

Introducción

Tu jefe está loco, todos lo sabemos, porque si no, no es lógica la conversación que has tenido esta mañana con él.

Ha llegado con los pelos canosos, largos y revueltos y lo primero que te ha dicho es que tiene una gran idea innovadora: “quiere hacer una herramienta donde los usuarios se puedan registrar y mandarse mensajes entre ellos!”. Vamos, pura innovación...

Y te ha pedido a tí que te encargues de desarrollar la API del proyecto, es el momento de demostrarle que no eres un gallina!

Te has sentido como Marty Mcfly en Regreso al Futuro, y le has preguntado si conocía las diferentes herramientas de mensajería que ya existen en el mercado, pero él se ha negado a escucharte y ha seguido con lo suyo, así que te has resignado y has empezado a desgranar el proyecto en historias de usuario para ponerte a desarrollar la API.



<> Prueba técnica backend Kubide

Tecnologías

En cuanto a tecnologías, no presta ninguna atención a ese punto, pero sí que te ha pedido que sea con **NestJS** porque ha oído por ahí que existe ese framework que mola mucho, así que ya sabes que muchas veces es mejor hacer caso a sus “locuras”. Además, ha leído que los ORM como **TypeOrm** ayudan mucho al desarrollador, así que te pide usar este en tu capa de acceso a datos conectado a **PostgreSQL**.

Para comprobar cómo has realizado el trabajo, tu jefe te pide que subas el proyecto a un sistema de control de versiones **GIT** (puede ser Github u otro) y que vayas haciendo subidas de código iteradas siguiendo **conventional commits**.

Y, por último, para no quedar mal delante del jefe, has oído por el pasillo que está empezando a leer algo sobre **Docker**, por lo que te apuntarías un buen tanto si la entrega la haces utilizando esta tecnología.

Casos de uso

Para cubrir todas las necesidades del loco de tu jefe, te pide que cumplas con los siguientes casos de uso:

- Como USUARIO quiero poder llamar a la API para crear usuarios, teniendo en cuenta que no se podrán registrar dos veces con el mismo email.
- Como USUARIO quiero poder llamar a la API para hacer login y obtener un token de autenticación para las siguientes peticiones.
- Como USUARIO quiero poder llamar a la API para editar mis datos.
- Como USUARIO quiero poder llamar a la API para consultar mis datos.
- Como USUARIO quiero poder llamar a la API para indicar si estoy activo o no.
- Como USUARIO quiero poder llamar a la API para obtener la lista de usuarios activos.
- Como USUARIO quiero poder llamar a la API para crear un nuevo mensaje.
- Como USUARIO no quiero recibir mensajes si no estoy activo.
- Como USUARIO quiero poder llamar a la API para consultar mis mensajes.
- Como USUARIO quiero que se guarden notificaciones cada vez que recibo un mensaje.
- Como USUARIO quiero poder llamar a la API para consultar mis notificaciones.



Documentación

No olvides que tu jefe es un apasionado de la documentación y buenas prácticas, así que si documentas los endpoints de la API con **Swagger** mucho mejor.

Además, asegúrate de que el fichero **README** de tu proyecto lleva bien explicados los pasos a seguir para levantar el proyecto y ponerlo a funcionar y, como consejo, si le adjuntas una colección de **Postman** con los endpoints ya preparados para que los pruebe... éxito asegurado!

