UNIVERSIDAD NACIONAL DE LA PLATA

FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

TÍTULO: Dispositivos de Interacción auditiva como interfaces de contenidos Web

AUTORES: Torre Manuel, Ripa Gonzalo

DIRECTOR: Gustavo Rossi

CODIRECTOR: Sergio Firmenich

ASESOR PROFESIONAL:

CARRERA: Licenciatura en Sistemas

Resumen

Los asistentes de voz, particularmente los nuevos dispositivos conocidos como altavoces inteligentes, permiten a los usuarios finales interactuar con aplicaciones por medio de comandos de voz. Usualmente, los usuarios finales son capaces de instalar aplicaciones (también llamadas skills) que se encuentran disponibles en repositorios y cumplen con múltiples propósitos. En este trabajo presentaremos un entorno para usuarios finales que permitirá definir habilidades ("skills") para asistentes de voz, en base a la extracción de contenidos presentes en la Web y su organización en diferentes patrones de navegación por voz.

Palabras Clave

Asistentes de voz, Programación de usuario final, Contenido Web, VUI, Web Scrapping

Conclusiones

Presentamos un enfoque de desarrollo para los usuarios finales, que permite la creación de skills propias basadas en VUI, para que sean usadas con fuentes de información y servicios Web preferidos. Creemos que la creación de VUI basadas en contenidos Web podría ser una manera interesante de otorgar más control a los usuarios mientras interactúan con sus dispositivos.

Trabajos Realizados

Discutimos el fundamento y las mecánicas para adaptar contenido Web dentro de una VUI.

Presentamos nuestro entorno EUD, incluyendo el template de extracción para bloques de contenido y SkillMaker, nuestra herramienta EUD utilizada para crear VUI basadas en bloques de contenido.

Creamos SkillHub, aplicación con la que el usuario interactuará para obtener contenidos web en formato de voz.

Finalmente, realizamos pruebas de usuario para verificar la factibilidad y usabilidad de nuestra solución.

Trabajos Futuros

Extender los templates de extracción definidos, permitiendo al usuario definir más elementos que sean parte de la estructura de los contenidos abstraídos.

Mejorar la obtención de la ruta perteneciente a cada contenido dentro de un sitio web. Nuestra solución se basa en obtener expresiones xpath directamente desde el DOM de una página web.

Encontrar un método (alternativo al uso de la librería Puppeteer) más eficiente para la obtención del texto de los contenidos.

Contemplar la posibilidad de incorporar el uso de motores de búsqueda dentro de los sitios web, como nuevo medio de navegación, adaptados a una solución que permita interactuar con ellos por medio de VUIs.

Fecha de la presentación: 15 de Mayo 2020

Dispositivos de Interacción auditiva como interfaces de contenidos Web

Ripa Gonzalo, Torre Manuel

15 de mayo de 2020

Resumen

Los asistentes de voz, particularmente los nuevos dispositivos conocidos como altavoces inteligentes, permiten a los usuarios finales interactuar con aplicaciones por medio de comandos de voz. Usualmente, los usuarios finales son capaces de instalar aplicaciones (también llamadas skills) que se encuentran disponibles en repositorios y cumplen con múltiples propósitos. En este trabajo presentaremos un entorno para usuarios finales que permitirá definir habilidades ("skills") para asistentes de voz, en base a la extracción de contenidos presentes en la Web y su organización en diferentes patrones de navegación por voz. Describiremos el enfoque, el entorno de desarrollo para usuarios finales, y finalmente presentaremos algunos casos de estudio basados en el servicio Alexa y los dispositivos Amazon Echo.

Índice general

Capítulo 1

Introducción

Desde la aparición de internet a escala global, tenemos la capacidad de consumir, con creciente facilidad, información generada y distribuida desde cualquier rincón del mundo. Internet, a través de todas sus interfaces conocidas, ha sido el principal impulsor en el ámbito de la investigación y comunicación, acelerando los procesos de obtención de información y conexión de ideas a una velocidad sin precedentes. La computadora personal fue el principal interlocutor que hemos podido apreciar durante todos estos años. Luego surgieron las computadoras portátiles, las cuales permitieron trasladar la computadora personal como ya la conocíamos hacia cualquier espacio físico y así utilizarla para tareas que antes no eran posibles. Por ejemplo, poder llevarla a una reunión o a un viaje con la familia. Por si fuera poco, la aparición de dispositivos móviles(como los smartphones, tablets o las PDA yendo más atrás en el tiempo) lograron que el acceso a la información sea aún más cómodo. El hecho de generar, distribuir y reproducir información en cualquier momento y en cualquier lugar restableció las reglas de juego. El ritmo es cada vez más veloz, tanto que cambió el concepto de lo inmediato y de aquello que ya es "viejo". Pero el punto más importante, es que la cantidad de información generada volvió a impulsarse ascendentemente, alimentada por el consumo ininterrumpido. Esta información es accedida casi únicamente a través de un dispositivo con una pantalla. Nuestra atención debe enfocarse en esta pantalla y en leer los contenidos.

En los últimos años, ha surgido una nueva forma de interacción con los dispositivos mediante el uso de la voz; es así que los dispositivos existentes fueron incorporando nuevas funcionalidades y adaptando la manera en que los usuarios interactúan con estos. Esto llevó también a la incursión en el mercado de nuevos dispositivos inteligentes, que funcionan únicamen-

te mediante el uso de comandos de voz, produciendo respuestas mediante el sonido, como son los smart speakers. Estos dispositivos no disponen de ninguna pantalla que brinde algún tipo de interacción visual.

1.1. Motivación

La "red mundial de internet" es la principal fuente de información y plataforma de servicios. Un usuario promedio utiliza la web con múltiples fines. Quizás está haciendo una tarea en el hogar, mientras consume servicios de audio para escuchar música y envía mensajes instantáneos a sus amigos, o está en su rato de ocio y ve una película mediante un servicio de streaming. Otros acceden a la información de sus cuentas bancarias y tarjetas de crédito e incluso realiza pagos, cobros, transferencias y distintos tipos de solicitudes. Asimismo, es tan amplia la diversidad de información que se encuentra disponible (y las formas de acceder a ella) que casi todos los empleos de hoy en día perciben la web como una herramienta infaltable para el desarrollo de sus tareas.

Paralelamente, los algoritmos de reconocimiento de voz y las tecnologías relacionadas, han ido experimentando intensos avances en los últimos años, alcanzando un amplio consumo por parte de los usuarios finales.

En el último tiempo, se comenzó a gestar una nueva interfaz de interacción con las aplicaciones que brindan información. Comenzaron a surgir nuevos dispositivos enfocados en interactuar y representar información a través del sonido. Uno de los principales exponentes al día de hoy es un dispositivo producido por la empresa norteamericana Amazon, llamado Echo. Este dispositivo permite, además de muchas funcionalidades propias de un asistente personal, obtener información desde servicios específicos de internet y redactarla al usuario de manera natural, mediante una voz clara y fácil de entender. Echo tuvo gran aceptación por parte de los usuarios. En base a sus experiencias, la mayoría lo ha calificado de forma muy positiva (tomando como muestra aproximadamente 600 evaluaciones). Se ha verificado además que el usuario que tiende a "personificar" al dispositivo posee una mejor recepción del mismo, pudiendo derivarse ésto de las facilidades que ofrece con respecto a la interacción [1]. Este buen recibimiento por parte de los usuarios, sumado al trabajo y desarrollo que se comenzó a evidenciar en el último tiempo, potenciará con los años aún más su utilización. De hecho, se estima que, durante la próxima década, este tipo de dispositivos podrá llevar a cabo tareas por parte de cada usuario tanto en el mundo físico como digital de manera autónoma [2].

Como cualquier otro tipo de dispositivo inteligente, estos dispositivos permiten a los usuarios instalar aplicaciones (también llamadas skills en el caso de Amazon Echo) que ofrecen servicios de información específicos [3]. Algunas aplicaciones de los altavoces inteligentes están relacionadas con el manejo de otros dispositivos inteligentes, o incluso con mashups de dispositivos inteligentes gracias al uso de plataformas IoT como por ejemplo IFTTT [4] o Node-Red [5]. Sin embargo, otros tipos de aplicaciones para altavoces inteligentes están más enfocados en leer informacion e interactuar con servicios ya publicados por las aplicaciones Web existentes, como puede ser leer las noticias de algún portal de noticias, o preguntar por el precio de un producto de la tienda Amazon.com.

Se comenzó a llevar a cabo un proceso de adaptación de contenidos desde la Web hacia los dispositivos de interacción por voz, similar a la adaptación de contenidos Web en los teléfonos inteligentes. Por ejemplo, el sitio de reservas Expedia.com ofrece una skill para Amazon Echo, que permite a los usuarios interactuar con la voz buscando precios de alojamientos y vuelos. Aunque hay que resaltar que existe una diferencia entre ambos procesos, dado que en el caso de los dispositivos móviles, existe aún la posibilidad de visitar cualquier sitio Web por medio de algún navegador Web desde el propio móvil. En cambio, en el caso de los altavoces inteligentes, no se brinda esta posibilidad, por lo que no será posible acceder a servicios y contenidos que no sean brindados por alguna aplicación nativa (Skill).

A pesar del avance en la Ingenieria Dirigida por Modelos [6] y en las interfaces de usuario multi-modales [7], la gran mayoría de los sitios Web no se encuentran desarrollados con estas especificaciones, por lo que crear aplicaciones específicas para este tipo de dispositivos (por ejemplo proporcionando el acceso a voz) será generalmente muy costoso. De esta manera, podemos deducir que la capacidad de interacción de este nuevo jugador con los proveedores de información se ve limitada por la necesidad de adaptar sus contenidos a esta nueva tecnología. Si bien la implementación de una skill para Amazon Echo puede ser una opción para aquellas empresas o proveedores de información que deseen trasladar sus contenidos a este tipo de interfaz, para el usuario siempre existirá la condición necesaria de que el proveedor del contenido desarrolle una skill que permita consumirlo. Cada portal debería implementar su servicio propio y ofrecerlo en la tienda de skills disponibles para Echo. Esto reduce significativamente el acceso a la información de esta nueva interfaz. ¿Qué pasa si el usuario suele leer las noticias en un portal que no tiene planificado implementar su propia skill? Deberá hacer uso de algún otro dispositivo que prefiera y leer de la pantalla.

Como dijimos, las aplicaciones Web juegan un rol muy importante en la

vida diaria de los usuarios; las usamos para leer noticias, trabajar, e incluso para interactuar con dispositivos inteligentes en el mundo de la "internet de las cosas" (IoT). Aquí es donde surge la necesidad de poder adaptar cualquier portal disponible en la web hacia esta clase de dispositivos. La gran fuente de información de internet se encuentra en formato web, lo cual implica que es representada visualmente.

Este trabajo tiene por objetivo llenar la brecha existente entre las aplicaciones disponibles para altavoces inteligentes, y los servicios y contenidos presentes en la Web, que los usuarios consumen diariamente por medio de un navegador Web. Proponemos un entorno de desarrollo orientado a usuarios finale [8], en el que los usuarios puedan crear sus propias extensiones para asistentes de voz basándose en la abstracción y extracción de contenidos y servicios Web que están acostumbrados a usar.

En nuestro trabajo, utilizaremos una herramienta basada en el concepto de ingeniería de software para el usuario final (End-User Software Engineering), mediante la cual los usuarios finales podrán definir modelos de información de manera guiada. Esta herramienta, además, favorecerá la reutilización de los modelos definidos, ya que podrán ser utilizados por distintas aplicaciones que requieran de la abstracción de los contenidos web [9]. Nuestra investigación está enfocada en utilizar esta herramienta para obtener la información requerida, y lograr que los usuarios puedan acceder y escuchar información dinámica que se encuentre disponible en sitios web seleccionados por ellos mismos, a través de una interfaz distinta, más cómoda y accesible como la que brinda Amazon Echo.

El trabajo se organiza como sigue. El capitulo 2 presenta un trasfondo en diferentes aspectos relacionados a este enfoque. En el capitulo 3 se llevó a cabo un análisis acerca de la posibilidad de adaptar los contenidos de la Web hacia una VUI. El capitulo 4 introduce nuestro enfoque y presenta la razón subyacente de nuestro entorno de desarrollo orientado a usuarios finales, el cual es descrito detalladamente en el capitulo 5. En el mismo, además de los detalles de implementación, se muestra un ejemplo de uso de interacción final con nuestra skill diseñada para Amazon Echo. En el capitulo 6, se llevó a cabo una evaluación de nuestro enfoque mediante las pruebas de usuario. Finalmente, brindamos una conclusión y trabajos futuros en el capitulo 7.

1.2. Objetivo general

El objetivo de nuestro trabajo será diseñar un enfoque que permita, a los usuarios finales sin conocimientos de programación, especificar extensiones

para dispositivos con interfaz de voz basadas en contenidos Web existentes. De esta manera, se brindará una adaptación real y accesible que funcione como alternativa a las interfaces ya existentes hasta el momento.

Con esto se lograría poder adaptar cualquier contenido disponible en la web para reproducirlo de forma auditiva, sin que sea necesario que cada sitio web tenga que implementar su propio servicio.

A continuación se describen los objetivos epecíficos:

- Realizar un análisis sobre los distintos patrones de interacción entre el usuario y Amazon Echo que podrían adaptarse a la aplicación propuesta. Se espera que el usuario logre consumir contenidos de manera cómoda y eficaz.
- Desarrollar una aplicación "skill" con la cual el usuario podrá interactuar para la obtención de contenidos.
- Desarrollar una herramienta que permita al usuario extraer contenidos web de sitios web definidos por él mismo.
- Implementar una interfaz de usuario para la configuración de la aplicación skill por parte del usuario.
- Generar un web service que logre obtener el texto correspondiente a partir de una URL e identificadores propios del DOM (Document Object Model) y retornarlo para su eventual reproducción por parte del dispositivo.
- Implementar un modelo que se adecúe a los objetivos de la aplicación. Esto incluiría la relación de los usuarios con sus contenidos personalizados para su reproducción.
- Verificar la aceptación de la tecnología en cuestión por parte de los usuarios para un uso cotidiano orientado al consumo de contenidos web.

Capítulo 2

Marco Teórico

Como hemos mencionado en la introducción de esta tesina, nuestra intención es la de investigar y desarrollar, a partir de tecnologías ya existentes, un procedimiento que facilite la adaptación de los contenidos que están expresados en forma de texto en la web hacia un medio auditivo.

En este contexto, es necesario llevar a cabo un relevamiento de los distintos estudios e investigaciones ya elaborados que guardan relación con nuestra investigación y que nos resultan útiles para el desarrollo de la misma. A continuación, se presentarán algunos antecedentes relevantes.

2.1. Background

2.1.1. Interfaces de usuarios basadas en voz (VUI)

Los agentes conversacionales y asistentes virtuales no son un concepto nuevo. Ya en el año 1960, Licklider introduce el interés en "hablar con las computadoras" como una dimensión a contemplar en la interacción humano-computadora [10]. Hasta ese momento, la producción de voz fue fácilmente ejecutada por los sistemas electrónicos, aunque aún existían problemas severos en el proceso de reconocimiento de la voz. Los algoritmos de reconocimiento de voz han ido evolucionando rápidamente en las últimas décadas; hemos podido apreciar distintos trabajos de investigación relacionados con los agentes conversacionales (también conocidos como interfaces conversacionales) desde hace casi 20 años [11] [12].

En este contexto, disponemos de muchas variantes para definir el tipo de interacción entre el usuario y los sistemas que se basan en la interacción por voz. Por ejemplo, McTear [13] define dos términos para definir este concepto: "Spoken Dialog System" (SDS) y "Voice User Interface" (VUI). Ambos se

utilizaron durante los años 90 para definir las interacciones con el usuario a partir de la voz y el lenguaje natural. Durante esa década, la empresa ATT fue una de las incursoras en este tipo de tecnologías. Implementó el sistema "How may I help you" ("Cómo puedo ayudarlo" en español) que era utilizado para poder comprender las demandas de los usuarios, y así redirigirlo hacia el sector adecuado para la atención con un empleado que pueda resolver la demanda específica del cliente. Más allá de que los términos anteriormente mencionados (SDS y VUI) utilizan el mismo tipo de tecnología y muchas veces son utilizados como sinónimos, McTear [13] resalta una diferenciación entre ambos: el término SDS es mayoritariamente utilizado dentro del ámbito de la investigación, mientras que VUI es utilizado por los desarrolladores principalmente en el ámbito estricto de implementaciones orientadas a negocio.

Adicionalmente, podemos encontrar también la nomenclatura "Conversational web interface", definida así por Baez [14], quien hace foco en cómo los usuarios podrían eventualmente tener una comunicación verbal con un agente conversacional, permitiendo la navegabiliidad de un sitio web mediante un flujo de informacion bidireccional adecuado para los distintos sitios web. En este sentido, se hace referencia a especificaciones como WAI-ARIA y estándares como VoiceXML, aunque aclara que no están diseñados y no son óptimos para escenarios de comunicación mediante agentes conversacionales.

Como describen Hauswald [15] y Pradhan [16] los asistentes de voz son también llamados "Dispositivos personales inteligentes" (Intelligent Personal Assistant o IPA) porque reúnen además ciertas características que los diferencian de los medios usuales. En los artículos, se denota que este tipo de dispositivos poseen características que los diferencian del resto no solamente por el tipo de medio, sino también por su interfaz, que está compuesta además por un componente "inteligente" que procesa las entradas de los usuarios en base al contexto y expone una respuesta adecuada teniendo en cuenta este aspecto. Hauswald [15] utilizó técnicas de procesamiento de lenguage natural ("Natural language processing" ,NLP) que realiza una comparación efectiva de las entradas del usuario, las reconoce y las clasifica en tres tipos diferentes: VC(Voice command), VQ (Voice query), y VIQ (Voice-image query). Cada tipo de entrada es derivado a un servicio distinto que realiza las tareas necesarias para producir la salida adecuada. Esta primera clasificación de la entrada significa la primer arista de lo que luego será la VUI proporcionada al usuario para la explotación de los servicios que se ofrecen.

Dentro de otras referencias al término VUI, creemos adecuado mencionar la de Cohen[17], que identifica a la VUI como aquello con lo que una persona

interactua cuando utiliza una tecnología orientada a la interacción por voz. Se diferencian tres conceptos fundamentales: Prompts (en español "aviso" o "respuesta"), gramática y lógica de diálogo. Estos conceptos se utilizan de manera coordinada para la creación de las interfaces diseñadas. Además, se indica que las VUIs se diferencian en muchos aspectos con respecto a las interfaces corrientes, pero que sin embargo, también tienen muchos aspectos en común, que pueden ser de utilidad a la hora de desarrollar interacciones fluidas y eficientes.

Más allá de las distintas alternativas que encontramos para referirnos a la interacción por medio de la voz para el acceso a contenidos web, hemos llegado a la conclusión de que el término **VUI** es el que mejor se aplica a las distintas circunstancias dentro del ámbito de investigación e implementación de interfaces de voz para el acceso a información en la red.

Hoy en día, las VUIs han sido desplegadas para distintos tipos de interacción y dispositivos, como pueden ser los teléfonos inteligentes, altavoces inteligentes, etc. Este tipo de interfaces nos permite interactuar con un producto de una manera distinta, sin el uso de las manos ni de la vista, pudiendo enfocar nuestra atención en alguna otra cosa.

Aunque las VUIs comenzaron a ser ampliamente utilizadas con su inclusión en los teléfonos inteligentes (como es el caso de Siri en iPhone [1]), con la aparición de nuevos dispositivos como son Google Home o Amazon Echo, ha ido cambiando su uso diario y ha crecido su presencia en los ambientes. Estos altavoces inteligentes permiten a los usuarios iniciar una interacción conversacional mediante un comando de voz expresado en lenguaje natural, tal como "Alexa, tell me the news", en el caso del servicio Alexa proporcionado por Amazon. Los altavoces inteligentes cuentan con un conjunto base de comandos y capacidades; por ejemplo, relacionados al tiempo y clima, o alguna interacción VUI del tipo pregunta-respuesta que consuma servicios de venta. Además, a partir de la instalación de skills de terceros en el caso de Amazon Echo, podremos reproducir música, leer noticias, etc...

Actualmente, solo analizando un caso de dispositivos, el repositorio de Skills de Alexa organizado en categorías, ofrece más de 50.000 skills [2], casi doblando el número de skills disponibles al final del año 2017 [18].

En la tabla 1 se listan las categorías más relevantes, el número de skills por categoría, un ejemplo de skill por categoría, y algunos comandos disponibles para ese ejemplo de skill.

[1] Siri, https://www.apple.com/es/siri/, last accessed 3/14/2019

[2] Alexa skill repository: https://www.amazon.com/alexa-skills/b?ie=UTF8&node=13727921011, accessed February 20th 2019.

Table 1.

Categoría de Skill	Cantidad de Skills	Ejemplo de Skill	Ejemplos de comandos de Skill
Business & Finance	over 1.000	Marketplace	*"Alexa, what's my Flash Briefing?"*, *"Alexa, what's in the news?"*
Communication	over 1.000	Mastermind	*"Alexa, Ask Mastermind to text <someone>"*, *"Alexa, ask Mastermind to ring my phone"*
Education & Reference	over 8.000	Couriosity	No direct commands, this skill offer aleatory content that end user may skip.
Games & Trivia	over 10.000	Twitch	*"Alexa, ask Twitch for followed channels"*, *"Alexa, ask Twitch to play Monstercat"*
Lifestyle	over 7.000	Sleeptracker	*"Alexa ask Sleeptracker how I slept last night"*
Movies & TV	619	MDb's What's On TV Briefing	*"Alexa, what's my Flash Briefing?"*
Music & Audio	over 6.000	Connect Control for Spotify	*"Alexa, ask Connect Control to play on device 2"*
News	over 4.000	The Washington Post	*"Alexa, ask Washington Post for headlines"*
Shopping	153	Opening Times	*"Alexa ask Opening Times for Tesco Redruth Extra"*
Smart Home	over 1.000	Smart Life	*"Alexa, set hallway light to 50 percent"*
Sports	over 1.000	PGA Tour	*"Alexa, ask PGA TOUR for the leaderboard."*
Travel & Transportation	808	Madrid Transport	*"Alexa, open Madrid Transport"* *"incoming buses at 70"*
Weather	663	Temperature Now	*"Alexa, Temperature Now"*

Como se puede notar, el rango de servicios ofrecidos por las skills es muy amplio, siendo importante destacar que para algunas categorías existen más de 4.000 skills creadas (como por ejemplo en la categoría "News").

De todos modos, esto no sería tan sorprendente si consideramos que disponemos de una gran variedad de fuentes y servicios en la Web para poder lograr un mismo propósito. Consecuentemente, sería sencillo plantear la posibilidad de poder crear nuevos tipos de skills usando los contenidos y servicios disponibles públicamente en la Web (tanto a partir de APIs Restful como directamente "parseando" y extrayendo el contenido Web deseado).

Más allá de este análisis cuantitativo, un estudio reciente [19] muestra que los usuarios de los altavoces inteligentes (el estudio fue realizado con usuarios de Google Home) utilizan skills (por orden de relevancia, desde los skills más usados a los menos usados) relacionados con la Música, Información, Automatización, Conversación, Alarma, Clima, Video, Tiempo, Listados, Otros. Esto nos permite realizar un análisis cualitativo relacionado con los tipos de skills preferidos por los usuarios. Una vez más, podemos apreciar que para la mayoría de estas categorías, existen varias contrapartes de aplicaciones Web desde donde los usuarios finales pueden leer información o completar algún proceso de negocios utilizando un dispositivo que soporte la navegación Web normal.

Por otro lado, podemos decir que un rango muy amplio (aunque no en su totalidad) de skills destinadas a la automatización de dispositivos de IoT, poseen o permiten a su vez contra-partes de funcionalidades en la web.

La posibilidad de crear comandos de voz personalizados es también relevante y ya ha sido estudiada en el contexto de las interfaces de usuario multi-modales [20]. De todos modos, aunque este sistema de personalización ofrece algún tipo de flexibilidad, los usuarios finales no son capaces de manejar por ellos mismos una especificación de VUI completa.

En nuestro trabajo, investigamos sobre cómo los contenidos Web pueden ser extraídos, procesados y utilizados (como respuesta) por aplicaciones que definen una VUI, en particular por los altavoces inteligentes.

-Diseño de VUIs

No se pueden aplicar las mismas pautas de diseño que se utilizan en interfaces de usuario gráficas para las VUI. En una VUI, al no existir la chance de poder visualizar la interfaz, muchas veces las opciones para la interacción provistas al usuario no satisfacen las necesidades del mismo y no conducen hacia una navegación eficiente dentro de la aplicación. Por este motivo, cuando se diseñan acciones en una VUI, es importante que el sistema establezca claramente cuáles son las posibles opciones de interacción,

informándole al usuario qúe funcionalidad está usando, cúal es el contexto en el que se encuentra, y limitando la cantidad de información que se le otorga al usuario a una cantidad que pueda recordar.

Jeong, J. [21] realizó un estudio en el que se discute sobre cómo algunos factores (entre ellos las limitaciones relacionadas con la tecnología o las características de los usuarios) influyen en el diseño de una VUI. En el mismo, realiza una descripción sobre los componentes que pueden propiciar interacciones basadas en voz más eficientes y efectivas entre humanos y máquinas. El estudio se centra en poder mejorar la experiencia del usuario con una VUI dentro del ambiente de los smartphones. Busca demostrar si las variantes que existen, tanto en el género (masculino o femenino) como en los modos (dinámico o más calmado) de la voz utilizada en las VUI influyen en la percepción y la experiencia de los usuarios. Se tuvieron en cuenta distintos aspectos relacionados con la percepción que tienen los usuarios durante la interacción: se pudo observar que estos se sienten más cómodos y confían más en un tipo de voz cercano a la interacción humana, y con el género masculino, asi como también con los modos de habla que son más dinámicos.

Por otro lado, se han ido desarrollando distintos patrones de diseño para simplificar las tareas a la hora de diseñar una VUI [22]. Los creadores de estos patrones buscaron abordar distintos factores que representan dificultades que se evidencian a la hora de crear una aplicación VUI, como pueden ser las limitaciones propias del medio de voz (poco tiempo disponible, las personas escuchan más lento de lo que leen, etc), la calidad de la síntesis y el rendimiento del reconocimiento de la voz.

Con la implementación de estos patrones, buscan ayudar a los diseñadores de VUIs para que puedan resolver estos problemas con éxito, ya que poder obtener un diseño de calidad de las interfaces de voz no es una tarea trivial. Plantean algunas decisiones de diseño centradas en tres aspectos diferentes de las aplicaciones VUI: estrategia de diálogo, respuesta del sistema y escenarios de uso.

En nuestro caso, nos resultó de gran utilidad poder estudiar estos patrones de diseño para VUIs, ya que nos permitió introducirnos en una nueva modalidad de diseño a la que no estabamos acostumbrados, y nos facilitó la tarea a la hora de diseñar nuestra interfaz propuesta, pudiendo estructurar la información brindada de una manera eficiente.

- Usabilidad y accesibilidad de VUIs

Algunos aspectos fundamentales a tener en cuenta a la hora de diseñar este tipo de soluciones, son la usabilidad y accesibilidad, donde se debe pres-

tar especial atención a las características de los distintos tipos de usuarios y a las dificultades que pueden llegar a experimentar a la hora de interactuar con las VUIs.

Schlögl, S. [23] plantea observar cómo interactuarán las personas mayores con una VUI: cómo estas interfaces pueden mejorar sus experiencias a la hora de interactuar con la tecnología y qué características deberán tener las interfaces para satisfacer las necesidades de este grupo reducido de usuarios.

Los autores llevaron a cabo un experimento, en el que detectaron dificultades de los usuarios a la hora de interactuar con una pantalla táctil, por lo que se puede pensar que los métodos de entrada de audio pueden ser una buena alternativa a las entradas táctiles.

Se detectó una preferencia por parte de los usuarios en las interacciones mediante una VUI respecto a la interacción con una GUI (Interfaz de Usuario Gráfica). Teniendo en cuenta las interacciones mediante VUIs, existe una preferencia de los usuarios hacia un lenguaje natural por sobre los comandos de voz. Por otro lado, queda claro que este tipo de interacciones deberá centrarse en brindar una buena usabilidad, pudiendo ofrecer estrategias de recuperación de errores, feedback y alguna forma de retroceder en caso que surjan problemas o la interacción se vuelva tediosa, especialmente para este tipo de usuarios.

En concordancia con nuestro enfoque, se buscó probar que los usuarios se sienten más cómodos a la hora de interactuar con algunas funcionalidades presentes en los dispositivos tecnológicos, mediante una interacción VUI respecto a las interacciones clásicas GUI que estamos acostumbrados a usar.

2.1.2. End-User Programming

El término End-user programming (EUP) o End-user development (EUD) hace referencia a las actividades y herramientas que permiten a los usuarios finales (personas que no tienen conocimiento en el desarrollo de software) poder programar. Los usuarios finales pueden usar herramientas EUD para crear o modificar artefactos de software y objetos de datos complejos sin tener conocimientos de ningún lenguaje de programación.

En la actualidad existen varios enfoques EUD, por lo que es un tema de investigación muy activo dentro del campo de ciencias de la computación y de la interacción humano-computadora.

Un ejemplo de este tipo de herramientas es Contour [24]: un prototipo que permite al usuario final crear contenido basado en text-to-speech o TTS (texto transformado a voz) con un tono emocional usando la voz como método de entrada. El objetivo es permitir al usuario crear TTS que sean más

expresivos. Dicho prototipo cuenta con una GUI que se utiliza para controlar parámetros del tono de la voz, y una VUI, que consiste en una interfaz de voz personalizada y una serie de algoritmos, capaces de analizar la señal del speech o audio para re-sintetizarla con el objetivo de que se aproxime al estilo buscado.

En base a los resultados del experimento llevado a cabo por los autores, se llegó a la conclusión de que el workflow (flujo de trabajo) con entrada de voz que ellos proponen es más eficiente en términos del tiempo que toma completar las tareas y cantidad de iteraciones en el diseño, respecto al workflow con entrada parametrizada, mientras que ambos mantienen el mismo nivel de eficacia en términos de calidad de producción preferida por el usuario. En términos generales, los usuarios del experimento prefirieron en su mayoría usar el workflow con entrada de voz propuesto.

A pesar de que el enfoque del trabajo descrito en el párrafo anterior es diferente al de nuestra investigación, contribuyó a darnos cuenta de la necesidad que existe de poder crear TTS que sean mucho más expresivos y cercanos al tono de conversación real, frente al creciente avance de la tecnología de los asistentes de voz, y al creciente consumo de estos dispositivos. Este concepto nos permitió diseñar de una manera más apropiada las interacciones que un usuario deberá llevar a cabo con nuestra aplicación desarrollada para Alexa.

Por otro lado, hay que destacar que nos inclinamos por utilizar el servicio de Alexa para crear nuestra aplicación ya que, entre otras cosas, posibilitó la configuración de distintos parámetros destinados a mejorar la expresividad en las interacciones por voz (por ejemplo relacionados con el tono de voz o la velocidad del habla).

2.2. Trabajos relacionados

En esta sección se incluirán trabajos existentes que plantean soluciones a problemas relacionados a la problemática de nuestro trabajo.

Soic, R. [25] llevó a cabo una investigación que se centra en el desarrollo de un modelo que brinde la posibilidad de traducir los contenidos web desde el formato texto a voz. Uno de los mayores problemas a los que se enfrenta se relaciona con la poca relevancia que se le otorga a la semántica utilizada en los sitios web en general.

El desarrollo de la solución presenta algunos inconvenientes, ya que requiere que se analice previamente el sitio completo además de obtener la semántica a partir de cada línea del código fuente del mismo. Cabe destacar

que el modelo está diseñado para funcionar con sitios web específicos que se adecúen a ciertas reglas.

Al tratarse solo de un prototipo que no permite aplicar la herramienta a la mayoria de los sitios web existentes, ya que depende mucho de la estructura que posean, no es posible resolver la problemática planteada en nuestra investigación. Además, hay que destacar que tampoco se tienen en cuenta otros factores relacionados con la interacción por parte de los usuarios finales.

SpeechEnabler [26] es un framework que permite a los creadores de los sitios web existentes, poder adaptar sus funcionalidades para que sean más accesibles a personas no videntes, y que de esta manera no tengan que depender de ninguna herramienta extra que les brinde asistencia, como pueden ser los screen readers (lectores de pantalla).

Un screen reader es un software que trata de interpretar aquello que se muestra en una pantalla, para luego utilizar un mecanismo sintetizador de texto a voz para representar el contenido al usuario. Este tipo de herramientas, a pesar de ser útiles para estos usuarios cuando realizan tareas simples en la web, presentan complicaciones a la hora de realizar tareas más complejas como puede ser recorrer tablas o llenar formularios.

Los autores brindan a los creadores de los sitios web la posibilidad de solucionar problemas de accesibilidad desde el lado del servidor de forma automatizada. Se proponen mejorar la experiencia de los usuarios con dificultades visuales, otorgándole la posibilidad de realizar tareas complejas como puede ser hacer una reserva (con el uso de los screen readers, se deben recorrer todos los links de una página para acceder al link que sea de interés para el usuario, lo que dificulta la experiencia), a partir de un sistema aparte basado en la voz que sea accesible desde la misma página web.

Los desarrolladores de los sitios deberán entonces definir cuáles son sus funcionalidades clave y asociarlas a "atajos" dentro de la misma página web que "disparen" una interacción por voz, en la que el usuario recibirá cada respuesta a sus acciones en formato de audio. Sin embargo, hay que tener en cuenta que por cada acción el usuario deberá ingresar todos los datos de entrada mediante el teclado, por lo que no es una interacción puramente auditiva.

Voice Browser es una aplicacion de software que presenta una VUI interactiva para el usuario, que funciona de forma análoga a cómo funcionan los Web Browsers al interpretar contenido HTML (Hypertext Markup Language). Los documentos de diálogo interpretados por un voice browser por lo general están encodeados en lenguajes de marcado basados en estándares,

como Voice Dialog Extensible Markup Language (VoiceXML), un estándar reconocido por la World Wide Web Consortium.

Un voice browser presenta información de forma auditiva, reproduciendo archivos de audio pregrabados o mediante software de síntesis text-to-speech. Obtiene la información utilizando tanto reconocimiento de voz como entradas de teclado.

Con Sasayaki [27] los autores se plantearon mejorar la calidad de la navegación dentro del entorno de un voice browser mediante la creación de un prototipo de agente de voz. El mismo se enfoca en resolver las dificultades que experimentan las personas con problemas en la visión al querer navegar en la Web.

El trabajo consiste en el desarrollo de un plug-in para un voice browser específico (aiBrowser), que apunta más a resolver los problemas relacionados con la pérdida de detalles de las páginas Web que ocurre cuando se intenta adaptarlas a un nuevo formato de voz, y principalmente con la pérdida de contexto que experimentan los usuarios cuando navegan en una página web al interactuar con este tipo de navegadores.

En base a un estudio piloto se pudo llegar a la conclusión de que este agente puede mejorar considerablemente la experiencia de navegación de los usuarios que poseen dificultades visuales. Mediante su uso, los usuarios demoraron menos tiempo navegando hacia elementos específicos de una página web y pudieron recuperar la información requerida con mayor rapidez.

Asimismo, este estudio piloto no refleja con datos certeros que los sitios web específicos accedidos por los usuarios contengan en su implementación todas las características necesarias para su interpretación y posterior navegación por medio del Voice browser “aiBrowser”. Por lo que, si bien es una iniciativa muy auspiciosa para la navegación de sitios web a partir de la interfaz de voz, aún así está limitada por las implementaciones de los sitios web, que en general no siguen los estándares normalizados y estipulados para poder brindar accesibilidad. Por este motivo, nuestra solución no implicará la adopción de estándares por parte de terceros, para lograr así una correcta ejecución de la implementación realizada.

Capítulo 3

Análisis de contenidos Web y posibilidad de adaptación hacia VUIs

3.1. Disposición de contenidos dependiendo del medio

El consumo de información por parte del usuario a través de una aplicación web, como ya hemos discutido anteriormente, es parte de la evolución de internet y es uno de los servicios brindados más comunes actualmente. Como consecuencia de esto, se ha extendido un amplio campo para la investigación y estudio, retroalimentado por el mismo consumo y las experiencias de los usuarios.

A partir de la observación de distintos sitios con contenidos web, podemos evidenciar la existencia de patrones de lectura e interacción del usuario concretos y claramente establecidos. Estos patrones se ven aún más marcados en sitios de contenidos de noticias, aunque esto no implica que sean exclusivos de los mismos. También se pueden observar en sitios web de contextos diversos (como pueden ser sitios corporativos, gubernamentales, etc). Los patrones que se evidencian, no son más que estructuras, mediante las cuales los sitios web enfatizan sus contenidos y exponen aquella información que se quiere resaltar circunstancialmente en ese momento. Más allá de su objetivo, hay que destacar que estas estructuras surgen como respuesta a las limitaciones que existen en los sitios web. Al igual que sucede con los medios de noticias tradicionales (como por ejemplo diarios o revistas), los sitios web disponen de un espacio físico limitado [28]. Cada apartado de contenido del

sitio se verá restringido por las limitaciones propias del medio.

Estas estructuras de contenidos mencionadas son parte de estrategias utilizadas dentro de un marco exclusivamente visual. Por lo tanto, no se condicen con las alternativas que se utilizan, por ejemplo, en otros tipos de medios como la radio. En este contexto, el único patrón que se puede establecer es la división por secciones. Las noticias son enunciadas comenzando por el título, siguiendo luego con su contenido concreto, una tras otra y por lo general siendo separadas por sonidos que indican que se comienza a leer la próxima noticia (de manera secuencial). Podemos inferir entonces que existe un orden para la redacción de las noticias, establecido probablemente en base a la importancia de las mismas.

Sin embargo, se podría decir que estamos en presencia de una estrategia mucho menos efectiva respecto a las estrategias visuales, debido a que el medio en cuestión (el sonido) no permite brindar un panorama inicial al lector (en este caso oyente) para que disponga de todos los contenidos en simultáneo. Otro punto interesante a contrastar respecto a los sitios de contenidos web son las limitaciones que posee este medio. Como se ha mencionado anteriormente, el espacio visual limita las posibilidades de mostrar los contenidos. En el caso de las interfaces de medios auditivos tenemos otro tipo de limitación, que si bien no es tan evidente como la anterior, juega un rol muy importante. Estamos hablando del tiempo. El mismo limita los contenidos de medios auditivos desde otro ángulo, ya que la percepción e interpretación de la información por parte del usuario se genera en otro ámbito sensorial, donde la capacidad de poder expresar conocimiento en un período de tiempo corto de manera eficaz, es la principal herramienta que disponemos para poder favorecer la asimilación de la información y la consecuente satisfacción del usuario.

3.1.1. Disposición de contenidos en interfaces de usuario gráficas

Para dar noción de los patrones de exposición de contenidos web en los diferentes sitios, nos parece importante analizar un caso donde se evidencie un patrón de exposición de noticias en la portada de un sitio web. En este caso optamos por el portal de noticias “New York Times”.

A continuación analizaremos distintos casos usuales de visualización de contenidos.

Noticias de portada

En la portada del sitio podemos ver una noticia ubicada en el centro

que predomina **(2)** y que ocupa prácticamente el doble de espacio que el resto. Estamos en presencia de la noticia principal, conformada por un título, una breve introducción y contenido multimedia reproducible. Luego, en la izquierda, tenemos una noticia bastante más reducida en tamaño **(1)** en comparación con la noticia principal, pero que muestra el título mucho más grande que el resto de las noticias de la portada. En el costado derecho de la portada, se muestran distintos artículos de opinión de la editorial, destacándose entre ellos los más populares **(3** y **4)** conformados por un título,una introducción y una imagen, y otros **(5)** en los que sólo figura el título. En todos los casos es posible acceder a la noticia completa, presionando en el link que representa cada título de cada noticia.

Figura 1: Distribución de contenidos en la portada de un sitio

Noticias de una sección en particular

En el caso que se quiera acceder a una sección en particular, primero se deberá seleccionar una sección del menú principal situando el mouse sobre alguna de las secciones disponibles, y luego habrá que seleccionar mediante un click alguna subsección del submenú desplegable (en el ejemplo, "World" y "Americas" respectivamente). Una vez que se ingresa en la sección elegida, podemos ver que se listan las últimas noticias pertenecientes a la misma, sin

distinciones marcadas entre unas y otras, siguiendo un ordenamiento cronológico basado en el momento en que fueron subidas al sitio: encontrándose en la parte superior las noticias más recientes de la sección, y en la parte inferior las noticias más antiguas. Se puede apreciar que todas están conformadas por un título, un resumen o introducción y una imagen.

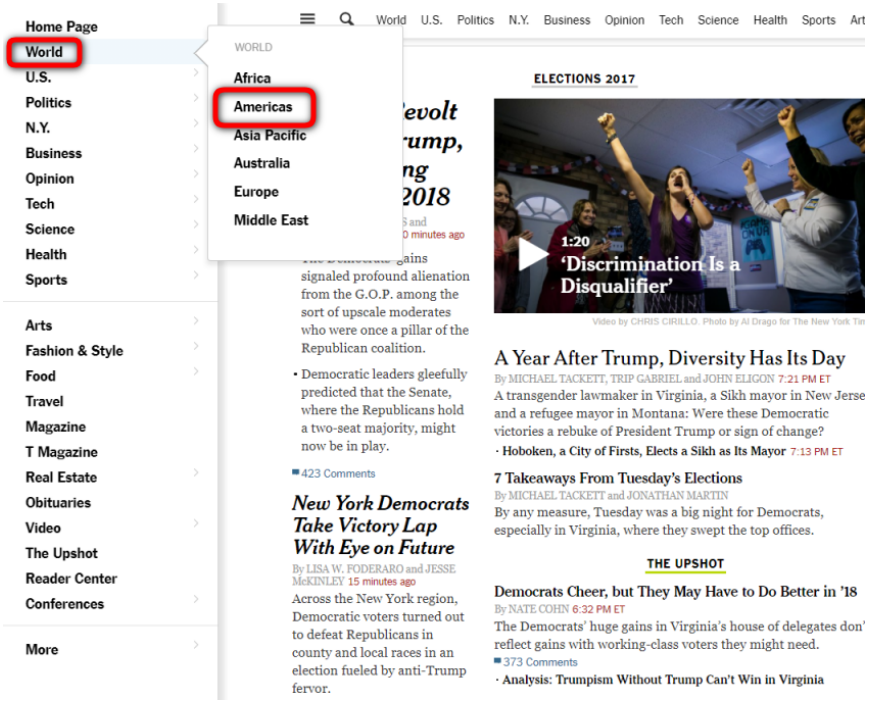


Figura 2: Acceso a una sección en particular de un sitio

Figura 3: Distribución de contenidos en una sección particular de un sitio

Noticia específica y sugeridos

Al ingresar en una noticia específica desde la portada o desde una sección particular, podemos ver que la misma está estructurada de la siguiente manera:

En la parte superior, se puede observar además de la sección a la que pertenece, el título de la noticia **(1)** con un formato de letra más grande que el resto del contenido. Luego podemos ver la presencia de una imagen junto a su pie de foto y, debajo de la misma, el resto del contenido de la noticia separado por párrafos.

Por otro lado, pero no menos importante, podemos observar que a la derecha figura un apartado **(2)** en el que se listan otras noticias relacionadas con la noticia en cuestión. El listado parece mostrar las noticias en un orden

aleatorio. Por cada noticia relacionada, se puede ver que se dispone del título junto con una imagen.

Figura 4: Distribución de la información de un contenido específico

3.1.2. Ejemplificación de contenidos varios (e-commerce)

Hasta el momento, nos hemos referido a sitios web cuyo contenido hace referencia a noticias o información que está distribuida con un formato bastante general y preestablecido. Sin embargo, los contenidos web pueden ir variando su estructura, ya que su propia naturaleza exige una diferenciación de características relacionadas con su propio fin o tipo de datos concretos. Por este motivo, podemos definir como contenido web también a otro tipo de información, que posee otros fines y fundamentos para ser visualizado en la web.

Si bien los ejemplos definidos anteriormente son representativos para una gran cantidad de sitios web (inclusive para aquellos sitios que no son concretamente portales de noticias, sino que expresan contenidos actualizados de manera esporádica o dan información acerca de una temática en particular),

aún existen otros tipos de contenidos web cuyos patrones de visualización no coinciden con los mencionados previamente. Podemos incluir dentro de estos tipos de contenidos a aquellos referidos al "e-commerce". El e-commerce consiste en la compra y venta de productos o de servicios a través de medios electrónicos, representado principalmente por empresas como Amazon, Ebay y AliExpress en todo el mundo, y MercadoLibre en latinoamérica. Este tipo de sitios ofrece una interfaz de usuario que no sólo apunta a lograr que el usuario encuentre lo que está buscando, sino que también intenta facilitar el proceso de compra y mantener al usuario visualizando productos la mayor cantidad de tiempo posible [29]. Por este motivo, las compañías no escatiman en tiempo ni gastos a la hora de definir las estrategias de visualización de contenidos, por lo que los patrones que se evidencian en este tipo de sitios merecen un cuidadoso análisis. Nos enfocaremos en Amazon para ejemplificar los patrones de interfaz de usuario gráfica, y también podremos analizar cómo la empresa plasma sus principios de interacción de usuario por voz en su dispositivo Amazon Echo. Vale aclarar que la interfaz web de este sitio suele actualizarse frecuentemente, realizando distribuciones de contenidos diferentes en base a los eventos del momento.

Contenidos de portada

Para empezar, nos centraremos en la portada del sitio. En esta sección se puede diferenciar concretamente una secuencia de imágenes cambiantes ubicadas en el tope de la página, que muestra aquellos productos o servicios que se consideran más atractivos en ese momento **(1)**. Debajo de esta serie de imágenes, se deja ver una categorización de productos variada y combinada (por ejemplo Holiday Toy List, que significa juguetes para las vacaciones) **(2)**. Y debajo de estas secciones se proporcionan listados de productos de categorías específicas **(3)**.

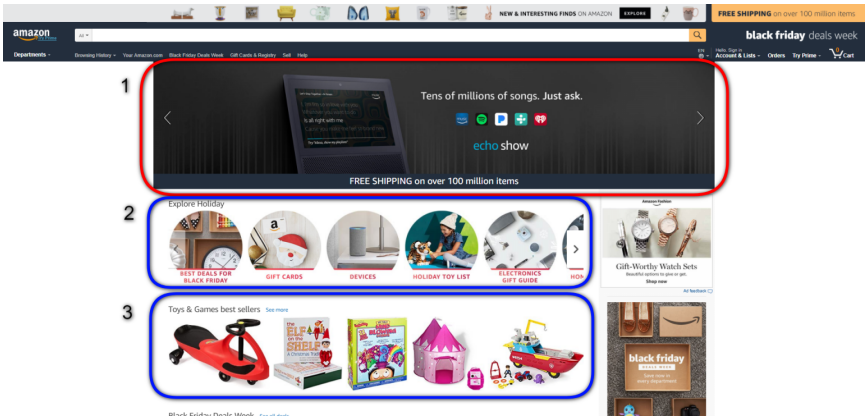


Figura 5: Distribución de contenidos en la portada de un sitio e-commerce

Contenidos de una sección en particular

Al margen de la diversidad de los contenidos web que podemos ubicar en los sitios, se pueden detectar patrones de distribución de información similares en todos ellos. A continuación, vamos a ejemplificar este concepto a partir de un patrón muy común que podemos encontrar prácticamente en cualquier sitio de contenidos. Este tipo de acceso y distribución ya fue mostrado en la ejemplificación del portal 'New York Times'. En Amazon, también nos encontramos con la posibilidad de utilizar clasificaciones de contenidos (en este caso, productos) y poder luego seleccionar uno en particular. Al seleccionar la opción "Departamentos", se nos provee un listado de clasificaciones de productos al que podemos acceder **(1)** situando el mouse sobre el mismo y luego otro listado de subclasificaciones a los cuales podremos acceder mediante un click **(2)**. Por último, tendremos disponible una categorización propia del tipo de producto seleccionado **(3)** y el listado de productos ordenados por algún criterio **(4)**. Cada producto dispone de un nombre, calificación (estrellas), descripción corta y precio (además de una etiqueta propia del sitio que diferencia productos que pueden ser adquiridos a un menor precio si contamos con una suscripción "Prime").

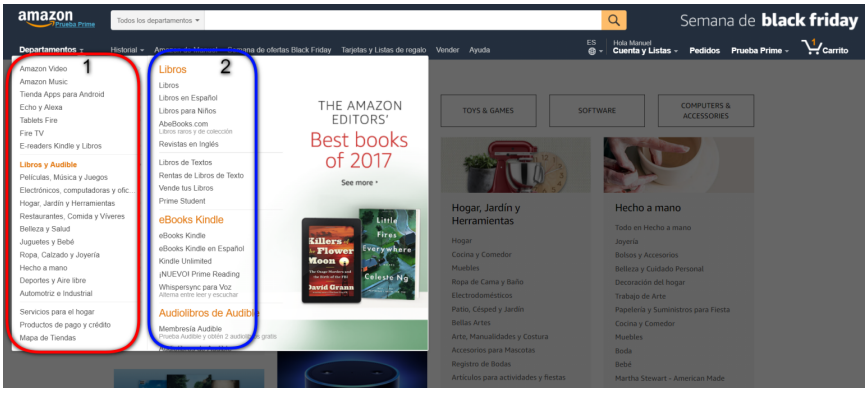


Figura 6: Clasificaciones de productos en un sitio e-commerce

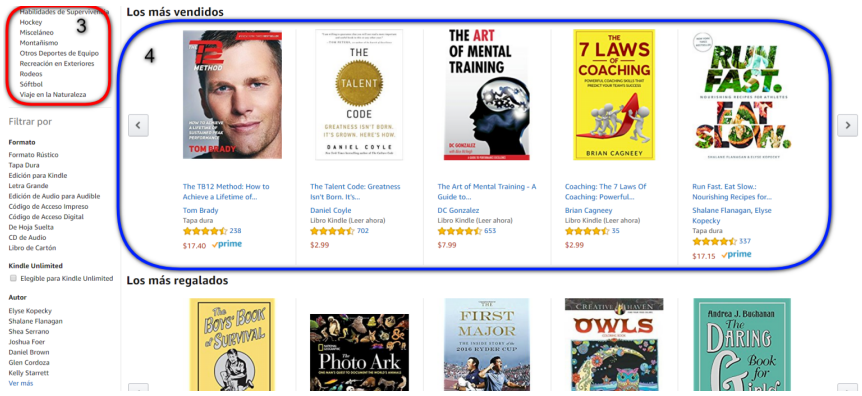


Figura 7: Distribución de productos de la clasificación seleccionada

3.2. Adaptación de contenidos web hacia interfaces de usuario por voz

Ya habiendo mencionado brevemente distintos criterios y estrategias de disposición de contenidos en medios visuales como los sitios web, seguramente podemos pensar que estas estrategias (en conjunto con muchas otras) son fáciles de entender e implementar. Esto se debe a que desde hace 15 años como mínimo, nos encontramos diariamente con este tipo de implementaciones, y la realidad indica que fueron muy efectivas a pesar del paso

del tiempo. Ya estamos acostumbrados a encontrarnos con interfaces de tipo visual, por lo que nos resultan muy naturales en nuestra vida diaria.

Sin embargo, el surgimiento de los dispositivos de interfaz de voz significó un gran avance para la accesibilidad y el consumo de contenidos, ya que logran brindar al usuario una interfaz que le permite no solamente adquirir contenidos sin utilizar las manos para generar una entrada, sino que también logra que el mismo interactúe sin la necesidad de utilizar el sentido de la vista [21]. Además, la capacidad de oír el contenido favorece su asimilación, de manera que los costos para la absorción e interpretación de información disminuyen. Con la aparición de los dispositivos de interfaz de voz, nos encontramos ante un mundo nuevo de estrategias y patrones a los cuales todavía no estamos acostumbrados. Será responsabilidad de quienes implementen estas nuevas interfaces permitir que los usuarios se acostumbren a las mismas y renovar eficazmente su diseño mediante el feedback provisto por los mismos. Más allá de esto, resulta por demás interesante analizar y proponer alternativas que podrían implementarse.

La tecnología seleccionada para la edificación de la propuesta tiene como principal atractivo y fundamentación la utilización de la voz para la obtención de información preexistente en la web en forma de texto. Si bien podemos pensar en muchos aspectos positivos para esta nueva interacción entre los usuarios y los dispositivos, es necesario analizar en detalle las distintas maneras que se pueden plantear en el momento de interacción entre estos. El concepto de comunicarnos a través del sonido y obtener la información requerida es claramente muy auspicioso. Evidencia un claro avance en la forma de comunicarnos y el lugar que le damos a los dispositivos en los ambientes.

No obstante, el alto nivel de accesibilidad que brinda tener este tipo de relación entre el usuario y el sistema significa, de la misma manera, un alto nivel de costos en tiempos de diseño, análisis y composición de diseños de interacción que se deben implementar ante los diversos casos de utilidad. Aunque las interfaces de usuario auditivas sean accesibles para los usuarios sin la necesidad de utilizar la vista, se advierten grandes dificultades para diseñarlas de manera eficaz. Los usuarios, de manera frecuente, tendrán modelos mentales errados y poco consistentes de lo que pueden (o no) decir para interactuar de manera correcta con el sistema [30]. No podemos comparar lo fácil que nos resulta interpretar nuestro campo de acción en un contexto concreto cuando lo leemos en comparación de cuando disponemos de sentidos alternativos. Los usuarios no disponen de las mismas herramientas para analizar sus posibilidades cuando la interfaz involucra la voz y el oído. La interacción con esta nueva interfaz de usuario no solo estará determinada

por las limitaciones naturales de un canal de voz, sino que también juegan un papel decisivo los factores humanos: capacidades auditivas y de orientación, atención, claridad, dicción, velocidad y ruido ambiental [22]. Al mismo tiempo, cabe destacar que los usuarios al no estar familiarizados a interactuar mediante la voz con la tecnología, desconocen el alcance que puedan llegar a tener sus funcionalidades.

Entre las principales dificultades que podemos evidenciar, está la necesidad de memorizar el contexto en el que se encuentran dentro de la aplicación, debido a que es casi imposible obtener todos los factores contextuales y suposiciones necesarios en un breve intercambio, por lo que habrá que encontrar maneras de guiar al usuario para ubicarlo en contexto. En las interfaces gráficas de usuario, los usuarios, por ejemplo, pueden ver cuando entran en una nueva sección de un sitio. En cambio, en las interfaces de voz de usuario, necesitan conocer qué funcionalidad están utilizando, ya que pueden llegar a confundirse fácilmente acerca del contexto en el que se encuentran dentro de la aplicación, o pueden querer activar por error alguna funcionalidad que no está permitida. Para que las interfaces de voz puedan tener éxito, y lograr que el usuario las adapte para el uso cotidiano, deberán tener estrategias eficientes de recuperación ante errores, brindar feedback suficiente para el usuario, así como también la posibilidad de "escapar" del contexto de la aplicación en el que se encuentra un usuario, para los casos en donde este se encuentre "atrapado" [23].

Ejemplificando estos conceptos, situémonos en el contexto de un juego de cartas. Para representar la situación no es necesario nombrar cuál sería el juego en cuestión. Basta entender que el jugador tiene un conjunto de cartas disponibles y en la mesa hay otro conjunto de cartas. El jugador puede interactuar con las mismas y de esa manera resolver y ganar el juego. Si esta ejemplificación la planteamos en términos de una GUI (Graphical User Interface), no precisamos demasiada imaginación para entender que debemos mostrar de manera diferente las cartas del usuario con respecto a las cartas de la mesa. Rápidamente podemos pensar en mostrar las cartas propias del jugador más cerca que las de la mesa. En cambio, llevando la situación a una VUI, podemos sentirnos faltos de ideas y podemos pasar un rato largo diseñando alternativas que apliquen a la situación del juego y su jugabilidad..

No nos centraremos en la resolución de la problemática anterior, sino en presentar alternativas de diseño de VUIs ante nuestro contexto actual. Por lo cual, a continuación analizaremos qué opciones tenemos a nivel diseño de interfaz para lograr que un usuario pueda acceder a su contenido favorito mediante la utilización de la voz.

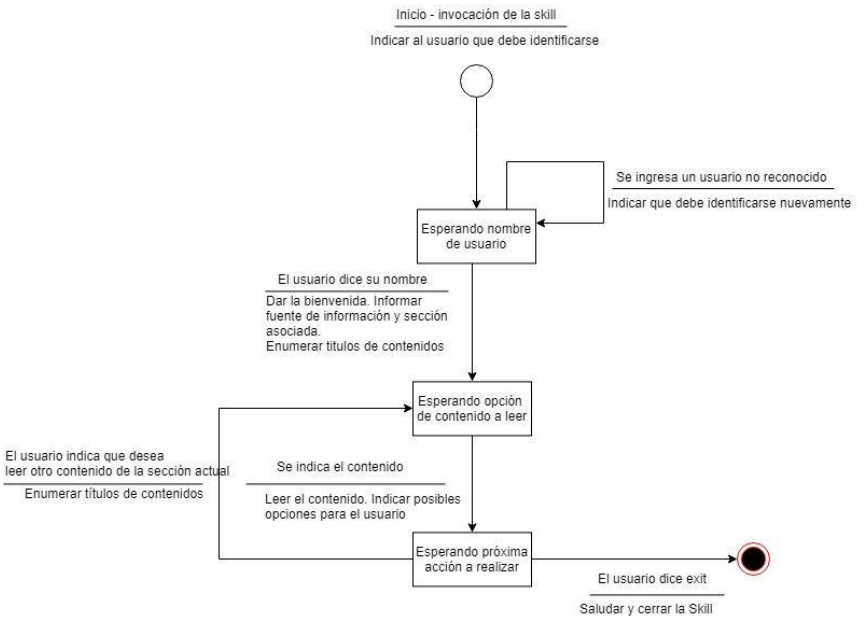
3.3. Escenarios de análisis

Ante el contexto planteado y contextualizando el ambiente propuesto para nuestro desarrollo, hemos propuesto alternativas que verifican el tratamiento de la información y le dan herramientas al usuario para poder interactuar eficazmente y obtener los resultados esperados.

Mediante diagramas de estado, se describen algunos diseños de interfaz auditiva que ejemplifican cómo podría ser el flujo de interacción entre un usuario y la aplicación que proporcionará el servicio. Los escenarios que se describen a continuación pueden derivarse en nuevos diseños mejorados o simplificarse de ser necesario.

Escenario 1

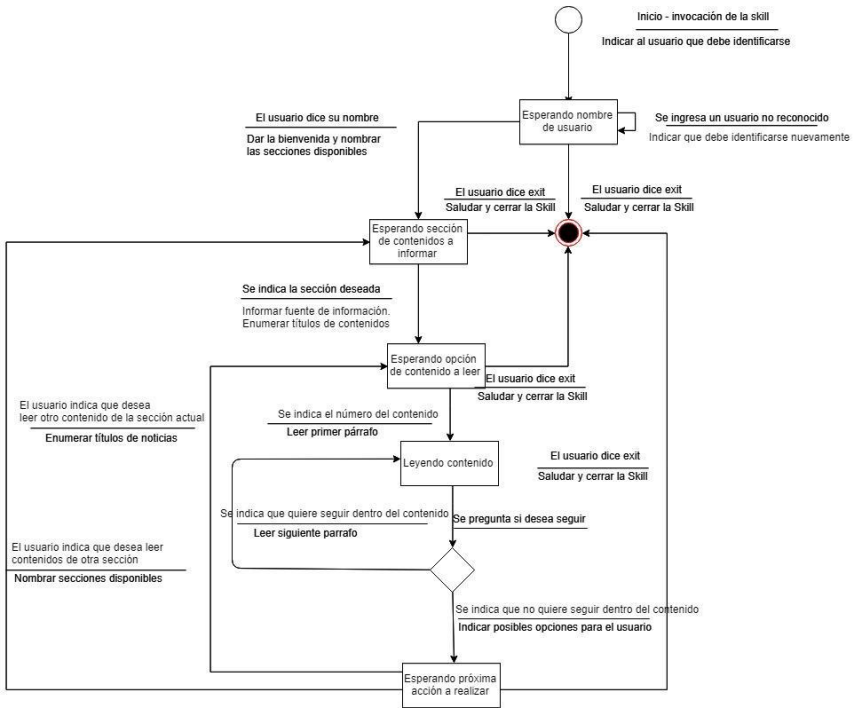
El usuario ingresa mediante su nombre de identificación al sistema y el sistema le da la bienvenida mediante un saludo. Luego de esto, el sistema le informa cada sitio web disponible desde donde podrá obtener los contenidos junto con la sección asociada. Ej: "Ud tiene disponible la sección 'X' para el sitio 'Y'. A continuación, se comenzarán a listar los títulos de los contenidos". Al finalizar la lectura de todos los títulos, el usuario deberá indicar el título que desea escuchar y el sistema comenzará a leer todo el contenido. Una vez escuchado, se le consultará al usuario si desea seguir escuchando contenidos de la misma sección o si desea salir de la aplicación.



Escenario 2

El usuario ingresa mediante su nombre de identificación al sistema y el sistema le da la bienvenida mediante un saludo. Luego de esto, el sistema lo ubica en contexto y le informa las secciones que tiene disponible listándolas. Ej: "Ud dispone de las siguientes secciones. Si ya sabe cuál es la sección a la que desea ingresar, nómbrela. Sino diga 'my feed' y se le nombraran las secciones que tiene asociadas". Si el usuario indica la sección a la que desea ingresar, la aplicación le informará la sección elegida y comenará a leer los títulos de cada contenido, no sin antes indicar de qué sitio se están leyendo las noticias. Ej: "Sección 'deportes basquet' del diario Olé noticias de basquet".

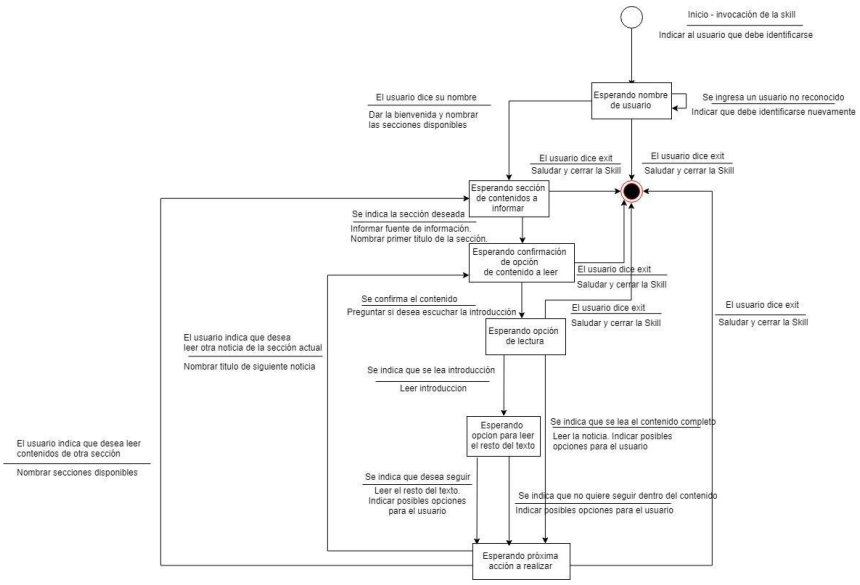
Luego de cada título leído, se dirá un número asociado para que el usuario, en caso de sentirse interesado por el mismo, recuerde. Al finalizar la lectura de todos los títulos, el usuario pronunciará el número del título que desea escuchar y el sistema comenzará a leer todo el contenido. Por cada párrafo, el sistema consultará al usuario si desea seguir escuchando el resto del contenido. Una vez escuchado en su totalidad, se le proporcionará al usuario la capacidad de seguir en la misma sección o seleccionar una nueva. Luego de la selección de la sección, el flujo se repite de la manera ya descrita.



Escenario 3

El usuario ingresa mediante su nombre de identificación al sistema y el sistema le da la bienvenida mediante un saludo. Luego de esto, el usuario podrá indicar directamente (sin escuchar las opciones de secciones que tiene disponibles) la sección que desea escuchar. Si no desea hacer esto, debe indicarle al sistema que desea escuchar las secciones que dispone, para luego nombrar una al final de la lectura.

Una vez pronunciada la sección, el sistema la informa y comienza a leer los titulos que la componen. Ante cada título leído, el sistema consulta al usuario si desea escuchar primero una introducción o directamente el contenido completo. Luego de escuchar la introducción, se vuelve a consultar al usuario para ver si éste desea escuchar el resto del contenido. Una vez escuchado el contenido en cuestión, el sistema consulta si desea seguir escuchando contenidos de la sección actual o si desea pasar a otra sección.

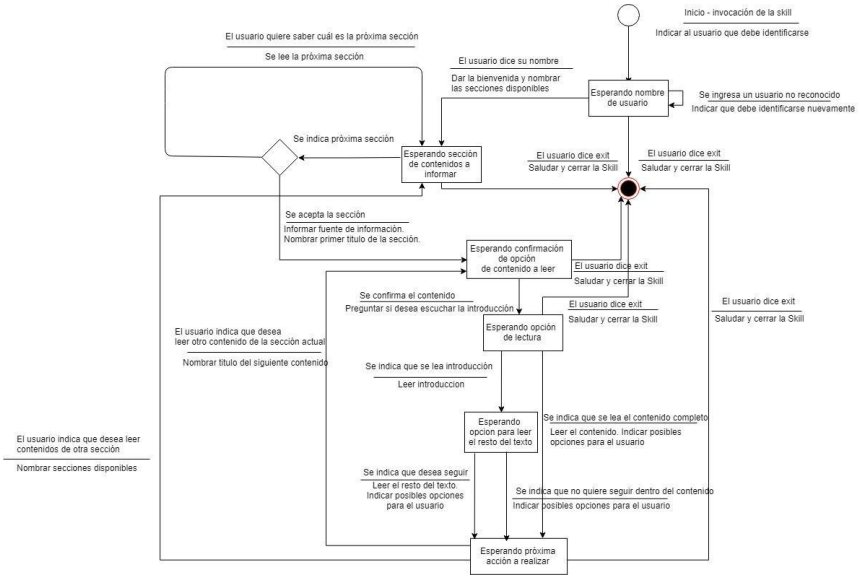


Escenario 4

El usuario ingresa al sistema indicando su nombre de identificación y este le da la bienvenida mediante un saludo. Luego, el usuario podrá indicar directamente (sin escuchar las opciones de secciones que tiene disponibles) la sección que desea escuchar. Si no desea hacer esto, debe indicarle al sistema que desea escuchar qué secciones dispone, por lo que el sistema comienza a nombrar de a una sección por vez, esperando que el usuario defina, al término de cada lectura de sección individual, qué es lo que desea hacer, pudiendo elegir entre dos opciones: si le interesa la sección nombrada, se deberá indicar que se seleccionará la misma; si por el contrario no le interesa, se deberá pedir al sistema que se le nombre la siguiente sección disponible, y así siguiendo hasta que se seleccione alguna sección. A diferencia del escenario anterior, existe una mayor participación del usuario a la hora de interactuar con el sistema, ya que es el usuario quien lleva la conversación adelante, indicando cada paso a seguir.

Una vez elegida una sección, el sistema informa la sección seleccionada y comienza a leer los títulos. Ante cada título leído (se indica también la web de la que se podrá obtener el contenido), el sistema consulta al usuario si desea escuchar una introducción o directamente el contenido en su totalidad. Luego de escuchar la introducción, se vuelve a consultar al usuario para ver si éste desea escuchar el resto del contenido.

Una vez escuchado el contenido en su totalidad, el sistema consulta si desea seguir escuchando contenidos de la sección actual o si desea cambiar de sección. En el caso que se decida seguir en la misma sección, se procederá a repetir el proceso de lectura de títulos. Se analizará la posibilidad de agregar una opción de "ayuda" en la que se podrán listar ejemplos de interacción que sirvan de guía para el usuario.



Escenario 5 (seleccionado)

Más allá de todas los escenarios antes detallados, creemos que el usuario común de la web dispone de muchas alternativas para la lectura de los contenidos. Por este motivo, el objetivo principal que nos propusimos lograr es que el usuario sea capaz de poder alcanzar la mayoría de las posibilidades y elecciones que realiza al consumir la web en su formato original.

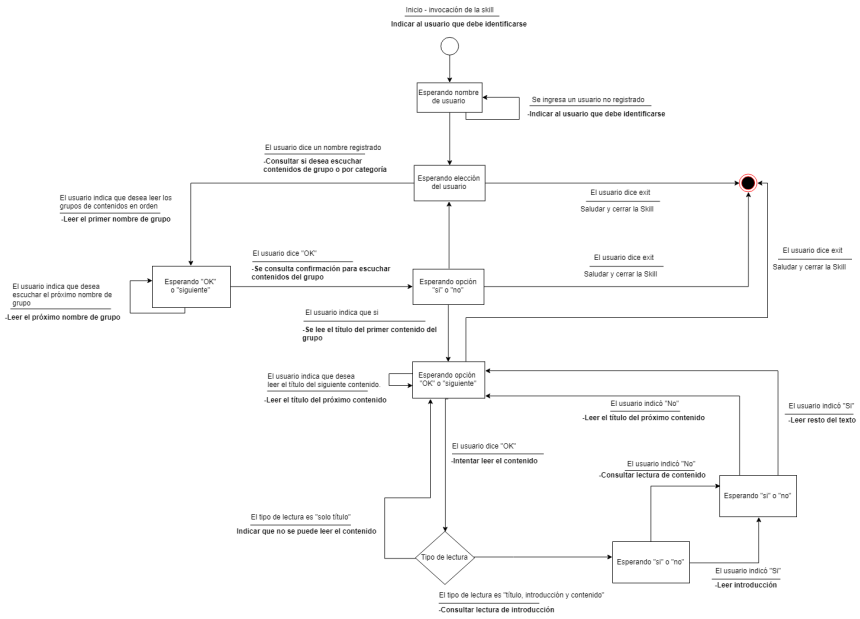
Finalmente nos decidimos por una idea más ambiciosa y personalizada, que ofrezca un trato diferenciado para los contenidos, permitiendo agruparlos de forma automática (por categoría o temática) o de forma personalizada mediante la creación de grupos de contenidos. Todas estas posibilidades que se le brindan al usuario implican una interacción mucho más compleja, que debe ser especificada criteriosamente. La interacción del usuario será de la siguiente manera:

El usuario ingresa al sistema con su nombre registrado previamente. Inmediatamente es consultado acerca de si desea leer sus contenidos definidos

agrupados bajo el concepto de “grupos de contenidos” o si, en cambio, desea que sus contenidos sean leídos en base a la categoría del mismo (deportes, política, espectáculos, etc). Si el usuario decide leer sus contenidos en base a los “grupos de contenidos”, entonces se leerán los nombres de los grupos a medida que el usuario lo indique, mediante la palabra “siguiente”. Una vez que el usuario encuentra un grupo de contenidos en particular para escuchar, lo indica mediante la orden “OK”.

Al indicar “OK”, el sistema informará cuál es el tipo de patrón de lectura que se utilizará para leer los contenidos del grupo indicado (más allá de que también cada contenido puede tener indicaciones específicas acerca de cómo debe ser leído). Luego, se indica al usuario que confirme si desea leer los contenidos en cuestión mediante los comandos “Si” o “No”. Si indica “No”, se le consultará nuevamente si desea escuchar los contenidos por grupos de contenidos o por categorías.

Una vez que el usuario decide escuchar los contenidos del grupo, se comenzará a listar uno por uno los títulos mediante la palabra “siguiente”. Si el usuario desea que se le lea alguno de los contenidos nombrados, debe indicarlo mediante “OK”. A su vez, cada contenido puede tener una indicación en particular (más allá del patrón de lectura seleccionado para el grupo de contenidos). Si no tiene ninguna indicación particular, la manera en que se leerá el contenido será el especificado a la hora de la definición del grupo. Los distintos patrones de lectura de contenidos disponibles son: “Leer solo título”, “Leer título, introducción y contenido” o “Consultar antes de leer”. El método de lectura de contenidos del grupo se verá modificado en base a estos tres tipos de lectura.



3.3.1. Interfaz de voz implementada por el sitio de e-commerce Amazon

Anteriormente en este capítulo, hemos utilizado la web de Amazon para referenciar y mostrar cómo funciona un sitio con contenidos relacionados al e-commerce. Por lo tanto, resulta muy interesante analizar y contrastar la implementación de interfaz por voz realizada para esta misma tienda. Entre las posibilidades que brinda la interfaz, tenemos la capacidad de pedir un producto en particular mediante el comando de voz “Alexa, order [producto deseado]”. En este caso, el dispositivo nos recomendará un producto en particular basado en el precio, la calidad y el envío del mismo. Luego de esto, el usuario puede confirmar la acción y comprar el producto en cuestión. De manera similar funcionan otras opciones, como por ejemplo la que permite comprar nuevamente un producto en particular. La aplicación detecta si ya hemos comprado el tipo de producto y nos sugiere exactamente el producto que ya hemos comprado anteriormente.

Podemos encontrar también el concepto de carrito de compras, que está totalmente sincronizado con la interfaz web. El carrito, si bien tiene el mismo concepto, funciona de manera diferente. El usuario puede agregar al carrito ciertos productos. Si el tipo de producto ya se compró anteriormente, añadirá al carrito un producto de la marca y modelo del producto compra-

do previamente. Si en cambio, el tipo de producto no fue comprado antes por el usuario, una vez que el usuario ingrese mediante la interfaz web, se le proporcionará al usuario un conjunto de sugerencias y éste tendrá la posibilidad de seleccionar un producto en particular para ese tipo de producto del carrito. Entre otras características importantes, tenemos la posibilidad de preguntar al dispositivo acerca de las ofertas más importantes, mediante el comando "Alexa, what are your deals?", así como también la chance de verificar el estado y lugar en el que se encuentra un producto que hayamos comprado mediante el comando "Alexa, where is my stuff?".

Vale la pena aclarar que la empresa Amazon proporciona en algunos casos un descuento aplicable a la compra de productos seleccionados mediante el servicio Alexa. Esto favorece la utilización de esta nueva tecnología y facilita la obtención del feedback necesario para iterar en el desarrollo de la misma.

Habiendo mostrado previamente los ejemplos mediante los diagramas de estado, y descrito el funcionamiento de la interfaz de la tienda de Amazon, podemos analizar las distintas alternativas y verificar que la realización de una interfaz auditiva para una aplicación de usuario puede ser altamente compleja de diseñar e interpretar. Hemos representado los ejemplos comenzando por el caso más simple y finalizando con el caso más complejo. Las principales diferencias que reconocemos entre una implementación de interfaz de usuario auditiva con respecto a una gráfica, tienen que ver con la existencia de una mayor cantidad de pasos necesarios para completar acciones y la limitación del tiempo presente en la interacción entre el usuario y la aplicación. Además, podemos observar que la interfaz auditiva de la tienda de Amazon no ofrece todas las funcionalidades que sí es posible utilizar en la interfaz web. Esto nos hace pensar, al menos por ahora, que la inserción de funcionalidades y opciones para el usuario puede ser mucho más costosa que en una interfaz gráfica. No obstante, hay que destacar que ya existen desarrollos para el dispositivo Amazon Echo que se utilizan diariamente para distintas tareas, demostrando que este tipo de aplicaciones son efectivamente reconocidas por los usuarios como útiles, intuitivas y de fácil adaptación.

3.4. Administrando contenido Web existente y de terceros

La idea de extracción de información que contemplamos en nuestro enfoque posee similitudes con algunas técnicas utilizadas en Web Scraping [18]. Se conoce como Web scraping al proceso llevado a cabo para la extracción de

datos no-estructurados (o débilmente estructurados), usualmente simulando la actividad de un navegador Web. Es utilizado comúnmente para automatizar la extracción de datos, con el fin de obtener más información a través de un procesamiento previo.

Una técnica de extracción de información orientada a los usuarios finales común es la anotación de contenidos Web. Algunos sitios Web ya se han encargado de etiquetar sus contenidos, permitiendo a otros artefactos de software (por ejemplo, una extensión de un navegador Web) procesar esas anotaciones y "aumentar" la interacción con estos contenidos estructurados. Un enfoque conocido destinado a darle significado a los datos de la Web son los Microformatos [31]. Algunos enfoques aprovechan el significado subyacente otorgado por los microformatos, identificando esos objetos presentes en las páginas Web y permitiendo a los usuarios interactuar con ellos de nuevas maneras. De acuerdo con [32], solo un 5,64 % entre más de 40 millones de sitios Web proveen algún tipo de datos estructurados (Microformatos, Microdatos, RDFa, etc.). Ésta realidad reafirma la importancia de empoderar a los usuarios para que puedan agregar estructuras semánticas cuando no estén disponibles.

Existen varios enfoques que permiten a los usuarios agregar estructura a contenidos existentes para facilitar el manejo de objetos de información relevante. Por ejemplo, Atomate it! [33] brinda una plataforma que permite configurar una colección de objetos por medio de la definición de reglas. Luego el usuario podría ser informado cuando algo interesante (como puede ser una nueva película, o registro) sea agregado, editado o removido.

Algunos enfoques de desarrollo para los usuarios finales se plantearon darle más poder a los usuarios, para que puedan cumplir con sus necesidades particulares por su cuenta. Por ejemplo, MashMaker [34] permite extraer widgets junto con sus propiedades, para luego poder insertarlos en otras páginas Web con el fin de modificar la aplicación. Otro trabajo propone la estructuración y extracción de modelos de datos del lado del cliente, para crear sitios Web personales que se ejecuten plenamente en el lado del cliente, por ejemplo los navegadores Web para los usuarios finales [9]. SearchAPI permite a los usuarios finales que no posean habilidades de programación, crear APIs de búsqueda mediante la selección visual de partes de la interfaz de usuario de los motores de búsqueda de las aplicaciones Web. De este modo, se pueden buscar los objetos de dominio que una aplicación ofrece simulando la interacción del usuario [18]. Han surgido enfoques similares en base a una técnica que es conocida como aumentación de la Web (Web augmentation), siendo una tecnología todavía prometedora para el desarrollo destinado a los usuarios finales [35].

Sin embargo, hasta donde llega nuestro conocimiento, no existen enfoques de desarrollo orientado a usuarios finales que permitan el desarrollo completo de especificaciones VUI reutilizando contenido Web existente.

Capítulo 4

Especificación de VUIs por los usuarios finales y basada en contenidos Web

En esta sección, presentamos nuestro enfoque para definir VUIs a través del parseo de contenidos Web. Primero presentaremos nuestro enfoque de un vistazo, y luego en la sección **4.2**, explicaremos en detalle los diferentes aspectos relacionados. Es válido indicar que, si bien hemos hecho referencia en nuestro trabajo a todo tipo de información presente en la web (por ej. información de productos de e-commerce o reseñas), creemos que el enfoque puede ser explicado de forma más sencilla en ámbitos de contenidos de información del estilo de noticias y notas edificadas a partir de textos definidos con una estructura más común (título, introducción, párrafo), de gran presencia en la web.

4.1. El enfoque en pocas palabras

La base de nuestro enfoque de desarrollo de VUIs orientado a usuarios finales se divide en tres partes:

1. Un mecanismo que permite a los usuarios seleccionar y definir bloques de contenido Web. Para esta tarea, utilizamos definiciones y anotaciones de contenidos Web, por medio del uso de herramientas visuales y una simple configuración. En el resto del documento, los llamaremos **bloques de contenido**.

2. Una manera de especificar cómo deberían ser usados estos bloques de contenido dentro de una VUI, y cómo debería comportarse dicha VUI. Creemos que los diagramas de flujo permiten modelar adecuadamente una estructura de comportamiento VUI; en los mismos, cada nodo representará un bloque de contenido específico, y las conexiones entre ellos, harán referencia a cómo se deberán leer y organizar los contenidos que componen cada grupo de contenidos.

3. Un intérprete, que se encargue de procesar una especificación VUI, y obtenga dinámicamente los elementos del DOM (modelo de objetos del documento) de los sitios Web, para luego proveerlos al skill y finalmente retornar la respuesta al usuario con el texto del contenido extraído.

Nuestra idea principal radica en que los usuarios finales puedan diseñar diagramas de flujo utilizando los bloques de contenido que crearon previamente. El proceso comienza con la definición de un bloque de contenido, que podrá ser usado luego en el editor de grupos para crear la VUI correspondiente, y finalmente utilizar esas especificaciones desde una aplicación nativa en un dispositivo inteligente que utilice una interfaz por voz. La Figura 8 representa una visión general de nuestra idea, en donde una especificación VUI (basada en un grupo de contenidos) contiene el bloque de contenido A (perteneciente a la página Web A), y otros dos bloques de contenido que pertenecen a la página Web B, uno de ellos nombrado como B (correspondiente a una colección de contenidos Web relacionados) y otro individual nombrado como C. Suponiendo el caso en el que un usuario quiere crear una VUI para noticias pertenecientes al sitio Web 'The New York times', en el que el mismo quiere incluir algunos elementos desde la portada del sitio (Bloques 1,2,3,4 y 5 de la Figura 9a). Estos bloques de contenido serían extraídos individualmente, como sucede con los elementos A y C en el ejemplo genérico de la Figura 8. No obstante, el usuario también querrá obtener todas las noticias pertenecientes a una sección específica de 'New York Times' (como se muestra en la Figura 9b con los elementos resaltados); nuestro enfoque permitirá entonces definir un conjunto de elementos 'hermanos' que componen un único bloque de contenidos.

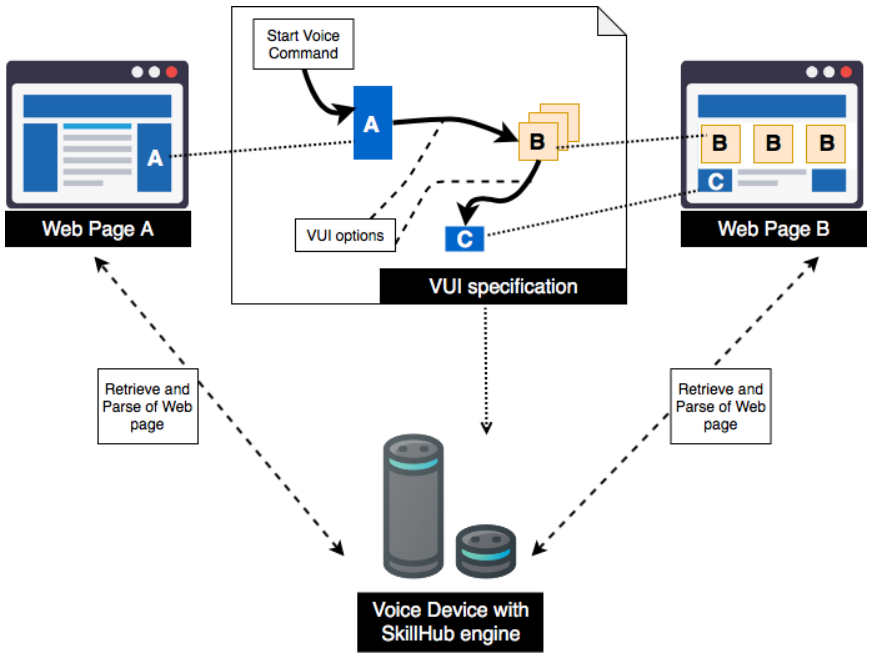


Figura 8: Nuestro intérprete VUI procesará especificaciones VUI, que son grupos de contenidos que definen cómo se van a leer las "partes" de las páginas Web en una interacción por voz

4.2. Base lógica: yendo de interfaces de usuario web a interfaces auditivas

En esta sección presentamos las cuatro dimensiones que definen nuestro entorno EUD (end user development) para la creación de VUIs.

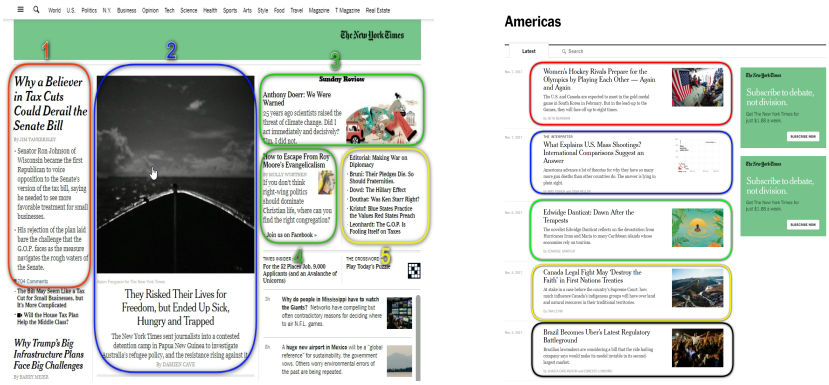
Definición de bloques de contenido

Pensamos en dos métodos para definir los bloques de contenido. Uno de ellos se basa en seleccionar individualmente cada "parte" de un sitio web que podría resultar útil, como se muestra en la Figura 9a. En este caso, el usuario deberá seleccionar un elemento de la interfaz de usuario (un elemento del DOM) para crear su bloque de contenido correspondiente. La otra forma consiste en contemplar un conjunto de elementos de la interfaz de usuario (que posean algún tipo de estructura en común) como si fuesen un único bloque de contenidos, lo cual se encuentra representado en la Figura 9b.

Comúnmente, las aplicaciones Web manifiestan en su interfaz de usuario una representación de sus objetos de dominio, como pueden ser noticias, productos, artículos, etc. Esto significa que en el "lado del cliente" de la Web, un usuario podría simular un modelo de dominio simple basado en los atributos presentes en la interfaz de usuario. Por ejemplo, para el caso de las noticias presentes en la Figura 9a, los atributos título, resúmen, fecha y autor podrían obtenerse desde la misma interfaz de usuario. Lo mismo pasará si buscamos productos en el sitio Amazon, cuya UI (interfaz de usuario) presentará atributos como el nombre del producto, precio, descripción, etc. El proceso de anotación necesario para definir un bloque de contenido podría tener en cuenta esta especificación semántica, o puede ser más directo y simple, considerando todo el elemento del DOM como bloque de contenido. En este último caso, al parsear (analizar) el elemento del DOM, será posible descomponerlo y detectar automáticamente distintas partes que sean relevantes para la VUI (links, texto, etc).

Otro aspecto importante a tener en cuenta será determinar si un bloque de contenido es 'navegable' o no. Esta característica aparece comúnmente en los sitios Web, los cuales presentan extractos de información de un ítem determinado, y mediante la acción de clickear en un link, ofrecen la navegación hacia una página Web específica correspondiente a ese ítem; este es el caso de las noticias presentes en la Figura 9a. Esta opción de navegación, que permite obtener más información sobre un bloque de contenido, será también considerada en nuestro enfoque.

Finalmente, los bloques de contenido deberán categorizarse con el fin de permitir una mayor flexibilidad al momento de definir el comportamiento de las VUIs. De esta manera, se podrían definir comandos de voz como 'pedir por las principales noticias', 'pedir por la información del tiempo', etc.



(a) Diferentes partes de una misma página Web

(b) Elementos hermanos de una misma página Web

Figura 9: Bloques de contenido Web: (a) selección de contenidos individuales (b) selección de contenidos hermanos.

Se podría diseñar un ejemplo de skill para utilizar los bloques de contenido de la Figura 9b, la cual se basa en un conjuntos de elementos hermanos que comparten un mismo tema. La skill podría leer al usuario los 'Temas de noticias' cuando pronuncia estas palabras como un comando de voz. Ésta podría responder y leer el título de la primera noticia, para luego preguntar al usuario final si desea escuchar más acerca de la noticia en cuestión o continuar con la siguiente noticia (este comportamiento será especificado por el usuario al momento de definir la VUI):

- Comando de voz del usuario: "Temas de noticias"
- Amazon Echo: "Women's hockey rivals prepare for the olympics by playing each other again and again. **Quieres seguir escuchando más acerca de esta noticia?**" (esta respuesta comienza leyendo la primer noticia del tema, y luego pregunta al usuario si desea seguir escuchando).
- Comando de voz del usuario: "Si".
- Amazon Echo: "BOSTON — Three days after the United States women's hockey team lost to Canada, 5-1, in an exhibition game here on Oct. 25, USA Hockey unexpectedly added Cayla Barnes, an 18-year-old freshman at Boston College, to its roster. **Quieres seguir escuchando más acerca de esta noticia?**"

Acceso a los bloques de contenido

La Web se basa en el concepto de navegación entre recursos que son accesibles por medio de una localización de recursos universal (URL). Teniendo en cuenta esto, se puede recuperar el contenido de un sitio Web específico si se conoce la dirección URL asociada. Cuando nos enfrentamos a una URL dinámica, que va cambiando en base al contenido publicado o la sesión del usuario, la navegación será una estrategia que permitirá obtener el contenido de una Web, a partir de la sección 'home' de la misma. Sin embargo, cuando existen grandes conjuntos de datos dentro de un mismo sitio Web, los motores de búsqueda se convierten en una herramienta fundamental a la hora de obtener elementos de información relevantes. Con esto en mente, nuestro enfoque va a contemplar tres formas distintas de acceder al contenido Web:

- Acceso Directo: es posible retornar una página Web a partir de una URL (que puede ser estática o basada en una URL obtenida por una API, que permita cambiar algunos valores de parámetros). Este método es útil para obtener el estado actual de un sitio Web que ofrece información que es actualizada frecuentemente, como pueden ser sitios de noticias, clima, etc. Los contenidos de los casos que aparecen en los ejemplos (a) y (b) de la Figura 2 pueden ser accedidos de esta forma.

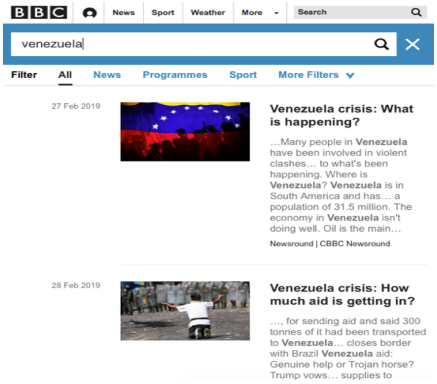
- Navegación: cuando no se puede acceder a un contenido deseado por medio de una URL predefinida, es posible navegar a través de enlaces que son parte de la página Web retornada. La navegación también es importante para obtener más información acerca de un bloque de contenido. Por ejemplo, leer en detalle una noticia principal podría implicar seguir un enlace, que permita a los usuarios finales navegar desde la portada de un sitio Web a la página Web específica de la noticia en cuestión, cuya URL difícilmente se podría conocer de antemano. Suponiendo que un usuario desea conocer más acerca de la noticia correspondiente al elemento 3 de la Figura 9b, se debería retornar una página Web similar a la que se encuentra representada por la Figura 10a, donde está presente el resto de la noticia en cuestión. En casos como este, se utiliza la navegación para obtener el bloque de contenido deseado.

- Búsqueda: en casos donde la VUI requiera consultar una aplicación Web para obtener información específica, se podrían utilizar motores de búsqueda específicos para automatizar el proceso [36]. Por ejemplo,

si sólo se requieren elementos relacionados con un dominio específico (por ejemplo, noticias relacionadas con 'Venezuela', como se muestra en la Figura 10b), puede ser útil simular cómo el usuario debería buscarlo dentro de la aplicación Web. Este método de acceso a contenido Web es importante también en los casos de sitios e-commerce, sitios de reservas de vuelos y alojamiento, etc.

Sin importar la técnica que se utilice para retornar la página Web que contenga el contenido deseado, una vez que se obtiene la misma, será necesario poder extraer cada bloque de contenido y parsearlo. Para este propósito, cada bloque de contenido tendrá un template de extracción, el cuál será definido por los usuarios finales por medio de una serie de herramientas de selección y anotación. Estas anotaciones forman parte de la definición de una VUI y contienen (entre otros aspectos) las expresiones XPath que permitirán extraer un elemento de información específico de una página Web.

(a) Página Web de noticias



(b) Resultados de búsqueda en un portal de noticias

Figura 10: Bloques de contenido Web: (a) detalles de items (b) resultados de búsqueda de items.

Orden en el que un mismo comando de voz leerá varios bloques de contenido

Como ya mencionamos, este enfoque propone el uso de diagramas de flujo para organizar cómo estarán dispuestos los bloques de contenido en la VUI, debido a que como respuesta a un comando de voz específico, se leerá una secuencia de bloques de contenido. Una vez definidos los bloques de

contenido, es importante también definir el orden en el que se van a leer los mismos y bajo qué interacciones de voz. Por ejemplo, para los bloques de contenido que aparecen en la Figura 9a, el comando de voz podría ser 'Leer las noticias de hoy', y el orden en el cual se van a leer las noticias, podría ser (Bloque 1, Bloque 2, Bloque 3, Bloque 4, Bloque 5), en el que cada número de bloque se corresponde con los números que figuran en la Figura 9a, o también podría ser cualquier otro orden que el usuario final defina.

4.3. Configuración del comportamiento de las VUI

Como ya mencionamos anteriormente, en nuestro enfoque se definirá la estructura principal de las respuestas VUI por medio de un diagrama de flujo. Además del orden establecido, se deberán definir otros aspectos del funcionamiento de una VUI:

1. **Cómo leer un bloque de contenido:** cuando un usuario dice un comando de voz, la VUI responderá con uno o un conjunto de bloques de contenido. Sin embargo, podrían ser distintas las partes o propiedades que se usen en la lectura de esos bloques de contenido para distintos escenarios de uso. Por ejemplo, un usuario podría estar interesado en leer sólo el título principal o el bloque de contenido completo para algunos contenidos, o de forma general, un conjunto específico de las propiedades semánticas definidas para ese bloque de contenido. Además, en el caso de que el bloque contenga un elemento 'navegable', entonces se podría brindar la posibilidad de obtener más información en profundidad, pudiendo extraerla una vez obtenida la página Web definida en el enlace del bloque de contenido, etc.

2. **Cómo continuar con el contenido de bloque siguiente:** más allá de cómo se lea un bloque de contenido específico, cuando la lectura de este finalice, habrá diferentes posibilidades de continuar con otros bloques relacionados. Esto es parte de la definición del funcionamiento de la VUI, por el cual el usuario final deberá poder definir entre distintas opciones: leer el bloque siguiente sin preguntar antes, leer el bloque siguiente directamente pero pronunciando previamente un texto predefinido, preguntar al usuario si desea continuar con la lectura, etc.

Con estos dos aspectos en mente, buscamos brindar soporte a la variabilidad de comportamiento para una VUI propuesta. Sin embargo, dado que

sería muy tedioso y propenso a errores definir cada uno de estos aspectos para cada elemento presente en un grupo de contenidos de la VUI (teniendo en cuenta tanto nodos como enlaces), proponemos utilizar como opción por defecto algunos patrones de VUI. Un template de patrón definirá el comportamiento transversal para administrar los bloques de contenido (1) y la transición entre ellos (2).

De todas formas, para permitir que los usuarios finales puedan personalizar su VUI obteniendo una mayor variabilidad, brindaremos la posibilidad de modificar el comportamiento en base al patrón elegido, tanto para un bloque en particular como para la transición hacia otro bloque de contenido. Hasta el momento, en nuestro enfoque se requerirá poseer habilidades de programación en Javascript para poder definir un nuevo patrón VUI. Básicamente, estos patrones son máquinas de estado que definirán cómo será el comportamiento conversacional.

Capítulo 5

Herramientas

5.1. SkillMaker: Un entorno de especificación de VUI basado en el navegador Web

En esta sección, presentaremos **SkillMaker**, nuestro entorno EUD, a través de un ejemplo. Presentaremos la herramienta que permitirá definir templates de extracción para los bloques de contenido, de modo que luego puedan ser utilizados por el editor de VUI basado en diagramas de flujo. El entorno fue deployado en su totalidad como una extensión de navegador Web (en particular como una extensión del navegador Google Chrome).

5.1.1. Definición de bloques de contenido

Utilizamos la anotación de contenidos como método de definición de un bloque de contenido [37]. El proceso comienza cuando el usuario decide definir un bloque de contenido para el sitio Web actual, lo cual se realiza presionando en el botón principal de la extensión Web SkillMaker – punto (1) de la Figura 11 –. Como resultado, se van a resaltar los elementos del DOM cuando el puntero pase por sobre estos. Cuando el usuario escoja un elemento del DOM específico, podrá arrastrar y soltar el mismo – punto (2) en la Figura 11 – dentro de la caja de definición de templates de extracción – punto (3) en la Figura 11 –.

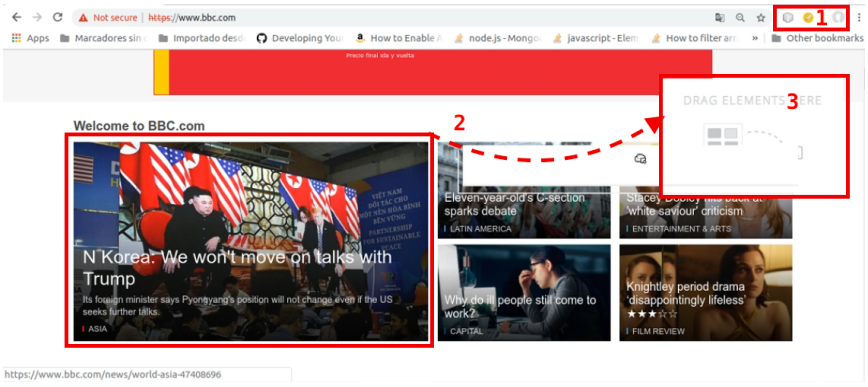


Figura 11: Seleccion de contenido Web para definir un bloque de contenido

Una vez seleccionado un elemento del DOM, comenzará el proceso de anotación agregando la semántica de los sub-elementos del DOM, como muestra la Figura 12. Se puede observar una vista detallada en la Figura 13a, donde aparece un cuadro de confirmación para la propiedad del título del contenido.

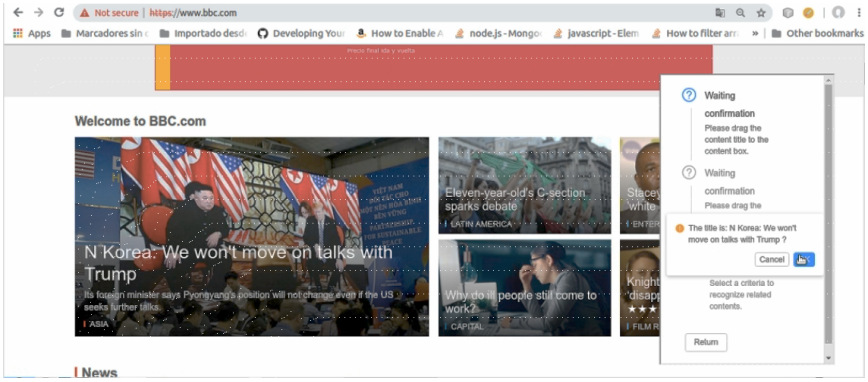
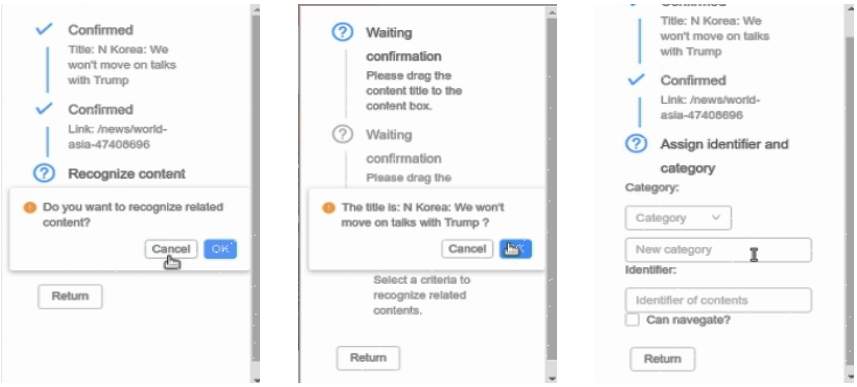


Figura 12: Definición de un bloque de contenido Web

El proceso continua (Figura 13b) preguntadole al usuario si se deben considerar contenidos relacionados (elementos hermanos) al bloque de su interés, para admitir bloques de contenido como los que aparecen en la Figura 9b. Se puede observar también en la Figura 13b que la herramienta detecta automáticamente el enlace de navegación del elemento del DOM

seleccionado. De esta manera, el usuario final podría tener en cuenta esta navegación (notar el checkbox “Can navigate?” de la Figura 13c). La Figura 13c muestra un formulario de edición, para completar otras propiedades obligatorias como son la categoría y el nombre.



(a) Edición de atributos semánticos (b) Edición de contenidos relacionados (c) Edición final del bloque

Figura 13: Vistas de las diferentes etapas de edición durante el proceso de definición de un bloque de contenido

Cuando el usuario final confirma la creación del bloque de contenido, la herramienta se encarga de almacenar todos los datos y ofrece dirigirse al editor de grupos de contenidos, que será explicado en el apartado siguiente. De todas formas, igualmente el usuario podrá continuar con la creación de bloques de contenido para la misma página Web o cualquier otra que desee.

5.1.2. Despliegue y definición de VUI a través de ejemplos

El editor de grupos, desplegado también dentro de la misma extensión Web, tendrá acceso a los bloques de contenido definidos por el usuario final, los cuales podrán ser arrastrados y soltados dentro del canvas del editor de diagramas, como se puede observar en la Figura 14. En este ejemplo, el bloque de contenido seleccionado es único pero representa todos los elementos hermanos de la Figura 9b.

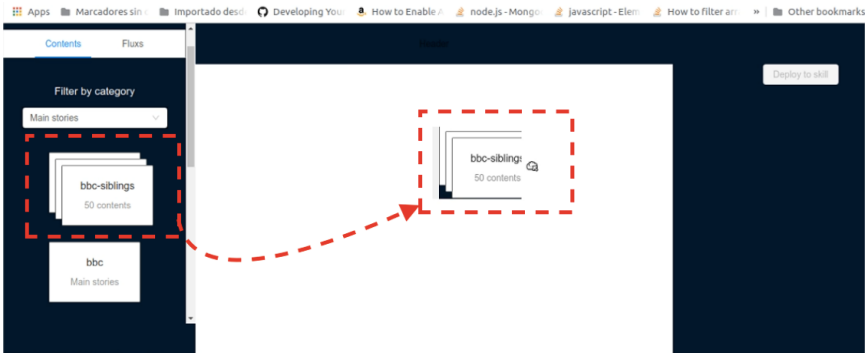


Figura 14: Definición de un grupo para leer un conjunto de bloques de contenido hermanos

Luego de agregar algunos bloques más al canvas, y de conectar los nodos a través de enlaces, se podrá ver un grupo de contenidos como el que se muestra en la Figura 15. En este caso, los bloques de contenido representan varias de las noticias principales seleccionadas de distintos portales.

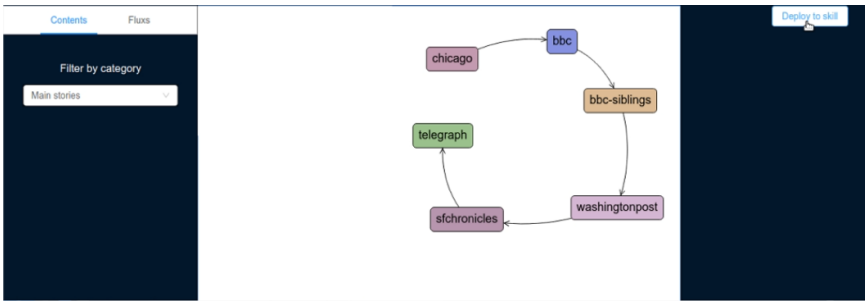


Figura 15: Grupo definido para leer las noticias principales de varios portales de noticias

Cuando el usuario presiona el botón "Deploy to skill", el editor abrirá una ventana modal (Figura 16) pidiendole al usuario:

1. Un nombre que identifique al grupo de contenidos, el cual será utilizado luego como comando de voz como respuesta a la petición de la skill.
2. Un patrón de lectura de contenidos que va a ser utilizado en las respuestas correspondientes. Esto último está relacionado con la discusión

de la configuración del comportamiento de las VUI introducido en la sección previa.

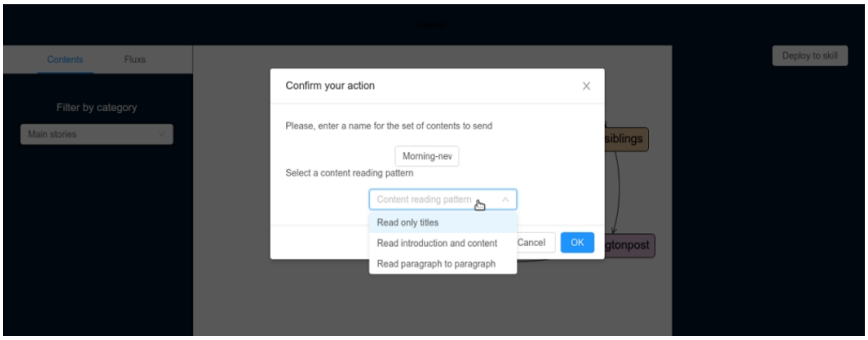
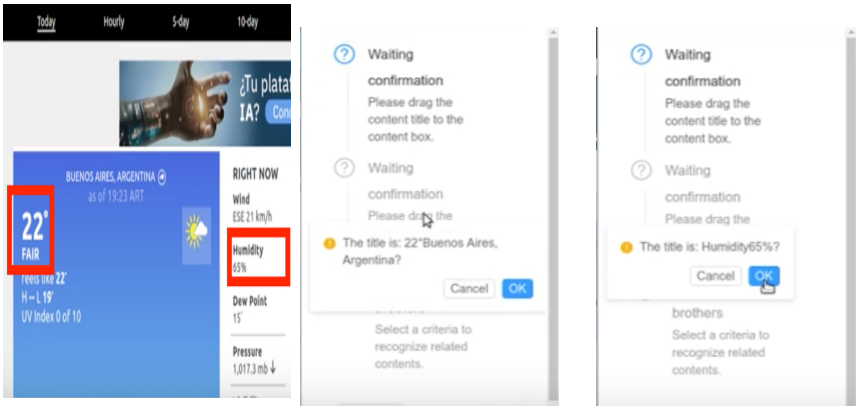


Figura 16: Edición del nombre del grupo y patrón de lectura de contenidos

Las skills diseñadas en el grupo de la Figura 15 utilizarán el patrón para "Leer únicamente los títulos". En este ejemplo, los 6 bloques de contenidos del grupo "main news" fueron definidos para 6 portales de noticias (Chicago Tribune, BBC, Washington Post, Telegraph, Sfchronicles). Un ejemplo de interacción con esta VUI podría ser el siguiente:

- User's Command voice: "Main News headlines"
- Amazon echo: "N Korea We won't move on talks with Trump, next news..."

Un segundo caso de estudio se basó en skills existentes que son usadas frecuentemente, como las que se relacionan con el clima. Esta skill se definió para poder responder con la temperatura y la humedad de la ciudad de Buenos Aires, cuando el usuario diga "Weather in Buenos Aires".



(a) Contenidos Web destacados dentro de una página Web

(b) Bloque para la temperatura

(c) Bloque para la humedad

Figura 17: Definición de Bloques de contenido para el grupo del clima

El grupo fue creado utilizando información del sitio "weather.com". La Figura 17 muestra el proceso de definición de los bloques. Se definieron dos bloques de contenido, uno para la temperatura y otro para la humedad. En la Figura 18 se puede observar el grupo definido correspondiente a estos bloques.

Dado que se pudo detectar que el contenido de interés involucra sólo títulos, utilizamos el patrón "Read only titles". Un posible extracto de conversación es el siguiente:

- User's Command voice: "Buenos Aires weather"
- Amazon Echo: "22, humidity 65 %"

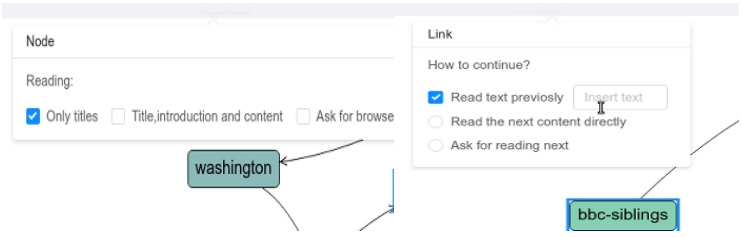
Figura 18: Definición de VUI para un grupo de contenidos del clima

5.1.3. Aspectos relacionados con el desarrollo orientado a usuarios finales

Anteriormente mencionamos que la variabilidad es un aspecto clave en entornos de EUD; por lo tanto, las reglas generales establecidas para leer bloques de contenidos y los enlaces utilizados en el patrón podrían ser reemplazados, editando manualmente la forma de leer cada uno de los elementos. Para el caso de los nodos que representan los bloques de contenido, en la Figura 19a se observan las opciones disponibles, donde se podrá seleccionar entre "Leer todo" o "Leer sólo los títulos"; también se podrá habilitar la opción para que la VUI pregunte al usuario antes de leer el resto del contenido de un bloque.

En la Figura 19b se puede observar cómo es la configuración para los enlaces. En este caso, las opciones disponibles son para poder leer un texto particular antes de comenzar con la lectura del bloque de contenido, y para elegir si el skill deberá preguntar o no antes de pasar a leer el siguiente bloque de contenido dentro del grupo.

Es importante mencionar que estas opciones provienen del análisis de varios ejemplos que fueron útiles para definir la expresividad del enfoque (no incluímos otros detalles). Sin embargo, fácilmente se podrían programar nuevos tipos de controles.



(a) Edición de opciones de la VUI para un nodo

(b) Edición de opciones de la VUI para un link

Figura 19: Edición de opciones para la VUI en reemplazo de las reglas de patrones

Nuestro entorno tiene en cuenta especialmente el asunto de debugging descrito por Ko. [38] en el contexto de enfoques de ingeniería de software para usuarios finales. Esto es dado que nuestro enfoque se basa en contenidos Web de terceros. El template de extracción definido en nuestro entorno posee referencias a elementos del DOM expresadas en xPath. Si una página Web

cambia su estructura del DOM subyacente, estas expresiones xPaths podrían dejar de funcionar. Aunque existen otras maneras más robustas [39], aún existiría la posibilidad de que un cambio sustancial en el DOM de la página Web destruya o inhabilite las referencias. En este sentido, el entorno provee de un feedback visual en el editor de grupos de contenidos, cuando un bloque de contenido particular parece tener una referencia rota, como muestra la Figura 20 (estos bloques aparecerán en el menú con un fondo de color rojo). Haciendo click en uno de estos bloques, comenzará un procedimiento de redefinición de un template de extracción en la página Web correspondiente. Si no es posible encontrar la página Web correspondiente, el usuario podrá definir un nuevo bloque de contenido en cualquier otro sitio Web y guardarlo con el mismo nombre de bloque, o uno distinto si lo desea.

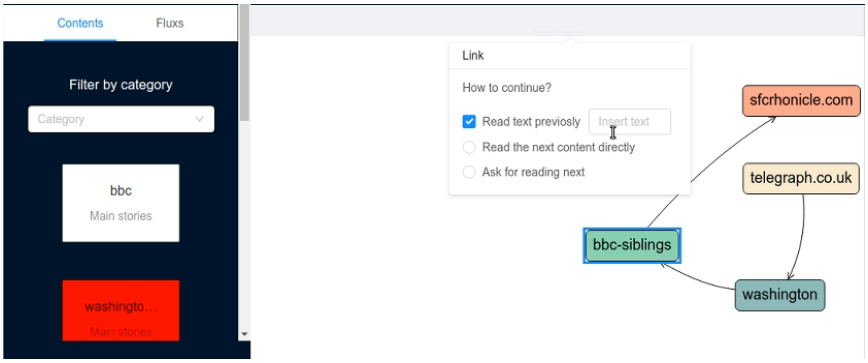


Figura 20: Bloque de contenido con referencias del DOM que dejaron de funcionar

5.2. Implementación

La implementación de la herramienta que utilizamos para llevar a cabo la propuesta, se ha realizado teniendo en cuenta no solamente el factor tiempo, sino que también se han seleccionado tecnologías que favorecen conceptos importantes como la mantenibilidad del código, extensión y encapsulamiento. Su correcta implementación y la utilización de buenas prácticas de programación son de suma importancia. Su desarrollo ha sido planificado con antelación, teniendo en cuenta los lineamientos propios del dominio y la interacción con el ambiente del sistema propuesto.

La herramienta en cuestión está implementada en base a microservicios, por lo cual contamos con un ecosistema de aplicaciones desacopladas que pueden ser extendidas de manera independiente. Además, se han desarrollado con tecnologías actuales que impulsan la extensibilidad a través de nuevos desarrolladores que puedan incluirse eventualmente. Los servicios están separados en base al stack al que pertenecen. Es decir, contamos con servicios orientados al manejo de información necesaria para el desarrollo (backend) y, por otra parte, disponemos de una serie de aplicaciones que se implementaron de manera coordinada para satisfacer los requerimientos funcionales del usuario (frontend).

Entre las aplicaciones frontend, desarrollamos una extensión del navegador Chrome que logra vincular al usuario mientras navega por la web con las funcionalidades desarrolladas para la abstracción del contenido. Esta extensión de Chrome es la primer herramienta con la que el usuario interactúa.

La extensión provee un ambiente de desarrollo aislado, que logra comunicar a otras aplicaciones con el DOM de los sitios accedidos por el usuario. El usuario es capaz de arrastrar, con el mouse, contenidos desde cualquier sitio hacia la extensión. La misma transmite la información del evento, que incluye información del contenido arrastrado, hacia la segunda aplicación dentro del flujo del proceso de abstracción de contenidos. Una extensión Chrome está compuesta comúnmente por una serie de archivos destinados a trabajar dentro de ambientes diferentes para cumplir roles distintos. Los archivos básicos presentes en las extensiones Chrome son:

- Manifest: provee al navegador de información sobre la extensión. Por ejemplo, indica los archivos y funcionalidades nativas que utilizará.
- Background script: es donde se almacena el código propio de la extensión capaz de escuchar y disparar eventos desde y hacia el browser.
- Content script: Contiene el código Javascript que se ejecuta en el contexto de un sitio web del browser. El código presente en este archivo puede leer y modificar el DOM de un sitio web.

Las extensiones de Chrome son capaces de desenvolverse dentro de parámetros muy similares a los de un sitio web común y corriente. Esto es, pueden renderizar contenido HTML y ejecutar de forma nativa código escrito en Javascript. Esto puede realizarse a partir de un contenedor propio de la extensión (popup) o con otras alternativas. En nuestro caso, optamos por generar, a partir de la extensión, un iFrame dentro del sitio web visitado

por el usuario. Los iFrames son elementos pertenecientes al DOM de un documento HTML, dentro de los cuales puede renderizarse paralelamente otro documento HTML. Su uso es una práctica bastante común dentro del desarrollo web desde hace varios años. El mismo estará parcialmente acoplado al resto del contenido de interés, facilitando la vinculación de información y la capacidad de comunicarse entre sí.

El flujo lógico dentro de nuestra extensión, a grandes rasgos, puede definirse de la siguiente manera: una vez que el usuario abre la extensión por medio del ícono de la misma dentro de la barra de tareas, se utilizan las funcionalidades que brinda el archivo content script para crear un iFrame acoplado al sitio web visitado. Este iFrame renderizará mayoritariamente un documento HTML desarrollado de manera independiente, al cual denominaremos **Content Parser**. El usuario podrá interactuar con este documento libremente y seleccionará las acciones a realizar. A partir de esta interacción necesaria, la extensión eventualmente reaccionará a un evento de mensajería que le indique que va a recibir información desde el DOM. A partir de este momento, la extensión (gracias al archivo background) "escuchará" a eventos de tipo "Drop" que puedan llegarle. En este instante, la aplicación *Content Parser* (visualizada a través del iFrame) mostrará al usuario un esquema que representa los pasos necesarios para la integración del contenido que desee, brindandole también un feedback al usuario para ubicarlo en el contexto actual de la aplicación: se indicará que debe arrastrar el contenido deseado.

Cuando el usuario arrastre un contenido hacia la extensión, la misma entregará la información referente al contenido hacia la aplicación renderizada dentro del iFrame (a través del servicio de mensajería, implementado en Javascript mediante la función "PostMessage").

Una vez entregada la información referida al contenido escogido por el usuario, *Content Parser* nos solicitará que confirmemos si la información del contenido es la adecuada. Una vez confirmado, podremos escoger entre las características propias del contenido (como la categoría que le asignemos o el nombre con el cual identificaremos al mismo), como así también decidiremos si deseamos obtener más información sólo del contenido arrastrado o si, en cambio, queremos obtener ademas aquellos considerados como "hermanos" (que posean características similares) si existiesen. Por características similares nos referimos al formato de los contenidos o su distribución dentro del sitio.

Luego de que todos los campos necesarios fueran completados, podremos confirmar el/los contenidos que finalmente podremos consumir en una VUI. El usuario podrá repetir todo el proceso de selección de contenido las veces

que desee y con los contenidos elegidos. Además, será posible "volver atrás" y repetir el proceso sin refrescar la página, si así lo quisiera.

A partir de este momento, podemos garantizar que logramos completar el proceso de abstracción de un contenido web hacia un tipo de dato complejo, que puede ser interpretado y utilizado luego para obtener la información que resulta realmente importante para el usuario final. Esta meta-información obtenida, permite que los contenidos web en cuestión, puedan ser procesados, abstraídos y finalmente reproducidos a través de cualquier medio entendible para el ser humano. En nuestro caso, a partir de la voz.

5.2.1. Detalles de implementación de Content Parser

La aplicación en cuestión fue desarrollada como una aplicación web mediante HTML + CSS + Javascript, utilizando la librería ReactJs mayoritariamente para lograr una implementación basada en componentes. De esta forma, pudimos identificar cuáles son los principales sujetos dentro de nuestra herramienta y representarlos de forma unívoca y estratégica en nuestra implementación. ReactJs utiliza una arquitectura "top-down", por lo que partimos de componentes más generales que engloban a otros componentes, cada vez más pequeños e identificables. Además, la lógica de la implementación también se ve afectada por esta arquitectura, ya que los parámetros para el renderizado y el manejo de eventos siguen este mismo criterio.

La estructura básica de la aplicación ContentParser es la siguiente:

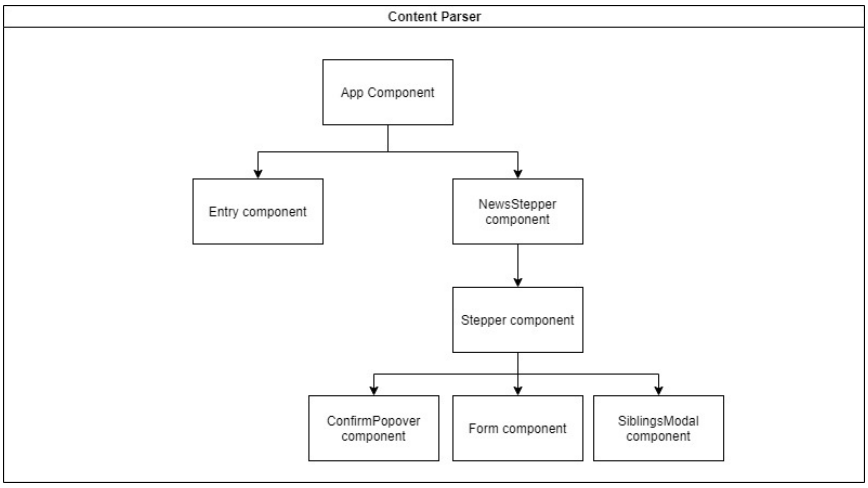
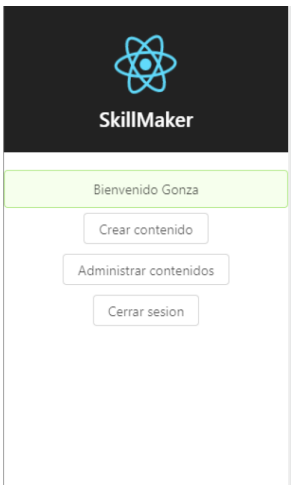


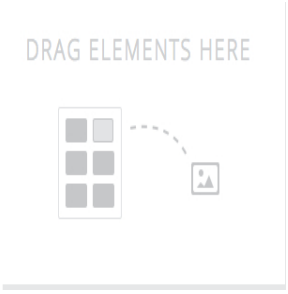
Figura 21: Content Parser

Centrándonos en la implementación del *Content Parser*, podemos identificar como primer elemento al componente **App**, que actúa como contenedor de toda la aplicación. En este caso, el componente en cuestión sólo es utilizado para contener la lógica y determinar qué información se deberá mostrar, dependiendo de la instancia en la que se encuentra el usuario. Esto es, si el usuario acaba de ingresar a la aplicación, se le mostrará el componente **Entry**. Este componente renderiza los botones adecuados para que el usuario comience a utilizar la herramienta.



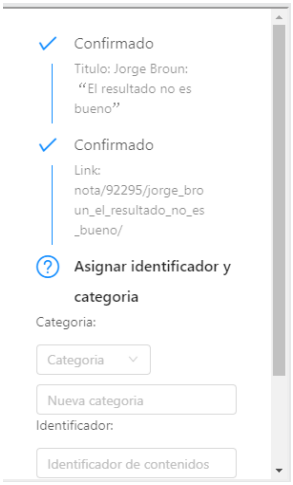
Si el usuario hace click en "Cerrar sesión", se le mostrará el login de la aplicación. Por otra parte, si el usuario ingresa a "Administrar contenidos" se le abrirá una nueva pestaña en el navegador, que abrirá la aplicación **Content Admin**, la cual será descripta en esta misma sección un poco más adelante.

Suponiendo que el usuario ingresa a "Crear contenido", la aplicación le mostrará una imagen en la que se indica que el contenido a crear debe ser arrastrado hasta la ventana de la extensión. En este punto, la aplicación comienza a interactuar con el navegador y el sitio web en el que el usuario está trabajando, mediante el envío de mensajes que disparan eventos.



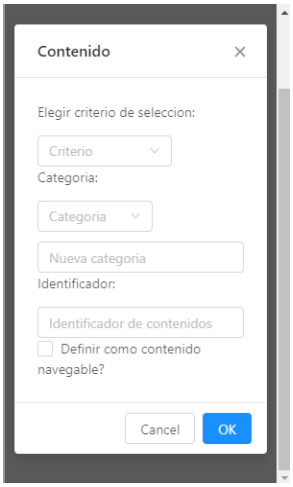
Una vez hecho esto, dentro de la misma aplicación se mostrará un nuevo componente: **NewsStepper**, el cual contiene la lógica referida a los distintos pasos que el usuario deberá completar para crear un nuevo contenido administrable y reproducible por la skill de Alexa. Más allá de la lógica que concentra este componente, se encarga a su vez de renderizar otro componente menor (basándonos en la estructura jerárquica mencionada anteriormente): **Stepper**, cuya función consiste en guiar al usuario, mostrando el estado actual en el que se encuentra dentro del proceso de abstracción del contenido.

A su vez, también se invoca al componente **Form**, el cual se encarga de mostrar los distintos campos que el usuario debe rellenar para completar el proceso.

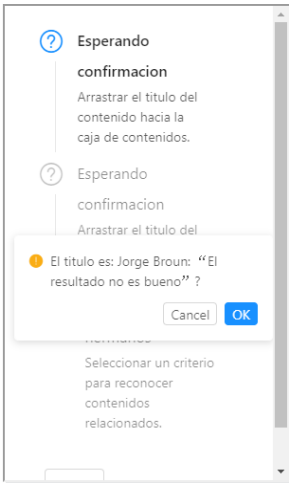


or otro lado, el mismo componente *Form* es utilizado en un nuevo contexto mediante la invocación de otro componente: **SiblingsModal**. Como

su nombre lo indica, representa al modal que se utilizará para la abstracción de los contenidos relacionados al contenido que se eligió crear en un primer momento ("contenidos hermanos"). Aquí evidenciamos cómo podemos reutilizar componentes dentro de la aplicación, ya que utilizamos el mismo componente *Form* en dos contextos diferentes.



También son invocados otros componentes "menores" que, mediante su renderización, ayudan al usuario a comprender el proceso que están llevando a cabo. Un ejemplo de esto es el componente **ConfirmPopover**. Este componente es el encargado de crear los distintos mensajes que se mostrarán al usuario, de manera que comprenda lo que va sucediendo a medida que va interactuando con la aplicación en pos del objetivo de crear su contenido. En el ejemplo de la imagen siguiente, podemos ver que se utiliza este componente para confirmar si el contenido en cuestión es el deseado por el usuario.



5.2.2. Detalles de implementación de Content Admin

Una vez completado el proceso de abstracción, toda esta información deberá ser administrada consistentemente y replicada de forma conjunta y estructurada en nuestros servicios back-end. Por este motivo, luego de confirmar el envío de la información, *Content Parser* nos permitirá acceder a la otra pierna de nuestra implementación: **Content Admin**.

Habiendo accedido a *Content Admin*, el cual también fue implementado de forma independiente para luego ser acoplado dentro de la misma extensión, podremos conformar estructuras de contenidos compuestos. Cada contenido abstraído previamente se podrá arrastrar con el mouse desde el panel izquierdo de la aplicación hacia el diagrama de contenidos. A su vez, tendremos la posibilidad de interactuar con cada uno de ellos y vincularlos entre sí, conformando una cadena lógica de contenidos.

El diagrama es fácilmente comprendido por el usuario, ya que no ofrece mayor complejidad que la expuesta en el presente párrafo. Una vez que tenemos nuestra estructura consistentemente armada (esto implica tener todos los contenidos enlazados, con un contenido inicial y un contenido final), podremos confirmar y enviar los contenidos en forma de unidad. Con esto, concluimos con el proceso de administración de contenidos, y a partir de este momento, podremos utilizarlos para reproducirlos a través de SkillHub dentro de Amazon Echo.

Con respecto a la implementación de la aplicación, al igual que *Content Parser*, está desarrollada con tecnologías HTML, CSS y Javascript, ya que también fue conformada como una aplicación web. Además, también se utili-

za ReactJs para la orquestación de los distintos componentes que se utilizan para la solución diseñada.

Dentro de los componentes más importantes, podemos observar al componente **Container**, cuya función consiste en actuar como puerta de entrada y contenedor del resto de los componentes. Por este motivo, aquí es donde se realiza el primer llamado al servidor para obtener los contenidos y los grupos para ser administrados.

Container renderiza a otro componente importante, llamado **Diagram**. Este componente utiliza la librería Javascript **goJs** para representar cada uno de los elementos dentro del diagrama. Sin este componente, no se podría representar ni utilizar los contenidos y grupos.

A su vez, el componente *Container* renderiza al componente **LeftPanel**, donde se muestran dos pestañas. La primer pestaña muestra todos los contenidos asociados al usuario logueado dentro de la aplicación, mientras que la segunda muestra los grupos de contenidos disponibles. Además, dentro de la pestaña de contenidos, se muestra un campo de búsqueda, donde el usuario podrá seleccionar una categoría en particular para filtrar los contenidos que dispone.

Asimismo, la aplicación fue implementada con clases que reflejan los conceptos mencionados previamente. El modelo de clases 22 que surje de la aplicación *ContentAdmin* refleja no solamente cómo está implementada la misma, sino que también da noción de cómo está pensada la solución integral del problema propuesto.

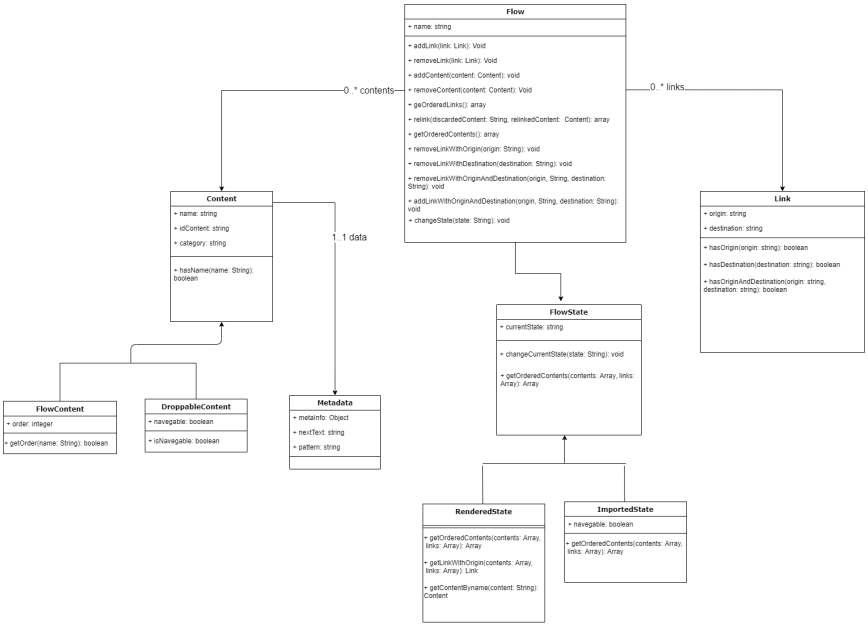


Figura 22: Modelo de clases

En el diagrama en cuestión, se evidencia que **Flow** (o grupo de contenidos) está compuesto por **Contents** y **Links**. *Flow* es capaz de manipular los mismos para realizar distintas tareas (agregar, eliminar o modificar tanto los contenidos como así también las conexiones y el orden de lectura entre ellos). Nuestra UI utiliza instancias de *Flow* para la solución de la implementación. *Flow* se encarga de obtener los contenidos ordenados en base a diferentes criterios (dependiendo de la ocasión), por ejemplo, al obtener los contenidos desde el backend o al utilizar el diagrama de la UI para editarlos. Por esto, utilizamos el patrón de diseño **"state"** dentro de nuestra implementación, para abstraer a la clase *Flow* del criterio correspondiente, y que sea el .estado.el que realice el ordenamiento en base al criterio adecuado y retorne los contenidos ordenados al propio *Flow*.

5.2.3. SkillHub

Como mencionamos al comienzo de este documento, creamos **SkillHub**, una skill para el servicio Alexa de Amazon Echo, que incluye nuestro intérprete Javascript para las VUIs definidas mediante **SkillMaker**. *SkillHub* y *SkillMaker* se encuentran sincronizados, de manera que cuando se

cree un nuevo grupo de contenidos, automáticamente esté disponible para ser consumido a través de Amazon Echo.

Las especificaciones VUI definidas mediante *SkillMaker* se almacenarán en un formato JSON y serán interpretadas en este formato por *SkillHub*. Ésta skill de Amazon Echo nos permitirá proveer VUIs creadas mediante *SkillMaker* en un escenario real.

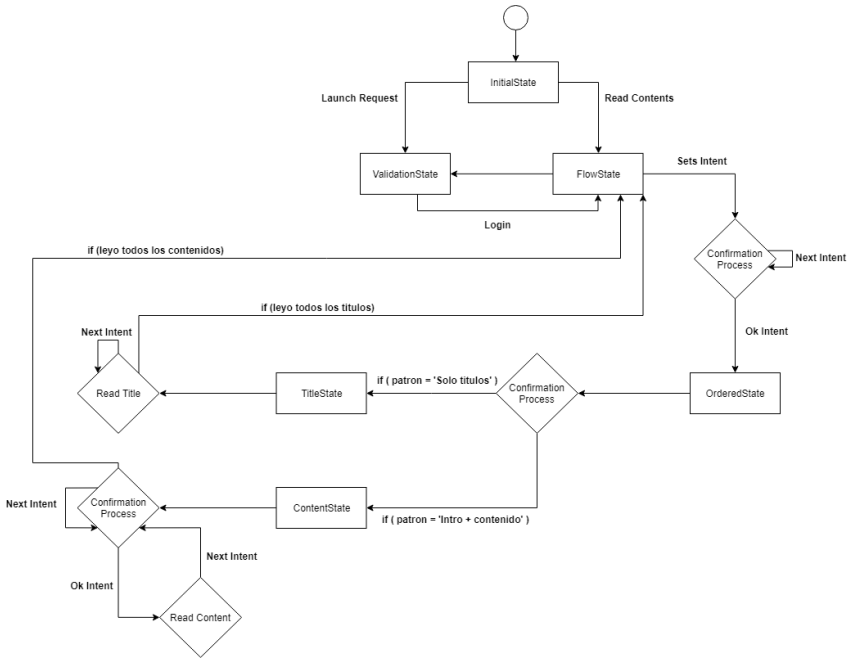


Figura 23: Diagrama de transición de estados de SkillHub

El diagrama en cuestión provee detalles propios de la implementación de la Skill desarrollada. Podemos observar que el mismo está basado en estados, los cuales definen comportamientos y respuestas diferentes ante los distintos comandos que el usuario pronuncia. Aunque estos comandos pueden repetirse, el comportamiento cambiará dependiendo del estado en el que se encuentre la aplicación.

A continuación, se detalla un ejemplo de interacción de un usuario con la Skill de Alexa, suponiendo que haya configurado un grupo de contenidos llamado “Noticias” con el patron de lectura para “Leer la introduccion y el resto del contenido”:

Usuario: Alexa, pide al lector de contenidos que lea mis contenidos

Amazon Echo: Cual es tu usuario?

Usuario: Gonza
Amazon Echo: Hola Gonza. Quieres escuchar tus grupos de contenidos o los contenidos filtrados por categorias?

Usuario: Mis grupos
Amazon Echo: Indica Ok para confirmar y siguiente para avanzar al siguiente grupo. Tus grupos son: 'Noticias'

Usuario: Ok
Amazon Echo: Has escogido el patron para 'Leer introduccion y contenido'. Quieres escuchar el listado de titulos?

Usuario: Si
Amazon Echo: Indica ok para confirmar y siguiente para pasar al siguiente titulo. //Se lee el titulo y se indica a que web pertenece

Usuario: Ok
Amazon Echo: Quieres escuchar una introducion antes?

Usuario: Si
Amazon Echo: //Se lee la introduccion. Quieres continuar con el resto del contenido?

Usuario: Si
Amazon Echo: //Se lee el resto del contenido

Usuario: Siguiente
Amazon Echo: Este es el ultimo contenido del grupo. Indica siguiente para volver a la lista de grupos. //Se lee el titulo y se indica a que web pertenece

Usuario: Siguiente
Amazon Echo: Quieres escuchar tus grupos de contenidos o los contenidos filtrados por categorias?

Usuario: Salir
Amazon Echo: Adios gonza. Nos vemos pronto

5.2.4. Microservicio para interacción con el DOM

Como comentamos anteriormente, para poder obtener el contenido concreto que se corresponde con los contenidos definidos por el usuario durante el proceso de abstracción, necesitamos acceder al DOM perteneciente a cada sitio web de interés para el usuario. Una vez que tenemos acceso al DOM de

una página web, necesitamos que a partir de una expresión xpath, se pueda obtener el texto asociado al elemento accedido mediante la misma.

Para manejar esta interacción con el DOM, fuimos haciendo uso de distintas librerías Javascript como son "xpath" junto con "xmlDom", "phantomjs", "osmosis", etc; pero a la hora de probar con cada una de estas alternativas, experimentamos distintos inconvenientes al momento de acceder al DOM en la mayoría de los sitios web probados. Por esto, luego de analizar las distintas opciones y tecnologías, nos terminamos decantando por implementar un servicio que haga uso de un **Headless-Browser** (navegador sin interfaz gráfica), el cual permite tener control de una página web en un entorno similar al de un navegador convencional, aunque ejecutándolo desde una consola de comandos.

Para esto, utilizamos **Puppeteer**, una librería del lenguaje NodeJs que provee una API de alto nivel para poder controlar los navegadores Chrome o Chromium a través del protocolo DevTools. *Puppeteer* se ejecuta en modo "headless" (se incorporan todas las características de la plataforma web proporcionadas por Chromium a la línea de comandos) por defecto, pero podría ser configurado para ejecutar Chrome o Chromium en modo full ("non-headless"). Con esta librería se pueden hacer la mayoría de las cosas que permite un navegador, como por ejemplo:

- Generar screenshots y archivos pdf de las páginas.
- Automatizar el envío de formularios, UI testing, entrada de teclado, etc.
- Testear extensiones de chrome.
- Correr tests directamente en la última versión de Chrome.

Así es que por medio de dos endpoints definidos, permitimos obtener texto dinámico con sólo disponer de una dirección URL y una expresión xpath: el primer endpoint ***getTitle*** se utilizará a la hora de obtener sólo el título de un contenido. Mientras que ***getBodyContent*** se usará para navegar hacia una nueva página a partir de un elemento "link" y así poder obtener, además del título, otras partes del contenido como son la introducción y el resto del texto en nuestro caso.

Con la creación de una instancia del browser, abriremos una nueva página a partir de una dirección URL (perteneciente a algún contenido abstraído por el usuario en *SkillMaker*) que será enviada mediante la invocación de cada endpoint, ya sea desde el servicio creado para la interacción con la BBDD

o desde la skill ejecutada dentro del contexto de una función "Lambda"(un servicio web provisto por Amazon que facilita el proceso de deploy de una skill).

Una vez cargado el documento de la página web, podremos obtener el título del contenido (en el caso de *getTitle*) utilizando la función ***evaluate*** provista por Javascript para evaluar expresiones xpath dentro del documento. A su vez, también tendremos la posibilidad de acceder a una nueva página web a partir de la dirección URL asociada a un elemento link, la cual contendrá toda la información relacionada con el contenido correspondiente a ese link (en el caso de *getBodyContent*).

En conclusión, gracias a esta librería podremos interactuar con el DOM de una página web de la misma forma que lo haríamos mediante la consola presente dentro del ambiente DevTools de un navegador, evaluando las expresiones xpath dentro del documento para así obtener la información que necesitamos.

5.2.5. Microservicio para la interacción con la base de datos

Desde el lado *backend* de nuestra herramienta, disponemos de un servicio Restful implementado con la tecnología NodeJs junto con el framework Express. Definimos distintos endpoints y dividimos las funcionalidades del servicio en dos partes: una que agrupa todos los endpoints que actúan como nexo, comunicando *SkillHub* con el servicio que permite la interacción con el DOM de los sitios web descrito en el apartado anterior; y una segunda parte, en la que definimos los endpoints encargados de manejar la interacción con la base de datos NoSql, implementada con tecnologías MongoDb con la ayuda del framework/ODM Mongoose.

MongoDB es un sistema de base de datos NoSQL orientado a documentos. En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos BSON (una especificación similar a JSON) que poseen un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida. MongoDB es una base de datos con múltiples funcionalidades entre las que destacamos:

- Capacidad de almacenar datos en documentos flexibles, con una estructura similar a JSON, lo que significa que los campos pueden variar de un documento a otro (en una misma colección, puede haber documentos con distinta cantidad de campos) y la estructura de datos puede ir cambiando a lo largo del tiempo.

- El modelo de documentos mapea con los objetos del codigo de la aplicación, haciendo que sea más fácil trabajar con los datos.
- Ejecución de JavaScript del lado del servidor: tiene la capacidad de realizar consultas utilizando JavaScript, haciendo que estas sean enviadas directamente a la base de datos para ser ejecutadas.

El servicio implementado permitirá interactuar tanto con la skill como con las herramientas frontend empaquetadas dentro de la extensión desarrollada. Entre otras cosas, permitirá obtener los nombres de los grupos y/o de las categorías definidos por el usuario para que sean listados dentro del entorno de la skill, obtener contenidos en orden filtrados por grupo/categoría, obtener toda la información perteneciente a un usuario, deshabilitar un contenido cuyo xpath haya dejado de funcionar, agregar contenidos durante el proceso de abstracción por parte del usuario, crear y actualizar grupos definidos durante el proceso de administración de contenidos.

5.2.6. Modelo de datos

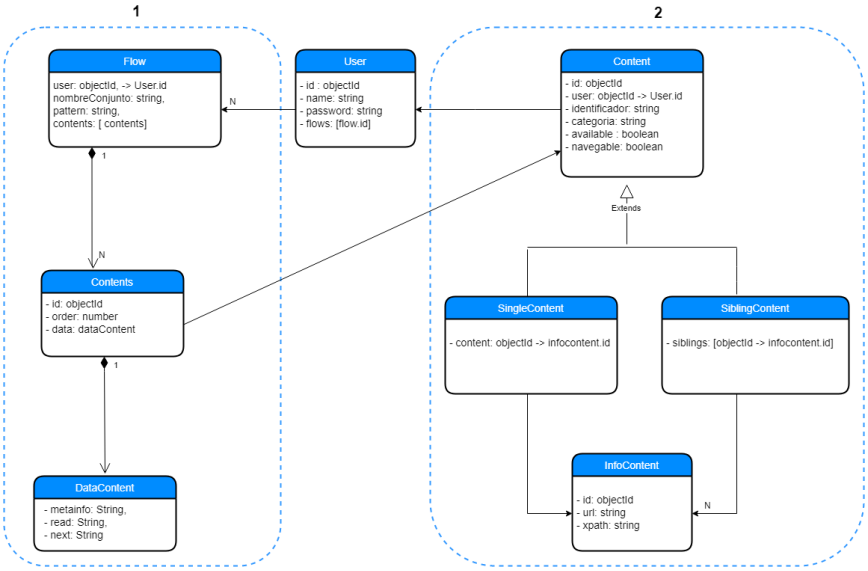


Figura 24: Modelo de datos

El modelo de datos está compuesto por distintos esquemas, a partir de los cuales se crean las colecciones de la base de datos:

- **User**: compuesto por los campos name, password, id y un arreglo de ids (flows) que referencia a los documentos presentes en la colección *Flow*.

- **Flow**: tiene asociado un id (user) que referencia a la colección User. Posee los campos nombreConjunto, pattern y un arreglo de objetos (contents) que posee información relacionada a los documentos de la colección *Contents* (id, order y un objeto "data" compuesto por los campos metainfo, pattern y next)

- **Content**: tiene asociado un id (user) que referencia a la colección *User*. Posee los campos identificador, categoría, available (para el caso de contenidos "deshabilitados" mencionado anteriormente), navegable, y además un discriminador que permite diferenciar entre contenidos simples (tendrán asociados un id de la colección InfoContent) y contenidos compuestos (dentro de un array tendrán asociados varios id de la colección InfoContent)

- **InfoContent**: contendra la informacion que extrajo el usuario en el proceso de abstracción de los contenidos, con los campos url, xpath y id.

Los esquemas agrupados dentro de **2** hacen referencia a la información que es generada por la aplicación *ContentParser*, mientras que los esquemas agrupados en **1** refieren a la información generada mediante la otra pierna de la extensión la cual denominamos *ContentAdmin*.

Capítulo 6

Pruebas de usuario

6.1. Evaluación de la herramienta

En esta sección, se realizará una evaluación del enfoque para verificar si es posible lograr la misma experiencia usando productos obtenidos a partir de nuestro enfoque respecto a productos nativos. En primer lugar, se definen los **objetivos**, **hipótesis** y **variables** del experimento. A continuación, se procede a definir los **materiales** considerados. Después de eso, se detallan **protocolos** utilizados. Luego se realizará un **análisis de resultados** y su **implicación**.

6.1.1. Objetivos, hipótesis y variables

En esta investigación, el objetivo es diseñar un enfoque que permita, a usuarios finales sin conocimientos de programación, especificar extensiones para dispositivos con interfaz de voz basadas en contenidos Web existentes.

La principal pregunta de investigación (**RQ**) es: ¿La experencia de los productos (skills) obtenidos con nuestro enfoque es similar a la de un producto (skill) nativo? Con el fin de responder a esta pregunta, se diseñó un experimento donde los sujetos fueron invitados a probar que es posible consumir contenido web a partir de la interaccion con interfaces auditivas satisfactoriamente. Se asignaron 2 casos distintos a los participantes, de forma aleatoria, en los que harían uso de un producto nativo ademas de un producto obtenido con nuestro enfoque, pudiendo ser un skill para obtener noticias u otro para obtener el clima. Para esta RQ, consideramos como hipótesis nula **H0** que la media del uso de nuestro enfoque es más compleja que la media del uso de un skill nativo. Como hipótesis alternativa **Ha**, se considera que la media del uso en ambos productos son similares.

En este experimento, se enfocó en medir el tiempo requerido para la tarea, satisfacción junto con la complejidad percibida, además de eficacia respecto si pudo completar la tarea o no.

6.1.2. Materiales

Para realizar las tareas del experimento, se preparó el ambiente para que no existan circunstancias externas que afecten el experimento (se llevó a cabo en un ambiente sin ruidos ni pantallas que distraigan al participante). Se instruyó a los usuarios con una serie de comandos que debian pronunciar para realizar las tareas de entrenamiento previas al experimento, utilizando skills nativas propias de la tecnología. Una vez completado el entrenamiento, el usuario estaba en condiciones de poder afrontar el experimento ya con alguna experiencia previa en la tecnología.

Para poder calcular la eficiencia de los participantes, una vez finalizadas las tareas del experimento, cada uno debió completar un cuestionario con preguntas acerca de su vivencia utilizando la herramienta y sus percepciones personales.

6.1.3. Protocolo

Durante el experimento, los participantes recibieron dos encuestas y un documento con las instrucciones necesarias para realizar las tareas.

Los participantes eran 20 personas sin distinción de edad ni sexo, pertenecientes a distintos rubros laborales. Un grupo de 4 participantes realizó el experimento con la skill creada por nuestro enfoque (tanto para la tarea de noticias como del clima). Otro grupo de 4 participantes realizó el experimento con skills nativas, mientras que un grupo restante de 12 participantes realizó el experimento intercalando tanto el uso de la skill creada con nuestro enfoque como la skill nativa. Ninguno de los participantes conocía previamente con qué tipo de skill estaba interactuando.

El protocolo del experimento se ejecutó de la misma manera con todos los participantes de ambos grupos y comprendió un flujo de trabajo bien definido. En primer lugar, se pidió a los participantes que completaran una encuesta demográfica, luego que leyeran la descripción del experimento, llevaran a cabo las tareas y finalmente completaran un cuestionario.

6.1.4. Análisis

Para lograr analizar los resultados de los participantes se volcaron todas las respuestas en un archivo excel. En el mismo figuran todas las preguntas

que fueron realizadas en el cuestionario, junto con la hora de inicio y de fin de las tareas llevadas a cabo. Se reporta el archivo excel como anexo en la sección 8.

6.1.5. Evaluación de Resultados e Implicación

Para responder a la pregunta de investigación, se han evaluado las muestras de los participantes y se determinó el resultado de las mismas en base al nivel de aceptación percibido por los sujetos, la completitud de las tareas llevadas a cabo y la comparación de resultados entre los dos casos de estudio del experimento.

Para esto utilizamos un SUS (System Usability Scale) una herramienta simple y confiable, útil para medir la usabilidad de los sistemas, permitiéndonos diferenciar entre sistemas utilizables e inutilizables. El mismo consiste en un cuestionario de 10 preguntas generales con cinco opciones de respuesta para los encuestados, que varían de "Totalmente de acuerdo" a "Totalmente en desacuerdo". Con todas las respuestas, es posible obtener un puntaje promedio final por cada participante, pudiendo sacar conclusiones acerca de si el sistema pudo cumplir o no con las expectativas del usuario.

Adicionalmente, se calculó el tiempo requerido para que cada individuo complete el experimento. Se pudo completar la tarea en todos los casos. El tiempo máximo de duración fue de 12 min, y el mínimo de 4 min, siendo la duración promedio de 6 min. La variación de tiempos pudo verse afectada por la diferencia de contenidos consumidos en los distintos días en que se llevó a cabo el experimento.

En su gran mayoría se detectó una gran aceptación por parte de los participantes hacia la tecnología probada, indicando que fue sencillo utilizar la herramienta y que no se detectaron errores durante las pruebas. En cuanto a los resultados del SUS utilizado, de un total de 20 encuestados, sólo uno no logró superar el puntaje promedio de usabilidad, mientras que el resto logró superarlo, en su mayoría holgadamente.

El puntaje SUS promedio dentro del grupo de 4 participantes que utilizaron SkillHub es igual a 88.75, mientras que el puntaje promedio dentro del grupo de 4 participantes que utilizaron skills nativas es igual a 86.87. Podemos observar que en ambos casos el puntaje promedio se acerca bastante al puntaje ideal de un SUS (100), destacando además que el puntaje obtenido por el grupo que utilizó SkillHub (nuestro enfoque) es 2 puntos superior al que utilizó skills nativas.

Por lo tanto, se puede llegar a la conclusión que es posible consumir contenido web a partir de la interaccion con interfaces auditivas satisfacto-

riamente con nuestro enfoque.

Capítulo 7

Resultados y discusiones

En base a la presente tesina, hemos desarrollado un paper (End-User Development of Voice User Interfaces based on Web content) que fue publicado y presentado en el marco del congreso bianual internacional ISEUD 2019, que tuvo lugar en la Universidad de Hertfordshire, Reino Unido. El trabajo de investigación fue presentado oralmente por los autores de esta tesina.

7.1. Resultados esperados

Buscamos que la implementación de nuestra herramienta les permita a los usuarios de dispositivos de interacción auditiva obtener cualquier contenido que pueda ser accedido mediante un navegador web. Esto facilitaría el acceso a la información a través de una tecnología más cómoda y accesible, inclusive para personas con discapacidades motrices o de visión.

Por otra parte, la inclusión de nuevas interfaces como la mencionada anteriormente, significaría un nuevo medio de acceso hacia una gran cantidad de información que ya existe, pero que sólo puede ser accedida por medio de una interfaz puramente visual.

Se busca, a partir de nuestro desarrollo, potenciar aún más el uso de esta clase de dispositivos para lograr sacar mayores conclusiones con respecto a esta nueva forma de interacción entre los usuarios y los contenidos web disponibles.

7.2. Conclusiones y trabajos futuros

Las VUI han ido incrementando su uso para permitir la comunicación con los dispositivos de interacción auditiva. En general, estos dispositivos permiten a los usuarios instalar skills de terceros para incorporar nuevos comportamientos.

En este trabajo, presentamos un enfoque de desarrollo para los usuario finales, que permite la creación de skills propias basadas en VUI, para que sean usadas con fuentes de información y servicios Web preferidos. La creación de VUI basadas en contenidos Web, podría ser una manera interesante de otorgar más control a los usuarios mientras interactúan con sus dispositivos.

Discutimos el fundamento y las mecánicas para adaptar contenido Web dentro de una VUI, el cual consiste en extraer bloques de contenido Web para luego organizarlos dentro de diagramas de flujos que puedan ser interpretados para responder a comandos de voz. También presentamos nuestro entorno EUD, incluyendo el template de extracción para bloques de contenido y SkillMaker, nuestra herramienta EUD utilizada para crear VUI basadas en bloques de contenido. El tiempo durante el proceso de abstracción de contenidos fue muy corto; sólo tomó algunos minutos poder definir bloques de contenido para usarlos luego en SkillMaker. Como prueba de concepto, desarrollamos SkillHub, un skill de Amazon Echo que implementa nuestro enfoque. Utilizamos SkillHub para interactuar con los contenidos definidos en SkillMaker.

Por último, nos parece interesante proponer algunas posibles mejoras a nuestro enfoque como trabajos futuros potenciales:

- Extender los templates de extracción definidos, permitiendo al usuario definir más elementos (además del titulo y cuerpo) que sean parte de la estructura de los contenidos abstraídos, otorgando una mayor variabilidad y empoderando aún más a los usuarios finales.

- Mejorar la obtención de la ruta perteneciente a cada contenido dentro de un sitio web. Nuestra solución se basa en obtener expresiones xpath directamente desde el DOM de una página web, haciendo uso de una libreria Javascript (Puppeteer). Aun así, nos encontramos con el problema de que estas direcciones xpath poseen un tiempo de vida útil, ya que las mismas pueden variar si el sitio web subyacente actualiza su estructura. Desde ese momento, se convierten en direcciones “rotas” dentro de nuestra solución, ya que no permiten ubicar al elemento al que referenciaban en un primer momento.

- Encontrar un método más eficiente para la obtención del texto de los contenidos (como alternativa al uso de la librería Puppeter utilizada en este trabajo). En el servicio creado para tener acceso al DOM de los sitios web, nos encontramos con el problema de que en algunas circunstancias, era demasiado elevado el tiempo de respuesta que conlleva obtener el texto de una página a la cual se accede mediante el método de navegación.

- Contemplar la posibilidad de incorporar el uso de motores de búsqueda dentro de los sitios web, como nuevo medio de navegación, adaptados a una solución que permita interactuar con ellos por medio de VUIs.

Capítulo 8

Anexo

En esta sección se reporta la tabla generada mediante un archivo formato excel con los resultados obtenidos durante el experimento llevado a cabo con los usuarios, el cual fue debidamente reportado en la sección 6 del presente documento.

Timestamp	Experimento noticias	Experimento clima	Creo que me gustaría utilizar este sistema con frecuencia
11/24/2019 17:47:12	Skill Hub	Skill Hub	5
11/24/2019 18:01:28	Skill Hub	Terceros (meteorologia Barcelona)	4
11/24/2019 18:17:40	Terceros (noticias del espacio)	Skill Hub	5
11/24/2019 18:28:51	Terceros (noticias del espacio)	Terceros (meteorologia Barcelona)	5
11/24/2019 18:37:35	Skill Hub	Skill Hub	5
11/24/2019 18:45:10	Skill Hub	Terceros (meteorologia Barcelona)	5
11/24/2019 18:55:33	Terceros (noticias del espacio)	Skill Hub	5
11/24/2019 19:03:06	Terceros (noticias del espacio)	Terceros (meteorologia Barcelona)	5
11/24/2019 19:30:24	Skill Hub	Terceros (meteorologia Barcelona)	5
11/24/2019 19:36:24	Terceros (noticias del espacio)	Skill Hub	4
11/28/2019 11:29:59	Skill Hub	Skill Hub	4
11/28/2019 14:25:51	Skill Hub	Terceros (meteorologia Barcelona)	5
11/28/2019 19:15:10	Terceros (noticias del espacio)	Skill Hub	5
12/1/2019 21:19:05	Terceros (noticias del espacio)	Terceros (meteorologia Barcelona)	3
12/1/2019 21:19:08	Skill Hub	Skill Hub	4
12/1/2019 21:19:09	Skill Hub	Terceros (meteorologia Barcelona)	3
12/2/2019 19:18:14	Terceros (noticias del espacio)	Skill Hub	5
12/2/2019 19:23:07	Terceros (noticias del espacio)	Terceros (meteorologia Barcelona)	4
12/2/2019 19:27:52	Skill Hub	Terceros (meteorologia Barcelona)	4
12/2/2019 19:32:27	Terceros (noticias del espacio)	Skill Hub	5

Encontré el sistema innecesariamente complejo	Pienso que el sistema fue fácil de usar	Creo que necesitaría el apoyo de una persona técnica para poder usar este sistema
1	5	1
1	4	2
1	5	1
1	5	1
1	5	1
2	5	1
1	5	1
1	5	1
1	5	1
1	5	1
1	5	1
4	5	3
1	5	2
2	4	1
1	5	2
1	4	2
1	5	2
2	3	4
1	5	1
3	3	4

Descubrí que las diversas funciones de este sistema estaban bien integradas	Pienso que había demasiada inconsistencia en este sistema
4	1
5	1
3	1
5	1
5	1
5	1
5	1
4	1
5	1
5	1
4	2
5	1
4	1
4	1
4	2
5	1
3	1
4	1
4	1
4	2

Me imagino que la mayoría de la gente aprendería a usar este sistema muy rápidamente	Encontré el sistema muy engorroso de usar	Me sentí muy seguro de usar el sistema
5	2	2
4	1	4
5	1	5
5	1	5
5	1	5
5	1	4
5	2	5
5	1	5
5	1	5
4	1	5
5	2	4
5	1	3
5	1	5
4	1	5
5	4	5
3	2	5
4	1	4
4	1	3
4	1	4
3	3	3

Necesité aprender muchas cosas antes de poder comenzar con este sistema	SUS SCORE	Hora de inicio del experimento	Hora de fin del experimento
1	87.5	5:30:00 PM	5:40:00 PM
1	87.5	5:50:00 PM	5:56:00 PM
1	95	6:00:00 AM	6:12:00 AM
1	100	6:18:00 AM	6:24:00 AM
1	100	6:26:00 PM	6:35:00 PM
2	92.5	6:41:00 PM	6:50:00 PM
1	97.5	6:55:00 PM	7:03:00 PM
1	97.5	7:06:00 PM	7:10:00 PM
1	100	7:27:00 PM	7:34:00 PM
1	95	7:50:00 PM	7:59:00 PM
1	87.5	11:13:00 AM	11:18:00 AM
3	77.5	2:12:00 PM	2:16:00 PM
1	95	6:56:00 PM	7:01:00 PM
2	82.5	9:05:00 PM	9:10:00 PM
2	80	9:10:00 PM	9:15:00 PM
1	82.5	9:00:00 PM	9:05:00 PM
2	85	7:12:00 PM	7:18:00 PM
3	67.5	7:20:00 PM	7:28:00 PM
1	90	7:30:00 PM	7:36:00 PM
4	55	8:03:00 PM	8:10:00 PM

¿Considerás que es difícil usar la herramienta? ¿Por qué?
No, no lo considero dificil
No, se puede manejar fácilmente sin estar muy informado sobre el tema.
No
No
No
No
No, una vez que sabes que decir es realmente sencillo.
No es para nada difícil, sobre todo para quienes habituamos usar dispositivos.
No, para nada.
No
No, porque trabajo en el sector de sistemas.
No es difícil usar la herramienta con un entrenamiento previo o una explicación básica del uso del dispositivo
NO, me resultó sencillo y amigable
No
No, no es difícil, fue práctica y consisa
No
No, me resultó sencilla ya que la aplicacion misma te iba guiando
No, pero me gustaría tener más ayudas en cada paso
No, al estar acostumbrado a usar la tecnologia me parecio sencilla
Al principio me costo entender lo que tenia que decir, pero despues me resulto mas facil

¿Qué otros usos de la herramienta se te ocurren?	¿Detectó algún problema en el uso de la aplicación? ¿Cuál/es?
integración con reconocimiento de imagen	No
Enseñanza.	Si, en la pronunciación.
escuchar música	no, ninguno
Ninguno	No
Ninguno	Ninguno
Ninguno	Ninguno
No sabría decir	No
Nada que agregar al lector de contenidos.	No detecte ningún problema.
No se me ocurre en éste momento.	Ningún problema detectado durante el experimento.
Ninguno	No
Poder organizar las noticias por temas y solicitar posibles alertas meteorólogicas.	No.
Escuchar música	Ninguno
me gustaría consumir los contenidos seleccionando los temas de modo aleatorio	NO
Que me lea novelas de ficcion	No
Obtener información sin la necesidad de estar al lado de la pantalla	No
Obtener información sin la necesidad de escribir	No, ninguno
Obtener recetas de cocina	No, ninguno
Obtener información de series y películas	No
Poder hacer busquedas en google	No
Que te pueda recordar las tareas del dia	No

¿Crees que hay algún aspecto de la herramienta que podemos mejorar? ¿Cuáles?
Si, la adaptación a un lenguaje "de la calle"
Si, en la anteriormente mencionada.
no
La forma de invocar podría ser más reducida
Ninguna
Ninguno
No se me ocurre
No realmente.
No
No
No.
Se podría evitar algunos detalles extras (por ejemplo al informar el sitio desde donde se obtiene el contenido)
ya lo mencioné en respuesta anterior
No
Si, el dialecto de alexa
Si, el vocablo de Alexa
No
Como comenté antes, se podrían brindar más ayudas
La pronunciacion
Me costó diferenciar el texto de la noticia con las palabras que tenia que decir para continuar

¿Tiene algún otro comentario sobre la herramienta, el enfoque o el experimento?
Tiene muchas aplicaciones futuras, especialmente en procesos de enseñanza y
No.
no
No
Me pareció muy útil
Me parece ideal para estar actualizado
Es una gran herramienta. Soy docente y poder corregir y escuchar las noticias al mismo tiempo sería algo sumamente útil para mi día a día.
Fue un experimento sencillo de realizar, no resultó aburrido ni extenso.
Fue entretenido realizar el experimento, me gustaría poder utilizar la herramienta a diario.
No
Me hubiese gustado probar más funcionalidades orientadas a situaciones del día a día.
Me gustaría utilizarlo en la vida diaria, mientras realizo las actividades del hogar
Es una herramienta innovadora y útil en orden de prioridades: mayores - con discapacidad visual - niños - el resto de la población
No
No, ninguno
Me parece genial para usarla mientras cocino o estoy manejando
Me pareció muy util para mis actividades diarias
No
Estuvo buena la experiencia, creo que la usaria bastante durante el dia

Bibliografía

[1] A. Purington, J. G. Taft, S. Sannon, N. N. Bazarova, and S. H. Taylor, "Alexa is my new bff: social roles, user satisfaction, and personification of the amazon echo," in *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 2853–2859, ACM, 2017.

[2] P. Cohen, A. Cheyer, E. Horvitz, R. El Kaliouby, and S. Whittaker, "On the future of personal assistants," in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 1032–1037, ACM, 2016.

[3] N. Zhang, X. Mi, X. Feng, X. Wang, Y. Tian, and F. Qian, "Understanding and mitigating the security risks of voice-controlled third-party skills on amazon alexa and google home," *arXiv preprint arXiv:1805.01525*, 2018.

[4] "Ifttt and amazon alexa," 2019.

[5] A. Rajalakshmi and H. Shahnasser, "Internet of things using node-red and alexa," in *2017 17th International Symposium on Communications and Information Technologies (ISCIT)*, pp. 1–4, IEEE, 2017.

[6] M. Brambilla, J. Cabot, and M. Wimmer, "Model-driven software engineering in practice," *Synthesis Lectures on Software Engineering*, vol. 3, no. 1, pp. 1–207, 2017.

[7] N. Elouali, J. Rouillard, X. Le Pallec, and J.-C. Tarby, "Multimodal interaction: a survey from model driven engineering and mobile perspectives," *Journal on Multimodal User Interfaces*, vol. 7, no. 4, pp. 351–370, 2013.

[8] G. Ripa, M. Torre, S. Firmenich, and G. Rossi, "End-user development of voice user interfaces based on web content," in *International Symposium on End User Development*, pp. 34–50, Springer, 2019.

[9] S. Firmenich, G. Bosetti, G. Rossi, and M. Winckler, "End-user software engineering for the personal web," in *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*, pp. 216–218, IEEE, 2017.

[10] J. C. R. Licklider, "Man-computer symbiosis," *IRE transactions on human factors in electronics*, no. 1, pp. 4–11, 1960.

[11] J. Cassell, T. Bickmore, M. Billinghurst, L. Campbell, K. Chang, H. Vilhjálmsson, and H. Yan, "Embodiment in conversational interfaces: Rea," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 520–527, ACM, 1999.

[12] M. W. Kadous and C. Sammut, "Inca: A mobile conversational agent," in *Pacific Rim International Conference on Artificial Intelligence*, pp. 644–653, Springer, 2004.

[13] M. F. McTear, Z. Callejas, and D. Griol, *The conversational interface*, vol. 6. Springer, 2016.

[14] M. Baez, F. Daniel, and F. Casati, "Conversational web interaction: Proposal of a dialog-based natural language interaction paradigm for the web," in *International Workshop on Chatbot Research and Design*, pp. 94–110, Springer, 2019.

[15] J. Hauswald, M. A. Laurenzano, Y. Zhang, C. Li, A. Rovinski, A. Khurana, R. G. Dreslinski, T. Mudge, V. Petrucci, L. Tang, *et al.*, "Sirius: An open end-to-end voice and vision personal assistant and its implications for future warehouse scale computers," in *ACM SIGPLAN Notices*, vol. 50, pp. 223–238, ACM, 2015.

[16] A. Pradhan, K. Mehta, and L. Findlater, "Accessibility came by accident: use of voice-controlled intelligent personal assistants by people with disabilities," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 459, ACM, 2018.

[17] M. H. Cohen, M. H. Cohen, J. P. Giangola, and J. Balogh, *Voice user interface design*. Addison-Wesley Professional, 2004.

[18] E. Ferrara, P. De Meo, G. Fiumara, and R. Baumgartner, "Web data extraction, applications and techniques: A survey," *Knowledge-based systems*, vol. 70, pp. 301–323, 2014.

[19] F. Bentley, C. Luvogt, M. Silverman, R. Wirasinghe, B. White, and D. Lottrjdge, "Understanding the long-term use of smart speaker assistants," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 3, p. 91, 2018.

[20] E. Kurniawati, L. Celetto, N. Capovilla, and S. George, "Personalized voice command systems in multi modal user interface," in *2012 IEEE International Conference on Emerging Signal Processing Applications*, pp. 45–47, IEEE, 2012.

[21] J. Jeong and D.-H. Shin, "It's not what it speaks, but it's how it speaks: A study into smartphone voice-user interfaces (vui)," in *International Conference on Human-Computer Interaction*, pp. 284–291, Springer, 2015.

[22] D. Schnelle and F. Lyardet, "Voice user interface design patterns.," in *EuroPLoP*, pp. 287–316, 2006.

[23] S. Schlögl, G. Chollet, M. Garschall, M. Tscheligi, and G. Legouverneur, "Exploring voice user interfaces for seniors," in *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments*, p. 52, ACM, 2013.

[24] Y.-Y. Fan, S. Shin, and V. Samanta, "Contour: An efficient voice-enabled workflow for producing text-to-speech content," in *Adjunct Publication of the 30th Annual ACM Symposium on User Interface Software and Technology*, pp. 133–135, ACM, 2017.

[25] R. Soic, M. Vukovic, and Z. Car, "Enabling text-to-speech functionality for websites and applications using a content-derived model.," *ICST Trans. Ambient Systems*, vol. 4, no. 13, p. e3, 2017.

[26] P. Verma, R. Singh, and A. K. Singh, "A framework to integrate speech based interface for blind web users on the websites of public interest," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, p. 21, 2013.

[27] D. Sato, S. Zhu, M. Kobayashi, H. Takagi, and C. Asakawa, "Sasayaki: augmented voice web browsing experience," in *Proceedings of the SIG-*

CHI conference on Human Factors in Computing Systems, pp. 2769–2778, ACM, 2011.

[28] E. Goldstein, R. E. Machesky, M. Babineau, D. Krzanowski, and H. Thuma, "System, method and apparatus for selecting, displaying, managing, tracking and transferring access to content of web pages and other sources," May 8 2007. US Patent 7,216,290.

[29] G. Rossi, F. Lyardet, and D. Schwabe, "Patterns for e-commerce applications.," in *EuroPlop*, pp. 269–282, 2000.

[30] E. Corbett and A. Weber, "What can i say?: addressing user experience challenges of a mobile voice user interface for accessibility," in *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 72–82, ACM, 2016.

[31] R. Khare and T. Çelik, "Microformats: a pragmatic path to the semantic web," in *Proceedings of the 15th international conference on World Wide Web*, pp. 865–866, ACM, 2006.

[32] C. Bizer, K. Eckert, R. Meusel, H. Mühleisen, M. Schuhmacher, and J. Völker, "Deployment of rdfa, microdata, and microformats on the web–a quantitative analysis," in *International Semantic Web Conference*, pp. 17–32, Springer, 2013.

[33] M. Van Kleek, B. Moore, D. R. Karger, P. André, *et al.*, "Atomate it! end-user context-sensitive automation using heterogeneous information sources on the web," in *Proceedings of the 19th international conference on World wide web*, pp. 951–960, ACM, 2010.

[34] R. J. Ennals and M. N. Garofalakis, "Mashmaker: mashups for the masses," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pp. 1116–1118, Citeseer, 2007.

[35] I. Aldalur, M. Winckler, O. Díaz, and P. Palanque, "Web augmentation as a promising technology for end user development," in *New Perspectives in End-User Development*, pp. 433–459, Springer, 2017.

[36] G. Bosetti, S. Firmenich, A. Fernandez, M. Winckler, and G. Rossi, "From search engines to augmented search services: an end-user development approach," in *International Conference on Web Engineering*, pp. 115–133, Springer, 2017.

[37] S. Firmenich, G. Bosetti, G. Rossi, M. Winckler, and T. Barbieri, "Abstracting and structuring web contents for supporting personal web experiences," in *International Conference on Web Engineering*, pp. 77–95, Springer, 2016.

[38] A. J. Ko, R. Abraham, L. Beckwith, A. Blackwell, M. Burnett, M. Erwig, C. Scaffidi, J. Lawrance, H. Lieberman, B. Myers, *et al.*, "The state of the art in end-user software engineering," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 21, 2011.

[39] I. Aldalur and O. Diaz, "Addressing web locator fragility: a case for browser extensions," in *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, pp. 45–50, ACM, 2017.