



UNIVERSIDAD
NACIONAL
DE LA PLATA

FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

TÍTULO: Accesibilidad en documentos para personas con discapacidad visual y ceguera en CILSA.

AUTORES: David Leonardo Huertas Godoy.

DIRECTOR/A: Mg. Ivana Harari.

CODIRECTOR/A: Mg. Ana Paola Amadeo.

ASESOR/A PROFESIONAL:

CARRERA: Licenciatura en Sistemas

Resumen

La tecnología se encuentra cada vez más inmersa en nuestras vidas, incluyendo el ámbito del aprendizaje. En este contexto, la accesibilidad desempeña un papel fundamental al facilitar el acceso a la información y proporcionar recursos de calidad, especialmente para las personas con discapacidad visual y ceguera.

En esta Tesina, se presenta una herramienta web que tiene como objetivo abordar de manera automatizada los desafíos de accesibilidad en documentos de lectura digitales. Esta herramienta busca brindar soluciones prácticas para mejorar la accesibilidad y garantizar que todas las personas puedan acceder y utilizar dichos documentos de manera efectiva.

Palabras Clave

Prototipo, Herramienta, Accesibilidad Web, Accesibilidad en documentos, DOCX, PDF, Python, Discapacidad, Discapacidad visual y ceguera, Documentos digitales, Conversión, Automático.

Conclusiones

En esta tesina se llevó a cabo un estudio sobre la accesibilidad de documentos en un entorno educativo, específicamente en el contexto de CILSA ONG. Como resultado, se identificaron oportunidades para mejorar automáticamente el material de lectura. A partir del estudio de la accesibilidad en documentos digitales, se desarrolló una herramienta que facilita la accesibilidad de documentos para docentes o profesores. Esto demuestra que dicha herramienta constituye una opción prometedora para alcanzar dicho objetivo.

Trabajos Realizados

Se ha desarrollado un prototipo de una herramienta web que automatiza la accesibilización de documentos digitales. Para lograrlo, se llevó a cabo una investigación y análisis exhaustivo de los componentes, estructura e impacto de los documentos en la accesibilidad de personas con discapacidad visual y ceguera, especialmente en el ámbito educativo. Se recopilieron requisitos a través de cuestionarios dirigidos a capacitadores de CILSA ONG, los cuales sirvieron de base para el desarrollo del prototipo. Además, se realizaron análisis de accesibilidad tanto automáticos como manuales, involucrando a potenciales usuarios, y los resultados obtenidos confirmaron la satisfactoria usabilidad de la herramienta.

Trabajos Futuros

Permitir la detección y gestión de una mayor cantidad de componentes dentro de un documento. Lograr el manejo de otros formatos de documentos. Mejorar la interfaz y la lógica de programación a partir de los análisis realizados. Continuar relevando requerimientos de usuarios para una nueva versión del prototipo. Investigar tecnologías asociadas al reconocimiento de imágenes y OCR.

Fecha de la presentación: JUNIO 2023

Agradecimientos

A mi familia, por el acompañamiento durante el proceso de aprendizaje universitario.

A las personas con las que he compartido horas de estudio, ayudándome a recorrer este camino de manera más amena.

A CILSA ONG, y en particular al centro de La Plata, a Analía por brindarme su apoyo para el desarrollo de esta tesina y al personal de la Organización, alguno de ellos siendo parte de este documento.

A Nahuel González, por su libro, y a los autores del libro de Facultad de Informática sobre Accesibilidad web por su material de lectura, ambos fueron un gran apoyo para el enriquecimiento de conocimientos sobre la temática de la tesis.

A Joana Sarmiento por su amabilidad y tiempo brindado.

A la Facultad de Informática y a los y las docentes, por su vocación y metodología de enseñanza para la transferencia de conocimientos.

A Dios, por permitirme llegar a este momento donde se cierra una etapa en mi vida.

Contenido

Agradecimientos.....	1
Capítulo 1: Introducción.....	4
1.1 Motivación	4
1.2 Objetivo.....	7
1.2.1 Objetivo general	7
1.2.2 Objetivos específicos	7
1.3 Organización de la tesina	7
Capítulo 2: La Inclusión y la Discapacidad	9
2.1 ¿Qué se entiende por discapacidad?	9
2.2 Educación inclusiva.....	12
Capítulo 3: Accesibilidad y tecnologías	16
3.1 Accesibilidad y Accesibilidad web	16
3.2 Accesibilidad y diseño inclusivo	20
3.3 Productos de apoyo y accesibilidad	23
3.4 Accesibilidad y documentos digitales.....	27
3.5 Herramientas para accesibilizar documentos.....	36
Capítulo 4: Construcción del prototipo.....	45
4.1 Requerimientos del prototipo	45
4.2 Desarrollo web del prototipo.....	48
4.3 Lógica de conversión del prototipo.....	69
Capítulo 5: Funcionamiento de AcceDocx.....	73
Capítulo 6: Análisis sobre AcceDocx	82
6.1 Análisis de accesibilidad de AcceDocx.....	82
6.2 Análisis desde la perspectiva del Capacitador	90
6.3 Análisis con persona con discapacidad visual	95
Capítulo 7: Conclusiones y Trabajos futuros	97
7.1 Conclusiones	97
7.2 Trabajos futuros	99
Referencias.....	100
Anexo 1: Inicialización de AcceDocx	105
Anexo 2: Componentes detectables con potencial implementación.....	106
Anexo 3: Encuesta SUS sobre la herramienta.	108
Anexo 4: Utilización de AcceDocx.....	111
Paso 1: Ingreso al sitio web.	111

Paso 2: Listado de documentos	111
Paso 3: Carga de documento	112
Paso 3: Proceso de Accesibilizar	114
Paso 4: Descarga en DOCX.....	116
Paso 5: Conversión de PDF a DOCX.....	117
Paso 6: Eliminar un documento	118
Anexo 5: Resultados de las encuestas SUS.....	119

Capítulo 1: Introducción

1.1 Motivación

La discapacidad se entiende como el resultado de la interacción de una persona con sus posibles limitaciones, y las barreras del entorno que impiden su participación plena y efectiva en la sociedad, en igualdad de condiciones que los demás. En este sentido, observamos que la tecnología se encuentra cada vez más inmersa en nuestras vidas, y puede ser fundamental para garantizar autonomía, como sucede en el caso de un proceso de aprendizaje ¹.

Se entiende que la accesibilidad supone facilitar el acceso a la información y en el caso de un ambiente de enseñanza, los recursos brindados de manera digital al lector necesitan ser accesibles. En el caso del grupo de personas con discapacidad visual y ceguera ², consumir el material de lectura supone tener un recurso de calidad que pueda servirle a un software o producto de apoyo, como los lectores de pantallas, para su acceso y lectura por medio de una voz y lograr que se exprese oralmente el contenido del documento.

Con el objetivo de brindar documentos de lectura accesibles para personas con discapacidad visual y ceguera, se llevará adelante un caso de estudio en el Programa de Oportunidades de Formación en el Centro de Inclusión Libre y Solidario de Argentina (CILSA), teniendo como propósito promover el acceso y la capacitación en el uso de las Tecnologías de la Información y la Comunicación, y la preparación para la inclusión

¹ Casi 900.000 personas tienen dificultades visuales en Argentina, según el Indec. Los datos surgen del Estudio Nacional sobre el Perfil de las Personas con Discapacidad, implementado en localidades urbanas de 5.000 y más habitantes del país durante abril y mayo de 2018.

² En Argentina, la educación superior de personas con discapacidad se encuentra mencionada en la Constitución Nacional (1994, Arts. 14 y 75) y en la Constitución de la Ciudad Autónoma de Buenos Aires (1995, Art. 42).

laboral de poblaciones en situación de vulnerabilidad social ³.

Dentro del área de formación existen documentos digitales, utilizados para el proceso de enseñanza brindado por la Organización a personas con y sin discapacidad, donde se encuentran incluidos las personas con discapacidad visual y ceguera. Estos documentos, son de distintos orígenes y tiempos, y no tienen una uniformidad concreta en cuanto al formato, estilo y otros criterios de accesibilidad. El personal capacitador con rol de profesor, son quienes dictan las actividades de enseñanza-aprendizaje, teniendo para ello, estos recursos digitales.

A partir de las directrices establecidas en cuanto al enfoque inclusivo y como proceso de mejora de la Organización, se realizaron charlas introductorias sobre el concepto de accesibilidad hacia los profesores y profesoras en CILSA. Estas charlas estuvieron a cargo de un personal experto en la temática.

En este contexto, se dictaminó la actividad de revisión de algunos documentos con el fin de detectar y en lo posible corregir la conformación de esos documentos utilizando los conceptos y criterios de accesibilidad. La actividad era organizada en grupos de 2 personas con asignaciones de algún documento, donde cada profesor o profesora debía ocuparse, dentro de su horario laboral, a ingresar a una computadora y utilizar un procesador de texto. Esta actividad duró un periodo inicial de 6 meses, donde había encuentros en una fecha estipulada para poder revisar avances cada 20 o 30 días.

A partir de la organización de esta actividad surgieron varias cuestiones pertenecientes a lo subyacente a una actividad para el personal capacitador dentro de su ambiente de trabajo, su cronograma de actividades personales principales y horarios laborales.

Surgen cuestiones como:

- contratiempos para finalizar esta actividad por cronograma de actividades de enseñanza personales,
- contratiempos para trabajar junto con el personal compañero de grupo,
- revisión de conceptos de accesibilidad a partir de dudas que surgen,
- tiempo no eficiente en el uso del software procesador de texto Word.

³ Centros de Formación CILSA. Centros tecnológicos comunitarios donde se dictan cursos que tienen por objetivos promover el acceso y la capacitación en el uso de las Tecnologías de la Información y la Comunicación, y la preparación para la inclusión laboral de poblaciones en situación de vulnerabilidad social.

Este procedimiento de accesibilizar documentos de lectura digitales se realizó durante el año 2022, los documentos quedaron con un nivel de accesibilidad alto y el personal encargado de la organización de este proceso resuelve que han sido satisfactorios los resultados obtenidos.

Considerando los resultados y analizando cuestiones básicas de accesibilidad en documentos digitales, se puede llegar a describir las implementaciones que se repiten para todos los documentos, estos criterios son básicamente sobre: fuente de texto y su tamaño, interlineado, existencia de encabezados para títulos, descripción de imágenes, entre otros posibles criterios sobre elementos del contenido de un documento. Es decir, que puede pensar en que estos criterios iniciales se podrían automatizar si existiera una herramienta accesible que permita trabajar sobre un documento.

1.2 Objetivo

1.2.1 Objetivo general

Contribuir en el área de la accesibilidad, a través de una herramienta tecnológica en línea, que permita accesibilizar documentos de lectura, observando particularmente la situación en CILSA.

1.2.2 Objetivos específicos

Investigar normativas de accesibilidad sobre recursos digitales educativos, herramientas de generación y validación.

Investigar sobre la accesibilidad en documentos, con especial atención en el impacto que tiene sobre grupo de personas con discapacidad visual y ceguera, en un ambiente de enseñanza.

Investigar sobre software y herramientas existentes para la gestión de documentos y el procesamiento de textos, y las características de accesibilidad que poseen.

Investigar la accesibilidad en documentos respecto de los aspectos disciplinares y componentes que pueda contener tales como textos, imágenes, y distintos elementos y estructuras.

Diseñar y desarrollar una herramienta que permita, en forma automática, solucionar problemas de accesibilidad en documentos de lectura, atendiendo las necesidades de las personas con discapacidad visual y ceguera.

1.3 Organización de la tesina

La presente tesina se estructura de la siguiente manera: el Capítulo 1 describe los aspectos motivacionales y objetivos del documento. En el Capítulo 2 se aborda el tema de la discapacidad y la inclusión de las personas en situación de discapacidad en un ambiente de formación. En el Capítulo 3 se aborda el concepto de accesibilidad junto con las tecnologías que acompañan. En el Capítulo 4 se describe la construcción del desarrollo

propuesto y sus bases tecnológicas. En el Capítulo 5 se describen las funcionalidades del prototipo desarrollado y el alcance. En el Capítulo 6 se analiza el prototipo sobre 3 perspectivas y se lo evalúa. En el Capítulo 7 se presentan las conclusiones, a partir del trabajo desarrollado, y los trabajos futuros que pueden continuar como etapa posterior del proyecto u otro trabajo relacionado.

Capítulo 2: La Inclusión y la Discapacidad

En este capítulo, se hará un revisionismo sobre los conceptos y terminologías que involucran la discapacidad, conceptos históricos, la inclusión, leyes y hasta escenarios que se presentan en la actualidad en un ambiente de enseñanza.

2.1 ¿Qué se entiende por discapacidad?

La convención Internacional sobre los Derechos de las Personas con Discapacidad, aprobado en el año 2006 por la Asamblea General de las Naciones Unidas [1], y en Argentina en 2008 a través de la Ley 26.378, en el Artículo 1° se cita [2]:

“Las personas con discapacidad incluyen a aquellas que tengan deficiencias físicas, mentales, intelectuales o sensoriales a largo plazo que, al interactuar con diversas barreras, puedan impedir su participación plena y efectiva en la sociedad, en igualdad de condiciones con las demás.”

En este sentido, podemos entender a la discapacidad como el resultado de la interacción de una persona con sus posibles limitaciones y las barreras del entorno que impiden su participación plena y efectiva en la sociedad, en igualdad de condiciones que los demás.

Dentro del grupo de personas con discapacidad, existen personas con discapacidad visual y ciega, y como la presente tesina se trabaja con este grupo de personas, a continuación, nos adentramos a definir las características de esta discapacidad.

Discapacidad visual y ciega

La Organización Mundial de la Salud (OMS) define a una persona con baja visión como aquella que presenta agudeza visual (AV) inferior a 6/18 (0,3 decimal) en el mejor ojo con la mejor corrección posible, o un campo visual menor o igual a 10 grados desde el punto de fijación, con visión útil para desarrollar una tarea [3].

Según la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE) en su undécima revisión (CIE-11) publicado en inglés en 2018 [4], clasifica a la deficiencia visual en los siguientes 2 grupos:

- Deterioro de la visión distante: aquí tenemos distintos niveles como, Leve, Moderado, Grave y Ceguera.
- Deterioro de la visión cercana: aquí tenemos la agudeza visual.

Así mismo, según la OMS, aproximadamente 1300 millones de personas en el mundo sufren ceguera o algún tipo de discapacidad visual [3].

De manera que, las personas con discapacidad visual son aquellas que con la mejor corrección posible podrían ver o distinguir, aunque con una gran dificultad, algunos objetos a una corta distancia. En la mejor de las condiciones, algunas de ellas pueden leer la letra impresa cuando ésta es de suficiente tamaño y calidad, pero por lo general, de forma más lenta, con un considerable esfuerzo y utilizando ayudas especiales. En otras circunstancias, una discapacidad visual puede manifestarse como una dificultad para la identificación de objetos que se hallan delante, a los lados, por encima o por debajo de los ojos de acuerdo al tipo de afección [5]. El término ceguera se aplica a aquellas personas que no ven nada en absoluto o solamente tienen una ligera percepción de la luz. Entendiendo las características mencionadas anteriormente, los términos como “baja visión” y la ceguera representan conjuntamente el total de casos de discapacidad visual.

Conceptos históricos.

Un paradigma o cosmovisión es una estructura de pensamiento, un modelo para pensar que sirve para reflexionar sobre distintas cosas del mundo. En Ciencias Sociales, el término de cosmovisión se utiliza para describir el conjunto de experiencias, creencias y valores que afectan la forma en que un individuo percibe la realidad y la forma en que responde a esa percepción.

La discapacidad es tan antigua como la humanidad, pero su concepto comienza a presentarse luego de finalizar la II Guerra Mundial, ya que los países que participaron en la misma contaban con muchos ciudadanos sobrevivientes que coexistían con sus cuerpos destrozados. En este momento socio-histórico, los países en su totalidad se vieron con la necesidad de enfrentar esta situación [6].

A lo largo del tiempo, se fueron construyendo distintas maneras de pensar la discapacidad. La podríamos categorizar en cuatro grandes paradigmas, asociados a su vez a diferentes maneras de nombrarla. Estos paradigmas o cosmovisiones se superponen y coexisten:

- Paradigma de exclusión: Representaba el conjunto de actitudes sostenidas por una sociedad antigua que asignó a la exclusión con las personas con discapacidad, colocándolas en un lugar de rechazo, dependencia, sometimiento y ocultamiento. Conceptos como: “discapacitado”, “impedido” o “minusválido” eran utilizados. Era entender como que esa persona no estaba capacitada para hacer cosas “normales”.
- Paradigma de segregación: Se caracteriza por la reclusión de las personas con discapacidad, con lo cual debían ser apartadas del resto de la sociedad en un espacio preparado especialmente para ellas. Conceptos como patología y fantasías de contagios, como castigo eran parte de este paradigma.
- Paradigma médico-rehabilitador: Un modelo de integración/normalización, tomando formas de actividades de rehabilitación para curar y “normalizar” los cuerpos y mentes “desviados” de los que era la “normalidad”. Introducción de la voz médica, teniendo como objetivo la integración de las personas con discapacidad a la sociedad intentando “curarlas” de su situación. Términos como “capacidades diferentes” o “necesidades espaciales” eran utilizados de manera positiva, teniendo la intención de integrar a esas personas con discapacidades, sosteniendo que el “problema” está colocado en la persona.
- Paradigma de inclusión: En el pensamiento de inclusión, la discapacidad se la considera dentro de un contexto, donde depende de múltiples factores y se configura a través de distintos aspectos socioambientales. Se incluye el término diversidad como parte de la condición humana y, por ende, con la necesidad de crear un mundo accesible para todas las personas.

Teniendo en cuenta estos paradigmas, podemos concluir que la discapacidad no puede pensarse de manera fija o rígida, sino que supone una construcción social y dinámica, variando según la situación histórica, política, económica y social que atraviesa una cultura. En función de cómo una sociedad comprenda el mundo se determinará el modo de comprender el concepto de discapacidad.

La discapacidad, incluida la discapacidad visual, es una condición que puede afectar la vida de las personas de diferentes maneras, y es importante reconocer que la discapacidad no define a una persona ni limita su valía o potencial, sino que cada persona independientemente de su discapacidad, posee habilidades y capacidades únicas.

El grupo de personas con discapacidad visual y ceguera, enfrentarán dificultades en la interacción con el mundo. Sin embargo, serán capaces de enfrentarse a los desafíos, adaptándose y superándolos mediante sus otros sentidos, y con la tecnología, la cual presentará herramientas que permitirán ser un apoyo. Las herramientas tecnológicas serán presentadas más adelante.

2.2 Educación inclusiva

A continuación, se estudia la inclusión de la persona con discapacidad en el ámbito educativo, sus oportunidades de éxito y los desafíos existentes.

Legislación sobre discapacidad en Argentina.

La Ley N° 24.901 es una legislación argentina que se promulgó en el año 1997 y se denomina "Sistema de Prestaciones Básicas en Habilitación y Rehabilitación Integral a favor de las Personas con Discapacidad" [7]. Tiene como objetivo principal promover la igualdad de oportunidades y el pleno ejercicio de los derechos de las personas con discapacidad, asegurando su inclusión social, educativa y laboral. La Ley N° 26.206 [8] brinda a las personas con discapacidad educación para su integración y el desarrollo de sus posibilidades y derechos. La Resolución CFE N 311/16 [9] detalla que los Estados Partes asegurarán que:

- Las personas con discapacidad no queden excluidas del sistema general de educación por motivos de discapacidad, y que los niños y las niñas con discapacidad no queden excluidos de la enseñanza primaria gratuita y obligatoria ni de la enseñanza secundaria por motivos de discapacidad;

- Las personas con discapacidad puedan acceder a una educación primaria y secundaria inclusiva, de calidad y gratuita, en igualdad de condiciones con las demás, en la comunidad en que vivan;
- Se hagan ajustes razonables en función de las necesidades individuales;
- Se preste el apoyo necesario a las personas con discapacidad, en el marco del sistema general de educación, para facilitar su formación efectiva;
- Se faciliten medidas de apoyo personalizadas y efectivas en entornos que fomenten al máximo el desarrollo académico y social, de conformidad con el objetivo de la plena inclusión”.

La Ley N° 26.061 [10] establecerá los procedimientos y recursos correspondientes para identificar tempranamente las necesidades educativas derivadas de la discapacidad o de trastornos en el desarrollo, con el objeto de darles la atención interdisciplinaria y educativa para lograr su inclusión desde el nivel inicial. En este sentido, estas leyes protegen los derechos humanos básicos de una persona con discapacidad, promoviendo la inclusión e igualdad de oportunidades en todos los ámbitos de la vida, incluyendo la educación, un área fundamental para el desarrollo del ser humano.

En el caso puntual de la educación superior, la Ley Nacional de Educación Superior N° 24.521 de 1995 [11] establece que las instituciones de educación superior deben promover la igualdad de oportunidades para el acceso y la permanencia de las personas con discapacidad. Por otro lado, la Ley N° 25.689 del 2002 [12] de Promoción de la Educación Superior para Personas con Discapacidad establece medidas específicas para garantizar la accesibilidad y la igualdad de oportunidades en la educación superior. Como el propósito de la presente tesis se enfoca sobre el ámbito de enseñanza, a continuación, se describe alguno de estos conceptos enfocados en la educación.

Igualdad de oportunidades.

El derecho a la educación ha sido una lucha y conquista de la humanidad, y poder ejercerlo en igualdad de condiciones es fundamental en la realización de las oportunidades de vida de todos los seres humanos.

Sin embargo, aún hoy en día, no todas las personas tienen el mismo acceso a la educación, y en el caso de las personas con discapacidad el panorama resulta particularmente complejo.

Tener una discapacidad implica muchas veces estar apartado de la sociedad ya que en ellas existen muchas desventajas que se imponen socialmente, como puede ser el no poseer recursos adecuados. Es por esto que las personas con discapacidad conforman un colectivo vulnerable, conviviendo con distintas barreras, por ejemplo, físicas o sociales. La igualdad de oportunidades y la integración plena de personas con discapacidad son actividades que en la actualidad están siendo temas con mayor importancia, como el acceso a recursos de información para el aprendizaje, aunque aún resulta complejo por diversos aspectos que deben de complementarse y asociarse para poder lograr avances significativos.

Barreras para el aprendizaje y la participación

Cuando se menciona a la educación y la discapacidad, podemos incluir el estudio que realizaron Tony Booth y Mel Ainscow (2002) [13], sobre el enfoque de la inclusión en este aspecto, donde mencionan lo siguiente:

“El enfoque de barreras para el aprendizaje y la participación se basa en el modelo social de la discapacidad, a partir del cual se entiende que las limitaciones o restricciones enfrentadas por las personas con discapacidad surgen en contextos sociales particulares”. (Booth y Ainscow,2002).

El enfoque de barreras para el aprendizaje y la participación, desarrollado por Booth y Ainscow, permite profundizar sobre las necesidades de aquellos estudiantes que se consideran en situación de desventaja o con necesidades educativas especiales, exponiendo la importancia de su interacción con el contexto donde se incluyen las instituciones educativas. En este sentido, en un ambiente de enseñanza es fundamental fomentar entornos inclusivos que reconozcan y valoren las habilidades y fortalezas individuales, y que brinden oportunidades equitativas para que todas las personas puedan aprender y desarrollarse plenamente.

De acuerdo a los conceptos analizados en este capítulo, podemos reflexionar acerca de la inclusión educativa y el aprendizaje. Con el objetivo de promover inclusión educativa la Educación Superior como derecho que debe garantizar el estado, deberían involucrar cambios en el marco regulatorio hasta programas de accesibilidad de las universidades nacionales.

En algunos casos, y especialmente en aquellos en que se dispone de un sistema de otorgamiento de becas específicas por discapacidad, se cuenta con algunos datos, pero estos no resultan significativos sobre el total de la población estudiantil caracterizada ⁴. Algunos ajustes razonables que deberían ofrecer las organizaciones que brindan enseñanza son las grabaciones de clases y el aporte de tecnologías existentes y emergentes. La adecuación de recursos para un entorno de aprendizaje y la capacitación sobre cómo hacerlos, es una actividad que lleva un periodo de tiempo en adaptar materiales y recursos, aun así, esta actividad debe crecer. Así mismo, con el avance tecnológico, existen sistemas o entornos web donde los materiales que se brindan en un proceso de enseñanza permanecen de manera conjunta y con una organización en la misma, siendo una herramienta muy útil, lo cual es algo para celebrar y promover. Finalizando estos puntos reflexivos, podemos concluir que, pese a las acciones tomadas para garantizar el acceso al sistema educativo de las personas en situación de discapacidad, esto sigue siendo complejo en su dimensión práctica.

⁴ Censo de estudiantes universitarios de la Universidad de Buenos Aires de 2011, arroja sobre un total de 274.667 alumnos se encontraron 1869 con algún grado de discapacidad (0,68%). De estos solo 154 presentaban discapacidad visual grave o ceguera. En la facultad de Medicina, se encontraron solo 3 alumnos con discapacidad visual. En 2015 se realizó un nuevo censo, pero no hubo información pública sobre él.

Capítulo 3: Accesibilidad y tecnologías

En este capítulo, se hará una descripción de los conceptos y terminologías involucradas en la temática de la accesibilidad junto con tecnologías o productos de apoyo para poder adaptar una computadora para su uso efectivo, superar barreras y participar plenamente en actividades digitales. Además, se dará a conocer las herramientas tecnológicas que existen para trabajar sobre la accesibilidad sobre recursos digitales como lo son los documentos digitales.

3.1 Accesibilidad y Accesibilidad web

A continuación, se estudia la accesibilidad en función del uso en la web y el enfoque de diseño inclusivo.

La accesibilidad se define según la Ley N° 22.431 en 1981, capítulo IV, artículo 20 [14] como:

“La posibilidad de las personas con movilidad reducida de gozar de las adecuadas condiciones de seguridad y autonomía como elemento primordial para el desarrollo de las actividades de la vida diaria sin restricciones derivadas del ámbito físico urbano, arquitectónico o del transporte para su integración y equiparación de oportunidades.”

De esta manera se describe la obligación de garantizar la accesibilidad en los espacios públicos y los distintos servicios para poder asegurar la igualdad de oportunidades para las personas con discapacidad.

La accesibilidad web se refiere a la práctica de diseñar y desarrollar sitios web de manera que sean accesibles y utilizables para todas las personas. Es decir que todas las personas, más allá de ciertas limitaciones, puedan entender, percibir, navegar e interactuar con la web, tanto en su estructura como en sus contenidos. En este sentido, la accesibilidad web implica eliminar barreras y obstáculos que impidan el acceso a la información en la web, a través del cumplimiento de reglas. Estas reglas, principalmente, han sido establecidas por la W3C (WWW Web Wide Web Consortium), un consorcio internacional dedicado a establecer estándares y pautas

técnicas para el desarrollo y evolución de la World Wide Web (WWW), el cual fue lanzado por en 1999 por Tim Berners, donde surgió la primera guía sobre accesibilidad web, denominada WCAG 1.0 (Web Content Accessibility Guide), con una serie de 65 recomendaciones sobre accesibilidad dirigidas a los desarrolladores web y su sitio oficial de referencia es w3.org. Las mismas nos orientan sobre cómo una persona con discapacidad y los adultos mayores ven la web actual y cómo debería ser diseñada una página web y sus contenidos para que todos podamos acceder y comprender.

Recomendaciones de la W3C

La Iniciativa de Accesibilidad Web (WAI) [15] del W3C desarrolla especificaciones técnicas, pautas y recursos de apoyo que describen las soluciones de accesibilidad y son consideradas estándares internacionales para la accesibilidad web.

La WAI emite directrices estandarizadas y reconocidas a nivel internacional para crear contenidos y páginas web más accesibles, agrupadas en tres áreas:

- Accesibilidad del contenido web (WCAG)
- Accesibilidad de las herramientas de autor (ATAG)
- Accesibilidad de los agentes de usuario (UAAG)

En este caso, la implementación de las recomendaciones WCAG permitirá a más personas a participar y beneficiarse de la Web.

Los principios de accesibilidad son los conceptos fundamentales que guían el desarrollo de las WCAG. Cada principio aborda una característica de la accesibilidad y establece objetivos generales que deben cumplirse para que un contenido sea accesible. Estos principios son los siguientes:

1. Perceptible: la información debe presentarse de diferentes maneras más allá del formato visual (por ejemplo: en Braille, con diferentes tamaños de texto, ofreciendo la conversión de texto a voz).
2. Operable: la funcionalidad de una aplicación tiene que poder utilizarse a través de diferentes dispositivos (teclado, mouse, comandos de voz, pantalla táctil).
3. Comprensible: los contenidos y la información provista en las páginas web deben ser inteligibles, contar con ayudas y disposición predecible de la información.

4. Robusto: el contenido debe poder interpretarse de manera confiable mediante una amplia variedad de navegadores, reproductores multimedia y tecnologías de asistencia.

Para poder evaluar y medir la accesibilidad de un contenido se necesita de ciertos criterios.

Los criterios de conformidad son los estándares específicos que se utilizan para evaluar si un sitio web o una aplicación cumple con los principios de accesibilidad establecidos en las WCAG. Los criterios de conformidad se agrupan en niveles [16]: A, AA y AAA, y cada nivel tiene sus propios criterios de éxito que deben cumplirse para alcanzar ese nivel de conformidad.

Guía de accesibilidad WCAG 2

Las WCAG definen cómo crear contenido web más accesible para las personas con discapacidad. Seguir estas definiciones o pautas permite crear un contenido más accesible para un mayor número de personas con discapacidad, incluyendo ceguera y baja visión. Estas pautas también ayudan a que el nivel de usabilidad del contenido se incremente.

Iniciando desde la WCAG 2.0 [17] publicada en 2008 se convirtió en estándar ISO en el 2012 [18], el sucesor fue la WCAG 2.1 [19] convertidas en recomendación el 5 de junio de 2018. Todos los requisitos (criterios de conformidad) de la versión 2.0 están incluidos en 2.1, siendo exactamente iguales. Pero esta actualización de la WCAG proporcionó 17 criterios de conformidad adicionales que hay que tener en cuenta [20].

A continuación, se los menciona:

- Para nivel A: Gestos del puntero, Cancelación del puntero, Etiqueta en el nombre, Actuación por movimiento, Atajos de teclado.
- Para nivel AA: Orientación de la pantalla, Identificación del propósito del campo, Reajuste de elementos, Contraste no textual, Espaciado del texto, Contenido en hover o focus, Mensaje de estado.
- Para nivel AAA: Identificación del propósito, Animaciones desde interacciones, Tamaño del área de interacción, Mecanismo de entrada concurrentes, Límites de tiempo.

Actualmente a mayo 2023, existe la WCAG 2.2 [21], y se encuentra en versión candidata a recomendación. El objetivo de la WCAG 2.2 es continuar el trabajo de la WCAG 2.1, de manera que son una nueva ampliación de criterios, de modo que si se

cumple con lo que recomienda la guía WCAG 2.2 se cumplirá también con la WCAG 2.1. La WCAG 2.1 aporta pautas relacionadas con el contenido móvil, la accesibilidad para personas con discapacidades visuales y sobre necesidades de personas con discapacidades cognitivas y de aprendizaje. Dentro de estas pautas que fueron agregadas sobre la persona con discapacidad visual, se pueden resaltar: “Espaciado de texto (AA)”, “Contenido en el *hover* o en el foco (AA)”, “Contraste de los elementos que no son texto (AA)”, “Etiqueta en el nombre (A)” y “Mensajes de estado (AA)”. Una de las novedades de los cambios es que se elimina el criterio “Parsing” de nivel A y se refería a la capacidad del contenido web de ser analizado por los agentes de usuario sin generar errores de sintaxis o de estructura, lo cual se consideró redundante y se eliminó, ya que la corrección del análisis sintáctico es ampliamente abordada por las tecnologías web modernas y los navegadores. De esta manera, dentro de las actualizaciones de la WCAG 2.1, principalmente se añadieron 9 criterios: 2 de nivel A, 4 de nivel AA y 3 de nivel AAA. Además, el criterio Foco Visible pasa del nivel AA al nivel A, de manera que aumenta su exigencia.

Habiendo presentado a las guías de recomendaciones WCAG en sus distintas versiones, se deja en claro que existe actividad relacionada a la accesibilidad de manera vigente en la actualidad, de hecho, se está trabajando sobre la creación de WCAG en su versión 3.0.

Teniendo en cuenta tanto WCAG 2.0 como WCAG 2.1, ambos son relevantes y válidas para utilizarse en una evaluación de sitio web o aplicación, aunque la recomendación es trabajar hacia la conformidad con la WCAG 2.1, ya que aborda un conjunto más amplio de necesidades de accesibilidad y considera las tecnologías y los dispositivos actuales.

Respecto a las pautas o guías de recomendaciones WCAG vale destacar que la misma es de suma importancia para el cumplimiento de la Ley N° 26.653 [22] de Accesibilidad de la Información en las Páginas Web de sitios gubernamentales aprobada en 2010. A partir de esto, mediante la resolución 69/2011 el 27 de junio de 2011 se determinó la adherencia de la Argentina como miembro del consorcio W3C, y al ONTI (Oficina Nacional de Tecnologías de Información de la Secretaría de Gobierno Digital e Innovación Tecnológica) como el organismo encargado de reglamentarla. De esta manera, tenemos garantía y ampliación el acceso a la información pública para las personas con discapacidad, y así también para personas de la tercera edad, siendo

grupos de personas vulnerables en la Web local. En el proceso de reglamentación de esta ley, se encuentran entre otros, el Decreto 355/2013 del 5 de abril de 2013 y la Disposición 2/2014 del 20 de agosto de 2014 de la ONTI, donde la primera hace pública su reglamentación y vigencia, y la segunda presenta las aprobaciones de normas y requisitos de accesibilidad, compuesta de 25 criterios de nivel A, las cuales se encuentran en la guía WCAG 2.0. Con el avance tecnológico, fueron surgiendo herramientas y funcionalidades disponibles en soluciones y plataformas para el desarrollo de sitios web, en este contexto, el 27 de septiembre de 2019, la ONTI pública la Disposición 6/2019, donde se eleva el nivel de exigencia de las pautas de accesibilidad a cumplir en los sitios web, incorporando criterios de conformidad de nivel AA de la WCAG 2.0. De esta manera, se establecieron un total de 38 criterios de conformidad a cumplir como mínimo, para desarrollar páginas accesibles nacionales, encontrándose las mismas en el Anexo II de la Disposición 6/2019 [23].

De acuerdo a las reglamentaciones sobre la accesibilidad web en el ámbito nacional que sucedieron y la necesidad de acceso de la información digital, la sanción de la Ley N° 26.653 y sus adhesiones provinciales, representaron un avance significativo en la promoción y protección de los derechos y la dignidad de las personas con discapacidad pretendiendo eliminar las barreras que impiden su participación plena y efectiva en la sociedad, en igualdad de condiciones con los demás.

3.2 Accesibilidad y diseño inclusivo

Frente al objetivo de la accesibilidad web de permitir que todas las personas, independientemente de sus habilidades o discapacidades, puedan acceder y utilizar los entornos digitales, aparece el concepto de Diseño universal.

La Tecnología Accesible supone, por un lado, una herramienta facilitadora del acceso a la información y comunicación de personas con y sin discapacidad. Es decir, cuando esta herramienta es pensada desde un Diseño Universal, facilita el acceso a todos, sin importar su condición.

Con respecto a este diseño, la Convención Internacional de los Derechos de las Personas con Discapacidad promueve el diseño universal, que se entiende como:

El diseño universal es el diseño de productos, entornos, programas y servicios que puedan utilizar todas las personas, en la mayor medida posible, sin necesidad de adaptación ni diseño especializado. El “diseño universal” no excluirá las ayudas técnicas para grupos particulares de personas con discapacidad, cuando se necesiten.

Con el objetivo de lograr un diseño para todas las personas, la Convención acepta el concepto de ajustes razonables, adaptaciones necesarias y adecuadas cuando se requieran en un caso particular. Podríamos utilizar el concepto de “diseño inclusivo” para referirnos al conjunto de recursos necesarios que permitan el acceso garantizado de una persona con cierta discapacidad sobre un producto tecnológico. Para lograr un diseño de calidad, se tienen 7 principios del diseño inclusivo [24] que se deberían cumplir:

- 1) Equivalencia de uso: el diseño debe proporcionar un grado de alcance equitativo de los usuarios con el contenido.
- 2) Flexibilidad: el diseño debe permitir la adecuación de una página hacia las preferencias del usuario.
- 3) Simple e intuitivo: el diseño debe permitir el fácil aprendizaje, rápido acceso y comprensión del contenido.
- 4) Percepción del contenido: el diseño debe permitir al usuario percibir el contenido de manera independiente a sus capacidades sensoriales.
- 5) Tolerancia a errores: debiendo minimizar las acciones accidentales o casual que puedan tener consecuencias adversas o no deseadas.
- 6) Mínimo esfuerzo físico y mental: lograr que el usuario, de una manera sencilla y clara, encuentre, utilice y evalúe los resultados de las funciones existentes.
- 7) Espacio y tamaño para el acercamiento y uso: proporcionar el tamaño y espacio adecuado para utilizar el producto sin problemas, independientemente de las características del usuario o su movilidad.

Estos principios nos ayudarán, junto a las pautas de la WCAG, a crear mejores experiencias para todas las personas, considerando la accesibilidad y la inclusión en un proceso de ingeniería de software aplicados al desarrollo web.

En este sentido, se puede pensar que el diseño inclusivo sirve para equiparar el acceso a la información y comunicación, derribando las barreras con las que se encuentra una persona con discapacidad en su cotidianidad.

En el proceso de diseño se hace uso de ciertas facilidades que ayudan a retirar los obstáculos o barreras del entorno, consiguiendo que las personas con discapacidad realicen la misma acción que pudiera llevar a cabo una persona sin ningún tipo de discapacidad. Estas facilidades son llamadas ayudas técnicas o productos de apoyos, se mencionan más adelante en este documento. Del mismo modo, podemos lograr desarrollar un buen diseño o mejorarlo a partir de las interacciones que tiene el usuario sobre una página. En este sentido, un término utilizado es el Diseño Centrado en el Usuario.

Diseño Centrado en el Usuario (DCU)

El diseño centrado en el usuario es un enfoque de diseño que pone al usuario como actor central de un proceso de desarrollo, involucrándose de manera activa y continua durante todas las etapas del proceso de ingeniería de un software. Esto incluye la investigación de usuarios, la generación de ideas, la creación de prototipos, las pruebas de usabilidad y la retroalimentación del usuario. El objetivo es crear soluciones que sean intuitivas, eficientes y satisfactorias para los usuarios finales. Dentro de las recomendaciones en las normas ISO, se han establecido como estándar este proceso de ingeniería para la interfaz del usuario como estándar ISO 13407 (*Human-Centred Design Process*) [25] y su reemplazo, la 9241- parte 210 [26]. Las mismas definen un proceso general para la inclusión de actividades centradas en el humano a través de un ciclo de vida de desarrollo. Dentro de ella se describe al DCU como “una actividad multidisciplinar, que incorpora factores humanos y conocimientos ergonómicos con el objetivo de mejorar la efectividad y eficiencia, las condiciones de trabajo y contrarrestar los posibles efectos adversos de su uso, en relación con la salud, la seguridad y el rendimiento”. Algunos de los aspectos clave del diseño centrado en el usuario incluyen:

- Investigación de usuarios.
- Creación de perfiles de usuarios.
- Iteración y prototipado.
- Pruebas de usabilidad.

- Consideración de la experiencia del usuario.

Adoptando el enfoque centrado en el usuario se pueden desarrollar tecnologías que se ajusten mejor a las necesidades y preferencias de los usuarios, y al mismo tiempo mejorando la usabilidad, la satisfacción del usuario y el éxito del producto o servicio final. La necesidad de trabajar la accesibilidad desde las primeras etapas de la ingeniería de software e incorporar a personas con discapacidad durante los procesos de evaluación a partir de métodos y técnicas correspondientes, es fundamental para garantizar que los productos y servicios digitales sean accesibles para todas las personas. Las razones para abordar la accesibilidad desde el inicio del proceso son:

- Costos y esfuerzos: donde incorporar la accesibilidad desde un principio puede ayudar a evitar costos y esfuerzos adicionales en el futuro.
- Experiencia del usuario mejorada: incorporando la accesibilidad mejora la satisfacción general del usuario.
- Alcance mayor: los productos y servicios llegan a una mayor audiencia, tanto a personas con discapacidad como a personas mayores, personas con baja visibilidad, usuarios con falta de recursos, entre otros.
- Cumplimiento de normas y leyes: para cumplimentar las leyes y regulaciones exigidas desde un inicio.
- Responsabilidad social: demostrar compromiso con el proceso de inclusión e igualdad de oportunidades.

3.3 Productos de apoyo y accesibilidad

Son muchos los avances en accesibilidad que se han logrado gracias a la tecnología, y justamente han nacido productos como tecnologías en formato de software permitiendo a la persona con discapacidad visual y ciega, poder consumir de algún recurso digital en una computadora. A continuación, se hará mención a los productos de apoyo más importantes, incluyendo al lector de pantallas.

Productos de apoyo

Producto de Apoyo se define en la norma UNE-EN ISO 9999 del año 2007 [27] como:

“Cualquier dispositivos, equipo, instrumentos y software, fabricado especialmente o disponible en el mercado, utilizado por o para personas con discapacidad y/o adultos mayores destinado a: Facilitar la participación; Proteger, apoyar, entrenar, medir o sustituir funciones/estructuras corporales y actividades; o Prevenir deficiencias, limitaciones en la actividad o restricciones en la participación.”

Los productos de apoyo, también conocidos como ayudas técnicas, son aquellos dispositivos o recursos (programas, aplicaciones, etc.) que permiten disminuir las barreras a las que se enfrenta cualquier persona. Una de esas barreras existentes se centra en el acceso a la información y al uso de recursos tecnológicos, por ejemplo, para utilizar una computadora y acceder a una página web. En este sentido, los productos de apoyo cumplen una labor fundamental para tener acceso a los dispositivos que se utilizan hoy en día.

Entre los productos de apoyo que funcionan como dispositivos de interacción de entrada, se tienen:

- Switch: Es un dispositivo que se conecta a una computadora o a un dispositivo (juguete adaptado) y permite realizar acciones de selección o la de activación (en general, en la computadora el clic con el botón izquierdo del mouse). Existen distintos tipos de switch de acuerdo al tipo de activación (por mano, por pie, por pera o mentón, por soplido, con la cabeza, por tacto). Un switch permite la escritura a través de un teclado en pantalla o con la interacción con programas de causa-efecto o de estimulación mediado por la acción de una persona.
- Mouses adaptados: Se trata de un dispositivo que cumple con las mismas funciones de un mouse convencional, pero posee un acceso modificado. Existen distintos tipos de mouses adaptados: por botones, por palanca, por tacto, ocular, por movimiento de cabeza.
- Teclados adaptados: Son los dispositivos que cumplen con las mismas funciones de un teclado convencional, pero posee un acceso modificado como agrandar el tamaño de las teclas, colores distinguibles sobre ellos o con la colocación de algún producto de superficie rígida que se coloca sobre el teclado. Existen diferentes tipos de teclados adaptados: sobre-teclado o cobertor para teclado, pegatinas de contraste, teclado expandido, entre otros. Un ejemplo es el software Teclado en Pantalla en una computadora con sistema operativo Windows.

- Joysticks y palancas de mando: Dispositivos de control que permiten el movimiento y la navegación en la pantalla, útiles para personas con discapacidad motora.
- Reconocimiento de voz: los programas de reconocimiento de voz convierten la voz hablada en texto en la pantalla, permitiendo hacer control de la computadora y escribir mediante comandos de voz. Por ejemplo, el software Narrador en una computadora con Windows, TalkBack, o Voice Assistant en teléfonos inteligentes o Tablet.

Entre los productos de apoyo que funcionan como dispositivos de interacción de salida, se tienen:

- Lectores de pantalla: Son programas que convierten el texto sobre la pantalla, en una voz sintetizada, ayudando a personas con discapacidad visual a acceder a la información en la computadora.
- Magnificadores de pantalla: Dispositivos que permiten aumentar un sector de la pantalla para facilitar su visualización. Por ejemplo, la lupa en Windows, Zoom en Mac OS.
- Impresoras braille: Dispositivos que convierten el texto en la pantalla en Braille, permitiendo ser leídas mediante el tacto.
- Sistemas de comunicación aumentativa y alternativa: Incluye dispositivos y software que permiten a personas con discapacidad comunicarse mediante símbolos, imágenes o voz sintetizada.

Además, hay de entrada y salida, como los de línea braille, un dispositivo electrónico de entrada y salida que combina en un sólo dispositivo una línea braille, que permite la salida de contenido en código braille, y un teclado braille.

Con el avance tecnológico, las alternativas van aumentando, por lo que hoy en día las personas se pueden encontrar con OrCam MyEye ⁵, un dispositivo revolucionario activado por voz y se puede sujetar a casi cualquier tipo de gafas. Utiliza una cámara de alta resolución y un procesador interno para capturar imágenes y procesar la

⁵ La página oficial de OrCam MyEye es www.orcam.com

información visual en tiempo real. El dispositivo utiliza reconocimiento óptico de caracteres (OCR) para leer y convertir texto impreso en voz. Puede leer una variedad de fuentes de texto, como libros, periódicos, señales, etiquetas de productos y pantallas digitales. Además, también puede reconocer y memorizar rostros, objetos y productos, teniendo la posibilidad de ser controlado mediante gestos o movimientos de la cabeza, pudiendo ser configurado y personalizado mediante una aplicación móvil. De manera que promete estar diseñado para adaptarse a las necesidades individuales de cada usuario.

La elección de los dispositivos adecuados dependerá de la necesidad de la persona, de manera que pueda lograr utilizar una computadora o un dispositivo móvil y poder acceder a recursos como documentos digitales o una página web.

Como el objetivo de la presente tesis se enfoca en la accesibilidad sobre recursos tecnológicos del grupo de personas con discapacidad visual y ceguera, a continuación, se hace una descripción de los lectores de pantalla.

Lector de pantalla

Un lector de pantalla es un software o programa que lee el texto en voz alta a través de una voz sintetizada o lo convierte en braille, es decir, que puede describir los eventos e información que aparecen en la pantalla, a medida que utilizamos la computadora o el dispositivo móvil. El usuario puede controlar el lector de pantalla mediante comandos de teclado o gestos táctiles, dependiendo del dispositivo utilizado.

Para su uso sobre Internet, los lectores de pantalla tienen la capacidad de interpretar el código HTML [28] de una página web para proporcionar acceso y comprensión del contenido a las personas con discapacidad visual. Cuando un lector de pantalla interactúa con una página web, analiza el código HTML y extrae información como encabezados, párrafos, listas, enlaces, tablas, formularios y otros elementos. Además de interpretar el HTML, los lectores de pantalla también pueden reconocer atributos y etiquetas específicas que ayudan a mejorar la accesibilidad, como el atributo “alt” para descripciones de imágenes o las etiquetas semánticas que indican la función o el propósito de ciertos elementos. Puede leer el contenido en tiempo real sobre documentos y páginas web.

Algunos ejemplos de aplicaciones o softwares disponibles en la actualidad son:

- Sobre una computadora: el “Narrador” de Windows, Voice Over en Mac OS, el software gratuito “NVDA” [29] y la aplicación de pago “Jaws” [30] ambos sobre Windows. Por otro lado, tenemos Orca, un lector de pantalla de software libre flexible y extensible, utilizado mediante una voz sintetizada y/o braille, este lector logra acceder a aplicaciones y toolkits.
- Sobre dispositivos móviles: TalkBack en Android, VoiceOver en iOS, Voice Dream Reader en iOS.

Para una actividad de enseñanza, tareas sobre una computadora, manejar documentos o para navegación por internet, el grupo de personas con discapacidad visual y ciega utiliza lectores de pantalla sobre una computadora, siendo generalmente sobre el sistema operativo Windows, esta situación ocurre, por ejemplo, en el espacio de enseñanza dentro de CILSA ONG. De manera que se puede identificar dos lectores de pantalla que sobresalen del resto por su popularidad, el NVDA y el JAWS [31].

3.4 Accesibilidad y documentos digitales

Si pensamos en los distintos archivos que existen en un ambiente de enseñanza, y enfocándonos principalmente en el entorno de enseñanza de CILSA ONG, podríamos hacer hincapié en los documentos digitales de estilo Word [32] y PDF [33]. Estos formatos de documentos conforman un estimado del 90% del material de lectura que se encuentra disponible en la Organización. Con el fin de gestionar documentos y garantizar su accesibilidad, es fundamental considerar los principales elementos que conforman un documento, en particular en el formato *DOCX*. Este formato no solo posibilita la creación y edición de los documentos, sino que también desempeña un papel crucial en el ámbito educativo de CILSA ONG al ser el formato fundamental utilizado para la creación del material de enseñanza. Es importante destacar que, dentro de este entorno, los participantes, incluidas las personas con discapacidad visual y ciega, muestran preferencia hacia estos formatos. Los documentos en formato *DOCX* ofrecen a los capacitadores la posibilidad de gestionar el contenido y estructurar cada sección de manera que se brinde un recurso accesible. Posteriormente, es posible generar un archivo *PDF* para cada recurso creado en formato *DOCX*. De esta manera, se garantiza la disponibilidad de materiales accesibles y bien estructurados para su uso en

el ámbito educativo. Para poder manejar el contenido de los documentos, debemos conocer los software o herramientas que existen en la actualidad para poder manipularlos.

Recomendaciones de accesibilidad para documentos

Los tipos de documentos digitales de texto y lectura que predominan en la Web son los formatos del paquete Office de Microsoft (Word, Excel, PowerPoint), especialmente el formato *DOCX*, y por otro lado el *PDF* de Adobe [34]. Cada uno de estos formatos requiere una serie de acciones para que la información que contengan sea accesible. A continuación, se resumen los puntos de accesibilidad a revisar en los 2 tipos de documentos:

1- Documentos Word.

- Ortografía.
- Vista y zoom.
- Contraste de fuentes.
- Comentarios.
- Tablas.
- Imágenes.
- Tamaño de fuente y contraste cromático.
- Formularios accesibles.

2- Documentos PDF.

- Formatos de encabezados y estilos.
- Contrastes y el uso de colores exclusivos.
- La existencia de imágenes e imágenes de texto.
- Tablas y listas.
- Orden de lectura.
- Cuestionarios accesibles.
- Si se crea con herramientas automáticas, comprobar la accesibilidad del documento de origen.
- Una vez creado el *PDF*, hacer la comprobación de accesibilidad y guardarlo como texto para comprobar el resultado final.

En particular, como resultante de un análisis de criterios de accesibilidad, se puede acceder a distintas guías de recomendaciones para documentos de estilo *Word* y *PDF*,

desde las recomendaciones de la WCAG 2.0 para textos planos [35] y para documentos *PDF* [36], como también un documento de acceso público desde 2017 en idioma español, titulado “Recomendaciones para la redacción de documentos digitales de texto accesibles” [37], dispuesto por la ONTI e incluyen recomendaciones para documentos *DOCX*, *ODT* (archivos de texto del formato de código abierto *OpenDocument*.) y *PDF*, basadas en la WCAG 2.0. En este sentido la ISO 14289-1:2012 [38], también conocida como *PDF/UA* (*PDF/Universal Accessibility*), establece los requisitos técnicos para crear y verificar la accesibilidad de los documentos *PDF*. Esta norma se basa en las pautas de accesibilidad establecidas por el WCAG 2.0 y proporciona directrices específicas para garantizar la accesibilidad en documentos *PDF*.

De manera de síntesis podemos mencionar que las recomendaciones sobre documentos son:

- Poseer una estructura: Utilizar títulos, párrafos, para poder facilitar la navegación y comprensión del documento.
- Utilizar un tamaño de fuente legible: Tamaño de al menos de 12 puntos para garantizar una buena legibilidad.
- Fuentes claras y legibles: Tales como Arial, Helvetica o Verdana, evitar fuentes decorativas o estilizadas.
- Espaciado adecuado: Espaciado adecuado entre líneas y párrafos para mejorar la legibilidad.
- Evitar uso excesivo de efectos visuales: Limitar el uso de animaciones parpadeantes o en movimiento.
- Colores y contrastes adecuados: Evitar combinaciones de colores que dificulten la legibilidad.
- Descripciones para imágenes: Incluir un texto descriptivo para las imágenes.
- Enlaces descriptivos: Indicar claramente el destino de esos enlaces.
- Tablas accesibles: Encabezados claros y organización lógica de filas y columnas.
- Evitar bloqueo de contenido: Evitar elementos que no se puedan seleccionar, como imágenes de texto o escaneos.
- Utilizar marcadores y listas: Si existen contenido con viñetas o numeración, utilizar las funciones que permitan estructurarlas.

- Verificar la accesibilidad: Antes de finalizar el documento, realizar pruebas de accesibilidad utilizando herramientas de verificación o solicitando retroalimentación de personas con discapacidad.

A partir del estudio de esta serie de recomendaciones, se podrá tener una serie de conceptos puntuales que se van a considerar a la hora de iniciar la elaboración de un documento digital accesible mediante procesadores de texto tales como son Microsoft Word, Documentos de Google y LibreOffice Writer, y considerar las funciones de accesibilidad que estos ofrecen para garantizar el cumplimiento de las mismas.

Siendo los archivos *DOCX* y *PDF* los formatos utilizados para el propósito de la presente tesis, se hará un análisis del formato *DOCX* y se mencionarán características de los *PDF*.

Cuestiones técnicas sobre archivos *DOCX*

Los archivos con esta extensión son conocidos por ser el formato de documentos de Microsoft Word. Fue lanzado en el mercado desde el 2007 a partir del paquete de aplicaciones de Microsoft Office 2007.

Este tipo de documento tenía una nueva estructura a los que existían anteriormente. Pararía de ser un simple archivo binario plano y sencillo, a ser un archivo binario combinado con XML (*Extensible Markup Language*), un lenguaje de marcado que define un conjunto de reglas para la codificación de documentos donde su sitio oficial de referencia es xml.com. De esta manera surge el estándar Open XML, o también llamado Office Open XML (OOXML) [39], un formato de archivo abierto y basado en XML desarrollado por Microsoft. Este estándar fue introducido por primera vez en 2007 con el lanzamiento de Microsoft Office 2007 como el formato de archivo predeterminado para los programas de Office, como Word, Excel y PowerPoint. Open XML utiliza XML como base para describir la estructura y el contenido de los archivos, y lo utiliza a partir de archivos comprimidos en formato ZIP, permitiendo que sean archivos más compactos y eficientes en cuanto al almacenamiento y la transferencia de datos. Este estándar, es un formato extensible que permite la creación de extensiones personalizadas y la incorporación de metadatos adicionales en los archivos, brindando flexibilidad para adaptar los archivos a necesidades específicas y permite la integración con otros sistemas y flujos de trabajo.

De esta manera, un archivo *DOCX* puede analizarse a partir del contenido que puede descomprimirse del mismo, como se visualiza en la siguiente figura (Fig. 1):

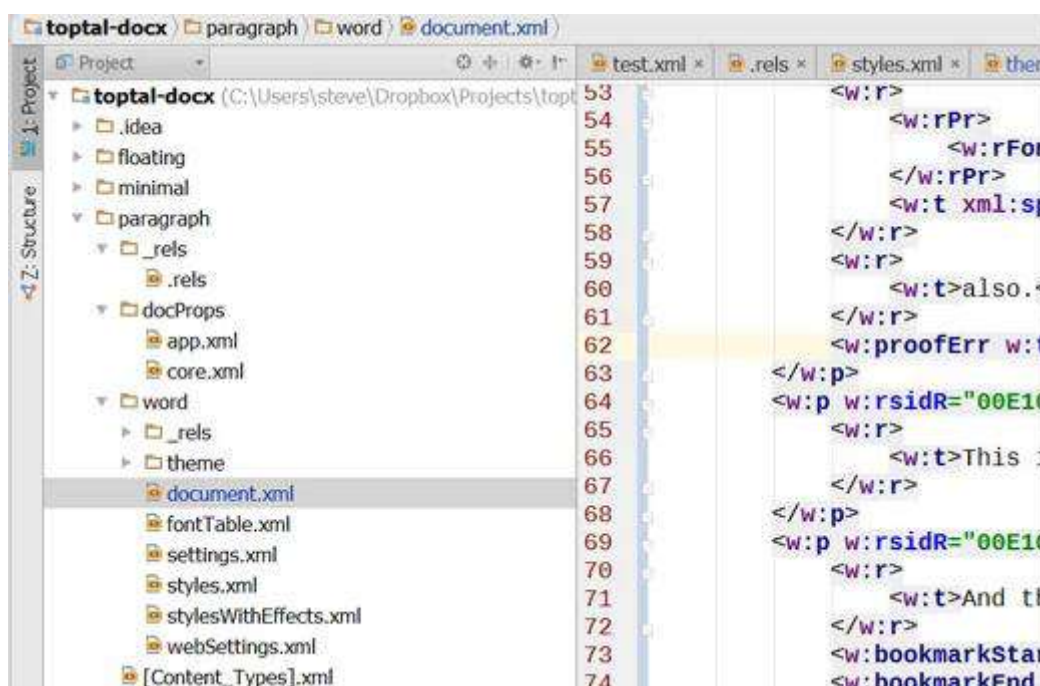


Fig. 1: Ejemplo de la estructura interna de un documento *DOCX*.

Fuente: [<https://www.toptal.com>]

A partir de poder acceder a las estructuras y elementos de un archivo *DOCX* se podrá estudiar la sintaxis y así tener una comprensión técnica de la misma, siendo importante para el proceso de accesibilización automático que propone la presente tesina. En este sentido, es de importancia el análisis de esta estructura y sus elementos que la componen, de manera que al querer trabajar con documentos *DOCX* se pueda tener conocimiento de su interior. Términos que van a ser útiles para ese caso son: Nodos, Párrafos, Propiedades sobre textos, Estilos, entre otros. Respecto de los Nodos, la especificación de Open XML comprende un archivo XML llamado document.xml, con la estructura de nodos interna. Cada nodo tiene un comienzo y un final que encapsula los nodos adicionales o el contenido. Un ejemplo simplificado de un archivo XML de este tipo se puede observar en la siguiente figura (Fig. 2):


```

<w:document>
  <w:body>
    <w:p w:rsidR#"005F670F" w:rsidRDefault#"005F79F5">
      <w:r><w:t>Documento de Ejemplo</w:t></w:r>
    </w:p>
    <w:sectPr w:rsidR#"005F670F">
      <w:pgSz w:w#"12240" w:h#"15840"/>
      <w:pgMar w:top#"1440" w:right#"1440" w:bottom#"1440"
w:left#"1440" w:header#"720" w:footer#"720"
      w:gutter#"0"/>
      <w:cols w:space#"720"/>
      <w:docGrid w:linePitch#"360"/>
    </w:sectPr>
  </w:body>
</w:document>

```

Fig. 2: Contenido del archivo document.xml que representa a los nodos y elementos que la componen.

El nodo principal <w:document> representa el documento en sí mismo, el <w:body> contiene párrafos y anidadas dentro de <w:body> están las dimensiones página definidas por <w:sectPr>.

Sobre los párrafos hay que considerar que, un simple documento consiste de párrafos, representados por el elemento <w:p>, un párrafo consiste de una o más secuencias de <w:r>, denominados “runs”, que contienen una serie de textos con un estilo propio. Las secuencias consisten de caracteres tal como etiquetas <w:t>, que contienen textos y pueden haber más de uno dentro de una secuencia o “run”. Además de los “runs”, los párrafos también pueden contener otros elementos del documento, como hipervínculos, comentarios, etc. A continuación, se muestra en la siguiente figura (Fig. 3) una estructura de párrafo de ejemplo:

```

<w:p>
  <w:pPr>
    <w:pStyle w:val#"MyStyle"/>
    <w:spacing w:before#"120" w:after#"120"/>
  </w:pPr>
  <w:r>
    <w:t xml:space#"preserve">Un párrafo es un contenedor principal
dentro de un documento que además consiste de uno a más runs donde el
texto del párrafo es actualmente contenido.</w:t>
  </w:r>
</w:p>

```

Fig. 3: Estructura de un párrafo en document.xml

Las propiedades básicas del texto son fuente, tamaño, color, estilo, etc. Hay alrededor de 40 etiquetas que especifican la apariencia del texto. En la siguiente figura (Fig. 4) se puede observar un “run” compuesto de texto con sus propias propiedades dentro de `<w:rPr>`, especificando `<w:color>`, `<w:rFonts>` y negrita `<w:b>`.

```
<w:r>
  <w:rPr>
    <w:rFonts w:ascii="Arial" w:hAnsi="Arial" w:cs="Arial"/>
    <w:b/> <w:color w:val="000000"/>
  </w:rPr>
  <w:t xml:space="preserve" text</w:t>
</w:r>
```

Fig. 4: Propiedades en texto en document.xml

Fuente: [<https://docs.fileformat.com>]

Sobre los estilos en un documento se puede mencionar que, existe un grupo de opciones en los procesadores de texto dedicado a los estilos: normal, sin espaciado, título 1, título 2, entre otros. Esta información se almacena en un archivo `styles.xml`, una vez aplicado un estilo sobre un texto, surge una referencia a ese texto y su valor del estilo. En nuestro documento principal de nodos aparece la etiqueta `<w:pPr>` y en el archivo `styles.xml` aparece la etiqueta `<w:pStyles>` que contiene otras etiquetas propias del elemento y su información relacionada.

Otros elementos que pueden ubicarse dentro de los párrafos son imágenes, aquí aparecen etiquetas como `<w:drawing>` haciendo que surja el atributo ID para referenciar esa imagen, esta información se encuentra contenida en `documents.xml.rels`. La imagen podría estar entre texto y con algún tipo de alineación, como puede ser de manera flotante, esto hace que debamos reemplazar la etiqueta anterior por `<wp:anchor>`.

Otro elemento a mencionar es una tabla, la misma es representada con una lógica similar a lo que sucede en el lenguaje de marcado HTML. Aparecen etiquetas como por ejemplo, `<w:tbl>` para definir el componente tabla, `<w:tblPr>` para definir propiedades, `<w:gridCol>` para definir propiedades de una columna, que se encuentra dentro de la etiqueta `<w:tblGrid>`. Cada fila definida por etiquetas `<w:tr>`. Además, los hipervínculos también tienen su representación lógica dentro de un documento a través de etiquetas, donde se almacena su dirección y el texto a mostrar.

Podríamos seguir mencionando otros componentes muy valiosos para lograr accesibilizar un documento, como las listas o un encabezado. Los mismos siguen una lógica determinada por las especificaciones impuestas por el lenguaje de etiquetas y su estructura de nodos del OpenOffice XML, sin embargo, para no extendernos en el

lenguaje a un nivel bajo, donde la programación a veces escapa y gestiona el archivo a un nivel superior, los más representativos fueron los mencionados anteriormente.

Por otro lado, el formato *PDF* tendrá una estructura que organice su contenido, la misma se describe a continuación.

Características técnicas sobre *PDF*

En cuanto a un documento *PDF*, éste se compone de una estructura de código y datos, teniendo como finalidad la transmisión de información y no fue pensado para la edición. Básicamente un *PDF* es un archivo de texto ASCII, aunque incluya datos binarios. Su estructura interna se basa en el formato estándar de Adobe *PDF* [40], que define cómo se organizan y almacenan los datos dentro del archivo. Un *PDF* se organiza utilizando un código de caracteres ampliamente utilizado para representar caracteres alfabéticos, numéricos y símbolos en dispositivos electrónicos y sistemas informáticos denominado ASCII (*American Standard Code for Information Interchange*), excepto ciertos elementos que pueden tener contenido binario, siendo este un sistema de numeración en el que los números son representados utilizando únicamente dos cifras: 0 (cero) y 1 (uno). El formato de archivos *PDF* usa una estructura fija compuesta de 4 partes o secciones y se puede observar en la siguiente figura (Fig. 5):

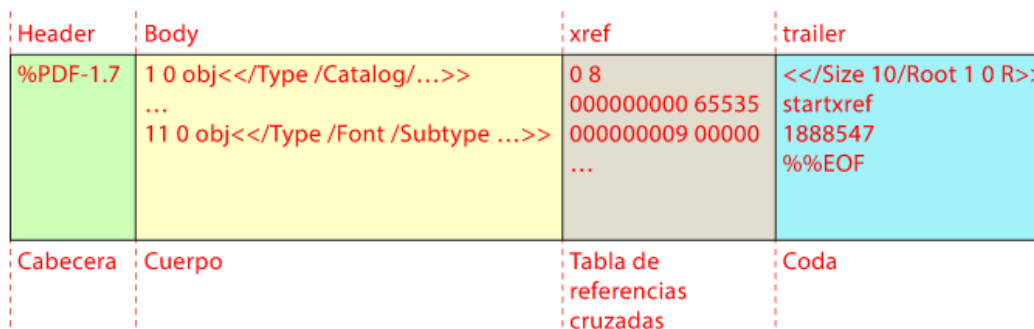


Fig. 5: Ejemplo de la estructura interna de un documento *DOCX*.

Fuente: [<http://www.gusgsm.com/>]

Siempre contiene 4 secciones:

- 1) Una cabecera (*header*): contiene la información de la versión del estándar *PDF* utilizada, tal como "%PDF-1.x", donde "x" es un número que representa la versión específica;
- 2) Un cuerpo (*body*): contiene la descripción de cada uno de los elementos que existen dentro de las páginas (estos elementos se consideran objetos), se organizan de manera jerárquica pero no siguen un ordenamiento ni tienen un nombre en particular, y la única información que existe sobre la jerarquía es que en su nivel superior se encuentra un objeto raíz (root) que es el diccionario catálogo, el punto de entrada al contenido;
- 3) Tabla de referencias cruzadas (*xrefs table*): contiene información sobre las relaciones y conexiones entre diferentes partes del contenido del *PDF* y los elementos;
- 4) Coda (*Trailer*): indica al programa que interpreta el documento donde se encuentran algunos elementos esenciales para su lectura, como la tabla de referencias cruzadas, y el objeto raíz (root) en la jerarquía del documento.

A partir de la estructura de un *PDF* podemos mencionar que los elementos básicos que la componen se denominan objetos y los principales son: matrices, valores booleanos, valores reales, valores enteros, cadena de caracteres, diccionarios, flujos de datos y el elemento vacío. Dentro de este formato, podemos encontrar información adicional sobre el archivo, denominados “metadatos”, los cuales proporcionan, por ejemplo, el título y autor. Estos metadatos se almacenan en objetos con sintaxis en XML y acompañan a la composición del documento.

A partir de las características de ambos formatos, *DOCX* y *PDF*, observamos que tenemos una cierta cantidad de información acerca de la estructura de un documento de lectura digital. En este sentido, a partir del conocimiento de distintos lenguajes de programación que permitan el acceso a esta estructura, se podrá incursionar sobre la gestión del contenido de la misma y aplicar criterios de accesibilidad que son fundamentales para generar documentos que puedan ser leídos por todas las personas. De esta manera, podemos avanzar sobre la implementación de la accesibilidad en documentos digitales de lectura a partir de herramientas que puedan existir para este propósito, de manera que en la siguiente sección se describirán algunos de ellos en cuanto a sus características principales y su enfoque accesible.

3.5 Herramientas para accesibilizar documentos.

La accesibilidad se refiere a la capacidad de una persona con discapacidad o de otra situación de vulnerabilidad, de acceder, utilizar y comprender de manera efectiva un producto, servicio o entorno. En este sentido, existen aplicaciones, plataformas online y herramientas tecnológicas programáticas que permiten el acceso a cada una de ellas tanto a una persona con conocimiento básico en computación o nivel avanzado de informática. De esta manera, debemos reconocer el alcance y las posibilidades que un usuario pueda utilizar estas herramientas con el propósito de accesibilizar un documento. A continuación, se mencionan estas distintas posibilidades y características de accesibilidad de cada una de ellas.

Herramientas sobre *DOCX*.

- 1) Herramienta de Microsoft: Este software se instala en una computadora para su uso. Además de poseer las funcionalidades básicas para gestionar el contenido de los elementos en un documento tales como aplicar estilos, creación de títulos, índice, entre otros elementos y operaciones, desde la versión 2016 incorporó una herramienta dentro de sí mismo denominada “Comprobar accesibilidad”, ubicado en la cinta de opciones, en la pestaña “Revisar”, dentro del grupo “Accesibilidad”. La herramienta de accesibilidad incorporada en el software permite inspeccionar el documento y devolver como resultado errores y advertencias, junto con sugerencias para aplicar una corrección sugerida. Estos resultados tienen etiquetas tales como:
 - “Falta el texto alternativo”: cuando una imagen no tiene una descripción asociada,
 - “Imagen y objeto no alineados”: cuando un elemento Forma no está insertado de manera correcta.
 - “Contraste de texto difícil de leer”: cuando el color de fuente y el contraste no son los adecuados para una lectura mínima del texto.Se ve reflejado una lógica de evaluación aplicada en base a ciertos criterios mínimos que se fundamentan dentro del marco de la accesibilidad web y las

pautas de accesibilidad, y a partir de la versión 2019 posee un lector en voz alta incorporado como funcionalidad. Es importante destacar, en este marco, que Microsoft Word posee un lector en voz alta. Es un software instalable cuya licencia es propiedad de Microsoft, no siendo un software de distribución gratuita de manera que para utilizarlo de manera ilimitada y hacer uso de todas las funciones se debe adquirir una licencia; aunque por otro lado es un procesador de texto muy popular del mercado [41]. Está disponible en versión en español, y además existe una versión en la nube, es decir, podemos utilizar una versión online del software, a través del navegador con muchas de las funcionalidades que se mencionaron anteriormente, pero esta vez de manera gratuita, aunque teniendo que registrar una cuenta sobre la plataforma Microsoft.

- 2) Documentos de Google: En cuanto a la creación de documentos accesibles, mediante el procesador de texto en línea de la suite Google Docs Editors, éste permite dar estilos a cada componente del documento, y además existe la posibilidad de incorporar el complemento “Accessibility Checker for Docs”. Este complemento es instalable sobre esta herramienta, pudiendo lograr a través de pautas, garantizar que el idioma esté configurado para el documento, que el documento tenga un título, que la fuente y los colores de fondo tengan una alta relación de contraste, que todas las imágenes tengan texto alternativo y muchas otras comprobaciones.

Esta herramienta se utiliza de manera online, está disponible en lenguaje español y de manera gratuita al usuario general, ya sea por medio de un vínculo compartido o a través de la creación de una cuenta en la plataforma Google.

- 3) LibreOffice: Sobre la creación de documentos accesibles, permite crear todo tipo de elementos dentro de un documento, aplicar estilos, títulos, entre otras funciones. Además, permite el guardado al formato *DOCX*. Tiene una herramienta incorporada llamada “Revisión de accesibilidad” el cual realiza una verificación de la estructura del documento e informa los problemas. No posee un lector de texto a voz.

Herramientas sobre formato *PDF*

Existen, por ejemplo, páginas web, software y complementos que se obtienen a partir de acceder a una plataforma online, los cuales permiten realizar un proceso de accesibilizar documentos en formato *PDF*. A continuación, se mencionan algunos de ellos:

1) Adobe Acrobat Reader: El software Adobe Acrobat Reader es el estándar global gratuito y de confianza para ver, imprimir, firmar, compartir y comentar archivos *PDF*. En cuanto a la accesibilidad, para su uso, ofrece distintas funcionalidades, por ejemplo:

- Personalización de preferencias: donde se puede cambiar las configuraciones del programa y adaptarlas según las necesidades.
- Trabajar con ampliadores: posee varias opciones con respecto a los tamaños y modos de lectura.
- Lector de archivo en voz alta: posee esta función integrada, y también permite trabajar con software profesional de lectura de pantalla.

Existe la versión gratuita y en español. Por otro lado, existe una versión paga, versión Pro. La misma incorpora mayores funcionalidades sobre la versión gratuita, principalmente la de creación del documento. Para generar documentos accesibles, está la opción de utilizar la versión Pro donde se permite más operaciones tales como:

- Marcado y navegación estructurada: agregar etiquetas de estructura a los elementos del documento *PDF*, como encabezados, párrafos y tablas, para facilitar la navegación y la comprensión del contenido por parte de los lectores de pantalla.
- Ajustes de contraste y color: opciones para ajustar el contraste y el color de fondo de los documentos *PDF*, lo que ayuda a mejorar la legibilidad y facilitar la lectura para personas con discapacidad visual o sensibilidad a ciertos colores.
- Accesibilidad de formularios: posibilidad de crear formularios *PDF* accesibles al utilizar etiquetas y descripciones adecuadas para los campos de formulario.
- Verificación y reparación de accesibilidad: herramientas para verificar y corregir la accesibilidad de los documentos, pudiendo identificar problemas de accesibilidad tales como falta de etiquetas de estructura o texto alternativo en imágenes, y teniendo la posibilidad de realizar ajustes.
- Texto alternativo para imágenes: posibilidad de agregar texto alternativo descriptivo a las imágenes.

- Tablas accesibles: se permite agregar etiquetas de encabezado de columna y fila a las tablas en los documentos *PDF*, mejorando la comprensión y navegación sobre las mismas.
- 2) PAVE⁶: Es una herramienta gratuita desarrollada por el Laboratorio TIC- Accesibilidad de la *ZHAW*, en colaboración con la asociación de ciegos suizos (SBV). Esta herramienta analiza la accesibilidad de los documentos *PDF*. En algunos casos es capaz de corregir automáticamente los problemas encontrados. Posee las siguientes características:
- Es una herramienta en línea.
 - Se proclama ser sencillo e intuitivo.
 - Está en idioma alemán e inglés, por lo que si no sabes alguno de estos puede que sea algo más dificultoso.
 - Realiza modificaciones automáticas, esto podría convertir un elemento dentro del documento en otro tipo de elemento no deseado en algunos casos.
 - No cambia el diseño visual de su *PDF*.
 - El documento *PDF* no debe superar los 5 megabytes.
- 3) CommonLook⁷: Complemento de tipo Plugin que trabaja junto con el producto de software Adobe Acrobat que permite validar documentos de tipo *PDF*, cumpliendo con los estándares de accesibilidad WCAG en su proceso de validación. Características:
- Trabaja como Plugin dentro de *Adobe Acrobat*
 - Verificaciones y validaciones de cumplimientos de manera automática.
 - El chequeo se realiza sobre documentos en formato *PDF*.
 - Implementa verificaciones de estándares WCAG 2.0, WCAG 2.1, entre otras.
 - Gratuito y con idioma principal en inglés.

⁶ La página oficial de PAVE es <https://pave-PDF.org/>

⁷ La página oficial de CommonLook es <https://commonlook.com/accessibility-software/PDF/>

- 4) PDF Accessibility Checker (PAC) ⁸: Es un Software gratuito de la fundación suiza “Access for all” para pruebas de accesibilidad de *PDF*, logrando realizar verificaciones sobre estándar *PDF/UA*. Características:
- Software gratuito, descargable e instalable (sólo en Windows).
 - Comprobaciones sobre estándar WCAG 2.1
 - Previsualización de la estructura lógica del documento.
 - Vista de árbol de etiquetas con todas las propiedades y atributos.
 - Software en su versión 2021 con idioma principal en inglés.

Herramientas programáticas

La programación está inmersa en cualquier herramienta o software que utilizamos en la vida cotidiana, de hecho, cada herramienta está programada con uno o varios lenguajes para que puedan funcionar y proveer de servicios al público en general.

En este sentido podemos encontrar que la programación es aplicable en ciertos ámbitos de un software en producción, ya sea desde su código de origen, por ejemplo, desde el código fuente de una plataforma online o generando un producto desde cero, o desde alguna herramienta que permita la incorporación de código programático sobre el mismo software en ejecución.

A continuación, se describirán algunas de las tecnologías que permiten al programador utilizar de ellas con la finalidad de gestionar documentos y su accesibilidad, entre los que se encuentra las denominadas Bibliotecas o Librerías, que no son más que una tecnología creada con la programación que permite ser utilizada por otro programador a través de un lenguaje particular. Estas tecnologías son las siguientes:

- 1) Script Lab: es un proyecto de *Microsoft Garage*, de código abierto a la comunidad, disponible en su repositorio en la web [42]. Es una herramienta de tipo complemento, que permite comunicarse con la API (Application programming interface) de *JavaScript* de *Office* sin tener que salir de *Excel*, *Word* o *PowerPoint*. Esto significa que, existe una *API* disponible para comunicarse con el software (por ejemplo, *Word*) y está implementado para que esa comunicación se realice mediante el lenguaje de programación *JavaScript* [43]. A partir de esta *API*, tenemos la posibilidad de crear, ejecutar y compartir

⁸ La página oficial de PAC es <https://PDFua.foundation/en/PDF-accessibility-checker-pac/>

"Fragmentos de Código del complemento de *Office*" desde *Excel*, *Word* o *PowerPoint*. Estos fragmentos se denominan *Snippets*, su término en el idioma. El complemento *Script Lab*, se obtiene a partir de la pestaña del *Word* llamada Insertar, y luego ingresando al grupo Complementos, donde a partir de la búsqueda online se logra la instalación. A partir de la correcta incorporación del complemento a nuestro software, se puede comenzar a programar. Algunas de las acciones que podemos realizar con este complemento son:

- Crear y editar fragmentos de código que pueden incluir *JavaScript*, *HTML*, *CSS* [44] y referencias a bibliotecas y datos en la web.
- Ejecutar los fragmentos de código y ver instantáneamente el resultado en un panel de tareas y en el documento de *Office*.
- Comenzar rápidamente con una selección de fragmentos de código de muestras disponibles predefinidas.
- Compartir y guardar los fragmentos de código con *gists* en *GitHub*, un repositorio de código online gratuito.
- Aprendizaje de la *API* de *JavaScript* con la ayuda de *IntelliSense* mientras edita.
- Programar en *TypeScript* (pudiendo implementar funciones modernas), incluso las versiones superiores a la 2.0 que permite usos como *async/await*.
- Desarrollar un prototipo de un complemento.

De esta manera tenemos la posibilidad de generar estructuras y contenido a nuestro documento, con el uso de la *API* en *JavaScript*, desde crear los distintos estilos de Títulos, colores, interlineado, y gestionar diversas configuraciones para que se logre un documento con un cierto nivel de accesibilidad aceptable. Este proyecto funciona en *Excel Word* y *PowerPoint* desde *Office* 2013 y versiones posteriores, *Office Online* y *Office* para *Mac*.

- 2) Officegen: es una biblioteca *JavaScript* de código abierto que permite a los desarrolladores de software trabajar con archivos *Office Open XML* y crear *Word (DOCX)* para *MS Office* 2007 y versiones posteriores, dentro de sus propias aplicaciones *JavaScript*. Además, permite trabajar otros tipos de archivos que son manejados por las aplicaciones del ambiente propietario de *Microsoft*. Sus características son las siguientes:

- Es muy flexible y se puede utilizar en varios entornos: los entornos que admiten *Node.js* son totalmente compatibles con él, como *Linux*, *OSX* y *Windows*.
- Compatible con varias funciones de procesamiento de *Word* para documentos de *Microsoft Word* (archivo *DOCX*): a) creación de documentos de *Word*; b) incorporación de uno o más párrafos; c) incorporación de imágenes; d) compatibilidad con encabezados y pies de página; e) marcadores e hipervínculos.

Officegen promete a los programadores de software, la posibilidad de crear un nuevo documento de estilo *Word* a partir del lenguaje de programación *JavaScript*, además de permitir agregar uno o más componentes según sus necesidades. La documentación existente está en un repositorio en *GitHub* [45], en idioma inglés, y es suficiente para poder iniciar con esta tecnología, teniendo una comunidad activa considerable.

- 3) Docx: *Docx* de *JavaScript* [46] es una biblioteca que promete generar de manera simple archivos con extensión *.docx*, con el lenguaje *JavaScript* y *TypeScript*. El marco de trabajo se realiza con *Node* y sobre el navegador. Provee distintas funcionalidades que permiten generar estructuras y dar estilos a textos. También permite el manejo de imágenes y estructuras complejas como tablas, y se puede trabajar con tecnologías complementarias al lenguaje de *JavaScript*, por ejemplo, *HTML/JS*, *Angular*, *React*, *Vue.js*. Esta herramienta de *JavaScript* está entre las más populares para gestión de documentos ⁹ y su comunidad es activa. Si bien el trabajo de generar un documento es muy completo en cuanto a las posibilidades de manejo de componentes que pueden existir en un archivo de este tipo, el proceso de extraer la información de un documento existente aparece como

⁹ Publicación: “Mejores bibliotecas para construcciones de Archivos *Docx* con *JavaScript*” en <https://openbase.com/>. Lista cuidadosamente seleccionada por expertos de *Openbase*. Sitio web con más de 50.000 reseñas y calificaciones, relevando la experiencia de otros desarrolladores con cada paquete existente en la plataforma.

punto central en esta situación. Para esto, se debe rescatar párrafos, imágenes y todos los componentes de un documento, haciendo que el programador analice otras tecnologías para hacer esto posible.

- 4) Docxtemplater: Biblioteca para generar documentos *DOCX* desde un *template DOCX*. Es utilizado con JavaScript en un ambiente sobre *NodeJS* o sobre el navegador con *React*, *Angular*, *VueJS* y *NextJS*. Esta herramienta permite usar archivos *DOCX* como “*template*”, es decir, como una plantilla y rellenarlos con variables de *JavaScript*, basándose en el concepto de etiquetas (concepto similar al utilizado con *HTML*). Cuenta con una popularidad superior al resto de las herramientas existentes con este lenguaje⁵, siendo de mayor uso en proyectos reales en la web. Su documentación en *GitHub* [47] es completa para poder introducirse en esta tecnología, complementarse con otras y avanzar en procedimientos más complejos. Es una tecnología Libre el cual permite iniciarse dentro de la misma, pero también existen lo que se denomina módulos, los cuales representan funcionalidades determinadas (considerados *Plugins*) adicionales a las provistas en un principio y son de pago.
- 5) Python-docx: *Python* [48] es un lenguaje de programación multipropósito, y en este caso, puede trabajar sobre documentos, por lo que existen bloques de códigos denominados bibliotecas que brindan funcionalidades. En este sentido, existe una tecnología llamada “*python-docx*” [49]. *Python-docx* es una biblioteca de *Python* que permite crear y modificar archivos de *Microsoft Word* en formato *DOCX*, esto es posible a partir del uso de una *API* de alto nivel sin necesidad de conocer todo el esquema interno en la que se basa esta biblioteca para representar un documento y su estructura de contenidos. Realizar la escritura de un párrafo o modificar un título es posible a partir de sus métodos que son provistos sobre el documento en que se está trabajando. Además, esta tecnología permite el manejo de estilos sobre textos y párrafos tales como tamaño, fuente y espaciado, entre otras propiedades. Así mismo permite la detección de imágenes, hipervínculos, entre otros componentes que pueden existir en un documento. Si bien podemos editar un documento, obteniendo cada componente y alterarlo durante el proceso de modificación, debemos diferenciar los componentes para poder saber si dentro del proceso el elemento que estamos captando es una imagen o un texto. Posee documentación con información sobre los

componentes que son alcanzados. La instalación se realiza sobre el entorno de *Python* de manera sencilla a través del comando “pip install python-docx”.

Capítulo 4: Construcción del prototipo

En este capítulo se presentan los requerimientos para el escenario del caso de estudio, y el proceso de diseño e implementación de una herramienta que se denominará “AcceDocx”. Este nombre resulta de complementar la palabra “Accesible” y “Docx”. La primera palabra hace referencia al objetivo de la herramienta, accesibilizar un documento digital. La segunda palabra referencia al tipo de archivo con extensión “.docx”, el cual es el formato mayormente utilizado por los capacitadores en el caso de estudio y también por preferencia de las personas con discapacidad visual y ceguera que han participado del proceso de enseñanza. Este prototipo surge como alternativa de los procesadores de texto privativos, con el propósito de independizar la actividad de accesibilizar un documento sin la obligación de algún software. Así también, esta nueva herramienta es basado en web, por lo cual se tiene la potencialidad de acceder al mismo, no sólo por una computadora, sino también a través de un dispositivo móvil. De esta manera la construcción de este prototipo representa una herramienta libre, accesible, y con la posibilidad de hacer accesibilizar documentos digitales.

4.1 Requerimientos del prototipo

Como se mencionó inicialmente en la presente tesina, el caso de estudio por el cual se desarrolla este proyecto, es el escenario que sucede en el ambiente de enseñanza en CILSA ONG. Luego de finalizar las actividades anuales en la Organización, se realizó un breve cuestionario a un grupo del personal con rol de docente o profesor, más precisamente sobre 7 personas en total. Las preguntas fueron enfocadas a los conocimientos previos del profesor y sobre la actividad que debió realizar sobre accesibilidad; los resultados fueron los siguientes:

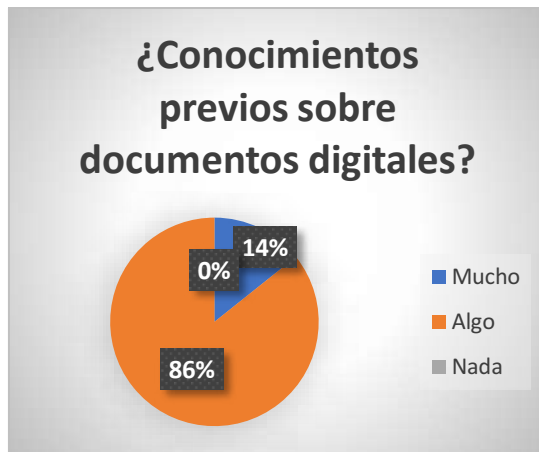


Fig. 6: Resultado sobre conocimientos previos en documentos digitales.

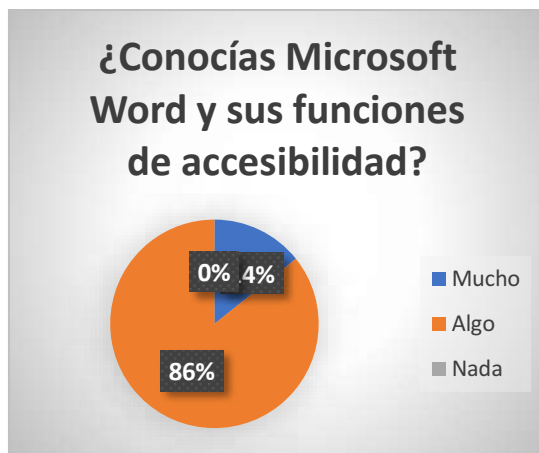


Fig. 7: Resultado sobre conocimientos previos en Microsoft Word y accesibilidad.

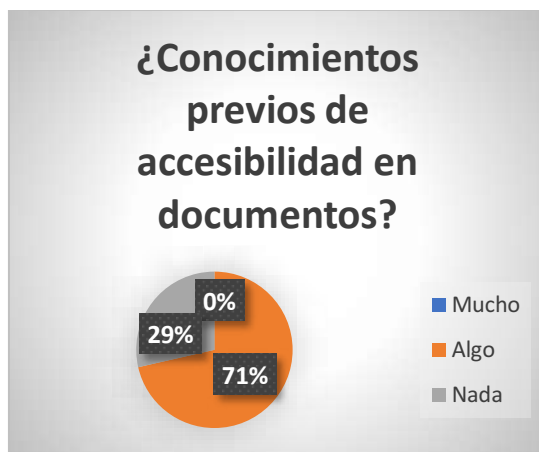


Fig. 8: Resultado sobre conocimientos previos en accesibilidad en documentos.



Fig. 9: Resultado sobre contratiempos sobre actividad.



Fig. 10: Resultado sobre observaciones repetitivas durante la actividad.

Este cuestionario permite relevar la información siguiente: la mayoría tenía un nivel básico sobre un procesador de texto, como Microsoft Word, además se observó que algunas características sobre estilos se podrían automatizar, y la aparición de algunos contratiempos en la convivencia de esta actividad y el manejo del horario laboral con las tareas principales del profesor.

En cuanto al alcance de los documentos, los formatos de documentos digitales de lectura que serán objeto del producto final de este proyecto, son los archivos *DOCX* y *PDF*, siendo estos tipos de documentos que se utilizan como material de estudio en un proceso de enseñanza, tal como sucede dentro del Programa de Oportunidades de Formación en el Centro de Inclusión Libre y Solidario de Argentina (CILSA) donde se brindan cursos de capacitación.

A partir de la información relevada por el escenario inicial y los cuestionarios, los requerimientos que se definieron para el prototipo incluyen:

- Facilidad de uso: cualquier persona debe poder utilizar el prototipo.
- Capacidad de subir documentos al sistema: podrá subir el documento en el sistema y permitirá al usuario visualizarlo y mantenerlo disponible.
- Capacidad de cambiar propiedades de un documento: el sistema permitirá realizar cambios en las propiedades del documento como autor y título del mismo.
- Capacidad de eliminar documentos: el prototipo debe permitir eliminar documentos del sistema que no se utilicen.
- Capacidad de aplicar criterios de accesibilidad: se permitirá trabajar con la accesibilidad del documento.
- Capacidad de descarga de documentos: el sistema permitirá descargar el documento luego de haberlo accesibilizado.
- Capacidad de conversión a *DOCX*: el sistema debe permitir la conversión de un documento *PDF* a formato *DOCX* para luego poder trabajar sobre él.

A continuación, se describe el desarrollo web con el que el usuario final interactúa para lograr su objetivo, de manera que se definen las herramientas utilizadas para la construcción de la solución y las justificaciones sobre las elecciones de las mismas. Así mismo, se menciona la implementación de la parte lógica del prototipo el cual se encarga de realizar el proceso de conversión de un documento hacia un documento accesible.

4.2 Desarrollo web del prototipo

El desarrollo de una aplicación web es fundamental a la hora de crear una herramienta que pueda ser accesible para un usuario a la hora de utilizar una herramienta independiente del dispositivo y lograr su propósito.

Para este proyecto se desarrollará el prototipo de una herramienta web que sea sencilla de utilizar por un usuario final con capacidades básicas de manejo de computadoras, intentando ser lo más simple y conciso posible en el diseño y armado del producto web.

El mismo debe ser un producto usable y accesible, lo cual fue estudiado en el capítulo 3, siendo la guía de recomendaciones de la WCAG 2.1 la utilizada para este propósito, dado que como se mencionó anteriormente, en la actualidad junio del 2023 es la guía establecida como recomendación.

A partir de los requerimientos principales del sistema a desarrollarse, mencionados anteriormente, las funcionalidades que deben implementarse serán las siguientes:

- Listar documentos.
- Subir un documento en formato *DOCX*.
- Eliminar un documento.
- Accesibilizar un documento.
- Descargar un documento accesible.
- Convertir un archivo *PDF* a *DOCX*.

A partir del relevamiento de toda la información mencionada anteriormente, se procede a definir el diagrama de los casos de uso, a partir de la siguiente figura (Fig. 11):

Casos de uso

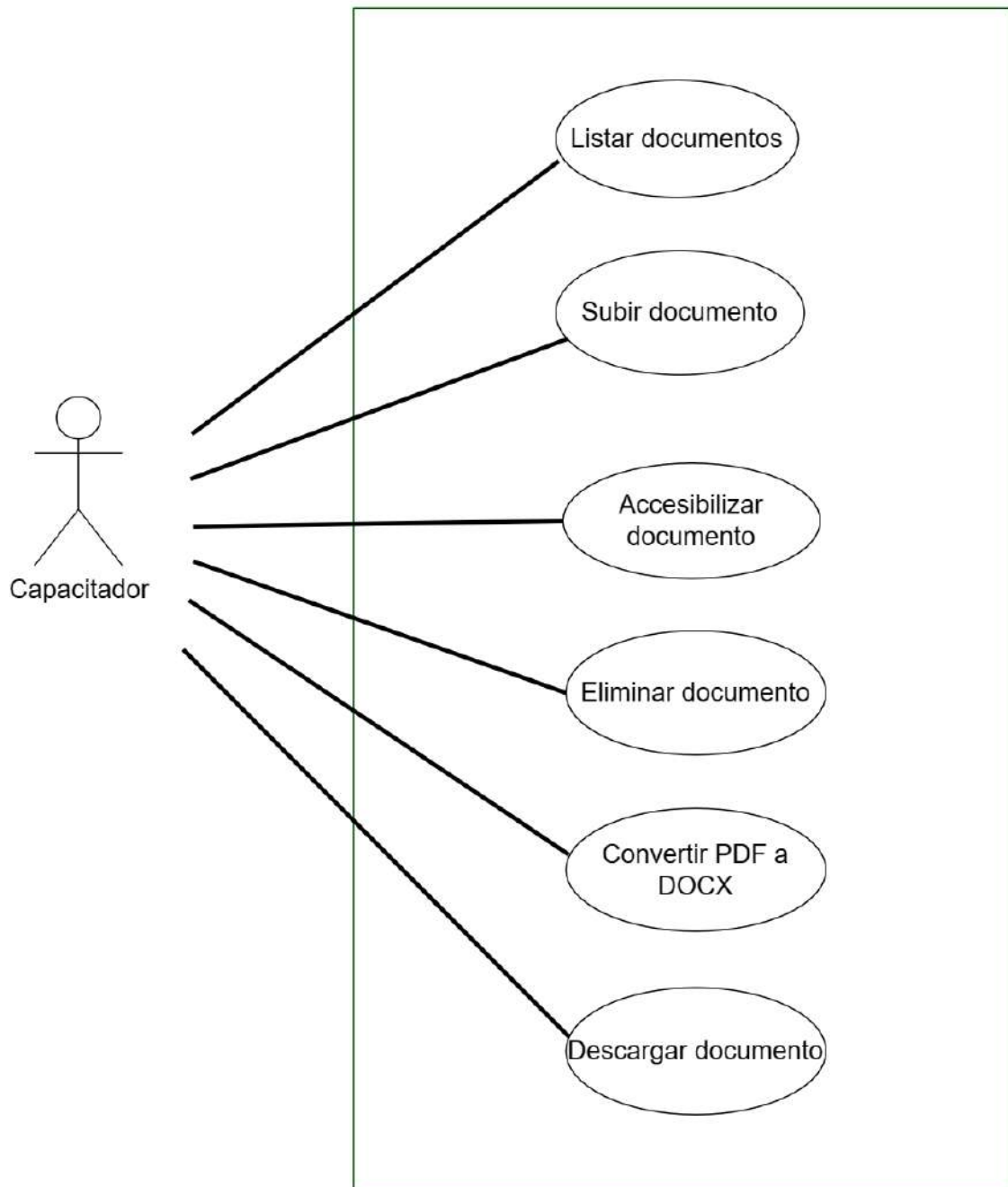


Fig. 11: Diagrama de los casos de uso entre el usuario y la aplicación web.

Teniendo en cuenta el diagrama anterior, se describen los casos de uso correspondientes:

CU-01	Listar documentos	
Fecha	16/01/2023	
Actor	Capacitador	
Precondición	El capacitador ha accedido a la aplicación web.	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el capacitador acceda a la sección "Documentos".	
Secuencia normal	Paso	Acción
	1	El capacitador presiona el botón Documentos, del menú de opciones en la parte superior de la página.
	2	El sistema muestra un listado de los documentos existentes en el sistema junto con sus acciones posibles representados en botones.
Postcondición	El capacitador visualiza la lista de documentos.	
Excepciones	Paso	Acción
	2	Si no existen archivos en el sistema, devuelve un listado vacío sin botones.
Comentarios	-	

CU-02	Subir documento	
Fecha	16/01/2023	
Actor	Capacitador	
Precondición	El capacitador ha accedido a la sección "Documentos".	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el capacitador presione en el botón "Subir documento".	
Secuencia normal	Paso	Acción
	1	El capacitador presiona el botón "Subir documento" que se encuentra en la parte superior izquierda.
	2	El sistema muestra una nueva página con un cuadro descriptivo que contiene información y un botón sobre la operación de subida.
	3	El capacitador presiona el botón "Comenzar" que se encuentra en el cuadro descriptivo.

	4	El sistema muestra una nueva página con un formulario.
	5	El capacitador completa los campos título y autor, con un botón para adjuntar un archivo <i>DOCX</i> , luego presiona el botón "Comenzar".
	6	El sistema realiza el almacenamiento del documento en el sistema.
Postcondición	El capacitador visualiza la lista de documentos.	
Excepciones	Paso	Acción
	5	El capacitador presiona el botón "Cancelar".
		El sistema muestra la página con el listado de documentos.
	6	Si el capacitador no adjunta un documento <i>DOCX</i> .
		Se cancela el caso de uso.
Comentarios	-	

CU-03	Accesibilizar documento	
Fecha	16/01/2023	
Actor	Capacitador	
Precondición	El capacitador ha accedido a la sección Documentos.	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el capacitador presione en el botón "Accesibilizar" sobre una fila del listado.	
Secuencia normal	Paso	Acción
	1	El capacitador presiona el botón "Accesibilizar" que se encuentra como una de las acciones disponibles.
	2	El sistema muestra una nueva página con información del documento y distintas secciones sobre el contenido del mismo.
	3	El capacitador completa los campos y selectores que se le presentan en distintas secciones, y presiona el botón "Terminar".
	4	El sistema realiza el almacenamiento del documento accesible en el sistema.
Postcondición	El capacitador visualiza la lista de documentos.	
Excepciones	Paso	Acción

	3	El capacitador presiona el botón "Cancelar".
		El sistema muestra la página con el listado de documentos.
	4	Si el capacitador no completa de manera correcta las secciones
		El sistema se mantiene en la página e informa sobre los campos que faltan completar.
Comentarios	-	

CU-04	Eliminar documento	
Fecha	16/01/2023	
Actor	Capacitador	
Precondición	El capacitador ha accedido a la sección Documentos.	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el capacitador presione en el botón "Eliminar" sobre una fila del listado.	
Secuencia normal	Paso	Acción
	1	El capacitador presiona el botón "Eliminar" que se encuentra como una de las acciones disponibles.
	2	El sistema muestra una ventana emergente solicitando una confirmación de la operación.
	3	El capacitador confirma la operación de eliminación.
	4	El sistema realiza la eliminación del documento del sistema.
Postcondición	El capacitador visualiza la lista de documentos.	
Excepciones	Paso	Acción
	3	El capacitador presiona el botón "Cancelar".
		El sistema muestra la página con el listado de documentos sin modificaciones.
Comentarios	-	

CU-04	Convertir PDF a DOCX	
Fecha	16/01/2023	
Actor	Capacitador	

Precondición	El capacitador ha accedido a la sección Documentos.	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el capacitador acceda a la sección "Convertir PDF".	
Secuencia normal	Paso	Acción
	1	El capacitador presiona el botón "Convertir PDF" del menú de opciones en la parte superior de la página.
	2	El sistema muestra una nueva página con un cuadro descriptivo que contiene información y un botón sobre la operación de conversión.
	3	El capacitador presiona el botón "Comenzar" que se encuentra en el cuadro descriptivo.
	4	El sistema muestra una nueva página con un formulario.
	5	El capacitador completa los campos título y autor, con un botón para adjuntar un archivo PDF, luego presiona el botón "Convertir".
	6	El sistema realiza la conversión del archivo a DOCX e inicia la descarga en la computadora.
Postcondición	El capacitador visualiza la lista de documentos.	
Excepciones	Paso	Acción
	3	El capacitador presiona el botón "Cancelar".
		El sistema muestra la página con el listado de documentos.
	4	Si el capacitador no adjunta un documento PDF.
		Se cancela el caso de uso.
Comentarios	-	

CU-04	Descargar documento	
Fecha	16/01/2023	
Actor	Capacitador	
Precondición	El capacitador ha accedido a la sección Documentos.	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el capacitador presione en el botón "Descargar" sobre una fila del listado.	
Secuencia normal	Paso	Acción

	1	El capacitador presiona el botón “Descargar” que se encuentra como una de las acciones disponibles.
	2	El sistema inicia la descarga del archivo <i>DOCX</i> , que se encuentra en el sistema, sobre la computadora.
Postcondición	El capacitador visualiza la lista de documentos.	
Excepciones	Paso	Acción
Comentarios	-	

A partir del escenario inicial, las funcionalidades requeridas y los casos de uso, lo siguiente es realizar el diseño del desarrollo y la selección de las tecnologías adecuadas para la implementación del prototipo.

Diseño e implementación del prototipo

Como se mencionó inicialmente en este documento, el proyecto que se lleva a cabo tiene como objetivo el desarrollo de una herramienta para accesibilizar documentos digitales de lectura, de manera que sea sencillo de usar y accesible para los distintos usuarios. En este sentido, la aplicación web se centra en ser lo más accesible posible desde su comprensión y su diseño, sin cargarlo de estilos avanzados ni componentes dinámicos que no son de aporte principal al objetivo del producto, y evitando que el usuario se vea afectado al momento de interactuar en el navegador.

A continuación, podemos observar una figura (Fig. 12) que representa un modelo preliminar, estilo mockup, de la página principal de AcceDocx presentada al usuario sobre un navegador web en una computadora.

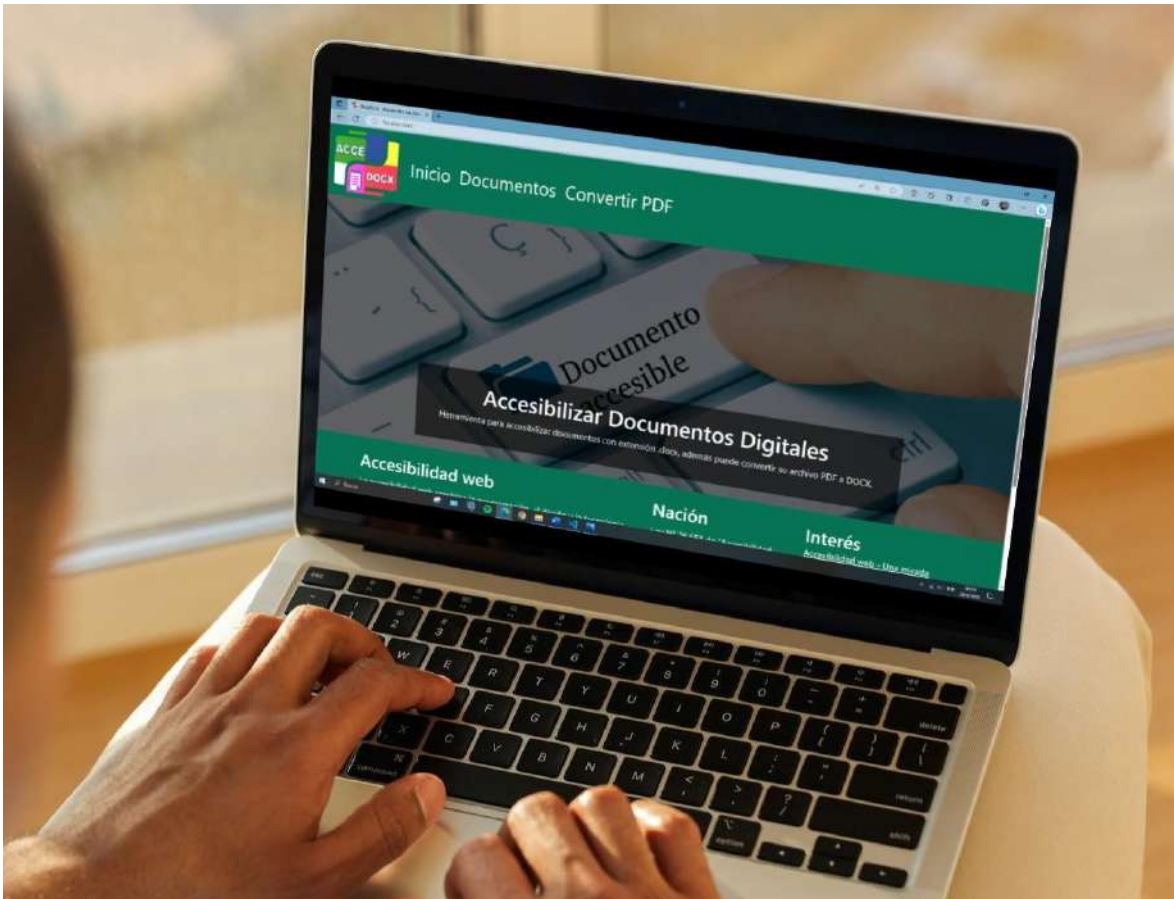


Fig. 12: Mockup de la página principal de la AcceDocx sobre una computadora.

En la figura anterior se observan 3 secciones:

- 1) Menú de opciones: El menú de opciones en la parte superior representa la posibilidad de acceder a distintas páginas que conforman el sitio web, las cuales son: Inicio, Documentos y Convertir *PDF*;
- 2) Contenido: En el centro de la pantalla se observa un espacio donde se presenta la información relacionada a la página en donde se encuentra posicionado el usuario, en este caso, la página donde comienza la aplicación es la pantalla principal Inicio.
- 3) Pie de página: Información relacionada a la temática y referencias sobre accesibilidad.

Así mismo, el prototipo se debe adaptar a distintos dispositivos con pantallas de diferentes tamaños, de manera que a continuación se muestra una figura (Fig. 13) representativa de AcceDocx desplegado en un dispositivo móvil.

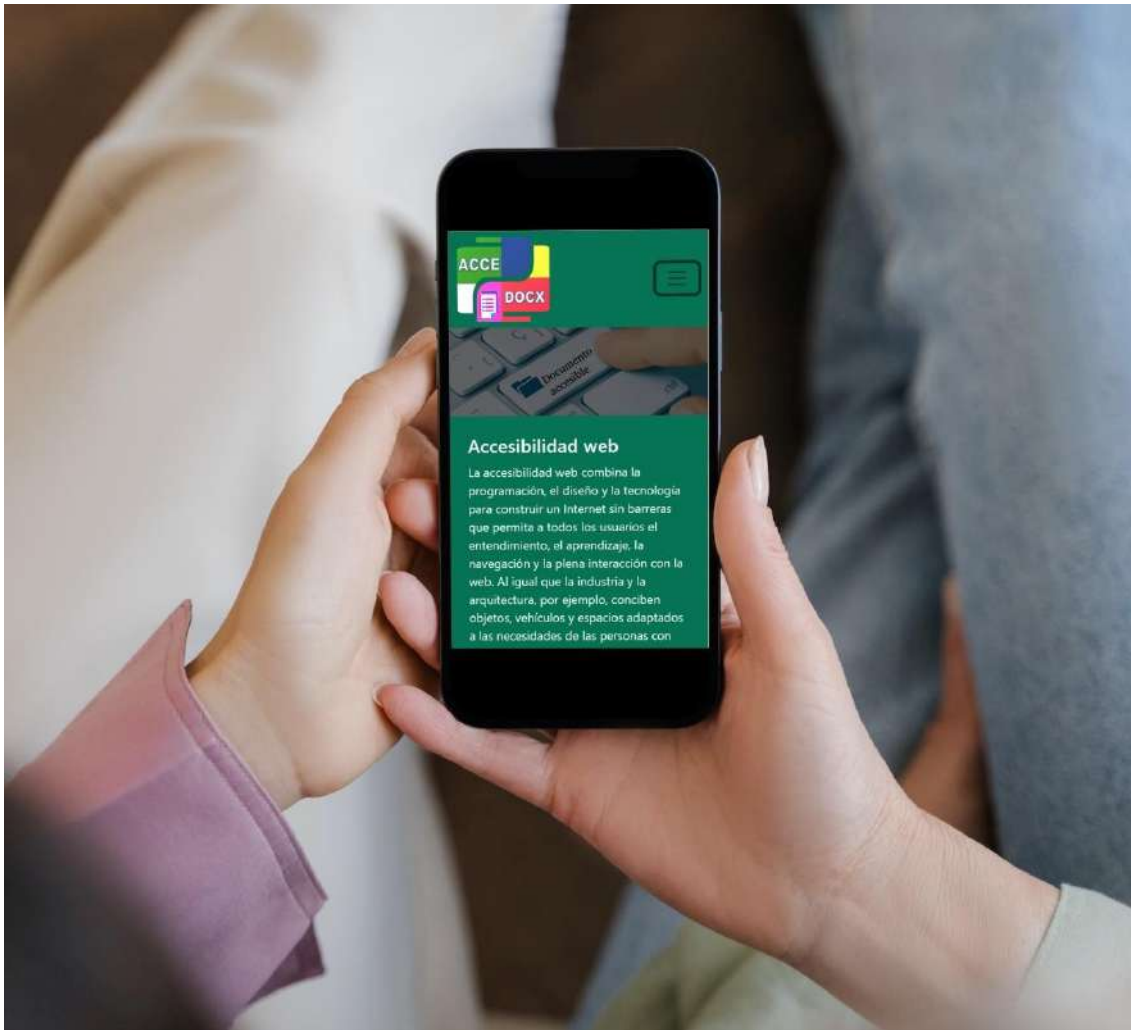


Fig. 13: Mockup de la página principal de la AcceDocx sobre un dispositivo móvil.

De esta manera, la página se adapta correctamente al tamaño de la pantalla, y colocando el menú como un elemento a desplegarse en caso que el usuario requiera.

El prototipo web comprende una serie de páginas, como la página principal mencionada anteriormente, y otras 2 más las cuales se accede a través del menú de opciones, por ejemplo, en la siguiente figura (Fig. 14) se observa una página para visualizar los documentos en el sistema, el cual se accede sobre la opción Documentos.

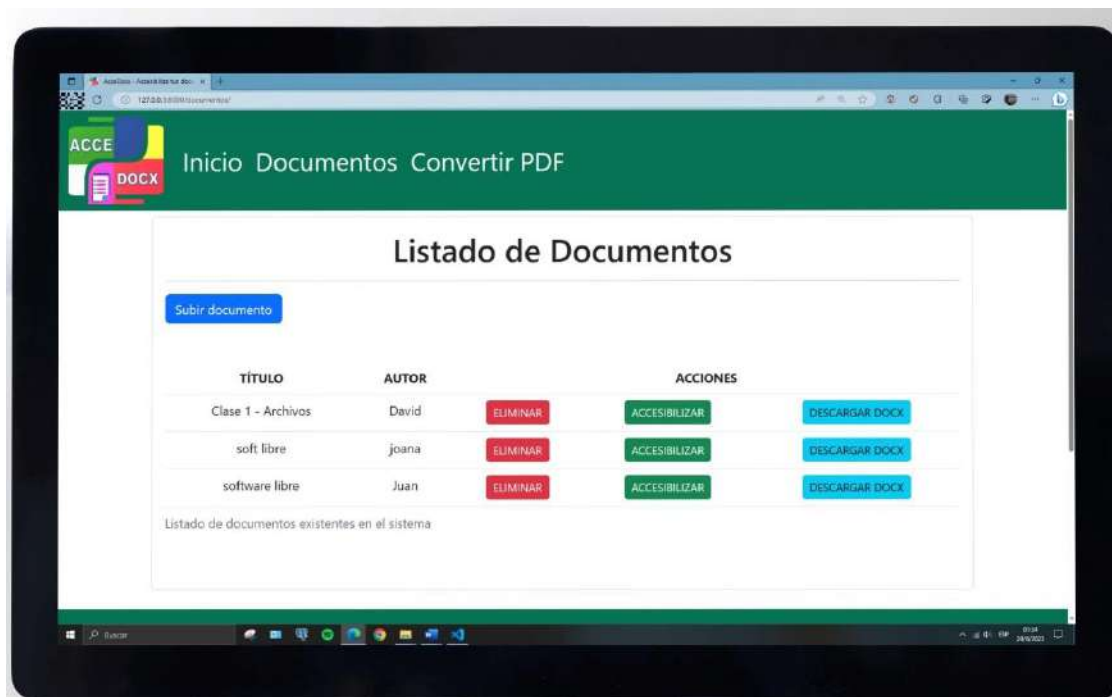


Fig. 14: Mockup de la página Listado de Documentos de la AcceDocx sobre una computadora.

En la figura anterior, se observa una página con el título Listado de Documentos, la cual representa el espacio donde los archivos que han sido subidos a la plataforma, y donde se podrán realizar operaciones sobre ellos, tales como: “eliminar”, “accesibilizar” y “descargar docx”. Además, a través de un botón llamado “Subir documento” que se encuentra en la parte superior izquierda del espacio de contenido de la página, se permite la acción de subir un documento al sistema. Este proceso se observa en la siguiente figura (Fig. 15) donde se encuentra la pantalla en cuestión.

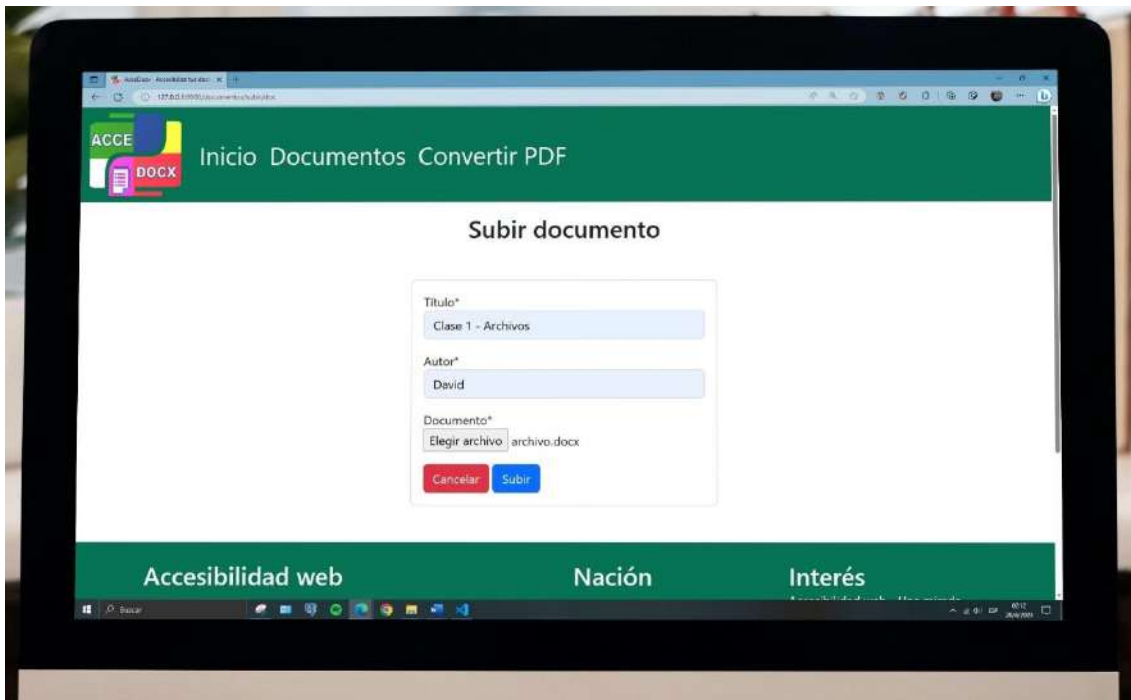


Fig. 15: Mockup de la página Subir documento de AcceDocx sobre una computadora.

Del mismo modo, para accesibilizar un documento, existe una página denominada “Accesibilizando” la cual se encarga de contener información relacionada al documento y poder interactuar con el usuario para aplicar accesibilidad sobre el documento. Esta página se observa a continuación (Fig. 16).

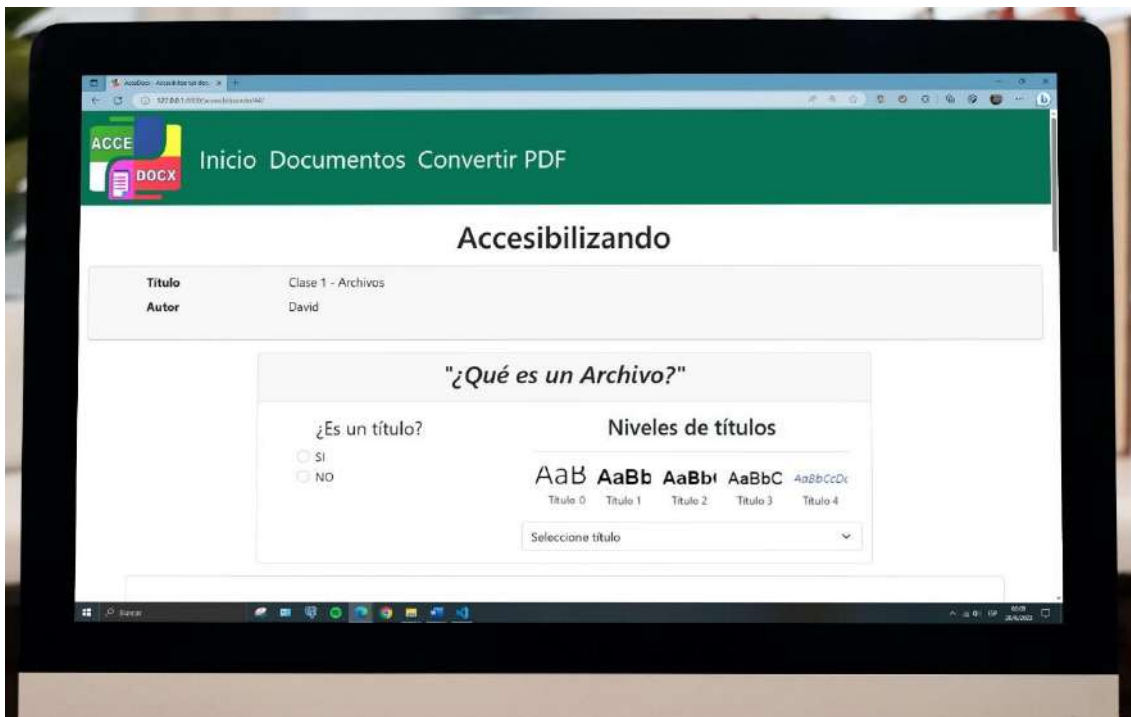


Fig. 16: Mockup de la página Accesibilizando de AcceDocx sobre una computadora.

Además, se puede observar en la siguiente figura (Fig.17), la página para convertir un *PDF* a un *DOCX*, la cual se accede sobre la opción *Convertir PDF* del menú.

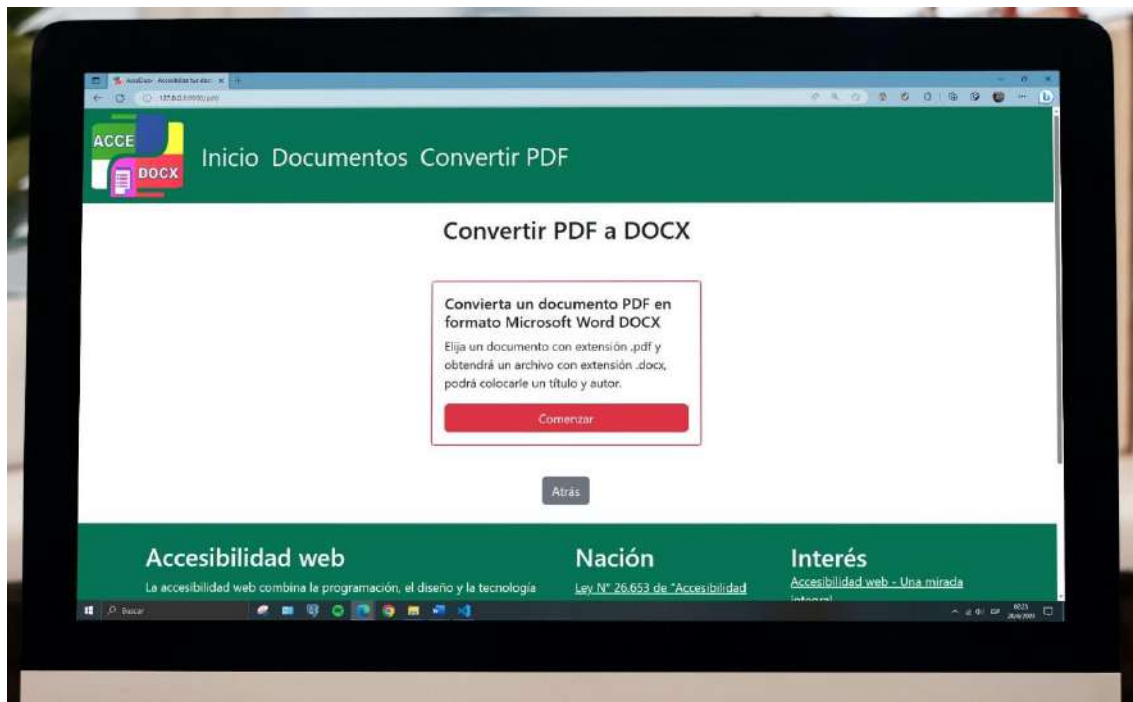


Fig. 17: Mockup de la página *Convertir PDF a DOCX* de *AcceDocx* sobre una computadora.

A partir del diseño preliminar del prototipo y teniendo la información previa, de las funcionalidades y requerimientos del desarrollo, se debió llevar a cabo el diseño del frontend, lugar donde el usuario inicia la comunicación con el producto web, como también del backend, la parte que se encarga de los datos que puedan ingresar a través del navegador y el almacenamiento de los mismos.

Para desarrollar el frontend, la toma de decisión fue pensada en la accesibilidad, de manera que se hicieron uso de tecnologías recomendadas por la WAI. El total de las tecnologías utilizadas para implementar el frontend del prototipo son las siguientes:

- HTML5 [50]: Última versión del estándar *HTML*, siendo este un lenguaje de etiquetas que permite construir la estructura de una página web.
- CSS: Lenguaje que nos permite generar “hojas de estilo en cascada” para poder crear y generar cambios en un documento *HTML* sobre su diseño y formato de presentación.

- JavaScript: Lenguaje de programación que nos permite desarrollar páginas web interactivas y dinámicas, permitiendo añadir funcionalidades.
- Bootstrap [51]: Un framework *HTML*, *CSS* y *JavaScript* para crear interfaces de usuario web limpias y adaptables.

A continuación, se presentará a través de una figura (Fig. 18), las tecnologías que conforman el frontend del prototipo.

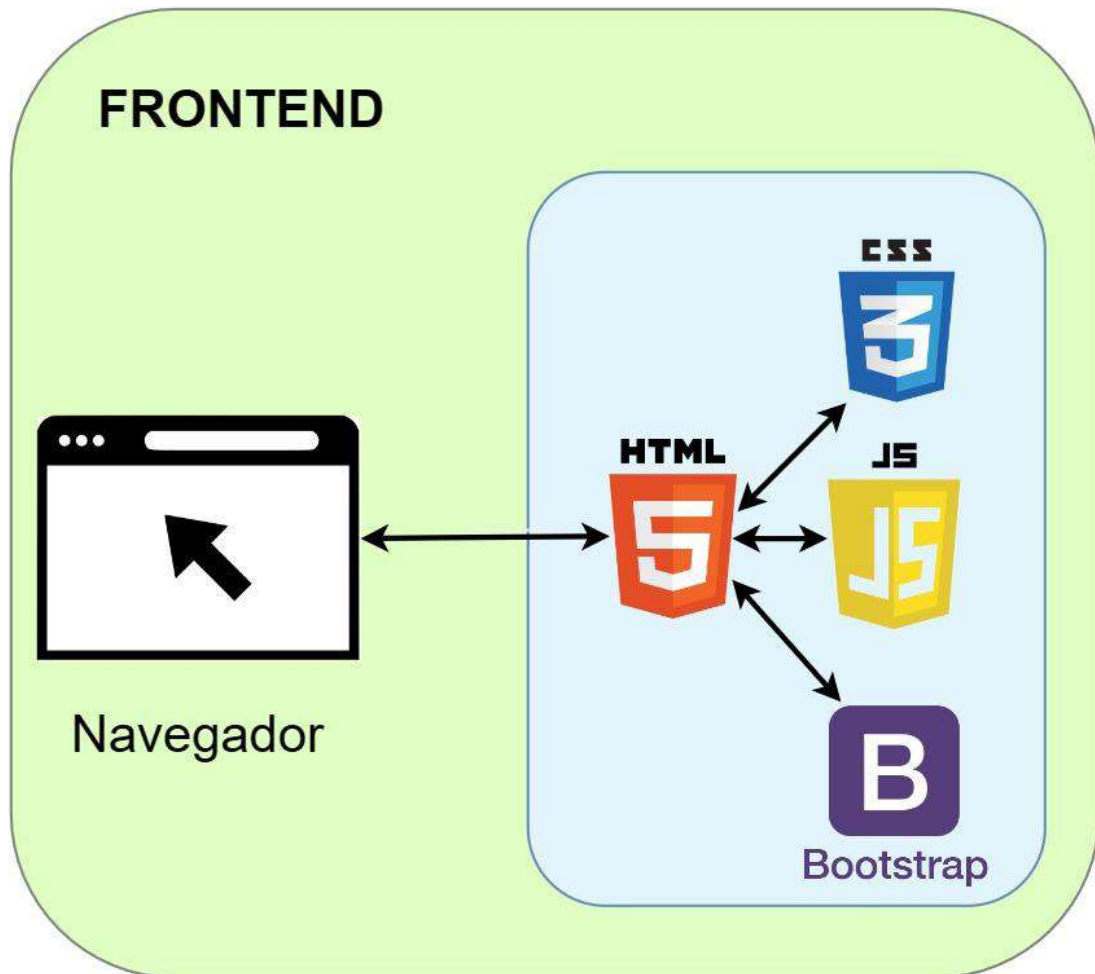


Fig. 18: Composición de la estructura Frontend de la aplicación web.

Para entender las tecnologías del frontend es necesario revisar las características principales de cada una de ellas.

Primeramente, *HTML* (*HyperText Markup Language*) es el lenguaje de marcado estándar, creado por Tim Berners-Lee, utilizado para crear y estructurar páginas web.

Algunas de las principales características de *HTML* son:

- Estructura básica: HTML utiliza una estructura básica que incluye elementos fundamentales como `<html>`, `<head>` y `<body>`, los cuales ayudan a organizar y estructurar el contenido de la página.
- Etiquetas y elementos: Se basa en el uso de etiquetas para marcar y definir diferentes partes del contenido.
- Hipertexto y enlaces: *HTML* permite crear enlaces a otras páginas web, recursos externos o ubicaciones dentro de la misma página.
- Imágenes y multimedia: Permite la inclusión de imágenes y otros medios, como videos y audio, en una página web.
- Semántica: A través de la última versión de *HTML*, *HTML5* se introdujo una serie de elementos semánticos que brindan significado adicional al contenido.

A partir de *HTML5*, la última versión del estándar *HTML*, se pueden encontrar distintas etiquetas, que conforman los elementos, los cuales ayudan a estructurar y describir mejor el contenido de una página web. *HTML5* permite una mejor experiencia de usuario al proporcionar elementos y atributos específicos para la creación de los distintos componentes de la página, también facilita la implementación de interacciones intuitivas, mayor accesibilidad a partir de los elementos semánticos incorporados, así como también la compatibilidad con distintos dispositivos móviles, proporcionando capacidades para adaptar y diseñar páginas web de forma sensible y optimizada para diferentes tamaños de pantalla y dispositivos. De esta manera se concluye en que *HTML5* se utiliza para el desarrollo de páginas web debido a su estandarización global, sus características mejoradas, su capacidad para crear experiencias interactivas y ricas, su enfoque en la accesibilidad y su capacidad de adaptarse a diferentes dispositivos y tamaños de pantalla. Teniendo esto en cuenta, permite desarrollar de manera completa la estructura de AcceDocx.

El lenguaje *CSS* (son las siglas en inglés de *Cascading Style Sheets*, que significa “hojas de estilo en cascada”) es un lenguaje que se usa para definir el estilo y la presentación de los elementos escritos en un lenguaje de marcado como *HTML* a la hora de armar una página web. Sus características principales son:

- Separación de contenido y presentación: Permite separar el contenido *HTML* de su presentación visual, encargándose de aplicar estilos y diseños a ese contenido.

- Selectores y reglas: Utilización de selectores para seleccionar elementos *HTML* específicos.
- Diseño responsive: *CSS* ofrece técnicas y propiedades para crear diseños responsivos, es decir, diseños que tienen en cuenta la adaptación a dispositivos móviles o tablets.

La importancia de *CSS* para crear páginas web radica en la posibilidad de tener control de la presentación (colores, fuentes, tamaños, etc.), separación sobre el contenido (estructura y contenido en *HTML*), flexibilidad y adaptabilidad a partir de que proporciona una amplia gama de propiedades y técnicas para crear diseño flexible y adaptativos, y además ayuda a la mantenibilidad y reusabilidad teniendo código que puede ser utilizado sobre múltiples elementos en una página.

De esta manera, *CSS* es fundamental para desarrollar la parte visual de *AcceDocx*.

Por otro lado, el lenguaje de programación *JavaScript* aporta interactividad y dinamismo a las páginas web. A partir de la manipulación de los elementos *HTML* de una página, se pueden realizar acciones como creación, modificación y eliminación de estos. Ayuda en la validación de datos ingresados por el usuario, aporta herramientas para la mejora de la usabilidad de una página web, tales como menús desplegables, autocompletado en formularios, validaciones en tiempo real, y acciones contextuales; estas características pueden beneficiar a personas con discapacidades motoras o cognitivas al hacer que el contenido sea simple de navegar y utilizar. En este sentido, *JavaScript* puede utilizarse para garantizar la compatibilidad con tecnologías de asistencia, como lectores de pantalla y teclados virtuales. Al aplicar las mejores prácticas de desarrollo web, como el uso adecuado de atributos *ARIA* [52] y la implementación de elementos interactivos accesibles, es posible garantizar que estas tecnologías puedan interpretar y comunicar correctamente el contenido y las funcionalidades a los usuarios con discapacidades. De este modo, *JavaScript* aporta a *AcceDocx* funcionalidades específicas para que pueda ser dinámica y para su correcta interacción con el usuario, por ejemplo, para el proceso de eliminación de un elemento de la página.

La última tecnología a mencionar es *Bootstrap*, un *framework*, es decir, una herramienta que ofrece una base sólida y un conjunto de funcionalidades predefinidas para simplificar el proceso de desarrollo de aplicaciones o sitios web. Esta herramienta es de código abierto, y proporciona un conjunto de herramientas, componentes y estilos predefinidos que facilitan el diseño y desarrollo de páginas web responsivas y con un

aspecto visual atractivo. Se basa en *HTML*, *CSS* y *JavaScript*. Algunas de las formas en las que *Bootstrap* ayuda en el desarrollo de páginas web son:

- Diseño responsive: *Bootstrap* está diseñado con un enfoque en la creación de sitios web responsivos, proporcionando una cuadrícula flexible y clases *CSS*.
- Componentes predefinidos: Ofrece una amplia gama de componentes y elementos de interfaz de usuario listos para usar, como botones, barras de navegación, formularios, tarjetas, carruseles, entre otros.
- Estilos y temas predefinidos: Proporciona estilos *CSS* predefinidos que permiten que los sitios web tengan una apariencia visual coherente y atractiva, además de la posibilidad de personalizar y adaptar los mismos según las necesidades del proyecto.
- Compatibilidad de navegadores: *Bootstrap* se ha probado y optimizado para garantizar la compatibilidad con distintos navegadores web.
- Documentación y comunidad: *Bootstrap* cuenta con una extensa documentación y una comunidad activa de desarrolladores. De esta manera *Bootstrap* aporta muchas herramientas a la hora de desarrollar una página web, como es el caso del diseño de AcceDocx.

Sobre el *backend* del prototipo web, tenemos la siguiente figura (Fig. 19) la cual muestra las tecnologías que la componen.

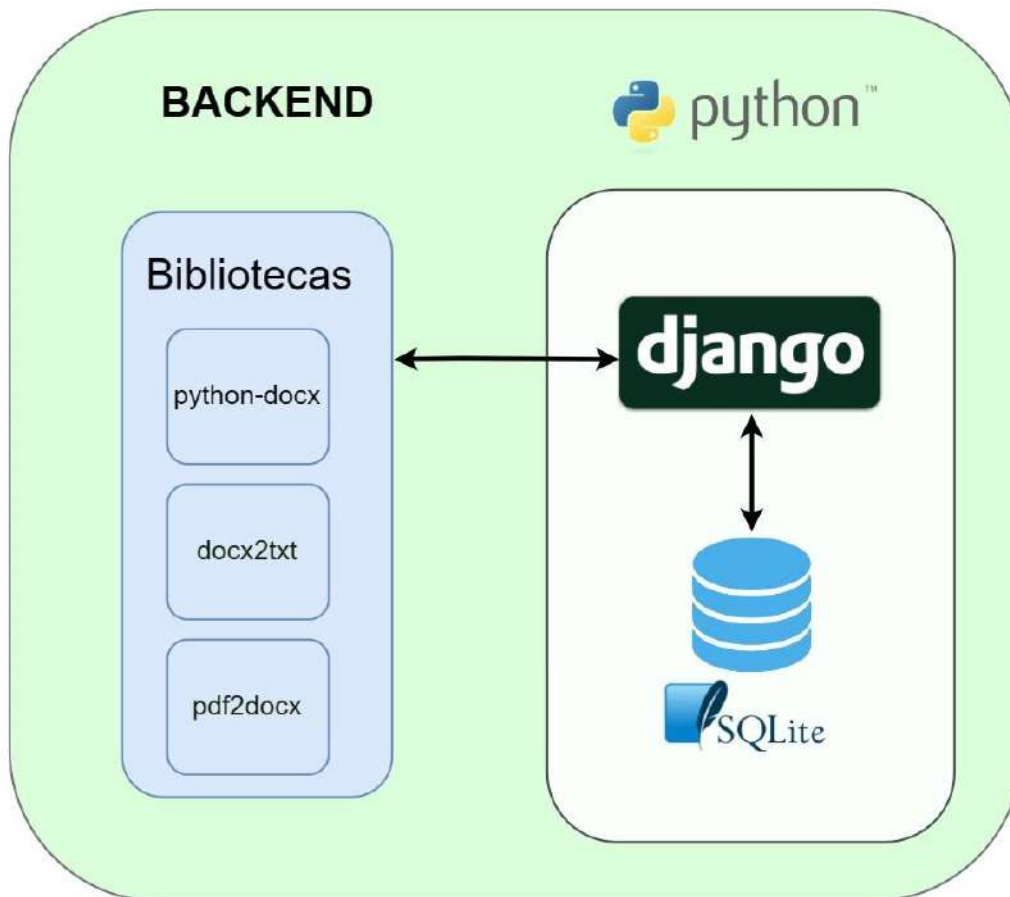


Fig. 19: Composición de la estructura *backend* de la aplicación web.

Para el desarrollo del *backend*, se decidió utilizar el lenguaje de programación *Python*, siendo un lenguaje que se suele enseñar en distintos ambientes de enseñanza para desarrollos web, como sucede en la Facultad de Informática de la UNLP [53]. Junto con *Python*, se hará uso de *framework Django* [54] que es muy popular para el desarrollo web [55]. Es una herramienta de alto nivel, es decir, que abstrae muchas tareas y detalles técnicos complejos del desarrollo web. Además, es de código abierto y está escrito en *Python*. Este framework tiene varias características que la hacen ideal para el desarrollo de páginas web, algunas de ellas son:

- **Productividad:** Proporciona una estructura y una arquitectura bien definidas que facilitan el desarrollo web, y ofrece una amplia gama de características integradas que aceleran el proceso de desarrollo tales como la generación de formularios, autenticación de usuario y administración de bases de datos.
- **Seguridad:** Django provee múltiples herramientas y mecanismos de protección contra ataques como inyección de SQL (lenguaje de consultas sobre bases de

datos), ofrece un sistema de autenticación y autorización robusto, tiene un sistema de validación de formularios incorporado, entre otros mecanismos de protección.

- Escalabilidad: Está diseñado para manejar aplicaciones web de gran escala, por lo que utiliza un enfoque basado en componentes reutilizables, lo que permite que el código sea modular y fácilmente escalable.
- Comunidad y documentación: Existe una gran comunidad de desarrolladores activos [56] que contribuyen con soluciones a problemas comunes y con nuevas implementaciones de código.
- Versatilidad: Se adapta a una amplia gama de proyectos web pudiendo construirse todo tipo de aplicaciones y sitios web, además de ser compatible con una variedad de bases de datos donde se incluye *SQLite* [57].
- Adherencia a estándares: Se adhiere a las mejores prácticas y estándares de desarrollo web, lo que promueve un código limpio, bien estructurado y fácil de mantener.

Respecto de la popularidad, en la siguiente figura (Fig. 20) se observa el resultado de una encuesta del sitio web *Statista* donde se encuentran los frameworks más populares a nivel mundial en 2022 [58].

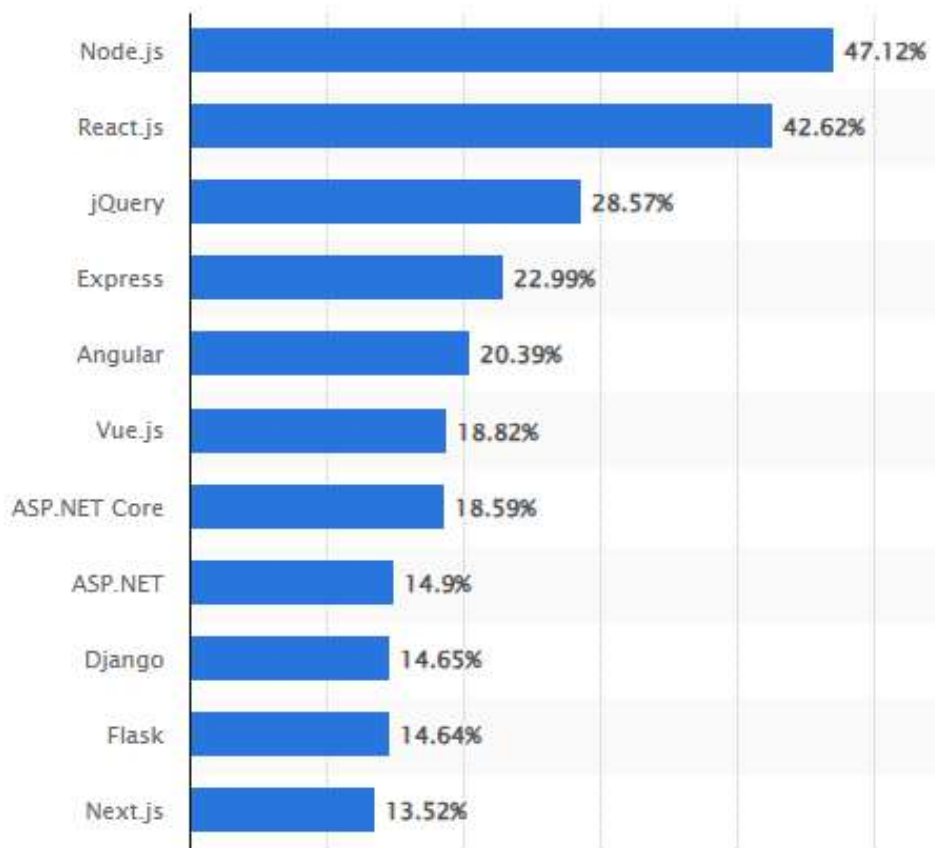


Fig. 20: La popularidad de varios frameworks web en 2022.

Fuente: [<https://www.statista.com>]

Este *Framework* nos permitirá crear un ambiente seguro donde se ejecute la aplicación y se pueda almacenar datos e información, como, por ejemplo, documentos. De esta manera, podremos manipular documentos de una manera sencilla.

Teniendo en cuenta las características de *Django* mencionadas anteriormente y su información relacionada, podemos concluir en que es un *framework* poderoso y eficiente, y es la opción más popular en la actualidad con el lenguaje *Python*, para el desarrollo de páginas web, y además satisface los requerimientos para un desarrollo web completo y seguro lo cual es ideal para el prototipo AcceDocx.

Para el almacenamiento de información relacionada con la aplicación y su actividad con el usuario y sus archivos, se decidió utilizar la tecnología *SQLite*, un gestor de bases de datos relacional que se destaca por ser liviano, rápido, fácil de configurar y compatible con el lenguaje SQL. *SQLite* es una base de datos incorporada en la mayoría de las instalaciones de *Python* y no requiere una instalación o configuración adicional. Es adecuado para aplicaciones web pequeñas o medianas con un volumen de datos

moderado y no requiere un servidor de base de datos separado. En este sentido, una herramienta que provee de manera predeterminada esta tecnología es el *framework Django*, además de que su configuración se aplica sobre un proyecto desde su inicio, de manera que no se necesitará instalar ningún otro componente adicional para comenzar a programar. Con la necesidad de almacenar la información de un documento, se diseña el siguiente modelo (Fig. 21):

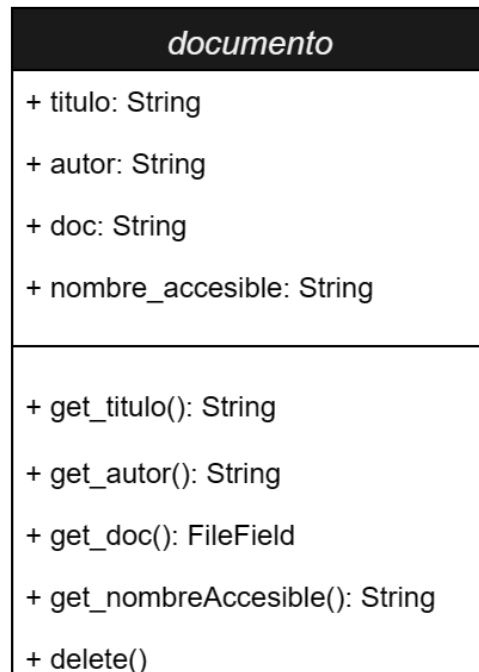


Fig. 21: Representación del documento en la base de datos SQLite.

Teniendo en cuenta lo anterior, *SQLite* es un gestor de bases de datos suficiente para el lograr desarrollar el prototipo.

De esta manera, queda definido el diseño y las tecnologías para el desarrollo web, así como también los tipos de documentos que serán aceptados por el sistema. La siguiente figura (Fig. 22) muestra la arquitectura resultante del prototipo, a partir de lo mencionado anteriormente.

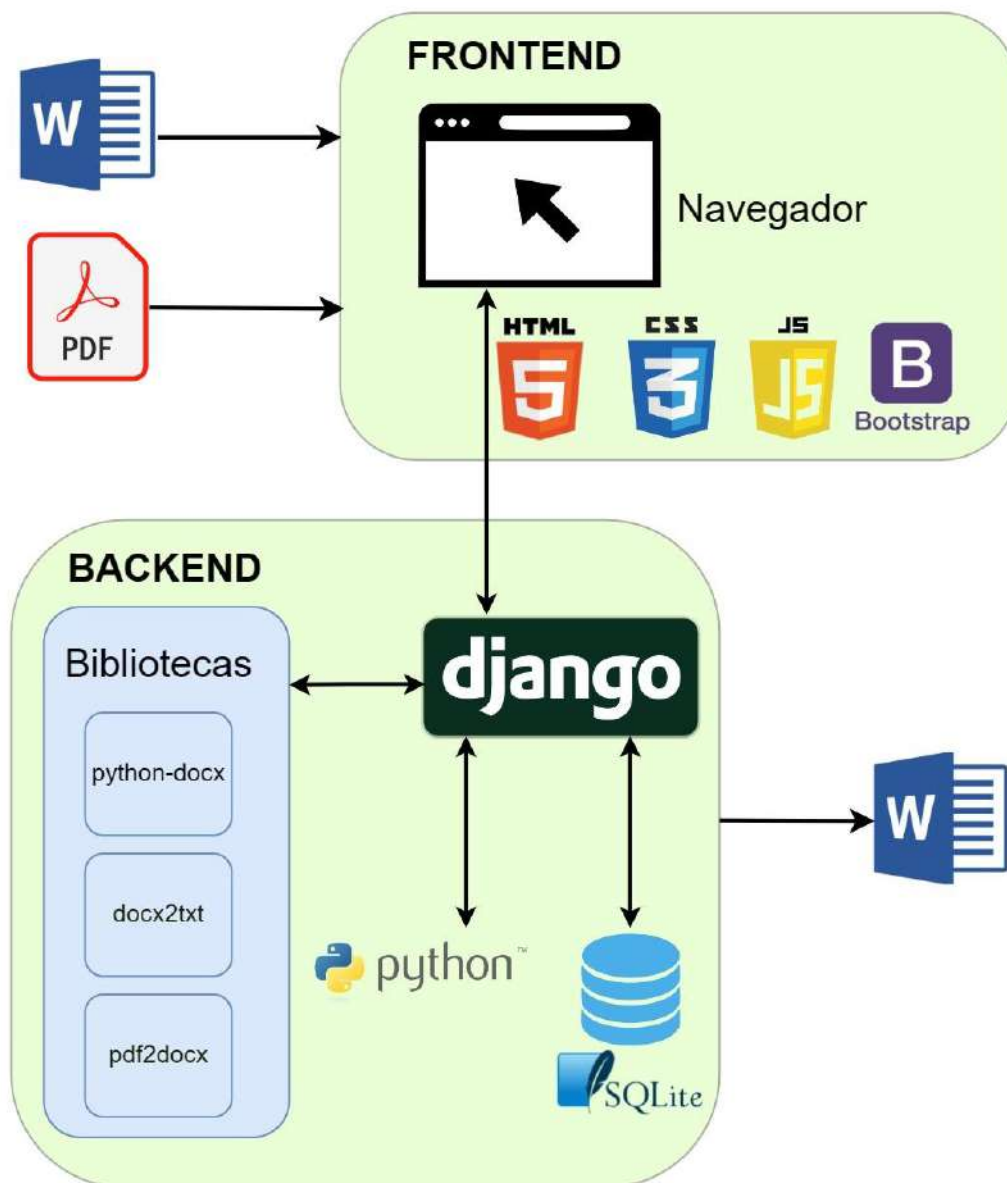


Fig. 22: Arquitectura del prototipo.

4.3 Lógica de conversión del prototipo

La herramienta de conversión funciona por detrás del diseño web que se visualiza, y es la encargada de realizar de manera lógica la conversión de un archivo digital de lectura a un archivo accesible. Este desarrollo agrupa varios archivos generados para este propósito. Para llevar a cabo el desarrollo programático de esta parte de AcceDocx, se decidió continuar utilizando *Python* junto con bibliotecas para el mismo lenguaje. Estas bibliotecas no son más que tecnologías que proporcionan un conjunto de

funcionalidades. Las bibliotecas principales que se utilizan para desarrollar esta parte lógica son 3:

- 1) python-docx: Esta biblioteca de *Python* se concentra en los archivos con extensión “.docx”, y sus características fueron mencionadas en el capítulo anterior.
- 2) docx2txt [59]: Biblioteca de *Python* que permite extraer el texto contenido en documentos de *Microsoft Word* en formato *DOCX*, y adicionalmente, las imágenes que existan son detectadas y guardadas en alguna carpeta destino definida. Proporciona una forma sencilla de convertir documentos de Word en texto plano, lo que resulta útil en aplicaciones donde solo se necesita extraer el contenido de un archivo de Word sin conservar el formato o estilos. Se puede instalar usando el gestor de paquetes de *Python* “pip” mediante el siguiente comando: `pip install docx2txt`.
- 3) docx2PDF [60]: Es una biblioteca de *Python* que permite convertir archivos de *Microsoft Word* en formato *DOCX* hacia el formato *PDF*. Proporciona una forma de convertir documentos de Word en archivos *PDF* donde no es necesario tener instalado *Microsoft Word* o algún otro *software* adicional. La funcionalidad principal de *docx2PDF* es tomar un archivo *DOCX* existente y generar un archivo *PDF* correspondiente. Esta conversión conserva el contenido, el formato y los elementos del documento original. Para instalar esta biblioteca, se puede utilizar el gestor de paquetes “pip” y luego ejecutar el siguiente comando: `pip install docx2PDF`.

A partir de las funcionalidades que ofrece *python-docx* para manejar el contenido de un *DOCX* y teniendo en cuenta el estudio de las recomendaciones de accesibilidad para documentos basados en la WCAG 2.1, dentro del proceso de conversión se establecieron los formatos sobre los textos y párrafos con valores adecuados, tales como el estilo Arial y tamaño 12 para textos, como así el interlineado de 1.5 para párrafos. Lo que se obtiene de la programación que resuelve este escenario de accesibilidad, es un conjunto de archivos que se relacionan con la ejecución principal de la aplicación web, comportándose como módulos del programa principal.

Las bibliotecas como *python-docx*, se utilizan para procesar contenido de documentos y se orientan para la creación de un nuevo documento y no de la modificación del mismo. Por esto tanto, la lógica implementada a través de 2 archivos principales, representan 2

operaciones: Procesar y Convertir. Para que estos archivos realicen su trabajo, deben existir documentos en el sistema. En este sentido, la primera operación procederá a revisar el contenido del documento y rescatar el contenido del mismo. De esta forma, la lógica implica, por un lado, el rescate de las imágenes en una carpeta, y por otro lado el rescate de texto y de estructuras. Toda la información sobre el documento se mantiene almacenado en un archivo de texto plano, el cual sirve de fuente de consulta para otra parte de la lógica implementada. La segunda operación, procede a generar un nuevo documento con el contenido del mismo desde su archivo origen, con las imágenes y los textos que la componen. Además, sobre el contenido de este nuevo documento, se aplican los criterios de accesibilidad que el usuario haya decidido.

La lógica desarrollada permite detectar y procesar los siguientes elementos y características del contenido de un documento:

- Títulos
- Párrafos
- Estilo de fuente
- Tamaño de fuente
- Interlineado
- Imágenes
- Hipervínculos
- Listas

Además, permite transformar un documento del formato *PDF* a *DOCX*, así como también colocar el título o nombre del archivo y el autor al documento resultante dentro de sus propiedades. En el mismo sentido, se agregó una acción sobre los documentos en el sistema, que permita la descarga del archivo *DOCX* en formato *PDF*.

La lógica de este desarrollo, se basa en el uso fundamental de la biblioteca *python-docx* que sirve de puerta de acceso a un documento *DOCX*, y donde sus características principales fueron mencionadas en el capítulo anterior. Esta tecnología permite procesar el contenido de un documento, y poder gestionar cada componente del mismo, aunque a pesar de esto, se presentan distintas dificultades a la hora de modificar el contenido, principalmente las siguientes 3:

- 1) El tipo de listas: Si bien las listas en un documento .docx son detectables por esta tecnología, no se pueden diferenciar entre listas numeradas o listas no-numeradas.
- 2) Texto alternativo: Si bien las imágenes se detectan, la tecnología no tiene acceso aún a la propiedad de la imagen de texto alternativo, por lo que se puede implementar de manera alternativa un párrafo debajo de cada imagen.
- 3) Los hipervínculos: La información del hipervínculo es detectable y también su creación, pero hay dificultad lógica para la colocación de ese elemento dentro de un párrafo.

El diseño de esta lógica programada tiene como objetivo principal cubrir las funcionalidades mencionadas anteriormente, permitiendo gestionar tanto el contenido del documento en los puntos previamente mencionados como también la información y el tipo de archivo resultante.

En el siguiente capítulo se podrá observar el funcionamiento del prototipo, pudiendo además observar parte de los componentes alcanzados.

Capítulo 5: Funcionamiento de AcceDocx.

En este capítulo se detalla el funcionamiento del producto final desarrollado. De modo que se describirán con mayor detalle las funcionalidades existentes y su uso. Para hacer uso de AcceDocx y ponerlo en marcha se requiere de unos pasos previos definidos en el Anexo 1 “Inicialización de AcceDocx”.

Las funcionalidades que fueron definidas previamente, se lograron implementar y como resultado se describen a continuación junto con figuras que apoyan la usabilidad del mismo dentro de la página:

- 1) Listado de documentos: Para listar los documentos existentes en el sistema se accede a la sección Documentos a partir de un botón con el mismo nombre. Al pulsar sobre éste, se puede visualizar los documentos existentes en la plataforma y las funciones permitidas sobre cada uno de ellos. En la siguiente figura (Fig. 23) se puede observar esta sección.



Fig. 23: Sección Listado de documentos almacenados en el sistema.

Los elementos a mencionar son los siguientes:

- Botón Documentos (1): Pulsando sobre él accedemos al listado.
- Tabla (2): La tabla presenta las filas con los documentos que serán subidos al sistema, junto con los encabezados que determinan la información disponible y las acciones que se pueden realizar sobre cada uno.

- 2) Carga de documentos en formato *DOCX*: Para cargar un documento primero se listan los documentos (proceso descrito en el punto anterior). En la siguiente figura (Fig. 24) se observa la pantalla que permite realizar este primer paso.



Fig. 24: Acceso a la carga de documentos formato *DOCX*.

Los elementos que aparecen en esta página se describen a continuación:

- Listado de Documentos (1): La página es la correcta, donde se encuentran listados los archivos cargados en el sistema.
- Botón Subir documento (2): Pulsando sobre él iniciamos el primer paso para realizar la carga de un documento. Se accede a una nueva pantalla.

Un segundo paso para lograr el objetivo de realizar la carga de un documento, se procede a una pantalla con un cuadro descriptivo con el procedimiento que se iniciará. Este cuadro se observa en la siguiente figura (Fig. 25).

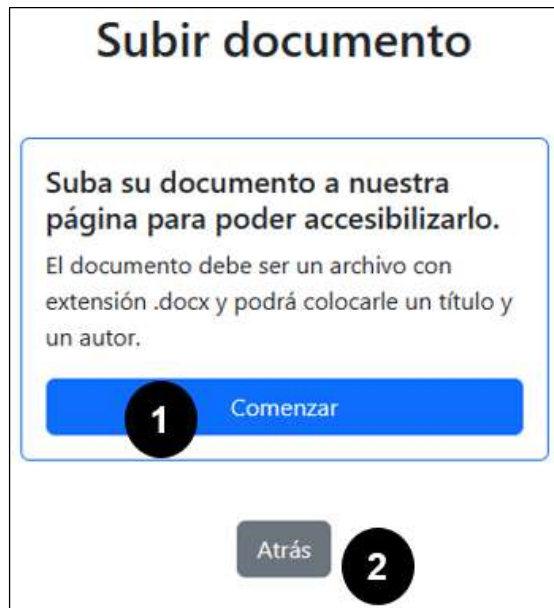


Fig. 25: Cuadro descriptivo para iniciar la carga de un documento.

Los elementos a mencionar son los siguientes:

- Botón Comenzar (1): Pulsando sobre él se inicia el proceso de subir un documento.
- Botón Atrás (2): Vuelve a la página anterior.

Si el usuario da inicio al proceso de carga de ese documento, accede al formulario respectivo en la siguiente figura (Fig. 26), en cual se debe completar para finalizar la carga del archivo al sistema. El mismo incluye la generación de un título y un autor sobre el documento.

Fig. 26: Formulario para la carga de documento *DOCX*.

Este formulario se compone de la siguiente manera:

- Campo Título (1): En él se ingresa el título con el que será guardado el archivo en la computadora y como será mostrado en la sección del listado de documentos.
- Campo Autor (2): En él se ingresa el autor del documento, el cual será almacenado en las propiedades del archivo real.
- Botón Elegir archivo (3): Este botón permite seleccionar el archivo que se cargará en el sistema.
- Botón Cancelar (4): Presionando sobre él cancelamos la operación actual, volviendo a la pantalla inmediatamente anterior a esta.
- Botón Subir (5): Presionando sobre él realizamos la carga del documento en el sistema, siendo redireccionado al listado de documentos.

- 3) Eliminación de un documento cargado: La eliminación de un documento se realiza a través de las opciones disponibles en cada fila de la tabla, dentro de la sección del Listado de Documentos. En la siguiente figura (Fig. 27) se aprecia lo descrito:



TÍTULO	AUTOR	ACCIONES
Clase 1 - Archivos	Capacitador David	ELIMINAR ACCESIBILIZAR DESCARGAR

Fig. 27: Acciones disponibles sobre un documento cargado en el sistema.

El botón correspondiente a esta operación será el llamado “Eliminar”.

- 4) Accesibilizar documento: Para realizar el proceso de “Accesibilizar” un documento, se accede a partir de las acciones permitidas sobre los documentos listados en la tabla general de documentos incorporados en el sistema. Esta posibilidad se observa en la figura anterior (Fig. 27).

Al momento de iniciar el proceso de accesibilizar el documento, se presenta una pantalla con información del documento como se observa en la siguiente figura (Fig. 28).



Fig. 28: Información del documento que se va a accesibilizar.

Esta misma pantalla, además nos presenta los elementos dentro del documento detectados por la herramienta. Estos se representan a través de dos componentes: un posible título o encabezado, y una imagen.

Sobre cada una de estas secciones generadas a través de la ejecución de la lógica de procesamiento de la herramienta sobre el documento, debemos tomar decisiones. La siguiente figura (Fig. 29) muestra una sección para trabajar sobre un posible título:



Fig. 29: Sección para definir un posible título.

Esta sección se compone de la siguiente manera:

- Cadena detectada (1): Aquí aparece la cadena de caracteres que fue detectado por la herramienta como un posible título.

- Botón de opción (2): Para el posible título o encabezado detectado elegimos una opción de selector: SI, en caso que sea tratado como un título (encabezado), caso contrario, se selecciona el NO. Conlleva la etiqueta “¿Es un título?”.
- Selector de opciones (3): Sobre él se elige su nivel de Título de los existentes (0, 1, 2, 3, 4) en un documento de tipo *DOCX*. Conlleva la etiqueta “Niveles de títulos” con una imagen representativa de los posibles títulos a elegir.

Para la imagen detectada se puede apreciar la siguiente sección (Fig. 30):



Fig. 30: Sección para colocar una descripción a una imagen.

Esta sección se compone de la siguiente manera:

- Imagen (1): Aquí aparece la imagen detectada por la herramienta.
- Campo descripción (2): Aquí se ingresa la descripción correspondiente a la imagen. Sobre este campo existe una etiqueta a modo de título del campo con la frase “Ingrese una descripción”.

En la parte inferior de la página, existe tenemos opciones para finalizar el proceso de accesibilizar. Esto se puede observar en la siguiente figura (Fig. 31).



Fig. 31: Opciones para finalizar el proceso de accesibilizar un documento.

- 5) Conversión de archivo *PDF* a *DOCX*: En caso de tener un documento en formato *PDF*, la herramienta nos permite convertirlo en formato *DOCX*. Esta opción llamada “Convertir *PDF*” se encuentra en la barra de navegación superior el cual se puede observar en la siguiente figura (Fig. 32):



Fig. 32: Barra de navegación superior.

Esto resulta de utilidad para el momento en que tenemos un documento inicial con este primer formato, ya que el proceso de carga de documentos y el de accesibilizar un documento solamente se realiza sobre documentos *DOCX*. A continuación, tenemos una siguiente pantalla con una información preliminar sobre el proceso que se llevará a cabo y los requerimientos para que funcione correctamente.

En la siguiente figura (Fig. 33) se hace foco principalmente en el componente de la página recién mencionada anteriormente.



Fig. 33: Pantalla preliminar al proceso de conversión de *PDF* a *DOCX*.

Existe el botón “Comenzar” el cual permitirá acceder a un breve formulario para la conversión de *PDF* a *DOCX*. En la siguiente figura (Fig. 34) se observa el formulario para realizar la conversión:

La imagen muestra un formulario con el título "Convertir PDF". El formulario contiene los siguientes elementos: un campo de texto etiquetado "Título:" con un número 1 en un círculo negro; un campo de texto etiquetado "Autor:" con un número 2 en un círculo negro; un botón etiquetado "Elegir archivo" con un número 3 en un círculo negro, y el texto "No se eligió ningún archivo" a su derecha; un botón rojo etiquetado "Cancelar" con un número 4 en un círculo negro; y un botón azul etiquetado "Convertir" con un número 5 en un círculo negro.

Fig. 34: Formulario para conversión de *PDF* a *DOCX*.

Este formulario se compone de la siguiente manera:

- Campo Título (1): En él se ingresa el título con el que será guardado el archivo en la computadora.

- Campo Autor (2): En él se ingresa el autor del documento, el cual será almacenado en las propiedades del archivo físico.
 - Botón Elegir archivo (3): Este botón permite seleccionar el archivo con formato *PDF* que será convertido al formato *DOCX*.
 - Botón Cancelar (4): Presionando sobre él cancelamos la operación actual, volviendo a la pantalla inmediatamente anterior a esta.
 - Botón Convertir (5): Presionando sobre él realizamos la conversión del documento y su almacenamiento en una carpeta contenedor específica del sistema. Luego seremos redireccionados al listado de documentos.
- 6) Descargar documento: En caso querer descargar el documento accesible, la herramienta nos permite hacerlo a través de su botón “Descargar”. El mismo se puede observar en la figura 27 (Fig .27) donde se presentan las acciones disponibles sobre un documento cargado en el sistema.

A partir del uso de *AcceDocx* y del proceso de desarrollo, se observó que ciertos componentes y características eran detectables por *python-docx*, y otros tuvieron los resultados esperados a partir de un análisis previo del alcance de la biblioteca sobre los documentos *DOCX*, estos se mencionaron en el capítulo anterior. En este sentido, luego de implementar el prototipo y teniendo en cuenta el alcance de la biblioteca, ciertos componentes, que pueden existir dentro de un documento tales como Tablas, Índice, Encabezado y Pie de página, son detectables y se podría trabajar sobre ellos, con una mayor complejidad lógica por analizarse. De esta manera, que estos componentes que no fueron alcanzados en esta etapa de inicial de desarrollo, quedan como parte de una futura implementación para un trabajo siguiente.

En el Anexo 2 “Componentes detectables con potencial implementación” se puede visualizar una investigación preliminar sobre estos componentes teniendo la posibilidad de una pronta inclusión al prototipo.

Capítulo 6: Análisis sobre AcceDocx

Los test de accesibilidad son herramientas y procesos que se utilizan para evaluar la accesibilidad de un sitio web, aplicación o sistema digital. La importancia radica en garantizar la igualdad de oportunidades y el acceso equitativo a la información y los servicios en línea para todas las personas. Al realizar test de accesibilidad, se pueden identificar barreras y deficiencias en la usabilidad de un sistema, permitiendo su corrección y mejoramiento. Los mecanismos de evaluación de la accesibilidad pueden ser automáticos, manuales o basados en la interacción con usuarios, y se pueden combinar para obtener una evaluación más completa. Desde un enfoque de evaluación automática, se puede buscar problemas de accesibilidad en el sitio web a partir del uso de herramientas y software especializado que realizan un análisis automático. Desde un enfoque manual, los evaluadores revisan y prueban manualmente un sistema. De esta manera se busca identificar problemas de accesibilidad, pudiendo verificar elementos como la navegación, la estructura de contenido y otros aspectos que no son fácilmente detectables por herramientas automáticas. Por último, desde una evaluación con usuarios, se permite involucrar a usuarios reales con discapacidades, lo cual proporciona retroalimentación sobre usabilidad y accesibilidad del sistema, brindando una perspectiva más auténtica y valiosa sobre cómo las personas con discapacidades interactúan con el sistema.

Teniendo en cuenta lo anterior, a continuación, se describe el análisis del prototipo desarrollado, frente a 3 puntos de vista o enfoques: desde la accesibilidad de la herramienta, desde la perspectiva del capacitador, y desde la experiencia del uso de la misma con una persona adecuada.

6.1 Análisis de accesibilidad de AcceDocx

En esta sección, se describe un análisis enfocado en la accesibilidad del prototipo y sobre los documentos resultantes.

El prototipo web Accedocx fue analizado sobre su accesibilidad según las normas establecidas sobre la WCAG, como fue descrito anteriormente en el momento de desarrollarla, de manera que a continuación se presentan los análisis y resultados de los mismos.

Análisis de accesibilidad web sobre AcceDocx.

El prototipo web Accedocx fue testado por usuarios que, al realizar test de usabilidad sobre ella que se presenta a continuación, han mostrado conformidad sobre la navegación y su interfaz.

Para complementar el análisis del prototipo sobre la accesibilidad que posee, se pone a prueba al mismo, utilizando una herramienta de evaluación de accesibilidad web llamada Wave Web Accessibility [61] en su última versión actualizada en marzo 2023. Wave es una herramienta en línea que permite a los usuarios evaluar la accesibilidad de un sitio web, basándose en las pautas de accesibilidad web establecidas por el Consorcio World Wide Web (W3C). Esta herramienta realiza un análisis completo de la página web y presenta un informe detallado que identifica los problemas de accesibilidad que deben abordarse ofreciendo sugerencias y soluciones para resolver los problemas de accesibilidad identificados.

La evaluación del prototipo con Wave se llevó a cabo sobre dos de las páginas principales de la aplicación web: la página del “Listado de documentos”, y sobre la página para Accesibilizar un documento, llamada “Accesibilizando”:

- Evaluación sobre Listado de documentos: El resultado de la prueba de accesibilidad, sobre la página donde se visualiza el listado de documentos en el sistema, arroja un resultado que se presenta a continuación en la siguiente figura (Fig. 35):

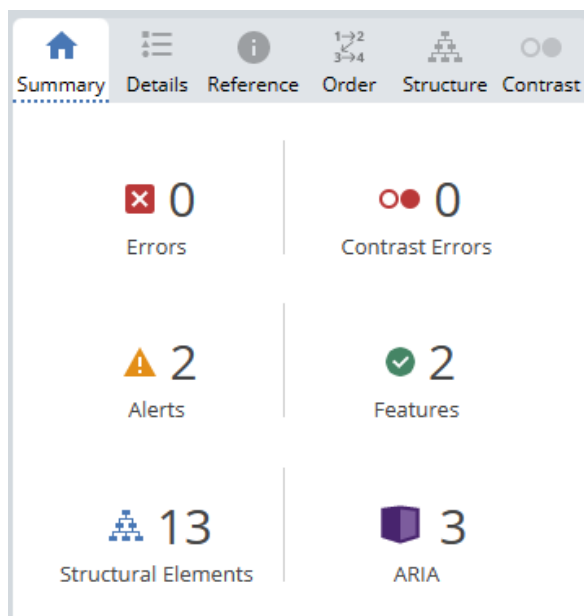


Fig. 35: Resumen del resultado en Wave sobre la página con el listado de documentos.

En la figura anterior se puede observar que el resultado arrojado demuestra que la página no presenta errores. Detecta las estructuras y distintas características del armado de la página. Solamente se observa simples alertas que nacen de los vínculos que existen hacia archivos, que son los cuales existen en la tabla que engloba los distintos documentos que existen en el sistema.

- Evaluación sobre Accesibilizando: Se procede a realizar la prueba de accesibilidad, sobre la página donde se aplica los criterios de accesibilidad del usuario sobre los elementos detectados en el documento. En la siguiente figura (Fig. 36) se puede visualizar de manera general la página para accesibilizar un documento, en este caso en el listado de documentos en el sistema, arroja un resultado que se presenta a continuación:



Fig. 36: Prueba de accesibilidad con Wave sobre la página con el proceso de accesibilizar un documento.

Se observa en la figura anterior que sobre cualquier documento se obtiene un resultado aceptable sobre la prueba realizada. A continuación, se observa de manera detallada los resultados (Fig. 37):

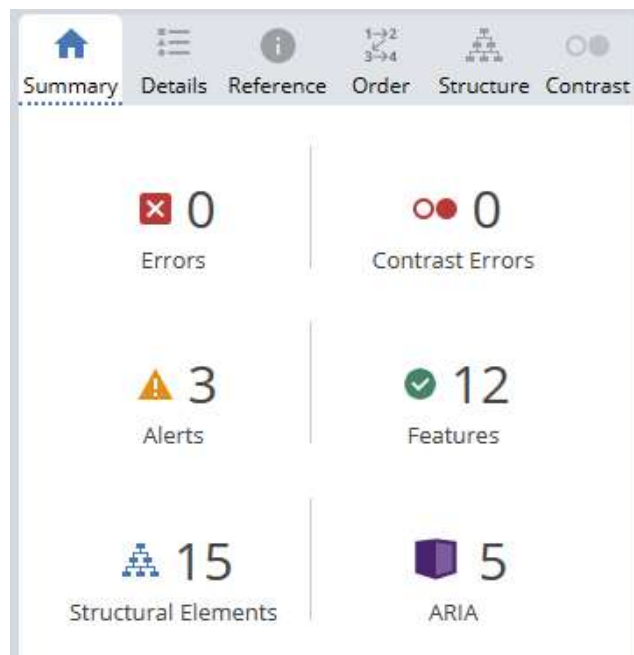


Fig. 37: Resumen del resultado en Wave sobre la página con el proceso de accesibilizar un documento.

Frente a los resultados obtenidos en la figura anterior se observa que no existen errores. La existencia de alertas es sobre la detección del link hacia un elemento web en formato *PDF*, mencionado anteriormente; y otra alerta sobre alguna

etiqueta que fue detectada como elemento con falta de información en cuanto a su identificador.

Los resultados a partir de la prueba con la herramienta Wave fueron favorables hacia la obtención de una herramienta web que permita accesibilizar documentos digitales de manera automática.

En cuanto al análisis de su estructura, no fueron detectados errores y las advertencias tienen una solución considerablemente sencilla pudiendo modificar y agregar algunos elementos HTML, quedando como actividad futura aplicar una nueva revisión del código y una nueva prueba con Wave.

Sobre la experiencia del resultado final y el propósito del proyecto, el documento resultante es leído por un lector de pantalla de manera natural, y pudiendo dejar un documento uniforme en cuanto a la estructura del mismo y su diseño visual.

Por consecuencia de lo mencionado anteriormente, y sobre los resultados sobre el uso de la herramienta por los usuarios, se puede considerar que el prototipo web, AcceDocx es en una gran parte accesible. De manera que la navegación sobre ella no lograría generar inconvenientes tanto de manera visual y comprensiva, como así tampoco para las herramientas de apoyo como son los lectores de pantalla.

Análisis de Accesibilidad sobre documentos generados

Como primera acción se realiza el proceso de accesibilizar el documento mediante la herramienta, en este caso un capacitador realiza esta acción y se genera finalmente el documento resultante el cual es analizado según sus componentes detectados y luego analizar la accesibilidad del mismo. En principio, se obtienen las siguientes consideraciones sobre el documento resultante:

- En cuanto a las estructuras de texto como párrafos y títulos, no existieron inconvenientes logrando cumplir con los criterios de accesibilidad.
- En particular, el espaciado de las líneas de textos y párrafos, los interlineados, resultaron expresarse en el documento de manera correcta, respetando los criterios de accesibilidad.
- Además, los tamaños y estilos sobre los textos fueron resueltos por la herramienta de manera satisfactoria respetando los criterios de accesibilidad.

- Como la herramienta genera un documento sin formato de color sobre los fondos y contrastes, no existieron inconvenientes en ese aspecto.
- Sobre los elementos de listas, se pudieron traspasar desde el documento original hacia el resultante.
- Los hipervínculos fueron rescatados y colocados dentro del documento resultante.
- Un detalle encontrado en sobre las listas que existen en un documento, es que no es posible aún, con la biblioteca *python-docx*, diferenciar entre listas ordenadas o no-ordenadas. En este sentido, se llevó a cabo la transformación de todas las listas existentes en el documento origen hacia un formato de lista no-numeradas en el documento resultante.
- Si bien las imágenes fueron detectadas, la descripción de la imagen se realiza mediante la colocación de un párrafo debajo de la imagen. Incluir un texto alternativo sobre la imagen no es una función que exista aún dentro de la tecnología utilizada *python-docx*.
- Sobre los hipervínculos dentro de un párrafo, se lograron rescatar tanto el nombre del link colocado, si es que tiene, y su dirección asociada. Son esta información es detectable, pero al momento de la conversión son colocados luego de finalizar el párrafo donde fueron detectados.
- Los inconvenientes encontrados fueron sobre componentes especiales que existían dentro del documento como tablas, en este caso, se perdía el formato de manera detallada de las líneas delimitadoras. También el encabezado y pie de página no es abarcado con la herramienta, como resultado de la prueba.

Sobre este documento generado, a partir del proceso que realiza el prototipo, se debió realizar una comprobación de la accesibilidad que poseen los mismos. En este caso, se puso a prueba a través de la plataforma web PAVE la accesibilidad de los mismos, esta herramienta fue mencionada anteriormente y trabaja sobre archivos *PDF*.

Como primer paso, se realizó la conversión del archivo *DOCX* resultante del prototipo, a un formato *PDF*. Este documento se compone de títulos (encabezados), párrafos, imágenes e hipervínculos.

Como siguiente paso se utilizó PAVE para subir el archivo sobre su sistema online como se muestra en la siguiente figura (Fig. 38), por consecuencia se ejecutó el proceso de análisis del documento para verificar su accesibilidad.



Fig. 38 Contenido del documento cargado en la plataforma PAVE.

Por último, se pudo recuperar la información que devuelve PAVE sobre el documento, tanto las propiedades del documento como las estructuras de textos tuvieron un resultado satisfactorio como se muestra en la siguiente figura (Fig. 39):



Fig. 39: Resumen del resultado sobre las propiedades del documento.

Los resultados sobre las imágenes resultaron la detección de falta de texto alternativo como se muestra en la siguiente figura (Fig. 40), siendo este resultado esperado, ya que el prototipo solo permite agregar una descripción a cada imagen como texto debajo de cada imagen. No permite la colocación de texto alternativo sobre la imagen porque la biblioteca *python-docx* aún no tiene esa funcionalidad disponible.

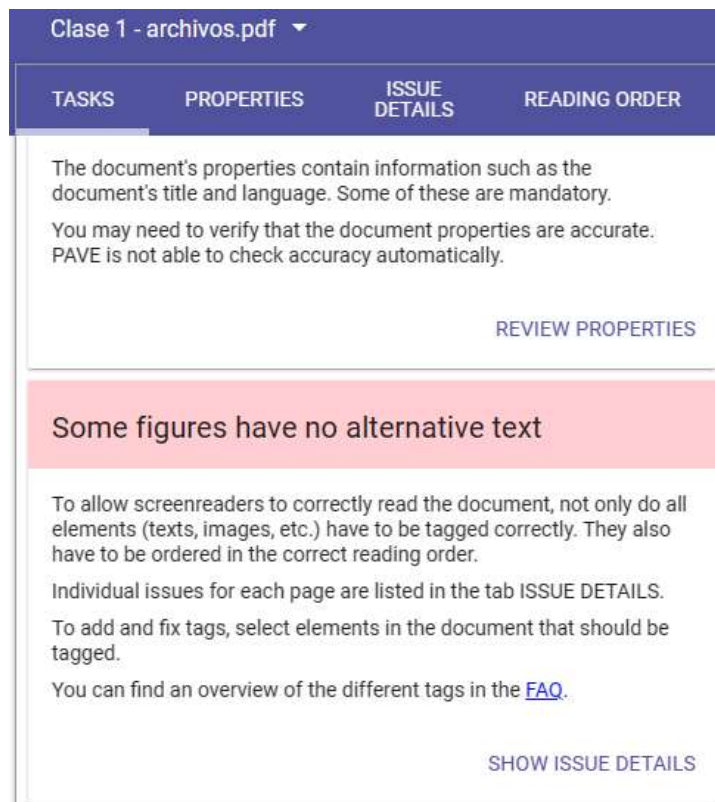


Fig. 40: Resumen del resultado sobre las imágenes del documento.

Teniendo en cuenta los resultados obtenidos sobre la accesibilidad sobre un documento, el mismo estaba compuesto de componentes o elementos que estaban alcanzados por el prototipo y otros elementos que no. En cuanto, a los componentes alcanzados, fueron correctamente detectados y pudieron ser manipulados durante todo el proceso, con algunos detalles mencionados anteriormente. Sobre los componentes no alcanzados, se pudieron lograr detectar desde la lógica y la biblioteca *python-docx* hace mención a la posibilidad de manipularlos. Estos componentes se encuentran descritos en el Anexo 2 “Componentes detectables con potencial implementación” junto con una información al respecto. La incorporación del manejo de estos elementos no alcanzados en esta etapa inicial del proyecto será parte de un proceso de mejora de la herramienta que se podría implementar como trabajo futuro.

6.2 Análisis desde la perspectiva del Capacitador

En esta sección se describe el caso de estudio elegido donde se probó el prototipo y su impacto sobre el mismo.

Frente a la problemática planteada inicialmente, con el propósito de que el capacitador pueda utilizar la herramienta para accesibilizar documentos, se llevó a cabo la prueba del prototipo.

Pruebas de usuario

Se tomó a una cantidad de personas reducida, considerando ser lo suficientemente aceptable para la cantidad total existente dentro del ambiente del caso de estudio. Este proceso tendrá la posibilidad de luego ser replicado en otros ambientes, para una mayor cantidad de personas; dentro de una extensión del trabajo en un futuro.

El proceso de prueba se realizó como prueba funcional del software y se compuso de las siguientes especificaciones:

- Requisitos: se basan en las funcionalidades de la herramienta que se ha desarrollado.
- Diseño y planificación: los casos de prueba tendrán una descripción de la funcionalidad que se va a probar y los pasos necesarios.
- Ejecución y registro: se llevará a cabo la acción de los casos de prueba y se obtendrán los resultados.
- Análisis y revisión: a partir de los resultados se analizarán cada uno de ellos y se generará una revisión final de esta actividad.

Cada etapa del proceso de pruebas se describe a continuación.

Requisitos

Anteriormente, se pudieron determinar las funcionalidades de la herramienta necesarias para lograr el objetivo de accesibilizar un documento en formato *DOCX*. Además, se

dispuso de los ordenadores necesarios y personal adecuado con los que se llevó a cabo esta actividad.

Diseño y planificación

Los usuarios que conforman la población para las pruebas son una porción del personal del Programa de Oportunidades de Formación en el Centro de Inclusión Libre y Solidario de Argentina (CILSA), en particular, los capacitadores con rol de profesor/ra encargados de realizar el dictado de clases de cursos dentro del establecimiento. Los capacitadores involucrados en estas pruebas poseen distintas características, las mismas se visualizan en una tabla a continuación:

Tabla de información de los profesores

Profesor / Datos	Edad	Conocimientos de PC	Conocimientos en accesibilidad	Curso	Perfil
Juan	E3	Básico	Básico	Operador de PC	Asesor lingüístico. Profesor en educación especial. Discapacidad auditiva.
Jesús	E2	Intermedio	Básico	JavaScript intensivo	Analista de Sistemas. 3 años de experiencia. Conocimientos básicos en distintos lenguajes de programación.
Julio	E3	Básico	Básico	Introducción al desarrollo web	Analista en Informática Aplicada. Instructor en cursos. Experiencia superior a 3 años.
David	E2	Intermedio	Ninguno	Programación en Python	Estudiante de Sistemas. Conocimientos avanzados en programación. 1 año de experiencia.

Aclaración sobre valores en la tabla:

- 1) Columna Edad, contiene los siguientes valores:
 - E1 = edad menor a 30
 - E2 = edad entre 30 y 40
 - E3 = edad entre 40 y 50
 - E4 = edad mayor a 50
- 2) Columnas que implica conocimientos, con los posibles siguientes valores:
 - Ninguno
 - Básico
 - Intermedio
 - Avanzado

Como metodología para las pruebas, se decidió tomar a 4 capacitadores, cada uno con acceso al prototipo AcceDocx utilizando un ordenador y con la disponibilidad de un conjunto de archivos pertenecientes al material de distintos cursos. Se definieron 4 acciones a realizarse: Subir un documento, Accesibilizar un documento, Descarga del Archivo y Convertir un archivo *PDF* a *DOCX*.

Además, los profesores tendrían como actividad completar un cuestionario respecto del uso de la herramienta, específicamente una encuesta de estilo SUS (Simple Usability Scale), por lo que deberían responder 10 preguntas enfocadas en la usabilidad. El cuestionario se encuentra en el Anexo 3 “Encuesta SUS sobre la herramienta”.

Ejecución de tareas y registro

Se les pidió a los 4 capacitadores que realicen las 4 acciones definidas, y se les brindó algunos documentos preseleccionados tanto de formato *PDF* como de tipo *DOCX*. En este sentido uno de los capacitadores que realizó la prueba fue, Juan, de nombre completo, Juan Manuel Barbato, donde se puede observar su participación en la siguiente imagen (Fig. 41). Los pasos necesarios para poder utilizar las funcionalidades descritas se encuentran disponibles en el Anexo 4 “Utilización de AcceDocx”.

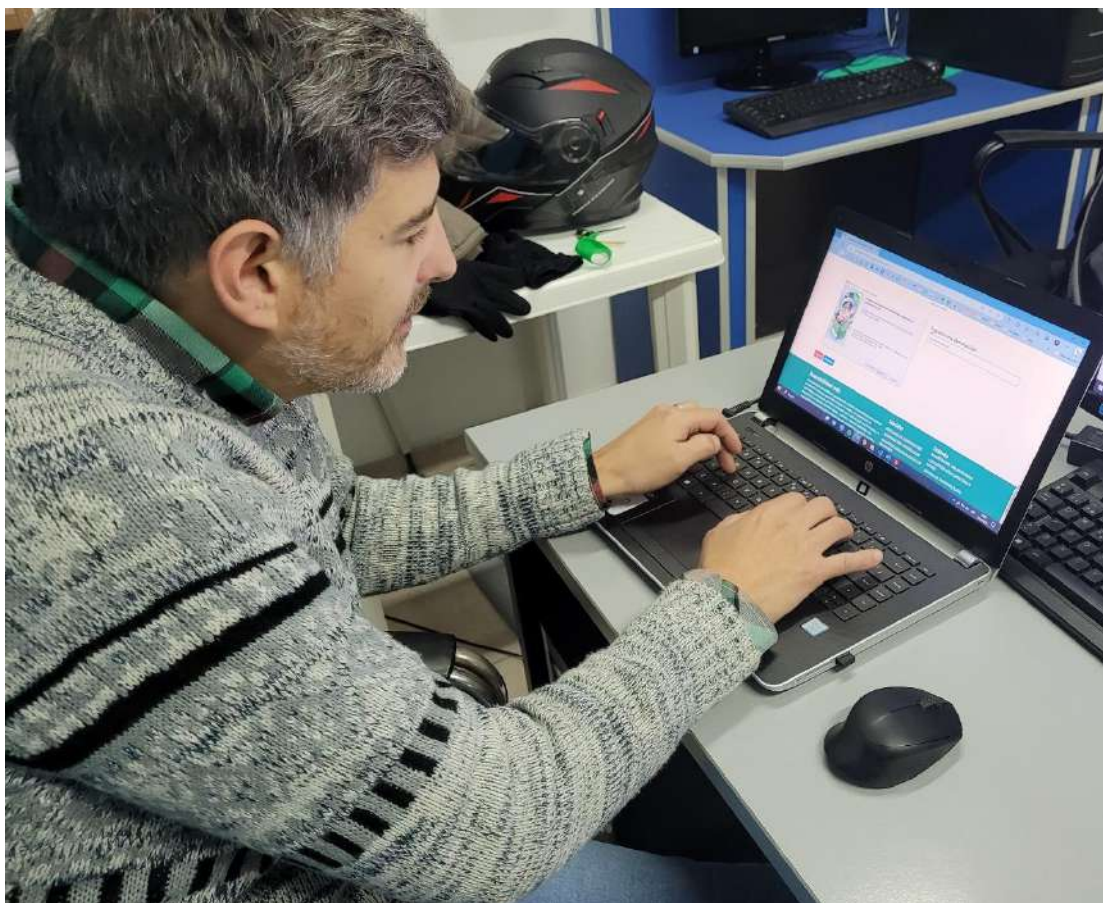


Fig. 41: Capacitador Juan utilizando AcceDocx.

Luego de ejecutarse las pruebas, se realizó la recopilación de los resultados obtenidos, tanto del uso de la herramienta como de los resultados finales sobre los documentos.

De manera adjunta, cada profesor completó el cuestionario de estilo SUS mencionado anteriormente, donde los resultados de manera resumida fueron los siguientes:

Participante	Resultado
Participante 1	90
Participante 2	87,5
Participante 3	90
Participante 4	97,5

Este cuestionario dio 91,25 puntos como resultado del promedio entre los puntajes de los 4 participantes dentro del proceso de evaluación de SUS, sobre

un valor teórico máximo de 100. Las respuestas al cuestionario se podrán encontrar de manera completa en el Anexo 5 “Resultados de las encuestas SUS” respectivamente.

Análisis de los resultados

Luego de obtener los resultados del cuestionario (Anexo 2) y de las pruebas realizadas por los profesores, se pudo detectar que la herramienta tiene un nivel de sencillez para el uso del mismo notable, y en cuanto a las funcionalidades que propone y los resultados, tienen correlación con el objetivo de accesibilizar un documento existente en el ambiente de los cursos que se dictan en el espacio de la organización.

En cuanto a las funcionalidades que provee el prototipo y la composición de los documentos digitales que se usaron en esta etapa de prueba, cumplieron con el objetivo de resolución de los componentes de cada documento utilizado.

En cuanto al diseño web, expresan el buen nivel de accesibilidad de la herramienta, tanto en la parte el formato de colores y contrastes, como de los tamaños y textos, que se encontraron durante el proceso. Los cuestionarios también fueron completados.

El profesor Juan comentó:

“Es interesante como ayuda a darle un formato accesible a los documentos con un proceso rápido, además ayuda a quienes poseen poco manejo de herramientas como procesadores de textos.”

Como resultado general, este procedimiento ha logrado permitir al capacitador acelerar en el proceso de preparación del material para el dictado de cursos.

6.3 Análisis con persona con discapacidad visual

La siguiente experiencia de usuario, sucedió en las instalaciones de CILSA ONG en la ciudad de La Plata, y tuvo como protagonista a Joana Sarmiento, persona con discapacidad visual. Ella conoce el lector de pantalla NVDA y tiene un buen manejo de las teclas para poder navegar sobre una computadora. Teniendo listo el ambiente de prueba del prototipo AcceDocx en la computadora, y un dispositivo para la grabación, se dio comienzo al proceso.

Una primera actividad inició y primera grabación se registró [62], con el objetivo era realizar una navegación sobre la página pudiendo detectar cada elemento que la compone y, a través de una persona que sirve de asistencia o guía, se le describió los detalles del mismo. Además, utilizó la funcionalidad de subir un documento en el sistema y realizar el proceso de accesibilizar al mismo.

Luego de navegar sobre AcceDocx, realizó unos comentarios finales.

Una segunda actividad se registró [63] y consistió en revisar el documento resultante y poder comparar con el original, donde el original no tiene una organización interna definida.

Una tercera se registró [64] y consistió en accesibilizar un documento que contenía imágenes, luego se verificó el resultado en el documento resultante.

Finalmente, se registró a Joana realizando algún comentario final dando cierre a su experiencia con el prototipo [65]. La participación de Joana se puede observar en la siguiente imagen (Fig. 42).

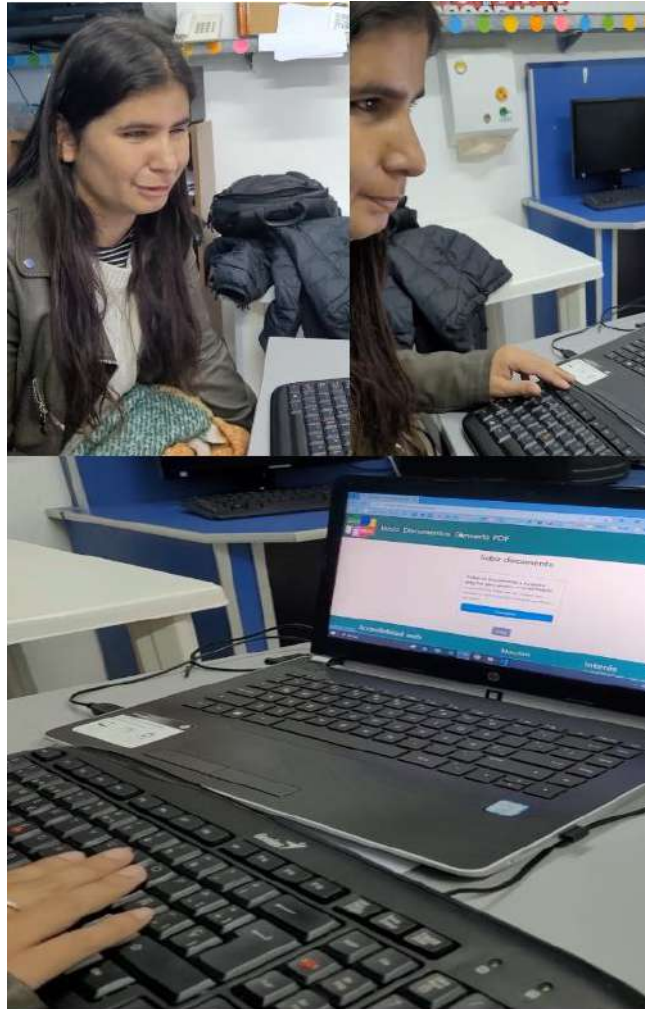


Fig. 42: Joana utilizando AcceDocx.

En pocas palabras, Joana puso a prueba la accesibilidad de la herramienta y sus documentos resultantes, por lo que logró considerar que el prototipo web es accesible y que no tuvo inconvenientes en la navegación. Además, comentó que es una herramienta con potencial utilidad dentro de un ambiente como una biblioteca, siendo utilizado por personal a cargo del espacio de lectura.

Capítulo 7: Conclusiones y Trabajos futuros

En este capítulo se presentan las conclusiones y los trabajos futuros sobre el desarrollo propuesto para accesibilizar documentos.

7.1 Conclusiones

En este informe de tesis se pudieron abordar distintos aspectos de la accesibilidad, tanto en el ámbito web como en documentos de lectura con el objetivo de examinar su aplicación en un entorno educativo.

Se investigó y se analizó la accesibilidad en documentos de lectura sobre el grupo de personas con discapacidad visual y ceguera, en un ambiente de enseñanza. En este sentido, el material educativo accesible es de suma importancia para que una persona acceda a la información y participe plenamente en el proceso de enseñanza. Al proporcionar materiales que se ajusten a las necesidades de cada estudiante, se eliminan las barreras y se fomenta la inclusión. Al considerar la accesibilidad desde el principio, se crea un entorno educativo equitativo y se potencia el aprendizaje para todos los estudiantes.

Se realizó una investigación de las herramientas disponibles en el área de la accesibilidad de documentos, logrando obtener un estudio con las características de cada una de ellas. Sobre las herramientas para accesibilizar un documento, se pudo observar que existen diversas plataformas y softwares, libres y privativos, donde cada uno ofrece ventajas y desventajas. Se identificaron diversas limitaciones en cada una de ellas, y con el objetivo de superar estas deficiencias, se procuró incorporar las mejoras correspondientes en AcceDocx.

En el presente informe, se llevó a cabo una investigación y análisis sobre la accesibilidad en documentos de lectura sobre el grupo de personas con discapacidad visual y ceguera, en el contexto específico de CILSA ONG como entorno educativo. Desde el punto de vista tecnológico, se deseaba desarrollar una herramienta que permita solucionar problemas de accesibilidad en documentos de lectura, mediante un proceso de conversión. Esto mismo, aplicando criterios de accesibilidad de manera automática,

por lo que, a través del desarrollo del prototipo AcceDocx, se logró atender esta necesidad. Mediante la creación y prueba de esta aplicación web como prototipo, se pudo comprobar que es posible generar una herramienta de apoyo, a partir de la programación con el lenguaje *Python* y tecnologías asociadas como la biblioteca *python-docx* y el framework *Django*.

Se realizó la evaluación de la herramienta y los resultados obtenidos en relación a la usabilidad y accesibilidad del prototipo fueron prometedores. Para las pruebas se lograron utilizar documentos que son parte del material dictado por el personal con roles de profesor y profesora para los cursos existentes.

Finalmente, a partir de las experiencias de uso del prototipo, se pudo obtener comentarios positivos sobre la accesibilidad. Desde el punto de vista de la accesibilidad del prototipo, para los capacitadores resultó ser una herramienta accesible y útil en el proceso de accesibilizar documentos. Además, existieron comentarios positivos en torno a la accesibilidad de los documentos generados a partir de esta herramienta. De esta manera, AcceDocx logra ser una herramienta de libre acceso, y al ser basado en web, permite hacer uso de ella de manera online mediante una computadora o cualquier dispositivo móvil.

Se pretendía diseñar y desarrollar una herramienta que no solamente fortalezca los derechos del alumno con y sin discapacidad, sino brindar la posibilidad que los mismos puedan obtener un rol de capacitador. Con este propósito, se logra concluir con el desarrollo de una herramienta que promueva la inclusión en un ambiente de enseñanza con el fin de garantizar la igualdad de oportunidades.

7.2 Trabajos futuros

Definidas las conclusiones y el trabajo propuesto en esta tesina, se incluyen como posible trabajo futuro una serie de nuevas funciones o características para añadir al prototipo AcceDocx. A continuación, se presenta un listado de las mismas:

1. Poder cargar directamente en el sistema los documentos con otros formatos. Actualmente, se permite el formato *DOCX*, y existe una funcionalidad para convertir *PDF* en *DOCX*. Un formato para agregar puede ser *PPTX*, ya que numerosos archivos en ese formato se utilizan para el dictado de temas en un ambiente de enseñanza.
2. Posibilidad de gestionar una mayor cantidad de componentes dentro de un documento.
3. A medida que se va utilizando la herramienta, poder continuar relevando requerimientos de usuarios para una nueva versión del prototipo.
4. Investigar sobre tecnologías asociadas al reconocimiento de imágenes para obtener descripciones sugerentes sobre imágenes y sobre caracteres para trabajar con OCR.

Referencias

- [1] ONU (2006). Convención internacional sobre los derechos de las personas con discapacidad. Disponible en:
<https://www.un.org/esa/socdev/enable/documents/tccconvs.PDF>
- [2] Ley N° 26.378, Argentina (2008) “Convención sobre los Derechos de las personas con discapacidad y su protocolo facultativo”. Disponible en:
<https://www.argentina.gob.ar/normativa/nacional/ley-26378-141317/texto>
- [3] Ceguera y discapacidad visual. (13 de octubre de 2022). Disponible en:
<https://www.who.int/es/news-room/fact-sheets/detail/blindness-and-visual-impairment>
- [4] OMS. CIE-11. Guía de Referencia versión 2019 (14 de noviembre de 2019). Disponible en:
[https://icd.who.int/es/docs/Guia%20de%20Referencia%20\(version%2014%20nov%202019\).PDF](https://icd.who.int/es/docs/Guia%20de%20Referencia%20(version%2014%20nov%202019).PDF)
- [5] National Eye Institute. Enfermedades y afecciones de los ojos. Disponible en:
<https://www.nei.nih.gov/espanol/aprenda-sobre-la-salud-ocular/enfermedades-y-afecciones-de-los-ojos>
- [6] University of Kent, English Heritage. Disability Since 1945 (2013). Disponible en:
<http://www.english-heritage.org.uk/discover/people-and-places/d> (kent.ac.uk)
- [7] Sistema de prestaciones básicas en habilitación y rehabilitación integral a favor de las personas con discapacidad – Ley N° 24.901 (1997). Disponible en:
<http://servicios.infoleg.gob.ar/infolegInternet/anexos/45000-49999/47677/norma.htm#:~:text=Las%20personas%20con%20discapacidad%20tendr%C3%A1n,su%20rehabilitaci%C3%B3n%20e%20inserci%C3%B3n%20social.>
- [8] Ley de educación nacional – Ley N° 26.206 (2006). Disponible en:
<http://www.bnm.me.gov.ar/giga1/documentos/EL002610.PDF>
- [9] Resolución CFE N° 311/16. Disponible en:
<https://www.argentina.gob.ar/sites/default/files/res-311-cfe-58add7585fbc4.PDF>
- [10] Honorable Congreso de la Nación Argentina. Ley de protección integral de los derechos de las niñas, niños y adolescentes - Ley N° 26.061. Disponible en:
<https://www.argentina.gob.ar/normativa/nacional/ley-26061-110778/texto>
- [11] Ley 24.521/1995. Disponible en:
<https://www.argentina.gob.ar/normativa/nacional/ley-24521-1995-25394>

- [12] Ley 25689/2002. Disponible en:
<https://www.argentina.gob.ar/normativa/nacional/ley-25689-2002-81041>
- [13] Booth, T. y Ainscow, M. (2002). Índice de Inclusión. Desarrollando el aprendizaje y la participación en las escuelas. Unesco. Oficina Regional de Educación para América Latina y atención educativa el Caribe /Centre for Studies on Inclusive Education.
Disponible en: <https://www.redalyc.org/PDF/447/44746861005.PDF>
- [14] Ley N° 22.431, Sistema de protección integral de los discapacitados, 1981.
Disponible en: <https://www.argentina.gob.ar/normativa/nacional/ley-22431-20620/actualizacion>
- [15] Web Accessibility Initiative (WAI). Disponible en <https://www.w3.org/WAI/>
- [16] W3C (2018). Niveles de adecuación de WCAG 2.1. Disponible en:
<http://accesibilidadweb.dlsi.ua.es/?menu=niveles-2.1>
- [17] W3C (2008). Web Content Accessibility Guidelines (WCAG) 2.0. Disponible en <https://www.w3.org> › WCAG20
- [18] ISO (2012). ISO/IEC 40500:2012. Information technology and W3C Web Content Accessibility Guidelines (WCAG) 2.0. Disponible en: <https://www.iso.org> › standard.
- [19] W3C (2018). Web Content Accessibility Guidelines (WCAG) 2.1. Disponible en: <https://www.w3.org/TR/WCAG21>
- [20] W3C. Novedades en WCAG 2.1. Disponible en:
<https://www.w3.org/WAI/standards-guidelines/wcag/new-in-21/es>
- [21] W3C (2023) Web Content Accessibility Guidelines (WCAG) 2.2. Disponible en: <https://www.w3.org/TR/WCAG22/>
- [22] Honorable Congreso de la Nación Argentina (2010). Ley 26653 sobre accesibilidad en los contenidos de las páginas web. Disponible en:
<http://servicios.infoleg.gob.ar/infolegInternet/anexos/175000-179999/175694/norma.htm>
- [23] Anexo II de la Disposición 6/2019. Disponible en:
<https://www.argentina.gob.ar/sites/default/files/infoleg/disp6-2.PDF>
- [24] DO-IT. University of Washington. Universal Design: Process, Principles, and Applications. Disponible en: <https://www.washington.edu/doiit/universal-design-process-principles-and-applications>
- [25] Travis, David (2011). ISO 13407 is dead. Long live ISO 9241-210. Artículo disponible en: <http://www.userfocus.co.uk/articles/iso-13407-is-dead.html>.

- [26] ISO 9241-210 Ergonomics of human-system interaction- Part 210: Human-centred design for interactive systems. Disponible en:
<https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en>
- [27] ISO 9999 (2007). Assistive products for persons with disability — Classification and terminology. Disponible en: <https://www.iso.org/standard/38894.html>.
- [28] WHATWG. (2023, 30 de junio). HTML Standard. Disponible en:
<https://html.spec.whatwg.org/multipage/>
- [29] NonVisual Desktop Access – NVDA (2006). Disponible en:
<https://www.nvaccess.org/>
- [30] JAWS Screen Reader (1995). Disponible en: <https://www.freedomscientific.com/>
- [31] WebAIM (2019). Screen Reader User Survey #8. Disponible en:
<https://webaim.org/projects/screenreadersurvey8/>
- [32] Microsoft Learn. [MS-DOCX]: Word Extensions to the Office Open XML (.docx) File Format. Disponible en: https://learn.microsoft.com/en-us/openspecs/office_standards/ms-docx/b839fe1f-e1ca-4fa6-8c26-5954d0abbccd
- [33] Adobe Acrobat. Disponible en: <https://www.adobe.com/ar/acrobat.html>
- [34] Duff Johnson. *PDF Association*. *PDF's popularity online*. Artículo disponible en: <https://PDFa.org/PDFs-popularity-online/>
- [35] W3C. Plain Text Techniques for WCAG 2.0. Disponible en:
<https://www.w3.org/TR/WCAG-TECHS/text.html>
- [36] W3C. *PDF Techniques for WCAG 2.0*. Disponible en:
<https://www.w3.org/TR/WCAG-TECHS/PDF.html>
- [37] ONTI. Recomendaciones para la redacción de documentos digitales de textos accesibles. Versión preliminar. Disponible en: https://github.com/argob/accesibilidad-web/blob/master/docs/recomendaciones_textos_accesibles.PDF
- [38] ISO 14289-1 (2012). Document management applications. Electronic document file format enhancement for accessibility — Part 1: Use of ISO 32000-1 (*PDF/UA-1*). Disponible en: <https://www.iso.org/standard/54564.html>
- [39] Daniel Dick. Office Open XML. Disponible en: <http://officeopenxml.com/>
- [40] Dejan Lukan. Infosec Resources. *PDF file format: Basic structure*. Artículo disponible en: <https://resources.infosecinstitute.com/topic/PDF-file-format-basic-structure/>
- [41] Microsoft Word market share and usage statistics. Disponible en:
<https://webtechsurvey.com/technology/microsoft-word>

- [42] OfficeDev. (2018). Script Lab. Disponible en: <https://github.com/OfficeDev/script-lab>
- [43] JavaScript MDN Web Docs (2005). Disponible en: <https://developer.mozilla.org/es/docs/Web/JavaScript>
- [44] W3C. Cascading Style Sheets. Disponible en: <https://www.w3.org/Style/CSS/Overview.en.html>
- [45] Ziv Barber. (2013). Officegen Proyecto. Disponible en: <https://github.com/Ziv-Barber/officegen>
- [46] DOCX. Easily generate and modify .docx files with JS/TS. Works for Node and on the Browser. Disponible en: <https://docx.js.org/#/>
- [47] Docxtemplater Library. Disponible en: <https://github.com/open-xml-templating/docxtemplater>
- [48] John M. Zelle (2003) “Python Programming: An Introduction to Computer Science.” Disponible en: https://jntukucen.ac.in/ebook_files/93.PDF
- [49] Python Software Foundation (2021). Python-docx Library. Disponible en: <https://pypi.org/project/python-docx/>
- [50] W3C. HTML5. Disponible en: <https://dev.w3.org/html5/spec-LC/>
- [51] Bootstrap. The most popular HTML, CSS, and JS library in the world. Disponible en: <https://getbootstrap.com/>
- [52] W3C. ARIA in HTML. Disponible en: <https://www.w3.org/TR/html-aria/>
- [53] Python para el Desarrollo web – Enseñanza en la Facultad de Informática UNLP. Disponible en: <https://www.info.unlp.edu.ar/wp-content/uploads/2022/03/Proyecto-de-Software.PDF>
- [54] Django Framework. Disponible en: <https://www.djangoproject.com/>
- [55] Django como herramienta para Desarrollo web. Disponible en: <https://djangostars.com/blog/10-popular-sites-made-on-django/>
- [56] Django Project. Django Community. Disponible en: <https://www.djangoproject.com/community/>
- [57] SQLite. SQLite Home Page. Disponible en: <https://www.sqlite.org/index.html>
- [58] Statista. Most used web frameworks among developers 2022. Disponible en: <https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/>
- [59] PyPI. Docx2txt. Disponible en: <https://pypi.org/project/docx2txt/>
- [60] PyPI. Docx2PDF. Disponible en: <https://pypi.org/project/docx2PDF/>

[61] WAVE Web Accessibility Evaluation Tools. Disponible en:

<https://wave.webaim.org/>

[62] David Huertas Google Drive. (29 de mayo de 2023). Experiencia Joana “Primer video”. Disponible en:

https://drive.google.com/file/d/1vskemAaBOTWXU3rz37mx7hv4FofuuMvw/view?usp=share_link

[63] David Huertas Google Drive. (29 de mayo de 2023). Experiencia Joana “Segundo video”. Disponible en:

https://drive.google.com/file/d/1LOjovc8dm09Y5t8UVFUIMp84lyserFTI/view?usp=share_link

[64] David Huertas Google Drive. (29 de mayo de 2023). Experiencia Joana “Tercer video”. Disponible en:

https://drive.google.com/file/d/1mOv9bOIw4uBdSHMIvkXRAe74MMaaOi6K/view?usp=share_link

[65] David Huertas Google Drive. (29 de mayo de 2023). Experiencia Joana “Cuarto video”. Disponible en:

https://drive.google.com/file/d/1x7duhVuqkNUB_xDGhnPtaY0jxsWyMxx2/view?usp=share_link

Anexo 1: Inicialización de AcceDocx

La herramienta desarrollada se encuentra en un ambiente local, es decir, no se encuentra en producción (en un ambiente online). De manera que nuestro producto web se encuentra en un ambiente de desarrollo, por lo que para iniciarse necesitamos ejecutar una instrucción sobre el ambiente de Django.

A continuación, se hace mención de los pasos a seguir para iniciar la aplicación web, para que el lector, con cierto grado de conocimiento técnico, pueda hacer uso de la misma:

- 1) **Abrir una terminal:** abrir un editor de código con una consola o terminal disponible, o directamente abrir una terminal de comandos (CMD) respectivo del sistema operativo.
- 2) **Iniciar el servidor web de Django:** para iniciar el servidor web de Django, se debe estar en la ubicación correspondiente a la carpeta del proyecto, y ejecutar sobre una terminal la siguiente instrucción: `python .\manage.py runserver`
- 3) **Iniciar la aplicación web:** para esto, debemos ingresar en la barra de direcciones de un navegador e ingresar dentro de la dirección at `http://127.0.0.1:8000/` y seguido colocando la dirección con la cual la aplicación fue programada para su página inicial.

En el caso de estar en un ambiente de producción, este procedimiento se deja de lado, ingresando simplemente a la dirección en internet de la aplicación web.

Anexo 2: Componentes detectables con potencial implementación.

A partir de un proceso de investigación, se describen a continuación, componentes detectables por la biblioteca python-docx que no fueron incorporados en esta etapa. De manera adicional, se incorporan porciones de código en Python a cada uno de ellos.

Tablas

Se representa dentro del documento *DOCX* como un elemento `<w:tbl>`. La biblioteca permite el armado de este componente, pudiendo agregar columnas y filas. Un código asociado a la creación de una tabla es el siguiente:

```
from docx import Document
# crear documento
doc = Document()
# agregar una tabla
tabla = doc.add_table(rows=2, cols=2, style="Table Grid")
```

Encabezado y pie de página

Componentes asociados a cada página del documento, en este caso existirán secciones con la posibilidad de acceder a alguno de ellos y poder determinar su encabezado y pie de página. Un código asociado al proceso de agregar texto al encabezado es el siguiente:

```
documento = Document()
seccion = documento.sections[0]
encabezado = seccion.header
parrafo = encabezado.paragraphs[0]
parrafo.text = "Este es mi encabezado"
```

Tabla de contenido

Este componente es posible generarlo de manera más elaborada que los anteriores, no existen métodos propios para este fin, sino creando algún espacio llamado campo complejo donde se irán agregando ciertos elementos.

Un código asociado a este objetivo es el siguiente:

```
fldChar = OxmlElement('w:fldChar') # crea un nuevo componente
fldChar.set(qn('w:fldCharType'), 'begin') # define atributos del elemento
```

```
instrText = OxmlElement('w:instrText')
instrText.set(qn('xml:space'), 'preserve') # define atributos del elemento
instrText.text = 'TOC \o "1-3" \h \z \u' # cambia 1-3 dependiendo de los niveles de
títulos que se requiera.
```

Una alternativa para la generación de este componente es utilizando la biblioteca Win32 para Python, pywin32. La misma se puede obtener desde el sitio pypi.org, y nos permite interactuar y manipular funciones del sistema operativo Windows. En esta herramienta podemos encontrarnos con funcionalidades asociadas a un documento estilo Word y una de ellas es sobre la tabla de contenidos.

Anexo 3: Encuesta SUS sobre la herramienta.

A continuación, se describen las preguntas realizadas en el cuestionario incorporado en la etapa de pruebas de usuarios:

1. Creo que me gustaría utilizar la herramienta AcceDocx con frecuencia:
 - Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo

2. Encontré a AcceDocx innecesariamente complejo:
 - Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo

3. Pensé que AcceDocx era fácil de usar:
 - Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo

4. Creo que necesitaría el apoyo de un técnico para poder utilizar AcceDocx:
 - Totalmente en desacuerdo
 - En desacuerdo
 - Neutro
 - De acuerdo
 - Totalmente de acuerdo

5. Encontré que las diversas funciones de AcceDocx estaban bien integradas:
 - a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutro
 - d. De acuerdo
 - e. Totalmente de acuerdo

6. Pensé que había demasiada inconsistencia en AcceDocx:
 - a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutro
 - d. De acuerdo
 - e. Totalmente de acuerdo

7. Me imagino que la mayoría de la gente aprendería a utilizar AcceDocx muy rápidamente:
 - a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutro
 - d. De acuerdo
 - e. Totalmente de acuerdo

8. Encontré a AcceDocx muy complicado de usar:
 - a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutro
 - d. De acuerdo
 - e. Totalmente de acuerdo

9. Me sentí muy seguro usando AcceDocx:
 - a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutro
 - d. De acuerdo
 - e. Totalmente de acuerdo

10. Necesitaba aprender muchas cosas antes de empezar con AcceDocx:
- a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutro
 - d. De acuerdo
 - e. Totalmente de acuerdo

Anexo 4: Utilización de AcceDocx.

Paso 1: Ingreso al sitio web.

Al iniciar la aplicación web, se nos presenta la página principal (Inicio) que se muestra en la figura siguiente (Fig. 43).



Fig. 43: Página inicial de la aplicación web

Paso 2: Listado de documentos

Con la aplicación web iniciada, el profesor accede a la sección “Documentos” donde aparecerá el listado de documentos existentes en el sistema. Para esto ubicamos el botón Documentos en la barra de navegación el cual se observa en la siguiente figura (Fig. 44), y presionamos sobre él.



Fig. 44: Barra de navegación superior donde se destaca el botón Documentos.

Luego de haber realizado lo anterior, el profesor se encontrará en la sección del listado de documentos como se observa en la siguiente figura (Fig. 45).



Fig. 45: Pantalla con el listado de documentos donde se destaca la fila existente.

Paso 3: Carga de documento

Lo siguiente que el profesor debe realizar es la carga de un documento de lectura con extensión .docx, para hacer esto debe presionar el botón correspondiente con el nombre “Subir documento” que se visualiza a continuación en la siguiente figura (Fig. 46).

Listado de Documentos



Fig. 46: Pantalla con el listado de documentos donde se destaca el botón “Subir documento”.

El profesor continúa sobre una nueva pantalla donde se observa el elemento principal que es un cuadro descriptivo sobre el procedimiento que llevará a cabo, sobre esta pantalla deberá presionar sobre el botón “Comenzar” como se observa en la siguiente figura (Fig. 47).

Subir documento

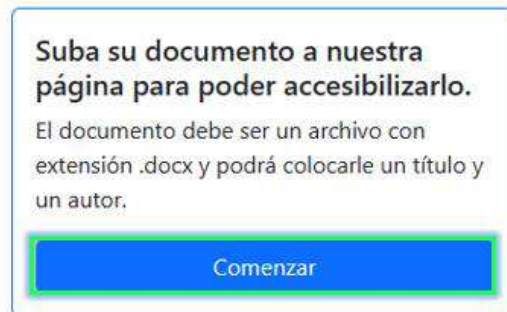


Fig. 47: Cuadro descriptivo para subir un documento donde se destaca el botón “Comenzar”.

A continuación, el profesor deberá completar los campos solicitados por el formulario que se le presenta frente a él, tal como se muestra en la siguiente figura (Fig. 48). Completar los campos, elegir el archivo que desea cargar en el sistema, y luego presionar el botón “Subir” son los pasos que debe realizar el profesor.

Subir documento



Título*
Clase 2 - Uso y descarga de Software Libre

Autor*
Capacitador David

Documento*
Elegir archivo Uso y descar...re Libre.docx

Cancelar Subir

Fig. 48: Formulario para cargar de documento *DOCX* donde se destaca el botón “Subir”.

Lo siguiente que sucederá es que el profesor podrá visualizar el archivo que fue cargado en el sistema dentro del listado de documentos que fueron subidos sobre la plataforma. La siguiente figura (Fig. 49) muestra el listado de documentos con el archivo nuevo recién cargado, y sus acciones permitidas sobre éste.

TÍTULO	AUTOR	ACCIONES		
Clase 1 - Archivos	Capacitador David	ELIMINAR	ACCESIBILIZAR	DESCARGAR
Clase 2 - Uso y descarga de Software Libre	Capacitador David	ELIMINAR	ACCESIBILIZAR	DESCARGAR

Fig. 49: Listado de documentos donde se destaca la fila con el nuevo archivo cargado al sistema.

De esta manera el archivo es subido al sistema y se finaliza este segundo paso.

Paso 3: Proceso de Accesibilizar

Posteriormente a la carga del archivo al sistema, el profesor debe visualizar la fila correspondiente al archivo sobre el cual va a trabajar, y luego sobre la misma debe

presionar en el botón “Accesibilizar” que se encuentra dentro de las Acciones posibles, tal como se visualiza en la siguiente figura (Fig. 50).



Fig. 50: Fila de un archivo con las acciones disponibles, destacando el botón “Accesibilizar”.

Continuando en la siguiente página, el profesor puede observar los datos del documento, como son el título y el autor, que se muestran en la siguiente figura (Fig. 51).



Fig. 51: Página dentro del proceso de accesibilizar, destacando los datos del documento.

Iniciado este proceso, el profesor visualizará distintas secciones con los elementos detectados. Una primera sección de un posible título, será completado por el profesor, seleccionando alguno de los botones de opciones, y seleccionando la opción del selector del nivel del título. Estos elementos y la selección de las opciones se visualizan en la siguiente figura (Fig. 52):



Fig. 52: Sección para definir un posible título destacando los elementos disponibles para definir y las opciones seleccionadas.

La sección sobre cada imagen deberá aparecer si este elemento existe en el documento, por lo que el profesor deberá ingresar la descripción en el lugar correspondiente, tal como se visualiza en la siguiente figura (Fig. 53):



Fig. 53: Sección para colocar una descripción a una imagen donde se destaca el campo a completar.

Para finalizar el proceso, el profesor deberá presionar sobre el botón Terminar que se encuentra al final de la página como se muestra a continuación (Fig. 54).

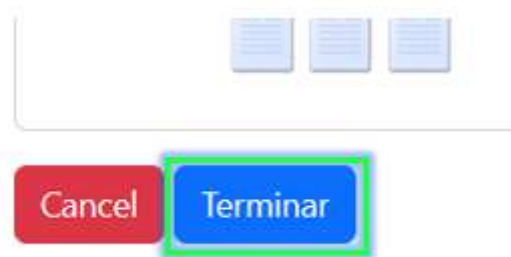


Fig. 54: Opciones para finalizar el proceso, donde se destaca el botón “Terminar”.

Paso 4: Descarga en DOCX

Para descargar el documento ya accesible, el profesor debe verificar que el botón “Descargar” se encuentra habilitado como se muestra en la siguiente figura (Fig. 55), por lo que luego deberá presionar sobre él y el documento iniciará su descarga en el ordenador.

TÍTULO	AUTOR	ACCIONES		
Clase 1 - Archivos	Capacitador David	ELIMINAR	ACCESIBILIZAR	DESCARGAR

Fig. 55: Ventana con el listado de documentos donde se destaca el botón “Descargar”.

Paso 5: Conversión de PDF a DOCX

Para realizar la conversión, lo primero que debe hacer el profesor es ingresar a la sección “Convertir *PDF*” que aparece en la barra de navegación de la página como se muestra a continuación (Fig. 56).



Fig. 56: Barra de navegación superior donde se destaca la opción “Convertir *PDF*”.

A continuación, el profesor deberá completar el formulario como se observa en la siguiente figura (Fig. 57) y luego de esto debe presionar sobre el botón “Convertir”.

Convertir PDF

Título:

Autor:

Elegir archivo

Fig. 57: Formulario para conversión de *PDF* a *DOCX* donde se destaca el botón “Convertir”.

Una vez presionado el botón mencionado, se iniciará la descarga en el dispositivo actual, listo para ser cargado en el sistema para poder accesibilizar.

Paso 6: Eliminar un documento

Para eliminar un documento del sistema, el profesor debe visualizar la fila correspondiente al archivo sobre el cual va a trabajar, y luego sobre la misma debe presionar en el botón “Eliminar” que se encuentra dentro de las Acciones posibles, tal como se visualiza en la siguiente figura (Fig. 49). Una vez presionado el botón mencionado, se iniciará el proceso de eliminación, por lo que el profesor tendrá que confirmar la operación. Una vez confirmada, el documento ya no pertenece al sistema.

Anexo 5: Resultados de las encuestas SUS.

A continuación, se muestran los resultados obtenidos de las preguntas realizadas en el cuestionario SUS. Para visualizar los resultados, se arma una tabla correspondiente con la información necesaria para generarlos, teniendo los siguientes criterios de evaluación:

- Cada enunciado tiene una respuesta en la escala de Likert, que equivaldrá a 1, 2, 3, 4 o 5, en función de la respuesta en el orden que aparece.
- Se realiza las siguientes operaciones:
 - Suma de las respuestas de los enunciados impares y después se le resta 5
 - Suma de las respuestas de los enunciados pares y se le resta ese total a 25
 - Suma de ambos resultados y se multiplica por 2,5.

Dado un valor máximo teórico de 100, un puntaje de 68 se considera “bueno” y uno cercano a los 85 es “excelente”, estando por sobre el promedio de resultados en esta encuesta. Un puntaje inferior a 68 se considera por debajo del promedio, y su consideración varía entre “Ok” y “Pobre”.

Tabla resultante:

Participante / Pregunta	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Resultado
Participante 1	4	2	4	1	5	1	5	1	4	1	90
Participante 2	4	1	4	2	4	1	4	1	5	1	87,5
Participante 3	4	2	4	1	5	1	4	1	5	1	90
Participante 4	5	1	5	1	5	1	4	1	5	1	97,5

Este cuestionario ha dado 91,25 puntos como resultado del promedio entre los puntajes de los 4 participantes.