

Recuperación de información y recomendaciones en la web

Informe final - Grupo N° 16

Facultad de Ingeniería - Udelar - 2018

Agustín Talagorria 4.747.056-9

Mathias Mendes Lopes 4.895.346-9

Gonzalo Rodriguez 4.671.685-5

Martín Pérez 4.591.307-2

Índice

Introducción	4
Motivación y Descripción del problema.	5
Enfoque de la solución	6
Diseño e implementación	7
Descripción general	7
Limitaciones	7
Arquitectura	7
Bases de datos	8
Tecnologías Utilizadas	9
Android	9
NodeJS	10
Funcionamiento	10
¿Cuándo utilizarlo?	10
Herramientas	11
Google Flights	11
Lugares de Destino - Google	11
Scraping	12
Recuperación de información desde la Web	12
Puppeteer	12
Generación de combinaciones	12
Paralelización	12
Conclusiones	13
Trabajos a futuro	14
Referencias	15

Introducción

En el siguiente documento se describe el conjunto de etapas abordadas para la construcción del sistema “Travel Solutions”.

A lo largo del documento se elaboran distintas secciones en las cuales se explica detalladamente las etapas de la construcción del sistema. En particular, se especifica el problema a abordar, el enfoque de solución propuesto, las funcionalidades del sistema implementado y sus limitaciones. Por otra parte se presentan los resultados obtenidos y se incluye una sección en la cual se describen sugerencias de funcionalidades y/o mejoras a incluir en el sistema en futuras versiones del mismo.

Para la realización del obligatorio se identifica un problema el cual puede ser resuelto mediante la extracción de información disponible en la web, la cual luego de procesada y presentada al usuario, permite resolver el problema identificado. Se pretende además que el sistema facilite el proceso de toma de decisiones a los usuarios.

Motivación y Descripción del problema.

Teniendo en cuenta la dificultad que requiere planear un viaje al exterior en lo que refiere a la búsqueda de aerolíneas disponibles, precios de los vuelos, combinaciones entre distintos vuelos, o simplemente la elección de los diferentes destinos turísticos y sumado a esto, el interés de reducir el tiempo empleado que genera el planear un viaje de este estilo así como también el dinero invertido, se decidió abordar dicha problemática.

Por consiguiente, se elige como problema a tratar la recuperación de información desde la web sobre precios de viajes en distintas agencias de vuelos, combinaciones entre vuelos, disponibilidad de los mismos e información sobre los distintos lugares turísticos en los que el usuario podría estar interesado.

Para el usuario promedio, la principal limitante a la hora de plantear un viaje al exterior, es el tema económico, por dicho motivo sería de utilidad tener una herramienta que permita optimizar el recurso dinero, sin hipotecar la cantidad de destinos ni la calidad del viaje en sí.

Por otra parte un factor que genera dificultad para el usuario es la búsqueda de aerolíneas disponibles y las combinaciones que se deben realizar para llegar al destino elegido, ya que desde nuestro país son muy pocos los destinos accesibles en vuelo directo.

Para abordar la elección de los atractivos que el usuario desearía visitar en cada destino de su viaje, se plantea que la herramienta permite parametrizar según sus intereses, y de esta forma poder presentar los lugares con mayor efectividad en cuanto a su gusto personal.

Enfoque de la solución

La intención del grupo es crear una aplicación, llamada “TravelSolutions” la cual permite en una primera instancia seleccionar un país de origen (Uruguay), varios países destino de interés (dentro de Europa), además, se le va a solicitar una fecha de inicio del viaje y la cantidad de días que piensa quedarse en cada destino. Para de esta forma, poder tener con mayor claridad la disponibilidad del usuario para recorrer cada destino. En particular esta información resultará de gran utilidad a la hora de especificar qué lugares y cuantos se le recomendará visitar. De esta forma se permite, realizar una búsqueda en base a esos datos.

Más precisamente, se intentará mediante la recolección de estos datos, obtener resultados de todas las combinaciones de vuelos posibles y además brindar la posibilidad de conocer detalles específicos como por ejemplo, ¿que se busca minimizar?, es decir, si el usuario desea minimizar según el precio, ó le interesa en mayor medida reducir el tiempo que le insumirá el traslado, (filtrar según la duración total del vuelo).

Si el usuario prioriza reducir los costos, con la información recogida, la aplicación podrá brindar un recorrido por todos los países de interés con el menor precio posible. En cambio, si se eligió minimizar el tiempo de viaje, se podrá sugerir un recorrido minimizando tiempos en las escalas en aeropuertos intermedios. Un parámetro interesante con el que se puede trabajar para lograr estos objetivos es la permutación de los destinos en función de la “facilidad” de locomoción para llegar a cada uno de ellos, así como también tener la capacidad de variar mínimamente las fechas de arribo y partida de cada destino. De esta forma se puede encontrar variantes en lo referente a tiempo de viaje y costo del mismo, permitiendo brindar un abanico más amplio de opciones.

Además para cada uno de los países destino, se recomiendan lugares turísticos los cuales visitar con información sobre los mismos. Dando un pantallazo general de cada sitio, descripción general, horarios en los que se puede visitar y una sugerencia de como llegar a los mismos.

Diseño e implementación

Descripción general

“Travel Solutions” permitirá a los usuarios conocer detalles en cuanto a precios y tiempo insumido para realizar viajes desde Uruguay a, potencialmente, varios destinos. Los países destino serán seleccionados desde una lista confirmada por 48 países pertenecientes al continente Europeo.

A su vez, el usuario deberá proporcionar una fecha de inicio del viaje y cuánto tiempo desea estar en cada destino seleccionado. Información que será utilizada por la aplicación en pos de realizar los cálculos necesarios para arrojar los datos solicitados, con la salvedad de que en ciertos casos se podrá tomar la libertad de variar mínimamente alguno de los plazos que el usuario estipulo estar en cada destino siempre y cuando esta modificación de los parámetros mejore la solución final.

Finalmente, para cada país seleccionado como destino, la aplicación proporcionará una serie de sitios de interés, brindados como sugerencia con el fin de facilitar al usuario la elección de qué lugares visitar en su estadía.

Limitaciones

La principal limitación del sistema construido es la escasa variedad de posibles destinos, así como también que está planteado para funcionar teniendo solamente a Uruguay como país de origen. Por otra parte los sitios de interés de cada destino brindados son genéricos, es decir no permite filtrar los resultados en base a los intereses de los usuarios. En la sección Trabajo a futuro se detallan una serie de mejoras que se podrían aplicar al producto para mejorar sus limitaciones y explotar sus virtudes.

Arquitectura

El sistema consta de una arquitectura cliente-servidor, donde la aplicación se comunica con el servidor. A su vez el servidor se comunica con un servidor de base de datos MySQL, donde se insertan los posibles países destino, conteniendo de cada uno un identificador, el nombre, una imagen y el código utilizado para identificarlo a la hora de realizar el scrapping para obtener la información de cada uno.

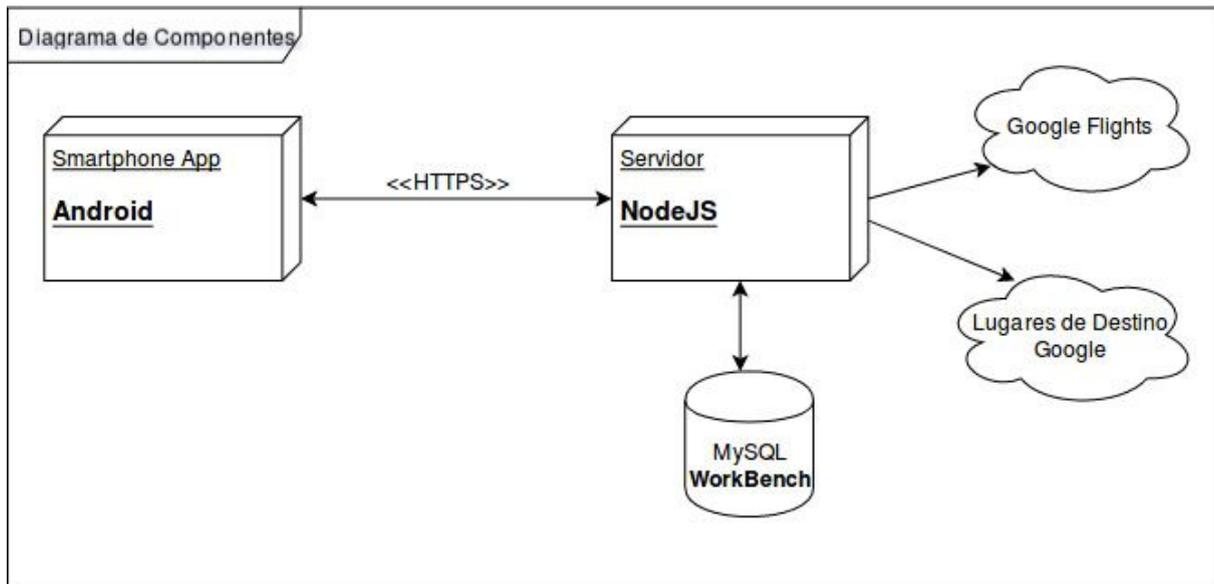


Figura 1 - Diagrama.

Bases de datos

Para el manejo de la base de datos MySQL se utilizó la herramienta de diseño visual WorkBench desde donde se crean las estructuras pertinentes y se carga la base con los datos antes mencionados referentes a cada país. En la figura 2 se muestra la instancia de la tabla “country” utilizada en el desarrollo de la aplicación.

Result Set Filter:

#	idcountry	countryName	countryImage	countryCode
1	1	Alemania	https://www.fotopaises.com/Fotos-Paises/t200/2003/3/8/30_3053.jpg	0345h
2	2	Albania	https://www.fotopaises.com/Fotos-Paises/t200/2002/11/29/30_676.jpg	0jdx
3	3	Andorra	https://www.fotopaises.com/Fotos-Paises/t200/2005/11/14/1498_1132054528.jpg	0hg5
4	4	Armenia	https://www.fotopaises.com/Fotos-Paises/t200/2005/9/15/1279_1126972944.jpg	0jgx
5	5	Austria	https://www.fotopaises.com/Fotos-Paises/t200/2007/3/8/3298_1173343916.jpg	0h7x
6	6	Azerbaiján	https://www.fotopaises.com/Fotos-Paises/t200/2006/8/4/2570_1154700682.jpg	0jhd
7	7	Bélgica	https://www.fotopaises.com/Fotos-Paises/t200/2007/3/1/3360_1172776481.jpg	0154j
8	8	Bielorrusia	https://www.fotopaises.com/Fotos-Paises/t200/2002/12/19/30_1152.jpg	0163v
9	9	Bosnia y Herzegovina	https://www.fotopaises.com/Fotos-Paises/t200/2002/12/23/30_1271.jpg	0166b
10	10	Bulgaria	https://www.fotopaises.com/Fotos-Paises/t200/2007/5/1/3777_1178183541.jpg	015qh
11	11	Chipre	https://www.fotopaises.com/Fotos-Paises/t200/2003/2/13/30_2513.jpg	01ppq
12	12	Croacia	https://www.fotopaises.com/Fotos-Paises/t200/2003/2/11/30_2467.jpg	01pj7
13	13	Dinamarca	https://www.fotopaises.com/Fotos-Paises/t200/2003/2/15/30_2557.jpg	121p5z1h
14	14	Eslovaquia	https://www.fotopaises.com/Fotos-Paises/t200/2005/9/15/1279_1126972944.jpg	06npd
15	15	Eslovenia	https://www.fotopaises.com/Fotos-Paises/t200/2004/3/5/83_8999.jpg	06t8v
16	16	España	https://www.fotopaises.com/Fotos-Paises/t200/2005/11/30/1586_1133343878.jpg	06mkj
17	17	Estonia y Herzegovina	https://www.fotopaises.com/Fotos-Paises/t200/2004/3/5/83_8999.jpg	02kmm
18	18	Finlandia	https://www.fotopaises.com/Fotos-Paises/t200/2003/2/23/30_2752.jpg	02vzc
19	19	Francia	https://www.fotopaises.com/Fotos-Paises/t200/2007/4/29/3463_1177888403.jpg	0f89c
20	20	Georgia	https://www.fotopaises.com/Fotos-Paises/t200/2003/5/12/30_4627.jpg	0d0kn
21	21	Grecia	https://www.fotopaises.com/Fotos-Paises/t200/2006/8/2/2515_1154564434.jpg	035qy
22	22	Hungría	https://www.fotopaises.com/Fotos-Paises/t200/2003/3/22/30_3399.jpg	03gj2
23	23	Irlanda	https://www.fotopaises.com/Fotos-Paises/t200/2008/8/24/1128_1219650222.jpg	03rt9
24	24	Islandia	https://www.fotopaises.com/Fotos-Paises/t200/2006/10/12/2874_1160915037.jpg	03rj0
25	25	Italia	https://www.fotopaises.com/Fotos-Paises/t200/2005/3/29/767_1112136747.jpg	03rj1

country 2 x

Exported resultset to /home/mathias/paises.xml

Tecnologías Utilizadas

A continuación se presentarán las tecnologías utilizadas para la implementación tanto del Frontend con Android como el Backend en NodeJS.

Android

Para el desarrollo del Frontend se decidió utilizar Android, las siguientes son algunas de las características que hacen que sea uno de los lenguajes de desarrollo más utilizados en el mercado tecnológico.

- El gran beneficio que ofrece Android es que no es solo un sistema operativo más, sino que se trata de un ecosistema completo que ha logrado cautivar a los usuarios, desarrolladores, fabricantes y operadores en una relación gana-gana.
- Está desarrollado bajo una plataforma de código abierto, lo que permite una amplia interacción por parte de los desarrolladores y usuarios.
- Comunidad de desarrolladores. Android tiene la ventaja de contar con la más grande comunidad de desarrolladores, lo que ha generado un concurrido entorno de eventos, foros, grupos de participación y concursos que optimizan los servicios y aplicaciones ofrecidos a usuarios.
- Es un sistema operativo creado por Google, lo que facilita y acelera el uso de varios de los principales servicios de Internet, como Gmail, Gtalk, Google Maps y YouTube.
- A pesar de ser un sistema operativo abierto creado por una sola firma, hay varias empresas fabricantes que lo apoyan a través de Open Handset Alliance y que lo incluyen en los nuevos teléfonos que lanzan al mercado.
- Precios más bajos. Al existir un amplio mercado de productos con Android, los precios de los dispositivos son más bajos a los ofrecidos con otros sistemas. De igual forma, en el mercado Android se encuentran equipos de todos los precios y diferentes gamas.
- Acceso a aplicaciones. De 20.000 aplicaciones que se tenían en la tienda de Android en 2008 (cuando se lanzó) se ha pasado a un poco más de 200.000 en el 2011 y en la actualidad es la que más aplicaciones gratuitas ofrece a los usuarios.
- Otros dispositivos. El sistema operativo está presente en teléfonos móviles, pero también en otros aparatos (por ejemplo tabletas) por lo que es muy fácil y 'amigable' continuar con los beneficios del sistema en otros ambientes.

NodeJS

Para el desarrollo de la API se decidió utilizar NodeJS, a continuación se expondrá una breve descripción del lenguaje y sus principales características.

Funcionamiento

NodeJS está basado en el motor V8 de Javascript de Google, el cual está diseñado para correr en un navegador y ejecutar el código de Javascript de una forma extremadamente rápida. La tecnología detrás de NodeJS permite ejecutar este motor en el lado del servidor, abriendo un nuevo abanico de posibilidades en cuanto al mundo de desarrollo se refiere. Para ello el entorno de NodeJS ha desarrollado un conjunto amplio de librerías comparables a las de otras plataformas. Y, por otro lado, se han eliminado algunas funcionalidades que en el entorno de servidor no tenían sentido como por ejemplo el uso de Document Object Model. NodeJS trabaja con un único hilo de ejecución que es el encargado de organizar todo el flujo de trabajo que se deba realizar.

¿Cuándo utilizarlo?

Básicamente NodeJS es adecuado para esos momentos en que necesitas hacer muchas cosas al mismo tiempo; sobre todo muchas operaciones I/O a la vez (acceso a ficheros, bases de datos, entre otros).

Y es especialmente bueno para aplicaciones en tiempo real, que necesitan mantener una conexión persistente entre el browser y el servidor (juegos online, chats, herramientas de colaboración, etc.).

Entre las cinco ventajas de utilizarlo, destacan:

- Con Node puedes utilizar javascript como lenguaje de scripting en tu consola (como bash, perl, etc.).
- Permite utilizar el mismo lenguaje (javascript), tanto en el cliente como en el servidor.
- Ofrece muy buena gestión de paquetes gracias a NPM (si quieres hacer algo, probablemente exista una librería/paquete que ya lo ofrezca). Detrás de Node hay una gran comunidad documentando, haciendo tutoriales y creando nuevos módulos.
- Con NodeJS es posible hacer en el servidor, todo lo que necesitas – acceso a ficheros, a bases de datos, conexiones de clientes, entre otros.
- Está basado en eventos, así que toda la filosofía asíncrona que ya utilizas con AJAX en tu cliente, la puedes pasar al servidor.

Herramientas

Como fuente para la obtención de datos se utilizó Google Flight en lo referente a la información de los vuelos y Lugares de Destino de Google para la extracción de los sitios de interés para cada destino seleccionado.

Google Flights

Google Flights es una aplicación de Google que ayuda a las personas que lo usen a conseguir vuelos; Google Flights te permite buscar cualquier tipo de vuelos, como por ejemplo, por región o país, además, también ofrece un listado con las diferentes opciones de precios que tienes en una misma búsqueda.

Lugares de Destino - Google

Lugares de Destino - Google es una aplicación de Google que ayuda a las personas que lo usen a informarse acerca de los lugares turísticos que podrían visitar en la ciudad dada; Lugares de Destino - Google te permite buscar cualquier tipo de lugar, ya sea buscando por ciudad o por país, brindando información básica tal como horario en los que está abierto, ubicación, y una breve reseña descriptiva.

Scraping

Recuperación de información desde la Web

Puppeteer

Puppeteer es una API para Node.js que entre otras cosas nos permite recuperar contenido de una página web, y además para trabajar con la nueva función de Chrome, Headless Chrome. Headless Chrome, nos permite ejecutar el navegador desde línea de comandos sin tener que abrir ni una sola de sus ventanas. El navegador “simulado” cuenta con las mismas características que nuestro Chrome de siempre. Es decir, tiene sus mismas funciones y es capaz de abrir páginas webs, ejecutar los distintos procesos, albergar caché, etc.

Para la recuperación del contenido que luego utilizaremos para mostrar la información acerca de la escala, precio, descripción y aerolínea desde el país origen a los países destinos, se cuenta con los “countryCode” (códigos identificadores de dichos países obtenidos desde la BD) en las webs de “**Google Flights**” y “**Lugares de Destino - Google**”, luego se procede a armar la URL de lugares turísticos para el país countryCode y se abre el navegador consultando y recuperando los datos.

Una vez obtenida la recuperación, gracias a los métodos “*page.evaluate*” y “*document.querySelector*” se puede obtener el contenido de un tag html determinado dentro del document perteneciente a la página web del país destino buscado, de esa forma se arma un objeto JSON con la información asociada a ese país con countryCode.

Generación de combinaciones

Para la generación de las combinaciones posibles entre los distintos destinos seleccionados, en primer lugar se define si el orden de las combinaciones se minimizará o bien por precio o por duración, luego se obtiene mediante lo que se seleccionó de la app los códigos de países por los que se va a realizar el viaje, con esos códigos se generan dos estructuras de mapas, una con todas las combinaciones posibles de los países incluyendo a Uruguay, esto se logra generando todas las permutaciones posibles dentro de los id de los países, y otra con los pares de países para saber en qué lugar me va a quedar el resultado del viaje entre los dos países. Para cada par de países obtengo los datos del vuelo entre ellos, una vez que tengo todos los resultados armó la respuesta. Recorro todas las combinaciones de países es decir la lista de los países, uno por combinación (notar que no está Uruguay en estas combinaciones) y para cada una de estas, consideró la combinación y se obtiene para cada país de la misma el viaje del anterior país hacia él (el primero siempre es Uruguay). Entonces se calcula el precio total, a duración total y las escalas totales de la combinación. Para terminar

se agrega el vuelo total de la combinación al arreglo de todos los vuelos y se ordena según el criterio seleccionado (precio o duración)

Paralelización

Para lograr una mejor performance, y lograr obtener los resultados de la recuperación de datos de la web, optamos por devolver por cada combinación de países una nueva promesa “*return new Promise(...)*” el cual ejecuta el código asociado a la función que se le pasa como parámetro a la Promise, devolviendo el resultado o error en caso de que lo haya. Todo esto lo hace asíncronamente (por medio de asignarle al resultado del Scarping el método “*async ()*”) de las demás Promise que se creen por cada combinación, entonces esto se puede ver como varios hilos de ejecución yendo a buscar un resultado de forma independiente cada uno y juntando los resultados de cada uno en un resultado final para luego devolver toda la información solicitada.

Flujo del Prototipo

A continuación se presenta brevemente el flujo de nuestro prototipo.

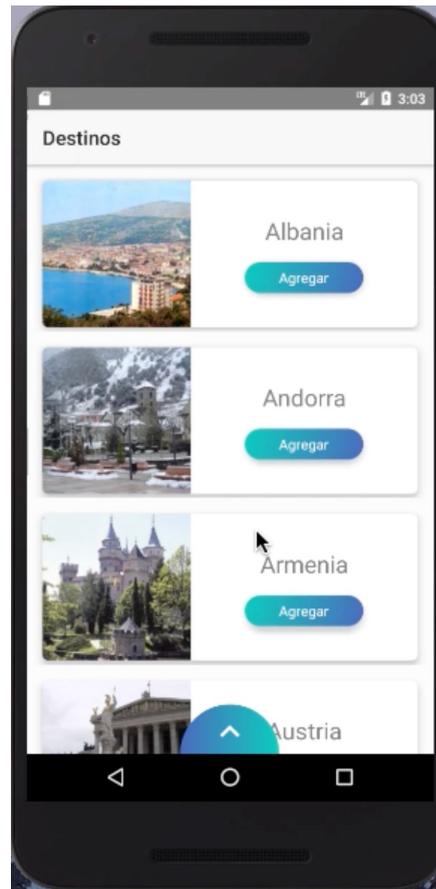


Figura 1.

Al ingresar a la aplicación desde el dispositivo móvil se puede ver la lista de países disponibles para seleccionar como destinos. Dichos países se encuentran ordenados alfabéticamente, por otra parte esta pantalla permite consultar información acerca de cada país (presentado en la figura 2) como también agregar a dicho país como destino, mediante el botón agregar (esto se presenta más en detalle en la figura 3).

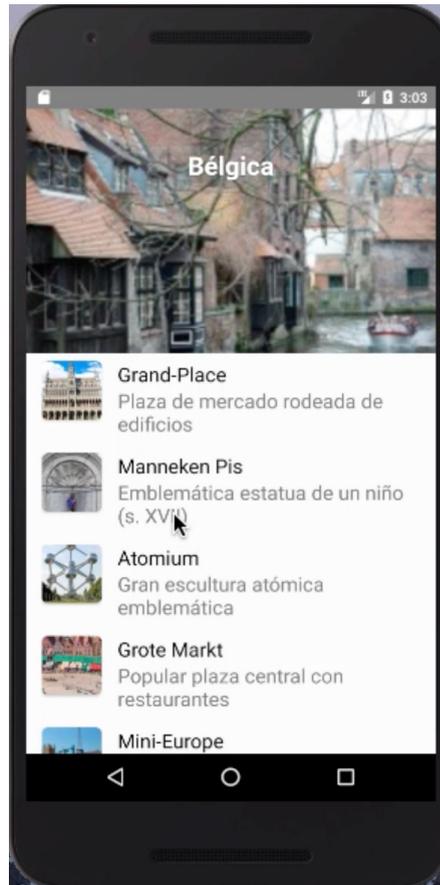


Figura 2.

Una vez que se accede a un determinado país a través de hacer click en su respectiva foto, se puede observar como se nos presenta una lista de distintos lugares turísticos que tiene el país seleccionado, con información básica de cada uno. Concretamente de cada lugar turístico se presenta una imagen y una breve descripción. Y brinda la opción de ver más información acerca de cada uno de estos (presentado en la figura 3).

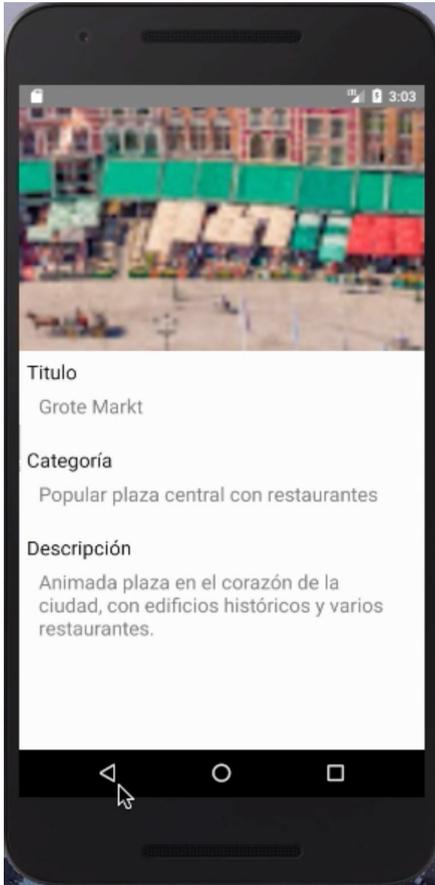


Figura 3.

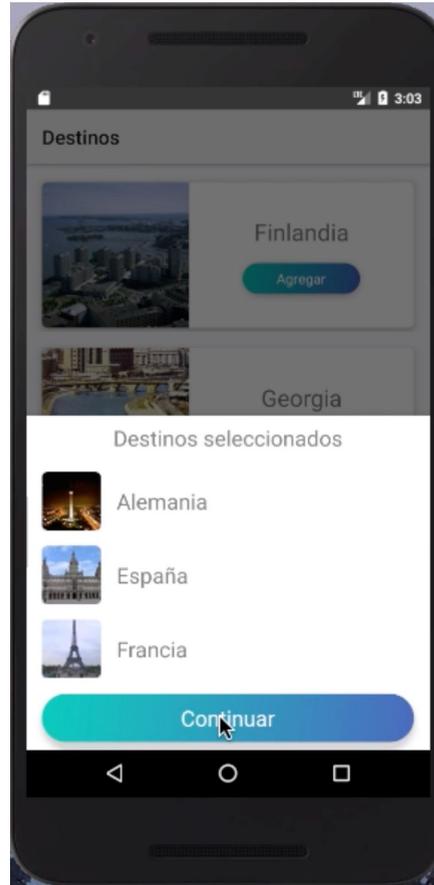


Figura 4.

Cuando se accede a alguno de estos lugares se muestra la información más detallada del mismo, conteniendo el nombre de la atracción una categorización de la misma y una descripción más extensa. Por otra parte, cada vez que se agrega un destino se va conformando una lista con los destinos seleccionados, tal cual se presenta en la figura 4. Como se mencionó previamente un usuario puede agregar un destino clickeando en el botón “agregar” que se muestra en la figura 1.

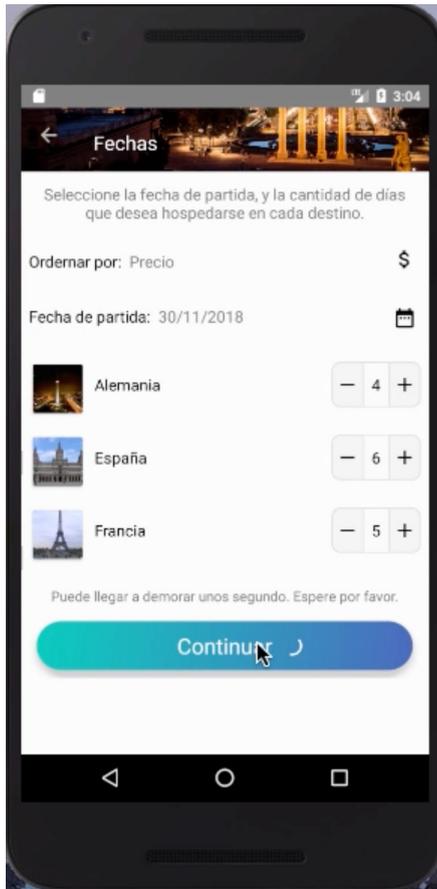


Figura 5.

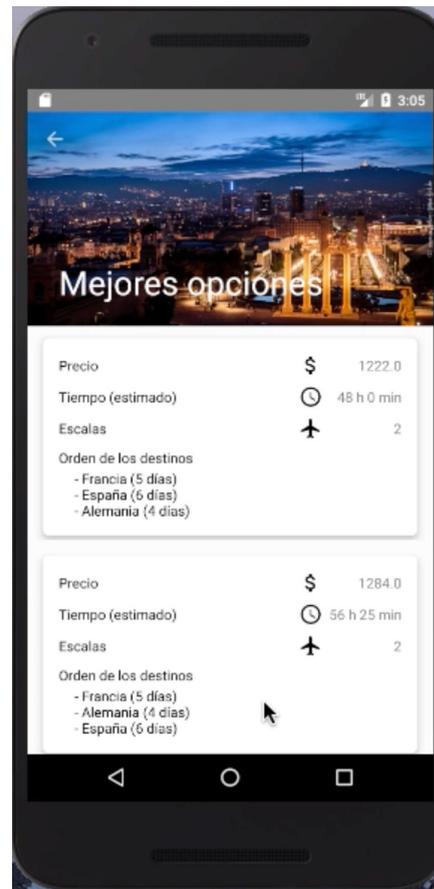


Figura 6.

Una vez que se agregaron todos los países deseados se va al botón “Continuar” (figura 4). Se selecciona el parámetro por el cual se quiere ordenar la combinación, la fecha de comienzo del viaje y la cantidad de días que se quiere estar en cada país (Figura 5). Finalmente, se clickea en el botón “Continuar” y esperamos a que se realice la búsqueda de vuelos y las combinaciones posibles. A continuación se muestran las opciones disponibles ordenadas como se indicó anteriormente, mostrando el precio total, el tiempo estimado, la cantidad de escalas y el orden de los destinos para esa combinación (Figura 6).

Pruebas

Se realizaron pruebas de tiempo, las mismas se efectuaron de dos formas, con fechas en cada destino fijas y en segundo caso con dichas fechas de días variables. Los resultados se muestran en la siguiente imagen.

Tiempos con fecha fija				Tiempo con días variables			
#Test	Países	Orden	Tiempo	#Test	Países	Orden	Tiempo
1		2 Precio	31601.770ms	1		2 Precio	45267.013ms
2		3 Precio	53264.073ms	2		3 Precio	158098.189ms
3		4 Precio	128388.689ms	3		4 Precio	2234682.340ms
4		5 Precio	1844662.420ms	4		5 Precio	3704672.322ms

Se puede ver que para todos los test, la prueba con días variables demora considerablemente más que con las fecha fijas, es más , en algunos casos la demora es más del doble de tiempo. Al probar con más de 6 países destinos, para ambos casos, el tiempo es bastante más alto y no es aconsejable para una app de usuario ya que el tiempo puede molestar a la experiencia de usuario, a su vez se dan problemas en la recuperacion de informacion ya que por motivos de ancho de banda y de procesamiento hace que haya resultados que no se recuperen correctamente. La solución a estos problemas se detalla en la sección de trabajo a futuro.

Conclusiones

Hoy en día las personas cada vez realizan más viajes al exterior, por lo cual está consideramos que esta temática debería ir tomado relevancia en el mercado de software. Dado que generalmente el proceso de búsqueda de aerolíneas disponibles, precios de los vuelos, combinaciones entre distintos vuelos, o simplemente la elección de los diferentes destinos turísticos es engorroso y estresante para los usuarios. Sumado a esto, creemos que el interés por reducir el tiempo empleado en la planeación de un viaje de este estilo así como también minimizar el dinero invertido en el mismo, son de vital importancia. En consecuencia creemos que se podría aprovechar de forma muy beneficiosa la extensa información contenida a lo largo de las páginas web con relación a este tema.

Dicha información, acompañada de herramientas como pueden ser el Scraping de las páginas o una API rest que permite consumir información de la misma, tienen la capacidad de crear aplicaciones como la propuesta por nosotros "Travel Solutions" que tiene un enorme potencial en facilitar problemas cotidianos, como lo es el mencionado anteriormente.

Pudimos a través de un prototipo básico, realizar una App que brinda grandes beneficios para los usuarios que viajan al exterior, lo cual es minimizar según duración y precio del viaje a los distintos destinos, y a su vez mostrar información de interés de los distintos lugares. Creemos que si bien como se detalla en la sección de "*Trabajo a futuro*" falta mucho por avanzar en este prototipo, es evidente el gran beneficio que ofrece la recuperación de información ya que como en este caso, se pudo resolver la problemática planteada en la sección "*Motivación y descripción del problema*" que satisface la necesidad del viajero.

Trabajo a futuro

Se buscará corregir las limitaciones mencionadas anteriormente haciendo que la App sea más escalable, para eso se intentará lograr que la variedad de los países destino no se vea limitada por solo Europa, sino que sea posible seleccionar como destino los países turísticos de los 6 continentes y sus respectivas ciudades más representativas. Además se buscará variar el país origen para que la combinaciones de viajes se calcule a partir de países que no necesariamente sean Uruguay, en si, se planea que sea como los destinos, parametrizable para cualquiera de los 6 continentes. Además se buscará ampliar la cantidad del conjunto de países destino para realizar el viaje, actualmente con 5 países seleccionado como destino la app creemos que demora lo máximo que puede demorar (esta demora se da por la cantidad de permutaciones entre los países y por ende por la cantidad de hilos de ejecución que trabajan abriendo un navegador para recuperar la información de la web) para que el usuario no se desagrada con la espera del resultado. Por lo tanto se trabajará a futuro en encontrar formas de aumentar ese conjunto de países sin afectar la performance y el tiempo de respuesta, para dicha solución dos posibles alternativas podrían ser encontrar una hora "D" en el día, o en la semana, para hacer la recuperación de de información de la web y volcar la misma en base de datos organizada de forma estratégica para su debida recuperación al momento de que sea necesaria, también agregar una caché para que el acceso a la información sea realizado a memorias rápidas y esto mejore la experiencia de usuario. A su vez si se quisiera recuperar información de los vuelos con más de un mes de anticipación, en un principio se hará el scraping en el momento.

Por otro lado, se agregan filtros según lugares de interés, atracciones, zona climática, por temática (ej: tranquilidad, diversión, salidas nocturnas), por estación del año y por culturas, además se permitirá ajustar la búsqueda a determinados parámetros, por ejemplo se podrá ingresar un monto de dinero específico y reducir los resultados de búsquedas a ese monto. A su vez, los resultados de búsqueda mostraran mas informacion, como por ejemplo información del clima que habrá según los días en que se realizará el viaje (calculado por la fecha de inicio y la cantidad de días que se estará en el destino seleccionado), además se agregara información más detallada sobre los países destinos, ciudades, etc.

En otras palabras se buscará compensar las limitaciones que se identificaron a lo largo del proyecto para de esta forma lograr una mejor experiencia de usuario.

Por último, se buscará que nuestra solución de ofrecer todas estas funcionalidades con respecto a los vuelos y lugares de interés se escale e implemente para otros

tipos de transporte, pj: Líneas de trenes, barcos, y ómnibus, dentro de cada continente.

Referencias

- [1] - Android
 - <https://www.dinero.com/negocios/tecnologia/articulo/las-ventajas-ofrec-e-android/118796>
- [2] - NodeJS
 - <https://www.facilcloud.com/noticias/5-ventajas-de-usar-node-js/>
- [3] - Google Flights
 - <https://www.google.com/flights>
- [4] - Lugares de Destino - Google
 - <https://www.google.com/places>
- [5] - Google Flights Description
 - <https://iiemd.com/google-flights/que-es-google-flights>