

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 propuesta 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, contará con 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 RRSS.
- F2: Búsqueda por nombre de usuario filtrando por Red Social.
- F3: Obtención y extracción de estadísticas dado un usuario en todas las RRSS (F1)
- F4: Obtención y extracción de estadísticas dado un usuario en una determinada Red Social (F2)
- F5: Generación de gráficas y bloques de datos a partir de la información obtenida en F3, en diferentes dimensiones:
 - F5.1: Para cada Red Social mostrar información general acerca de ella, en base a una frecuencia.
- F6: Interactuar con las gráficas de información general para obtener datos específicos (F7)
- F7: Generación de gráficas y bloques de datos a partir de la información obtenida en F4, en diferentes dimensiones:
 - F7.1: Mostrado del número de seguidores o suscriptores en el último año, dependiendo de la plataforma.
 - F7.2: Mostrado de la frecuencia de subida contenido multimedia a la plataforma por semana.
 - F7.3: Mostrado de número de 'likes' en contenido multimedia en los últimos 30 días, dependiendo de la plataforma.
 - F7.4: Mostrado de información general del último contenido multimedia subido, dependiendo de la plataforma.
 - F7.5: Mostrado del Avatar del usuario buscado.
 - F7.6: Mostrado de información específica de la plataforma.
- F8: Generación de gráficas con información general sobre las diferentes redes sociales.
- F9: Opción de hacer login y registrarse en la aplicación

- F10: Obtención de información del usuario en cuestión acerca de las búsquedas.
- F11: Generación de nick único del perfil para búsquedas posteriores enviando petición al administrador.
- F12: Creación del usuario, añadiendo éste a la Base de Datos
- F13: Asignación del usuario, por parte del administrador, a un nick específico

Bocetos de las pantallas de la aplicación

El primer mockup consiste en la pantalla principal de la aplicación web, ésta muestra una barra superior donde se encuentra el logo y el nombre de la app, además de una barra de búsqueda dónde poder conseguir información del usuario pudiendo filtrar por red social. En la zona media de la pantalla se encuentra la zona de búsqueda principal, es decir, dónde se tendrá la opción de buscar el usuario en todas las redes sociales. Además se incluye en la parte superior derecha la opción para loggearse o registrarse. Por último, en la zona baja se muestran tres gráficas de las redes sociales mostrando datos de carácter general y que no serán interactivos.

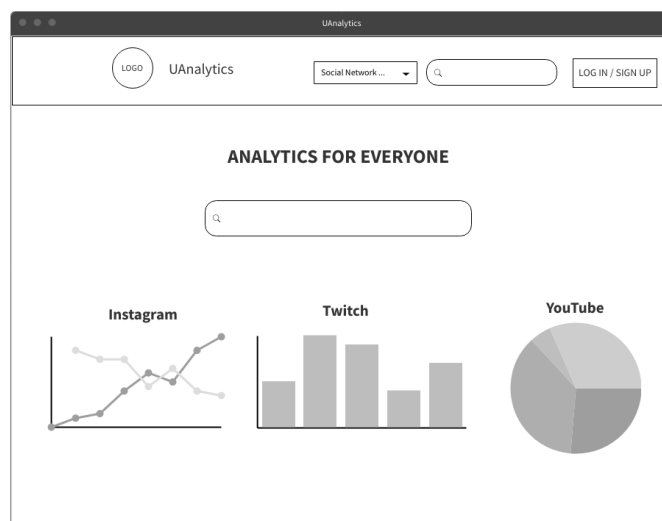


Figura 1: Mockup 1

El segundo mockup muestra la pantalla que enseña datos correspondientes al usuario pero de una forma general. Es decir, dentro de las tres redes sociales se observa información relevante al usuario pero no de forma específica, con el objetivo de captar los datos particulares al interactuar con las gráficas.

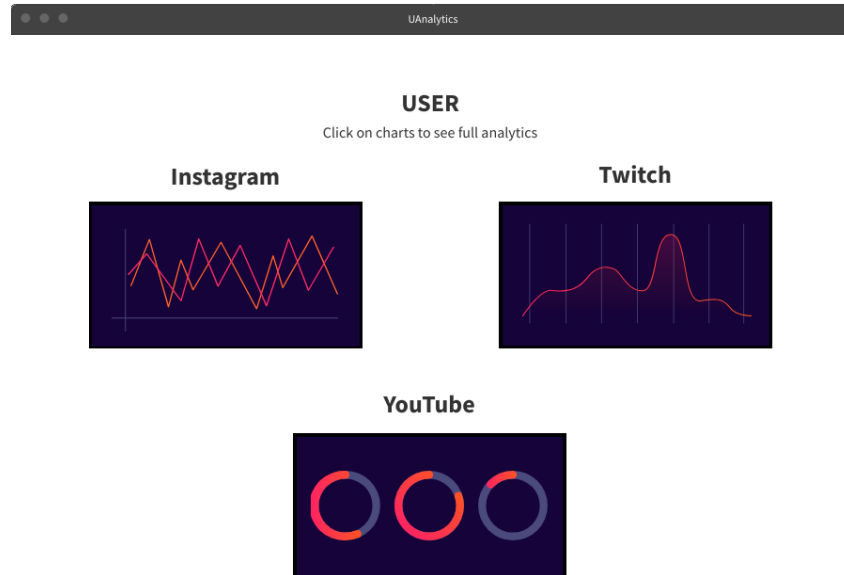


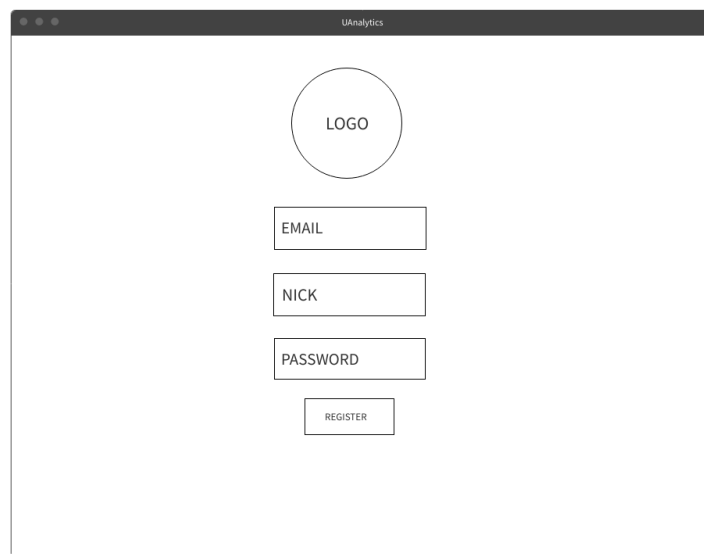
Figura 2: Mockup 2

El tercer mockup, representa la información detallada del usuario en la red social elegida mostrando diferentes gráficas y bloques de datos. Con el objetivo de realizar más funciones con el tratamiento de datos se ha decidido mostrar información específica y con distintas implementaciones, usando desde diagramas de sectores hasta tablas de datos ordenados.



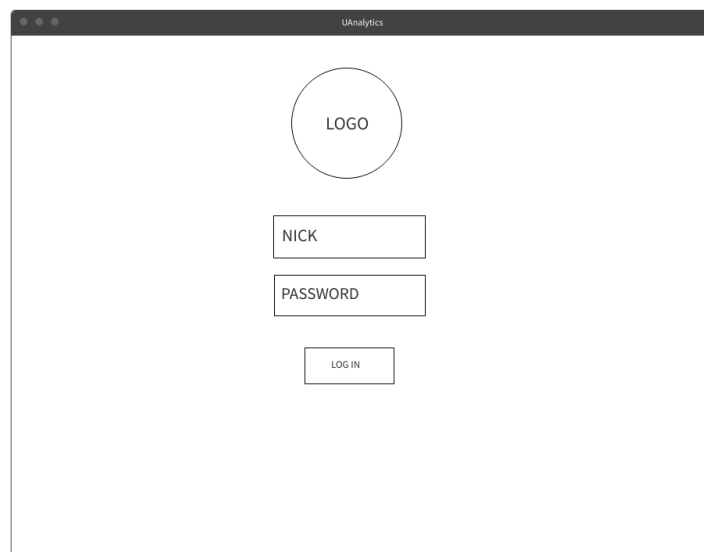
Figura 3: Mockup 3

Los dos siguientes mostrarían el registro y login de los usuarios a la propia aplicación.



A wireframe of a user registration form within a browser window titled "UAnalytics". The form is centered and consists of a circular logo placeholder at the top, followed by three stacked rectangular input fields labeled "EMAIL", "NICK", and "PASSWORD". Below these fields is a single rectangular button labeled "REGISTER".

Figura 4: Mockup 4



A wireframe of a user login form within a browser window titled "UAnalytics". The form is centered and consists of a circular logo placeholder at the top, followed by two stacked rectangular input fields labeled "NICK" and "PASSWORD". Below these fields is a single rectangular button labeled "LOG IN".

Figura 5: Mockup 5

Este último mockup muestra la página del usuario de la aplicación, en la cual se presenta una lista de búsquedas realizadas además del nick del usuario en las distintas redes sociales. En esos cuadros de texto se enviarán las peticiones cada vez que se quiera informar de su nick en las redes sociales presentes pulsando en el botón de enviar.

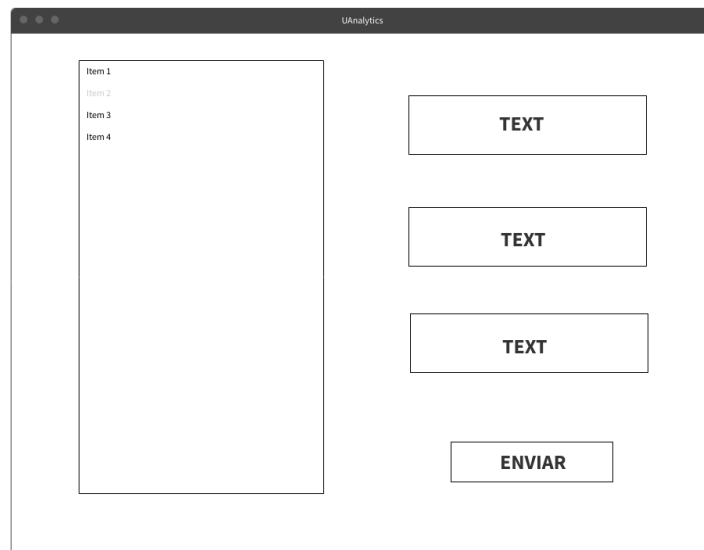


Figura 6: Mockup 6

Flujo de datos de aplicación

En el diagrama de flujo representamos gráficamente los procesos que podemos realizar dentro de nuestra página web. Vemos como desde la Pantalla de Inicio tenemos la opción de registrarnos o si ya tenemos cuenta, Iniciar Sesión, una vez hayamos iniciado sesión podremos entrar en Mi Cuenta donde veremos toda nuestra información. Iniciar Sesión no será obligatorio ya que podremos realizar todas las demás funciones sin necesidad de ello. También desde la Pantalla de Inicio tendremos como proceso principal Buscar Usuario, que una vez se haya completado con éxito veremos su información general además de tener la opción de poder verla más detalladamente.

DIAGRAMA DE FLUJO: UANALYTICS

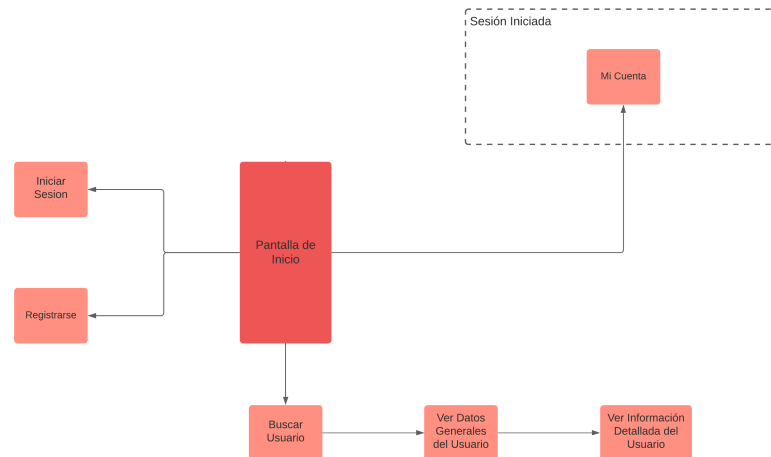


Figura 7: Flujo de datos de la aplicación

APIs utilizadas

- API: Instagram
- API2: Youtube
- API3: Twitch

Información a extraer del API1 - Instagram

De esta API se requerirá el acceso a datos básicos de cuentas de la plataforma. El objetivo principal es conseguir datos de usuarios como del número de seguidores, número de publicaciones multimedia realizadas, número de 'likes' obtenidos, entre otros. De esta forma podemos obtener lo que nuestra aplicación necesita, ya sea a nivel de una única publicación ('likes', comentarios o engagement) como a nivel de cuenta, pudiendo distinguir la variedad de datos para conseguir gran pluralidad. Es decir, lo que se busca es información esencial del usuario ya que para poder obtener datos de mayor nivel se necesitan permisos que no obtenemos. Para poder cumplir con nuestra propuesta se deberá usar la API Graph de Instagram que es la que nos proporcionará la información deseada, ya que existe la API de visualización básica de Instagram que se nos queda corta en cuanto a la recopilación de datos.

Información a extraer del API2 - Youtube

La API cuenta con una diferenciación notable con respecto a las otras dos APIs debido a la información concluyente de ésta, es decir, no podríamos obtener los mismos tipos de datos, pero si parecidos. Ya que nuestro objetivo es mostrar los más relevante posible para el usuario, podríamos obtener datos a nivel de canal, publicaciones o vídeos, lo que nos permitiría tratar la información entre apartados y obtener datos como número total de 'likes' o 'dislikes', el ratio entre ellos, cantidad de suscriptores por una frecuencia determinada, cantidad de publicaciones en un tiempo determinado, ratios de engagement en tiempo de reproducción, entre otros.

Información a extraer del API3 - Twitch

El objetivo de usar el API de twitch es conseguir modelos de información simplificados y estructuras consistentes, que es de lo que trata esta API. Dentro de las funcionalidades, se podría obtener información en tiempo real además de un tratado de errores íntegro. Por ello, podríamos obtener información del número de suscripciones, número de seguidores, últimos estados descritos, últimos juegos, engagement del perfil, fechas de streaming, tiempo de duración de directos, entre otros. El objetivo es tratar esta información de la forma más eficiente posible para mostrar los datos más relevantes al usuario.

Por lo tanto, el objetivo del uso de estas APIs, es recopilar la máxima información posible del usuario a buscar para poder realizar un tratado de datos eficiente con Pandas, por ello se han elegido estas opciones. Es decir, a parte de proporcionarnos grandes cantidades de información similar, también nos ofrecen un gran abanico de posibilidades para usarlos y mostrarlos como nos parezca más relevante.

Uso de Pandas dentro de la práctica

Dado que el objetivo de nuestra aplicación es el mostrado de la información más relevante de la forma más estilizada y eficiente posible, el uso de Pandas ha de ser exhaustivo. Por ello, se recoge información relevante a cada una de las APIs y se pretende recopilar datos muy distintos y de formas variadas por lo que se mostrarán algunos ejemplo cuyo objetivo será englobar el uso total de la librería.

Para la API de Twitch, que contiene bastante flexibilidad, se podría realizar el uso de `count()` para recoger el numero de elementos que pertenecen al campo de suscripciones de un usuario y junto a la información de fecha que cuenta ese atributo se concluiría con una frecuencia de suscripciones en un transcurso de tiempo.

Con la API de Youtube existen muchas series que contienen información irrelevante para el usuario, con lo que se usaría `dropna()` para eliminar esos datos desconocidos o nulos, además de poder ordenar la serie con `sort-values()` para poder dar información de forma ordenada.

Los DataFrame nos son de gran utilidad para esta aplicación, ya que mediante los diccionarios de datos que nos proporciona cualquiera de las 3 APIs podemos mostrar elementos generales de las redes sociales, pudiendo filtrar por filas que cumplen una determinada condición, como por ejemplo un número de seguidores mayor que 'x', por ello podemos echar mano de `df[condición]` usando una instancia del DataFrame y asignándole una condición.

Además Pandas nos ofrece un modo de previsualización de DataFrames en forma de tablas lo que nos ayudaría en gran manera a representarlo de esta forma al usuario para alguna funcionalidad que pueda necesitarlo.

Funcionalidades a implementar en la primera iteración de la práctica

El objetivo es implementar las funciones relacionadas con el funcionamiento básico de la aplicación, es decir, todas aquellas que se encuentren entre la F1 y F5. Dependiendo de la dificultad de la implementación de las funcionalidades se podría incluir la F6 y F7, sin embargo las últimas 4, al ser ajenas a la función de la aplicación se realizarán después de la primera iteración. Por ello el objetivo sería conseguir algo parecido a los dos primeros mockups con opción de implementar de forma íntegra el tercero.

Uso de librerías externas (opcional)

Dado que nos interesa mostrar los datos al usuario de forma que éste los comprenda de forma sencilla, es necesario hacer uso de gráficas mediante la librería Matplotlib que nos ofrece gran variedad de plots, pudiendo hacer diagramas de barras, histogramas, diagramas de sectores o mapas de color. Con ello podríamos elegir la mejor opción para representar la información deseada y no estar sujetos a una única interfaz.

Además nos vendría bien hacer uso de la librería Datetime, que maneje datos de tipo fecha, lo cual nos facilitaría usar eficientemente esta información a la hora de tratarla y no complicarse con parseos innecesarios.