### Background

A large coffee plantation community farm in Brazil has purchased a few autonomous tractors and drones to save labor and improve yields and operations. Managing all these autonomous vehicles remotely is challenging. For that purpose, we need your help in writing a frontend for our new revolutionary RaaS (Robot-as-a-Service) platform dashboard.

#### 

#### 

#### 

#### 

### Assignment

We would appreciate your help in building a slim working code project implementing a very basic frontend with one Vehicles page.

We want to see your coding skills, but we do not want to take up much of your valuable time. For that reason, we have limited the requirements to these basic few:

1. Build a one-page **responsive** app with a top bar and map area with **“moving” vehicles**, as illustrated below using React.js
2. bonus point: create application according to clean code principles and proper component structure
3. bonus point: use the latest react features (e.g. hooks)
4. extra bonus point: use typescript
5. Use any icons/assets/colors you see fit, from our site or elsewhere, no need to stick to the mockup exactly, but please stay roughly around those lines.
6. Create map using leaflet and react leaflet libraries
7. bonus point - use some advanced features of leaflet of your choice (e.g. clustering, zoom control etc)
8. Create a tiny Node.js RESTful API service for getting vehicle entities so you can test “end-to-end”. Just one GET API is enough, use hard-coded mock data, no DB required.
9. This is how a **Vehicle** object should look like:

{

"\_id": "1d7df7a80139c9c0312b1121",

"type": "tractor", // tractor | drone

"createdAt": "2020-03-11T17:33:53.119Z",

"updatedAt": "2020-03-11T18:33:53.119Z",

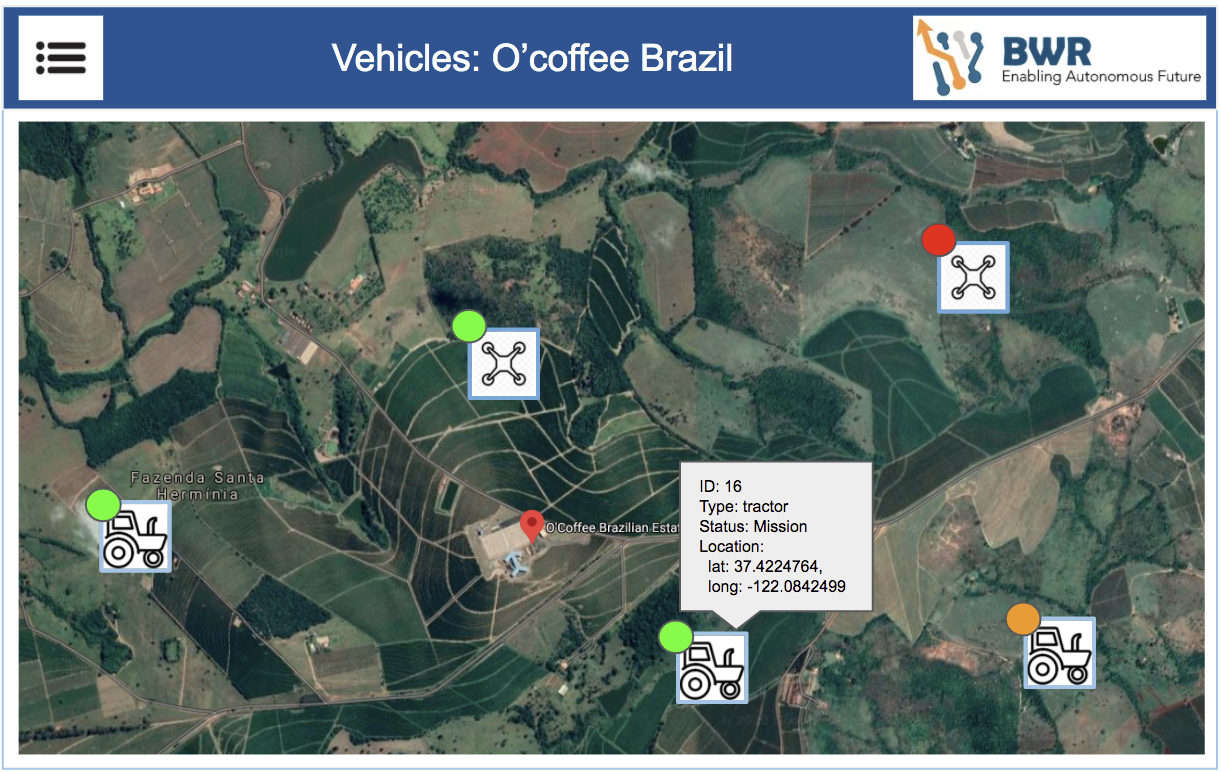
"status": "on-mission", // unreachable | idle | on-mission

"location": { "latitude": "37.4224764", "longitude": "-122.0842499",

"updatedAt": "2019-09-15T20:14:02.877Z" }

}

1. App should periodically refresh by getting A/B mock location data to show “movement” (fake is OK).
2. When hovering over a vehicle or clicking on it, a tooltip should open showing some data
3. Put your code on Git so we can check it out and comment on it.
4. Package it so that we can run it and share it with us.
5. Create a basic README with the project to explain how to run it.



Feel free to contact us for any questions or suggestions.

**We appreciate your time and energy, and we will do our best to make the process as efficient and pleasant as possible. Thank you!**