2014

Proyecto Final

Recuperación de Información y Recomendaciones en la Web

Ana Carrato - C.I: 4247054-2
Ana Menéndez - C.I: 4373979-9

Profesor: Libertad Tansini
05/12/2014

Contenido

1. INTRODUCCIÓN

El presente informe es el resultado del proyecto realizado por nuestro Grupo en el marco de tareas previstas en la materia electiva Recuperación de Información y Recomendaciones en la Web, edición 2014, de la carrera de Ingeniería en Computación, del Instituto de Computación de la Facultad de Ingeniería.

En esta sección se presenta la motivación del estudio, sus objetivos y un resumen del proyecto.

En la siguiente sección se describe el desarrollo del mismo desde la arquitectura, las herramientas seleccionadas hasta la implementación de la aplicación.

Finalmente se encuentran las conclusiones del proyecto, las referencias utilizadas y en el Anexo se incluye resultados de las herramientas utilizadas.

1.1. Tema de estudio

Dentro de las propuestas presentadas por la docente, seleccionamos la extracción de información de la web de forma automatizada, dado lo útil que personalmente consideramos el hecho de poder centralizar en un único sitio lo que muchas veces lleva tiempo recorrer diferentes links de sitios similares como en el caso de buscar inmuebles, donde todas las páginas son relativamente similares.

Por lo que el tema de estudio fue Web Scraping, Análisis e Integración de datos de páginas de inmuebles, incluyendo sugerencias al usuario.

1.2. Objetivos del estudio

El objetivo del `proyecto fue aplicar Web Scraping a una selección de páginas de inmuebles y lograr la integración de las mismas reconociendo ventajas y desventajas de estos procesos, desarrollando luego una aplicación que centralice el acceso a la información de diferentes fuentes, obtenidos previamente los datos con web scraping.

1.3. Resumen

Se investigó de Web Scraping, su definición, ejemplos de uso, y herramientas que la proveen.

Convenimos es utilizar una accesible Extensión de Chrome y al ver diferentes ejemplos de aplicación de la técnica, decidimos enfocarnos en la extracción de datos de diferentes páginas de inmuebles (El Gallito y Mercado Libre) para exponer al usuario en un solo sitio.

Esto requirió un estudio de la estructura de las páginas a “scrapear” y la forma de presentación de información de cada una. Aquí nos encontramos con varios puntos a considerar en la utilización de la extracción automática.

Una vez extraídos los datos, fueron trabajados y analizados para luego integrarlos en una única base de datos.

Luego se desarrolló una aplicación web que logra ofrecer al usuario un sitio que centraliza la información de varias páginas brindando al usuario una forma cómoda y rápida de acceder a la información.

A lo largo de este informe se encuentran los detalles de los puntos presentes en este resumen y una conclusión final sobre la experiencia.

2. DESARROLLO DEL PROYECTO

2.1. Web Scraping

Web Scraping [1] es una técnica utilizada mediante programas de software para extraer información de sitios web. Usualmente, estos programas simulan la navegación de un humano en la World Wide Web ya sea utilizando el protocolo HTTP manualmente, o incrustando un navegador en una aplicación.

El web scraping está muy relacionado con la indexación de la web, la cual indexa la información de la web utilizando un robot y es una técnica universal adoptada por la mayoría de los motores de búsqueda. Sin embargo, el web scraping se enfoca más en la transformación de datos sin estructura en la web (como el formato HTML) en datos estructurados que pueden ser almacenados y analizados en una base de datos central, en una hoja de cálculo o en alguna otra fuente de almacenamiento.

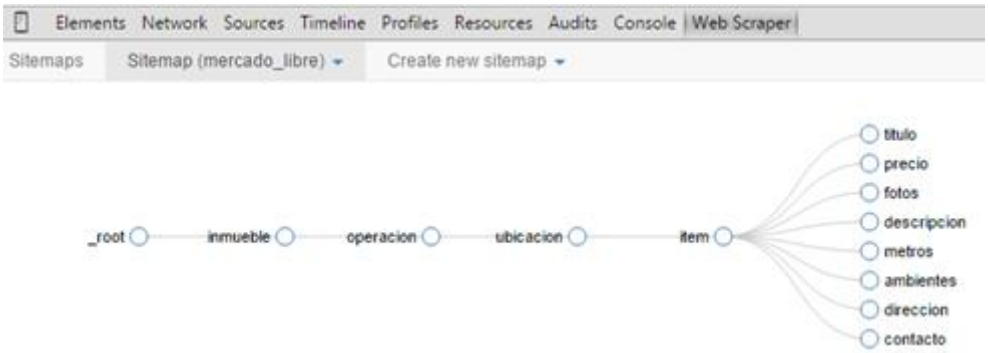
El término web scraping también está relacionado con la automatización de tareas en la Web, la cual simula la navegación de un humano utilizando un software de computadora. Alguno de los usos del web scraping son la comparación de precios en tiendas, la monitorización de datos relacionados con el clima de cierta región, la detección de cambios en sitios webs y la integración de datos en sitios webs.

Hay diversas herramientas como por ejemplo Apache Camel que es un motor de ruteo que se basa en reglas provistas por implementación en Java.
Para este proyecto se escogió la Extensión de Chrome Web Scraper que no requiere programación en lenguaje particular y es muy sencilla de aplicar.

2.2. Extensión de Chrome – Web Scraper

Web Scraper[2] es una extensión gratuita para el navegador Chrome construido para la extracción de datos de páginas web. Esta extensión requiere que el usuario cree un plan de ruteo donde indica cómo se debe navegar en el sitio web (mapa del sitio) y cuáles son los datos que deben extraerse. Una vez obtenido los datos pueden exportarse como CSV.

La aplicación de Web Scraper se hizo sobre páginas de inmuebles, donde la estructura de datos a obtener se refleja en el siguiente grafo.



La primera selección se hace sobre tipos de inmuebles (Casas, Apartamentos, etc.)
La segunda selección se hace sobre operación. (Venta, Alquiler, etc.)
La tercera selección se hace sobre ubicación. (Montevideo, Canelones, etc.)
Y finalmente se recaba la información siendo ítem el inmueble encontrado.

El ruteo se diseña sobre cada página elegida, por lo que va a haber un grafo por cada una.
Para este proyecto se seleccionó El Gallito Luis y Mercado Libre, pero puede extenderse a las páginas que se desee.
Si bien el ejemplo es el grafo sobre Mercado Libre, es la estructura base que buscamos obtener también en otras páginas.

Previamente debe estudiarse la estructura de la página a "scrapear" y al diseñar, debe indicarse cuál es la secuencia de links a elegir, por tal motivo el grafo queda dependiente de la presentación de la página web elegida.
Una vez diseñado el grafo, se le indica a la aplicación la acción 'Scrape' y en ese momento se abre una nueva ventana simulando la navegación humana extrayendo la información del ítem indicada.
La extracción consta en obtener los datos de títulos, etiquetas, links, etc. del código de la página según lo indicado en el ruteo y este es guardado para su devolución posterior.
Dependiendo de la cantidad de rutas que se naveguen esto puede demorar algunas horas.

Finalizado el ruteo puede exportarse la información recabada en archivo csv que si bien es un archivo plano, puede abrirse con Excel quedando identificadas las columnas según la ruta y los links que se fueron navegando.

- ✓ VENTAJAS:

Es relativamente sencilla la forma de diseñar el ruteo y poner a correr la extracción.
Una vez diseñada la ruta puede ejecutarse las veces que sea necesaria para obtener la información.

- ✓ DESVENTAJAS:

El ruteo queda dependiente del diseño de la página al momento de crearlo. Por lo que si la página fuente realiza alguna modificación en su estructura se deberá actualizar el grafo.
Algunos diseños con JavaScript o AJAX no pueden obtenerse fácilmente.

2.3. Páginas seleccionadas

Luego de seleccionar la página que se trabajará, es necesario estudiar bien la estructura de la misma y planificar como conviene extraer la información.

Entre las páginas que publican inmuebles en Uruguay, elegimos el Gallito Luis y Mercado Libre dado que son paginas conocidas, que presentan una estructura similar y contienen mucha información útil para el proyecto.

2.3.1. El Gallito Luis

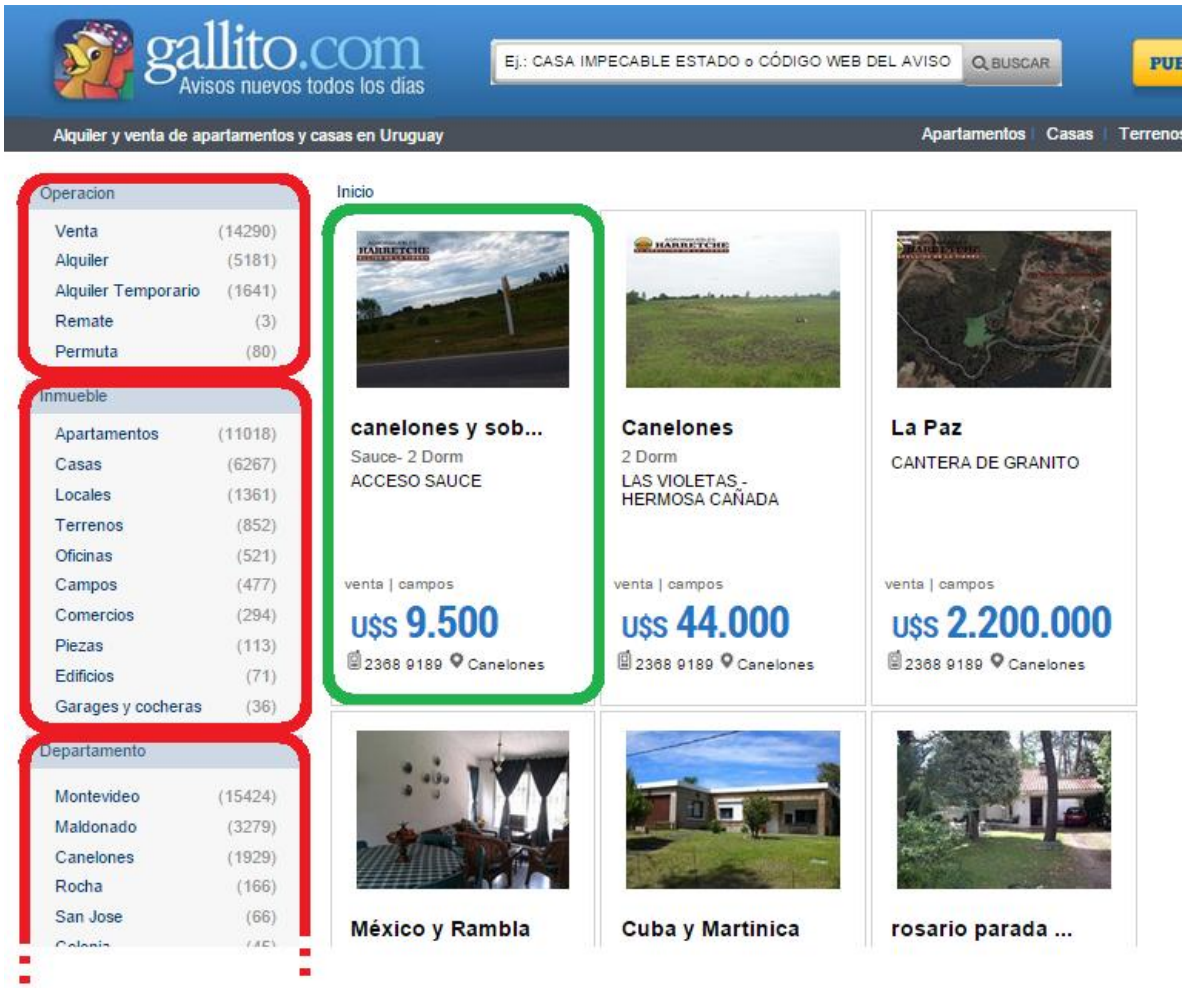
La página web utilizada de "El gallito Luis" http://www.gallito.com.uy/inmuebles , ofrece diferentes filtros para navegar en la oferta de diversos inmuebles en Uruguay.

Sobre el margen izquierdo se encuentra una variedad de filtros posibles que el usuario puede combinar para el filtrado de las búsquedas.

Los filtros que utilizamos son:

- Operación
- Inmueble
- Departamento

Existen en esta página otros filtros que se combinan dependiendo del tipo de inmueble, por ejemplo Dormitorios, Baños, Precio, Metraje Total, etc.



2.3.2. Mercado Libre

La pagina web utilizada de “Mercado Libre” http://inmuebles.mercadolibre.com.uy , ofrece diferentes filtros para navegar en la oferta de diversos inmuebles en Uruguay.

Sobre el margen izquierdo se encuentra una variedad de filtros posibles que el usuario puede combinar para el filtrado de las búsquedas.

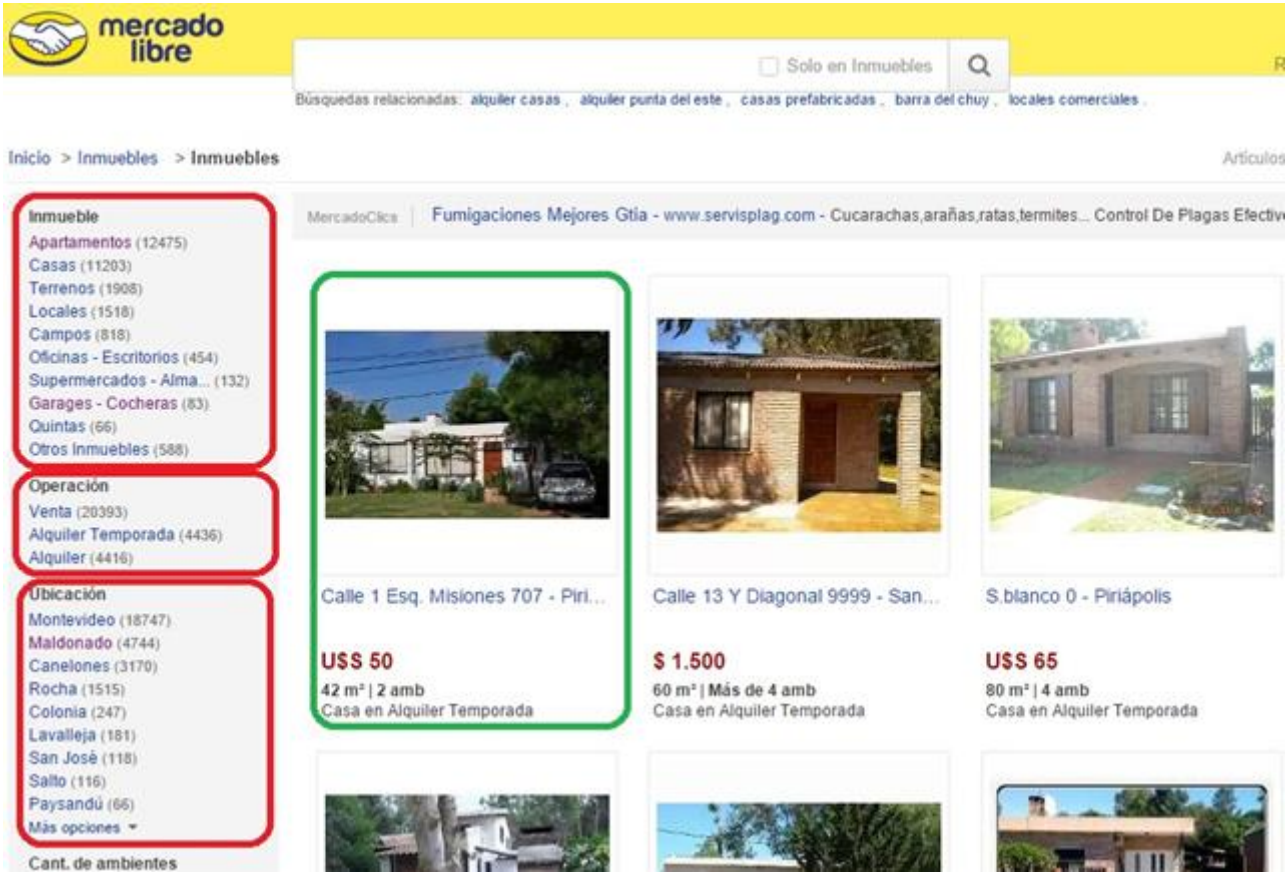
Los filtros que utilizamos son:

Operación

Inmueble

Departamento

Existen en esta página otros filtros que se combinan dependiendo del tipo de inmueble, por ejemplo Metraje Total, Rango de Precios, Cochera/s, etc.



2.4. Análisis de Datos

Una vez corrido el Scrape sobre las páginas, los datos son exportados a archivos .csv

Los datos necesitan de cierta depuración y normalización antes de agregarlos a la base de datos, ya que si bien los datos escogidos para ilustrar nuestro proyecto son puntuales, debido a que las publicaciones las realizan los propios usuarios, estamos dependientes a las interpretaciones que realice al ingresar un inmueble, lo cual lleva a que sea difícil la integración de diversas páginas.

2.4.1. Problemas encontrados

- En Gallito se habla de dormitorios y en Mercado Libre se habla de ambientes.
- Los datos de metraje si bien cuando se carga un inmueble existe un campo para escribir el valor, nos encontramos con que algunos le han agregado también la letra m o m2, etc. Esto nos dificulto al obtener un campo numérico único para guardar el valor de cantidad de metros como para poder utilizar como filtro.
- Puede ocurrir que el mismo inmueble esté publicado en ambas páginas, pero requiere de mayor trabajo poder identificar que se refieren al mismo. Por lo cual no sera tenido en cuenta en nuestra aplicación.
- En mercado libre algunos no tienen la dirección y en su lugar se encuentra el horario en que se puede llamar al contacto, lo que genera al scapear que se guarde un dato semánticamente incorrecto.
- En ambas páginas en ubicación no aparecen los 19 departamentos sino que se menciona algunos y luego ponen "Más Opciones", que seleccionando despliega otros departamentos, esto dificulta el scapear porque cuando el ruteo selecciona este campo toma "Más Opciones" como una ubicación más.
- Otros problemas son los típicos datos mal ingresados por ejemplo::
  - En el título ponen toda la descripción sobre el inmueble.
  - Ponen precios falsos del tipo 1 o 111111
  - Ingresos de metros incorrectos 1m2.
- Si bien en ambas páginas se dispone de más de una foto, con la herramienta solo pudimos guardar una única foto.
- En mercado libre el teléfono del contacto es un link dinámico por lo cual no fue posible obtenerlo con el scraper.

Por todos estos motivos es que consideramos que la calidad de datos debería analizarse cuidadosamente en caso de continuar este proyecto.

2.4.2. ETL de datos

Una vez obtenidos los archivos .csv con los datos extraídos de las páginas seleccionadas, se utilizó la herramienta Data Integration - Pentaho[3] para la realización del proceso ETL[4].

Con esta herramienta nos fue posible diseñar una transformación con una serie de pasos sobre los archivos para realizar una depuración básica.

Una vez realizado el diseño de la secuencia de procesamiento, esta transformación se guarda permitiendo ser aplicado en las sucesivas extracciones de los archivos. Esto da la ventaja de una vez estudiada la forma de los datos originales y a donde queremos llegar, se conserva el procesamiento.

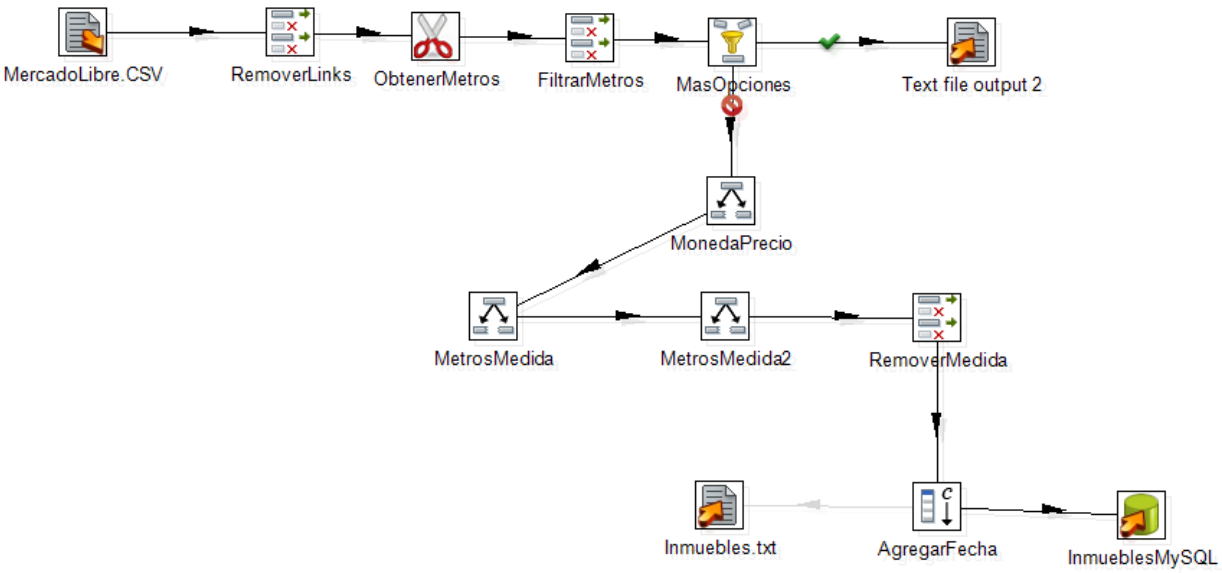
Similar al diseño del scraper, este proceso queda dependiente del formato del archivo original, por lo cual si se cambia la extracción y esto afecta al archivo generado hay que revisar el procesamiento planificado con esta herramienta.

De los pasos aplicados destacamos las siguientes:

- Eliminación de columnas con links que no se van a utilizar. En la extracción se guarda cada link recorrido, del cual en la mayoría de los casos nos interesa el valor seleccionado y no el link que contiene. El unico link que se deja es el link al inmueble y el de la foto.
- Sobre el campo recabado de metraje, el mismo contenía datos por ejemplo (100 m2 cubiertos), por lo que para extraer la parte numérica de ese campo primero se descartó la parte final de m2 cubiertos y luego se trato de dejar solo el número. Aqui los mas complicados son los que no cumplen el estándar y agregaron por ejemplo 100m m2 cubierto, y otros casos.
- El precio del inmueble en ambas páginas viene de la forma “Moneda Precio” por ejemplo U$S 50000 o $ 12000, por tal motivo se creó una columna moneda y se separó el dato que venía en el precio dejando separada la moneda del valor numérico.
- También se agrega un campo fecha para guardar la fecha de extracción de la muestra de datos.

Una vez finalizado el proceso de transformación sobre el archivo original, la salida la volcamos directamente sobre la base de datos.

El siguiente es el esquema del proceso diseñado para trabajar sobre el archivo scrapeado de Mercado Libre.



2.5. Integración de Datos

Para la aplicación se diseñó una base de datos en MySql que contiene una única tabla “inmuebles” con toda la información recabada de los inmuebles.

Para la clave primaria de esta tabla se agregó un atributo autogenerado “id”.
El resto de los campos se corresponden con los datos obtenidos de las páginas y la “fecha_extraccion” es la agregada en el proceso de transformación de los datos para controlar a qué día corresponde la extracción de ese dato.

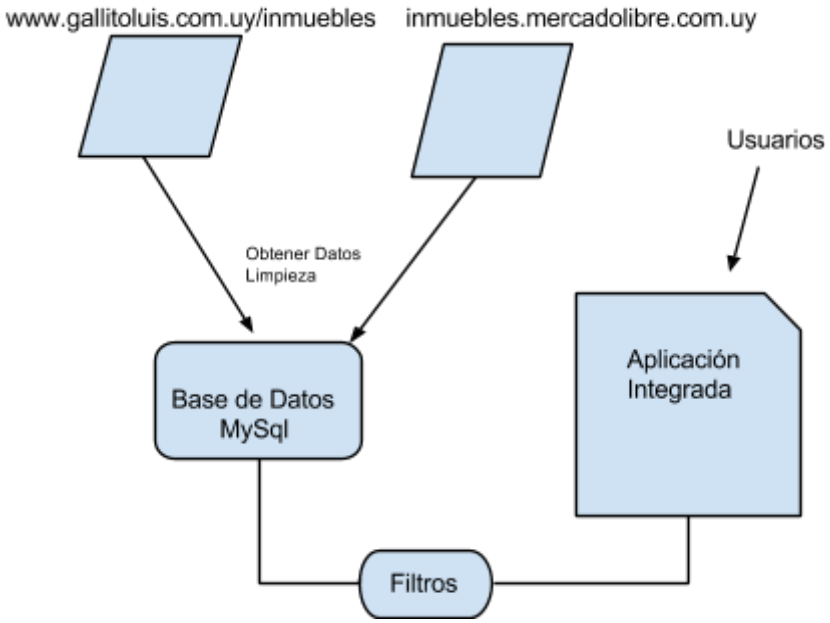
En la siguiente imagen se muestran los campos que contienen la tabla “inmuebles”

La carga de los datos en la base de datos se realizó mediante el proceso etl con la transformación correspondiente a la página extraída.

2.6. Arquitectura de la Aplicación

Con la base de datos diseñada, los datos integrados y cargados en la misma, se comienza a desarrollar la aplicación.

El esquema del proyecto es el siguiente:



Como se ha mencionado en los puntos anteriores, partimos de dos sitios conocidos de inmuebles, a los cuales realizamos web scraping para obtener la información de ellos.

Una vez que contamos con los archivos de extracción de datos, se aplica la transformación limpiando y formateando los mismos volcándolos a la base de datos MySql.

La aplicación web construida para explotar dicha información, permitirá que un usuario realice búsquedas de inmuebles ingresando en un único sitio pudiendo utilizar filtros para reducir y optimizar la búsqueda deseada.

NOTA: *La aplicación construida para este proyecto no está alojada en ningún servidor externo por lo que se ejecuta desde el localhost.*

2.7. Herramientas utilizadas

En el siguiente cuadro se resumen las herramientas elegidas para el desarrollo de todo el proyecto y sus versiones.

Scraper	Extensión de Chrome Web Scraper
Fix de datos	Data Integration - Pentaho 4.4.0
DBMS	MySQL (Version: mysql-installer-community-5.6.17.0)
IDE	Eclipse Luna (Version: eclipse-jee-luna-SR1)
SERVER	JBoss 7.1.1
TOOLS	Java 1.7 (Version: jdk1.7.0_71)
TOOLS	MAVEN (Version: apache-maven-3.1.1
SVN	Tortoise (Versión: tortoisehg-3.1.2)

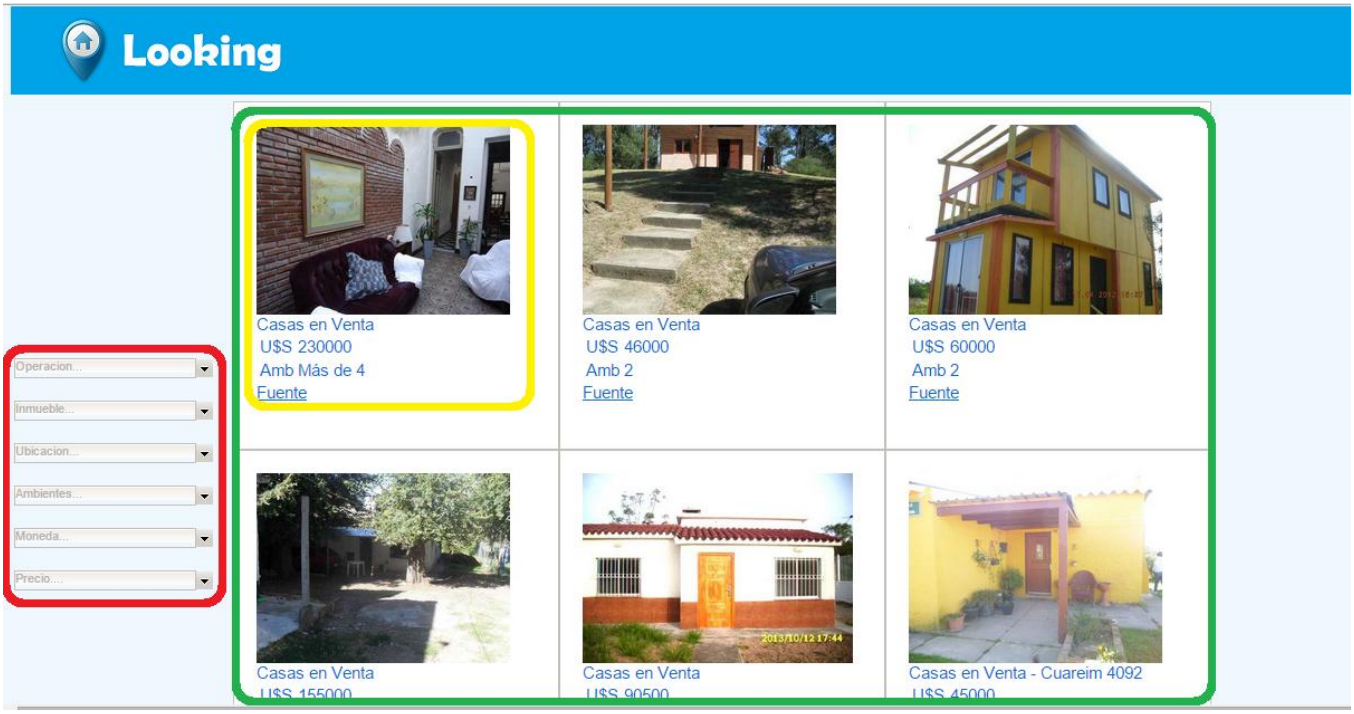
2.8. Aplicación Web

El objetivo final de este proyecto fue desarrollar una aplicación que a la visión del usuario centralice el acceso a la información de diferentes páginas de inmuebles.

Según experiencias personales del equipo, la tarea de buscar inmuebles por internet resulta complicado porque, si bien hay varios buscadores que centralizan a varias inmobiliarias, como www.casas.com.uy, www.apartamentos.com.uy, www.buscandocasa.com.uy estas muchas veces no bastan y de todas formas se termina manejando una lista importante de páginas web que debe ingresarse cada vez.

2.8.1. Web Looking

Conociendo la necesidad del usuario se desarrolló la página web llamada ***Looking*** que consta en principio de un único link de ingreso: localhost:8080/tradingProperties-web/



Al ingresar en la página, en la pantalla se muestran sobre el lado izquierdo diferentes filtros a aplicar en la búsqueda, ellos son:

- Operación [Venta, Alquiler]
- Inmueble [Casas, Apartamentos]
- Ubicación
- Ambientes
- Moneda [$, U$S]
- Precio [ascendente, descendente]

Las opciones de cada select, se cargan con los diferentes valores de los campos correspondientes en la tabla, a excepción del campo precio que ofrece un ordenamiento ascendente o descendente según el valor ese campo.

En la zona derecha de la pantalla se muestra una grilla de 6 items de acuerdo a la combinación de filtros elegida y mostrando la paginación al pie.

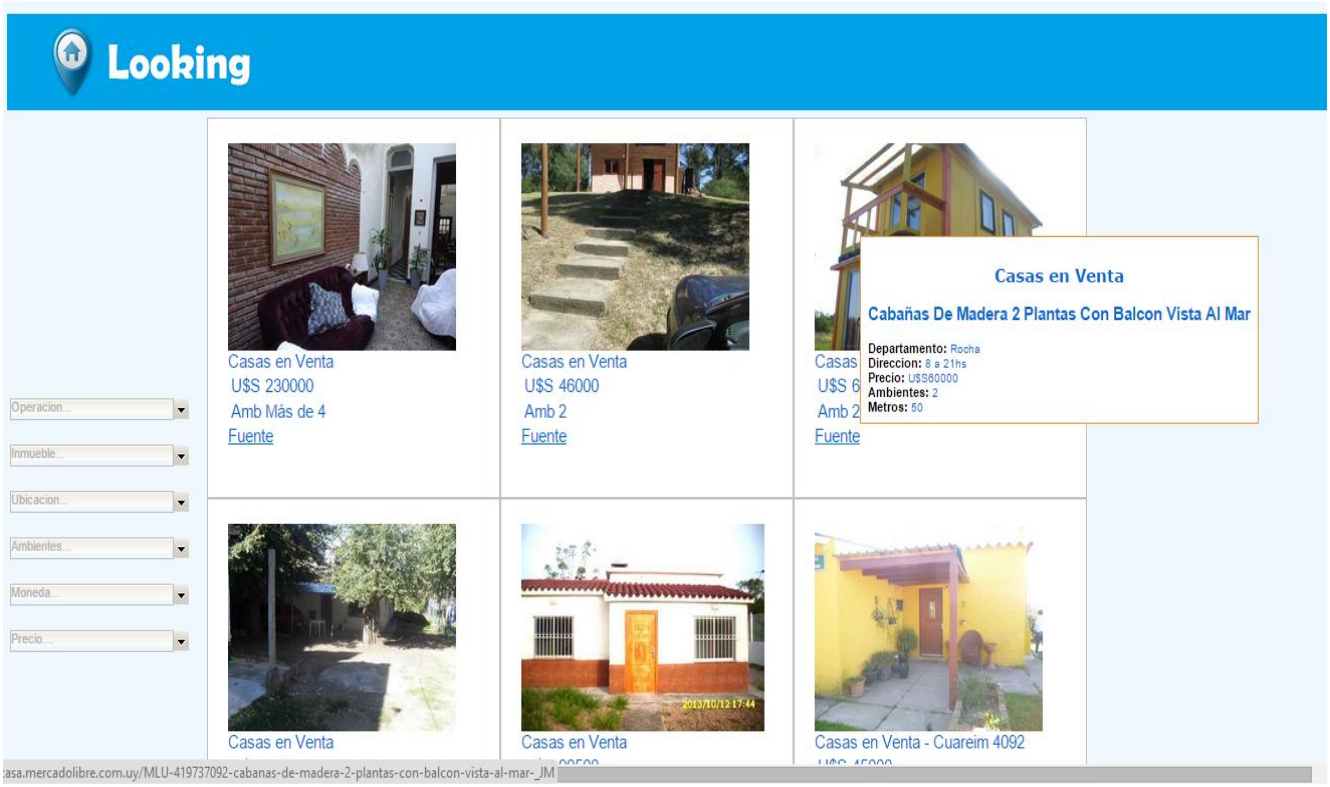
La primera vez que se ingresa a la web, la selección es realizada sin ningún filtro, por lo que trae 6 inmuebles de cualquier categoría.

A medida que van aplicándose los filtros de búsqueda, se va reduciendo la cantidad de items devueltos y se refresca la grilla de inmuebles a mostrar.

Sobre el contenido de la grilla, cada item de inmueble muestra la imagen obtenida del mismo.

Luego se muestra el título que acompaña el anuncio, la moneda, el precio, la cantidad de ambientes y un link Fuente que llevará al usuario al link original del anuncio pudiendo acceder a la información original del inmueble.

Al pararse con el mouse sobre cada item de la grilla, se muestra en formato tooltip toda la información que se posee del mismo, es decir, la ubicación, la dirección, el precio, los ambientes y el metraje.



Finalmente al pie de la página se muestra la sección ***Lo más buscado…*** que muestra 5 inmuebles de la base de datos.

La primera vez que se ingresa a la página, se muestra los primeros 5 inmuebles de la base de datos y a medida que se van aplicando los diferentes filtros se agrega a esta lista el primer inmueble que no esté en la lista de la selección que genera el filtrado.

2.8.2. Desarrollo de la aplicación

Para el desarrollo de la aplicación estuvimos investigando en principio diferentes framework ofrecidos en el mercado.

Dado que no encontramos alguno que satisfaga nuestro interés debido a que todos estaban diseñados a los negocios de e-commerce, por lo cual decidimos desarrollar nuestra propia web.

Para el desarrollo elegimos el lenguaje java generando un proyecto en Eclipse y utilizando el servidor JBoss.

La programación se centraliza principalmente en el archivo index.xhtml.

Investigamos varios temas referentes a html y css, en particular compatibilidad con xhtml [5] tanto para la programación en general como a los aspectos de diseño de la web.

También buscamos información y ejemplos sobre cómo utilizar componentes provistos de RichFaces[6] que fuimos agregando en nuestro diseño.

Los principales archivos del proyecto son:

- orm.xml
- SearchBean.java
- inmueblesDAO.java
- Inmueble.java
- index.xhtml

La aplicación se corre en forma local, por lo que debe tenerse creada y levantada la base de datos en el equipo que desee verse la página en funcionamiento.

Luego puede accederse desde cualquier navegador con la url:
localhost:8080/tradingProperties-web/

3. CONCLUSIONES

Finalizado el proyecto consideramos que hemos cumplido con los objetivos propuestos sobre la utilización de Web Scraping, logrando la integración de páginas de inmuebles e implementando una aplicación que centraliza los diferentes sitios.

Nos pareció interesante que a partir de lo antes mencionado logramos brindar al usuario comodidad y rápido acceso a la información, sin tener que navegar por varios sitios.

La mayor motivación de este proyecto fue centralizar la información de inmuebles, debido a las experiencias personales.

El hecho de realizar las búsquedas en los diferentes sitios resultan engorrosas debido a la gran cantidad de información que implica que al acceder a cada página es necesario aplicar los mismos filtros cada vez. También día a día tener que entrar a estas y cotejar si la información ha variado.

En cuanto a la recuperación de la información en la web pudimos ver que es una tarea interesante, pero requiere un análisis profundo debido a que la calidad de los datos extraídos puede impactar en la información que se quiere ofrecer.

Debido a las limitaciones del alcance de este proyecto no pudimos aplicar las recomendaciones al usuario que en principio nos hubiera gustado implementar para obtener una aplicación centralizada y robusta con funcionalidades que no se encuentran en las páginas ya existentes, por lo cual las dejamos planteadas como mejoras a futuro.

3.1. Mejoras futuras para el proyecto

Algunos de los puntos que consideramos podrían fortalecer el uso de esta web y la explotación de la información en cuanto a la continuación de este proyecto, son los siguientes:

- ✓ Registro de usuario. Se podría agregar un registro de usuarios, de forma de que cuando un usuario se loguee en la web, se pueda guardar información sobre el mismo, por ejemplo guardar preferencias en los filtros a aplicar, que permita que cada vez que ingrese al sitio figuren seleccionados los últimos filtros elegidos.
- ✓ Mejoras en las sugerencias a los usuarios. El logueo de usuarios también podría permitir guardar algún historial de las búsquedas realizadas y de forma de poder mostrar alguna sugerencia que pueda ayudar sobre sus preferencias.
- ✓ Enviar por mail sugerencias a los usuarios. Otra de las mejoras con el logueo de usuarios es poder generar algún resumen de publicaciones de nuevas propiedades que pueda interesarle de acuerdo a sus filtros.
- ✓ Agregar más opciones en operación e inmuebles. Dada la estructura de la BD y la web, es muy sencillo poder agregar más opciones de operaciones por ejemplo Remates ó de inmuebles por ejemplo Locales comerciales.
- ✓ Calidad de Datos. Mencionados los inconvenientes tenidos respecto a los datos, sería clave invertir más tiempo en los mismos. También debería agregarse lógica que permita

identificar publicaciones repetidas, es decir, que se identifique cuando un inmueble es publicado por diferentes fuentes, así como también detectarse inconsistencias en cuanto a que el mismo inmueble puede ser publicado en direcciones diferentes o pueden poner precios diferentes y tomar un criterio según el caso de que datos se guarda.

- ✓ Actualización de Datos. Para este proyecto se trabajó con solo la extracción de un día puntual, pero es necesario realizar la extracción de datos con frecuencia de forma de poder reconocer cuando aparecen nuevas publicaciones y las que siguen vigentes se actualicen si tuvieron algún cambio.
- ✓ Vigencia de los datos. Debería verse de qué forma se maneja la vigencia de la información, es decir hasta cuando se ofrece la información y cuando se considera obsoleta. En la BD siempre se cuenta con la fecha de extracción o actualización de los ítems.
- ✓ Mejores Filtros. Trabajando un poco más en la depuración de los datos extraído, podrían agregarse más filtros y/o mejorar los existentes, por ejemplo rango de precios, metraje, barrios, etc.
- ✓ Más fotos de los inmuebles. Si bien por simplicidad y limitaciones de la herramienta seleccionada estamos mostrando solo una foto, la mayoría de los sitios guardan más de una foto del inmueble que sería interesante poder ofrecer al usuario en esta web.
- ✓ Utilizar más páginas fuentes. Debería realizarse el trabajo de la fase inicial para otras páginas de forma de poder obtener los datos a volcar en la BD, pues una vez en la BD no se requiere cambios en la web para utilizar esta información con las precauciones mencionadas en los puntos anteriores.
- ✓ Mejoras en el diseño de la página. De continuarse con la programación consideramos que se debería dedicarle más tiempo al diseño gráfico de la página de forma de hacerla más amigable e interesante para los usuarios.

4. REFERENCIAS BIBLIOGRÁFICAS

[1]Web Scraping
http://en.wikipedia.org/wiki/Web_scraping

[2] Extensión de Chrome
http://es.schoolofdata.org/introduccion-a-la-extraccion-de-datos-de-sitios-web-scraping/
http://schoolofdata.org/handbook/recipes/scraper-extension-for-chrome/
http://webscraper.io/

[3] ETL
http://es.wikipedia.org/wiki/Extract,_transform_and_load

[4] Pentaho - Spoon
http://wiki.pentaho.com/display/EAIes/Manual+del+Usuario+de+Spoon

[5] Tutorial Html, CSS, Xhtml
http://www.w3schools.com/

[6] RichFaces
http://showcase.richfaces.org/

5. ANEXO

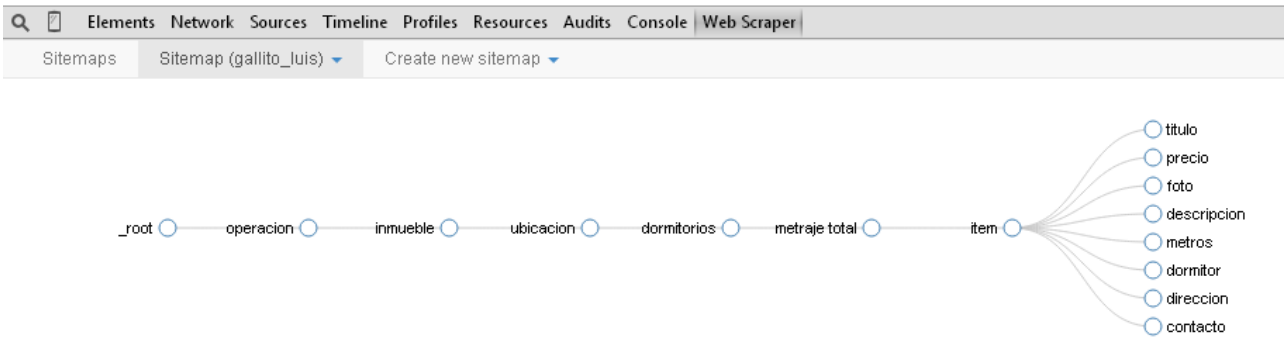
5.1. SiteMap Mercado libre


```
{"startUrl":"http://inmuebles.mercadolibre.com.uy/","selectors":[{"parentSelectors":["_
root"],"type":"SelectorLink","multiple":true,"id":"inmueble","selector":"dl#Inmueble
a","delay":""},{"parentSelectors":["inmueble"],"type":"SelectorLink","multiple":true,"i
d":"operacion","selector":"dl#Operación
a","delay":""},{"parentSelectors":["operacion"],"type":"SelectorLink","multiple":true,"
id":"ubicacion","selector":"dl#Ubicación > dd > a, dl#Ubicación dd.qcat-filter
a","delay":""},{"parentSelectors":["ubicacion"],"type":"SelectorLink","multiple":true,"
id":"item","selector":"h2.list-view-item-title a:nth-of-
type(1)","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multiple":false
,"id":"titulo","selector":"h1","regex":"","delay":""},{"parentSelectors":["item"],"type
":"SelectorText","multiple":false,"id":"precio","selector":"article.price
strong","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorImage","mult
iple":false,"id":"fotos","selector":"div.first-image
img","downloadImage":false,"delay":""},{"parentSelectors":["item"],"type":"SelectorText
","multiple":false,"id":"descripcion","selector":"h2.tit-description
span","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multipl
e":false,"id":"metros","selector":"dd.prominent
span","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multipl
e":false,"id":"ambientes","selector":"ul.technical-details:nth-of-type(1) li:nth-of-
type(3)
strong","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multi
ple":false,"id":"direccion","selector":"dd:nth-of-type(3) span.seller-info-
box","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multiple
":false,"id":"contacto","selector":"dd.seller-phone span.seller-details-
box","regex":"","delay":""}],"_id":"mercadolibre"}
```

5.2. SiteMap Gallito Luis


```
{"selectors":[{"parentSelectors":["items"],"type":"SelectorText","multiple":false,"id":
"datos","selector":"span.thumb_datos","regex":"","delay":""},{"parentSelectors":["_root
"],"type":"SelectorLink","multiple":true,"id":"operacion","selector":"div#Div_Cont_Oper
acion.criterio_orden
a.items_ordenamiento_azul","delay":""},{"parentSelectors":["operacion"],"type":"Selecto
rLink","multiple":true,"id":"inmueble","selector":"div#Div_Cont_Operacion.criterio_orde
n div.items_ordenamiento_filtro_azul:nth-of-type(1)
a.items_ordenamiento_azul","delay":""},{"parentSelectors":["inmueble"],"type":"Selector
Link","multiple":true,"id":"ubicacion","selector":"div#Div_Cont_Inmueble.criterio_orden
div.items_ordenamiento_filtro_azul:nth-of-type(1)
a.items_ordenamiento_azul","delay":""},{"parentSelectors":["ubicacion"],"type":"Selecto
rLink","multiple":true,"id":"dormitorios","selector":"div.item_op_med >
div.items_ordenamiento_filtro_azul:nth-of-type(1)
a.items_ordenamiento_azul","delay":""},{"parentSelectors":["dormitorios"],"type":"Selec
torLink","multiple":true,"id":"item","selector":"div.contiene_grilla div div div
a","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multiple":false,"id":
"titulo","selector":"div.titulo
h1","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multiple"
:false,"id":"direccion","selector":"section.datos
h2","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multiple"
:false,"id":"descripcion","selector":"p.desc","regex":"","delay":""},{"parentSelectors"
:["item"],"type":"SelectorText","multiple":false,"id":"precio","selector":"p.precio","r
egex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","multiple":false
,"id":"metros","selector":"ul.cfx li:nth-of-
type(5)","regex":"","delay":""},{"parentSelectors":["item"],"type":"SelectorText","mult
iple":false,"id":"dorm","selector":"ul.cfx li:nth-of-
type(5)","regex":"","delay":""}],"startUrl":"http://www.gallito.com.uy/inmuebles","_id"
:"gallitoluis"}
```


5.3. Grafo Gallito Luis



5.4. Script Base de Datos

```
CREATE DATABASE `webir` /*!40100 DEFAULT CHARACTER SET utf8 */;
CREATE TABLE `inmuebles` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `operacion` varchar(100) NOT NULL,  `inmueble` varchar(100) NOT NULL,
  `ubicacion` varchar(100) NOT NULL,
  `titulo` varchar(500) NOT NULL,
  `direccion` varchar(500) DEFAULT NULL,
  `precio` int(255) DEFAULT NULL,
  `moneda` varchar(50) DEFAULT NULL,
  `ambientes` varchar(50) DEFAULT NULL,
  `dormitorios` varchar(50) DEFAULT NULL,
  `metros` varchar(50) DEFAULT NULL,
  `descripcion` varchar(500) DEFAULT NULL,
  `fotos` varchar(500) DEFAULT NULL,
  `link_original` varchar(500) NOT NULL,
  `fecha_extraccion` date NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=936 DEFAULT CHARSET=utf8;
```

5.5. ETL Gallito Luis