--

***Software Requirements***

***Specification***

***for***

***StreetFoodLove***

***Version 0.6 approved***

***Prepared by Silvia Casaburi, Makoto Emura, Wenhuan Tan, and Colin Zhou***

***FoodApp***

***03/01/2022***

***Table of Contents***

# *1.* *Introduction ......................................................................................................................5-15*

## 1.1 Problem statement ...........................................................................................................................5 1.2 Summary ...........................................................................................................................................5 1.3 Product Scope .............................................................................................................................6-14

*1.3.1 In scope ....................................................................................................................................................6-12*

*1.3.2 Out of scope ...........................................................................................................................................13-14*

***1.4******References ..........................................................................................................................................15***

# *2.* *Overall Description ..........................................................................................................….16-21*

***2.1*** ***Compete analysis .............................................................................................................................16******2.2***

***User Classes and Customer Profile ................................................................................................17******2.3*** ***Design and Implementation Constraints ..................................................................................18-19******2.4*** ***Assumptions and Dependencies ................................................................................................20-21***

# *3.* *Specific Requirements ..........................................................................................................24-35*

***3.1******User Interfaces............................................................................................................................24-28******3.2******Functional Requirements ..........................................................................................................28-32******3.3******Logical Database ………………………………………………………………………………….32******3.4******Performance and Software Quality Requirements ...................................................................33-35***

**4.****Breakdown of work/ Project timeline plan .......................................................................36-38**

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | ***Date*** | ***Reason For Changes*** | ***Version*** |
| Silvia Casaburi, Makoto Emura, Wenhuan Tan, and Colin Zhou | *11/2/21* | *This is the initial version* | *0.1* |
| Makoto Emura | *01/18/22* | *Update supported browsers* | *0.2* |
| Silvia Casaburi | *03/01/2022* | *Update Documents for Review* | *0.3* |
| Makoto Emura | *03/05/2022* | *Update estimated costs* | *0.4* |
| Makoto Emura | *05/12/2022* | *Add user story 35 and update price estimate* | *0.5* |
| Makoto Emura | *05/22/2022* | *Update third-party dependencies to match SDD* | *0.6* |
| Wenhuan Tan | *05/31/2022* | *Update the User Experience, Functional Requirement and Breakdown Timeline plan* | *0.7* |

# 1. Introduction

***1.1 Problem statement***

As street food is becoming more popular and new street food businesses are popping up, it is also becoming harder for customers to pick from a large variety of options. Furthermore, it is challenging to track down food trucks as many of them change locations nearly every day. As a result, food trucks that frequently change locations lose a high percentage of customers. When a person’s favorite food truck is not in town, they can be hesitant to try new street foods as it is hard to find reviews of street food online. Reviews for street food vendors are hard to find in many existing services such as Yelp because they focus on restaurants and popular bars.

Street food businesses often misjudge food costs and inventory because no resources or data are shared with business owners. Such vendors often struggle against competitors who have access to marketing data.

## 1.2 Summary

StreetFoodLove is a service that helps users discover new street food in the Seattle area. Users will find the reviews and the search feature to be very helpful in deciding where to eat next. People can get updates on their favorite food trucks by visiting their vendor page or by receiving notifications from specific vendors. Users do not need to sign up for the service unless they want to review vendors or receive notifications.

Vendors must go through an approval process and agree to a monthly fee to get a vendor account, which is necessary to create their vendor page. In addition, they will gain access to the vendor interface which displays many useful charts and statistics, such as the most popular neighborhoods. Through the vendor interface, they can update their business info including their location and business hours.

The main goal of the StreetFoodLove project is to build a community that benefits both the customers and street food vendors.

In comparison to other review services, StreetFoodLove only focuses on street food. Other sites like Yelp use ranking algorithms that are biased towards big establishments like restaurants and bars, while reviews for street food are either hidden deep in their search listings or not found at all.

## 1.3 Product Scope

***1.3.1 In scope***

***User Experience 1: Street Food Business Owner***

Street food business owners can sign up for an account with their business information such as name, email, cuisine types, business areas, business hours, location, website, social media webpage, and contact information. After they click the “Finish” button, a street food vendor account is created.

After a vendor account is created, business owners will be able to get access to the vendor dashboard. On the vendor dashboard, a vendor page will be set up for them to display their basic information. If they need to change or update their information such as locations and business hours, they can go to the Customize Vendor Page section to make any updates. They can also upload photos/videos of the food items, menu, and outside look to the photo/video libraries in the Upload Pictures section. They will receive notifications when a new review from the customer is uploaded to the New Reviews section. They can also respond to customers’ reviews. Instructions/Guides for opening a new street food business or operating your existing business will be provided for them in the Guide section. In the Trend section, the dashboard will display 5 analytic graphs: Top 3 cuisine types in a certain area, Top 5 searches in the last month, Top 10 vendors in a certain location, new reviews in the last month, and the Average rating overtime. In the Guide section, you will find the most popular guides and most recent guides that we selected for you for your references.

***User Experience 2: Customers***

Customers can log in to our web application by signing up for a new account or continuing with their google account. After they logged in, if they click on their username, a drop menu will be displayed, and it will navigate them to the different pages: the profile page; a page called vendors near you, the help page, and the log-out page. Customers will see a map of street food vendors nearby immediately after they enter our website. By clicking the marks of the vendors on the map, they will see a shortcut to their vendor information: vendor name, average rating, total reviews, and a profile picture. When the customer clicks the shortcut, they will see detailed information about the food vendor such as location, business hours, photos/videos, website, social media links, contact information, and description of the vendor. If they are interested in going to the food vendor sites, they will get navigation help by clicking the “Map” section. After they purchase the food, they can upload a review with pictures or videos of the food they have purchased to show how they like/dislike the food.

They can click the star-shaped “Star” button on the vendor page to show they are interested in this vendor, and they can cancel it by clicking it one more time. By the “Star” button, a number shows the total Star amount that the vendor received. Under the “Star” button, a logo of Twitter will allow users to share the vendor page to their social media.

They can also search for specific food items by entering keywords such as cuisine type, main ingredients, and name, and then a list of food vendors will be displayed on the map. They can also filter and sort the results by price and cuisine types. Users will receive food/vendor recommendations based on their search query history and favorite vendor lists after they give queries to search for food at least 10 times and add at least 3 vendors to their favorite vendor list.

If a vendor has enabled the discount exchange function on their Customize Vendor Page, the customer will get a QR code after they give a review. The QR code can be scanned to get a discount on their next purchase. The QR code will be added to a QR code list on the customer’s profile page, and it will be deleted after it is scanned once.

**Criteria 1:**

*Given I searched street food vendors nearby*

When I enter keywords such as "**Pizza**" and click the "**enter**" button

A list of street food vendors will show on the map

**Criteria 2:**

*Given I searched street food vendors nearby*

When I click "Filters"

I can choose different filters, such as Price range and Cuisine Types to filter my search results.

**Criteria 3:**

*Given I have shared my location with the application*

By clicking "Vendors Near You" page in the dropdown menu of my username on the right top corner

A list of vendors near me will be shown on a table and if I click any vendor name in the table, I will be navigated to the vendor’s vendor page

**Criteria 4:**

*Given I see the locations of street food vendors on the map*

When I tap/click on the name of a food vendor on the map

The vendor’s page will be shown.

**Criteria 5:**

*Given I look at the profile page of the street food vendor*

When I scroll down

I will see detailed information about this food vendor: business hours, contact, websites, social media webpage, latest location, average review rating, total review numbers and so on.

**Criteria 6:**

*Given I am interested in driving to the vendor location to get some food*

When I click the "Map" section

Then I will get a navigation map to guide me to their location.

**Criteria 7:**

*Given I have browsed some food vendors*

If I like this vendor and I want to browse their vendor page again

Then I can click the “Star” button by the vendor's name on the vendor’s page, and I can check this vendor again on the “Vendor Starred” section of my profile setting page

**Criteria 8:**

*Given I have tried the food*

If I want to leave a review

Then I can click the attachment button to upload pictures or videos with my text to show other users how much I like/dislike the experience and food.

**Criteria 9:**

*Given I leave a review*

When other users ask me more questions about the food item

Then I can click the button on the right corner to respond to them to share more information

**Criteria 10:**

*Given as a user who wants to create an account*

When I want to log into the App

Then I can sign up with my email by clicking the "sign up" button or logging in with my google account by clicking the "Continue with Google" button

**Criteria 11:**

*Given that as a street food business owner, I am interested in joining StreetFoodLove*

When I want to create a vendor account

Then I can sign up for an account with my email by clicking the "Sign up" button and filling out a form with my business information.

**Criteria 12:**

*Given whether I like or dislike the food*

When I want to share my review of the food on my social media account

Then I can click the “Twitter” logo to share the link of the street food vendor on my social media account.

**Criteria 13:**

*Given I have a vendor account*

When I log into my account

In the dropdown menu of my account, I can find the Vendor Dashboard page to get into my dashboard. On the dashboard page, I can see 6 sections: Vendor page, Customize Vendor Page, Upload Pictures, Trends, New Reviews, and Business Guides.

**Criteria 14:**

*Given that I am on my dashboard page*

When I want to edit my vendor page

Then I can click the "Customize Vendor Page" section to update my vendor information, such as location, business hours, cuisine types, and enabling discount exchange function and so on

**Criteria 15:**

*Given that I am on my dashboard page*

When a new review is added to my profile

Then I can click the "New Reviews" section to look at the latest reviews and respond to the customers if desired

**Criteria 16:**

*Given that I am on my dashboard page*

When I want to upload more pictures to my photo gallery

Then I can click the "Upload Pictures" section to upload more pictures to the photo gallery

**Criteria 17:**

*Given that I am on my dashboard page*

When I want to know the local business trends information

Then I can click the "Trends" section, I can find 5 analytic graphs: Top 3 cuisine types in a certain area, Top 5 searches in the last month, Top 10 vendors in a certain location, New Reviews in the last month, and the Average rating overtime

**Criteria 18:**

*Given that I am on my dashboard page*

When I want to get some help with business operation or set up new street food businesses

Then I can click the "Business Guides" section to look at the most popular and latest guides articles to get some instructions

**Criteria 19:**

*Given as a street food business vendor*

When I want to set a photo as my profile photo

Then I can click the “Profile Setting” page in the dropdown menu by clicking my vendor account, the account setting section will allow me to upload a picture to set as my profile picture

**Criteria 20:**

*Given I have signed up on the App*

When I want to change my password, edit my account info and so on

Then I can go to the account settings page to update my information

**Criteria 21:**

*Given I have left a review to a vendor and the vendor has enabled discount exchange function*

When I submit my review

I can get a notice that tells me a discount QR has been added to my discount list on my profile page

**Criteria 22:**

*Given I have a QR code for a food vendor*

When I go to the food vendor to purchase food

Then the food vendor can scan the QR and give me a discount (the food vendor decides the discount range) on the item I purchase. After that, the QR code will be deleted from my QR code list.

**Criteria 23:**

*Given as a subscriber, I am interested in opening another street food business*

When I want to decide on business locations and set up my new business

Then I can reference the trend and business guide information provided on my vendor dashboard

**Criteria 24:**

*Given as a street food vendor, I will change my location frequently and share with the users*

When I change to a new location

### Then I can go to the “Customize Vendor Page” section to update my location by choosing “Use my GPS coordinates” even though I do not know my new address and customers will see a timestamp of the update, such as "Updated 3 minutes ago" on my vendor page

**Criteria 25:**

*Given as a customer, I have been using the StreetFoodLove web app for a while. I have searched for food/vendors at least 10 times and added 3 vendors to my favorite list.*

Then I will receive food/vendor recommendations that are like what I have searched for or shown interest in.

**Criteria 26:**

*Given I am a food vendor*

When I set my business address

Then the marker on the map representing my business changes position to that of my new address.

### 1.3.2 Out of Scope

**Criteria 27:**

*Given I am interested in purchasing an item*

When I want to order food online

Then I can place the order or pay for it

**Criteria 28:**

*Given as a prospective business owner, I am interested in opening a street food business*

When I want to decide on business locations and get my new business certification

Then I can rely on the app to make the decision or get the new business certification for me

**Criteria 29:**

*Given I have a vendor account and I want to promote my business*

When I pay for advertising

Then my business will be featured as a sponsored listing in search queries and the map

**Criteria 30:**

*Given as a food vendor and subscriber and I want to delete the bad reviews or improve my review rating*

When I pay for this service

Then I can delete all the bad reviews and improve my review rating

**Criteria 31:**

*Given I am given discounts/vouchers after I give a review*

When I have used the discount/vouchers one time

Then I can use the discount/vouchers multiple times

**Criteria 32:**

*Given I am a produce seller, and I am not considered as a street food vendor*

When I want to sign up for a food vendor account

Then I will sign up successfully

## 1.4 References

* Yelp-like app features list - <https://www.idea2app.dev/yelp-app-clone-development.html>
* W3C Web Content Accessibility Guidelines 2.1 - <https://www.w3.org/WAI/standards-guidelines/wcag/>
* Browserslist (As of 10/27/21) - <https://github.com/browserslist/browserslist>
* AWS documentation - <https://aws.amazon.com/>
* The Reasons to Use MySQL - <http://www.databasequest.com/index.php/product-service/mysql-dbquest>
* Why Food Trucks Fail - <http://restaurantmba.com/food-trucks-fail-part-2/>
* Google Maps Platform - <https://developers.google.com/maps/documentation/places/web-service/details>
* Why Use MongoDB and When to Use It? - <https://www.mongodb.com/why-use-mongodb>
* To create ER Diagram - <https://lucid.app>
* Yelp official website - <https://www.yelp.com>
* StreetFoodFinder official website - <https://streetfoodfinder.com>
* Google Map official website: - <https://www.google.com/maps>
* TripAdvisor official website - <https://www.tripadvisor.com/>
* Streefoodapp.com website - <https://streetfoodapp.com/>
* Amazon RDS DB instance storage documentation - <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_Storage.html>
* AWS latency test - <https://ping.psa.fun/>
* What is JavaScript? - <https://www.typescriptlang.org/why-create-typescript>
* What’s so great about Go? - <https://stackoverflow.blog/2020/11/02/go-golang-learn-fast-programming-languages/>

# 2. Overall Description

***2.1 Complete analysis***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Top Competitors** | **Strength** | **Weakness** | **Opportunities** | **Threats** |
| **Yelp** | Well-known, nice UI design, popular and has a strong user base | Not specific for street food vendors and street food vendors are easily ignored by users because users are more likely to view restaurants. No instructions or analysis about street food business available | Services specifically provided for street food vendors are still not highly developed. | If Yelp starts paying more attention to street food vendors and develops more features for them, it will take users away from us. |
| **Google Maps** | Well-known, popular | Not suitable for mobile food vendors. No instructions or analysis about street food business available | Services specifically provided for street food vendors are still not highly developed. | With Google Maps, customers may not be interested in downloading another app to look for street food specifically |
| **StreetFoodFinder** | Specific for street food vendors and customers who are looking for street food | No detailed information of street food vendors, no reviews, no business analysis information. Limited number of cities. No tracking of mobile food vendors. No ability to share on social media. | More services and functions can be developed to help more street food vendors and customers | If StreetFoodFinder realized what they could do to provide a better service, it may affect our user groups. |
| **TripAdvisor** | Gain attention from users before they travel to the destination cities | Not specifically made for street food vendors. We can’t find street food easily. | TripAdvisor has not gained a big customer foundation in the street food business | If TripAdvisor pays more attention to street food vendors, it may take users from us |
| **Streetfoodapp.com** | Specific for street food vendors and street food customers. Update location and business hours info continuously | No detailed information of street food vendors, no reviews, no business analysis information. Limited number of cities. No ability to share on social media. | Not well-known | If they develop their mobile apps and add more useful features, it may take users away from us |

## Summary: Information about street food vendors is still limited online, and challenges are for customers to find street food and for street food vendors to gain more customers. Instructions/tips and street food businesses related analysis is even more limited. As a high-demand, low-investment, and easy-startup business, street food businesses should be developing faster than they are now. A street food business will help new immigrants start a new life in the USA, and customers will have more choices for food.

## 2.2 User Classes and Customer Profile

## We expect users to be of any demographic who are interested in finding street food near them. Some users would want to share their reviews of the food they recently ate. Regular users are not expected to be tech-savvy and are expected to range from teenagers to the elderly. We anticipate that there will be users with disabilities. Users may be using laptops or desktop devices of any screen size.

## The second group of users will be existing street food business owners. They will be expected to utilize the dashboard and monitors on our platform to track the status of their business. These data analytics features of the app will be more appealing to business owners.

## We also want to attract people thinking of starting their own street food business who might need guidance. Understandably, these people may not be well-versed in technology or how to manage a business. Therefore, the app will assume a basic level of fluency in English and will attempt to help those unfamiliar with business management by providing tools and guides. However, regular users will not be able to access the data and charts that business owners will see.

|  |  |  |
| --- | --- | --- |
| Users | Description | Useful Features |
| Regular Users | Any demographic, of all ages may include people with disabilities. Not expected to be experts in technology usage. | Will be using maps to navigate around and reading the reviews on the user side. Also, we will be making accounts and uploading photos in the reviews |
| Vendor Owners | Any demographic who is interested in seeing the performance of their business. | Will be using the dashboard and monitors on the platform to track their business status. Mostly using data analytics tools provided. |
| Interested vendor owners | Any demographic, who may not be well versed in technology or business principles. Assumes a basic level of fluency in English. | Will be using the tools and guides provided in the users' dashboards to get started and educated on how to start such a business. Will have access to all features of regular users. |

## 2.3 Design and Implementation Constraints

## Portability with various devices and web browsers is essential. Some users will use the app on a desktop platform, and others will use it on a mobile platform. Users will view the app through different web browsers of various versions. Not only does the JavaScript code need to be compatible with various web browsers, but the UI needs to be usable on monitors and phone screens.

## Not all users have fast internet speeds. Some users will load the app with a slow cellular network.

## Some users have disabilities and will use assistive technologies to help them navigate through the app.

## Four developers will be working on the source code simultaneously, so a version control system is necessary.

## Slow UI response times will make the interface frustrating to use. This may deter customers from using our service.

## For the beta version, up to 100 users can access the API concurrently.

## If our servers are down, the app is unusable.

## The database and media storage must have the capacity to store all application data and images.

## Now, we have a very low budget, so we must rely on free third-party services.

## The MVP can be hosted in a single instance and will not need the ability to scale. This also means that a load balancer is not necessary.

## Tests are necessary to ensure correctness. In a layered web app architecture, integration tests are essential to ensure that all layers work together.

## Logs are necessary to fix server-side and client-side errors after they occur.

## Personal information and account credentials are sent over the internet, so data in transit must be secure.

## We have a strict secure-by-design policy.

## Database records should be backed up as they may be lost or changed inadvertently due to technical issues, human error, natural disasters, etc.

## The final product must be delivered by the end of the spring quarter.

## 2.4 Assumptions and Dependencies

### App dependencies

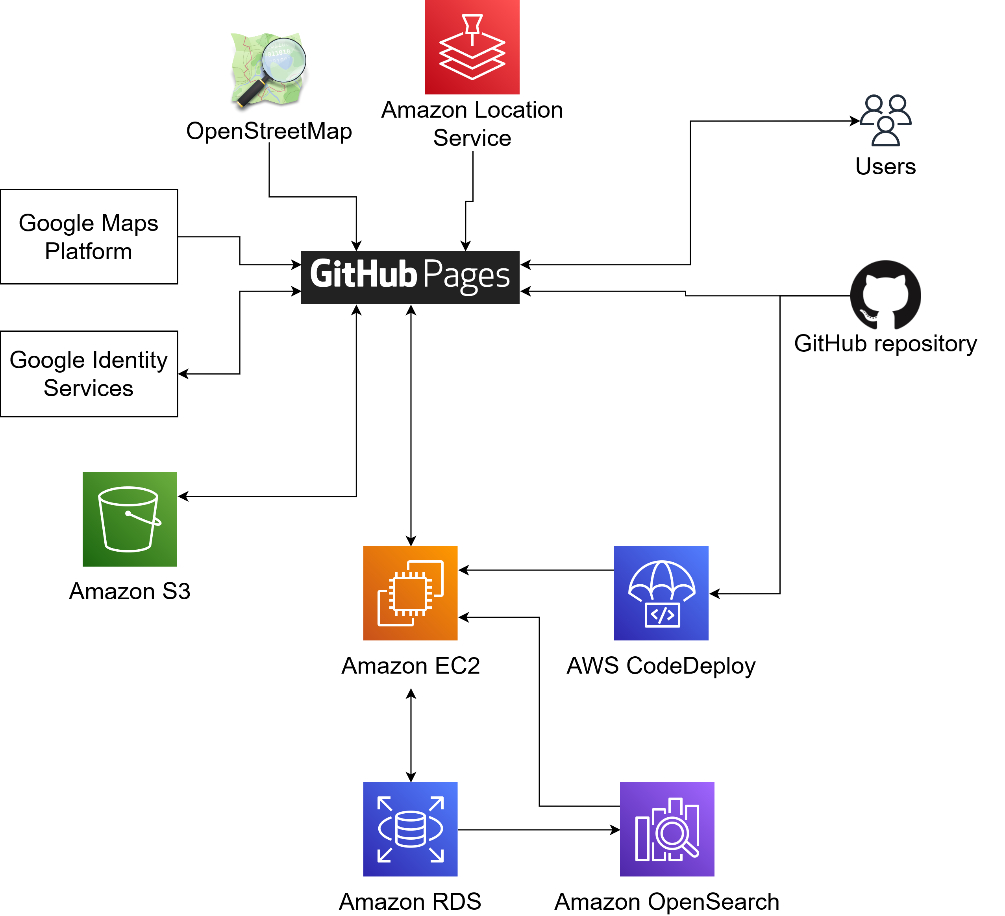
* Languages
  + TypeScript for the front-end app
    - As a superset of JavaScript, TypeScript adds type information to JavaScript code.
    - Type information provides compiler checks that reduce runtime errors. Two very common type-related errors encountered at runtime in JavaScript code are those related to missing properties in objects and those related to null values being treated as an object. These errors are mitigated in TypeScript through type checking.
    - Having type information for function parameters and return values is important for collaboration. Even if the types are defined in a comment, it is easily overlooked by other team members. Type information gives immediate feedback so that teams do not have to double-check that variable are used correctly.
    - Type checks reduce the number of production bugs and therefore reduce the labor needed to resolve those bugs.
    - The downside with the required type of information is that it requires more upfront work, which pays off in the long run.
  + Go for the backend
    - Go is a compiled, statically typed language that is commonly used for web apps.
    - Go has a simple syntax, unlike Java. This allows developers to focus on higher-level ideas rather than on the syntax and the technical details.
    - Go has a vast number of third-party libraries for web development. Libraries will help speed up the development of features. Go also comes with a very easy-to-use and modular package manager, unlike Python.
    - Although performance is not of priority in this project, programs written in Go tend to be faster and more memory efficient than JVM or Python programs. This in turn reduces the number of instances needed to serve a given number of users.
    - There are two cons with Go: Go does not have generics and Go code tends to be more verbose than other languages.
* Database
  + MySQL
    - MySQL is a relational database that has constraints to ensure referential validity and null safety.
    - MySQL was chosen over MongoDB because no team member has experience in MongoDB but we have learned MySQL. MongoDB has some limitations regarding document sizes and the need to correctly index items for optimum performance.
* Libraries
  + Gin: HTTP web framework
  + React: To build the user interfaces
  + Semantic UI: Theming
  + Leaflet: OpenStreetMap API and map presentation
  + Rechart: Data visualization component
  + Redux: To store the user interface state
  + Babel: TypeScript transpiler
  + webpack: JavaScript bundler
* Services for content and notifications
  + OpenStreetMap: Map tiles/data
    - The embedding capability of Google Maps does not allow us to overlay elements on the map, whereas OpenStreetMap provides raw tiles which allow us to overlay elements.
  + Amazon Location Service
    - Geocoding services
  + Google Maps Platform
    - For vendor page map
  + Google Identity Services
    - For Sign-in with Google
  + Amazon OpenSearch
    - Search engine
* Deployment
  + Amazon EC2: Compute instances
    - AWS provides a Free Tier that allows free use of certain services for small-scale projects.
  + AWS CodeDeploy: For deploying to EC2 instances
  + Amazon RDS: MySQL database
  + Amazon S3: Store media that is viewed and uploaded by the users

### Development tools

* Version control and collaboration
  + Git: Source code control and tracking
  + GitHub: Source code host and project collaboration
* Web browsers and test devices
  + Firefox, Chrome, Edge: View and debug the client app
  + PC, iPhone, Android phone: To test how the app would look like and perform with various devices

### System Architecture Diagram

This diagram shows how various services will interact with each other.



* **GitHub** will be used to store the source code and run CI/CD workflows
* On deployment, the client app will be compiled and copied to the **GitHub Pages** static host
* The server code will be compiled and deployed to an **EC2** instance using **CodeDeploy**
* The client interacts with the backend API
* The client requests map data from **OpenStreetMap** and **Google Maps Platform**
* The client uses **Google Identity Services** for the sign-in with Google authentication flow
* The client can view and upload images to an **S3** bucket
* The backend will store content data on an **RDS** database

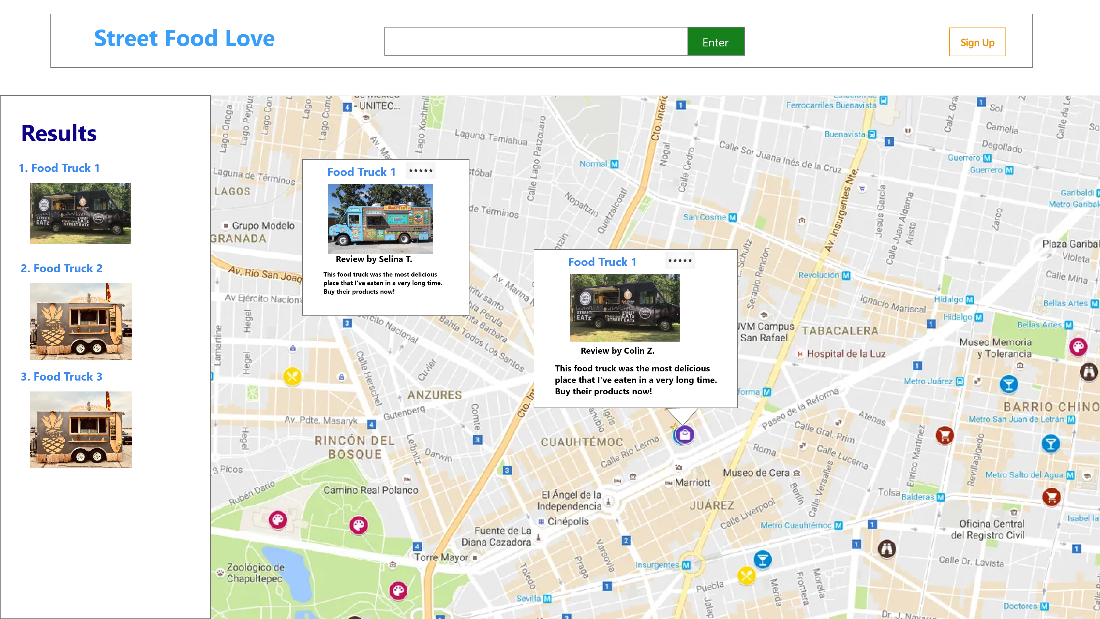
### Estimated costs of third-party services

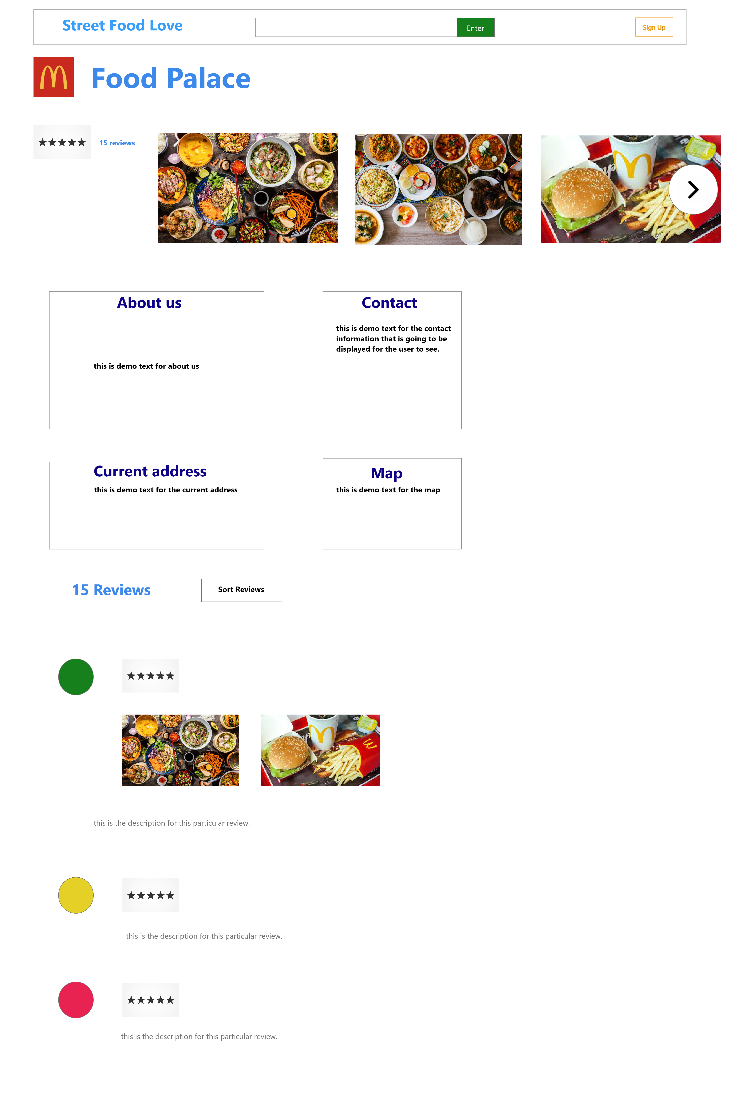
This is a list of third-party services that we will use and their estimated cost per month. We will use the Free Tier for all AWS services. Free Tier provides select AWS services for free for up to one year.

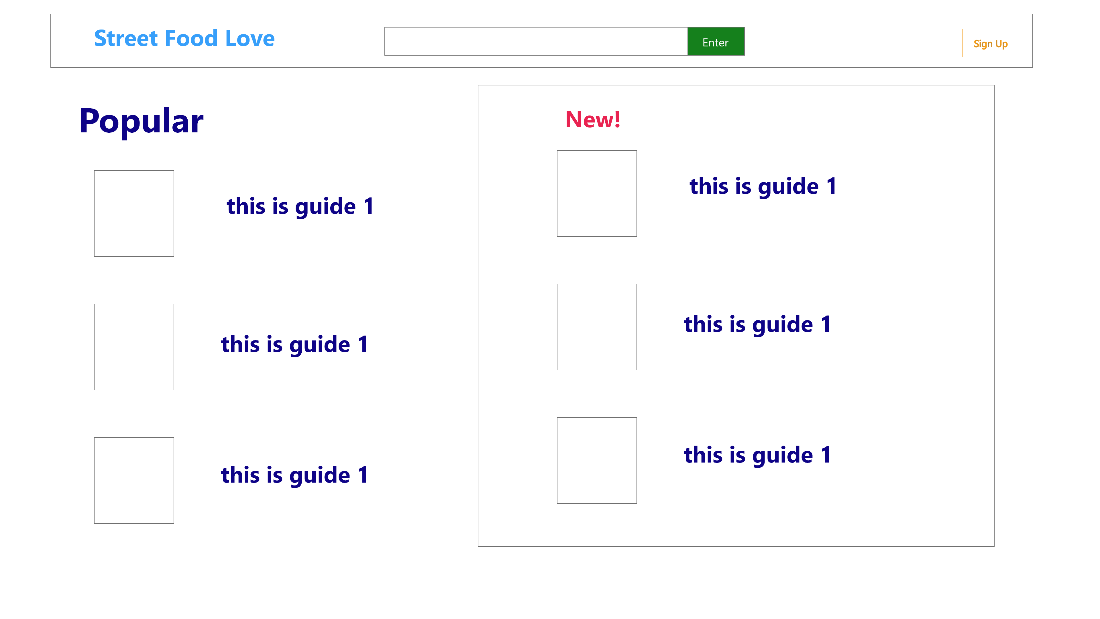
|  |  |
| --- | --- |
| Name of service and requirements | Cost per month |
| AWS CodeDeploy | $0.00 |
| Amazon EC2   * 1 instance * Instance type: t2.micro * Up to 750 hours/month | $0.00 |
| EC2 Elastic IP | $0.00 |
| Amazon Location Service   * 2 months | $0.00 |
| Amazon OpenSearch   * 1 t3 small.search instance * 1 node * 10 GB storage * 1200 hours/month | $5.00 |
| Amazon RDS   * db.t2.micro instance * Less than 1 GB of storage | $0.00 |
| Amazon Simple Email Service   * Less than 50 emails per month | $0.00 |
| Amazon S3   * Storage class: Standard * 0.1 GB per month * 500 total requests per month * Outbound data transfer: 10 GB per month | $0.00 |
| Google Maps Embed API | $0.00 |
| Google Identity Services | $0.00 |
| OpenStreetMap tile server   * Not for commercial use | $0.00 |
| Total | $5.00 |

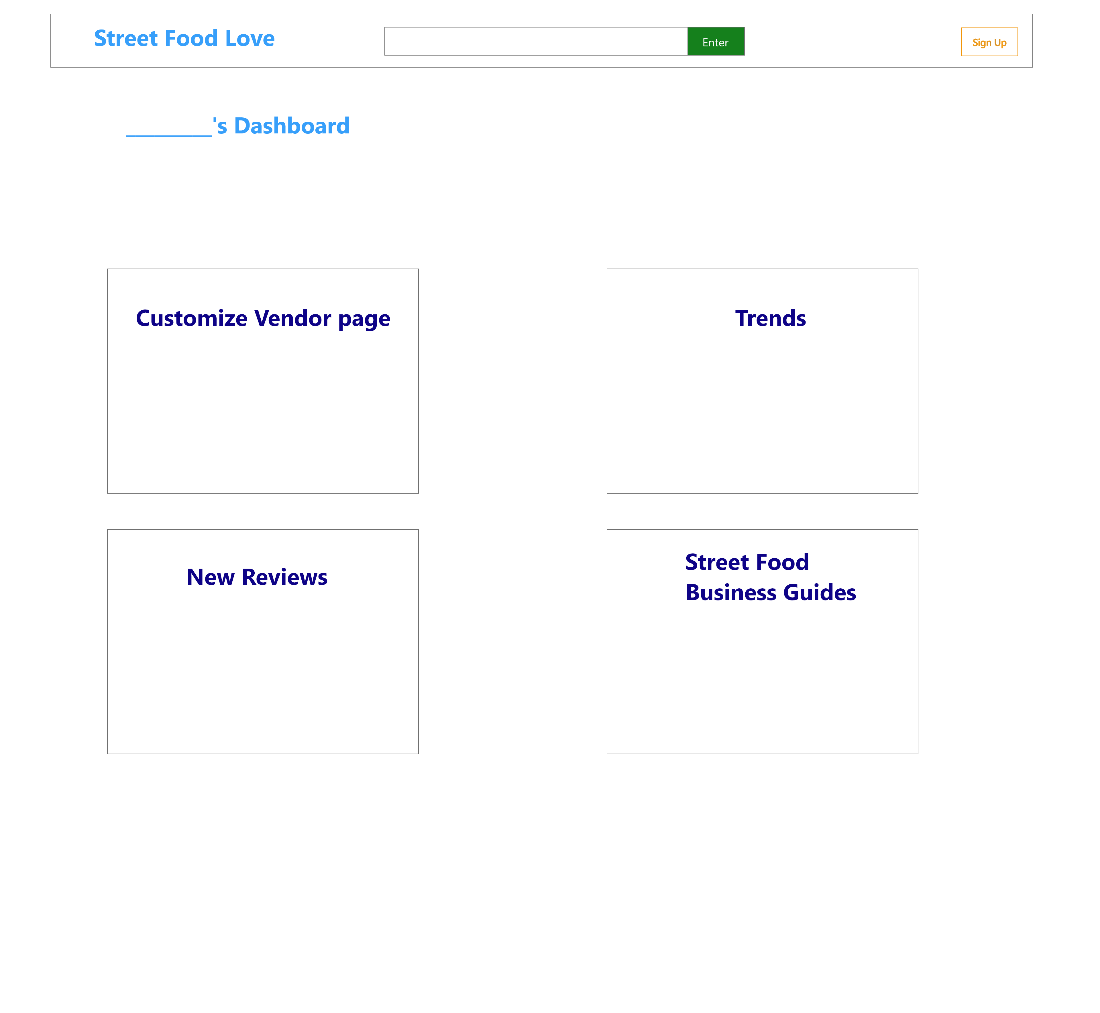
# 3. Specific Requirements

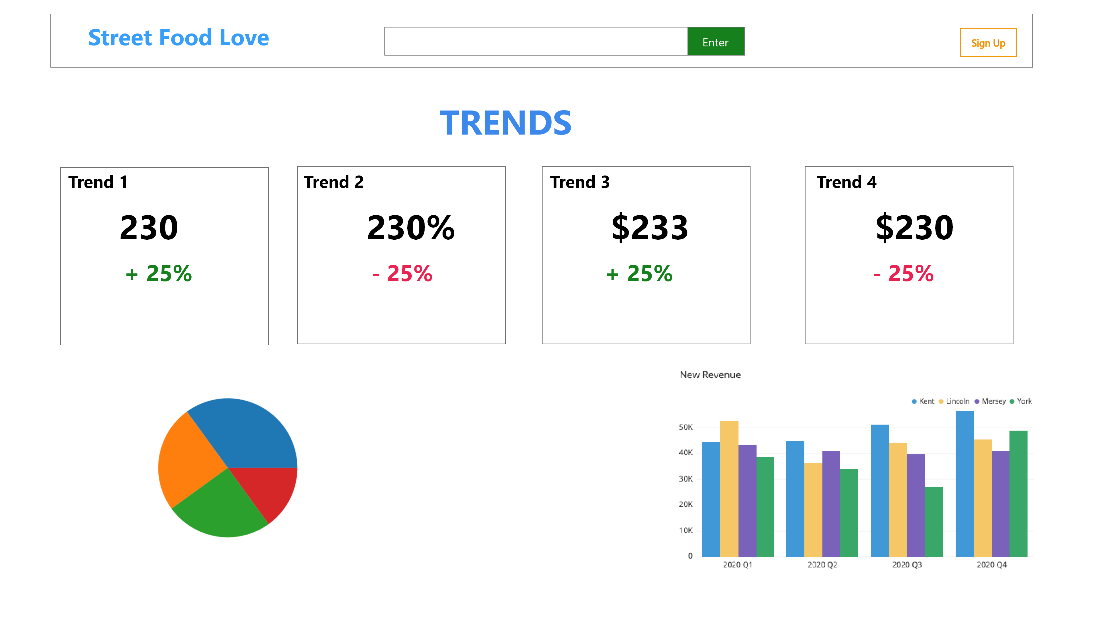
## 3.1 User Interfaces

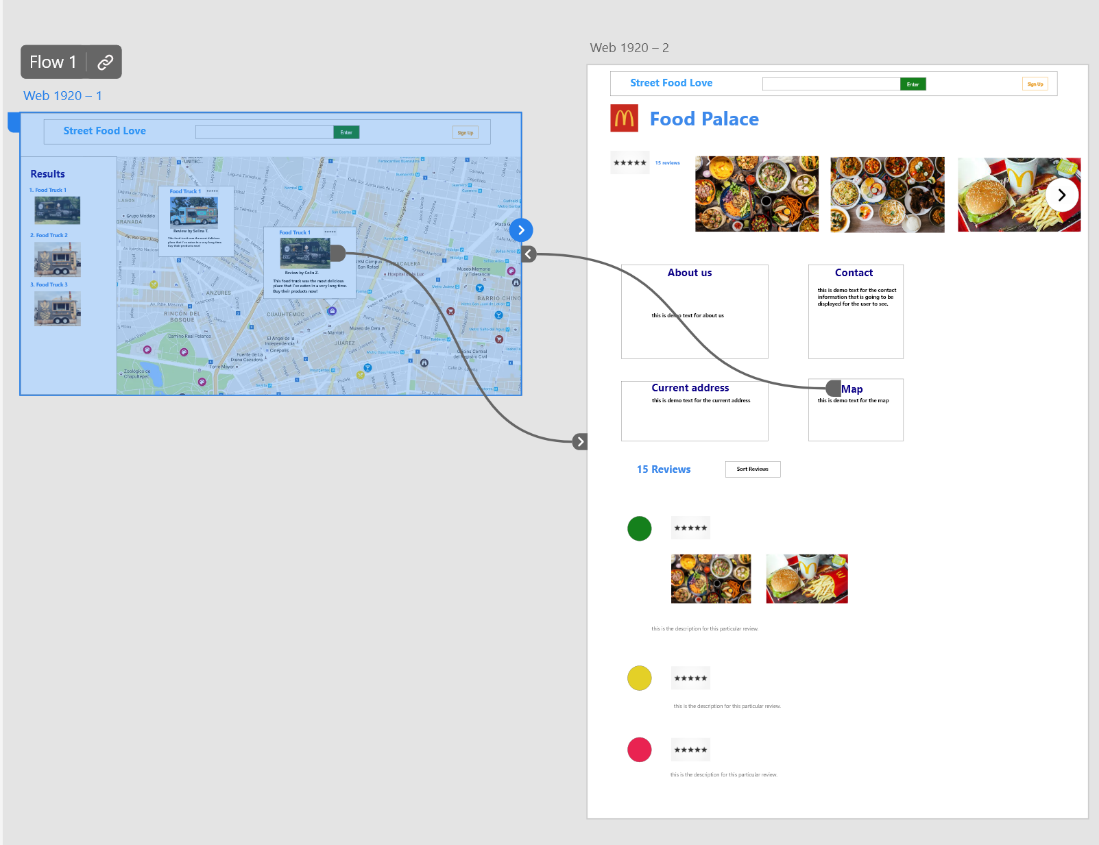


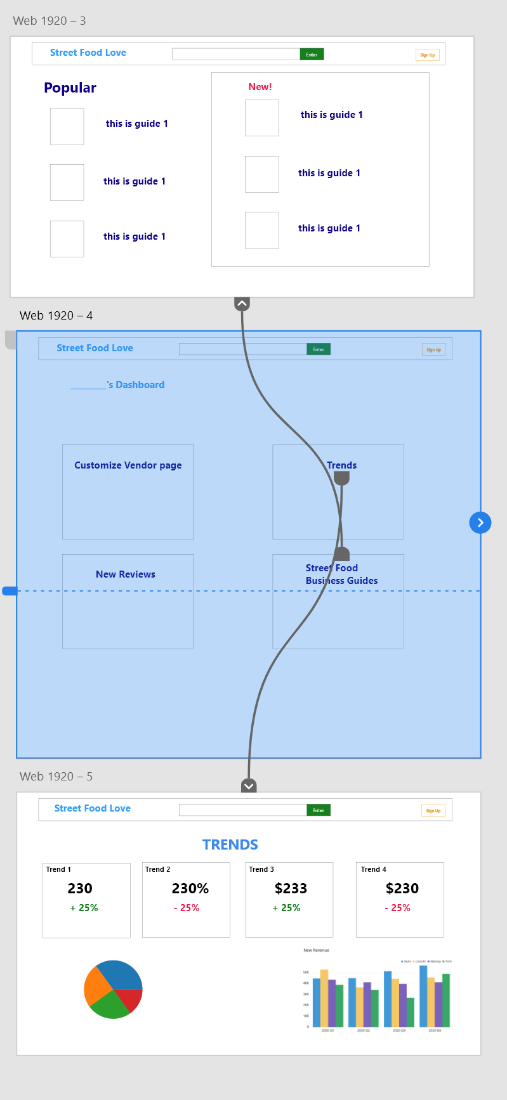


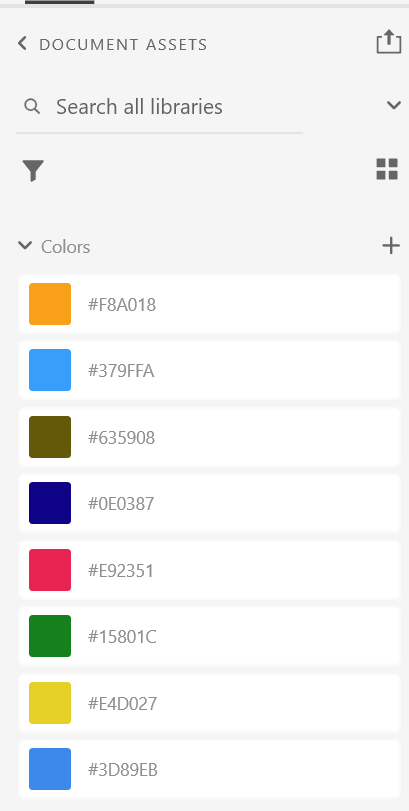












These are mockups of the app user interface. There will be a user interface needed for both the general users and the vendors. The user will be able to see the reviews and the graphical map, while vendors will see the graphs and progress monitors of their business. The graphs will be taken from external third-party libraries.

The colors in the color scheme shown above are the colors appearing on the final product. The purple color will be used to designate the categories for the user side, while black will be used for outlines in the vendor dashboard monitors as shown.

The orange sign-up button will be for creating new accounts on the app, and the enter box at the top will be a general search bar, helping users search for reviews and specific information.

Once the user navigates to the main page of a specific street food vendor, they will see a display of pictures of food, with a grid of business information highlighted in purple. In addition, there will be a map showing their location, which will shrink in size according to the size of the screen. This app will be fully responsive on the web.

Error messages will be displayed in different accents of the color red. This is to help distinguish between whether the error was caused by a network error or a user input error. The bright red and font size will be adjusted to fit this need.

## As this is a web-based graphical app, the only keyboard shortcuts available for the user are pressing “enter” in the search bar and pressing the tab to navigate between different interactive elements.

## 3.2 Functional Requirements

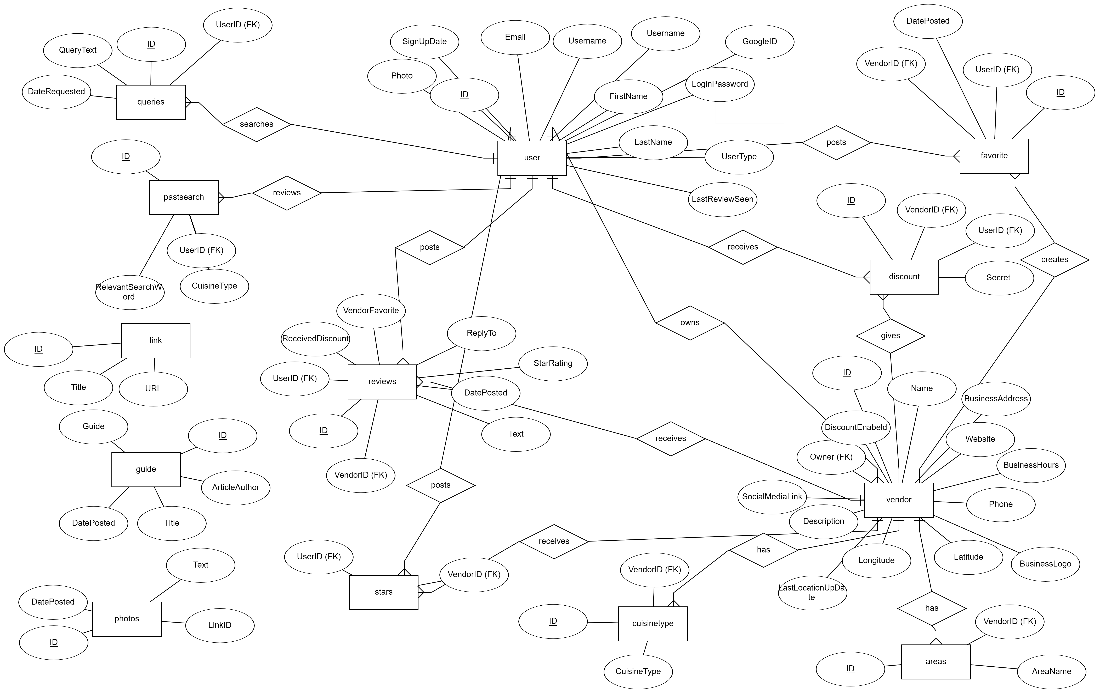
Note:

1. Features are sorted by priority
2. In the Users column, C = Customers and BO = Business Owners

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Features** | **Users** | **Functional Requirement** |
| **1** | **Vendor page** | **BO** | Business Owners can create and edit basic business information on their vendor page such as hours, location, contact information that is visible to other users and customers. The vendor page will also show people’s reviews and pictures/videos. |
| **2** | **Reviews and star rating** | **C/BO** | Customers can give reviews and upload pictures/videos about the food they bought, and Business Owners can respond to customer reviews. Other users can read the reviews and see the average rating of the food vendor. |
| **3** | **Photo Library** | **C/BO** | Business Owners can upload food pictures and videos in the Upload Pictures section on their vendor dashboard |
| **4** | **Sign Up and**  **Log In** | **C/BO** | Customers and Business Owners can sign up for an account with their email address and log into our app after signing up. Business Owners must sign up with their email to get a vendor account |
| **5** | **Map** | **C/BO** | Nearby street food and search results will be shown on the map. Users can click/tap on results to view the result in more detail. The vendor page also has a link to their location on Google Maps. |
| **6** | **Search** | **C/BO** | Users can enter search queries and see relevant results. Users can click/tap on results to view the result in more detail. |
| **7** | **Filter Items** | **C/BO** | Customers and Business Owners can filter search results by price and cuisine types. |
| **9** | **Location tracking** | **BO** | The app can track the locations of business owners with the GPS on their device if they choose to share their location. The time of when the location was updated will be displayed on the vendor’s page |
| **10** | **Star/Favorites list** | **C** | Customers can add their favorite food vendors to a favorites list. They can find them at the Vendor Starred section on their profile page |
| **11** | **Food Recommendation** | **C** | Recommend items to customers that they are likely to buy based on the customers' browsing behavior, search queries, and Star interactions. |
| **12** | **Settings** | **C/BO** | A list of settings for the product. These are: change password, deactivate account, and edit profile. |
| **13** | **Single Sign-On** | **C** | Users can log in using their Facebook/Twitter/Google account. No need to remember usernames and passwords. |
| **14** | **Vendor Dashboard** | **BO** | The vendor dashboard displays 6 sections:   * The vendor page entrance * Customize Vendor Page * Upload Pictures * Trends * New Reviews * Business Guides |
| **15** | **Business Analytics** | **Backend and Front end** | At the Trend section on the vendor dashboard, 5 analytic graphs are displayed:   * Top 3 cuisine types in a certain area * Top 5 searches in the last month * Top 10 vendors in a certain location * New Reviews in the last month * The Average rating overtime. |
| **16** | **Business Starting-Up and Operation Guides** | **BO** | Street Food business start-up instructions/tips/tutorials are available for vendors at the Business Guide section on their vendor dashboard |
| **17** | **Connect with/Share in social media** | **C/BO** | Customers and business owners can share the link of vendor page to Twitter account |
| **18** | **Discount QR code** | **BO/C** | If a business has enabled discount exchange to give discounts in exchange for reviews, the customer will get a QR code after they give a review. The QR code will be added to a QR code list on their profile page, and it will be deleted after it is scanned once. |

## Logical Database Requirements

The database used for this application is MySQL and an Entity Relationship Diagram can be viewed below.



## 3.4 Performance and Software Quality Requirements

* Usability and battery efficiency
  + All fonts in the app must be readable, whether on a monitor or a phone
  + In the mobile layout, the text needs to be readable, and links and buttons must be big enough to tap on
  + Any user interface actions that have response times that exceed 1 second on a slow device have an associated indicator such as a loading icon
  + The client app must not drain the device battery excessively
* Accessibility
  + The font color has sufficient contrast with the background
  + All images have alternative text
  + All pages, except the map, are usable without images
  + Use the semantically appropriate HTML tags
* Development and deployment
  + GitHub will be used to host the source code, and Git will be used as the version control system
  + When a commit is tagged, a GitHub Action will automatically deploy the new version
* Performance
  + The 90th percentile RTT of the API is 0.1 seconds (assuming the user is in the Seattle area)
    - The 0.1 second figure is based upon the following observations:
    - RDS has a typical response time of a few milliseconds
    - A single EC2 instance will be able to handle >100 concurrent users which is the estimated traffic load
      * A t2.micro instance has 1 GiB of memory and a goroutine and session state needs <10 KB
    - The processing period of most API calls would be <2 ms
    - ping.psa.fun reports that the RTT to us-west-2 is 27 ms and us-west-1 is 47 ms
    - The predicted best-case latency would be 30 ms. A ballpark figure for a slow response time is 100 ms.
    - Some resources, such as those for search queries, may take longer to process.
  + On a desktop device, the first contentful paint all pages must be under 1.0 seconds
    - A very simple React app is 0.5 s or less. The FCP of large apps is varied, such as airbnb.com which is 0.6 s, or reddit.com which is 1.0 s. Maps may take longer to render, but OpenStreetMap’s FCP was 0.5 s.
  + Pages must not jitter or flicker excessively when scrolling, both on mobile and on desktop
* Scalability, concurrency, and storage capacity
  + The backend is hosted on one Amazon EC2 instance of type t2.micro which scales to zero instances if unused
  + The client app is hosted on GitHub Pages
  + The relational database has a storage capacity of 10 GB
  + The media storage has a capacity of 5 GB
  + 100 users can access the app concurrently
* Availability and reliability
  + The app has a 99.5% availability guarantee (Less than 50.4 minutes of downtime a week)
  + There are less than five server-side errors per day
* Portability
  + The web app supports popular browsers (Chrome, Edge, Firefox, Safari, Chrome for Android, Safari for iOS) for versions released in the past 2 years at the time of release
  + The web app supports the above browsers on the latest versions of these OSes: Windows, macOS, Debian, Android, iOS
* Correctness, testability, and maintainability
  + Unit tests will be used to test small units of code
  + Integration tests will be used to test APIs. Integration tests will use a local test database.
  + Test coverage is greater than 90% of backend code and greater than 80% of frontend code
  + Every minor update of the app goes through alpha testing and beta testing stages before general release
  + Logs of all backend and client errors are stored, with the time, error code, error message, call stack, and other relevant info
  + In case a recently released version generates too many errors, the latest version can be rolled back to a previous version by rerunning the deployment action for the previous version
* Security
  + Web content is served over TLS
  + The database is stored in a secure AWS data center
  + Only the backend instance and authorized staff members can directly access the database
  + User passwords are hashed and salted using a secure algorithm
  + Session tokens are generated using a secure and trusted library
  + All staff members accounts for GitHub and AWS must use multifactor authentication
  + Libraries with known vulnerabilities are updated or removed
  + Login requests are rate-limited. Each consecutive login request in a short period of time takes more time to respond.
  + A backup of the database is made daily

# 4. Breakdown of work/ Project timeline plan

***User Story Milestone Assigned to***

*P0-REQ-1: The skeleton of the web application with the following basic functions:*

*1.1 A street food vendor can create a vendor page*  *M1 Wenhuan*

*1.2 A customer user can leave reviews and star ratings*  *M1 Silvia*

*1.3 A user can sign-up for the app*  *MI Makoto*

*1.4 The nearby food vendors can be listed and shown on the map M1 Colin*

*P0-REQ-2: Food vendors and other customers can respond to the reviews*  *M1*  *Makoto*

*P0-REQ-3: A user can upload pictures/videos to photo/video libraries*  *M1*  *Wenhuan*

*P0-REQ-4: A food vendor user can set a picture as their profile picture M1 Colin*

*P0- REQ-5: A user can search for street food vendors* M*1*  *Wenhuan*

*P0- REQ-6: A user can filter the search result*  *M2 Silvia*

*P0-REQ-7: The app can track food vendors’ location* *M2* *Makoto*

*P0-REQ-8: A customer user can star their favorite food vendors*   *M2*  *Silvia*

*P0-REQ-9: The app can recommend food items to the users*  *M2* *Colin*

*P0-REQ-10: Users can set their accounts M2*  *Makoto*

*P0-REQ-11: Users can log in using their google account M2 Wenhuan*

*P0-REQ-12: The users' activities (stars, search queries) can be analyzed M3*  *Silvia*

*P0-REQ-13: A dashboard will be displayed on vendor’s account M3 Colin*

*P0-REQ-14: Business Start-Up instructions are available M3*  *Wenhuan*

*P0-REQ-15: A user can share vendor page to social media M3 Wenhuan*

*P0-REQ-16: Discount QR code generator and list M3 Makoto*