Description

In the distant future, humans have colonised all 8 planets in our solar system. You work for Stomble, a shipping company trying to expand their operations to the whole solar system. You have been asked to develop a system to manage the logistics of Stomble's fleet of spaceships.

Requirements

You will develop a REST API that will store information about the location of the spaceships and the locations in which they are stationed. To accomplish this, your API must fulfil the following use cases:

- **Add** spaceships.
- **Update** the spaceship status: to one of the 3 possible states.
- **Add** a location.
- **Remove** spaceships.
- **Remove** location.
- **Travel** functionality: The travel function involves a spaceship moving from one location to another. Before carrying out the **travel** transaction, the spaceship should be ensured to have the status of operational.

Data Models

Spaceship:

A spaceship should include its name, model, and status. The status of the spaceship can be of 3 different states: decommissioned, maintenance or operational.

Locations:

A location should include its city name, planet name, and the capacity of said location (how many spaceships can be stationed at this location simultaneously).

Other data may be used to complete this task and help with storing and receiving the data.

Notes

- Feel free to use the programming language you are most comfortable with.
- You do not have to worry about coding a front end. However, you do need a way to show that your API functions correctly. We suggest using postman to show this and help with testing if you do not have a tool for preference, but anyway that you prefer is fine.
- Code must be placed in a public git repository (e.g., GitHub, GitLab), do not forget to use meaningful commit messages.
- Spaceship fuel levels do not have to be accounted for, and one can assume that they can travel to any location in a single trip.
- To simplify the task and avoid concurrency issues, travel from planet to planet is so fast that the time taken can be ignored (i.e. travel happens instantaneously).
- All spaceships must be stationed in a hangar somewhere, regardless of their status.
- Consider documenting and testing your API.
- If time permits, use a database.
- Here is a link to a tutorial to help set up a local server to create a Rest API with node.js/Express and MongoDB. This task does not care about the actual creation of the server, but it is needed to run and test your API endpoints.
    - https://www.codementor.io/@olatundegaruba/nodejs-restful-apis-in-10-minutes-q0sgsfhbd

- Feel free to include any planning that was done/additional information within your git repository for reference (not necessary for the task to be considered complete :D)