

Migration from CLI to Web Application

The purpose of this document is to provide an overview on the migration from the Proof of Concept to a fully featured web application and will identify a design approach, the best suitable technology and identify UI designs in regards to its technical implementation.

System Overview

1. Similar to the CLI, we would be using a RESTful service provided by the Socrata API to access the database for this application.
2. The application will be developed using the React js Library to create interactive UIs components and manage their own state . I'm choosing the framework because it boots productivity, its reusability, fast rendering and the strong community of users for resource sharing. Components such as logging, registration, finding a food truck, map based on the users location and chat will be considered.

Architecture

The system architecture describes how the functionality and the responsibilities of the system would be partitioned and assigned to components. The system is categorized into two sections: RESTful service and Web Application but the main focus will be on the web Application which will consume the RESTful service.

The web Application has the following components:

1. **Logging/ Registration:** This allows us to keep track of the number of registered users currently using the app. It also allows us to keep track of the user's preferences.
2. **Home Page:** Page the user sees once they visit the page, there can search for food trucks, save preferences and rate food trucks.
3. **Find a food truck:** rendered on the home page to search for trucks available
4. **Popular food truck searches/ratings /Meal Preferences:** rendered on the homepage to save user's preferences/ show popular searches
5. **Map/Geolocation Component based on user's location :** gets the users location and renders the food trucks closest to the user in order of ratings. Based on the API, we can do a conditional check to compare the user's current location to the locations of the food trucks.
6. **Footer:**

Administrator Module

1. Need to have a login portal for the administrator to have access to the web Portal.
2. This would allow the user to Delete/ Add/ Edit user profiles
3. Monitor users activities (Comments, Reviews)
4. Option to update password for user.

ScalingIt is important to consider the scaling when it pertains to web performance as traffic increases. Using tools like Real user Monitoring to capture and analyze user interaction and Content Delivery Network to decrease bandwidth and total requests by a high percentage.

Future Work

1. Scalability
2. Including Chat Services.
3. SMS verification.
4. Social Networks embedded signups.tw
5. AI Driven Preference classifier.