Product Backlog for the Food Truck Application

Product Backlog: This is a work in progress which will be constantly evolving based on collaboration, negotiation, customer feedback. The teams can stay focused on the MVP which we will agree upon discussions and improve things on priority with a quick measurable feedback loop to understand if the right things are being worked on

This backlog is created based on certain theoretical assumptions. These backlog items will be refined/changed based on team discussions and stakeholder communication. This is no means a set of concrete requirements, this is just some points to enable us to start having a communication during our planning and architectural discussion spike. This Backlog is a continuous work in progress.

Few Backlog Items are below:

1. As a user I want to be able to use a mobile app in IOS and discover food trucks with approved permit for my given location

**Details**: Location provided by latitude and longitude, address, or use current location. The app should ask the user permission to use the current location

**Acceptance criteria**: I need to see at least 5 food trucks in a table view in order of nearest to furthest with a variety of filters such as cuisine type, open timing, hours of operation.

Display detailed mockup needs to be provided.

1. As a user I want to be able to use a mobile app in Android and discover food trucks for my given location

**Details**: Location provided by latitude and longitude, address, or use current location

**Acceptance criteria**: I need to see at least 5 food trucks in a table view in order of nearest to furthest with a variety of filters such as cuisine type, open timing, hours of operation. A nice to have feature will be to have the menu link if available

Display detailed mockup needs to be provided.

1. As a user I want to be able to use a web browser and discover food trucks with approved permits for my given location

**Details**: Location provided by latitude and longitude, address, or use current location

**Acceptance criteria**: I need to see at least 5 food trucks in order of nearest to farthest with a variety of filters such as cuisine type, open timing, hours of operation,. A nice to have feature will be to have the menu link if available

The App must run in all latest web browsers.

In the manual testing- We need to see a table view and a map view. More details on the various paths will be provided after discussion with the Development team.

1. As a user if I pick a food truck I should have a option to get directions to get to it
2. As an app developer I want to get the food truck data from the San Francisco Food Truck Open data Set. A common data service that various views (Web, Mobile, CLI) can use.

**Details**: An App Token needs to be created to get the data. The details around how frequently the Master data is refreshed needs to be understood and based on that a cache might need to be maintained. We might also store the data in a local database with some indexes around geo hash and other filter criteria

1. As a user I can create an account and save my preferences for alerts and notification when preferred food truck is near my location

**Details** for preferences- Cuisine type, location, favorite vendor

**Acceptance criteria**- Based on my preference when a preferred food truck is near my location if I

have logged in then I should be able to get notification

1. As a user I should be able to opt out from sharing my personal information

Non-Functional Requirements

1. The systems should be able to support 800 concurrent users during peak lunch hours
2. The system should return search results within 20 seconds
3. the system should be scalable to other locations (cities) or food with minimal changes
4. The frontend, rest endpoint and other microservices should be containerized so that they can be independently deployed.
5. My REST service needs to have consistent API definitions with proper documentation
6. Error details should be user friendly.
7. The application code should be in a centralized location, with continuous
8. automated tests build and deployment.