

Trabajo evaluativo integrador Final

Grupo de hasta 3 personas, presentación individual

Desarrollaremos un sistema de publicación de clasificados, mediante el cual las empresas inmobiliarias puedan cargar sus bases de datos mediante el posteo de un archivo json con la información relacionada a las propiedades, los usuarios/navegantes puedan buscar esos clasificados desde la home del sitio en base a una oración y traiga los resultados priorizados que permitan ver el detalle de la publicación y enviar un mensaje con la información de contacto al vendedor.

Para esto, se piden desarrollar 5 microservicios:

1. búsqueda
2. usuarios
3. mensajes
4. publicaciones
5. frontend

Para el servicio (1) utilizar un motor de búsqueda que permita una indexación y búsqueda de los ítems por sus características (título, descripción, atributos, ciudad, estado, etc), que se nutra mediante notificaciones del servicio de ítems y busque la información de ese servicio.

Para el servicio (2) utilizar una base de datos relacional para una persistencia ACID de la información del usuario, utilizar un sistema de colas para notificar a los otros servicios de la modificación, borrado, y el agregado de usuarios nuevos. Si se da de baja el usuario se debe notificar como un borrado y borrar sus ítems asociados y la información en el motor de búsqueda para que no se listen, así también como sus mensajes asociados. Proveer un endpoint de login para el frontend.

Para el servicio (3) permitir agregar nuevos mensajes a una publicación y que se disponibilicen tanto por publicación como por publicador (usuario) en un listado. Los mensajes no necesariamente tienen que tener un ítem asociado, pueden ser notificaciones del sistema al usuario.

Para el servicio (4) tener un caché centralizado con los datos de contacto del usuario para no buscarlos en cada request que necesite la información del ítem y que se actualice escuchando mediante un consumer los mensajes de la cola. Si la información del usuario no está en cache, tiene que solicitarla al servicio correspondiente y almacenarla para que los próximos requests se sirvan del caché. Los datos del ítem se pueden almacenar no relacionalmente.

Lo importante acá es que las imágenes se sirven localmente, por lo que hay que prever una **descarga concurrente asíncrona no bloqueante** de las imágenes de los items provistas en la carga del catálogo inicial.

La carga de la información del item, la realiza el usuario posteando un archivo .json con un arreglo de items, una vez que obtengamos el archivo finaliza el request y el procesamiento se realiza de forma asíncrona, se le notifica al usuario utilizando el servicio de mensajería el resultado del proceso, X procesados correctamente, Y no procesados por W causas.

Para el servicio (5) proveer un frontend que tenga la vista de inicio con el input de búsqueda, el listado de publicaciones, el detalle de la publicación y la posibilidad de enviar mensajes al usuario. Proveer un panel de administración al usuario donde pueda ver sus mensajes y sus publicaciones, así también como borrarlas o borrar su cuenta completamente con un login para su ingreso.

Requerimientos no funcionales:

- 1) Proveer uno o varios balanceadores de carga para tener múltiples instancias de los servicios.
- 2) Proveer un servicio de escalado automático que ante la saturación o caída de un contenedor de un servicio pueda proveer una nueva instancia y la asocie al balanceador.
- 3) Utilizar github o similar para versionar el código fuente
- 4) Escribir la documentación en el readme del repositorio

Para el final:

1. Hacer todas las correcciones previstas en la entrega del trabajo.
2. Desarrollar los tests de carga que permitan ver los límites de concurrencia de cada servicio.
3. Desarrollar un cache local que permita incrementar esa capacidad.
4. Desarrollar un frontend (de administración) que pueda ver el estado de los contenedores de los servicios y me permita crear nuevos contenedores y destruirlos.

Nota: El desarrollo de todos los puntos es imprescindible para la aprobación del trabajo.

Ej de la estructura de un item: <https://api.mercadolibre.com/items/MLA1146082202>