



ADMINISTRACIÓN DE SISTEMAS

TRABAJO GRUPAL

Documentación



Jon Blasco
Xabier Dermit
Ibai Franko



DESCRIPCIÓN	3
SOFTWARE	3
<i>Twitch API</i>	3
<i>Python</i>	3
<i>Grafana</i>	3
<i>MySQL</i>	4
<i>AWS</i>	4
DESPLIEGUE	4
ARCHIVOS	4
<i>twitch.py</i>	4
<i>twitch.sql</i>	5
<i>Descripción.pdf</i>	5
<i>Documentación.pdf</i>	5
CONCLUSIONES	5
BIBLIOGRAFÍA	5

DESCRIPCIÓN

En este proyecto se ha llevado a cabo la creación de un Dashboard que resume e interpreta gráficamente datos extraídos de la plataforma Twitch, la cual se utiliza para la retransmisión de video en directo. A partir de estos datos, se crean diferentes gráficos que reflejan por ejemplo entre otras cosas, la cantidad de espectadores que posee cada videojuego en total en el momento que se consulta el Dashboard. Nos muestra un resumen de los datos más interesantes de una manera actualizada y fácil de entender gracias al uso de diferentes tipos de gráficos y colores.

SOFTWARE

Para el desarrollo de este proyecto se han utilizado diferentes herramientas software que analizaremos una a una, explicando su funcionamiento dentro del proyecto así como el por qué de su uso en el mismo.

Twitch API

Una API (interfaz de programación de aplicaciones) es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. En este caso, utilizaremos la API oficial de Twitch para recopilar los datos necesarios para nuestro Dashboard.

A pesar de haber otro tipo de fuentes de información para datos como pueden ser cuentas de Twitter que te dan los datos directamente sin necesidad de interactuar con ninguna API, hemos decidido hacerlo nosotros directamente, para familiarizarnos con el uso de una API y así garantizar la veracidad de los datos y la creación de un Dashboard real y de calidad.

Python

Lo primero que necesitamos a la hora de crear un Dashboard son los datos. En nuestro caso, hemos utilizado la API oficial de Twitch, que proporciona una infinidad de datos sobre su plataforma. Para conseguir extraerlos, decidimos usar Python, ya que su conexión con la API es ya conocida y por lo tanto aprender a utilizarla se nos haría más sencillo. Por lo tanto, creamos un archivo python en el que se crea la conexión con la API, la cual será nuestro proveedor de datos, y haremos que se ejecute de manera continuada en un periodo de tiempo para lograr tener un Dashboard con datos actualizados en todo momento.

Grafana

Otra herramienta fundamental es la que nos permitirá crear nuestros gráficos para el Dashboard. Hay varias alternativas en cuanto a herramientas para esta función, como puede ser la vista en clase Kibana, u otras como Grafana o Apache Superset. Después de analizar las ventajas y desventajas de cada una, decidimos usar una herramienta no vista en clase, para así añadir más valor a nuestro proyecto. Finalmente nos decantamos por

Grafana, ya que creemos que visualmente era más atractiva y aparentemente más sencilla de usar que Apache Superset.

MySQL

Los datos juegan un papel muy importante en nuestro proyecto, y Grafana admite una gran variedad de fuentes de datos, como diferentes bases de datos (MongoDB, MySQL...). Como hemos mencionado anteriormente, nuestro proveedor de datos es la API oficial de Twitch, la cual mediante nuestra aplicación Python creada, nos devuelve los datos en formato JSON. Debido a eso, y gracias a que python posee una librería que permite pasar resultados de JSON a una base de datos MySQL, nos decantamos por esta base de datos, la cual posteriormente añadiríamos a Grafana para crear los gráficos.

AWS

Para terminar, el único requisito funcional que tenía el proyecto era el que proporcionase la opción a no componentes del grupo de poder ver y probar por su cuenta nuestro proyecto. Para ello se ha hecho uso de AWS (Amazon Web Services), donde hemos creado una máquina virtual que muestra nuestro Dashboard.

DESPLIEGUE

El despliegue es muy sencillo, ya que hemos creado un nuevo usuario para poder entrar sin miedo de cambiar nada. Para acceder, solamente hay que ingresar la IP x.xx.xxx.xxx:3000 en nuestro navegador, y acceder al login de Grafana con los credenciales que se muestran a continuación:

Usuario: ProyectoGrupalAS

Contraseña: ProyectoGrupalAS

En la descripción del proyecto se muestra detalladamente la manera de ingresar al proyecto, en caso de tener dificultades, he aquí el enlace a dicho documento:

<https://docs.google.com/document/d/1nsGLrOoRuqDB0Bsm9OmxK55GwK9KjWi-uSo1eVNL6XE/edit?usp=sharing>

ARCHIVOS

En este apartado detallaremos los archivos proporcionados en la carpeta comprimida y su contenido.

twitch.py

Código fuente que crea la conexión con la API de Twitch y hace la conversión de los datos en formato JSON a base de datos MySQL



twitch.sql

Base de datos que almacena los datos y que servirá para que Grafana pueda actualizarse en cualquier momento.

Descripción.pdf

Documento que introduce el proyecto y resume de qué se trata, además de sus características y forma de uso.

Documentación.pdf

Documento que desarrolla las herramientas software utilizadas a lo largo del proyecto, y su papel dentro del proyecto. También incluye las pautas para desplegar el Dashboard.

CONCLUSIONES

Siendo el objetivo de la asignatura profundizar en diferentes técnicas y herramientas actuales para administrar sistemas informáticos, creemos que el planteamiento de este trabajo ayuda a reforzar la destreza en dichas competencias, ya que te obliga a investigar acerca de softwares de actualidad aunque desconocidos por nosotros, para después poder crear algo propio con dichas tecnologías.

BIBLIOGRAFÍA

En este apartado mencionaremos los documentos o enlaces que nos han servido de gran ayuda a la hora de crear el proyecto.

Twitch: página oficial

<https://www.twitch.tv>

Twitch API: documentación

<https://dev.twitch.tv/docs>

Grafana: documentación

<https://grafana.com/docs/grafana/latest/>

AWS: documentación

<https://docs.aws.amazon.com/>

Stack Overflow

<https://stackoverflow.com/>