

Task for a Great Backend Developer - Scootin' About

Description

A company called Scootin' About will deploy electric scooters in Ottawa and Montreal. Design and implement a backend service that exposes a REST-like API intended for scooter event collecting and reporting to mobile clients.

1. The scooters report an event when a trip begins, report an event when the trip ends, and send in periodic updates on their location. After beginning a trip, the scooter is considered occupied. After a trip ends the scooter becomes free for use. A location update must contain the time, and geographical coordinates.
2. Mobile clients can query scooter locations and statuses in any rectangular location (e.g. two pair of coordinates), and filter them by status. While there will be no actual mobile clients, implement child process that would start with main process and spawn three fake clients using API randomly (finding scooters, travelling for 10-15 seconds whilst updating location every 3 seconds, and resting for 2-5 seconds before starting next trip). Client movement in straight line will be good enough.
3. Both scooters and mobile client users can be identified by an UUID.
4. For the sake of simplicity, both mobile client apps and scooters can authenticate with the server using a static API key (i.e. no individual credentials necessary but will most probably be introduced as the project develops further).

Deliverables

- README.md file with instructions for building and running.
- Simple documentation for the API (can be in the same README.md or another format you prefer).
- An example database dump or migrations for the demo.
- Any supplementary material you think is useful.
- Extra kudos if docker compose can be used to easily run the application.
- All of that in a private GitHub.com/GitLab.com/Bitbucket repository of your choice. We will provide an account to share it with.

Tips

- Use any libraries/frameworks/databases that you see fit.
- Where spec is ambiguous make an assumption towards the simpler side or ask us. Document these assumptions made in an ASSUMPTIONS.md file. Simple bullet point list will be sufficient.
- Consider that many users could be using the application at the same time.
- Beautiful and expressive code, meaningful tests, and proper organisational technique are always nice to see.