

Guión TFG

Diapositiva 1. Título.

Muchas gracias a todos los miembros del tribunal por su atención, ahora voy a pasar a presentar mi trabajo de fin de grado [Decir el nombre si no se ha dicho].

Diapositiva 2. Índice.

Esta es la estructura de la presentación... Comenzamos con la introducción.

- Introducción
- Trabajos relacionados
- Propuesta
- Solución
- Demo
- Conclusiones

Diapositiva 3. Introducción.

Es normal que haya toma de decisiones en sistemas de alojamientos arreglados como es el caso de la ocupación hotelera, en la que sí se tiene información. Sin embargo, de los destinos turísticos de alojamientos no arreglados como es el caso de Airbnb, no se tiene información.

El proyecto tiene como objetivo el análisis y toma de decisiones sobre los alojamientos registrados en la página de Airbnb. Partiendo de esta base, se elabora una solución de inteligencia de negocio para cumplir dicho objetivo.

Diapositiva 4. Trabajos relacionados.

En el mercado actual existen soluciones que ofrecen una gestión de los datos sobre la página de Airbnb. Estas soluciones ofrecen funcionalidades muy útiles, pero que se mejoran con el proyecto propuesto.

Por un lado, tenemos Airdna que nos brinda un cuadro de mando de la ciudad seleccionada. Además, presenta un modelo de negocio en el que sus funcionalidades gratuitas son muy limitadas y para su expansión se debe abonar una cantidad mensual. Al tener un cuadro de mando preestablecido e invariable nos provee de un análisis limitado.

Por otro lado, Inside Airbnb nos ofrece poca variedad de ciudades, estas son fijas y los datos que nos ofrece son ficheros planos, como por ejemplo CSV's.

Diapositiva 5. Propuesta.

¿Qué se propone?

- Una gran variedad de opciones de análisis gracias a las herramientas empleadas tanto para el backend como para el frontend, lo que hace a la propuesta una herramienta de BI muy versátil.
- Se proponen indicadores personalizados, confeccionados a partir de los cálculos entre los datos de los distintos atributos de alojamientos y sus respectivos calendarios. **Los datos de los alojamientos** contienen campos como el **id** de alojamiento, su **número de habitaciones y baños** y **latitud-longitud**, entre otros. Por otro lado, los calendarios ofrecen por cada día la disponibilidad y el precio del alojamiento.
- En último lugar, es posible obtener información de cualquier ciudad al alcance de Airbnb, esto es porque la ciudad va parametrizada en la consulta que se realiza contra el servidor.

Diapositiva 6. Solución.

La solución planteada se puede dividir en dos partes:

Extracción de datos y gestión de los datos.

En la primera se plantea la extracción de los datos de la página web de Airbnb todo ello programado con Python. Concretamente se han trabajado dos spiders o bots de extracción de datos, uno para los alojamientos y el otro para los calendarios. El framework Scrapy ha sido el encargado de gestionar la estructura del código y sus librerías. Los datos extraídos se almacenan en ficheros json.

Se ha utilizado el Middleware Rotated proxies para evadir el baneo por parte del servidor. Este ofrece una rotación de una lista de IPs, con este middleware Scrapy actuará bajo una IP distinta cada vez que la extracción falle.

Una vez obtenidos los datos, se gestionan con el entorno de MongoDB de tipo **NoSQL**, en el que se encuentran:

El servidor de la base de datos que viene junto a una consola para realizar transformaciones de los datos,

el cliente para visualizarlos se llama Compass, en este cliente también se pueden realizar algunas transformaciones, pero por motivos de eficiencia se recomienda realizarlas en la consola,

En último lugar, MongoDB Charts se utiliza **para confeccionar los gráficos de los datos tratados y agregados.**

Diapositiva 7. Demo.

Ahora se dará paso a la Demo, en la que se verá el entorno de desarrollo y de producción.

Diapositiva 8. Conclusiones.

Las conclusiones se han dividido en 3 grupos:

Conclusiones referentes a la Tecnología:

- Se observa una clara sinergia entre Scrapy y el entorno de MongoDB por el uso conjunto de jsons.
- Scrapy es una muy buena opción **open source** para el **web scraping programable** ya que nos ha ofrecido muchas librerías para la gestión de los datos. (como deltatime, diccionarios, a parte de su fichero de configuración)

Trabajos futuros:

- El uso de crontab para la programación de tareas como la inyección de datos sería algo imprescindible para el avance del proyecto.
- Así como la automatización de la extracción de la lista de proxies.

Resultado:

- Entre las buenas prácticas aprendidas, encontramos el establecimiento de los indicadores antes de la obtención de los datos, ya que si nos falta algún dato se tendría que volver a realizar la extracción.
- Entre las competencias adquiridas durante este proyecto se encuentran el aprendizaje del lenguaje Python y una mayor profundidad en el conocimiento del campo de la inteligencia de negocio y Big Data.