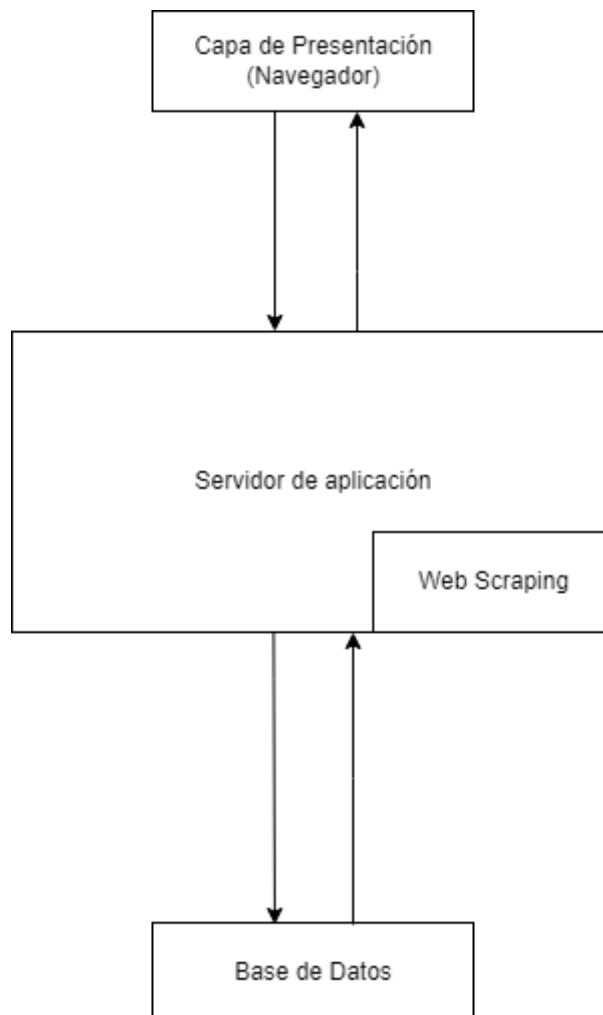


Diseño Arquitectónico

Raúl Rodríguez Rodríguez
Noah Wenck

Análisis del modelo 4+1 de Kruchten

Vista lógica

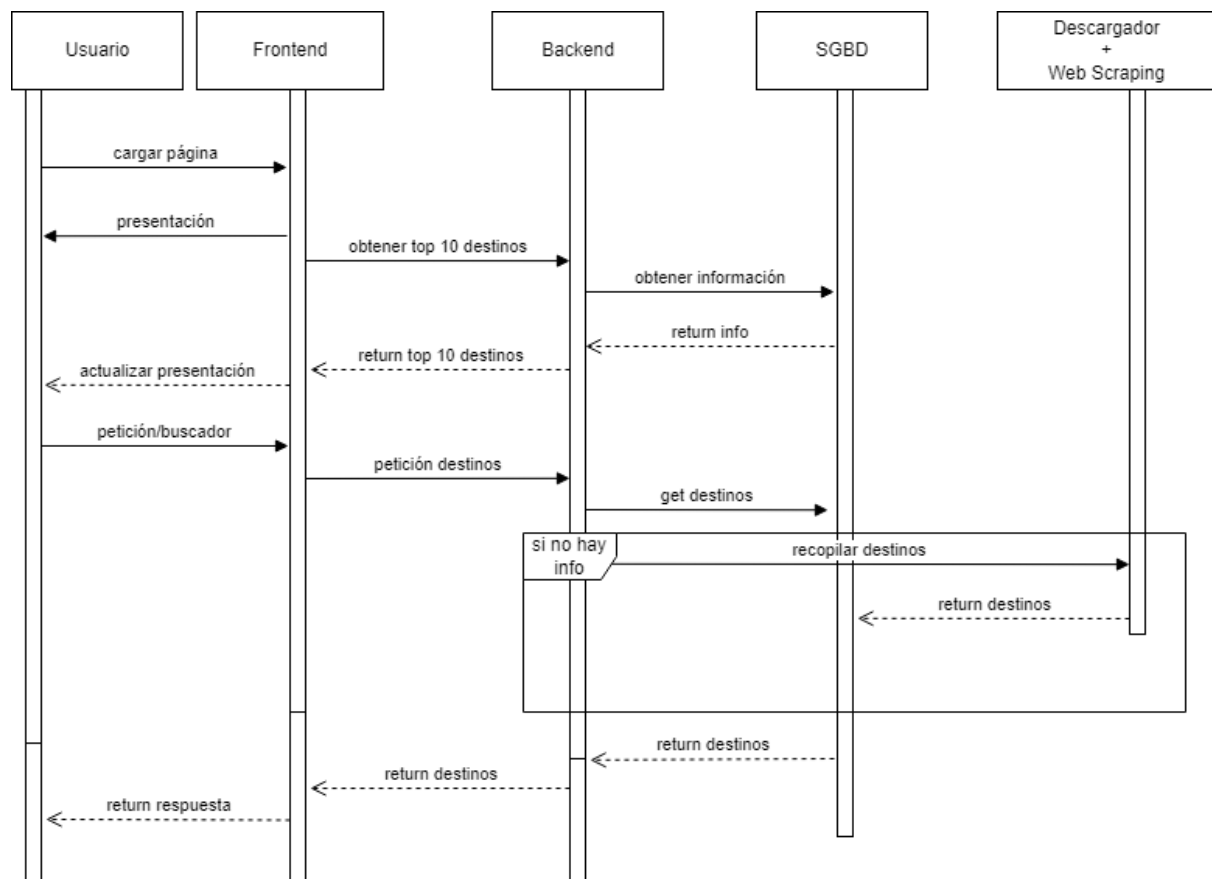


Se ha elegido esta opción o forma de planificar nuestro sistema para poder realizar correctamente los diferentes procesos sin ocasionar congestiones en ningún elemento, haciendo uso del **Front-End** en la **capa de presentación**, el **Back-End** en el **servidor de aplicación**, el cuál se encarga de **pedir al SGBD la información** necesaria para las

peticiones, y si no la encuentra se descarga (**API + Web Scraping**) la información necesaria para satisfacer las peticiones que no tengan respuesta de la BD.

No se han escogido otros métodos de esquematización porque tendría un rendimiento menor o no es una buena solución a la planificación del sistema.

Vista de procesos

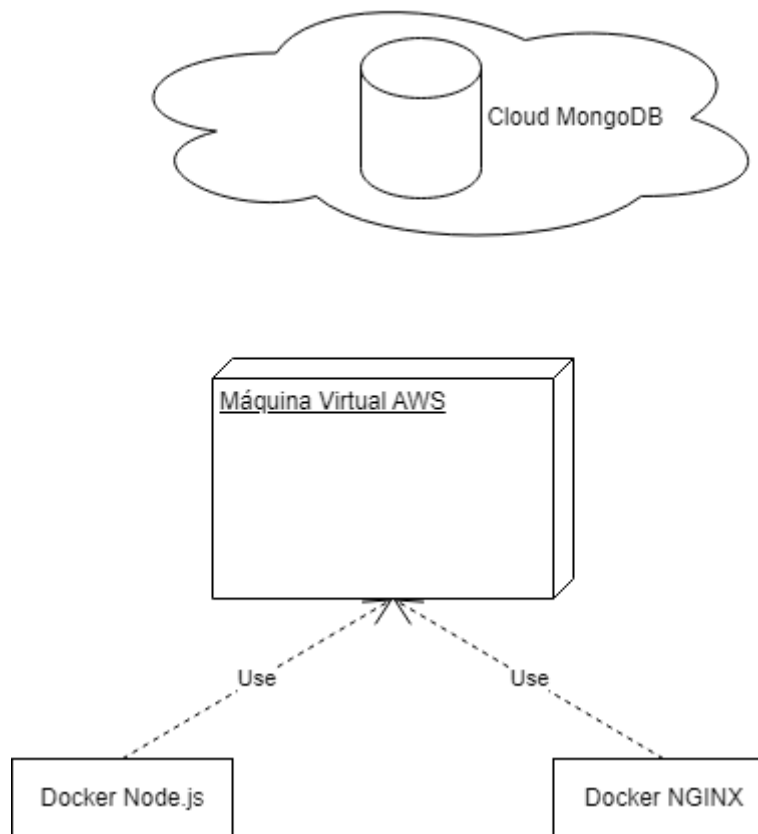


Para la vista de procesos se ha escogido este esquema porque da lugar (como ocurría antes con la vista lógica y siguiendo el sentido de la misma) a que se dé el intento de **búsqueda más rápido posible** recopilando **información previamente en la BD** y si no tuviera información de un lugar pedido, descargará por medio de una **API y Web Scraping** la información e imágenes que no esté disponible en la BD. Por lo tanto esta opción es la **más versátil en cuanto a tiempo de respuesta/satisfacción de peticiones**.

Otras formas de hacerlos que se han presentado son:

- Forma **más rápida u óptima pero no siempre complaciente**: solo entraría en el esquema el **SGBD o la BD** y se van a satisfacer las peticiones en las cuales haya información en la BD, sin hacer uso (más lento) de llamadas a la **API** y hacer uso de **Web Scraping** ("en caliente").
- Forma **más lenta pero siempre complaciente**: no se haría uso de la **BD** ya que usaría para cada petición llamadas a la **API y Web Scraping** para obtener la información, lo que sería muy lento.

Vista física



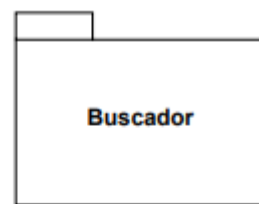
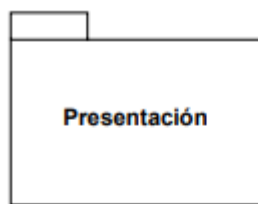
Con este esquema de la vista física se denota el **uso y distribución del trabajo** del **servidor** con el uso de los **Dockers** para cada función o utilidad, separando el trabajo en:

- **Docker** con **back-end** del servicio.
- **Docker** con **NGINX**, para servir el **front-end** de la página web.

La **base de datos** estará alojada en un servicio **Cloud de MongoDB**.

Otra forma de hacerlo podría hacer que no fuera tan eficiente la distribución de carga de trabajo, haciendo que se colapsara el sistema.

Vista de desarrollo



La forma elegida nos pareció la que mejor distribuye o clasifica mejor las funciones del sistema.