**INSTITUTO TECNOLÓGICO DE COSTA RICA**

**INGENIERÍA EN COMPUTACIÓN**

**Introducción al desarrollo de aplicaciones web**

**Investigación Web Scrapping**

**Efren Jiménez Delgado.**

**ELADIO CRUZ VÁSQUEZ**

**Marzo, 2016**

**Tabla de contenido**

[1. Resumen 3](#_Toc479016963)

[2. Introducción 3](#_Toc479016964)

[3. Metodología realizada 3](#_Toc479016965)

[4. Conclusiones 6](#_Toc479016966)

[5. Recomendaciones 7](#_Toc479016967)

[6. Bibliografía 7](#_Toc479016968)

# 1. Resumen

Este proyecto se tratará de explicar acerca de la metodología empleada para realizar el parseo de páginas web con la técnica Web Scraping para obtener diferentes datos de una o varios sitios web.

Para realizar esta actividad se cuentan con diferentes herramientas que se utilizaron como librerías y lenguajes de programación que se utilizaron para este trabajo de investigación.

# 2. Introducción

En esta investigación, se estará hablando acerca de los procedimientos que se emplearon en el Web Scraping, sobre todo el cómo manipular dicha información y los datos obtenidos.

Lo primero que hay que decir es que para realizar este trabajo se contaron con el lenguaje de programación Python, para extraer los datos con el uso de diferentes librerías, al igual que Postgress como base de datos para almacenar la información recolectada.

Como se dijo anteriormente la principal función de este proyecto es la extraer información de diferentes sitios web con el fin de contrastarla y así obtener algún beneficio que, haciendo manualmente esta actividad sería más complicado.

# 3. Metodología realizada

Para la realización de este proyecto de Web Scraping se extrajo los datos de dos sitios web de apuestas, el primero de ellos es Bet365 y el segundo Wannabet con el fin de comparar los datos de apuestas de la liga española para ver, primeramente, a quien se le está dando más porcentaje de triunfo, de los partidos y también para saber en qué página se está pagando más por la apuesta. Para esto se utilizará el lenguaje de programación Python utilizando dos librerías principales como BeatifulSoap y también selenium. Para la base de datos se cuenta con una conexión desde Python para crear las tablas y realizar las consultas a Postgress.

Cabe destacar que lo primero que se debe realizar es la instalación de las librerías de Python psycopg2 (para conectar con Postgress), selenium y beatifulsoap además de Postgress. Luego, debemos crear una base de datos, en este caso con el nombre “scrapping”. Una vez tengamos esto creado, debemos ejecutar el archivo de Python llamado database.py y luego podremos ejecutar el Web Crawler.py que se encuentran en la carpeta.

A continuación, se adjunta un diagrama de la metodología a utilizar, que fue explicado en el documento anterior:

Comparación de datos

Almacenamiento en base de datos

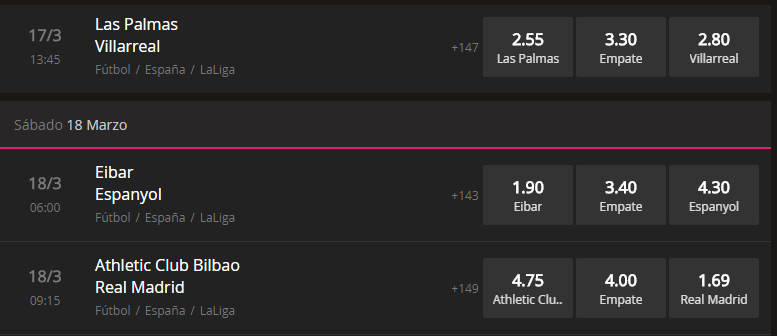
Lectura de las páginas web

Extracción de los partidos con sus porcentajes de apuestas

Mostrar el resultado final

Esta extracción de datos va a servir para comparar en cuál de las dos páginas es mejor apostar y así obtener una mejor ganancia, también, servirá para llevar una estadística de a qué equipo se le da más posibilidades de triunfo o empate.

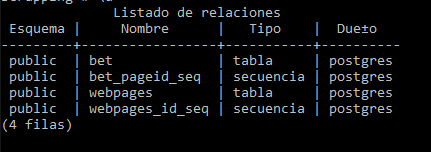
En esta toma de pantalla de página bet365, se observa la lista de elementos que se desea obtener (fecha del partido, hora, equipos y los porcentajes de apuesta para ambos equipos o empate) además posee un link de estadísticas que puede ser aprovechado después.

En esta otra imagen se puede apreciar los datos de las apuestas que son similares a los de bet365, pero se ve que las ganancias son diferentes, por lo que pueden llegar a contrastarse entre ellos y saber si es seguro apostar por 1 equipo o por el empate y en cuál de las dos casas de apuestas es mejor.

En resumen, se desea obtener los siguientes atributos, utilizando solamente las clases donde se encuentra el dato:

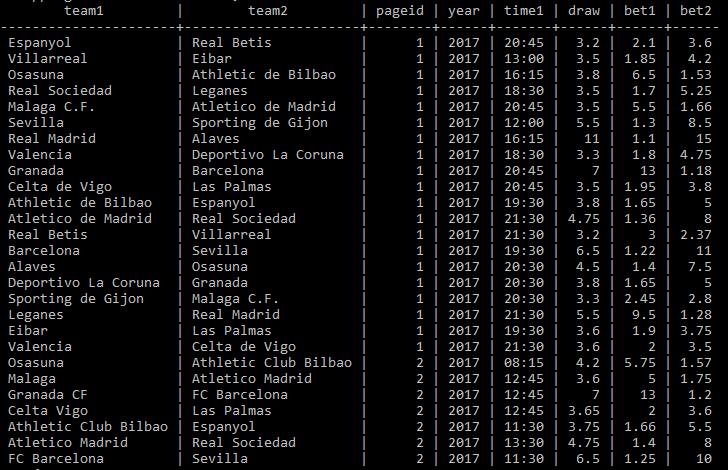
* **Equipo1 y Equipo 2:** Obtenidos en bet365 con la clase sl-CouponParticipantWithBookCloses\_Name o sl-CouponParticipantWithBookCloses sl-CouponParticipantIPPGBase sl-MarketCouponAdvancedBase\_LastChild y en wanabet con la claseKambiBC-event-participants\_\_name.
* **Año del partido:** Obtenido desde Python con las funciones para la fecha actual.
* **Hora del partido:** En wanabet se obtiene con la clase KambiBC-event-item\_\_start-time—time y en bet365 con la clase sl-CouponParticipantWithBookCloses\_BookCloses
* **Apuesta Equipo1, Apuesta Equipo1, Empate:** En bet365 se obtienen con la clase gl-ParticipantOddsOnly\_Odds y en wanabet con KambiBC-mod-outcome\_\_odds
* **Página donde se obtiene el dato:** Bet365, Wanabet.

Una vez hayamos ejecutado lo que anteriormente se mencionó, tendremos en la base de datos Postgress lo siguiente:



Dos tablas, la primera llamada “bet” es donde se guardarán los datos obtenidos en las páginas web de apuestas y también webpages que son las páginas web a las que le vamos a realizar el scrapping.

También cuando se ejecute correctamente el web crawler, en la base de datos la información se guardará de la siguiente manera:



# 4. Conclusiones

Para concluir se puede mencionar lo mismo que el documento anterior:

* Se debe indagar anteriormente a realizar el trabajo, los datos sobre la legalidad que tenga los sitios web, esto con el fin de no llegar a cometer algún delito grave.
* Se debería de enseñar esta práctica, ya que se puede lograr grandes beneficios al aplicar correctamente esta técnica.

# 5. Recomendaciones

* Se debe de investigar los términos legales de los sitios web para no cometer ningún delito.
* Se debería de indagar los mejores lenguajes de programación que permitan desarrollar una herramienta útil.

# 6. Bibliografía

Carey, H. J., & Manic, M. (2016). HTML web content extraction using paragraph tags. *2016 IEEE 25th International Symposium on Industrial Electronics (ISIE)*, 1099-1105.

Mahto, D. K., & Singh, L. (2016). A dive into Web Scraper world. *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 689-693.

Malik, S. K., & Rizvi, S. (2011). Information Extraction Using Web Usage Mining, Web Scrapping and Semantic Annotation. *2011 International Conference on Computational Intelligence and Communication Networks*, 465-469.