

Nodo SRTImporter

El nodo “SRTImporter” ofrece soporte para leer archivos de subtítulos con formato SRT y convertirlos a un formato que se pueda utilizar dentro del motor. Este es un nodo utilitario que está pensado para ser usado solo durante el desarrollo del juego, y no durante la ejecución. Permite al desarrollador seleccionar un archivo SRT, parsear el contenido de acuerdo a las especificaciones del formato, y guardarlo como una animación, un recurso que se almacena en disco y que se puede reproducir en tiempo de ejecución mediante la utilización de un AnimationPlayer.

La razón detrás de convertirlo al formato de Animation es que es un formato que el motor entiende y maneja inteligentemente. Godot se encarga de comprimir las animaciones para ahorrar espacio en disco y en memoria, mejorando la performance al momento de reproducirlas. Para el juego es mucho más performante leer un recurso animation creado con anticipación, que leer un archivo SRT, interpretarlo y reproducirlo en tiempo de ejecución.

Para poder crear la animación, la herramienta requiere de un nodo con soporte para mostrar texto formateado. Éste será el nodo en donde aparecerá el subtítulo, de acuerdo a la configuración ingresada por el desarrollador. La animación solo contiene información básica: el texto que se quiere mostrar, alineación del texto y formato básico (negrita, subrayado, cursiva, etc). Esto deja otros atributos importantes bajo el control del desarrollador: la tipografía, colores, tamaños, y espaciado de la fuente se pueden configurar de cualquier manera que el desarrollador desee, para encajar con el diseño del juego o para hacerlo más accesible a los jugadores.

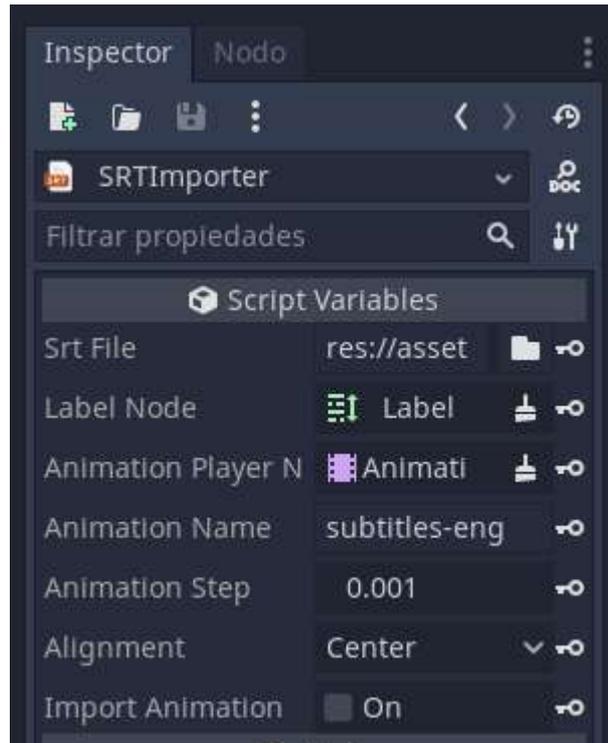


Figura 5.11 Vista de los atributos configurables del nodo “SRTImporter”

En la figura 5.11 se puede observar la vista de inspector del nodo “SRTImporter”. El nodo permite seleccionar el archivo de subtítulos en formato SRT que se desea importar, un nodo en donde queremos mostrar los subtítulos, y el nodo en donde almacenar la animación resultante del proceso de importación. Adicionalmente se permite configurar la resolución de la animación, y aplicar una alineación a los subtítulos (centrado, izquierda, derecha). Al clicar sobre el checkbox “Import Animation”, comienza el procesamiento del archivo y se crea la nueva animación. En la figura 5.12 se puede observar la animación creada en el editor de animaciones. Se crearon las diferentes claves de animación que contienen la información de cada línea de subtítulo: tiempo de inicio y fin, texto y formato.

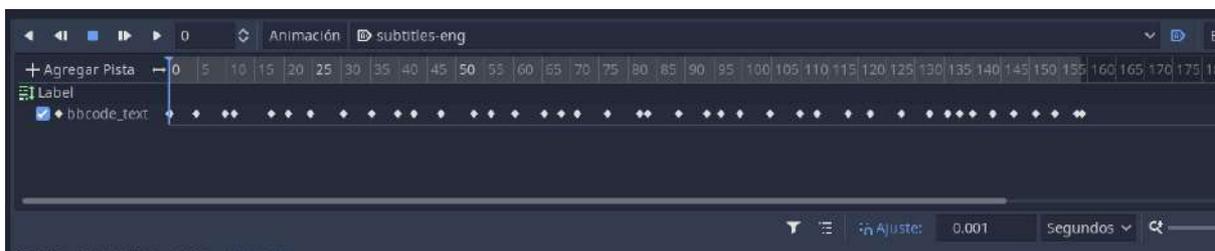


Figura 5.12 Animación resultado del proceso de importación de un archivo SRT.

Uno de los usos más comunes para este nodo es importar subtítulos creados en herramientas externas para trabajarlos en Godot. Se puede utilizar por ejemplo para subtítular el diálogo prediseñado entre personajes, o para mostrar subtítulos de un video o cinemática.

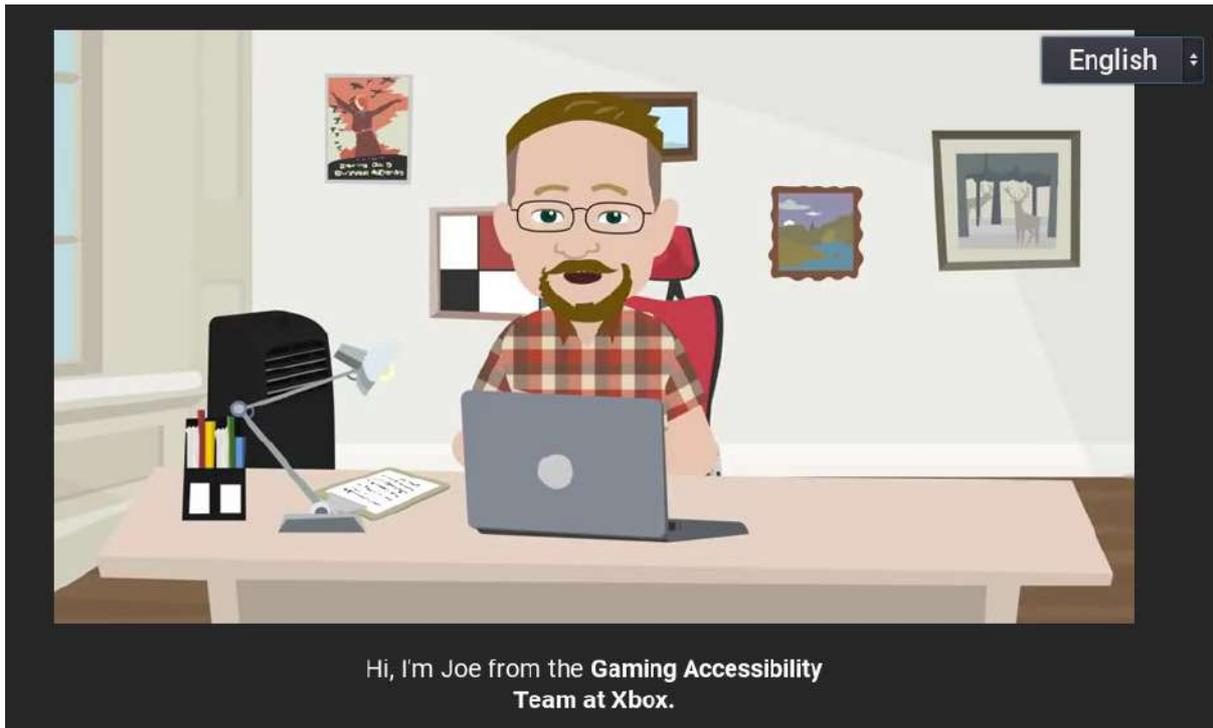


Figura 5.13 Ejemplo de un subtítulo SRT acompañando a un video dentro de Godot.

Nodo LiveCaption

El segundo nodo creado, llamado “LiveCaption”, permite generar subtítulos dinámicamente, y está pensado para usarse en tiempo de ejecución. A diferencia de los subtítulos SRT, que fueron creados con anticipación en una herramienta externa, este tipo de subtítulos o “captions” se van a generar dinámicamente a medida que se generan los eventos del juego. La idea de este nodo es no depender de ninguna información previa, y mostrar subtítulos sólo cuando el desarrollador lo indica.

Este nodo se puede utilizar para mostrar en pantalla el diálogo no guionado de los personajes, o para generar subtítulos descriptivos ante eventos dinámicos dentro del juego. Este plugin soporta la utilización de colores y nombres de personajes. De acuerdo a la guía para subtítulado de la BBC (BBC Subtitle Guidelines, 2022), el método preferido para identificar el personaje que pertenece a cada línea de diálogo es la utilización de colores.

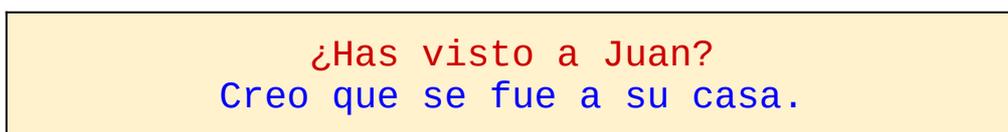


Figura 5.14 Ejemplo de subtítulo de diálogos usando colores para identificar diferentes personajes.

La segunda propuesta de la BBC involucra mostrar una etiqueta con el nombre del personaje antes del diálogo asociado.

MARCOS: ¿Has visto a Juan?
FLORENCIA: Creo que se fue a su casa.

Figura 5.15 Ejemplo de subtítulo usando el nombre del personaje antes del diálogo.

La guía “XBox Accessibility Guidelines” indica que el nombre del personaje solo se debe mostrar la primera vez que habla, y ocultar en las siguientes líneas de diálogo. El plugin desarrollado tiene en cuenta esta sugerencia y muestra el nombre del personaje sólo la primera vez. Adicionalmente el plugin permite configurar la cantidad de tiempo que el subtítulo se mantiene visible en pantalla antes de desaparecer.

En la figura 5.16 se puede observar la configuración para el nodo “LiveCaption”. Se permite configurar la duración de cada línea de diálogo en pantalla y el espacio de tiempo entre subtítulos. Se permite además forzar una alineación para los subtítulos (centrado, izquierda, derecha), cambiar el color por defecto de los subtítulos, y la configuración del efecto “shake”, uno de los efectos que se pueden aplicar a los textos en Godot.

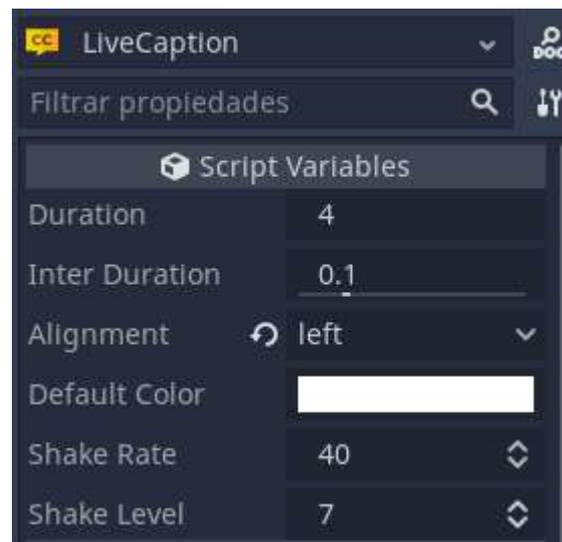


Figura 5.16 Vista de los atributos configurables del nodo “LiveCaption”

Se puede utilizar el nodo “LiveCaption” en múltiples circunstancias:

- Mostrar diálogo utilizando un formato acorde a las guías de accesibilidad.
- Configurar un color para el diálogo de cada personaje.
- Aplicar un “estado” al diálogo, cambiando el color y aplicando efectos sobre el texto.
- Implementar subtítulos descriptivos para algunas fuentes de sonido y sonidos clave.

Para utilizar la herramienta se debe incluir el nodo en cualquier escena. Se puede referenciar este nodo desde un script para acceder y utilizar cualquiera de los métodos que provee el nodo: `caption()`, `dialog()` y `descriptive()`. En la figura 5.17 se pueden observar algunos de los usos del nodo “LiveCaption”.

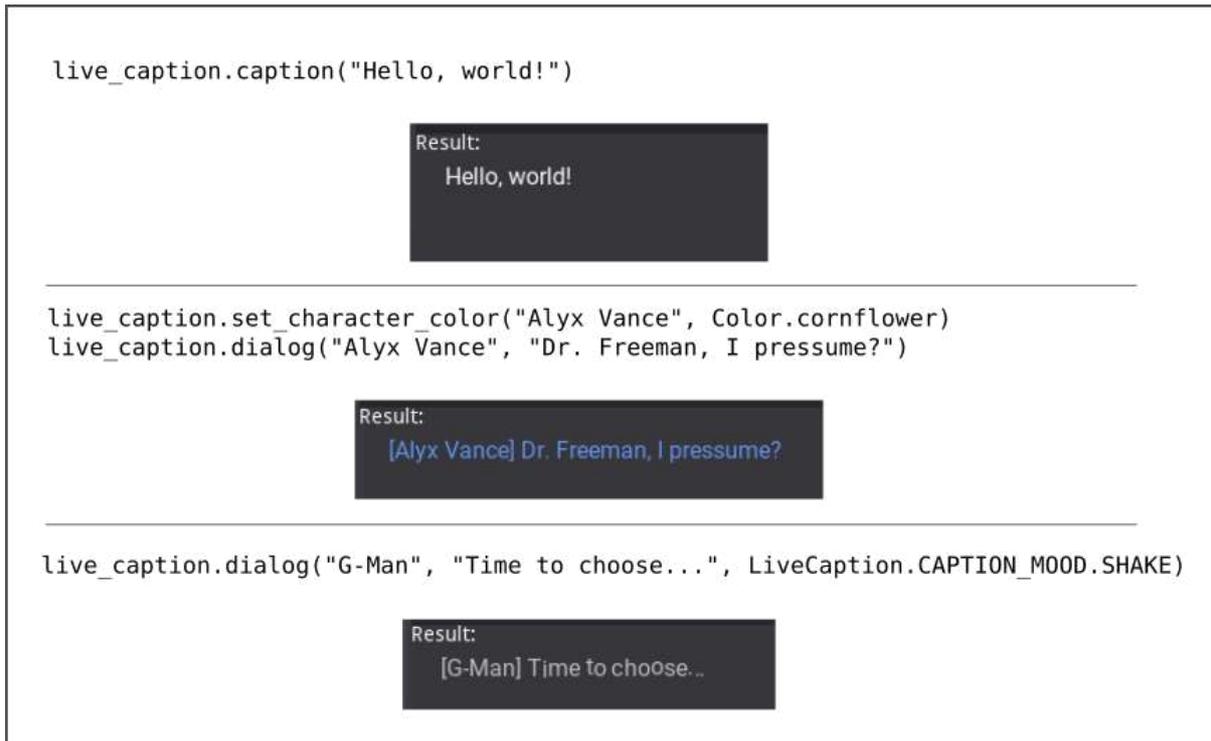


Figura 5.17 Ejemplos de utilización del nodo "LiveCaption" y el resultado mostrado en pantalla

Soporte para BBCode

El BBCode por las siglas en inglés de "Bulletin Board Code" es un lenguaje de marcado simple que fue creado para dar formato a mensajes dentro de foros de Internet. Es similar al código HTML en el sentido que los dos utilizan etiquetas para dar formato. Las etiquetas en BBCode se escriben con palabras claves encerradas entre corchetes. El plugin desarrollado aprovecha las ventajas del marcado BBCode para mostrar texto formateado, más allá de las etiquetas que permite el formato SRT original. Para esto se apoya en el nodo `RichTextLabel` y su propiedad `bbcode_text` para mostrar el texto formateado. El formato SRT original se transformará a un formato compatible con BBCode antes de ser convertido a una animación:

- `` se convierte en `[b]` (negrita).
- `</i>` se convierte en `[i]` (cursiva).
- `</u>` se convierte en `[u]` (subrayado).
- `...` se convierte a `[color=#FF0000]...[/color]`.

El plugin va a ignorar cualquier etiqueta en formato BBCode en el proceso de importación de los archivos SRT. De esta forma se pueden incluir etiquetas adicionales como por ejemplo las etiquetas `[wave]`, `[rainbow]`, `[url]`, o cualquier otra que soporte Godot (BBCode Documentation, Godot Website, 2022).

Publicación del plugin

En el sitio de GitHub (<https://github.com/FEDE0D/godot-subtitles>) se encuentra disponible el proyecto que ejemplifica cómo utilizar el plugin “Godot Subtitles Plugin”. Este proyecto incluye una demostración de cómo subtítular un video usando el plugin para importar un archivo de subtítulos en formato SRT. El proyecto también incluye un generador de diálogos dinámicos (figura 5.18), el cual utiliza el nodo “LiveCaption” para ejemplificar la utilización de colores y nombres de los personajes en los diálogos, además de la generación de subtítulos descriptivos.

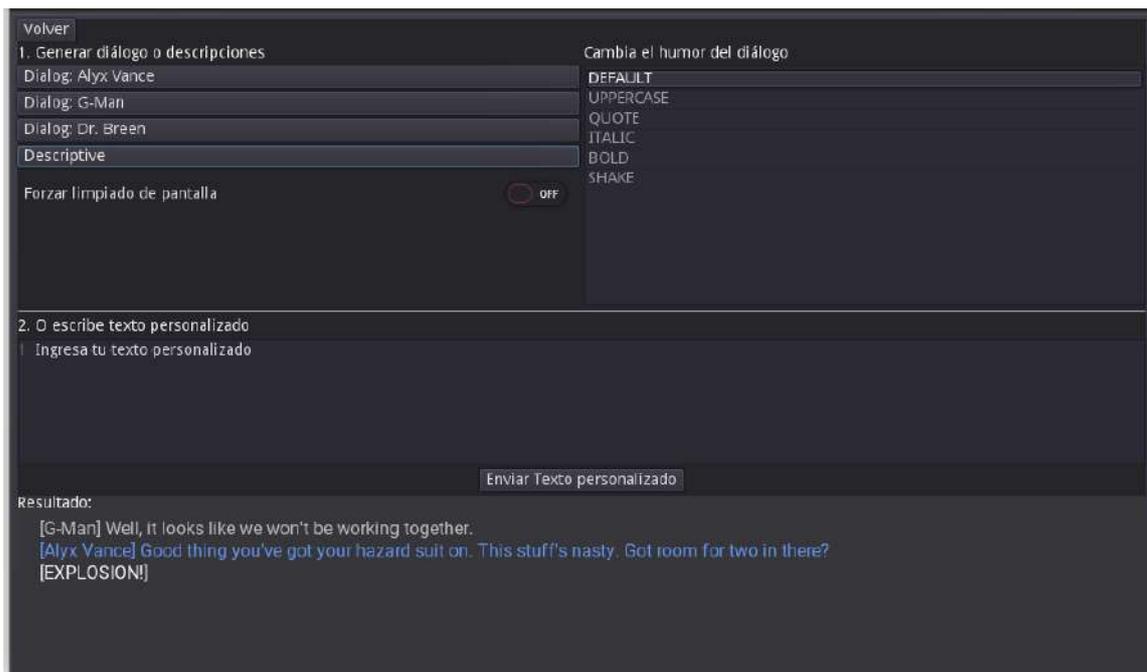


Figura 5.18 Generador de diálogo y subtítulos descriptivos que se incluyen en el proyecto demo, ejemplificando el uso del plugin “Godot Subtitles Plugin”

Adicionalmente, el plugin se aceptó dentro de la librería de assets oficial de Godot Engine (<https://godotengine.org/asset-library/asset/1434>), lo que permite a los desarrolladores de Godot descargar tanto el proyecto de prueba como el plugin con soporte para subtítulos desde dentro del editor.

5.4 Conclusiones

El motor Godot Engine presenta múltiples dificultades para la creación de juegos y aplicaciones accesibles, lo que afecta tanto a los jugadores como a los desarrolladores que usan el motor. Algunas de las herramientas disponibles en el mismo no permiten implementar fácilmente estrategias de accesibilidad, por lo que implementar estas estrategias es responsabilidad completa del desarrollador. La falta de recursos económicos y otras dificultades que suelen acompañar a los desarrollos independientes amenazan la

implementación de tales estrategias, por lo cual poner disponible más y mejores herramientas para trabajar la accesibilidad redonda en una ventaja para los usuarios finales de los productos realizados con Godot Engine. En este sentido el plugin “Godot Subtitles Plugin”, desarrollado en esta tesina, ofrece una solución a los desarrolladores independientes incorporando la facilidad de subtítulos, tanto para leer archivos de subtítulos con formato SRT para luego convertirlos al formato utilizado en Godot, como la de generar subtítulos dinámicamente desde el mismo editor.

En el siguiente capítulo se continuará con el trabajo dentro de Godot Engine, desarrollando un prototipo de juego enfocando brindar opciones de accesibilidad a los jugadores.

Capítulo 6: Prototipo de juego accesible

En los capítulos anteriores se abordó el concepto de accesibilidad, cómo puede afectar a los jugadores y cómo se aborda desde el punto de vista de los desarrolladores de videojuegos independientes. Además, se describió el estado del arte de la accesibilidad del motor para videojuegos Godot Engine, y se desarrolló un nuevo plugin “Godot Subtitles Plugin” para facilitar la utilización de subtítulos dentro de dicho motor.

6.1 Metodología

Como parte de la puesta en práctica de los conceptos trabajados en esta tesina, se propone implementar un breve prototipo de videojuego dentro del motor Godot Engine, con el objetivo de demostrar y aplicar algunas de las decisiones de diseño que se describieron durante el Capítulo 4, y poder ver con ejemplos prácticos cómo estas decisiones influyen en el desarrollo del prototipo. El producto de este capítulo será un prototipo ejecutable para navegadores web y sistemas operativos de escritorio (Linux, Windows), junto con el código fuente y recursos necesarios disponibles en un repositorio público de Github.

La implementación y el diseño de este prototipo se realizó y estuvo crucialmente influenciado por la investigación realizada durante el desarrollo de esta tesina. A continuación se realizó una serie de playtestings³³ con diversos participantes, con el objetivo de probar, mejorar y justificar las decisiones de diseño tomadas durante el desarrollo. El resto de este capítulo proporciona información acerca del desarrollo del prototipo y de las sesiones de playtesting realizadas. Este prototipo implementará algunas de las opciones de accesibilidad que se estudiaron durante el Capítulo 4, teniendo en cuenta los puntos de vista y dificultades expresadas por desarrolladores independientes en el Capítulo 3 y las necesidades y diversidad de los jugadores vistas en el Capítulo 2.

6.2 Sobre el proceso iterativo del desarrollo del prototipo

El foco del prototipo estuvo puesto en el diseño y rediseño del prototipo. El flujo está influenciado por un flujo de cinco pasos basado en una metodología de diseño conocida como Design Thinking (Plattner et al, 2014), pero adaptado para poder ajustarse a los objetivos de esta tesina. Este flujo no representa la realidad del desarrollo de videojuegos de ninguna manera, es meramente una manera de poder visualizar el proceso de desarrollo del prototipo y cómo afecta el diseño inclusivo. En la figura 6.1 se observa el flujo final tomado para el desarrollo.

³³ Serie de pruebas realizadas para identificar cómo un participante interactúa con el juego.

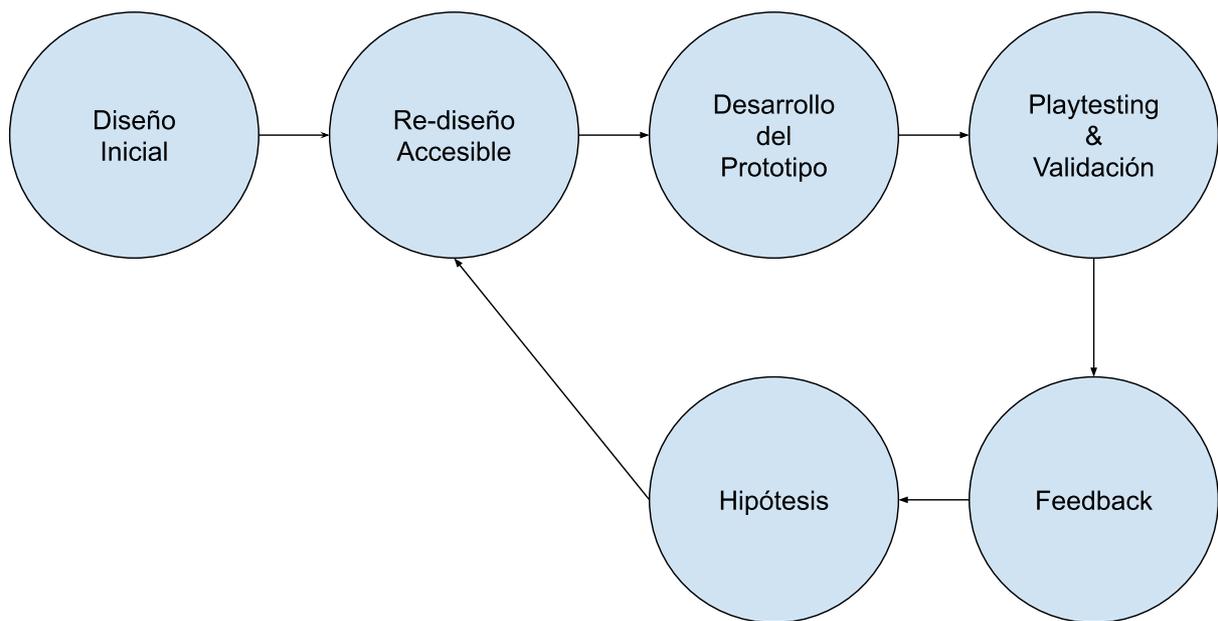


Figura 6.1 Mapa de flujo del proceso del diseño de juego utilizado para crear el prototipo

Este flujo cuenta con seis pasos definidos:

1. Diseño inicial.
2. Rediseño accesible.
3. Desarrollo del prototipo.
4. Playtesting y Validación.
5. Feedback.
6. Hipótesis.

Decidí separar la etapa de diseño en dos etapas independientes, para reflejar una situación que podría darse en el desarrollo de videojuegos independientes. Por una parte tenemos el diseño inicial del juego, con sus componentes puros tal cual fueron pensados y diseñados en un principio. En esta etapa se diseñan y planean todos los aspectos del juego: la historia, los personajes, los escenarios, el gameplay, los controles, el arte conceptual, etc.

Como segunda etapa tenemos el rediseño del juego, pero esta vez tomando como foco la accesibilidad. En esta etapa se repasan las ideas y conceptos diseñados durante la primera etapa, se definen objetivos de accesibilidad que se desean alcanzar, y se plantean mejoras y cambios en el diseño original que permitan alcanzar dichos objetivos.

El resultado de las dos primeras etapas es un documento de diseño de juego o GDD, por las siglas en inglés de Game Design Document. Este documento contiene todos los diseños, ideas, requerimientos técnicos, detalles de jugabilidad, y toda información que sirva para desarrollar el prototipo a continuación.

En la tercera etapa tenemos el desarrollo del prototipo en sí. En esta etapa se toman todos los bocetos, diseños, requerimientos y recursos generados en la etapa anterior y se

desarrolla el prototipo. Se desarrollan todos los elementos de accesibilidad previamente pensados para el prototipo.

Una vez que el prototipo está desarrollado de acuerdo a las especificaciones planteadas, se construye un ejecutable del mismo y se lleva a cabo el playtesting. Esta etapa consiste de una serie de sesiones de prueba en las cuales diversos jugadores prueban el prototipo en un ambiente controlado. Se realizan observaciones objetivas de la interacción del jugador con el prototipo y se toman notas. La etapa de playtesting nos permite realizar una validación de los objetivos de accesibilidad planteados durante la segunda etapa, para comprobar en qué nivel fueron cumplidos.

Posteriormente se toma un feedback de los jugadores. Este feedback brinda al jugador un momento para poder expresar sus pensamientos y opiniones acerca del juego que probaron. Este feedback es una de las piezas de información más importantes que se rescatan de la etapa de playtesting, dado que permite poner en palabras las fallas en accesibilidad y dificultades que experimentan los jugadores.

En la última etapa, partimos de los datos obtenidos durante el playtesting, identificamos las dificultades y fallos de accesibilidad de nuestro prototipo y realizamos varias hipótesis acerca de las causas que llevaron al prototipo a no alcanzar los objetivos de accesibilidad planteados originalmente. Con estas hipótesis se vuelve a la etapa de rediseño y se llevan a cabo modificaciones sobre el diseño y requerimientos del prototipo. Estas mejoras del diseño luego se vuelven a desarrollar y a probar con jugadores en una nueva etapa de playtesting. Este ciclo se vuelve a ejecutar cuantas veces hagan falta para alcanzar los objetivos de accesibilidad requeridos.

6.3 Sobre el diseño de juego

El género de videojuegos elegido para el prototipo es el género de aventura y plataformas en 2D. El juego cuenta con elementos de exploración, puzzles y combate. El jugador controla a Mahur, un misterioso personaje con poderes mágicos, quién se adentra en un bosque para tratar de librarlo del mal mientras resuelve puzzles para poder avanzar. El juego es un sidescroller³⁴ de plataformas clásico, orientado a la exploración de los niveles, impulsada por el diálogo con diferentes personajes que se presentan durante la aventura.

El documento anexo “Documento de Diseño de Juego” contiene todos los detalles sobre el diseño del prototipo y sobre los requerimientos técnicos y funcionales. El prototipo se centra en el desarrollo del primer nivel de la aventura, aprovechando la oportunidad para realizar

³⁴ También conocido como videojuego de desplazamiento lateral, es aquel en el que el movimiento del jugador se permite de una forma horizontal, a través de niveles que se exploran bidimensionalmente.

un tutorial que permita al jugador familiarizarse con los controles y el modo de juego propuesto.

El prototipo se desarrolló utilizando el motor Godot Engine, está disponible tanto para navegadores web como para sistemas operativos de escritorio (Windows y Linux). Adicionalmente se brinda soporte para controlar el juego mediante dos tipos de controladores: el teclado y joystick.

Como requerimientos no funcionales, el prototipo implementa de manera exitosa opciones de accesibilidad que sirvan para ejemplificar las ideas y temáticas examinadas durante este tesina, y debe ser posible realizarlo en un período corto de tiempo.

6.4 Sobre la implementación del prototipo

Arquitectura

El prototipo fue desarrollado de acuerdo a las especificaciones del documento de diseño dentro del motor Godot Engine. La arquitectura utilizada para la implementación consta de un objeto único en memoria (Singleton) que guarda información sobre el estado general del juego. Este objeto se utiliza como punto de conexión entre todas las entidades del juego, de manera que por ejemplo la entidad Jugador, por ejemplo, sabe en qué estado se encuentra el juego (si está pausado, si se está mostrando un subtítulo, etc).

En Godot Engine, así como en otros motores de desarrollo de videojuegos, existe un mecanismo que permite la comunicación entre grandes cantidades de entidades dentro del juego sin la necesidad de mantener referencias entre las mismas. Este mecanismo se utiliza como base para la comunicación entre entidades del juego, y termina moldeando la arquitectura del software. Las señales en Godot son una herramienta que permite a un nodo enviar mensajes a otro nodo sin necesidad de mantener referencias explícitas entre los mismos. Esto significa que un nodo podría emitir un mensaje sin preocuparse por quién lo va a recibir ni entender cómo o cuándo lo va a procesar, lo que brinda una gran escalabilidad y encapsulamiento en el desarrollo de las entidades. Las señales permiten definir un canal de comunicación asíncrono y asimétrico. Asíncrono porque un nodo puede emitir un mensaje y continuar su ejecución sin esperar una respuesta, y asimétrico porque un nodo puede recibir/emitir mensajes desde/hacia uno o más nodos.

Dentro de la implementación del prototipo existen múltiples entidades que necesitan enviar y recibir mensajes para que el juego funcione correctamente. Un ejemplo de esto lo podemos ver con las escenas "World" que representa a un nivel, "Player" que representa al personaje, y los elementos de la UI, por ejemplo los subtítulos. Cuando la entidad "Player" detecta que está en presencia de una entidad dentro de "World" con la cual puede interactuar, emite una señal a través de GameState. Esta señal luego es escuchada por el

nodo de la UI que muestra subtítulos, efectivamente iniciando el diálogo y mostrando subtítulos en pantalla. La figura 6.2 muestra un diagrama de una pequeña sección de la arquitectura de entidades dentro del prototipo que ejemplifica la utilización de señales.

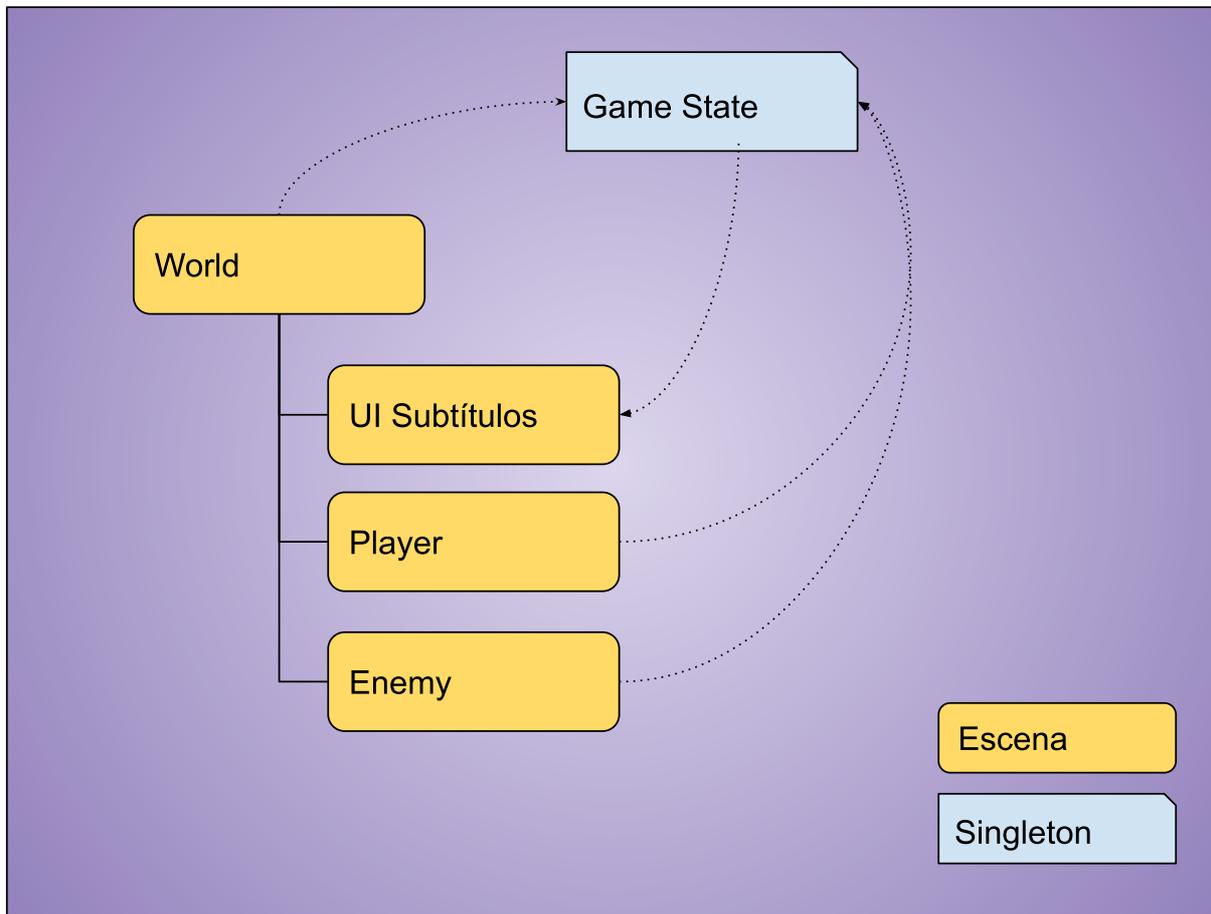


Figura 6.2 Ejemplo de utilización de señales para comunicar mensajes entre escenas

Escenas principales

A continuación se describe brevemente el funcionamiento de las escenas más importantes del prototipo.

Escena World

Esta es la escena principal, es la que se ejecuta durante el gameplay y representa un nivel jugable. Incorpora múltiples escenas bajo su jerarquía para crear un nivel completo:

- Fondo con efecto parallax³⁵.
- Cuerpos físicos que conforman el suelo y las paredes del mundo virtual.
- Nodos de UI como la pantalla de pausa y los subtítulos.
- Personajes:

³⁵ Es un efecto utilizado para crear la ilusión de profundidad usando capas en 2D que se mueven a diferentes velocidades.

- Controlado por el jugador.
- Enemigos controlados por el juego.
- Elementos visuales que sirven como decoración para el nivel.

En la figura 6.3 se puede observar una captura de pantalla del editor mientras se está editando la escena “World”: a la izquierda se visualiza la jerarquía de nodos que conforman la escena completa.

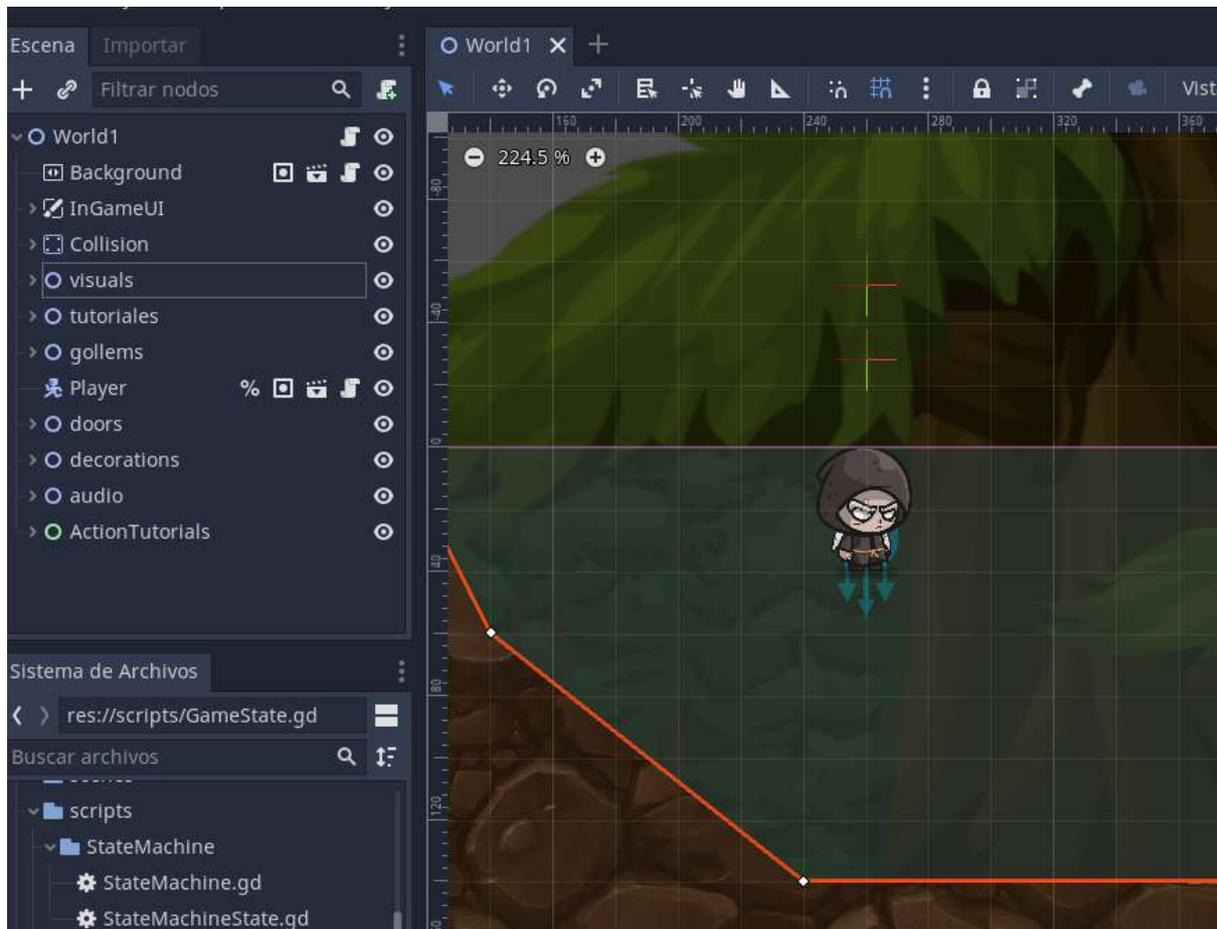


Figura 6.3 Captura del editor durante la edición de la escena “World”

Escenas Player y Enemy

La implementación de estas escenas son similares dado que ambas representan personajes dentro del juego. La única diferencia radica en que Player es un personaje controlado por el jugador, y Enemy es controlada por el juego. Para implementar un personaje se utilizaron combinaciones de nodos, cada una con su función específica:

- Elementos físico, que permiten introducir al personaje a una simulación de físicas y ejecutar colisiones con las paredes y el suelo del mundo virtual.
- Elementos visuales, conformados por Sprites³⁶ y animaciones.

³⁶ Imágenes optimizadas para ser utilizadas dentro un juego

- Una cámara virtual, que permite seguir al personaje mientras se mueve por la pantalla.
- Una máquina de estados, parecido a la implementación del patrón de diseño State, lo que permite que tanto el jugador como los enemigos se comporten de manera diferente de acuerdo a su estado actual.
- Elementos auditivos, por ejemplo para reproducir los efectos de sonido que se producen con cada salto del personaje

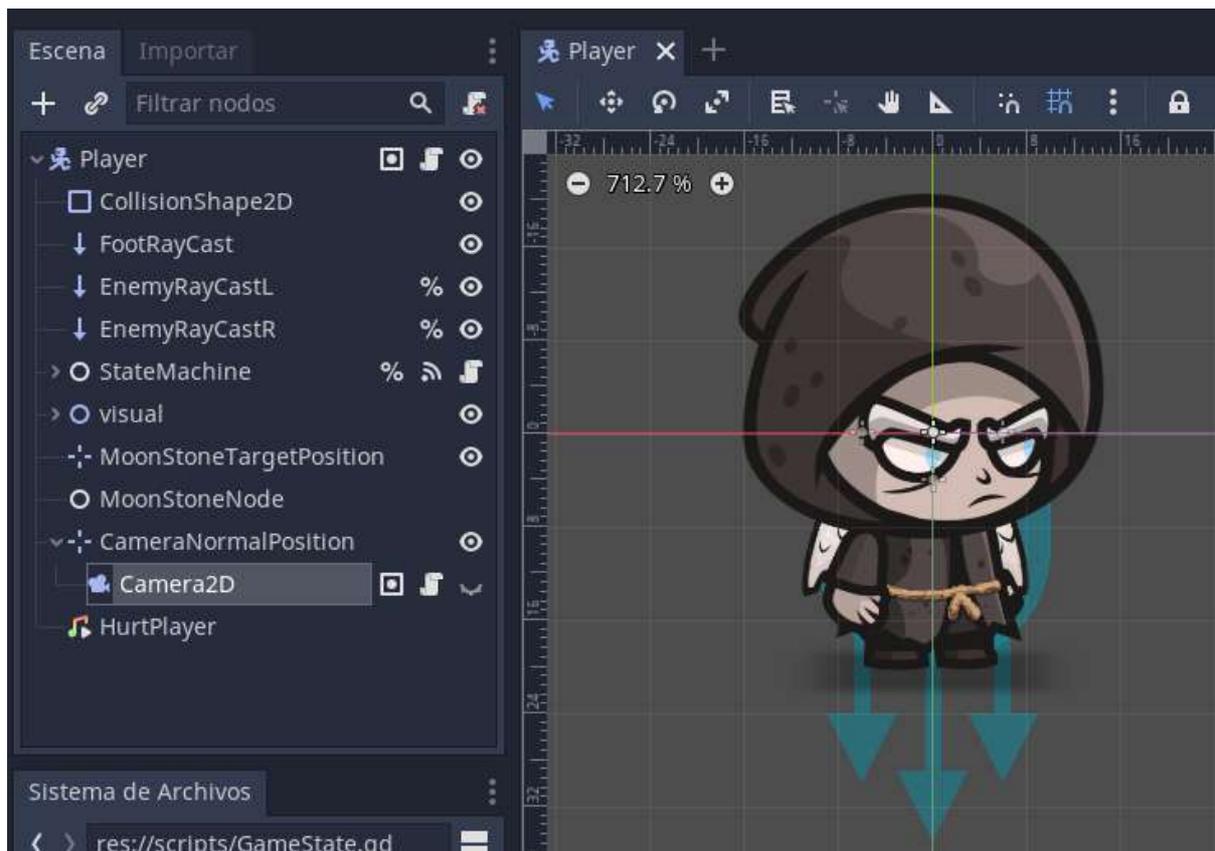


Figura 6.4 Captura del editor mientras se edita la escena "Player"

Interfaz de usuario

El prototipo contiene pocas interfaces de usuario y menús navegables. Sin embargo fueron implementadas con el mismo detalle que el juego principal. Por ejemplo en la pantalla principal se encuentra el menú de inicio, con opciones para iniciar una nueva partida, ir a la pantalla de configuración, o salir del juego. En la figura 6.5 se puede observar una captura de este menú y su versión final. Tanto el menú de inicio como el resto de las interfaces dentro del juego fueron diseñadas para poder ser navegables utilizando el mouse, teclado o joystick.

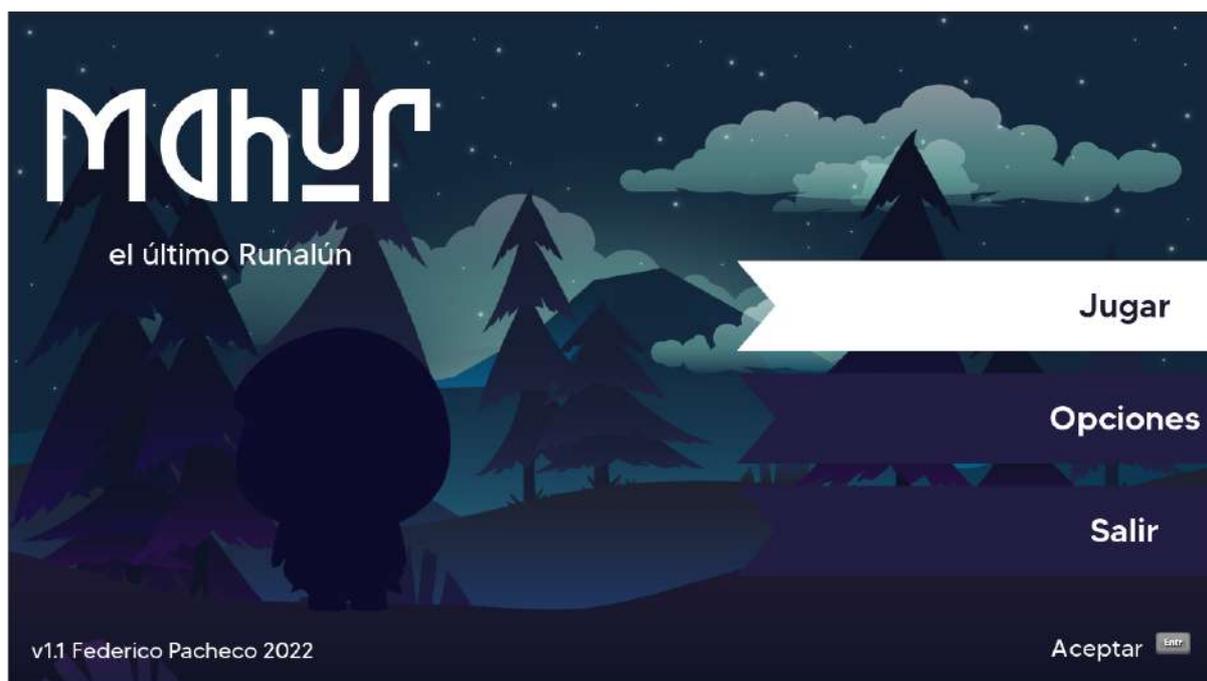


Figura 6.5 Captura de pantalla del menú principal del juego

La pantalla de configuración fue diseñada para poder ver todas las opciones desde una misma ventana, sin necesidad de entrar en submenús o navegar a otras pantallas. Para esto se organizaron las diferentes configuraciones disponibles en cuatro categorías: gráficos, controles, audio, juego. La figura 6.6 muestra una captura de esta pantalla en la versión final del prototipo.

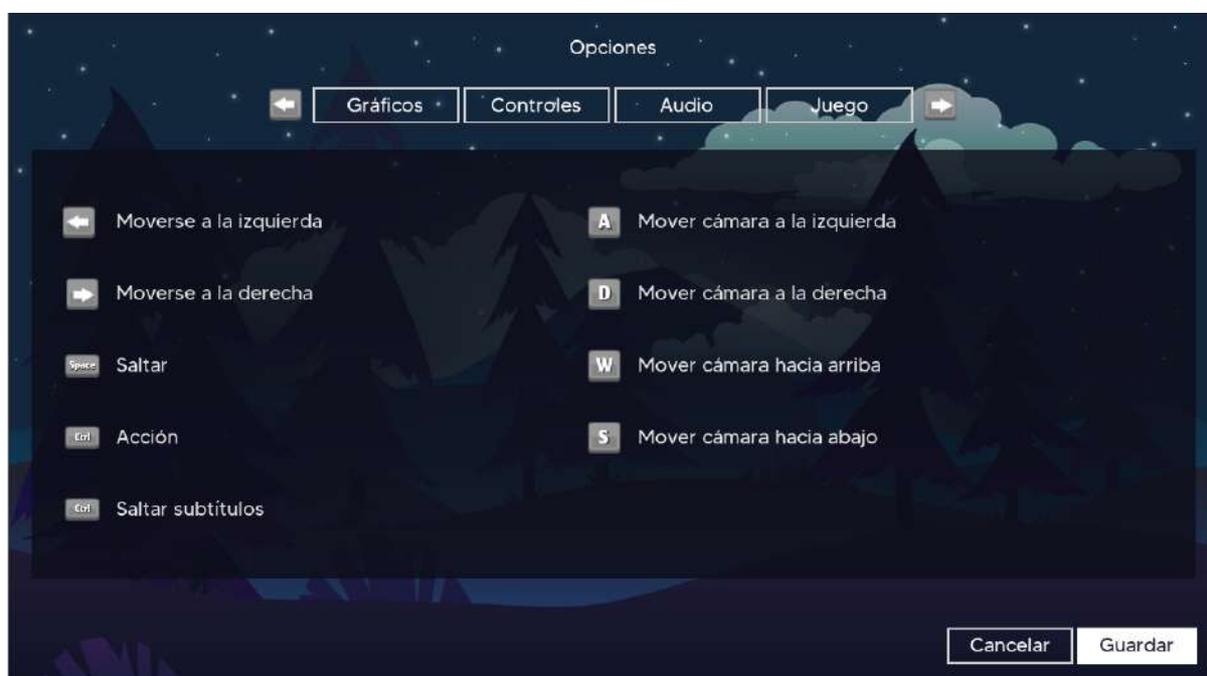


Figura 6.6 Captura de la pantalla de configuración del prototipo

El menú de pausa se presenta cuando el jugador presiona la tecla “escape” o botón “cancelar” del joystick. Esta pantalla parece simple a la vista, pero es una de las más

complejas en su implementación, dado que ocurren varias cosas bajo la alfombra. La pantalla de pausa se superpone sobre la pantalla de juego, por lo que las animaciones, subtítulos, movimiento de personajes y toda otra acción del juego debe ser detenida por un tiempo. Además de esto, en la esquina superior derecha de la pantalla se muestra la lista de objetivos. Esta lista se actualiza automáticamente a medida que el jugador avanza por el nivel. Por último cuando el jugador elige la opción “Salir”, se muestra un diálogo de confirmación que se superpone al resto de los elementos visuales anteriormente mencionados. La figura 6.7 muestra una captura de pantalla del menú de pausa.



Figura 6.7 Captura de pantalla del menú de pausa

Funciones de accesibilidad

El objetivo principal de este prototipo es el de ejemplificar la implementación de múltiples herramientas de accesibilidad para el mismo. Si bien estas opciones se centraron en asistir a jugadores con discapacidades auditivas, también se desarrollaron funciones que aplican a otras categorías de accesibilidad.

La lista completa de funciones relacionadas a la accesibilidad desarrolladas para este prototipo incluye:

- Posibilidad de cambiar el modo de juego entre pantalla completa y modo ventana.
- Posibilidad de ocultar el fondo de juego.
- Escalado de elementos de interfaz de usuario.
- Remapeo de controles.
- Ajustar el volumen de la música y efectos de sonido.
- Modo de audio mono-canal.

- Subtítulos descriptivos.
- Soporte Text-To-Speech.
- Asistencia: tutorial de controles.
- Asistencia: mostrar lista de objetivos.

Ocultar fondo

Esta función de accesibilidad tiene como objetivo simplificar el aspecto visual del juego, ofreciendo al jugador la posibilidad de ocultar el efecto parallax en el fondo del nivel. Este efecto puede causar una sobreestimulación visual y causar efectos negativos en el jugador. La figura 6.8 muestra una captura de pantalla del juego con esta opción activada.



Figura 6.8 Captura de pantalla del juego con la opción "Ocultar fondo" activa

Escalar pantalla

Esta opción está orientada a asistir a jugadores con dificultades visuales. Como se describió en el Capítulo 4, algunos jugadores pueden tener dificultades para interpretar los elementos visuales cuando el tamaño en pantalla no es el correcto. Esto puede darse por ejemplo en jugadores con baja visión, o en situaciones donde el tamaño de la pantalla es pequeño, o donde la pantalla se encuentra muy alejada del jugador. La función de escalar pantalla permite al jugador reducir o agrandar el tamaño de los elementos visuales. La figura 6.9 muestra una comparación de cómo se ven diferentes escalado de pantalla sobre el componente de subtítulos.

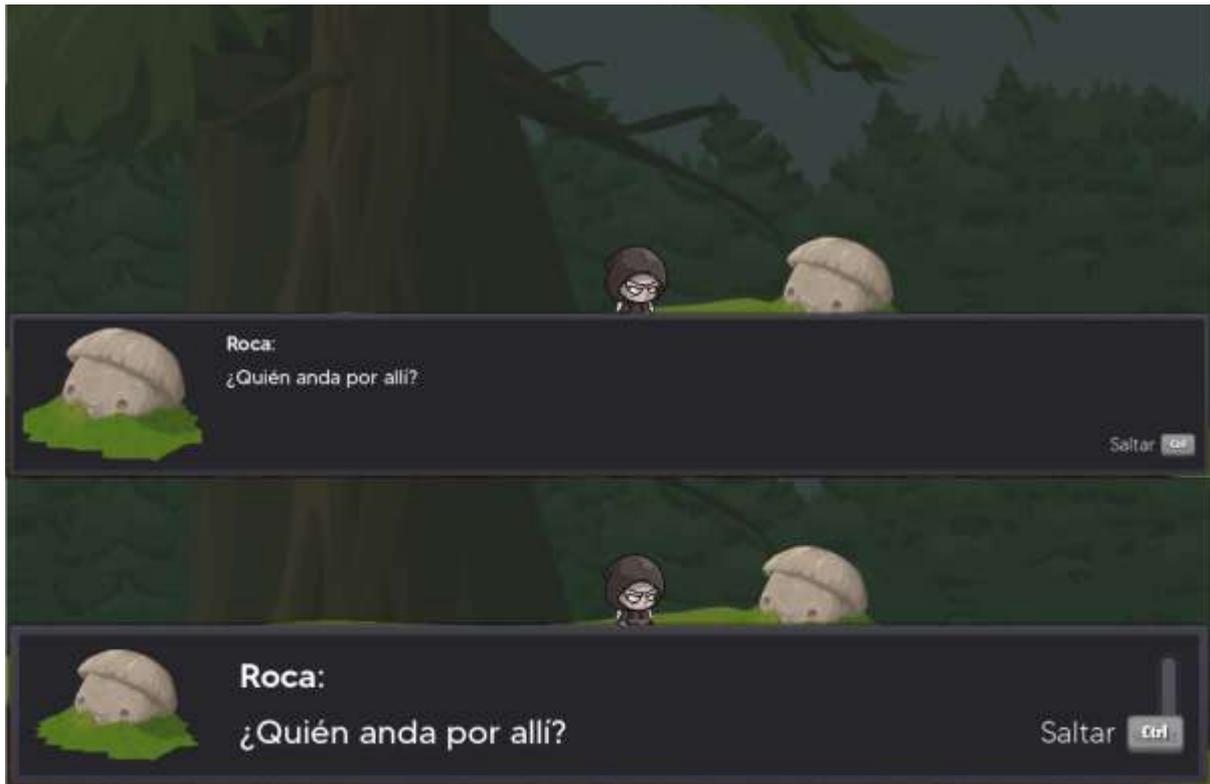


Figura 6.9 Comparación de diferentes niveles de escala de pantalla sobre el componente de subtítulos

Remapeo de controles

El remapeo de controles permite a los jugadores asignar cualquier tecla o botón del joystick para realizar una acción determinada. Esto se realizó utilizando el plugin “Input Button Plugin” de “Haunted Bees Productions”, un plugin que permite mostrar en pantalla una representación visual de los botones y teclas asignadas a acciones, y además detectar el cambio de tipo de controlador, mostrando íconos de teclas o botones de acuerdo al tipo de controlador detectado dinámicamente.

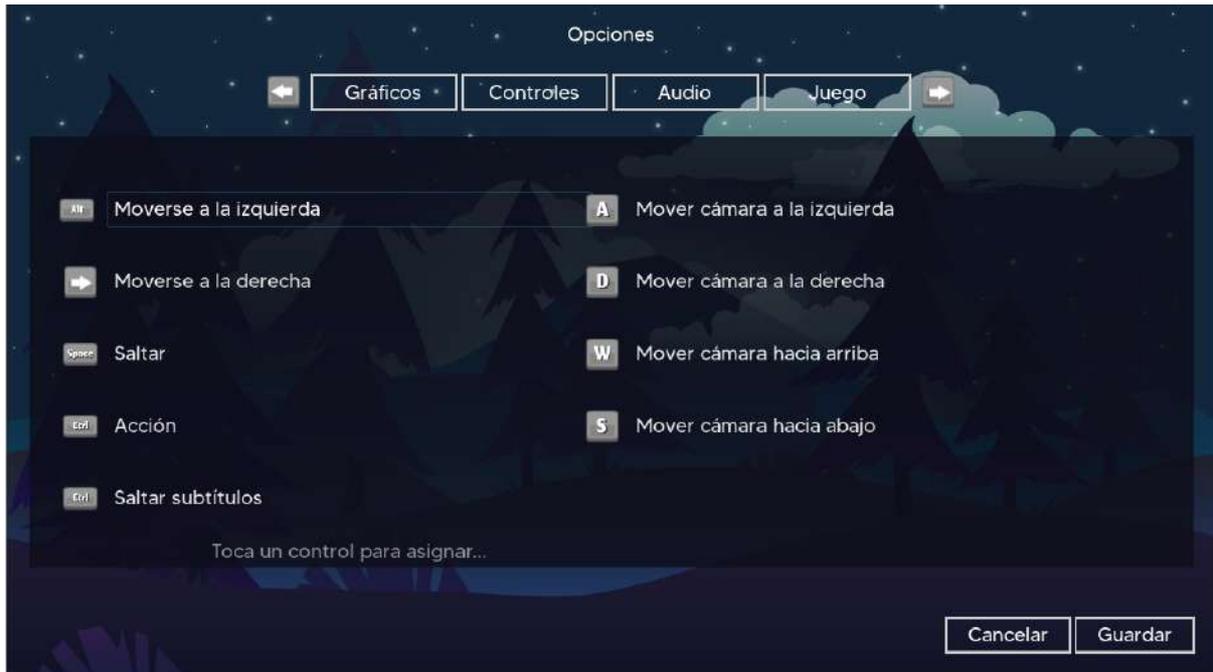


Figura 6.10 Captura de la pantalla de configuración de controles

Ajustes de audio

Las opciones de accesibilidad relacionadas al audio fueron el punto en el que más se trabajó dentro de las funciones de accesibilidad del prototipo. La figura 6.11 muestra una captura de la pantalla de configuración de audio, en donde se ven las opciones disponibles.

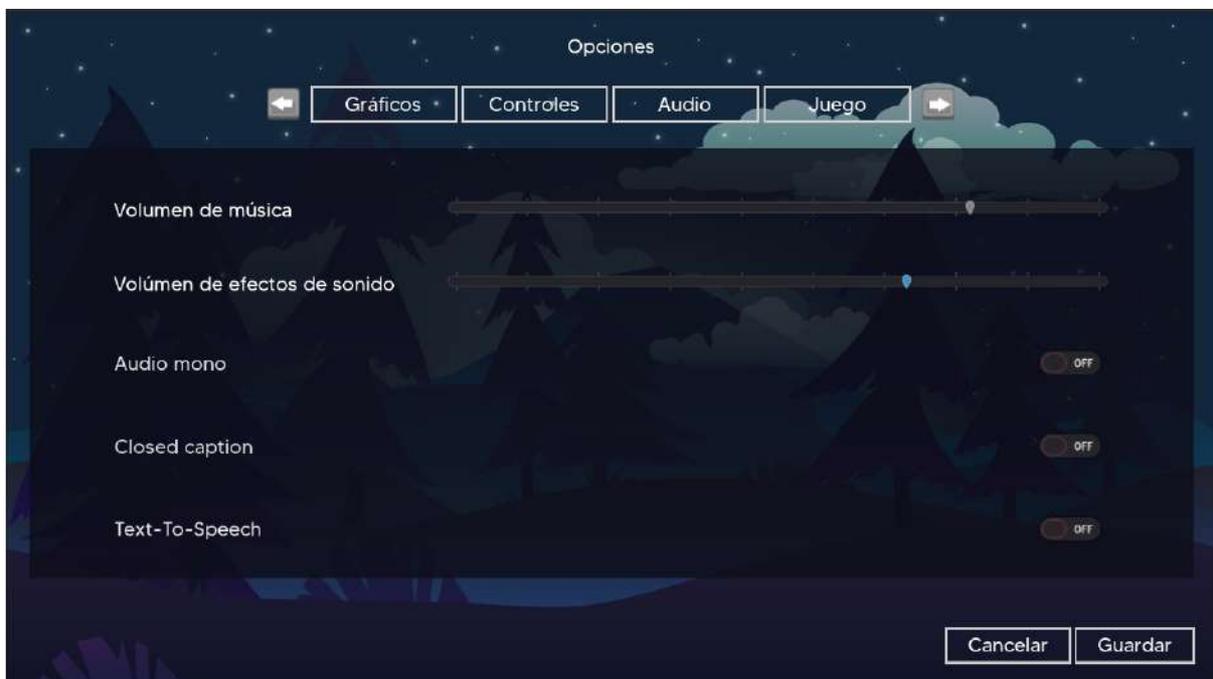


Figura 6.11 Captura de la pantalla de configuración de las opciones de audio

Se permite ajustar tanto el volumen de la música como el de los efectos de sonido de forma separada. Esto permite por ejemplo que el jugador apague la música y escuche solo los

efectos de sonido. En la pantalla de configuración se muestra una preview a medida que se ajustan los valores del volumen. Para implementar esto se utilizó la herramienta de buses de audio disponibles en el motor Godot Engine. Se asignaron las fuentes de sonido a diferentes buses de audio (figura 6.12), y de esta manera se ajustó el nivel de cada bus por separado para obtener el efecto deseado.

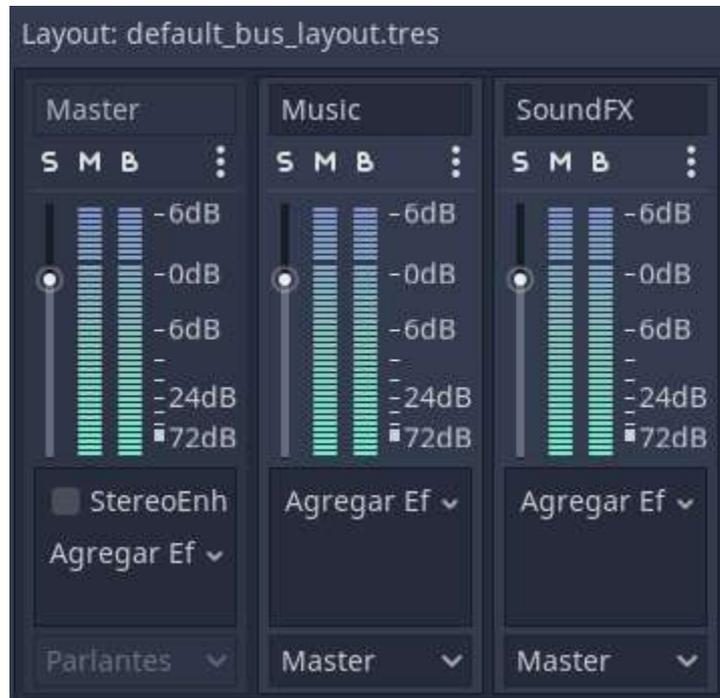


Figura 6.12 Captura de los tres buses de audio utilizados en el prototipo: efectos de sonido, música y un bus master

Para realizar el efecto de audio monocanal, se utilizó un efecto aplicado al bus de audio master (el principal). La figura 6.13 muestra la configuración del efecto de sonido “StereoEnhance”, que permite obtener un efecto de audio monocanal en el bus master.



Figura 6.13 Captura de pantalla de la configuración del efecto de sonido “StereoEnhance”

La función de subtítulos descriptivos utiliza el nodo “LiveCaption” del plugin desarrollado para esta tesina. La implementación consiste en detectar, cuando se produce un efecto de

sonido, y enviar una señal al componente “LiveCaption”, enviando un mensaje que contiene el texto correspondiente al subtítulo descriptivo correspondiente. La figura 6.14 muestra una captura del juego en donde se activaron los subtítulos descriptivos, indicando que se escucha el sonido de fuego en las cercanías.



Figura 6.14 Captura de pantalla que ejemplifica el uso de subtítulos descriptivos en el juego

Por último, la implementación del soporte Text-To-Speech fue posible gracias al plugin “Godot Accessibility” de Nolan Darilek, el cual contiene un utilitario para comunicarse con las capacidades TTS de la plataforma donde se ejecuta el juego. En este prototipo, cada línea de diálogo que se muestra en el componente de subtítulos también se envía mediante señales al componente TTS del plugin, consiguiendo que se escuchen las líneas de diálogo a través del plugin. Además de esto, el plugin se utilizó para leer en voz alta el nombre del componente de UI seleccionado, y de este modo se facilita la navegación por la pantalla a través del audio.

Asistencias dentro del juego

Las últimas funcionalidades de accesibilidad agregadas al prototipo están relacionadas a brindar cierta asistencia al jugador mientras está jugando. En el Capítulo 4 se menciona brevemente que las asistencias in-game mejoran la experiencia para algunos jugadores. En este prototipo se incluyen dos tipos de asistencia: tutoriales y objetivos. Los tutoriales están integrados dentro del nivel, y permiten al jugador aprender sobre los controles y las acciones disponibles dentro del juego de una manera más natural. El listado de objetivos surgieron de feedbacks por parte de los jugadores en las sesiones de playtesting, y permiten al jugador tener una ayuda extra y un objetivo claro a realizar para evitar que se sientan “sin rumbo” o perdidos dentro del juego. La figura 6.15 muestra el primer tutorial que

aparece en el juego, un indicador que enseña al jugador a saltar presionando la barra espaciadora del teclado.



Figura 6.15 Captura de pantalla del primer tutorial de controles que se ven en el juego

La figura 6.16 muestra el momento en el que aparece la lista de objetivos en la esquina superior derecha de la pantalla, en este caso indicando al jugador que debe “encontrar una entrada al interior del templo”.

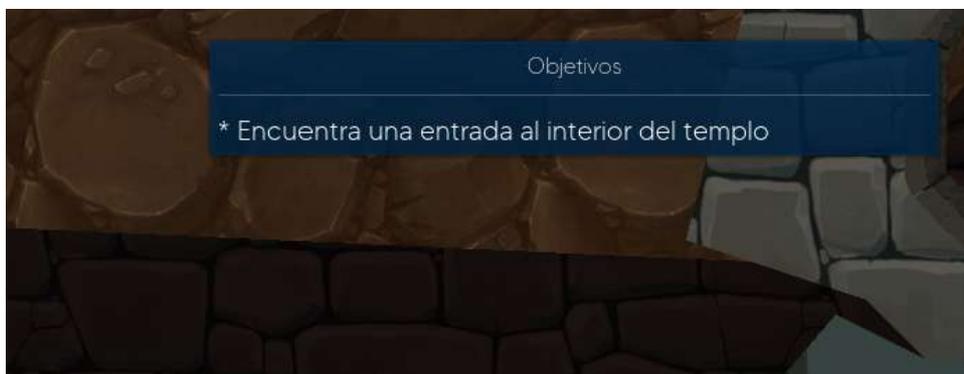


Figura 6.16 Ejemplo del listado de objetivos dentro del juego

Diálogos y subtítulos

Uno de los aspectos principales del diseño del juego es la transmisión de la historia a través del diálogo entre personajes. Para esto fue necesario crear un elemento visual que soporte tanto la reproducción de diálogos predefinidos, como subtítulos generados dinámicamente (para el soporte de subtítulos descriptivos). El componente “SubtitlesUI” fue creado con estas necesidades en mente. Consiste de un contenedor alineado a la parte inferior de la pantalla, y una imagen que representa al personaje correspondiente a las líneas de diálogo mostradas. Para implementar esto se utilizó el plugin “Godot Subtitles Plugin” creado para

esta tesina, permitiendo mostrar líneas de subtítulos dentro de este componente. La figura 6.17 muestra una captura de pantalla del editor mientras se está editando el componente “SubtitlesUI”, donde se observa la utilización del nodo “LiveCaption”.

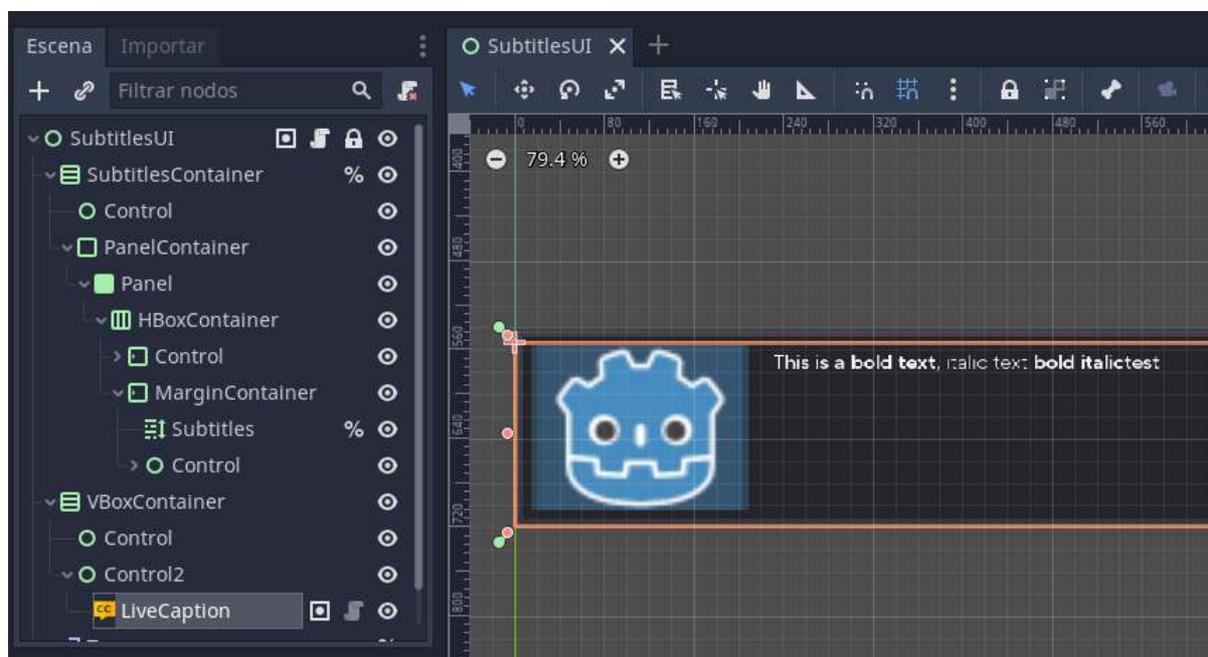


Figura 6.17 Captura de pantalla de la edición del componente “SubtitlesUI”, el cual permite mostrar diálogos y subtítulos descriptivos durante la partida

La tipografía final del prototipo es una tipografía Sans-Serif y puede ser leída con facilidad tanto en tamaños pequeños como grandes. El contraste del texto es el adecuado, tanto para los diálogos como para los subtítulos descriptivos, por lo que son sencillos de leer en cualquier contexto dentro del juego.

6.5 Sobre el proceso de testing

Como se explicó anteriormente, una de las etapas más importantes del proceso de desarrollo del prototipo es la etapa de playtesting. Esta etapa nos permite descubrir errores y fallas tanto en el diseño del juego en sí como en la implementación de opciones de accesibilidad. El objetivo del proceso de testing es descubrir cómo responden los participantes al prototipo en su estado actual de desarrollo y de esta manera encontrar múltiples puntos deficientes en el prototipo, por ejemplo:

- Fallas en diseños de niveles.
- Fallas en la implementación de mecánicas de juego.
- Expectativas que no se cumplen, por ejemplo cuando una mecánica no es tan divertida como se planteaba en el diseño original.
- Información que el juego presenta al participante de una forma poco clara, por ejemplo cuáles son los objetivos, o cuáles son los controles.

- Dificultad del participante para navegar el nivel.
- Dificultades para realizar acciones particulares, por ejemplo abrir una puerta.
- Dificultad para navegar el menú.
- Errores en la implementación de herramientas de accesibilidad.

De acuerdo al estudio “Playtesting for indie studios” (Mirza-Babaei et al, 2016), es conveniente planificar el proceso de testing y ajustarlo de acuerdo a las necesidades del estudio independiente. Para esto se tienen en cuenta ciertos recursos como dinero, tiempo disponible, disponibilidad de espacio y herramientas o equipo necesario para realizar las sesiones de prueba, etc. La metodología elegida para las pruebas debe contemplar las necesidades y deseos particulares de cada estudio independiente y también de cada juego en particular. Es por eso que se decidió ajustar la selección de participantes, la metodología utilizada durante las sesiones de playtesting, y la recolección del feedback a los modos y tiempos que permitan ser llevados a término correctamente dentro del marco de esta tesina. A continuación se describen las limitaciones y metodologías usadas para cada aspecto del proceso de testing.

Participantes

La selección de jugadores, referidos en este capítulo como “participantes” del proceso de testing, se enfocó en contar con diversidad de géneros, edades y capacidades distintas. Esto permite obtener distintos puntos de vista y maneras de interactuar con el prototipo, de esta forma es posible extraer un nivel más rico de información.

Participaron quince personas a las que se les comunicó el propósito de las sesiones de playtesting, el tipo de información que se recogería durante las sesiones, la confidencialidad de la participación, y la relación del proceso con esta testina. De las 15 personas, 47% corresponden al género masculino y 53% al femenino. En cuanto a el rango etario, el 27% de los participantes tienen menos de 20 años de edad, 53% tiene entre 20 y 40 años, y el 20% restante tiene más de 40 años de edad. En cuanto al rango de edades, podemos mencionar que el participante con menor edad tiene 12 años, mientras que el participante con mayor edad tiene 63 años, un rango de edad de 51 años. Entre las discapacidades y dificultades de algunos jugadores durante la realización de la prueba se puede observar: paraplejía, retraso madurativo, hipoacusia unilateral, ceguera parcial, síndrome de down, hipoacusia leve y dificultades visuales leves.

El rol del participante consiste en interactuar directamente con el juego durante las sesiones de playtesting y posteriormente brindar un feedback al observador. El participante de ninguna manera se encarga del setup del ambiente de prueba o de registrar datos sobre la sesión.

Observador

Si bien el rol de observador durante las pruebas puede ser tomado por una o más personas, durante el desarrollo de este prototipo, el rol de observador estuvo a cargo del tesista, siendo el único observador presente durante las pruebas.

El rol del observador consiste en dirigir la sesión de playtesting, armar el setup del ambiente de prueba, interactuar con el participante (no con el juego) y registrar tanto los datos observados objetivamente como el feedback de cada participante.

Playtesting

Las sesiones de playtesting se realizaron teniendo en cuenta los lineamientos planteados por Mirza-Babaei (2016), quien propone realizar un “*setup*” o configuración del playtesting involucrando múltiples aspectos:

- La preparación del lugar o espacio físico donde se realizan las pruebas.
- La preparación del equipamiento necesario para que el participante pueda probar el prototipo.
- La preparación de las herramientas de observación.
- La recolección de datos por parte del observador.

Se optó por utilizar el espacio físico personal del participante como lugar para realizar las pruebas y observación durante las sesiones de playtesting. Esto quiere decir que las sesiones se realizaron en el domicilio de los participantes, con la finalidad de favorecer la comodidad de los participantes y la fluidez en la toma de los resultados. Para algunos de los participantes la tranquilidad del ambiente familiar fue un fuerte requerimiento para poder realizar las pruebas.

El equipamiento preparado para ser presentado a los participantes durante las pruebas consistió en:

- Una computadora portátil (notebook) con sistema operativo Ubuntu 20.04.
- Un ejecutable de la última versión del prototipo previamente abierto en la computadora.
- Un joystick inalámbrico de Xbox 360.
- Auriculares del tipo over-ear.

Para la observación de la prueba no se utilizó ningún tipo de software de captura de pantalla ni captura de datos in-game³⁷. En lugar de esto, se optó por realizar una observación visual y toma de notas manualmente durante la prueba. Esta observación se realizó de una forma metódica, de acuerdo a una guía elaborada para este propósito. El documento anexo a esta

³⁷ Datos generados dentro del juego, capturados durante la partida para ser analizados posteriormente.

tesina, “Guía para la conducción de las sesiones de playtesting”, describe las pautas y protocolos utilizados para la realización de la prueba y la toma de datos.

Toma de datos y feedback

La recolección de datos durante la sesión de playtesting también fue realizada de forma metódica. Para ello se consideraron aspectos de la interacción del participante con el juego, que fueran sencillos de medir y de interpretar, y que permitieran observar un cambio a través del tiempo. La plantilla utilizada para la recolección de datos durante las sesiones de playtesting permitió realizar una validación de los objetivos de diseño con cada iteración, identificar fallos y puntos de mejora, pero también identificar el progreso.

La figura 6.18 muestra los datos recolectados durante una de las sesiones de playtesting. Por razones de confidencialidad se anonimizó la información personal. La planilla de registro del playtesting está organizada en 7 secciones:

1. Información del prototipo utilizado durante la prueba.
2. Información sobre la prueba.
3. Datos sobre el controlador utilizado.
4. Información sobre la partida.
5. Observaciones realizadas durante la partida.
6. Puntajes dados por el observador al final de la sesión.
7. Feedback dado por el participante.

Versión	mahur-0.1.0
Link	...
Prueba	
Fecha	31/08/2022
Participante	Raúl ...
Plataforma	Desktop (Linux)
Controlador	
Tipo	Teclado
Facilidad	Media
Partida	
Tiempo Bosque Largo	0,05
Tiempo Templo Chuluñe	1,46
Tiempo Runalún	4,06
Tiempo La Salida	7,05
Observaciones	

Accesibilidad	UI Menu principal, no pudo usarla con el teclado, tuvo que utilizar el mouse para poder avanzar.
Juego	Confundido, terminó el nivel muy rápido, no entendía qué estaba pasando o cual era el objetivo.
Control	Pudo usar los controles intuitivamente y pudo navegar con facilidad, excepto por el doble salto.
Otros	El nivel no estaba completo, difícil de comprender dónde terminar
Notas	Lo completó en su mayor parte solo, necesitó ayuda para comprender doble salto. Problemas para navegar el menú principal e iniciar el juego
Puntajes /5	
Complejidad	5
Navegación	4
Controles	3
UX/UI	1
Accesibilidad	1
Promedio	2,8
Feedback	El juego está sin terminar. No se entiende con facilidad hacia donde ir. La mecánica del doble salto se siente rara, la altura y velocidad del salto no se siente correcta.

Figura 6.18 Ejemplo de datos recogidos durante una sesión de playtesting

La primera sección de la planilla identifica la versión de prototipo utilizado durante la sesión de playtesting, se incluye el nombre de la versión y un link al ejecutable correspondiente. Esto fue necesario para mantener un registro sobre qué versión está probando cada participante.

En la sección de prueba se registra la fecha en la que se realizó la sesión, el nombre completo del participante (anonimizado), y la plataforma en la que se probó. Las plataformas en las que se testó el prototipo fueron dos: ejecutable nativo para Linux en Ubuntu 20.04 y una versión para navegador en Google Chrome.

La siguiente sección (controlador) registra el tipo de controlador (teclado, joystick, teclado+joystick) que utilizó el participante durante la partida. El tipo “teclado+joystick” significa que el jugador utilizó durante un tiempo considerable de la partida los dos controles.

La sección de la partida registra los tiempos en los que el jugador avanzó a través del prototipo. Se lleva registro del tiempo en el que alcanzó cada uno de los 4 puntos clave del nivel. Esto nos permite tener un registro aproximado del ritmo en el que el participante avanzó por el nivel. Este registro acompañado de notas manuales de observación permiten conocer por ejemplo, si el participante avanzó rápidamente porque los puzzles no fueron lo suficientemente desafiantes, o tuvo problemas para avanzar en alguna sección del prototipo.

A continuación, la sección de observaciones hace foco sobre cuatro aspectos relevantes para los fines de esta tesis, en relación al prototipo:

- **Accesibilidad:** observaciones relacionadas a la accesibilidad, como por ejemplo cuando un participante no puede leer las instrucciones en pantalla porque el texto es muy pequeño.
- **Juego:** observaciones relacionadas al diseño del juego, diseño de los niveles y el gameplay.
- **Control:** observaciones sobre el uso de los controles, como por ejemplo dificultad para realizar ciertas acciones con un tipo de controlador.
- **Otros:** observaciones que no entran en las categorías anteriores pero que son dignas de mención
- **Notas:** se puede ingresar una observación general que englobe la experiencia durante la sesión de playtesting.

La sección de puntajes se completa objetivamente teniendo en cuenta el desempeño del participante durante la sesión. Este puntaje es un número del 1 al 5, donde 1 es el peor puntaje y 5 es el mejor. Para registrar los datos de esta sección se permite al observador consultar con el participante y obtener más información para registrar un número que represente mejor la realidad. Los aspectos sobre los que se hace foco en esta sección son los que se desean monitorear a través del tiempo, para validar que los objetivos planteados en la etapa de diseño fueron cumplidos exitosamente. Los cinco aspectos registrados son:

- **Complejidad:** qué porcentaje del prototipo fue completado exitosamente durante la sesión de playtesting.
- **Navegación:** facilidad con la que el jugador pudo avanzar en el nivel, esto incluye la interacción con personajes y objetos interactivos.
- **Controles:** mide qué tan intuitivos y naturales se sienten o no los controles, esto incluye facilidad para mover el personaje e interactuar con el nivel.
- **UX/UI:** mide la correcta implementación de los elementos de la UI, como por ejemplo la utilización de tipografías y colores, y también la experiencia del usuario interactuando con los elementos de UI.

- Accesibilidad: puntaje asociado a la correcta implementación de estrategias de accesibilidad dentro del prototipo como el desarrollo de los subtítulos.
- Promedio: calculado automáticamente a partir de los puntajes de los aspectos anteriores, sirve como indicador general

La última sección de la plantilla se reserva para registrar una breve nota sobre el feedback brindado por el participante. Este paso fue importante para entender los problemas y necesidades particulares de cada participante con respecto a cada una de las iteraciones del prototipo. En el documento anexo “Guía para la conducción de las sesiones de playtesting” se agrega más detalles sobre la metodología utilizada para conducir las pruebas.

6.6 Publicación del prototipo

El proyecto original, junto con el código, y los recursos multimedia necesarios para construir el prototipo se encuentran disponibles en un repositorio público de Github en el siguiente enlace web: <https://github.com/FEDE0D/ryn-lyn>.

Una versión ejecutable en navegadores está disponible en el sitio web de Itch.io en el siguiente enlace web: <https://fede0d.itch.io/mahur-el-ultimo-runalun>

6.7 Conclusiones

De la construcción de un prototipo accesible usando Godot Engine se pueden recuperar los siguientes aprendizajes:

Como primer punto importante, la versión final del prototipo logró cumplir con los objetivos de accesibilidad planteados en el documento de diseño de juego. Si bien los objetivos de accesibilidad se fueron modificando a través del tiempo que duró el desarrollo, no lo hicieron mutando hacia ideas diferentes de las originales, sino que se fueron refinando y alcanzando una definición menos abstracta y más detallada. Un ejemplo de esto fue el objetivo planteado inicialmente de “brindar algún tipo de soporte para bajar la dificultad de la partida”. Este objetivo está asociado a la idea de que jugadores con capacidades cognitivas o motrices disminuídas puedan encontrar este tipo de asistencias útiles durante la partida. Inicialmente el diseño del juego contemplaba enemigos mucho más agresivos y en mayor cantidad, además de existir el concepto de “energía” o “vida” que el personaje principal podía perder al recibir un daño. Estas ideas fueron rápidamente descartadas en las primeras iteraciones, luego de verificar que para algunos jugadores esto significaba una dificultad muy grande. Esto generó la hipótesis de que algunos jugadores no eran lo suficientemente rápidos o hábiles con los controles, o no percibían lo suficientemente rápido las acciones de los enemigos, y que por lo tanto se decidió ajustar la velocidad de ataque

de los mismos. Luego de realizar la prueba con el prototipo reajustado, algunos jugadores seguían sin superar el combate con enemigos. Durante el rediseño de este objetivo de accesibilidad, se pensó en agregar una nueva opción al menú de accesibilidad para poder jugar partidas sin enemigos. A diferencia de lo esperado, este rediseño demostró ser una opción peor que la anterior, dado que al no existir enemigos, la experiencia de juego no era la misma ni tan divertida. Es por esto que se decidió cambiar la manera de alcanzar el objetivo de “brindar algún tipo de soporte para bajar la dificultad de la partida”, y se realizó un rediseño de los enemigos y los niveles. En este nuevo diseño el combate fue reducido al mínimo y los niveles contienen muy pocos enemigos, muchos de ellos fuera del alcance del jugador. Esto permitió que todos los jugadores, sin importar su capacidad cognitiva o motriz, tuvieran la misma experiencia de juego con respecto al combate y los enemigos. Lo mismo ocurrió con el resto de los objetivos de accesibilidad, que si bien mantuvieron sus conceptos originales, se adaptaron hacia objetivos cada vez más ajustados a la realidad del prototipo. El segundo punto importante a resaltar fue la experiencia utilizando una herramienta que permite agregar un efecto de desenfoque en el editor. Esta herramienta fue implementada mediante un shader muy simple y se utilizó para “simular” dificultades en la visión del jugador. Esto permitió validar ajustes de tipografías y tamaños de textos y elementos visuales del juego de una manera rápida y más confiable. La figura 6.19 muestra una captura de pantalla del editor mientras se está utilizando el efecto. La parte izquierda de la figura muestra el efecto aplicado con un valor de 2, y a la derecha con un valor de 0.



Figura 6.19 Ejemplo de aplicación del efecto de desenfoque dentro del editor de juego.

La utilización del shader durante el desarrollo fue una herramienta que sirvió como validación y cuyo costo de implementación fue muy bajo. La figura 6.20 muestra las 8 líneas de código del shader que permite agregar este efecto dentro del editor.

```

1  shader_type canvas_item;
2
3  uniform float lod: hint_range(0.0, 5) = 0.0;
4
5  void fragment(){
6  >   vec4 color = texture(SCREEN_TEXTURE, SCREEN_UV, lod);
7  >   COLOR = color;
8  }
```

Figura 6.20 Código de shader del efecto de desenfoque

Como tercer punto vale la pena hacer mención del costo extra que representó la implementación de la accesibilidad. Desde mi experiencia profesional, personal y subjetiva del desarrollo de este prototipo, puedo decir que el tiempo abocado a la implementación de la accesibilidad representó aproximadamente un 40% del tiempo total de desarrollo. Esto se puede interpretar de varias maneras. Bajo un punto de vista, se puede decir que el costo asociado a implementar estrategias de accesibilidad fue alto, dado que representó casi la mitad del tiempo de desarrollo. Esto puede verse como una “desventaja” en cierta manera, si lo que se desea es completar el desarrollo en el menor tiempo posible. Pero esto es así si se ven a las opciones de accesibilidad como algo “opcional”, como algo que no todos los jugadores van a experimentar o hacer uso durante la partida, algo que no es completamente cierto. Según mi experiencia trabajando en la accesibilidad para este prototipo, las estrategias de accesibilidad no solo tienen impacto sobre jugadores con discapacidad o sobre un grupo reducido del total de jugadores. Por el contrario, las exploraciones en el diseño resultantes de las pruebas de accesibilidad derivaron en rediseños sobre la experiencia general del juego, algo que afectó en menor o mayor medida en todos los jugadores. Ejemplos de esto son:

- Simplificación de los enemigos y el combate.
- Eliminación del double-tapping³⁸ necesario para realizar un salto en el aire y para activar las puertas.
- Simplificación de platforming³⁹ necesario para alcanzar ciertas áreas elevadas.
- Uso de efectos de sonido para indicar acciones importantes dentro del contexto de juego: salto del personaje, apertura de puertas, presentación de nueva área.

El último punto importante a resaltar es que si bien se alcanzaron los objetivos de accesibilidad propuestos en el documento de diseño, queda claro que la accesibilidad es un proceso iterativo de mejora continua. No existen objetivos predefinidos ni una meta a

³⁸ Acción repetida y rápida de presionar dos veces un botón del controlador para poder realizar una acción particular.

³⁹ Acción de moverse y saltar entre plataformas.

alcanzar, sino que siempre existe la posibilidad de seguir trabajando en este aspecto. En el caso particular de este prototipo, existen algunas mejoras pendientes que pueden ser trabajadas a futuro. En el caso de los diálogos, una característica para seguir iterando a futuro es la de ajustar la velocidad del texto en pantalla, la personalización de la tipografía y colores, y la posibilidad de contar con un historial de diálogo que se pueda consultar en cualquier momento durante el juego y releer diálogos pasados. En cuanto a los niveles, todavía existe mucho lugar para seguir iterando en el diseño general del platforming. Algunos de los caminos pueden ser poco claros, y ciertos saltos requieren una precisión que se esperaría de niveles más avanzados y no de un primer nivel.

7. Conclusiones finales

En este capítulo se presentan las conclusiones finales del presente trabajo y se plantean posibles líneas de trabajo futuras.

7.1 Sobre los objetivos propuestos en la tesina

Considero que los objetivos planteados para el presente trabajo de tesis fueron alcanzados de manera exitosa. El objetivo general de la tesina de *“contribuir en la creación de videojuegos inclusivos en el sector de la industria independiente, y específicamente colaborar con la comunidad de Godot Engine, haciendo foco en la discapacidad auditiva”* fue satisfecho de acuerdo a los resultados esperados para este trabajo. Por un lado el estudio y análisis del sector de videojuegos permitió identificar al “sector independiente”, y poder reconocer las problemáticas y dificultades que atraviesan a este grupo de desarrolladores y desarrollos. El Catálogo de buenas prácticas en el diseño de accesibilidad, elaborado como parte del trabajo de esta tesina, permitirá a los desarrolladores independientes poder contar con una estrategia metodológica para la validación y correcto diseño de la accesibilidad en sus videojuegos. Adicionalmente, el plugin “Godot Subtitles” y el prototipo de videojuego, desarrollados y publicados en un repositorio público, para esta tesina permitirá a los desarrolladores independientes contar con herramientas y ejemplos de implementación de estrategias de accesibilidad que apuntan a mejorar la accesibilidad de los juegos creados en Godot Engine.

7.2 Sobre la accesibilidad en la industria independiente

De los primeros capítulos de este trabajo se pueden extraer algunas conclusiones importantes.

En primer lugar se analizó al sector independiente de videojuegos, pudiendo identificar las características que lo definen en tres aspectos: independencia financiera, creativa y editorial. Se examinó el impacto económico y tamaño de la industria a nivel global y a nivel nacional.

Se analizaron casos de estudio de videojuegos independientes, y se realizaron entrevistas a desarrolladores de este sector, de las cuales se logró extraer un listado de carencias y dificultades en el desarrollo de estrategias de accesibilidad en los videojuegos independientes. La conclusión más importante a la que se arribó de este análisis es que los sectores independientes suelen considerar con baja o nula prioridad a las tareas de accesibilidad. La desvalorización de las tareas relacionadas a las mejoras de la accesibilidad dentro de los proyectos independientes se deben a múltiples factores, entre

ellos la falta de conocimiento sobre diseño de accesibilidad, y errores de planificación del proyecto, entre otros. Por estas razones se decidió construir, como parte del trabajo de esta tesina, el “Catálogo de buenas prácticas en el diseño de accesibilidad en videojuegos”. Este catálogo permite a los desarrolladores contar con descripciones detalladas y ejemplos de aplicación de estos conceptos de accesibilidad en la etapa de diseño del videojuego. Además, cuenta con un sistema de clasificación que permite conocer entre otras cosas el nivel de dificultad en la implementación, casos de estudio en videojuegos concretos, y recursos adicionales que amplían el concepto original.

7.3 Sobre el desarrollo en Godot Engine: el plugin “Godot Subtitles” y la construcción y testeo de un prototipo accesible

En el Capítulo 5 se describió la relación entre el proyecto de software libre Godot Engine, y la comunidad de desarrolladores de videojuegos independientes. Se analizó el estado actual del motor en cuanto a accesibilidad, y se identificaron limitaciones que atentan contra la implementación de la accesibilidad en los juegos creados. Se identificó la falta de soporte de subtítulos en Godot Engine como una de las limitaciones con mayor impacto para los jugadores argentinos, por lo cual se planteó el desarrollo de un plugin para dicho motor que agregue la funcionalidad necesaria para permitir la creación y edición de subtítulos.

En el Capítulo 6 se detalló el trabajo realizado para crear un prototipo que implementa estrategias de accesibilidad en un juego desarrollado en Godot Engine, incorporando el plugin con soporte para subtítulos. Para esto se diseñó y experimentó con una estrategia de validación de las funcionalidades de accesibilidad basada en sesiones de prueba con jugadores (playtesting).

De ambos desarrollos, plugin “Godot subtitles” y el prototipo, se pueden recuperar las siguientes conclusiones. En primer lugar, se puede observar que la existencia de herramientas que agregan soporte de accesibilidad en Godot Engine, como el plugin desarrollado en esta tesina, o el plugin que agrega soporte Text-To-Speech, facilitan en gran parte la implementación de estrategias de accesibilidad. Acortan el tiempo destinado al desarrollo de tareas de accesibilidad dentro del proyecto, algo que se aprovecha sobre todo en proyectos con tiempo de desarrollo corto y número de desarrolladores limitados, como es el caso de los desarrollos independientes. En segundo lugar, el conocimiento sobre diseño de accesibilidad permite mejorar en gran medida la planificación del proyecto. Pero no basta simplemente con agregar opciones de accesibilidad a los juegos, sino que también es importante realizar validaciones sobre estas opciones con jugadores reales, algo que se pudo comprobar durante las sesiones de playtesting y las iteraciones en el desarrollo del prototipo.

7.4 Líneas futuras de trabajo

Varias de las temáticas analizadas y estudiadas, en el presente trabajo de tesis, se abordaron desde el punto de vista de la diversidad funcional de los jugadores, centrándose en las dificultades auditivas. Las estrategias planteadas para el diseño de accesibilidad y los desarrollos realizados en Godot Engine tuvieron como objetivo mejorar principalmente la inclusión de jugadores con dificultades auditivas.

Para una futura línea de trabajo se propone analizar las limitaciones que presenta el desarrollo de accesibilidad bajo otros puntos de vista de la diversidad de jugadores, como puede ser jugadores con discapacidades visuales, motrices o cognitivas. En base al análisis de estas limitaciones, continuar con la propuesta de mejoras sobre el motor Godot Engine y la ampliación del “Catálogo de buenas prácticas en el diseño de accesibilidad en videojuegos”, es una línea de trabajo futuro de esta tesina que permitiría facilitar la implementación de accesibilidad a los desarrolladores independientes que utilizan este motor.

Se identifica también como una línea de trabajo futura la realización de mejoras sobre el plugin “Godot Subtitles” implementado en esta tesina. Una de las mejoras posibles, que tendría un gran impacto en los jugadores, es el soporte de personalización de los subtítulos, dado que ofrecería la posibilidad de cambiar el color, tamaño y tipografía de los subtítulos en los juegos creados con Godot Engine, beneficiando a gran cantidad de jugadores, sobre todo a aquellos con dificultades visuales.

Bibliografía

- A. L. Brooks, S. Brahnam, B. Kapralos, & L. C. Jain. (2017) Recent Advances in Technologies for Inclusive Well-Being.
- ADA National Network. "What is the Americans with Disabilities Act (ADA)". Recuperado de: <https://adata.org/learn-about-ada>
- ADVA Website. "¿Qué es ADVA?" (2022). Recuperado de: <https://www.adva.vg/conocenos/>
- Aarseth, E. (2014), "Ontology", In Wolf & Perron (eds.), The Routledge Companion to Video Game Studies, pp.484-92. New York and London: Routledge.
- Able Gamers Website. (2020) "Accessible Player Experiences (APX)". Recuperado de: <https://accessible.games/accessible-player-experiences>
- Able Gamers Website. "Get help" (2022) Recuperado de: <https://ablegamers.org/get-help/>
- Accessibility Unlocked Website. "Access Jam!" (2022) Recuperado de: <https://www.accessunlocked.games/accessjam>
- Altenburger, T. (2021). ScourgeBringer: an indie accessibility post-mortem. IGDA GASIG
- Andrew Kirkpatrick, Michael Cooper. (2018). "WCAG 2.1 IS A W3C Recommendation". Recuperado de: <https://www.w3.org/blog/2018/06/wcag21-rec/>
- Apple Accesibility. Recuperado de: <https://www.apple.com/accessibility/>
- AskUbuntu website. "Is it possible to force all audio to mono?" Recuperado de: <https://askubuntu.com/questions/1393150/is-it-possible-to-force-all-audio-to-mono>
- BBC Subtitle Guidelines, Version 1.2.1. (2022) Recuperado de: <https://www.bbc.co.uk/accessibility/forproducts/guides/subtitles/#Identifying-speakers>
- BBC Subtitle Guidelines. "Sound effects". Recuperado de: <https://www.bbc.co.uk/accessibility/forproducts/guides/subtitles/#Sound-effects>
- BBCode Documentation. Godot Engine Website. (2022). Recuperado de: https://docs.godotengine.org/en/stable/tutorials/ui/bbcode_in_richtextlabel.html
- BBCode Format. Recuperado de: <https://www.bbcode.org/reference.php>
- Baker, M. (2021). Where to Begin: Games Accessibility
- Baker, M. 2021 Predictions #1 Puzzle Games Becomes the Top Grossing Genre on Mobile (2021) Recuperado de: <https://www.deconstructoroffun.com/blog/2021/1/9/2021-predictions-1-merge-becomes-the-ew-match#>
- Banco Interamericano de Desarrollo. (2019). Los videojuegos no son un juego.
- Bartoli, L., Garzotto, F., Gelsomini, M., Oliveto, L., & Valoriani, M. (2014). Designing and evaluating touchless playful interaction for ASD children.

- Bateman, C. (2014). Meet bertie the brain, the world's first arcade game, built in toronto. Spacing Magazine.
- Benyon, D. (2019). Designing User Experience. London, UK: Pearson Education Limited.
- Biblioteca CEDIB-INADI. (2018). Accesibilidad e inclusión de personas con discapacidad. Guía Temática.
- Blacknut Website, (2022). Recuperado de: <https://www.blacknut.com/>
- Blacknut Website. Video games and handicap: Blacknut and HitClic (Handigamers) are committed to the accessibility of cloud gaming. (2021) Recuperado de: <https://www.blacknut.biz/press-release/video-games-handicap-blacknut-hitclic-handigamers-accessibility-cloud-gaming>
- Boudeguer Simonetti, Prett Weber, P., & Squella Fernández, P. (2010). Manual de accesibilidad universal : ciudades y espacios para todos (1a. ed.). Corporación Ciudad Accesible.
- Boutighmass, M. , 2022, The microchip crisis in the game console and electronics sector Box Office Mojo Website, Avengers: Endgame (2022). Recuperado de: <https://www.boxofficemojo.com/release/rl3059975681/>
- Brittne, Nelson-Kakulla, P.h.D. 2020 Gaming Trends of the 50+. Washington, DC: AARP Research, December 2019. <https://doi.org/10.26419/res.00328.001>
- Cairns, P., Power, C., Barlet, M., Haynes, G., Kaufman, C., & Beeston, J. (2019). Enabled Players: The Value of Accessible Digital Games. Games and Culture, 155541201989387. doi:10.1177/1555412019893877
- Can I Play That? Website. "About Can I Play That?" (2022) Recuperado de: <https://caniplaythat.com/about-can-i-play-that/>
- Carr, D. (2010). Constructing disability in online worlds: Conceptualising disability in online research.
- Casal, K., Coquette, B., Di Scala, D., Fernández, C., Pastore, M. y Zocola, L. (2012). Inclusión: Un acontecimiento para todos. Recuperado de: <http://www.ianamericas.org/inclusion-unacontecimiento-para-todos/>
- Chandler, H. (2006). Taking Video Games Global: An Interview with Heather Chandler, author of The Game Localization Handbook.
- Chen H., (2022). The Impact of the Covid-19 on the Video Game Market, p. 73
- Christina Gough (2022) "Revenue of the global eSports market 2020-2025". Statista. Recuperado de: <https://www.statista.com/statistics/490522/global-esports-market-revenue/>
- Clarke, M.J. & Wang, C. 2020. Indie Games in the digital Age.
- Coterón, A, Gámez, G, Rodriguez R. (2018). "Accesibilidad e inclusión: dos aspectos clave para las personas con discapacidad". Recuperado de:

<https://blogs.worldbank.org/es/latinamerica/accesibilidad-e-inclusi-n-dos-aspectos-clave-par-a-las-personas-con-discapacidad>

Cristina Belén Sampedro-Palacios (2019) Innovación Social como herramienta en la transformación de una sociedad inclusiva. Accesibilidad e Innovación Social.

Cunningham L, Murphy O. Embracing the Universal Design for Learning Framework in Digital Game Based Learning - A Set of Game Design Principles. Stud Health Technol Inform. 2018;256:409-420. PMID: 30371502.

DAGERSystem Website. "About us" (2022). Recuperado de:

<https://dagersistem.com/about-the-editors/>

DFC Intelligence. (2020) Global Video Game Consumer Segmentation.

Dalgleish, M. (2018). There are no universal interfaces: how asymmetrical roles and asymmetrical controllers can increase access diversity

Dayi Lin, Cor-Paul Bezemer, and Ahmed E Hassan. 2018. An empirical study of early access games on the Steam platform. Empirical Software Engineering 23, 2 (2018), 771–799.

Delfino, A. En la Argentina, el 90% de la facturación de videojuegos está en manos de cuatro empresas (2022) Recuperado de:

<https://www.telam.com.ar/notas/202202/584821-argentina-videojuegos-facturacion.html>

Departamento de Urbanismo | FAU | Universidad de Chile (2019) Claves en la formación del discurso de Inclusividad en la Arquitectura Chilena: 1960 - 2016

Deuze, M. & Martin, C.B. (2009). The Independent Production of Culture : A Digital Games Case Study. Games and Culture, Vol. 4, No. 3.

Deuze, M., Martin, C., & Allen, C. (2007). The professional identity of gameworkers.

Digital Education Strategies. Digital Accessibility as a Business Practice.

Disability Horizons. The Equality and Accessibility of eSports. (2019) Recuperado de:

<https://disabilityhorizons.com/2019/07/the-equality-and-accessibility-of-esports/>

Documentación de GDNative. Godot Engine Website. (2022) Recuperado de:

https://docs.godotengine.org/en/stable/tutorials/scripting/gdnative/gdnative_cpp_example.html

Documentación de GDScript. Godot Engine. (2022). Recuperado de:

<https://docs.godotengine.org/en/stable/tutorials/scripting/gdscript/index.html>

Documentación de módulos. Godot Engine Website. (2022) Recuperado de:

https://docs.godotengine.org/en/stable/development/cpp/custom_modules_in_cpp.html

Documentación de plugins. Godot Engine Website. (2022) Recuperado de:

https://docs.godotengine.org/es/stable/tutorials/plugins/editor/making_plugins.html

Donnelly, J. PC Gamer (2018). GTA 5 estimated to be the most profitable product of all time.

Recuperado de:

<https://www.pcgamer.com/gta-5-estimated-to-be-the-most-profitable-entertainment-product-of-all-time/>

Dupire J. & Westin, T. (2016). Evaluation and Redesign of a Curriculum Framework for Education About Game Accessibility

Díaz LS, Esquivel GEC, Buenrostro MG, et al. Factores de riesgo asociados al trastorno de procesamiento auditivo central. *Investigación en Discapacidad*. 2022;8(2):51-61. doi:10.35366/105478.

Dörner, R. 2016. *Serious Games, Foundations, Concepts and Practice*.

Editorial Aela. (2022) "Accessibility In Games And Inclusive Design". Recuperado de:

<https://aelaschool.com/en/userexperience/accessibility-games-inclusive-design/>

Ellis K., Kao, K. (2019): *Who Gets to Play? Disability, Open Literacy, Gaming*.

Ellis. B, Hamilton, I, Westin, T. (2020). *The Digital Gaming Handbook*.

Entertainment Software Rating Board (ESRB) Blog (2022) "Half of All ESRB Ratings Assigned in 2021 Were E for Everyone". Recuperado de:

<https://www.esrb.org/blog/half-of-all-esrb-ratings-assigned-in-2021-were-e-for-everyone/>

FUNDAV Website. "Cómo trabajamos" (2022). Recuperado de: <https://fundav.com/>

FUNDAV Website. "Game Work Jam" (2022). Recuperado de:

<https://fundav.com/game-work-jam/>

Family Gaming Database Website (2022). Recuperado de: <https://www.taminggaming.com/>

GA Website. "Conference" (2022). Recuperado de: <https://www.gaconf.com/conference/>

GMTK Game Jam, (2021) Recuperado de: <https://godotes.com/gmtk-game-jam-2021/>

Gadia, D., Granato, M., Maggiorini, D., Marras, M., & Ripamonti, L. A. (2017). A

Touch-based Configurable Gamepad for Gamers with Physical Disabilities. In CHIRA (pp. 67-74).

Game Accessibility Guidelines Website. "Why And How?" Recuperado de:

<https://gameaccessibilityguidelines.com/why-and-how/>

Game Maker's Toolkit Twitter account (2022), Recuperado de:

<https://twitter.com/gamemakerstk/status/1552183527847989248/>

Game Maker's Toolkit YouTube Channel. (2022). Recuperado de:

<https://www.youtube.com/user/McBacon1337>

GamesHub (2022) "How inclusive design in video games benefits everyone". Recuperado de:

<https://www.gameshub.com/news/opinions-analysis/accessibility-day-2022-how-inclusive-design-features-in-video-games-benefit-many-19161/>

Gauci, F. 2021. *Game Accessibility for Children with Cognitive Disabilities*

George H. Pike. (2010). "President Obama Signs the 21st Century Communications and Video Accessibility Act". Recuperado de:

<http://newsbreaks.infotoday.com/NewsBreaks/President-Obama-Signs-the-21st-Century-Communications-and-Video-Accessibility-Act-70569.asp>

Global Game Jam Website. "What is a game jam?" (2022). Recuperado de:

<https://globalgamejam.org/what-game-jam>

Gonzalo Arjona Jiménez (2012) "Historia de la Accesibilidad: De la eliminación de barreras a la Accesibilidad Universal, pasando por el Diseño para Todos y la Vida independiente".

Recuperado de:

<http://periodico.laciudadaccesible.com/portada/opinion-la-ciudad-accesible/item/2554-historia-de-la-accesibilidad-de-la-eliminacion-de-barreras-a-la-accesibilidad-universal-pasando-por-el-diseno-para-todos-y-la-vida-independiente>

Gonzalo Arjona. (2014) Movilidad, accesibilidad y discapacidad. Una historia de logros. Actualidad de la accesibilidad. ASEPAU. Implícate N°4

Gordon, L. Game Subscriptions Are On the Rise. Indies Could Suffer. (2021) Recuperado de: <https://www.wired.com/story/game-subscription-indie-developers-gamepass-playstation/>

Grammenos, Dimitris & Savidis, Anthony & Stephanidis, Constantine. (2009). Designing Universally Accessible Games. 10.1145/1486508.1486516.

Grazzini, M & Lomanto, J. (2022) "T2 - E1: Videojuegos desde el Estado". Early Access Podcast, Image Campus

Gu, T. (2022) "The Evolution of Hypercasual Mobile Games Across the East and West".

Recuperado de:

<https://newzoo.com/insights/articles/hypercasual-mobile-games-across-the-east-and-west>

H. Conception. (2017) Video Game Therapy as an Intervention for Children With Disabilities.

Hamilton I. (2021) Independent Games Summit: Accessibility on a Shoestring

Hartmann, A. (2018). Playing games and using AAC.

Holly Tuke. (2021) "Assistive technology devices: How disabled people use the web".

Recuperado de:

<https://business.scope.org.uk/article/assistive-technology-devices-definitions-how-disabled-people-use-the-web/>

I. Khaliq, I. Dela Torre. (2019) A Study on Accessibility in Games for the Visually Impaired

IGDA Game Accessibility Special Interest Group. (2020) Sig guidelines. Recuperado de:

<https://igda-gasig.org/get-involved/sig-initiatives/resources-for-gamedevelopers/sig-guidelines>.

IGDA-GASIG Website. "About the SIG" (2022). Recuperado de:

<https://igda-gasig.org/about-the-sig/>

INDEC. Agencia Nacional de Discapacidad. (2018): Estudio Nacional sobre el Perfil de las Personas con Discapacidad.

J. Molina-López, N. Medina-Medina. (2019). Design proto-patterns to improve the interaction in video games of people with color blindness

Josh Straub (2015) "Accessibility Game Review – Batman Arkham Knight". DAGERSystem.

Recuperado de: <https://dagersistem.com/disability-game-review-batman-arkham-knight/>

Juan Piñeiro-Chousa, M. Ángeles López-Cabarcos, Ada M. Pérez-Pico, Jérôme Caby.

(2023) The influence of Twitch and sustainability on the stock returns of video game companies: Before and after COVID-19.

Kate Sánchez (2019) "SXSW 2019: Michael Espinoza, Talks Accessibility in Gaming and Shares his Story". Recuperado de:

<https://butwhytho.net/2019/03/sxsw-2019-michael-espinoza-talks-accessibility-in-gaming-and-shares-his-story/>

Kovanto, A. (2013) The Improvements for Indie Game Development.

L. Pereira. (2021). The elusive "indieness": Measuring the defining characteristics of indie games.

Legislatura de la Ciudad Autónoma de Buenos Aires. Comisión para la Plena Participación e Inclusión de las Personas con Discapacidad (COPIDIS). "LEY 962. Accesibilidad física para todos". Recuperado de:

https://www.buenosaires.gob.ar/sites/gcaba/files/ley_962_-_codigo_de_edificacion_de_la_ciudad_de_buenos_aires.pdf

Livingston, Ian & Mandryk, Regan & Stanley, Kevin. (2010). Critic-proofing: how using critic reviews and game genres can refine heuristic evaluations.

M. Baker. (2020) Deaf Accessibility in Videogames

M. Garda, P. Grabarczyk. (2016). Is Every Indie Game Independent? Towards the Concept of Independent Game

M. Shin. (2021). A Growth of Accessibility in Video Games. DO-IT News March 2021.

Mahmood, N (2011). Designing video games and interactive applications to enhance learning in children with autism spectrum disorders.

Makuch, E. GameSpot Website (2013). Guinness claims GTAV day-one sales total 11.2 million copies. Recuperado de:

<https://www.gamespot.com/articles/guinness-claims-gtav-day-one-sales-total-112-million-copies/1100-6415455/>

Mangiron, O'Hagan, Orero. (2014). Fun for All: Fun for All: Translation and Accessibility Practices in Video Games.

Marissa Sapega (2020) The History of Digital Accessibility and Why it Matters.

Mark C. Barlet & Steve D. Spohn (2012). Includification: A practical guide to gaming accessibility. The Able Gamers foundation.

Marques, G. (2020). Improve accessibility features for keyboard navigation, UI scaling, TTS, voice recognition, colorblindness and subtitles. Recuperado de:

<https://github.com/godotengine/godot-proposals/issues/983>

Mason. Newzoo: games revenues will surpass \$200 billion in 2022. (2022). Recuperado de:

<https://game-news24.com/2022/05/05/newzoo-games-revenues-will-surpass-200-billion-in-2022/>

Mathew Forde (2019), How do indie developers address accessibility in video games?

Recuperado de:

<https://www.pocketgamer.biz/mobile-mavens/70628/how-do-indie-developers-address-accessibility-in-video-games/>

Microsoft Windows Accesibility Features. Recuperado de:

<https://www.microsoft.com/en-us/accessibility/windows>

Mirza-Babaei, Pejman & Moosajee, Naeem & Drenikow, Brandon. (2016). Playtesting for indie studios.

Molly Follette Story. (1998) The Universal Design File: Designing for People of All Ages and Abilities

Mordor Intelligence. (2020). Gaming Market - Growth, Trends, Forecasts (2020-2025).

Recuperado de: <https://www.mordorintelligence.com/industry-reports/global-games-market>

Moss, C. (2018). Classifying Videogames as Art and Why It Matters Now. MRes.

Newzoo Global Games Market Report 2022. (2022). Recuperado de:

<https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2022-free-version>

Newzoo. (2017) "The Argentinean Gamer | 2017" Recuperado de:

<https://newzoo.com/insights/infographics/the-argentinean-gamer-2017>

Nielsen. "3, 2, 1 ¡Vamos! El videojuego alcanza su máximo nivel durante COVID-19". (2020)

Recuperado de:

<https://www.nielsen.com/insights/2020/3-2-1-go-video-gaming-is-at-an-all-time-high-during-covid-19/>

O. Mayor, J. Bonada, and J. Janer, "Audio Transformation Technologies Applied to Video Games," Paper P1-2, (2011 February)

OMS, Informe mundial sobre discapacidad 2011.

OMS. (2018). "Tecnología de asistencia". Recuperado de:

<https://www.who.int/es/news-room/fact-sheets/detail/assistive-technology>

Omer Kaplan (2019) "Mobile gaming is a \$68.5 billion global business, and investors are buying in". TechCrunch. Recuperado de:

<https://techcrunch.com/2019/08/22/mobile-gaming-mints-money/>

- Organización Mundial de la Salud. "Discapacidad" (2022). Recuperado de: <https://www.who.int/es/news-room/fact-sheets/detail/disability-and-health>
- Papadatou-Pastou, M., Ntolka, E., Schmitz, J., Martin, M., Munafò, M. R., Ocklenburg, S., & Paracchini, S. (2020). Human handedness: A meta-analysis
- Parkin, S. (2011). The Making Of Fez, The Breaking Of Phil Fish. Gamasutra. Recuperado de: http://www.gamasutra.com/view/feature/134934/the_making_of_fez_the_breaking_of_.php
- Patrick Klepek (2016) "How A Player With Disabilities Convinced Naughty Dog To Add More Accessibility Options To Uncharted 4". Recuperado de: <https://kotaku.com/how-one-disabled-player-convinced-naughty-dog-to-add-mo-1777633667>
- Pinelle, D., Wong, N., and Stach, T. (2008). Heuristic evaluation for games: usability principles for video game design.
- Plattner, C. Meinel, and L. Leifer. (2014) Design thinking research: Building innovators.
- Potato. (2018) The Right to Play: Accessibility in Gaming. Recuperado de: <https://medium.com/potato/the-right-to-play-accessibility-in-gaming-a954b01023f>
- Rees-Jones, J. D., Brereton, J. S., & Murphy, D. T. (2015). Spatial audio quality and user preference of listening systems in video games. 223-230. Paper presented at 18th Int. Conference on Digital Audio Effects (DAFx-15), Trondheim, Norway. Recuperado de: http://www.ntnu.edu/documents/1001201110/1266017954/DAFx-15_submission_35.pdf
- RocketBrush Studio Blog (2022) "How much does it cost to develop a game". Recuperado de: <https://rocketbrush.com/blog/how-much-does-it-cost-to-develop-a-game>
- Romina Gala. (2019) Map of the Videogame Cultural Industry in Argentina: a state of affairs.
- S. Anderson, K. Schrier. (2021). Disability and Video Games Journalism: A Discourse Analysis of Accessibility and Gaming Culture.
- S. Hu. (2017) Continuity and discreteness: How input methods influence gaming experiences. Recuperado de: <https://www.gamedeveloper.com/design/continuity-and-discreteness-how-input-methods-influence-gaming-experiences>
- SRT File Format Website (2022). Recuperado de: <https://docs.fileformat.com/video/srt/>
- Schell, J. (2019). The Art of Game Design: A Book of Lenses, Third Edition.
- Schindel, D. Polygon. "The best video essays of 2018". (2018) Recuperado de: <https://www.polygon.com/2018/12/28/18156334/best-youtube-video-essays-2018-watch>
- Scott, M. 2013. Promoting Game Accessibility: Experiencing an Induction on Inclusive Design Practice at the Global Games Jam.
- Shakespeare, T. (2013). The social model of disability. In L. J. Davis (Ed.), The disability studies reader (4th ed., pp. 214–221). Emerald Group Publishing.

- Sinclair B. (2019) "Ubisoft sees high acceptance for opt-out subtitles". Recuperado de: <https://www.gamesindustry.biz/ubisoft-sees-high-acceptance-for-opt-out-subtitles>
- Sitio web argentina.gob.ar. "Ley 24314". Recuperado de: <https://www.argentina.gob.ar/normativa/nacional/ley-24314-713/normas-modificadas>
- Sitio web argentina.gob.ar. "Ley 26378". Recuperado de: <https://www.argentina.gob.ar/normativa/nacional/ley-26378-141317>
- Sitio web argentina.gob.ar. "Sobre la Agencia Nacional de Discapacidad". Recuperado de: <https://www.argentina.gob.ar/andis>
- Sitio web argentina.gob.ar. "JAM Crear juegos con accesibilidad". Recuperado de: <https://www.argentina.gob.ar/jefatura/mediosycomunicacion/crear/jam>
- Sitio web buenosaires.gob.ar. Comisión para la Plena Participación e Inclusión de las Personas con Discapacidad (COPIDIS). "Principios del Diseño Universal". Recuperado de: <https://www.buenosaires.gob.ar/copidis/accesibilidad/principios-del-diseno-universal>
- Sjardijn & Alyson (2019) Exploring accessibility in video games and one button game design. Masters thesis, University of Huddersfield
- Statista. Distribution of video gamers in the United States from 2006 to 2022, by gender. (Julio, 2022). Recuperado de: <https://www.statista.com/statistics/232383/gender-split-of-us-computer-and-video-gamers/>
- Statista. eSports market revenue worldwide from 2020 to 2025 (Abril, 2022). Recuperado de: <https://www.statista.com/statistics/490522/global-esports-market-revenue/>
- Stephen Wilds (2020) "For All The Players: A History Of Accessibility In Video Games". Recuperado de: <https://www.gamingbible.com/features/games-for-all-the-players-a-history-of-accessibility-in-video-games-20200124>
- Szopa, A. (2021). Comparison Analysis of Accessible Features Built into Operating Systems. In: Shin, C.S., Di Bucchianico, G., Fukuda, S., Ghim, YG., Montagna, G., Carvalho, C. (eds) Advances in Industrial Design. AHFE 2021. Lecture Notes in Networks and Systems, vol 260.
- The Game Awards Website. "Innovation in accessibility award added to the game awards" (2020). Recuperado de: <https://thegameawards.com/news/innovation-in-accessibility-award-added-to-the-game-awards>
- The SRT or Subrip subtitle format (2010). Recuperado de: <https://www.visualsubsync.org/help/srt>
- The World Wide Web Consortium (W3C). "How WCAG 2.0 Differs from WCAG 1.0". Recuperado de: <https://www.w3.org/WAI/WCAG20/from10/diff.php>

The World Wide Web Consortium (W3C). “Web Content Accessibility Guidelines 1.0”.

Recuperado de: <https://www.w3.org/TR/WAI-WEBCONTENT/>

Tom Wijman. (2022) “The Games Market in 2022: The Year in Numbers”. Recuperado de:

<https://newzoo.com/insights/articles/blue-games-market-in-2022-the-year-in-numbers>

U.S. Access Board. “Rehabilitation Act of 1973”. Recuperado de:

<https://www.access-board.gov/law/ra.html>

Ubuntu Official Documentation Website (2022) “Ubuntu Desktop Guide - Accessibility”.

Recuperado de: <https://help.ubuntu.com/stable/ubuntu-help/a11y.html.en>

Universidad Nacional de Rafaela, Santa Fe. (2022). Informe General del Observatorio de la Industria Argentina de Desarrollo de Videojuegos

Verschelde, R. (2022) Major milestone ready for testing: Godot 4.0 alpha 1 is out!

Recuperado de: <https://godotengine.org/article/dev-snapshot-godot-4-0-alpha-1>

Verschelde, R. GSoC 2021 - Progress report #1 (2021). Recuperado de:

<https://godotengine.org/article/gsoc-2021-progress-report-1#command-palette>

“Voiceovers and subtitles in different languages”. Grim Fandango Remastered Steam Forums. Recuperado de :

<https://steamcommunity.com/app/316790/discussions/0/620703493336505918>

Von Kameke, L. Gaming market in the Asia-Pacific region - statistics & facts. (2022)

Recuperado de: <https://www.statista.com/topics/2196/video-game-industry-in-asia/>

W3C. “What’s New in WCAG 2.1”. Recuperado de:

<https://www.w3.org/WAI/standards-guidelines/wcag/new-in-21/>

WPUSROBSEACC. “Breve historia de la accesibilidad universal.” Recuperado de:

<https://observatoriodelaaccesibilidad.es/archivos/30>

Warren, J. (2014). It’s time for us to stop calling games “indie”.

Watterton, C. (2022). An Indie Approach to Accessibility. IGDA GASIG.

Westin, Thomas & Ku, JaEun & Dupire, Jérôme & Hamilton, Ian. (2018). Game Accessibility Guidelines and WCAG 2.0 – A Gap Analysis. 10.1007/978-3-319-94277-3_43.

Wijman, T. Games Market Revenues Will Pass \$200 Billion for the First Time in 2022 as the U.S. Overtakes China. (2022). Recuperado de:

<https://newzoo.com/insights/articles/games-market-revenues-will-pass-200-billion-for-the-first-time-in-2022-as-the-u-s-overtakes-china>

Wijman, T. Newzoo’s Games Trends to Watch in 2020. (2019) Recuperado de:

<https://newzoo.com/insights/articles/newzoos-games-trends-to-watch-in-2020>

Wijman, T. Three Billion Players by 2023: Engagement and Revenues Continue to Thrive Across the Global Games Market. (2020) Recuperado de:

<https://newzoo.com/insights/articles/games-market-engagement-revenues-trends-2020-2023-gaming-report>

Wilde, T. Microsoft debuts new cloud-based tools targeting indie game developers. (2022)

Recuperado de:

<https://www.geekwire.com/2022/microsoft-debuts-new-cloud-based-tools-targeting-indie-game-developers/>

Wilson, A., Crabb, M. W3C Accessibility Guidelines for Mobile Games. *Comput Game J* 7, 49–61 (2018). <https://doi.org/10.1007/s40869-018-0058-7>

Wini Bhalla (2021). “Why Linux Is More Accessible Than Windows and macOS”.

Recuperado de:

<https://www.makeuseof.com/why-linux-is-more-accessible-than-windows-macos/>

Witton, C. (2010). Childhood auditory processing disorder as a developmental disorder: The case for a multi-professional approach to diagnosis and management.

Yuan and E. Folmer. (2008). Blind hero: Enabling guitar hero for the visually impaired.

Yuan, Bei, Elke Folmer and Frederick C. Harris. 2010. ‘Game accessibility: a survey’

¿Qué es una MiPyME? (2022) Recuperado de:

<https://www.argentina.gob.ar/produccion/registrar-una-pyme/que-es-una-pyme>

“2022 Essential Facts About the Video Game Industry”. (2022) Theesa. Recuperado de :

<https://www.theesa.com/resource/2022-essential-facts-about-the-video-game-industry/>

Anexo 1: Guía para la conducción de las sesiones de playtesting

El presente documento basado en el protocolo propuesto por Mirza-Babaei (2016), propone una serie de protocolos o reglas a seguir para poder realizar de una forma estandarizada y controlada las pruebas del prototipo o playtesting con jugadores.

Roles

Existen únicamente dos tipos de roles durante el playtesting: el observador y el participante.

Observador

El rol de observador puede ser tomado por una o más personas durante las sesiones. Este rol tiene como responsabilidad la preparación, conducción y registro de la sesión de playtesting. Consiste en dirigir la sesión entera de playtesting, armar el setup de las herramientas y ambiente de prueba, registrar datos objetivamente durante la sesión, e interactuar únicamente con el participante. El observador debe evitar interactuar con el juego.

Participante

El rol del participante es tomado por cualquiera de los jugadores. Este rol consiste en interactuar directamente con el juego durante las sesiones de playtesting, seguir las indicaciones del observador, y posteriormente brindar un feedback sobre la sesión. El participante de ninguna manera se encarga del setup de las herramientas o el ambiente de prueba o de registrar datos sobre la sesión.

Selección de participantes

La selección de participantes debe realizarse previamente al comienzo de las sesiones de pruebas. Es recomendable realizar un listado con información sobre cada uno de los participantes, incluyendo:

- Nombre y Apellido
- Fecha de nacimiento / Edad
- Género
- Datos de contacto: email, teléfono
- Notas / Observaciones

Estos datos son necesarios para poder contactar con el participante en caso de ser necesario, y además permitirán luego realizar un análisis agrupado por alguna de estas características, por ejemplo por edad o género de los participantes. Es importante asegurar

la privacidad de estos datos, y en caso de divulgación poner en conocimiento a los participantes involucrados.

La selección de participantes debe tener en cuenta las características demográficas que queremos alcanzar con nuestro juego. Esto va a permitir validar dichas características de una forma más precisa.

Es recomendable realizar una invitación formal al evento de playtesting. Se puede realizar una cita aclarando el lugar y horario, y opcionalmente enviar los datos por algunas de las vías de contacto proporcionadas por el participante. Se recomienda el uso de herramientas como Google Calendar para facilitar el registro de las citas. La invitación debe aclarar el propósito y la duración de la prueba. Evitar brindar información al participante sobre el juego que se quiere probar.

Preparación

El observador debe asegurar el espacio y las herramientas para poder realizar correctamente la prueba. Esto significa tener preparado el setup que utilizará el participante durante la sesión. Por ejemplo, podríamos tener preparada una notebook con el juego abierto en la pantalla de prueba, un joystick con baterías cargadas y un par de auriculares correctamente conectados a la computadora. Es importante que este setup esté correctamente probado antes de iniciar la prueba para evitar retrasos y alguna posible incomodidad en el participante.

El observador también debe preparar las herramientas designadas al registro de las observaciones realizadas durante la sesión. Por ejemplo, podría tener listo un anotador y lapicera, o una planilla lista para ser completada.

Inicio de la sesión

Al inicio de la sesión, el observador se presenta al participante y también presenta el juego. Durante esta presentación se debe indicar si se trata de una versión diferente a la que el participante había probado anteriormente. Evitar realizar aclaraciones sobre cuáles son las mejoras respecto de una versión anterior. Se recomienda indicar al participante el estimado de tiempo que llevará la sesión.

El observador indica al participante que inicia la prueba.

Registro de observaciones

Durante la sesión de prueba el observador sólo se limita a la toma de registros en base a observación visual del participante. Se recomienda realizar las observaciones en silencio y sin hablar con el participante. En caso de que el participante requiera algún tipo de

asistencia durante la prueba, dejar registro de la dificultad. Bajo ningún término se debe brindar asistencia o responder preguntas con respecto al juego. Preguntas como “¿por dónde avanzo?” y “¿lo estoy haciendo bien?” no deben ser respondidas por el observador. Se recomienda previamente a la prueba contar con una planilla que facilite y organice el registro de observaciones. A continuación se presenta un ejemplo de planilla de observación:

Versión	versión del juego que se está probando
Link	link a un ejecutable de la versión indicada
Prueba	
Fecha	fecha en la que se realiza la prueba
Participante	nombre y apellido del participante
Plataforma	plataforma en donde se está ejecutado
Controlador	
Tipo	tipo de controlador utilizado por el participante
Facilidad	grado de facilidad con el que el participante usó el controlador
Partida	
Datos propios de la partida: registro de tiempos en los que pasaron eventos importantes dentro de la partida (reseteos de partidas, avance de niveles, etc), puntos ganados, nivel alcanzado, etc.	
Observaciones	
Observaciones realizadas durante la sesión. Se recomienda separar las observaciones en categorías para facilitar el análisis: juego, control, accesibilidad, UI/UX, etc	
Puntajes	
Puntaje general que pone el observador en base a la sesión y a los objetivos planteados que se quieren validar. Se recomienda separar los puntajes en categorías para facilitar el análisis: juego, control, accesibilidad, UI/UX, etc.	

Recomendaciones

- Dejar que el participante se familiarice con los controles y los objetivos del juego. En caso de ser necesario brindar una asistencia, hacerlo respondiendo preguntas específicas y sin revelar más información de la necesaria.
- Prestar atención a las dificultades que tiene el participante durante la partida. Esto puede incluir dificultades como:
 - Navegación entre menús.

- Lectura de texto en pantalla.
- Control del juego, por ejemplo movimiento del personaje o de la cámara.
- Dificultad para entender los objetivos del juego.
- Dificultad para entender cómo avanzar o realizar una acción requerida por el juego.
- Registrar el tiempo que le toma al participante realizar ciertas acciones de interés, por ejemplo:
 - Tiempo que toma la navegación de menús.
 - Tiempo que toma en avanzar al siguiente nivel.
 - Tiempo que toma en completar el juego completo.
- Tomar notas de todo lo que se crea relevante de ser registrado
 - Dejar un espacio en la planilla para tomar notas abiertas sobre observaciones realizadas durante la sesión

Finalización

La finalización de la prueba se puede dar porque el participante terminó la prueba completa (por ejemplo terminó todos los niveles), o porque se terminó el tiempo designado para la sesión. Una vez terminada la sesión, retirar los controles al jugador y evitar responder preguntas sobre el juego, es posible que estas preguntas influyan en futuras pruebas del participante.

Terminada la sesión, agradecer al participante por el tiempo brindado para la prueba. Ofrecer un tiempo extra para realizar preguntas (no relacionadas al juego) que le puedan surgir al participante. Ofrecer también la posibilidad de brindar un feedback abierto y tomar nota. Este feedback es importante porque puede aparecer información nueva que no habíamos observado o registrado durante la sesión. Por ejemplo, el participante puede dar feedback como “no sentí cómodo el control de la cámara”, o “me pareció difícil entender qué hacer”, o “este enemigo me pareció muy rápido”.

Anexo 2: Documento de diseño de juego - Mahur: el último

Runalún

Introducción

En este documento se describen los conceptos, ideas y diseños aplicadas en el videojuego “Mahur: el último Runalún”, desarrollado en el contexto de trabajo de esta tesina. Como parte de cualquier proceso de desarrollo de software, este documento es esencial como punto de partida para el desarrollo del proyecto, y como punto de apoyo en caso de presentarse dudas sobre los requerimientos de diseño durante etapas futuras en el desarrollo del mismo.

Se trata de un documento en progreso, no representa la versión final del diseño. El diseño aquí presentado no es el de un videojuego completo, sino el de un prototipo sencillo que intenta probar el diseño de un videojuego inclusivo: mecánicas, controles, gameplay, interfaces visuales y audio. Sin embargo, la estructura de este mismo documento se puede adaptar para proyectos con una escala mayor al actual.

Concepto

Nombre

- Título: Mahur
- Subtítulo: El último Runalún

Descripción

El mundo tiene los días contados y solo una persona puede salvarlo. “Mahur: el último Runalún”, es un videojuego de aventuras y plataformas en 2D que sigue la historia de Mahur, el último sobreviviente de una antigua raza de guerreros con poderes mágicos. El jugador toma el control de Mahur mientras se dispone a explorar un mundo misterioso lleno de peligros y magia. Explora extraños escenarios, conversa con misteriosos personajes y encuentra pistas que lo ayuden a salvar el mundo.

Género

El género principal del juego es el de plataformas en 2D, influenciando principalmente el control del personaje y el diseño de los niveles. El género de plataformas es un género de videojuegos en donde el jugador controla un personaje que debe navegar el nivel corriendo,

saltando y escalando a través de obstáculos y enemigos. El objetivo primario es alcanzar el final del nivel evitando múltiples peligros y enemigos, juntando items y power-ups, y a menudo luchando con un jefe final.

En segundo lugar el juego toma del género de aventuras, el desarrollo de una historia completa con personajes, escenas, y diálogos entre personajes, con una trama que une todos los niveles del juego. Los videojuegos del género de aventura son un tipo de videojuego en donde se enfatiza la narración y la exploración. En los juegos de aventura, los jugadores toman el rol del protagonista, el cual generalmente tiene la tarea de resolver acertijos, interactuar con NPCs (personajes no jugables), y explorar el entorno para poder progresar a través de la narrativa del juego.

Algunos de los juegos que se pueden tomar como referencia son: Rayman Origins de Ubisoft, Nubarrón de Nastycloud, y Grim Fandango de LucasArts. Las siguientes imágenes son capturas de pantalla de dichos juegos.





Trasfondo

Este juego forma parte de la presente tesina, por lo tanto el alcance de este desarrollo está delimitado por el tiempo y las formas que son necesarias cumplir dentro del marco de una tesina de grado de la Facultad de Informática de la UNLP.

Plataformas de lanzamiento

Por limitaciones dentro del marco de la tesina se plantea proveer únicamente el lanzamiento de una versión para navegadores web.

Implementación técnica

El prototipo será implementado utilizando el motor Godot Engine, en su última versión disponible (3.5). Adicionalmente se utilizarán los siguientes plugins para resolver requerimientos de accesibilidad:

- Subtítulos: [Godot Subtitle Plugin](#) (desarrollado como parte del trabajo de esta tesina)
- Closed Caption: [Godot Subtitle Plugin](#)
- ScreenReader: [Godot Accessibility Plugin](#)
- Text-To-Speech: [Godot Accessibility Plugin](#)
- Input remapping: basado en el demo [GUI Input Mapping Demo](#)

Historia

La historia completa del juego consta de 4 grandes capítulos, más 1 capítulo extra en forma de epílogo a la historia principal. Cada capítulo se representa en el juego en forma de 4 niveles jugables. Por limitaciones, dentro del marco de la tesina se trabajará únicamente en el primer nivel del primer capítulo.

Capítulos

- Capítulo 1: Entrada al Bosque
- Capítulo 2: Bosque Largo
- Capítulo 3: Las Cuevas
- Capítulo 4: El Carnaval

Background

Mahur es un runalún, una raza de guerreros que podían controlar la magia de la luna. Proviene de la antigua Maralún, una ciudad creada por la diosa Lunye en conjunto con el pueblo runalún en una meseta en el norte de Anmer, la Tierra. Durante la segunda edad Maralún fue destruída y todos sus habitantes fueron obligados a vivir como nómades.

Al final de la tercera edad Mahur sigue el rastro de unas criaturas conocidas como tráucos al interior del Bosque Largo.

Capítulo 1

El primer capítulo es la introducción del jugador al mundo del juego. Se introduce al personaje protagonista y las mecánicas de juego más importantes.

La historia de este capítulo se refiere a la llegada de Mahur a la entrada del Bosque Largo. Mahur sigue el rastro de un grupo de tráucos que se mueven durante el día, al contrario de la mayoría de tráucos. En el camino se encuentra con Rumi, un personaje con el cual el jugador puede interactuar en varios puntos. Este personaje tiene como propósito dos objetivos: primero enseñar al jugador a hablar con los personajes para poder avanzar en el juego, segundo guiar al jugador a través de los primeros niveles sin romper la inmersión de la historia, y tercero dar a conocer información importante sobre la trama del juego.

Durante todo el capítulo el jugador sigue el rastro adentrándose cada vez más en el bosque, guiado por Rumi. A medida que se avanza por el bosque, se encuentran ruinas antiguas, de las cuales Mahur recolecta piedras lunares. Para poder abrir la entrada al Bosque Largo es necesario recolectar 4 piedras.

Diálogos

A continuación se detallan los diálogos del primer nivel, entre Mahur y Rumi.

1er encuentro: bienvenida (aparece)

0. La roca: ¿Quién anda por allí?
1. El hombre: ...
2. La roca: Así que no quieres hablar. No importa tu nombre, de igual forma me doy cuenta que estás perdido.
3. El hombre: No estoy perdido. Estoy persiguiendo a una jauría de tráucos. Les sigo el rastro desde hace dos días. ¿No los has visto por aquí?
4. La roca: Yo veo muchas cosas. Te veo a vos.
5. El hombre: ...
6. La roca: Está bien. Vi unos tráucos pasar por aquí en la noche. Más adelante hay un antiguo templo Chuluñe. Seguro los encuentras allí.

2do encuentro: antes del primer platforming

0. El hombre: Otra vez tú. ¿Cómo puedes estar en ambos lugares?
1. La roca: Yo estoy en todos los lugares porque soy parte de este bosque. Y este bosque es muy antiguo, ¿sabes?
2. La roca: Yo estoy aquí desde el principio. Vi nacer y crecer a este bosque. Desde antes que se levantara Endailú, las Montañas Negras.
3. El hombre: Muy interesante.
4. El hombre: Necesito encontrar a esos tráucos. ¿Los has visto o no?
5. La roca: Ya te he dicho que sí los he visto. Pero no los encontrarás despiertos bajo la luz del sol. Solo salen bajo la luz de la luna.
6. La roca: A propósito, mi nombre es **Rumi**.

3er encuentro: en la entrada del templo

0. Rumi: Si nos vamos a ver seguido al menos quiero saber tu nombre.
1. El hombre: Mi nombre es **Mahur**. Soy un runalún, descendiente de la gente de Maralún.
2. Rumi: ¿Maralún? Eso está muy lejos de aquí, ¿sabías?
3. Mahur: Maralún ya no existe, fue destruída hace mucho tiempo. Solo yo y un pequeño grupo pudimos escapar a tiempo.
4. Rumi: Entonces estás perdido.
5. Mahur: No estoy perdido.
6. Mahur: Cumpló una misión secreta que no puedo revelar.

4to encuentro: antes de las ruinas (aparece)

0. Rumi: Antes de que sigas avanzando quería decirte algo más.
1. Mahur: ¿Qué cosa?
2. Rumi: Que estás acercándote mucho a las antiguas ruinas del templo. A partir de aquí es peligroso. No me atrevo a seguirte dentro del territorio chuluñe.
3. Mahur: ...
4. Rumi: Del otro lado del templo hay una salida. Si logras escapar, nos veremos del otro lado.
5. Rumi: Buena suerte.

5to encuentro: en las ruinas del templo (aparece)

0. Rumi: Soy yo de nuevo. No me pude resistir echar un ojo dentro del templo.
1. Rumi: Muy bonito eso que tienes ahí, por cierto.
2. Mahur: Es una piedra de luna. Escuché hablar de ellas en antiguas canciones. No puedo creer que haya encontrado una en este lugar. En Maralún estaban por todos lados.
3. Rumi: ¿Y para qué sirven?
4. Mahur: ¡Para todo! Puedes usar su luz para ver en la noche, y su calor sirve de protección durante el invierno.
5. Mahur: En su superficie tiene grabadas algunas runas, aunque no consigo entenderlas. Creo que dicen algo sobre este lugar, sobre algo oscuro.
6. Rumi: ...
7. Rumi: Salgamos de aquí, esto me da mala espina.
8. Rumi: Vuelve al nivel inferior. Te espero a la salida del templo.

Personajes

Mahur

Es un descendiente de la raza Runalún, el último vivo de esta raza al momento de iniciar el juego. Los runalún son una de las cinco razas que nacieron cuando se creó el mundo, y su asentamiento se encontraba en Maralún, una ciudad creada sobre una meseta al norte de Anmer. Son descendientes de Lunye, por lo cual poseen poderes mágicos asociados a la luz de la luna. Mahur tiene la estatura de un hombre adulto, posee una barba larga y blanca que se mezcla con su pelo y le oculta gran parte de la cara. Viste ropa blanca, sucia por vivir como nómada durante los últimos cientos de años.



Tráucos

En la antigüedad eran conocidos como Khauk, y son los descendientes de Khekunu, una raza conocida como Chuluñe. Son los enemigos principales de gran parte del juego. Estas criaturas vivieron en guerra por mucho tiempo con la gente de Maralún, pero cuando fueron derrotados se refugiaron en bosques alrededor de las Montañas Negras. Son criaturas malévolas que viven en el descampado o en cuevas en las montañas. Son de baja estatura y su piel es de color oscuro. Visten con pieles y cueros de animales, y llevan adornos hechos con dientes, maderas o metales alrededor del cuello y las muñecas. Muchos utilizan una piedra afilada como arma blanca y pocos saben usar arco y flecha. Por lo general solo salen durante la noche, aunque en el juego vemos que unos pocos se mueven durante el día.



Rumi

Es un ente antiguo que existió en el bosque desde la creación del mundo. Es el único ser con estas características y no tiene un origen conocido. Fue compañero de la raza Nawan que vivía en el bosque antes de que fueran exterminados. De ellos obtuvo mucho conocimiento sobre la vida del bosque y el río. Rumi es un ente de piedra, una especie de monolito tallado que puede moverse por todo el bosque viajando por debajo de la tierra. Es quien durante el juego va a dialogar con Mahur para guiarlo y explicarle mucho sobre el lore del juego al jugador.



Diseño de niveles y ambiente

Ambiente

Todo el primer capítulo toma lugar en una zona boscosa. Por esto es que se ve mucho verde, árboles, arbustos y otra vegetación. La idea es que el ambiente de este capítulo tenga un gran contraste con el del segundo capítulo, en donde se verá un bosque más oscuro, con menos visión del cielo, y pobre de vegetación verde.

Secciones

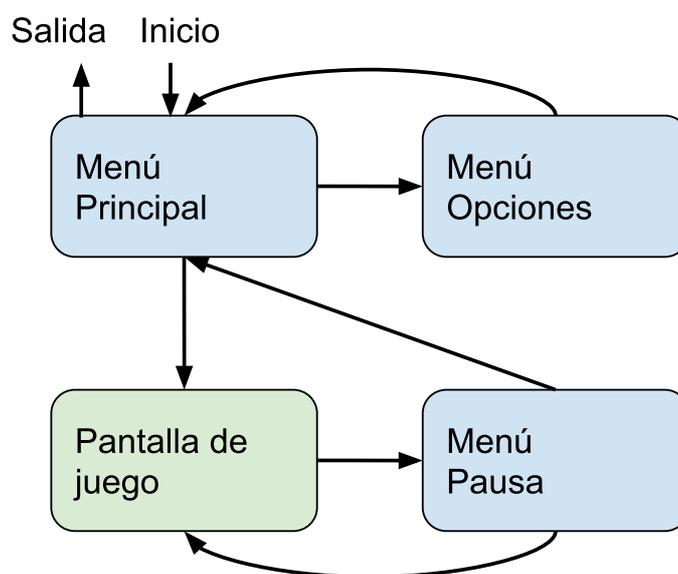
A continuación se listan las secciones del primer nivel en el Capítulo 1. Cada sección del juego se corresponde con una lista de objetivos a realizar. Durante el primer nivel encontramos objetivos simples y sin requerimientos especiales (como pueden ser encontrar una llave o activar un mecanismo).

Nombre	Objetivo
Bosque Largo	Explora el bosque
Templo Chuluñe	Encuentra una entrada al interior del templo
Runalún	Ingresa al templo
La salida	Sigue hasta la salida del templo

Durante el primer nivel, la navegación es lineal. Es decir que se puede avanzar por un único camino, y no es necesaria demasiada exploración para poder avanzar. Más adelante se puede presentar al jugador niveles en donde haya que realizar exploración en búsqueda de coleccionables, activar mecanismos para abrir caminos, etc.

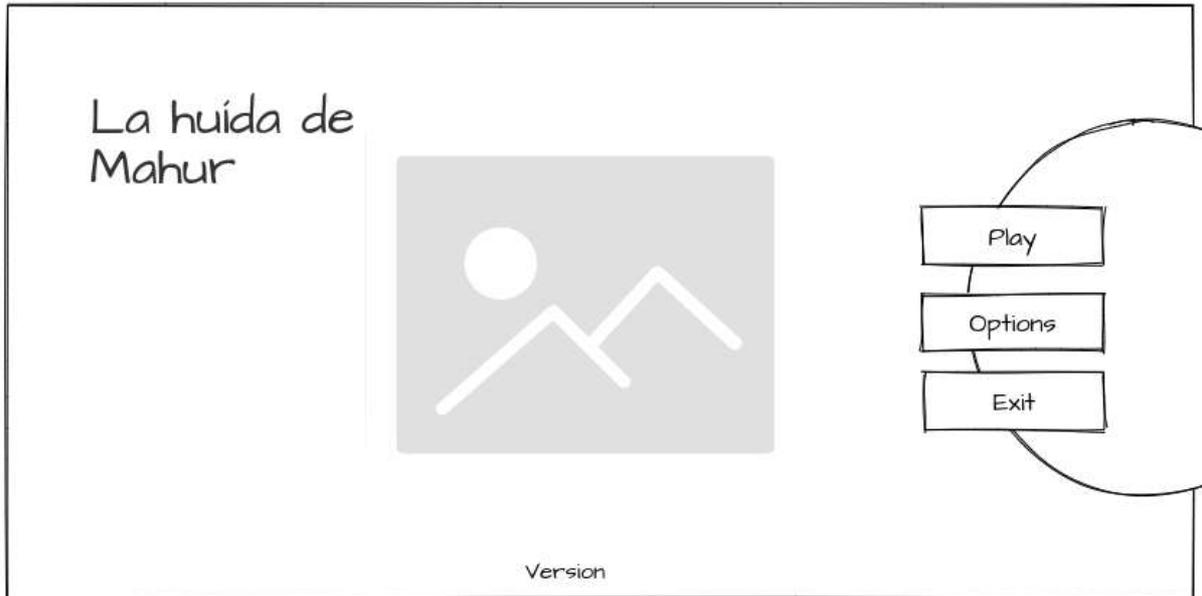
Flujo de Pantallas

A continuación se detalla el flujo propuesto para la navegación entre las pantallas del juego. Los nodos azules representan interfaces de usuario, y el nodo verde representa la pantalla de juego. Las flechas indican la dirección en la que se realiza la navegación.



Menú Principal

Es la pantalla inicial que ve el jugador cuando inicia el juego. El diseño debe ser minimalista, con el objetivo de favorecer la adaptación de la interfaz a distintos aspectos o resoluciones de pantallas. Debe incluir el título y tres botones que permitan realizar las navegaciones correspondientes. Además debe incluir un fondo animado y una pequeña descripción de la versión de aplicación.

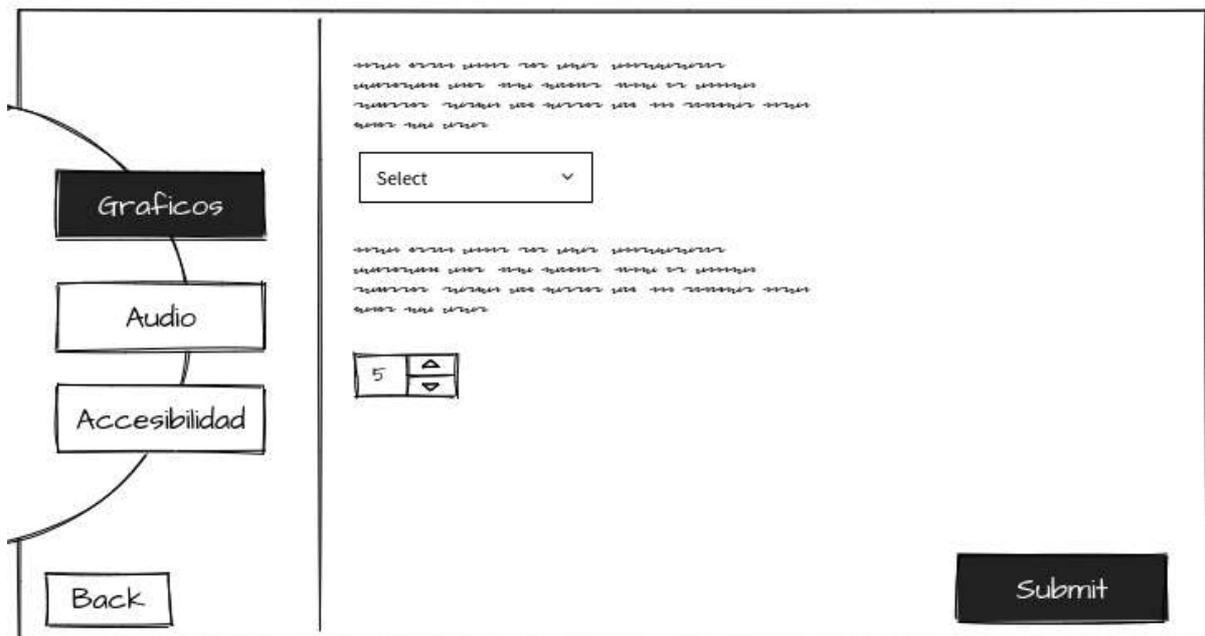


Menú Opciones

El menú de opciones debe permitir:

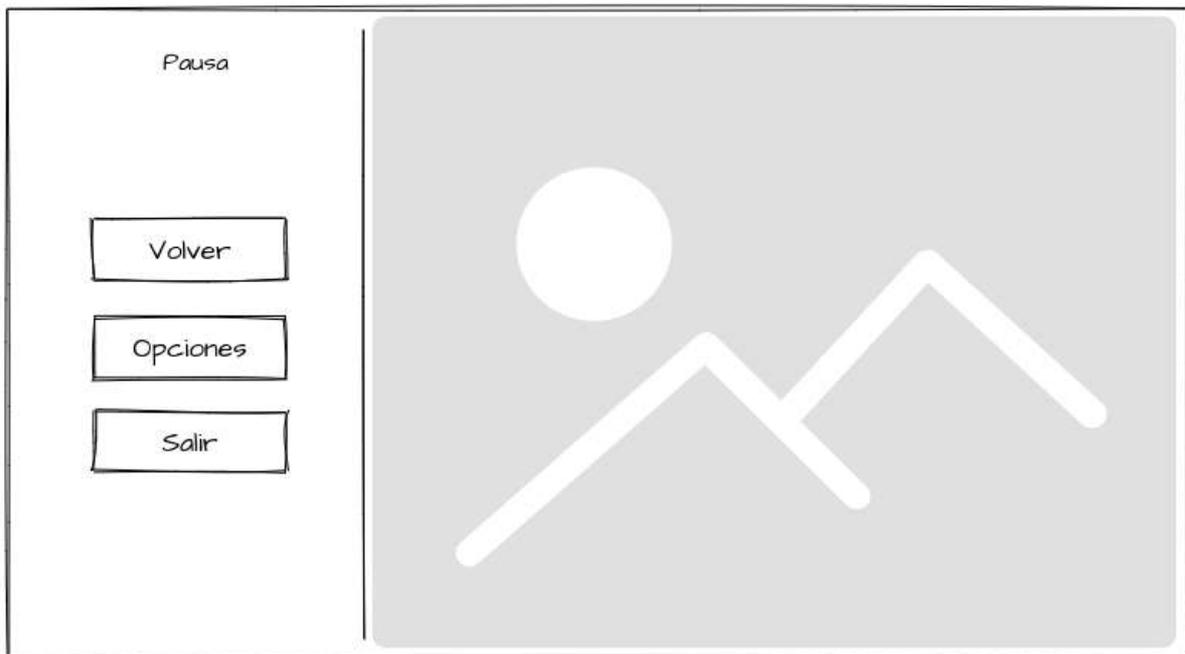
- ver la configuración actual
- cambiar una configuración
- aplicar los cambios
- persistir los cambios

Si bien todavía no se posee la lista completa de opciones con las que contará el juego, se propone dividir las opciones en categorías, donde cada categoría agrupa opciones del mismo tipo.



Menú de Pausa

Esta pantalla consiste de un panel que se despliega desde la izquierda encima de la pantalla de juego. Este panel cuenta con opciones para navegar a la pantalla de opciones, para volver al juego, o para volver al menú principal. El objetivo de ser un panel y no ocupar toda la pantalla es la de permitir al jugador poder pausar el juego sin perder el estado visual del mismo.



Gameplay

En esta sección se detallan las formas en la que el jugador interactúa con el juego.

Platforming

El movimiento se realiza al igual que otro juego de plataformas en 2D, mediante el control de la posición del jugador en la posición horizontal de la pantalla. Se permite saltar para mover al jugador en el espacio vertical.

El jugador tiene la posibilidad de realizar un doble salto. Esto se realiza al presionar la acción de salto mientras el personaje está en el aire, y permite que el platforming se sienta más natural.

Interacciones

Durante el juego se permite la interacción del jugador con varios tipos de elementos:

- personajes, para iniciar una conversación
- items, para tomarlos o usarlos

- mecanismos, para activarlos

Para esto se utiliza el botón de interacción, el cual es el mismo para realizar todas las acciones de interacción del juego. Esto hace que sea intuitivo el control del personaje.

Enemigos

Los enemigos son controlados por el juego, y pueden estar en diferentes estados de acuerdo al tipo de enemigo: dormido, despierto, en movimiento, alertado, en ataque, muerto. Los únicos enemigos que vemos durante el primer nivel son los tráucos de piedra. Se encuentran dispersos por el nivel, y atacan al jugador cuando este se acerca demasiado. La idea es que durante el primer nivel, aparezcan enemigos en diferentes ocasiones, cada una con una dificultad mayor:

- primer encuentro con un enemigo dormido, fuera del alcance del jugador
- segundo encuentro con un enemigo en solitario
- futuros encuentros con más de un enemigo al mismo tiempo

Piedras luna

Las piedras toman la funcionalidad de llaves, un coleccionable que es necesario para poder avanzar al siguiente nivel. La segunda utilidad es la de interactuar con enemigos y mecanismos que abren puertas y habilitan nuevos caminos

Hogueras

Se activan con magia y permiten abrir puertas.

Puertas pilares

Son pilares de piedra que actúan como compuertas. Cuando están cerradas impiden el paso del jugador, y cuando están abiertas permiten el paso. Se activan mediante las hogueras.

Arte

Cada capítulo tiene un ambiente totalmente único, independiente del resto del juego. A continuación se muestran algunos conceptos para comprender el estilo correspondiente a cada capítulo.

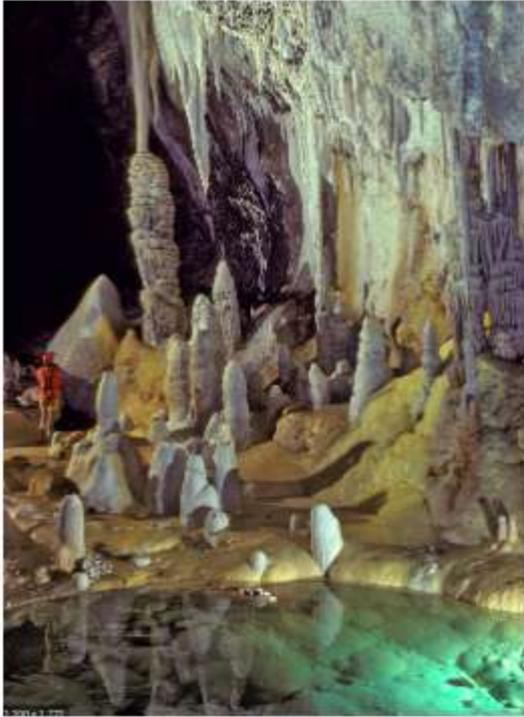
Capítulo 1: Entrada al bosque



Capítulo 2: Bosque Largo



Capítulo 3: Las cuevas



Capítulo 3: Calabozos



Sonido y música

No existe una restricción con respecto al diseño de la música más que debe evocar la sensación de soledad y algo de hostilidad del bosque durante el primer nivel, a la vez que transmite una sensación de movimiento al jugador. Esto hace que el jugador sienta que debe seguir avanzando en todo momento, y ayuda a transmitir la idea de que el bosque es un lugar vivo.

Los efectos de sonido deben resaltar por sobre el resto de las pistas de audio. Se puede pensar en agregar efectos de sonido “caricaturizados” para resaltar las acciones que realiza el jugador, como por ejemplo saltar o encender una hoguera.

Controles

Se permite el control del juego con dos tipos de controladores: joystick y teclado. A continuación se detallan las acciones dentro del juego y su correspondiente acción en los controladores.

Acción en el juego	Acción Joystick	Acción Teclado
Moverse a la izquierda	Left stick	Tecla izquierda
Moverse a la derecha	Right stick	Tecla derecha
Saltar	A button	Barra espaciadora

Interactuar	B button	Tecla Control
Avanzar subtítulos	B button	Tecla Control
Pausar juego	Start button	Tecla Escape
Mover cámara hacia la izquierda	Left analog stick	Tecla A
Mover cámara hacia la derecha	Right analog stick	Tecla D
Mover cámara hacia arriba	Up analog stick	Tecla W
Mover cámara hacia abajo	Down analog stick	Tecla S

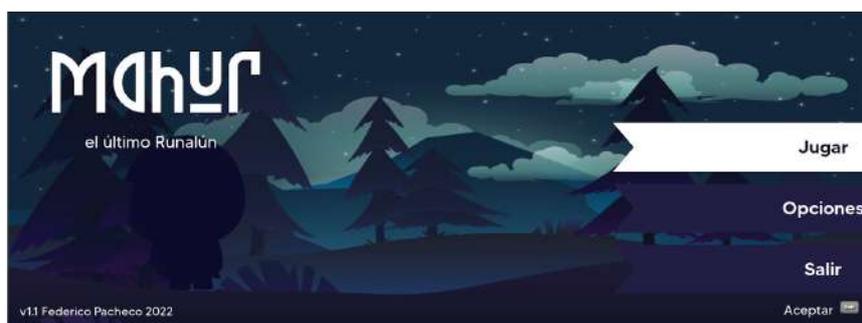
Accesibilidad

En esta sección se detallan las opciones de accesibilidad con las que cuenta el juego.

1. Posibilidad de cambiar el modo de juego entre pantalla completa y modo ventana

Esta opción permite al jugador controlar el tamaño de la pantalla que ocupa el juego. Por lo general el jugador usará el modo de pantalla completa, pero como se explicó en la tesina, esto puede obstaculizar la utilización de herramientas externas al juego en algunos casos. La implementación en Godot Engine es muy simple, con lo cual vale la pena ofrecer este tipo de opciones al juego.

Algo a tener en cuenta es que para que esto funcione correctamente el diseño de las interfaces de usuario deben funcionar correctamente bajo diferentes tamaños de pantalla. Esto incluye: paneles, contenedores, tamaños de botones, tamaño de textos, etc. Esto fue uno de los aspectos que se fueron ajustando a través de las diferentes iteraciones del prototipo construido para la tesina. A continuación un ejemplo de cómo se ve la pantalla principal en dos resoluciones y aspectos diferentes.





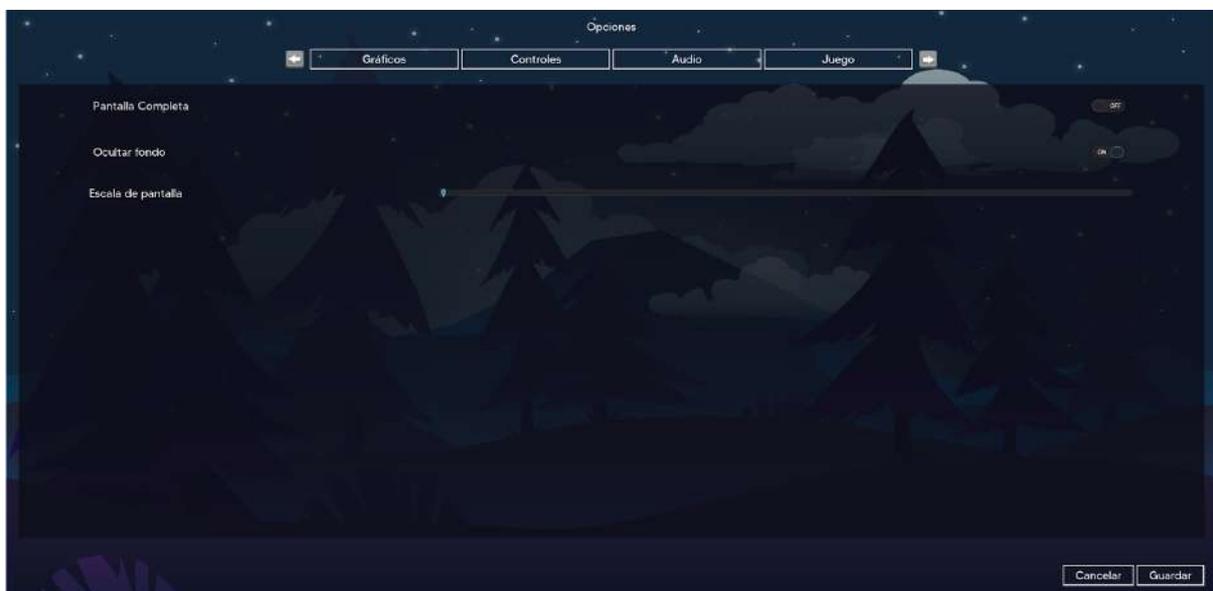
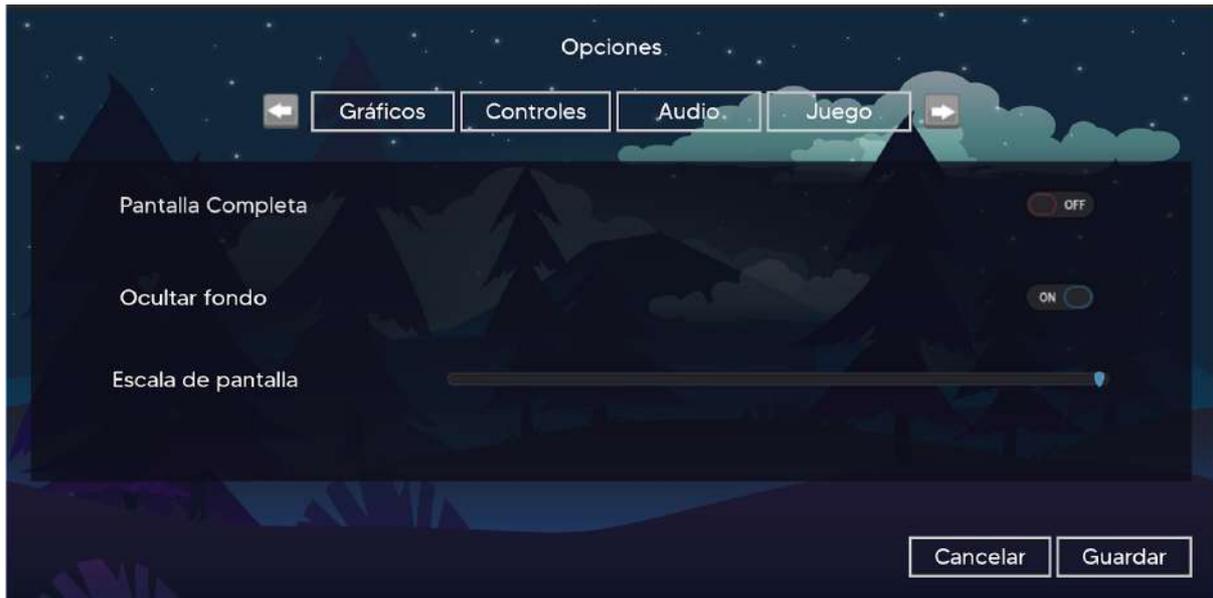
2. Posibilidad de ocultar el fondo de juego

Esta opción permite simplificar el fondo del escenario durante la partida. La tesina explica que un fondo de escenario complejo puede sobrecargar sensorialmente al jugador y provocar mareos, dolor de cabeza u otros tipos de incomodidades. La implementación en Godot Engine es simple y no requiere de un tratamiento complejo.



3. Escalado de elementos de interfaz de usuario

Esta opción aprovecha una de las features disponibles dentro del motor para escalar componentes de la UI y la expone al jugador para que pueda ajustarla a su antojo. Esto también requirió rediseños de algunos componentes para que se vean correctamente bajo todo tipo de condiciones. A continuación se muestra la pantalla de opciones gráficas bajo dos escalados completamente diferentes.

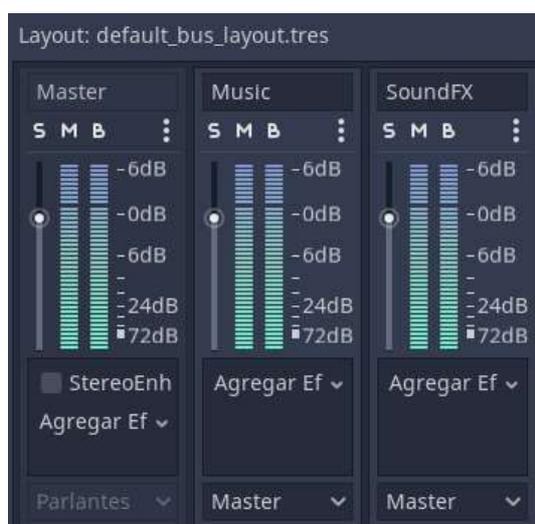


4. Remapeo de controles

Esta opción permite al jugador cambiar completamente los controles utilizados para controlar al juego. Permite al jugador configurar las acciones para conectarlas al input del joystick o del teclado. Esto se realizó gracias al plugin "GUI Input Mapping", el cual permite detectar dinámicamente la utilización de nuevos controladores durante el juego, cambiando el esquema de controles en el momento, sin necesidad de reiniciar el juego. El plugin además provee un conjunto de íconos para representar los botones y teclas asociados a cada controlador. Esta feature se aprovechó en las iteraciones más tardías para poder integrarlas a los tutoriales de controles.

5. Ajustar el volumen de la música y efectos de sonido

Esta opción permite al jugador controlar el volumen de la música y de los efectos de sonido de manera independiente. Esta feature de accesibilidad forma parte de los requerimientos desde el comienzo del desarrollo, lo cual facilitó su implementación. Cuando se creó el proyecto en Godot se agregaron varios canales (o buses) de audio, preparando las bases para el diseño del audio. A medida que se agregaron efectos de sonido y música, se asociaron a alguno de estos canales, organizando los audios en alguna de estas categorías (música o efecto de sonido), facilitando la implementación de esta feature. A continuación se muestran los tres buses principales de audio.



6. Modo de audio mono-canal

Esta feature de accesibilidad permite unificar todos los canales de audio en uno solo, para dirigirlo tanto al altavoz derecho como el izquierdo, efectivamente perdiendo información direccional. Esto hace que un efecto de sonido que solo se escucha por el altavoz izquierdo, luego de prender el modo de audio mono-canal se escuche por ambos altavoces. Esta feature surgió luego de varias iteraciones y sesiones de playtesting. Una de las razones para esto es que Godot 3.5 no dispone de una implementación de esta feature de manera nativa, por lo que el desarrollador debe probar soluciones alternativas. Una de estas soluciones, la cual fue utilizada para implementar esta feature, requiere la utilización de un efecto de sonido (StereoEnhance) que fue diseñada con otro objetivo.

7. Subtítulos descriptivos

Los subtítulos descriptivos se implementaron utilizando el nodo LiveCaption que provee el plugin "Godot Subtitles". Esto permite que los efectos de sonido tengan su correspondiente representación mediante subtítulos.



8. Soporte Text-To-Speech

El soporte TTS permite escuchar mediante una voz digitalizada tanto los diálogos de los personajes como los textos dentro de los elementos interactivos como los botones. Luego de varios ajustes e iteraciones en el diseño de este soporte, se lograron realizar los ajustes necesarios para que el usuario pudiera navegar los menús únicamente mediante el soporte TTS. Gracias al playtesting y a las iteraciones realizadas sobre el diseño del soporte TTS se pudo lograr que el jugador pudiera participar también de la historia del juego, algo que es muy importante en un juego de aventura.

Una nota importante sobre el diseño de este tipo de feature de accesibilidad para este juego es que está orientada únicamente a asistir al jugador con dificultades visuales. El diseño del juego se ajusta al marco de la tesina, la cual está orientada a las dificultades auditivas. Es por esto que el diseño de los niveles no considera a los jugadores con pérdida total de la visión. Esto requeriría un rediseño total de los niveles y profundizar en el diseño de los efectos de sonido para mejorar la información contextual que recibe el jugador.

9. Asistencia: tutorial de controles

Una de las primeras dificultades detectadas durante el playtesting fueron referidas a los controles. Se detectó que para los jugadores no eran intuitivos los controles, y sobre todo que el “doble salto” fue complicado de aprender.

El primer nivel del juego se rediseñó en base a esto, para poder introducir los controles de una manera paulatina. Además, el rediseño se apoyó en una serie de tutoriales, los cuales ayudan al jugador a entender los controles de una manera sencilla y directa. El tutorial consiste en mensajes que se integran directamente dentro del nivel, indicando con un breve

mensaje el objetivo de cada botón. Adicionalmente, se utilizó el plugin “GUI Input Mapping” para poder mostrar una representación visual del botón, lo que termina de explicar la acción requerida al jugador. Algunos jugadores indicaron que los tutoriales no les parecieron necesarios, por lo que son totalmente opcionales y se pueden activar o desactivar desde el menú de opciones. A continuación se muestra un ejemplo del primer tutorial que aparece en el juego, el cual enseña a saltar al jugador.



10. Asistencia: mostrar lista de objetivos

Esta fue la última feature de accesibilidad surgida de las sesiones de playtesting. Consiste en brindar información contextual al jugador de acuerdo al momento en el que se encuentre dentro del juego. Durante las sesiones de playtesting se detectaron situaciones en donde los jugadores no sabían qué era lo que tenían que hacer a continuación. Por estas razones se agregaron objetivos que se activan a medida que el juego recorre el nivel. Estos objetivos se muestran como mensajes que se pueden leer desde el menú de pausa, y apunta a orientar a los jugadores durante la partida. Este feature fue innecesario para algunos jugadores, por lo que se eligió brindar la posibilidad al jugador de activarlos desde el menú de opciones.