

## Resumen

Este proyecto consiste en la implementación de una aplicación web utilizando herramientas para el uso y tratado de datos con Python. Para ello hemos basado nuestra práctica en una aplicación de búsqueda de información de usuarios en determinadas redes sociales, haciendo uso de distintas APIs para cumplir tal propósito. A su vez, mostraría los datos más relevantes y específicos a la búsqueda mostrando, en la medida de lo posible, la información hallada en forma de diferentes estructuras y bloques de datos.

## Listado exhaustivo de las funcionalidades a implementar

- F1: Búsqueda por nombre de usuario en las diferentes Redes Sociales.
- F2: Obtención y extracción de estadísticas generales en algunas de las Redes Sociales.
- F3: Obtención y extracción de estadísticas dado un usuario en todas las Redes Sociales (F1).
- F4: Generación de gráficas y bloques de datos a partir de la información obtenida en F2, en diferentes dimensiones.
- F5: Interactuar con los bloques de datos de información para obtener información específica.
- F6: Generación de gráficas y bloques de datos a partir de la información obtenida en F3 en diferentes dimensiones:
  - F6.1: Mostrado del número de seguidores, suscriptores o visitas en una frecuencia determinada, dependiendo de la plataforma.
  - F6.2: Mostrado de la frecuencia de subida de contenido multimedia a la plataforma en específico.
  - F6.3: Mostrado del contenido multimedia más reciente, así como información más reciente si no hay contenido multimedia.
  - F6.4: Mostrado del Avatar o Logo del usuario buscado.
  - F6.5: Mostrar información relevante de la plataforma relacionada con el usuario.
- F7: Opción de hacer login y registrarse en la aplicación.
- F8: Obtención de información del usuario en cuestión acerca de las búsquedas.
- F9: Creación del usuario, añadiendo éste a la Base de Datos.

## Justificación de funcionalidades con respecto a primera iteración

Se puede apreciar como no se ajustan el número de funcionalidades al principalmente pactado. Esto se produjo dada el propio tránsito de trabajo a la hora de tratar con las distintas APIs y los consecuentes resultados. Es decir, a medida que realizábamos el trabajo nos dábamos cuenta como algunas funcionalidades no eran necesarias, ni importantes o se podían resumir dentro de otras. Por ejemplo, la generación de un nick único de perfil tenía como objetivo que un usuario pudiese introducir sus datos de usuario referente a las tres redes sociales para poder buscar directamente desde esos campos de información. Sin embargo, vimos que esto no era necesario, ya que las propias APIs ya realizaban una búsqueda exhaustiva e “insensitive” de los usuarios a encontrar, entonces no lo vimos necesario. Además el funcionamiento original de la aplicación consistía en una búsqueda de usuario en todas las Redes Sociales y, más tarde, tener la opción de elegir la Red Social para mostrar los datos. Obviamente este objetivo no es acertado dada la ineficiencia de las peticiones porque ha de buscar en las tres Redes Sociales a la vez, con lo que ello respecta. Por ello se ha decidido introducir algunas de las funcionalidades dentro de otras, ya que no cobraban mucho sentido y no tenían relevancia a la hora de ser implementadas.

## Flujo de datos de aplicación

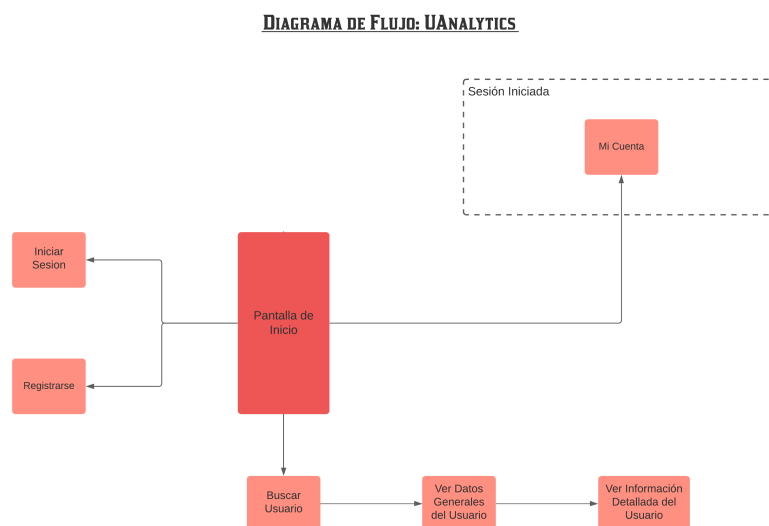


Figura 1: Flujo de datos de la aplicación

El flujon de datos de la aplicación no sufre cambio ya que se mantiene el recorrido que realiza la información a lo largo del transcurso de la ejecución de la aplicación.

## APIs utilizadas

- API de Twitch
- API de Youtube (Youtube Data API v3)
- API de Twitter

## **Información del API de Twitch**

Dentro del esquema general del mostrado de información, hemos optado por mostrar información general sobre las categorías más vistas en la plataforma, cuando se accede a la parte de Twitch pero sin escribir en el formulario. Una vez introducidos los datos, se muestra información específica del usuario y después muestras de datos recopilados y tratados, como es el caso de la gráfica que con Pandas se procede a agrupar por fecha, en este caso mes, para poder mostrar las visitas obtenidas durante esa franja de tiempo. Además se provee de datos que se encuentran especificados en las funcionalidades.

## **Información del API de YouTube**

Esta API contenía muchas limitaciones, con respecto a las otras dos APIs, sin embargo pudimos obtener información general, mostrada en las categorías antes de rellenar el formulario. Por otra parte se muestra información de una forma similar a las otras dos APIs, apostando por una unificación en el diseño. Se obtienen gran cantidad de datos que son tratados para poder obtener las gráficas tanto de likes como de comentarios, además del último contenido multimedia, por ejemplo.

## **Información del API de Twitter**

Esta última API tenía mucho potencial para poder tratar la información, ya que no limitaba abiertamente las búsquedas y se podía contar con grandes cantidades de información. Tal es el caso, que pudimos agrupar la información obtenida en días del mes, no sólo en meses, por ello cuenta con tres funciones de gráficas muy informativas. Además de información relevante para completar el conjunto.

## **Información específica**

La información relevante a la propia práctica, en cuanto a su ejecución, se ha depositado en el README. En este pdf se muestra información general relevante al funcionamiento de la misma.