

# Facultad de Ingeniería

Universidad de la República - Uruguay



## Recuperación de Información y Recomendaciones en la Web (WEBIR)

2017

### Renta de Autos

### Grupo 05

#### Integrantes del grupo:

Paula Martinez Vizoso - 4.756.077-8

Diego Rodriguez Caprani - 4.536.638-4

Diego Agustín Molina Arrieche - 4.815.354-8

Maria Belen Remedi Pertusatti - 4.682.533-9

# Índice

<b>Renta de Autos</b>	<b>1</b>
<b>Índice</b>	<b>2</b>
<b>Introducción</b>	<b>3</b>
<b>Problema y motivación</b>	<b>3</b>
<b>Enfoque de la Solución</b>	<b>4</b>
<b>Diseño e implementación</b>	<b>6</b>
Arquitectura:	6
Tecnologías utilizadas:	6
Filtros	7
<b>Conclusiones y Trabajos a futuro</b>	<b>8</b>
<b>Referencias:</b>	<b>9</b>

# Introducción

Dada la gran cantidad de empresas de alquiler de autos que existen en el mercado, ya sean internacionales o locales, se hace difícil elegir una y poder comparar precios o autos disponibles. Por eso se cree que es adecuado contar con un sitio centralizado donde se pudiera visualizar de forma unificada los distintos modelos de autos que brindan distintas empresas de alquiler de autos.

En este documento se describirá el problema planteado y la solución planteada, mostrando un esquema de la arquitectura desarrollada para el sitio, y mencionando todas las tecnologías utilizadas para el proyecto. Por último se hablará de los posibles trabajos a futuro que se puedan realizar.

## Problema y motivación

La motivación del proyecto surge de la gran cantidad de empresas de renta de automóviles que existen en Uruguay y el mundo, y cada una con su respectiva página web. Estas páginas web carecen de herramientas para realizar búsquedas y filtrados de vehículos según distintas características. Esto dificulta la navegabilidad del usuario y facilidad para encontrar lo que está buscando.

La idea que planteamos es que toda esta información se encuentre centralizada en un mismo lugar de forma homogénea, de forma que el usuario no deba ingresar a todas las páginas web por separado para encontrar lo que busca. Además, se busca que la información sobre los vehículos pueda ser filtrada según las preferencias del consumidor.

## Enfoque de la Solución

Para poder plantear un enfoque a la solución primero se tuvieron que destacar que funcionalidades deseábamos incluir como por ejemplo la implementación de los filtros, así como también que herramientas serían útiles para poder realizarlo.

En primer lugar se tuvo que realizar un fuerte análisis de las distintas empresas de renta de autos que existen. De manera de modelar la información recolectada se debía examinar qué información era proporcionada en cada página y cómo la misma era desplegada, para poder extraerla. También investigar cuáles eran los datos comunes que siempre se muestran de los autos, para así poder decidir qué datos serían importantes para nosotros ya sea para guardar en la base de datos y utilizarlos luego en nuestro sitio, como para poder implementar filtros que sean útiles y funcionales para el usuario.

Decidimos utilizar dos páginas web de empresas distintas como representantes: Plus rent a car y Multicar ambas empresas uruguayas. Esta elección se dio ya que nos pareció que la extracción de datos sería más fácil y ambas despliegan la flota de autos sin pedir antes ningún dato a ingresar lo que facilita el Scapping. Además de que los datos de ambas páginas eran bastante similares por lo que eran compatibles.

El sistema realizado cuenta con tres módulos que interactúan para poder brindar las funcionalidades necesarias. Los mismos son: WebScraper, un Servidor back-end, una interfaz Web y la base de datos.

A continuación realizamos una breve descripción de los mismos:

WebScraper: dicho componente se encargará de la extracción de información de las páginas utilizadas para luego almacenarla en una base de datos de manera estructurada. En particular, se buscan datos asociados a los autos junto con sus precios, y todas las características de cada uno. Se utilizó la librería de java jaunt para este propósito. Este mismo módulo es encargado de persistir la información obtenida por el procedimiento Scapping en una base de datos.

Base de datos: Dado que los datos que nos interesan son específicos, no es demasiada la información a administrar y es posible estructurarla, utilizamos una base de datos relacional, postgres.

Interfaz Web: se realizó una interfaz de usuario para mostrarle al mismo toda la información recabada de los dos sitios elegidos y poder aplicar filtros. Esta se encarga de consumir servicios web RESTful para obtener los datos requeridos por el usuario.

Servidor back-end: Este módulo se encarga de recuperar la información guardada en la base de datos y procesarla de modo de poder aplicar los filtros. Estas funcionalidades son utilizadas por los servicios web para obtener los datos y luego desplegarlos en el front-end.

Durante este proceso, se detectó que cada empresa de renta de autos exponía la información de formas distintas. Esto traía problemas al momento de tratar de unificar los datos de las distintas fuentes, ya que tanto el formato como los nombres de los atributos a recuperar eran diferentes. Otro problema que se observó fue la falta de estandarización. Por ejemplo en una página se hablaba de “valijas” y en otras de “valijas grandes” y “valijas chicas” que cabía en el auto.

Por lo tanto, como cada página web es distinta, se tuvo que realizar dos scraper distintos una para cada página web ya que ambas desplegaban sus datos de una forma distinta.

Sin embargo, como desventaja, esto hace que esta solución no sea muy escalable, ya que, al aumentar la cantidad de empresas, se debería agregar un nuevo scraper por cada uno.

En cuanto al acceso de la información, hemos constatado que no es fácil acceder a API's, además no se ofrecen muchas de forma pública, por eso decidimos hacerlo con Web Scraping que nos pareció más sencillo. Algunos ejemplos de páginas que brindaban una API son las siguientes que se mencionan a continuación:

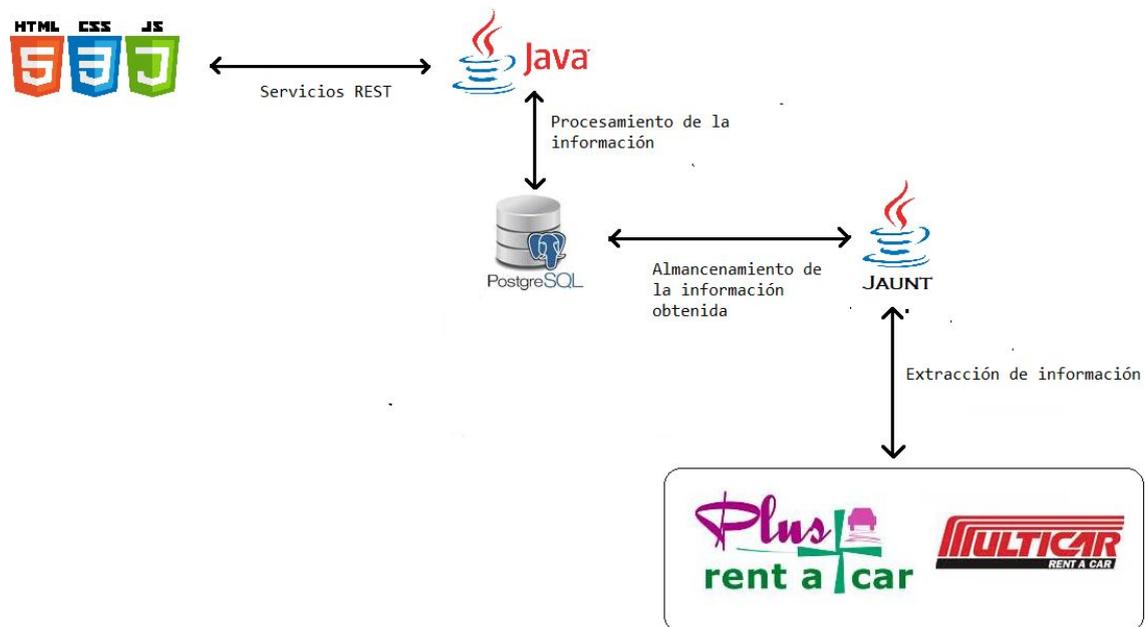
Expedia ofrece una API “car search”, solicitando una “api key” para poder acceder. El usuario puede buscar los autos de alquiler usando los códigos IATA de la ubicación de retiro junto con las fechas de retiro y entrega.

Hotwire “Rental Car Shopping API” permite al desarrollador buscar resultados de alquiler de autos utilizando parámetros similares a los utilizados para iniciar una búsqueda de alquiler en Hotwire.com, incluyendo: destino, fecha y hora de retiro y fecha y hora de entrega.

Los datos devueltos por la API incluye la ubicación y la información de tarifas para cada resultado.

# Diseño e implementación

## Arquitectura:



En el esquema se puede ver la arquitectura planteada para el prototipo con las tecnologías utilizadas. También se puede apreciar el flujo de la información y la interacción con los distintos módulos, desde que se extrae los datos necesarios de los autos de las páginas web de las empresas, hasta que se muestran en pantalla para el usuario en un nuevo sitio y con otra estructura.

## Tecnologías utilizadas:

A continuación se describirán las tecnologías que se utilizaron para la elaboración de este proyecto.

En primer lugar, se utilizó la librería de Jaunt de Java para la parte del web scraping. La misma es una nueva librería que salió en octubre de este año y es muy parecida a Jsoup.

En general, Jaunt parece trabajar en un nivel de abstracción más alto que Jsoup. También proporciona algunas características interesantes listas para usar (como almacenamiento en caché, descubrimiento de paginación web, entre otras cosas).

Además de las características HTML, Jaunt se ha integrado para trabajar con REST API y JSON.

Nos pareció más oportuna de utilizar por las nuevas funcionalidades y porque esta devuelve la información en un Json más fácil de manejar luego.

Utilizamos para la persistencia una base de datos postgres para guardar los datos extraídos de las páginas web.

Para la implementación del back-end, de los filtros y los servicios necesarios para desplegar la información en una página web se utilizó Java.

Y por último, para la parte de front-end del sitio web, se utilizó bootstrap, html, css y para poder invocar los servicios jquery, JavaScript y Rest.

## Filtros

Una de las mayores funcionalidades nuevas del proyecto presentado se basa en los filtros que el usuario puede utilizar para realizar sus búsquedas. Por eso mencionaremos cuales fueron los filtros que se consideraron de serian de mayor importancia y utilidad para el usuario.

Un primer filtro que se consideró es el precio. Este filtro logra que se le muestre al usuario solamente los autos que tengan un precio menor o igual al precio máximo que seleccionó el usuario. De esta forma puede buscar según un presupuesto y que no aparezcan autos a los cuales no podría acceder.

Luego, agregamos el filtro de cantidad de pasajeros, el cual está vinculado con la categoría del auto, es decir si es un auto familiar o estándar, etc. Por lo que solo le aparecerán los autos que mínimo tengan una cantidad de pasajeros determinada por el usuario.

Un tercer filtro es la transmisión del auto, si es automático o manual. De esta forma el usuario podrá filtrar por el que más le interese. Muchas veces es más solicitado y exigido los autos automáticos, por lo que es bueno saber en qué empresa hay disponibles de ese tipo de autos.

Y por último se implementó el filtro de cantidad de valijas que caben en el auto. Con este filtro el usuario puede asegurarse que tendrá espacio suficiente para sus valijas y se listaran los autos que mínimo tengan capacidad para la cantidad de valijas que el usuario digite.

## Conclusiones y Trabajos a futuro

Para resumir el curso de la materia se pudo experimentar el uso de las diferentes tecnologías y técnicas de recuperación, procesamiento y presentación de información de la web realizando un proyecto de corto alcance de forma de obtener resultados aplicando los conocimientos adquiridos en el curso.

Nuestro proyecto consistió en una aplicación web con el fin de brindar a los usuarios finales información de varias rentadoras a la vez. Se ofrece la posibilidad de consultar por los vehículos de acuerdo a ciertas características deseadas como son el precio, capacidad de personas, cilindradas, etc. De esta forma el usuario podría obtener de forma centralizada la información que necesita y no estar consultado varias páginas a la vez.

El producto final es un servicio simple y útil que cumple con los requerimientos iniciales propuestos por el equipo.

Como trabajo a futuro se podría mejorar la parte visual, también a su vez aunque no tan sencillamente, incorporar más empresas de renta de vehiculos, así como permitir a las empresas que aún no cuenten con alguna página web de su empresa, poder incluir sus vehículos en nuestra base de datos. Como otra mejora a considerar sería bueno incluir algún sistema de notificación cuando cambian los precios de los vehículos y/o se agregan nuevos.

Otra extensión que se le puede realizar al prototipo presentado es la adición de más filtros. El proyecto se basa en la atracción de los filtros, es por esto que se cree que se podría profundizar más en este aspecto. Por ejemplo agregar la localización de las empresas y poder filtrar por su ubicación, entre otros filtros que podrían agregarse.

## Referencias

- [1] <http://www.multicar.com.uy/>
- [2] <http://www.plusrentacar.com.uy/>
- [3] <http://jaunt-api.com/>