

TweetManager



Grupo Woody:

Víctor Lafaja Sanz (680794)
Santiago Morón Borraz (681417)
Iván Escuín González (684146)
Jorge Aguarón de Blas (698895)

29 de Agosto de 2018

ÍNDICE

1. Resumen del proyecto	3
2. Propuestas similares	4
2.1. <i>Hootsuite</i>	4
2.2. <i>TweetDeck</i>	4
2.3. <i>SpreadFast</i>	4
2.4. <i>Bit.ly</i>	4
3. Arquitectura de alto nivel	6

1. Resumen del proyecto

Este proyecto tiene como objetivo desarrollar un sistema que permite gestionar, mediante una interfaz típica de control de acceso de usuarios, una o varias cuentas de *Twitter*¹.

Para ello, se ha dividido el proyecto en *cuatro partes*:

1. *Sistema de control de acceso*: se restringirá o permitirá el acceso de un usuario a la aplicación mediante la autenticación por cuenta propia de la aplicación o mediante el acceso con cuentas de *Gmail*, *Facebook* o *Twitter*.
2. *Aplicación de usuario*: al acceder un usuario a la aplicación tendrá una vista principal del panel de control desde el que podrán gestionar sus cuentas de Twitter (añadir, eliminar).
3. *Análisis y estadísticas*: el usuario tendrá acceso a una vista desde la que tendrá una visión completa de cómo van sus cuentas.
4. *Interfaz de administración*: la aplicación tendrá un usuario administrador que, mediante su interfaz, se permitirá la gestión y administración de usuarios.

A nivel de implementación, se ha utilizado el stack MEAN para programar el sistema por completo.

¹ Es una red social en la que sus usuarios publican e interactúan mediante mensajes conocidos como “*tweets*”.

2. Propuestas similares

En el mundo competitivo de hoy en donde las empresas lanzan nuevos productos al mercado cada vez con mayor frecuencia, y en donde cada vez aparecen más empresas competidoras, se hace prácticamente una obligación realizar el análisis de la competencia.

La competencia puede ser directa cuando se trata de empresas que producen o venden productos similares a los de uno, o indirecta cuando se trata de empresas que producen o venden productos sustitutos a los de uno. Pero por lo general, al menos en el caso de las pequeñas y medianas empresas, para realizar el análisis de la competencia solo se suele tomar en cuenta la competencia directa.

En este apartado se van a analizar diferentes aplicaciones que hay en el mercado que tengan una funcionalidad similar (competencia directa) a *TweetManager*:

2.1. Hootsuite

Una de las herramientas *social media* más popular. Permite dirigir diferentes perfiles de redes sociales, programar mensajes o estados, publicar mensajes en varias redes de forma simultánea, etc. Además, su interfaz permite una cómoda visualización de cada plataforma. Una de las principales ventajas de *HootSuite* y, el objeto por el que fue creado, es la de la programación de publicaciones en el tiempo.

2.2. TweetDeck

Muchos lo consideran la competencia de *Hootsuite*, ya que las funciones de ambas herramientas *social media* son similares. Sin embargo, esta tiene una gran flexibilidad para su uso con todo tipo de plataformas y conecta a todos tus contactos a través de las diferentes redes sociales.

2.3. SpreadFast

SpreadFast es un sistema de manejo de redes sociales que te ayuda a organizar exitosamente tu presencia en las redes sociales y a poner en acción tus planes de una buena estrategia online. Este software activa un monitoreo y una aplicación social, haciendo posible organizar tu trabajo en un calendario global y utilizar herramientas organizacionales. Además, podrás utilizar contenido compartido y análisis comparativos para rastrear tanto tu trabajo como tus progresos. El contar con acceso móvil es otra prueba de la flexibilidad que esta plataforma ofrece.

2.4. Bit.ly

Una herramienta muy útil para ahorrar espacio y dar mejor estética a las publicaciones. Permite acortar las URL y, además, controlar las estadísticas de estos links. Estas funciones pueden mejorar la armonía visual, evitando enlaces demasiado largos, en redes como Facebook, pero en Twitter, se convierte en algo esencial para poder realizar publicaciones, ya que la extensión de contenidos se ve limitada a 140 caracteres.

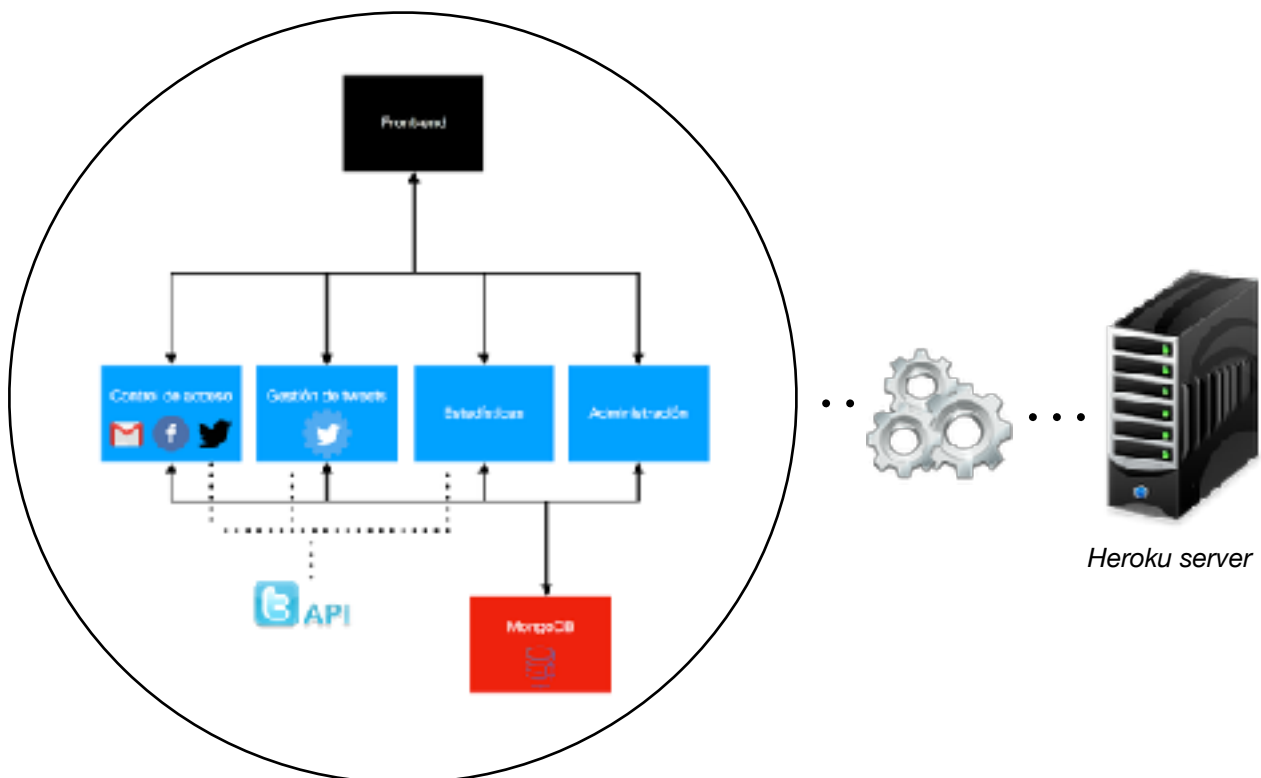
Hay muchas más aplicaciones que podríamos considerar competencia como: *Twiterfeed*, *Buffer*, *Blurtster*, *Sysomos*...

Puesto que que la competencia en la actualidad es muy grande, el objetivo debe ser implementar alguna funcionalidad nueva o tratar de hacer una aplicación más usable y accesible que las que hay en el mercado.

3. Arquitectura de alto nivel

A continuación se van a explicar los puntos más importantes de la arquitectura de la aplicación:

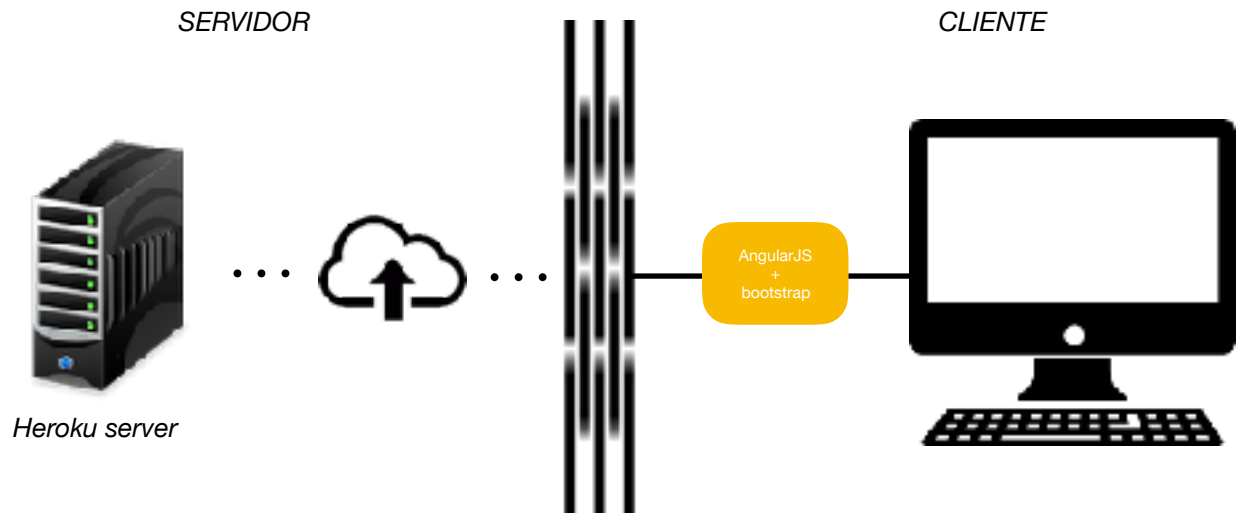
- ➔ El servicio principal de la aplicación es el servicio de gestión de tweets.
- ➔ Existen otros tres servicios adicionales, que se complementan entre ellos, como son: el control de acceso, análisis y estadísticas y administración.
- ➔ Todos estos servicios hacen uso de una base de datos no relacional² como es *MongoDB*.
- ➔ El control de acceso además de implementar una autenticación propia también hace uso de los sistemas de autenticación de Facebook, Google y Twitter.
- ➔ El sistema de autenticación de Twitter, el servicio de gestión de tweets y el servicio de estadísticas hacen uso de la *API de Twitter*.
- ➔ La parte de *front-end* se encarga de comunicarse con cada uno de los servicios y hacer las peticiones necesarias según las acciones solicitadas por el usuario.



Todos estos servicios se ejecutan en *Heroku*, uno de los *PaaS (Platform as Service)* más utilizados en la actualidad en entornos empresariales por su fuerte enfoque en resolver el despliegue de una aplicación. Además, Heroku permite manejar los servidores y sus

² Las bases de datos NoSQL están diseñadas específicamente para modelos de datos específicos y tienen esquemas flexibles para crear aplicaciones modernas. Las bases de datos NoSQL son ampliamente reconocidas porque son fáciles de desarrollar, su funcionalidad y el desempeño a escala. Usan una variedad de modelos de datos, que incluyen documentos, gráficos, clave-valor, en-memoria y búsqueda.

configuraciones, escalamiento y la administración. A Heroku basta con decirle qué lenguaje de backend se está utilizando o qué base de datos se va a utilizar y ya sólo hay que preocuparse únicamente por el desarrollo de la aplicación.



Los datos llegan al cliente a través de la conexión con los controladores. Cada una de las pantallas que muestran datos por pantalla utiliza un controlador que es el que se encarga de ejecutar una función en el servicio al que se conecta, que a su vez envía una petición a nuestra API (al servidor), el cual en muchos casos va a mandar una petición a la API de *Twitter*. Estos datos se devuelven a una variable del controlador asociado a la vista, que es accesible desde la vista.

//FALTA