

FlatFinder Lado del Servidor (Node)

Introducción

Nombre de la aplicación: Servidor Rest API de FlatFinder

Caso de Estudio

La base de clientes se ha ampliado. Aunque el sitio está operativo, su rendimiento es lento. Además, el servidor carece de flexibilidad y escalabilidad. Para abordar estos problemas, es necesario vincular el cliente a un servidor creado con Node.js.

Los desarrolladores colaborarán estrechamente, en parejas, uno se centrará en integrar las llamadas de los clientes con el servidor Node.js y el otro en desarrollar el lado del servidor.

Usa tu proyecto FlatFinder Lado del Cliente (React) y sigue:

Objetivos

Se espera que el proyecto haga que el estudiante obtenga confianza en las siguientes habilidades:

- Desarrollar el lado del cliente.
- Programar una REST API.
- Manejar autenticación y autorización.
- Manejar una base de datos.

Tecnologías

- Servidor - Node, Express
- Base de datos - MongoDB
- Test - Postman/Insomnia
- Seguridad - json web token

Habilidades Blandas

The journey of this project will include practicing and evolving the following skills:

1. Habilidades de revisión de código y revisión del trabajo de compañeros: practicar la capacidad de revisar el código de otra persona de manera constructiva, enfocándose en identificar áreas de mejora y al mismo tiempo reconocer las fortalezas.

2. Habilidades para levantar banderas: aprender a levantar banderas de manera efectiva, abordando inquietudes sobre problemas técnicos o la integridad del proyecto de manera clara y concisa.

3. Comunicación con los líderes de equipo (tu tutor): desarrollar habilidades para comunicarse con los líderes de equipo, incluida la capacidad de transmitir complejidades técnicas en términos comprensibles y facilitar la alineación y la toma de decisiones.

4. Diplomacia y Retroalimentación: Cultivar diplomacia cuando proveemos retroalimentación en la revisión de código, fomentando una cultura de crítica constructiva y crecimiento mutuo dentro del equipo.

5. Transparencia y Colaboración: Entender la importancia de la transparencia y la colaboración dentro del equipo, asegurando que las preocupaciones se abordan abiertamente y contribuyendo a un entorno de proyecto cohesivo.

6. Resolución individual de problemas: mejorar las habilidades de resolución de problemas abordando de forma independiente los desafíos encontrados durante el proyecto, fomentando la resiliencia y la creatividad para encontrar soluciones.

7. Habilidades de autoaprendizaje: enfatizando la importancia del autoaprendizaje continuo para mantenerse actualizado con nuevas tecnologías y metodologías, permitiendo el crecimiento personal y profesional en el campo en constante evolución del desarrollo full-stack.

Trabajo en Equipo

Un equipo tiene dos miembros y la división del trabajo es así:

Entidades

User (Programador A)

- id
- email
- password (contraseña)
- first name (primer nombre)
- last name (apellido)
- birth date (fecha de nacimiento)
- es admin
- lista de "favourite flats"
- created (creado)
- updated (actualizado)

Flat (Programador B)

- id
- city (ciudad)
- street name (nombre de la calle)
- street number (número)

- area size (área)
- has ac (tiene aire acondicionado)
- year built (año de construcción)
- rent price (precio de renta)
- date available (fecha disponible)
- owner id (id del dueño)
- created (creado)
- updated (actualizado)

Message (Programador B)

- id
- content (contenido)
- flat id
- sender id
- created

Puntos finales (End Points)

Respecto a los permisos de los siguientes endpoints: todos requieren que los clientes estén al menos conectados.

User

función	ruta	método http	permisos
getAllUsers	/users	GET	admin
getUserById	/users/:id	GET	
updateUser	/users	PATCH	admin/account owner (dueño de la cuenta)
deleteUser	/users	DELETE	admin/account owner
login	/users/login	POST	
register	/users/register	POST	

Flat

función	ruta	método http	permisos
getAllFlats	/flats	GET	
updateFlat	/flats	PATCH	flat owner (dueño del departamento)
deleteFlat	/flats	DELETE	flat owner
addFlat	/flats	POST	flat owner

función	ruta	método http	permisos
getFlatById	/flats/:id	GET	

Messages

función	ruta	método http	permisos
getAllMessages	/flats/:id/messages	GET	flat owner
getUserMessages	/flats/:id/messages/:senderId	GET	the sender (el remitente)
addMessage	/flats/:id/messages	POST	

Resumen

El desarrollo será hecho en el siguiente orden:

Tarea	Días
User API	1
Flat API	1.5
Messages API	1
Auth API	1.5
Total	5

¡Buena suerte!