

## final Project for SW Engineering Class CSC 648-848 Spring 2023

### **Food Feast**

#### **Team 01**

Elahe Bashiri ([Ebashiri@sfsu.edu](mailto:Ebashiri@sfsu.edu)): Team lead

Abbas Mahdavi: Back end lead

Megan Lew: Front end lead

Nathan Rennacker: Github master

Jed Graves: Back end Developer

Alexander Diaz: Front end Developer

**URL:** <http://35.160.127.228/>

**Date:**05/24/2023

## Product summary

### **Product Name: Food Feast**

Food Feast is a dynamic web application designed to make finding meals and food delivery more efficient for the San Francisco State University (SFSU) community, comprising students, staff, and faculty. It targets the challenges of busy schedules and limited food options within the campus, by providing an accessible and user-friendly platform to search for nearby restaurants, order meals, and have them delivered right on campus or to the dorms. Food Feast stands as a strong supporter of local businesses, providing a unique channel to reach a specific customer base.

### **P1 Functions:**

- Registration: Unregistered users shall be able to create an account using SFSU information.
- Browse Food Items: Unregistered users shall have the ability to view available food items.
- Map Accessibility: Unregistered users shall be able to access a visual map.
- Filter Application: Unregistered users shall be able to apply filters to restaurants.
- Restaurant Form Access: Potential restaurant owners shall have access to a registration form.
- Search Bar Usage: Users shall be able to use the search bar to find a specific restaurant.
- Login: Registered SFSU users shall have the capability to log in to their accounts.
- Order Placement: Registered SFSU users shall have the ability to select and order items.
- Restaurant Registration: Restaurant owners shall be able to register their establishments.
- Menu Upload: Restaurant owners shall have the ability to upload their menus.
- Delivery Interface Utilization: Delivery personnel shall have an interface to manage orders.
- Approval or Rejection of Restaurant Requests: The admin shall authority over approvals.
- Managing Approved Restaurants: The admin shall be able to revoke privileges of restaurant.
- User Management: The admin shall have the authority to manage the list of users.

**Unique Proposition:**

Food Feast's uniqueness lies in its tailored solution for the SFSU community, merging technology, convenience, and local business support. Unlike general food delivery services, Food Feast offers a campus-oriented service, providing fast delivery within the campus and the dorms. Furthermore, our student verification system offers exclusive discounts to verified students, making nutritious meals more affordable. We also accept payments through Gator Passes, providing a seamless and speedy payment process for the SFSU community.

URL: <http://35.160.127.228/>

## Milestone Documents

# **SW Engineering CSC648-848 Spring 2023**

## **Milestone 1: Use cases, High-Level Requirements and Architecture**

### **Food Feast**

#### **Team 01**

Elahe Bashiri ([Ebashiri@sfsu.edu](mailto:Ebashiri@sfsu.edu)): Team lead

Abbas Mahdavi: Back end lead

Megan Lew: Front end lead

Nathan Rennacker: Github master

Jed Graves: Developer

Alexander Diaz: Developer

### **Revision History**

Date Submitted	Date Revised
3/13/2023	3/18/2023

## **1. Executive Summary:**

Our team is developing a web application for restaurant search, order, and pickup called Food Feast. We are looking to address the issue of hungry SFSU students, staff, and faculty while supporting nearby local businesses. Our application is developed for exclusive use for the SFSU community. Many students and faculty usually have busy schedules and a limited amount of time to leave campus to search for food. It is important to have daily nutritious meals, but it may be difficult to find meals or have them delivered in time before the next class.

Food Feast will help users simplify the process of finding meals and having them delivered on campus or to their dorms. Our application will feature search/reviews of local restaurants, the ability of restaurants to register/advertise menus, the ability of restaurants to manage orders/delivery, and order meals. The delivery service will have access to a campus map for faster delivery times to classrooms and campus facilities. This makes our service unique because food delivery services usually deliver on a nearby street. Food delivery services are usually costly, our app will have a student verification system and offer students a 20% discount. We will also aim to provide a secure, fast payment system. Students can purchase their meals by linking their Gator passes.

Our team consists of six members, and we are a student startup team at SFSU.

## 2. Personae and main Use Cases:

### Faculty

#### Dr. Richard Demir



"Being very task-driven, it can be hard to find time to eat during the day!"

Age: 52  
Job: Professor of Psychology  
Family: Single

##### Bio

Dr. Richard Demir is a cognitive psychology expert and professor at the university. He earned his Ph.D. in psychology from the University of Heidelberg in Germany and later moved to San Francisco to continue his research and teaching.

##### Goals

- Balance lecturing, research, and meetings
- Eat a quick lunch during dynamic breaks in his schedule
- Often stays late at the office and needs to eat dinner there as well

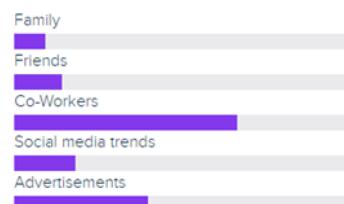
##### Frustrations

- Limited opportunities to eat with his busy schedule
- Long delivery times
- Inability to order directly to his office

##### Tech Usage



##### Influences



### Staff

#### Steven Morales



"I never compromise when it comes to good food."

Age: 34  
Job: Facilities Maintenance Manager  
Family: Married, two children

##### Bio

Steven is a 34-year-old facilities maintenance manager at the university. He was born and raised in Southern California, but has been living in San Francisco for the past 10 years. He is married with two young children and enjoys spending his free time with his family and playing soccer with his friends.

##### Goals

- Lead team to successfully complete maintenance tasks
- Needs to know when the food will arrive
- Wants his team members to be able to relax during their lunch breaks
- Wants quality food during his lunch break

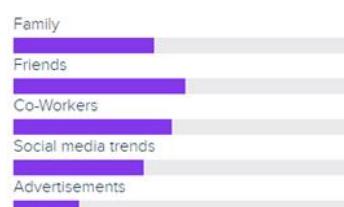
##### Frustrations

- Limited on campus delivery locations
- Long delivery times
- Confusion with campus delivery

##### Tech Usage



##### Influences



## Student

### Farah Ahmadi



*"With today's technology, there is no excuse for food delivery to be so difficult."*

Age: **28**  
Job: **Graduate Student**  
Family: **Single**

#### Bio

Farah is a 28-year-old Iranian-American pursuing her master's degree in computer science at San Francisco State University. Born and raised locally, she developed a love for tech and coding from a young age. Farah is a dedicated and hardworking student, with a passion for using technology to solve real-world problems. In her free time, she enjoys exploring the city and trying the wide variety of cultural food.

#### Goals

- Finish her master's degree
- Be as productive as possible during study time
- Have a hassle-free way to order food to her dorm room
- Be able to order late at night

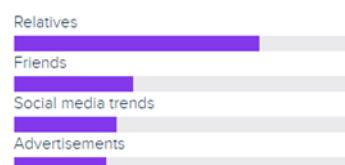
#### Frustrations

- Limited on-campus delivery locations
- Limited options for Middle Eastern food near her dorm
- High delivery fees

#### Tech Usage



#### Influences



## Driver

### Kylie Baker



*"I love being able to share with my friends. Food makes everyone happy!"*

Age: **20**  
Job: **Undergraduate Student**  
Family: Dating

#### Bio

Kylie is a 20-year-old undergraduate student at the university. She is a social butterfly and enjoys spending time with her friends and boyfriend. Kylie grew up in Arizona but moved to San Francisco in her teens and fell in love with the city's vibrant culture and diverse community. She is majoring in Communications and hopes to pursue a career in public relations or marketing. In her free time, Kylie enjoys exploring the city's restaurants, attending music festivals, and hiking in the nearby mountains.

#### Goals

- Have a great college experience
- Be able to order food for herself and her friends so they don't have to drive somewhere
- Feed everyone without breaking the bank

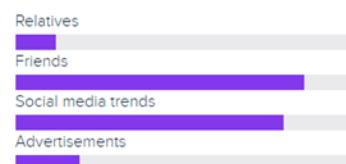
#### Frustrations

- Expensive food delivery
- Few group discounts
- Long delivery times

#### Tech Usage



#### Influences



## Admin

### Sarah Miller



*"Education doesn't run on an empty stomach!"*

Age: **30**  
Job: **System Admin**  
Family: **Married**

#### Bio

Sarah is a 30-year-old system admin from San Francisco. She graduated from SFSU and now works for a startup food delivery company. She loves technology and spends much of her time reading and researching new tech advancements. She can often be found hanging out in local coffee shops and eateries.

#### Goals

- Successfully maintain software for the campus food app
- Connect restaurants to the app
- Keep the app secure

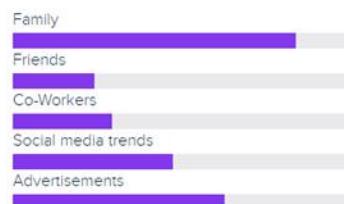
#### Frustrations

- Buggy software
- Managing confusing database schema
- Lack of communication from management

#### Tech Usage



#### Influences



## Restaurant Owner

### Haru Yamamoto



*"I'm bringing real Japanese cuisine to the bay area at affordable prices"*

Age: **57**  
Job: **Restaurant Owner**  
Family: **Married, one child**

#### Bio

Haru is a Japanese restaurant owner. He spent most of his life in Japan, but recently moved to the United States and opened a restaurant in San Francisco near the state university. The restaurant currently takes up most of his time, and he doesn't like to waste time on things he doesn't consider to be important or beneficial to the business.

#### Goals

- Increase the profitability of his restaurant
- Market his restaurant to the university students, staff, and faculty
- Add his restaurant to food apps without an excessive amount of effort

#### Frustrations

- Confusing software
- Long signup processes
- Food apps not delivering to a good portion of campus

#### Tech Usage



#### Influences



## **Use Cases:**

### **1. Ordering Food:**

Dr. Demir is a Professor of Psychology at San Francisco State University. He has an extremely busy schedule, combining meetings, lectures, and research sessions. After his morning class, he has a meeting with the department head and only a short break in between. He knows that he will not be at his peak performance for the meeting and afternoon class if he doesn't eat, the break is not long enough to go off campus for food, and being lunchtime, it is also a busy time for most delivery apps. He knows, however, that the campus food delivery app delivers only from nearby restaurants using drivers that are in the immediate area. He also knows he can have the food delivered directly to his office. He opens the app and is immediately greeted with the home page. He sees a lot of options and promotions but doesn't want anything really heavy, like pasta or pizza, because that might make him tired. He sees the option for sandwiches and immediately selects it. He is presented with several local, high-rated options and selects one that looks good. Being pressed for time, he decides to select the featured sandwich and a coffee. Tapping the checkout button, he is directed to the login page. Now that he is logged in, his payment information is saved in the app, so checkout is a breeze. It arrives at his door shortly after he returns to his office. This will keep him alert and his blood sugar up through the afternoon.

### **2. Searching and sorting:**

Farah is a graduate student at San Francisco State University studying computer science. This is one of the most demanding degrees, and she spends long hours studying and researching for her thesis. She enjoys her studies, but they are time-consuming, and she often finds that she has forgotten to eat dinner until she becomes extremely hungry around 9 p.m. On this particular night, as with many, she is at a critical place in her research and doesn't want to break her concentration for too long by leaving her dorm room or dealing with a long, annoying delivery app process. She opens the campus food delivery app and searches for Middle Eastern food. There are several options, so she selects the option to sort by highest rated restaurant. Since she has never used the service before, she is prompted to create an account. Filling out the form, she finds she can use her Gator card to pay. Convenient. A short time later, her food has arrived at her door, and she doesn't have to even change out of her pajama pants and slippers. She opens the bag. The heavenly smell reminds her of her mother's cooking.

### 3. Delivering:

Kylie is an undergraduate communications major. She is very motivated in her studies but also likes to have fun on the side. Like many college students, she often finds that her bank account gets uncomfortably low toward the end of the semester. She wanted to pick up a part-time job, but it's hard to find one that will fit her busy schedule. A few weeks ago, a friend recommended that she try delivering food for a delivery service. She doesn't have a car, so she wouldn't be able to make deliveries far from campus. She does have an electric scooter, though, and it is often faster to get around on than a car near campus. She then found out about the campus delivery app, which only does deliveries from restaurants near campus. Perfect! Since then, she has made nearly 100 deliveries to students, faculty, and staff on the SFSU campus. Using her scooter, she is able to deliver faster than any other food delivery service. She is also able to use the built-in maps feature to navigate safely and avoid high-traffic areas that might be dangerous for a scooter. Kylie now has a steady income that allows her to have more fun on the weekends and enjoy her college experience.

### 4. Group Order:

Steven is a facilities maintenance manager at SFSU. He is in charge of a team of 4 that specializes in utility maintenance. On this particular Friday, there was an unexpected pipe burst in the main sprinkler line on the campus lawn. It had been an unusually hot week for the bay area, and the sprinkler would have to be fixed before the weekend. Steven and his team worked in the heat for hours and finally got it fixed around two in the afternoon. They had worked through lunch, and Steven wanted to thank them for their good work. At this point, they're cooling off in the maintenance office, and he doesn't want anyone to have to go get food, so he opens the campus food delivery app. Ordering through this service will allow the food to be delivered directly to the maintenance office. It will also be fast because the delivery drivers never go far from campus. Steven is not very tech-savvy, and he doesn't care to be, so he doesn't use apps that aren't user-friendly or take more than a few taps to get what he wants. He sees the option for pizza. Perfect for a Friday afternoon. He knows that one of his team members, Cathy, has recently become vegan, so he chooses a restaurant that has vegan options. He orders the pizza, and it arrives quickly. Team morale has never been higher.

## 5. Registering Restaurant:

Haru is the owner of a Japanese restaurant near the SFSU campus. He has only lived in the United States for 3 years, and his English is not yet fluent. His employees all speak Japanese as well, so that is not a problem for him. He recognizes the potential of marketing to the nearby campus but is not really sure how to do it. He tried setting up a delivery app service before, but it was confusing and frustrating, so he gave up quickly. Then he found the campus delivery app. It is exactly what he needs to reach hungry university students. He goes into his office and logs into his desktop computer. The setup is very simple and straightforward. He fills out the form and uploads some photos of the restaurant and the menu items. He clicks submit and is prompted to make an account. Once it's submitted, he is informed that it may take up to one business day for his application to be approved. The next day, his restaurant was listed on the app, and he got his first order. It was quickly picked up by a student delivery person. Haru is now very excited that he was able to expand his customer base.

## 6. Approving Restaurants

Sarah fills the admin role for the backend of the food delivery app. She loves tech and is very proud of being good at her job, but one of her biggest pet peeves is poorly written-software. Fortunately for her, the campus food delivery app is a well-written piece of software, making her job a breeze. Today is a day just like any other. Working from home, she signs into her desktop computer and logs into the app as the admin user. There are three new applications from restaurants wanting to have their business listed on the app. She reviews the content to make sure it is in line with the content guidelines and that they have all of the necessary information, such as an address, photos, menu, and discounts. All three of them are complete today, so she accepts them. They are listed on the app, and the restaurants are notified that they are accepted.

### **3. List of main data items and entities – data glossary/description**

#### **1. Users:**

- Unregistered users: Anyone visiting the website that has not created an account, whom have limited access to the website, but can still browse restaurants and items, and create an account.
- Registered SFSU users: SFSU students, staff, and faculty who have full access to the website (Aside from admin, rest-reg, and driver section). Can log in and add items to the cart and place an order. Req Record: username, email, password
- Registered Restaurant owners: Users who have created a restaurant account while enrolling in their restaurant. Can manage their restaurant: delete, add/remove items, etc. Req Record: username, email, password, restaurant\_id
- Registered Drivers (Not a priority): Users that created a driver's account, can pick up active orders and deliver them, can manage order status, and have access to restaurant addresses and delivery locations, via a map to guide their delivery. Req Record: username, email, password

**2. Account:** The registered account of a user, including their data such as a username, email ending with @sfsu.edu (not required for restaurant owners and drivers), and password

**3. Restaurants:** The registered business by restaurant owners:

- Will include a unique ID, name, price range (1-5: \$), cuisine, description, rating (1-5), estimated delivery time, address, and main restaurant picture.

**4. Menu Items:** The list of the food for each specific restaurant, including the food item's unique id, name, description, price, corresponding restaurant id, and image.

**5. An order:** The record of a placed order that includes the user's info, the items in the order and their info, and the delivery location.

**6. Delivery:** The status of the order, and the driver's info

- 7. Location:** The location of the restaurant, the building of the delivery, and the room number of the user
- 8. Map:** A map available to the driver to help them deliver the food.
- 9. Payment:** A selection when ordering to use Student ID or in-person cash!
- 10. User App features:** Ability to see order status and ‘Favorite’ Restaurants/Food Items in particular restaurants

#### **4. Initial list of functional requirements**

##### **Unregistered Users:**

1. Map - Unregistered users shall be able to see a map of the area where the restaurants are in relation to you.
2. Search Bar - Unregistered users shall be able to type in the name of a restaurant and its plans to it on the map. This search bar could autofill with suggestions.
3. Browse Food Items - Unregistered users shall be able to see what restaurants have which items available for ordering.
4. Filter by different categories - Unregistered users shall be able to click on different filters (price, cuisine, distance) above the list of restaurants, the ordering of the list will be changed depending on delivery time.
5. View the popularity of restaurants - Unregistered users shall be able to view which restaurants are popular at which times depending on how many people order from a specific restaurant at different times of the day (could also tap into external analytics, but not sure of the logistics of it)
6. Estimated time for delivery vs. pickup - Unregistered users shall be able to see the estimate of the delivery. Each restaurant has an estimated time to complete an order, and then the system will use that for pickup time or add on additional delivery time for delivery.

7. Filing the restaurant form- Unregistered users shall be able to see the needed information and the policy for filing the form as a restaurant owner, it will be part of the registration choice.
8. Register - Unregistered users shall be able to register for an account (inputting sfsu.edu information if applicable)

**Registered SFSU Users:** inherit all functions from the unregistered users plus the below functions(users)

9. Login -users shall be able to log in
10. Place Order for pickup - users shall be able to select items from a menu from a specific restaurant (who has provided their menu already) and place an order with those items. This order will be conveyed to the restaurant, which will have it ready for you to pick up.
11. Place order for delivery - users shall be able to select items from a menu and place an order with those items. This order will be conveyed to the restaurant with a driver specified so that it can be delivered to you
12. View Order Status - users shall be able to check a page to see the status of their order, any delivery updates, and the estimated time it will arrive.
13. Update Order Status - users shall be able to cancel or add things to their order (this times out depending on how far along the restaurant is with completing their order
14. Write a review - users shall be able to get feedback. After you have ordered from somewhere, you will have the ability to give the restaurant a rating and include a written review if you choose
15. Save restaurants - users shall be able to save their favorite restaurants. A little star will appear next to restaurants that you can click on, and it will add it to your list of saved restaurants (a list that you can easily access when you want to see what you like)

**Restaurant Manager:**

16. Register for the service - Restaurants shall be able to opt-in to the service so that they will be displayed on the website as orderable. Will need specific information from them to ensure that they're legitimate.
17. Advertising - Restaurants shall be able to advertise on the website would include things like promoted restaurants that prominently display their food in the specific well has seen places on the website. You can click on these advertisements, and it will take you to the typical page where you can order from them.
18. Update menu - Restaurants shall be able to upload and update their menu in an easy-to-use form (that can include pictures), which will then allow users to select from it.
19. Instructions to delivery drivers - Restaurants shall be able to give instructions to drivers, including a list of items that can include a lot of information or very little. This typically will include a specific list of the items that the customer ordered so that the driver can ensure that everything is present.

**Driver:**

20. Delivery interface for drivers - Drivers shall be able to get their own orders which contain a map marked with the order number, etc. Drivers have a specific interface.
21. Report information to customer - Drivers shall be able to update the customer with any extra information that the customer may need, including. I'm here with your food, Increased wait time, etc.

**Admin:**

22. Approve pending requests from restaurants - Admin shall be required to approve pending requests from restaurants that are available to get orders from the website
23. Reject pending requests from restaurants - Admin shall be required to reject the requests which are inappropriate.
24. Control the list of approved restaurants - Admin shall be required to delete restaurants If sees an inappropriate username or receives a complaint from someone about anything

## **5. List of non-functional requirements**

- Refer to system properties and constraints such as:
  - Reliability
  - Response time
  - HW and networking requirements
  - Usability requirements
  - Marketing, legal, licensing
  - Media content (formats, size...)
  - Privacy: what is the data collected, how is the data used
  - Compatibility (e.g. which browsers...)
  - Can refer to:
    - Product (product behavior like speed, reliability)
    - Organization (e.g. process, standards used)
    - External factors (e.g. branding, legal disclaimers displayed)

## 6. Competitive analysis:

Feature	DoorDash	UberEats	GrubHub	Food Feast
Restaurant Search	++	+	+	+
Ease of Ordering	+	+	+	++
Delivery to specific places on campus	-	-	-	+
See what's popular to SFSU users	-	-	-	+
Ability to tip restaurant (supports restaurant workers)	-	-	-	+

Our app, designed for university students and staff, offers several advantages over its competitors: **Ease of ordering:** While competitors may have complex interfaces, our app streamlines the process for quicker and more efficient ordering, catering to the busy schedules of university users. **Delivery to specific locations on campus:** Unlike other food delivery apps, this app focuses on accurate delivery to designated spots within the university premises, ensuring timely and convenient food delivery for users in campus buildings or facilities. **Popular choices among university users:** Competitors may offer general popularity rankings, but our app displays the most popular menu items specifically among the university community, enhancing decision-making based on like-minded individuals' preferences. **Ability to tip restaurants:** In contrast to some competitors that only allow tipping drivers, our app provides users with the option to tip their favorite restaurants, fostering stronger relationships between customers and local eateries. By targeting these specific points and addressing the unique needs of the university

community, the proposed food delivery app aims to create a competitive edge over existing services in the market.

## **7. High-level system architecture and technologies used:**

A client-server model is used for the high-level system architecture. The client is built using React as it communicates with the server-side via HTTP requests.

The server side is built using Node.js and Express as it makes it easier to handle HTTP requests and middleware due to its functionality. The server is hosted on Amazon Web Services (AWS), and the operating system used is Amazon Linux 2, which offers a stable and secure Linux distribution optimized for AWS. The database used is MySQL v8.0, with MySQL known for storing and managing large amounts of data in databases. We will also be using Visual Studio Code, which is an IDE, to build and deploy web applications.

## **8. Team and roles:**

### **Team Lead: Elahe Bashiri**

Responsible for providing guidance and support. Assigning tasks to team members based on their strengths and abilities and ensuring that each team member clearly understands their responsibilities. Arranging weekly meeting groups to make sure about meeting deadlines.

### **GitHub Master: Nathan Rennacker**

Responsible for managing the repositories, ensuring that they are properly set up, maintained, and secured. Responsible for managing pull requests, ensuring that they are properly reviewed and merged and that any conflicts are resolved.

### **Back-end Lead: Abbas Mahdavi**

Responsible for the server-side web application logic, definition, and maintenance of the central database as well as integrating the work front-

end developers do. Responsible for advocating best practices and providing technical guidance to their team.

**Front-end Lead:** Megan Lew

Responsible for implementing visual elements that the user sees and interacts with within a web application as well as determining the structure of a web application. Responsible for advocating best practices and providing technical guidance to their team.

**Front-end Developer:** Alexander Diaz

Responsible for implementing visual elements that the user sees and interacts with in a web application, developing user interfaces, and determining the structure and build of web pages.

**Back-end Developer:** Jed Graves

Responsible for the server-side web application logic, definition, and maintenance of the central database as well as integrating the work front end developers do.

**9. Checklist:**

- So far all team members are engaged and attending team sessions when required. **(DONE)**
- Team found a time slot to meet outside of the class. **(DONE)**
- Back end, Front end leads, and Github master chosen. **(DONE)**
- Team ready and able to use the chosen back and front-end frameworks, and those who need to learn are working on learning and practicing. **(DONE)**
- Team lead ensured that all team members read the final M1 and agree/understand it before submission. **(DONE)**

- Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.). **(DONE)**
- NEW: Use of any GenAI tool like ChatGPT: say if you used ChatGPT or like and how and for what segment of Milestone 1 (brief paragraph). As per class policy: this is allowed as a help, BUT you cannot copy and paste its output and claim it is your own text, you need to put it in quotes or modify it, and only for short sentences You also are responsible for accuracy of your submission, so any ChatGPT content needs to be checked by you. **We don't use ChatGPT**

## **SW Engineering CSC648/848 Spring 2023**

### **Milestone 2 part I: More Detailed Requirements, Specs, Architecture, UI mock-ups**

#### **Food Feast**

#### **Team 01**

Elahe Bashiri ([Ebashiri@sfsu.edu](mailto:Ebashiri@sfsu.edu)): Team lead

Abbas Mahdavi: Back end lead

Megan Lew: Front end lead

Nathan Rennacker: Github master

Jed Graves: Developer

Alexander Diaz: Developer

#### **Revision History**

Date Submitted	Date Revised
3/31/2023	04/07/2023

## **1. Executive Summary:**

Our team is developing a web application for restaurant search, order, and pickup called Food Feast. We are looking to address the issue of hungry SFSU students, staff, and faculty while supporting nearby local businesses. Our application is developed for exclusive use for the SFSU community. Many students and faculty usually have busy schedules and a limited amount of time to leave campus to search for food. It is important to have daily nutritious meals, but it may be difficult to find meals or have them delivered in time before the next class.

Food Feast will help users simplify the process of finding meals and having them delivered on campus or to their dorms. Our application will feature search/reviews of local restaurants, the ability of restaurants to register/advertise menus, the ability of restaurants to manage orders/delivery, and order meals. The delivery service will have access to a campus map for faster delivery times to classrooms and campus facilities. This makes our service unique because food delivery services usually deliver on a nearby street. Food delivery services are usually costly, our app will have a student verification system and offer students a 20% discount. We will also aim to provide a secure, fast payment system. Students can purchase their meals by linking their gator passes.

Our team consists of six members, and we are a student startup team at SFSU.

## **2. List of main data items and entities (expand as necessary):**

### **1. Users:**

- Unregistered users: Anyone visiting the website that has not created an account, whom have limited access to the website, but can still browse restaurants and items, and create an account.
- Registered SFSU users: SFSU students, staff, and faculty who have full access to the website (Aside from admin, rest-reg, and driver section). Can log in and add items to the cart and place an order. Req Record: username, email, password
- Registered Restaurant owners: Users who have created a restaurant account while enrolling their restaurant. Can manage their restaurant: delete, add/remove items, etc. Req Record: username, email, password, restaurant\_id
- Registered Drivers (Not a priority): Users that created a driver's account, can pick up active orders and deliver them, can manage order status, and have access to restaurant addresses and delivery locations, via a map to guide their delivery. Req Record: username, email, password

**2. Account:** The registered account of a user, including their data such as a username, email ending with @sfsu.edu (not required for restaurant owners and drivers), and password

**3. Restaurants:** The registered business by restaurant owners:

- Will include a unique ID, name, price range (1-5: \$), cuisine, description, rating (1-5), estimated delivery time, address, and main restaurant picture.

**4. Menu Items:** The list of the food for each specific restaurant, including the food item's unique id, name, description, price, corresponding restaurant id, and image.

**5. An order:** The record of a placed order that includes the user's info, the items in the order and their info, and the delivery location.

- 6. Delivery:** The status of the order, and the driver's info
- 7. Location:** The location of the restaurant, the building of the delivery, and the room number of the user
- 8. Map:** A map available to the driver to help them deliver the food.
- 9. Payment:** A selection when ordering to use Student ID or in-person cash!
- 10. User App features:** Ability to see order status and 'Favorite' Restaurants/Food Items in particular restaurants

### **3. Functional Requirements - prioritized:**

#### **Priority 1**

- **Unregister Users**
  1. Register - Unregistered users shall be able to register for an account (inputting sfsu.edu information if applicable)
  2. Browse Food Items - Unregistered users shall be able to see what restaurants have which items available for ordering.
  3. Map - Unregistered users shall be able to see a map of the area with where the restaurants are in relation to you.
  4. Filter by different categories - Unregistered users shall be able to click on different filters (price, cuisine, distance) above the list of restaurants, the ordering of the list will be changed depending on delivery time
  5. Filing the restaurant form- Unregistered users shall be able to see the needed information and the policy for filing the form as a restaurant owner, it will be part of the registration choice.
  6. Search Bar - Unregistered users shall be able to type in the name of a restaurant and its to it on the map. This search bar could autofill with suggestions.

- **Registered SFSU users**( Reg.SFSU users inherit all functions of unreg users)

7. Login - Reg.SFSU users shall be able to log in.

8. Place order - Reg.SFSU users shall be able to select items from a menu and place an order with those items. This order will be conveyed to the restaurant with a driver specified so that it can be delivered to you.

- **Restaurant Managers**

9. Register for the service - Unregistered Restaurant owners shall be able to register and fill out the restaurant information form to get the service. Will need specific information from them to ensure that they're legitimate.

10. Upload menu - Registered Restaurant owners shall be able to upload their menu in an easy-to-use form (that can include pictures), which will then allow users to select from it.

- **Drivers**

11. Delivery interface for drivers - Drivers shall be able to get their own orders which contain a map marked with the order number, etc.

- **Admin**

12. Approve pending requests from restaurants - Admin shall be required to approve pending requests from restaurants that are available to get orders from the website.

13. Reject pending requests from restaurants - Admin shall be able to reject the requests which are inappropriate.

14. Control the list of approved restaurants - Admin shall be able to remove restaurants If sees an inappropriate username or receives a complaint from someone about anything.

15. Control the list of users - Admin shall be able to remove users.

## **Priority 2**

- **Unregister Users**

16. View the popularity of restaurants - Unregistered users shall be able to view which restaurants are popular at which times depending on how many people order from a specific restaurant at different times of the day (could also tap into external analytics, but not sure of the logistics of it)

17. Estimated time for delivery vs. pickup - Unregistered users shall be able to see the estimate of the delivery. Each restaurant has an estimated time to complete an order, and then the system will use that for pickup time or add on additional delivery time for delivery.

- **Registered SFSU users**( Reg.SFSU users inherit all functions of unreg users)

18. View Order Status - Reg.SFSU users shall be able to check a page to see the status of their order, any delivery updates, and the estimated time it will arrive.

19. Update Order Status - Reg.SFSU users shall be able to cancel or add things to their order (this times out depending on how far along the restaurant is with completing their order).

20. Save restaurants - Reg.SFSU users shall be able to save their favorite restaurants. A little star will appear next to restaurants that you can click on, and it will add it to your list of saved restaurants (a list that you can easily access when you want to see what you like)

- **Restaurant Managers**

21. Instructions to delivery drivers - Registered Restaurant owners shall be able to give instructions to drivers, including a list of items that can include a lot of information or very little. This typically will include a specific list of the items that the customer ordered so that the driver can ensure that everything is present.

### **Priority 3**

- **Registered SFSU Users** ( Reg. SFSU users inherit all functions of unreg users)

22. Write a review - Reg.SFSU users shall be able to get feedback. After you have ordered from somewhere, you will have the ability to give the restaurant a rating and include a written review if you choose.

- **Restaurant Manager**

23. Advertising - Registered Restaurant owners shall be able to advertise on the website would include things like promoted restaurants that prominently display their food in the specific well has seen places on the website. You can click on these advertisements, and it will take you to the typical page where you can order from them.

- **Drivers**

24. Report information to customer - Drivers shall be able to update the customer with any extra information that the customer may need, including. I'm here with your food, Increased wait time, etc.

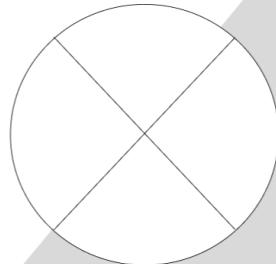
### **4. UI Storyboards for each main use case (low-fidelity B&W wire diagrams only):**

- **Ordering Food**

Dr. Demir is a Professor of Psychology at San Francisco State University. He has an extremely busy schedule, combining meetings, lectures, and research sessions. After his morning class, he has a meeting with the department head and only a short break in between. He knows that he will not be at his peak performance for the meeting and afternoon class if he doesn't eat, the break is not long enough to go off campus for food, and being lunchtime, it is also a busy time for most delivery apps. He knows, however, that the campus food delivery app delivers only from nearby restaurants using drivers that are in the immediate area. He also knows he can have the food delivered directly to his office. He opens the app and is immediately greeted with the home page. He sees a lot of options and promotions but doesn't want anything really heavy, like pasta or pizza, because that might make him tired. He sees the option for sandwiches and immediately selects it. He is presented with several local, high-rated options of sandwich specific restaurants and selects one that looks good. Being pressed for time, he decides to select the featured sandwich and a coffee. He already has an account so tapping the cart icon takes him to his order screen where he can edit any items in his cart and place his order. It arrives at his door shortly after he returns to his office. This will keep him alert and his blood sugar up through the afternoon.

## Food Delivery. Anywhere on Campus.

 Search for restaurant or cuisine... 

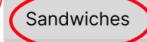


### Featured Restaurants

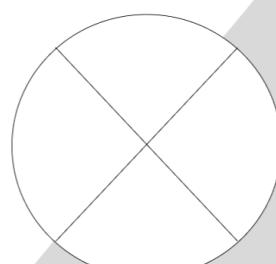


Clicking on any cuisine from the dropdown or searching for one will take you to a list of restaurants with ratings displayed.

## Food Delivery. Anywhere on Campus.

Italian  
Sandwiches   
Burgers  
Ramen  
Mexican  
Indian  
Etc.

Search F 



Fea



In this case clicking on the sandwich tile will take you to a list of sandwich specific restaurants (deli's etc)

## Category



Clicks here

Category

Restaurant A ★★★★★ XX - XX min away	Restaurant B ★★★★ XX - XX min away	Restaurant C ★★★★★ XX - XX min away
Restaurant D ★★★★★ XX - XX min away	Restaurant E ★★★★★ XX - XX min away	Restaurant F ★★★★★ XX - XX min away

Clicking on any of the tiles on the category page will take you to the restaurant page where you can order any items.

[Return to //Category//](#)

**Restaurant Name**

Most Popular Items

 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>	 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>
 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>	 <span style="background-color: #f0f0f0; border: 2px solid blue; padding: 5px;">Food Item Name Price</span>

Restaurant Defined Food Sub-genres

 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>	 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>
--	--

[Return to //Category//](#)

**Restaurant Name**

Ready to order?  
[Click here!](#)

Most Popular Items

Clicks/taps Here Food Item Name  
Price

 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>	 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>
 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>	 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>

Restaurant Defined Food Sub-genres

 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>	 <span style="background-color: #f0f0f0; padding: 5px;">Food Item Name Price</span>
--	--

SFSU Food Feast   ≡ Menu   Contact   About   For Restaurants   For Drivers   🔍 📍   Username

[⬅ Return to //Category//](#)

**Restaurant Name**

Clicks/taps Here

Ready to order?  
Click here! 🛒

Most Popular Items

<span style="font-size: 1.5em;">+</span>	Food Item Name Price	<span style="font-size: 1.5em;">+</span>	Food Item Name Price
<span style="font-size: 1.5em;">+</span>	Food Item Name Price	<span style="font-size: 1.5em;">+</span>	Food Item Name Price

Restaurant Defined Food Sub-genres

<span style="font-size: 1.5em;">+</span>	Food Item Name Price	<span style="font-size: 1.5em;">+</span>	Food Item Name Price
--	-------------------------	--	-------------------------

SFSU Food Feast   ≡ Menu   Contact   About   For Restaurants   For Drivers   🔍 📍   Login   Register

[⬅ Order more items](#)

**Your Cart**

Place Your Order

<span style="font-size: 1.5em;">-</span>  Qty	Food Item Name Price	<span style="font-size: 1.5em;">-</span>  Qty	Food Item Name Price
<span style="font-size: 1.5em;">-</span>  Qty	Food Item Name Price	<span style="font-size: 1.5em;">-</span>  Qty	Food Item Name Price
<span style="font-size: 1.5em;">-</span>  Qty	Food Item Name Price	<span style="font-size: 1.5em;">-</span>  Qty	Food Item Name Price

The screenshot shows the 'Your Cart' screen of the SFSU Food Feast app. At the top left is a back arrow and the text 'Order more items'. At the top right are search, location, login, and register buttons. The main area is titled 'Your Cart' and contains two rows of food item cards. Each card has a minus icon, 'Qty' below it, 'Food Item Name' and 'Price' next to each other, and another minus icon. At the bottom center is a button labeled 'Place Your Order'.

SFSU  
Food Feast

Menu Contact About For Restaurants For Drivers

Order more items

Your Cart

Food Item Name  
Price

Food Item Name  
Price

Food Item Name  
Price

Food Item Name  
Price

Place Your Order

Depending on how many items are in your cart, the place your order button either shows up at the bottom or the top right.

This screenshot is identical to the one above, showing the 'Your Cart' screen. However, a red oval has been drawn around the 'Place Your Order' button at the bottom center, and the text 'Clicks/taps here' is written in red to its left, indicating where the user should tap to place their order.

SFSU  
Food Feast

Menu Contact About For Restaurants For Drivers

Order more items

Your Cart

Food Item Name  
Price

Clicks/taps here

Place Your Order

Thank you for ordering with us!

Enjoy your food when it arrives!

Order delivery estimate:  
XX:XX - XX:XX

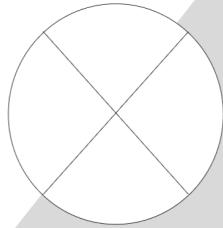
Clicking on the cart icon will take you to the current items you have selected to order, here you can remove items if you wish and also place an order.

### - **Login/ Registration**

A new user visits the Food Feast home page and wishes to access its features. To do so, they click on the "Register" button, which redirects them to the sign-up page. Here, the user enters their personal information, including their username, email address, and password, and confirms the password. Once the user has successfully created their account and the system validates the input, the user is automatically redirected to the login page. On the login page, the user enters their username and password and clicks on the "Sign In" button. Then the system redirects the user to the home page, where they can access the various features of the Food Feast application.

## Food Delivery. Anywhere on Campus.

 Search for restaurant or cuisine... 



### Featured Restaurants



Registration page

Navigation Bar



### Sign Up to have an account

Enter Username:

Enter Email:

Enter Password:

Re-Enter Password:

I accept the [terms & conditions](#)

Already have an account ? [Sign In here](#)

Login page

Navigation Bar



**Sign in to see more**

Username:

Email:

Password:

Remember Me      [Forgot Password ?](#)

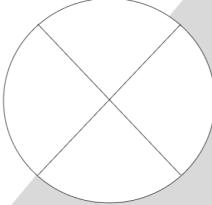
Don't have an account ? [Sign Up here](#)

SFSU Food Feast    For Restaurants    For Drivers    About    Contact    [!\[\]\(ec8bab73e52a81c95e69ed9d332ca55a\_img.jpg\) Login](#)    [Register](#)

## Food Delivery. Anywhere on Campus.

▼

🔍



Featured Restaurants



Restaurant A

★★★★★



Restaurant B

★★★★★



Restaurant C

★★★★★

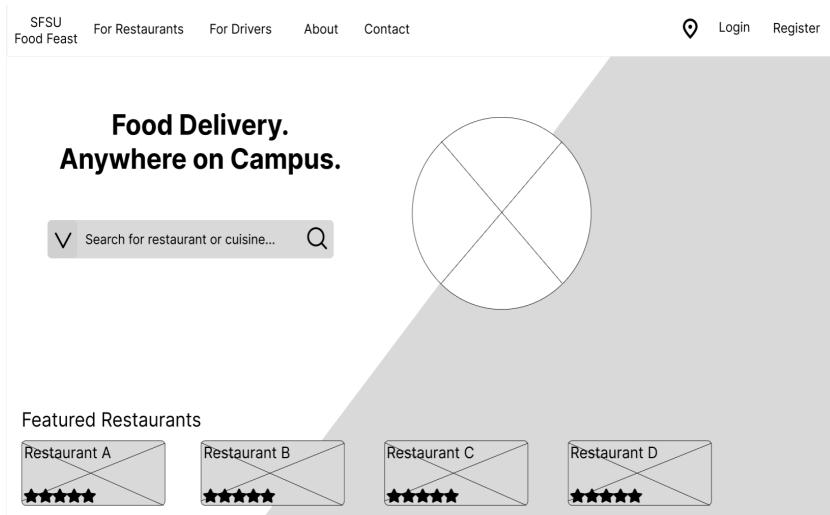


Restaurant D

★★★★★

- **Driver**

Kylie is a 20-year-old university undergraduate student who is majoring in communications and hoping to pursue a career in public relations or marketing. She enjoys hosting parties for her friends, but as a college student, money can be tight. She then found the campus delivery app and decided to deliver food to students from nearby restaurants. The setup is very simple. She opens the Food Feast home page and clicks on the For Drivers button to register for the app. Once registered, she can now access the Drivers page. Here she can see a list of all available orders that she can accept and deliver. The page also has options to view a local map and campus map, and each order tile displays the time to deliver the food, the restaurant contact information, and an accept button. Once the order is accepted, she can see the order details, and an option to mark when the order has been picked up. After choosing that option, she is then allowed to mark the order as delivered.



SFSU Food Feast Menu Home Register Restaurant Driver About Search for Restaurant or Cuisine ...

**Decline**

**Deliver by 9:23pm** 20

Chipotle  
3 items 0.2 miles

\$32.67

**Accept**

SFSU Food Feast Menu Home Register Restaurant Driver About Search for Restaurant or Cuisine ...

**Decline**

**Deliver by 8:53pm** 20

Marugame Udon  
6 items 0.3 miles

\$21.56

**Accept**

SFSU Food Feast Menu Home Register Restaurant Driver About Search for Restaurant or Cuisine ...

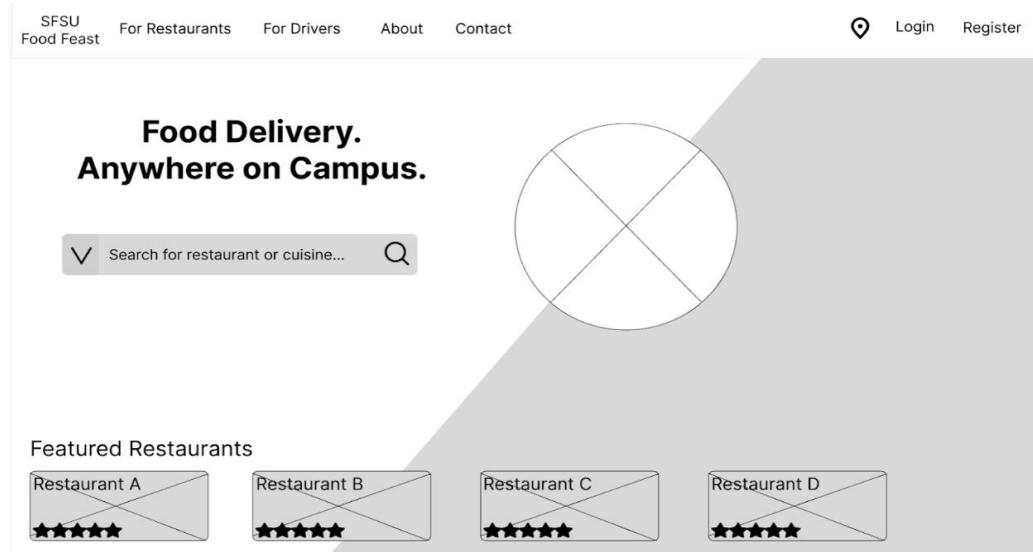
**Deliver by 9:23pm** 20

Marugame Udon  
6 items 0.3 miles

\$21.56

- **Restaurant Manager**

Haru is the owner of a Japanese restaurant near the SFSU campus. He has only lived in the United States for 3 years, and his English is not yet fluent. His employees all speak Japanese as well, so that is not a problem for him. He recognizes the potential of marketing to the nearby campus but is not really sure how to do it. He tried setting up a delivery app service before, but it was confusing and frustrating, so he gave up quickly. Then he found the campus delivery app. It is exactly what he needs to reach hungry university students. He goes into his office and logs into his desktop computer. The setup is very simple and straightforward. He fills out the form and uploads some photos of the restaurant and the menu items. He clicks submit and is prompted to make an account. Once it's submitted, he is informed that it may take up to one business day for his application to be approved. The next day, his restaurant was listed on the app, and he got his first order. It was quickly picked up by a student delivery person. Haru is now very excited that he was able to expand his customer base.



SFSU Food Feast      For Restaurants V      For Drivers      About      Contact       Login      Register

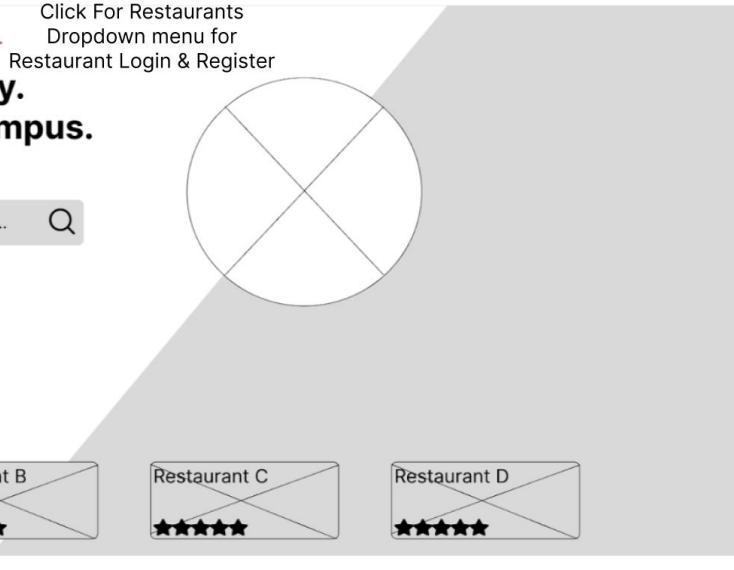
Click For Restaurants  
Dropdown menu for  
Restaurant Login & Register

**Food Delivery.**  
**Anywhere on Campus.**

V Search for restaurant or cuisine... Q

Featured Restaurants

Restaurant A      Restaurant B      Restaurant C      Restaurant D



Navigation Bar

### Register Restaurant

Name \*:

Email / Username \* :

Password \*:

Restaurant Name \*:

Restaurant Phone Number \*:

Restaurant Address \*:

Category \*

Add Images \*

Description \*

I accept the [terms & conditions](#)

Already Registered? [Login here](#)

Navigation Bar

After clicking  
Register

## Register Restaurant



Please allow 24hrs for  
admin to review and  
register your restaurant

Navigation Bar

## Restaurant Manager Login

Username:

Password:

Remember Me      [Forgot Password ?](#)

Want to Register your Restaurant ? [Register here](#)

Navigation Bar

## Main Restaurant Manager Dashboard

Logout

Main Landing Page

Main Dashboard

Welcome, restaurant manager name!

Message for driver

View Total Sales

Manage Orders

Order #12XXXX-XX

Quantity	Item	Price	Customer	Delivery Location
1	Sandwich	\$4.99	Student A	SFSU

Message to Driver :

Enter message here....

Send Message

Navigation Bar

### Menu Entry

Name: \*

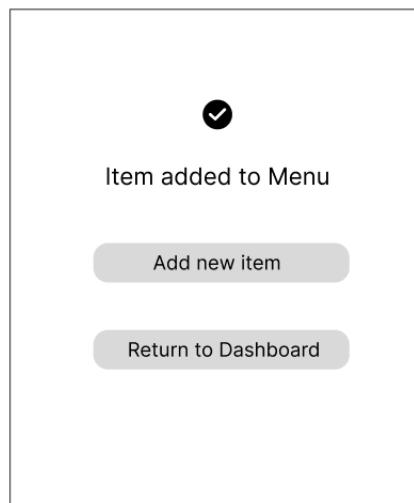
Price: \*

Text:   
Enter short description of menu item

Add Images \*

Submit

Navigation Bar



## 5. High-level Architecture, Database Organization summary only:

### DB organization:

- Table: users
  - Columns:
    - id int(11) AI PK
    - username varchar(225)
    - email varchar(255)
    - password varchar(255)
    - isAdmin tinyint(1)
    - isRestaurantOwner tinyint(1)
    - isDriver tinyint(1)
- Table: restaurants
  - Columns:
    - id int(11) AI PK
    - name varchar(25)
    - price varchar(5)
    - cuisine int(11)
    - description text
    - rating int(11)
    - est\_delivery\_time int(11)
    - address varchar(100)
    - picture varchar(255)
    - is\_featured tinyint(1)
    - phone varchar(20)
    - hours varchar(50)
    - owner\_id int(11)
- Table: food\_orders
  - Columns:
    - order\_id int(11) AI PK
    - customer\_id int(11)
    - restaurant\_id int(11)
    - order\_date datetime
    - order\_status enum('Pending','In Progress','Ready for Pickup','Out for Delivery','Delivered','Cancelled')
    - order\_total decimal(10,2)
    - delivery\_address varchar(255)
    - payment\_method varchar(255)
    - special\_instructions text
    - order\_accepted\_by\_driver tinyint(4)

- Table: menu\_items
  - Columns:
    - id int(11) PK
    - name varchar(45)
    - price decimal(10,2)
    - restaurant\_id int(11)
    - image varchar(255)
    - description text
- Table: order\_items
  - Columns:
    - order\_item\_id int(11) AI PK
    - order\_id int(11)
    - menu\_item\_id int(11)
    - quantity int(11)
    - price decimal(10,2)
    - item\_total decimal(10,2)
    - special\_requests text
    - item\_name varchar(255)
- Table: restaurant\_owners
  - Columns:
    - id int(11) AI PK
    - email varchar(255)
    - password varchar(255)
    - restaurant\_id int(11)
- Table: userFavorites
  - Columns:
    - user\_id int(11) PK
    - restaurant\_id int(11) PK
- Table: restaurant\_owners
  - Columns:
    - id int(11) AI PK
    - email varchar(255)
    - password varchar(255)
    - restaurant\_id int(11)

### Media Storage:

Images are stored using image links.

### Search/Filter architecture/implementation:

Search is implemented with React Fuse.Js dynamic search. It has fuzzy search capabilities and can handle misspelled words. It can specify three keys in our data that we want to search on which are name, cuisine, and description. Priority 2 is viewing the popularity of restaurants based on time. There can be a sorting algorithm to sort the restaurants with the busiest restaurant with the highest ranking.

## 6 . Identify actual key risks for your project at this time:

### **Technical Software Ability**

**Broad** - In a student-run team where we are required to make a full-stack web program, there might be issues in development where we aren't sure where to proceed in a specific part of our program due to a lack of expertise. This could be in the front end or back end. This is probably more complicated to resolve up front as problems like these are encountered during development and can't necessarily be predicted, but resolving to ask for help when needed and *early* is important in combating the issue.

**Insufficient Testing** - If we don't do our due diligence with testing, we may find ourselves down the line with problems from merges many days prior that will hinder our future development. To counter this, we will ensure that other members test each other's code to check for potential bugs.

**Github Usage** - Making sure that we manage our branches well is imperative, having too many branch conflicts can be a nightmare to resolve. Going over Github practices and having the GitHub person keep track of branches and merges goes a long way in mitigating this.

## Program Requirements

**Multiple Device Compatibility** - Keeping track of how our website looks on multiple different browsers, in addition to a phone screen, can be a daunting task. To combat these frequent check-ins during development for how each browser/browser side window looks with each page is a pretty good first step in helping resolve the issue.

**Security Vulnerabilities** - Even though this website won't be really going live with sensitive individual information, having good security practices when managing backend data is imperative in creating good full-stack services. Asking for help in this regard—someone to look over for vulnerabilities—might be a good step to take in order to shore up our software in this regard.

## Team

**Miscommunication/Poor communication** - Our communication has been pretty good thus far, however, that could change pretty rapidly depending on the situations presented to the team. Ensuring that we keep in contact every week at our specified meeting place—discord—and attending meetings will usually mean that this doesn't become an issue.

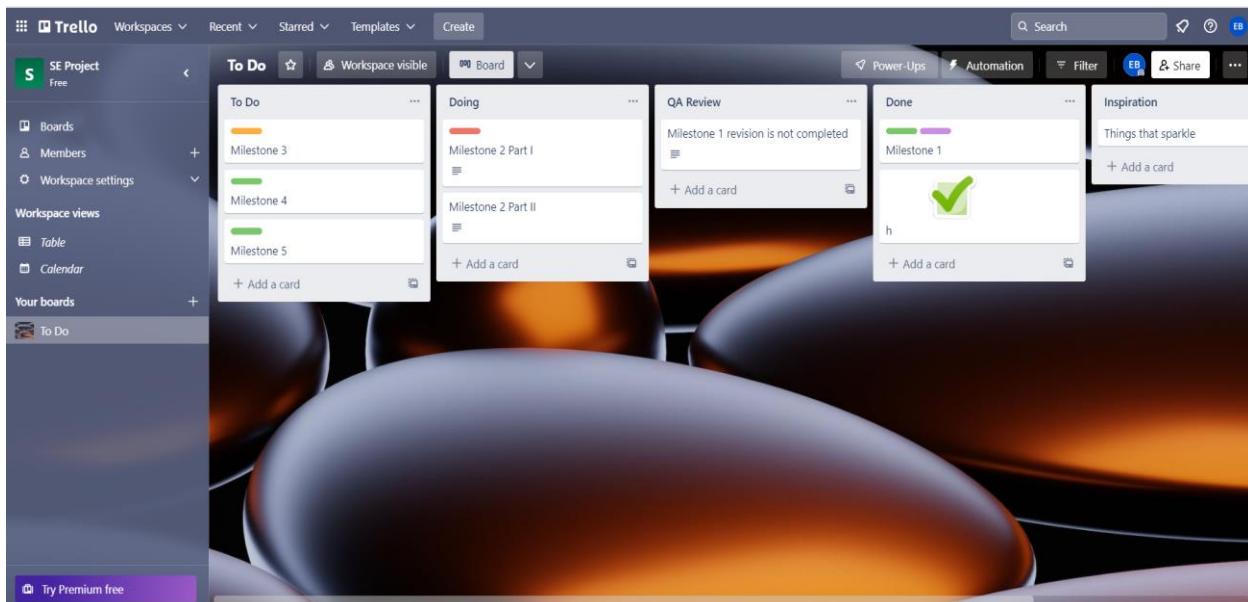
## 7. Project management:

As the team lead for our software engineering project, I have held regular team meetings every Friday to discuss project goals, expectations, and deadlines to ensure everyone is on the same page. During these meetings, I assign tasks and encourage team members to share their progress, concerns, and feedback so that we can identify potential issues early on and address them accordingly.

To help keep our team organized and focused, I create a google doc for every milestone that allows members to focus on the tasks, responsibilities, timelines, and resources needed to complete every milestone and future task. This google doc has been shared with all team members to ensure everyone understands their roles and responsibilities and working on them.

To monitor progress and track tasks, I will use a project management tool called Trello. This tool allows me to create boards for each project phase and assign tasks to team members with deadlines and priorities. It also enables me to monitor progress and identify potential delays.

In addition, I am also using communication tools such as Discord and Zoom to facilitate collaboration and communication among team members. These tools allow us to stay connected, share files, and have real-time discussions about the project.



## **SW Engineering CSC648/848 Spring 2023**

### **Milestone 3: Focused Group Feedback meeting (meeting notes)**

#### **Food Feast**

#### **Team 01**

Elahe Bashiri ([Ebashiri@sfsu.edu](mailto:Ebashiri@sfsu.edu)): Team lead

Abbas Mahdavi: Back end lead

Megan Lew: Front end lead

Nathan Rennacker: Github master

Jed Graves: Developer

Alexander Diaz: Developer

**Date:**04/26/2023

## Major Takeaways:

- Make search bar persistent and in the same place across all pages
  - Eliminate browse button
- Add cuisine table

## Other notes:

- Move SFSU project info bar to the top of the page
  - Make the menu more intuitive:
    - Tell the user what to do vs just having icons/pictures
  - Home page
    - Shrink the aesthetic features, take up too much space
    - “20-40 minutes delivery to SFSU” rather than “20-40 minutes” on restaurant tiles
  - User Registration page
    - Confirm email / password
  - Restaurant Registration Page
    - Increase text size in fields
    - Cuisine is a pulldown menu. User should not type cuisine manually
    - Where should the restaurant owner go after submitting? Button for homepage maybe?
  - Menu item entry
    - Submit/ next/ finish buttons
  - Driver page
    - Include the drivers name “Welcome <driver name> “
    - Include instructions for driver

- Display delivery schedule
  - Flag confirmed orders so other drivers can't accept it
  - Include campus map
- Driver Registration page
  - Confirm driver license checkbox
- Cart
  - Display username when logged in
  - Submit button
- Restaurant pages
  - Should link to a new page so you can click “back” on the browser

Github notes:

- Descriptive commit messages. No “bugfix” etc.
- Add inline comments for unusual or uncommon code

Database notes:

- “cuisine” column in restaurant is foreign key to cuisine table
- Add flag to restaurant table that signifies whether the restaurant is live on the site or not

P1 Tasks:

- Search
- Browse
- Results
- Order
- Restaurant functionality
- Registration
- Driver functionality

# **SW Engineering CSC648/848 Spring 2023**

## **Milestone 4: Product description, Beta Launch, QA, and Usability**

### **Testing**

### **Food Feast**

### **Team 01**

Elahe Bashiri ([Ebashiri@sfsu.edu](mailto:Ebashiri@sfsu.edu)): Team lead

Abbas Mahdavi: Back end lead

Megan Lew: Front end lead

Nathan Rennacker: Github master

Jed Graves: Developer

Alexander Diaz: Developer

### **Revision History**

Date Submitted	Date Revised
5/19/2023	-

## **1) Product Summary:**

Product Name: Food Feast

Food Feast is a dynamic web application designed to make finding meals and food delivery more efficient for the San Francisco State University (SFSU) community, comprising students, staff, and faculty. It targets the challenges of busy schedules and limited food options within the campus, by providing an accessible and user-friendly platform to search for nearby restaurants, order meals, and have them delivered right on campus or to the dorms. Food Feast stands as a strong supporter of local businesses, providing a unique channel to reach a specific customer base.

P1 Functions:

- Registration: Unregistered users shall be able to create an account using SFSU information.
- Browse Food Items: Unregistered users shall have the ability to view available food items.
- Map Accessibility: Unregistered users shall be able to access a visual map.
- Filter Application: Unregistered users shall be able to apply filters to restaurants.
- Restaurant Form Access: Potential restaurant owners shall have access to a registration form.
- Search Bar Usage: Users shall be able to use the search bar to find a specific restaurant.
- Login: Registered SFSU users shall have the capability to log in to their accounts.
- Order Placement: Registered SFSU users shall have the ability to select and order items.
- Restaurant Registration: Restaurant owners shall be able to register their establishment.
- Menu Upload: Restaurant owners shall have the ability to upload their menu.
- Delivery Interface Utilization: Delivery personnel shall have an interface to manage orders.
- Approval or Rejection of Restaurant Requests: The admin shall authority over approvals
- Managing Approved Restaurants: The admin shall be able to revoke privileges of restaurant
- User Management: The admin shall have the authority to manage the list of users.

## **Unique Proposition:**

Food Feast's uniqueness lies in its tailored solution for the SFSU community, merging technology, convenience, and local business support. Unlike general food delivery services, Food Feast offers a campus-oriented service, providing fast delivery within the campus and the dorms. Furthermore, our student verification system offers exclusive discounts to verified students, making nutritious meals more affordable. We also accept payments through Gator Passes, providing a seamless and speedy payment process for the SFSU community.

URL: <http://35.160.127.228/>

## **2) Usability test plan for a selected function: ( Search )**

**Test objectives:** The search function is being tested for our application. It is being tested because it is a priority one function. Unregistered users shall be able to type in the name of a restaurant or cuisine.

### **Test background and setup:**

System setup: Type the URL of the system to be tested.

Starting Point: Start on the homepage of the website.

Who are the intended users: The intended users of the search function would be anyone who is searching for food. This would be the faculty, staff, and students of SFSU.

URL of the system to be tested: Server URL: <http://35.160.127.228/>

### **What is to be measured: ( user satisfaction evaluation Likert scale )**

The efficiency of the search is being measured using a user satisfaction evaluation Likert scale. Users will use the same scale ( strongly agree, agree, neutral, disagree, strongly disagree). We can measure the average and standard deviation of the responses and the percentage overall of the responses.

**Usability Task Description:**

search for restaurants with American cuisine.

Using the cuisine filter, search for American cuisine.

Task	Description
Search	Search for American Restaurants
Search using cuisine filter	Using filter search for American
Successful Completion Criteria	Search results show American Restaurant

**Plan for evaluation of effectiveness:** First, test the search function using the search bar and using the cuisine filter. The task completion is measured by the percentage of people who were able to complete the defined task in the defined time. Count the number of errors by the number of restaurants showing up in the search result that were not defined by the search term or cuisine filter.

Can implement using table:

Test / Use Case	% completed	errors	comments
search	X%	X	X
search using filter	X%	X	X

**Plan for evaluation of efficiency:** For search, efficiency can be measured with efficiency in time and efficiency in content. Efficiency in time is the average time it takes the user to complete the search. Efficiency in content is the number of results from the search.

**Plan for evaluation of user satisfaction: ( provide 3 Likert scale questions)**

- The search was easy to use.
- The search results showed restaurants that matched the cuisine.

- The cuisine filter on the search showed search results matching the cuisine selected.

Scale is 1-5 with Strongly Disagree(1), Disagree(2), Neutral(3), Agree(4), Strongly Agree(5).

### 3) QA test plan and QA testing:

#### Unregistered Search Test

##### Setup:

Go to the home page of the application at <http://35.160.127.228/>, click on the yellow browse all restaurants button, no further setup is needed

#### Testing search functionality for unregistered users (in Chrome browser)

#	Title	Description	Input	Expected Result	Result
1	Empty Search	Testing empty search in search field	Press space and enter or simply press the search button on the right of the search bar	Eight restaurants should appear	PASS
2	% Like	Testing % like in search field for name	Type <i>indian</i> into the search bar and press enter	Two restaurants should show up in the results	PASS
3	Exact match	Testing exact match in the search field	Type <i>Roti Bistro</i> into the search bar and press enter	One restaurant called Roti Bistro should show up in the results	PASS
4	No matches	Testing no matches using the search field	Type <i>Ethiopian</i> into the search bar and press enter	No restaurants should show up in the results of the search	PASS
5	Long string	Testing excessively long search value	Type a string of 100+ characters or more into the search bar (use lorem ipsum for ease)	No restaurants should show up in the results of the search	PASS

## Registered Test Search

### Setup:

Go to the home page of the application at <http://35.160.127.228/> and press the login button at the top right.

In the username field, enter **Test10**

In the password field, enter **Test10**

Press the Sign in button, and you will be redirected to the home page as a signed-in user

Press the yellow browse all restaurants button

No further setup is needed

### Testing search functionality for registered users (in Chrome browser)

#	Title	Description	Input	Expected Result	Result
1	Empty Search	Testing empty search in the search field	Press space and enter or simply press the search button on the right of the search bar	Eight restaurants should appear	PASS
2	% Like	Testing % like in search field for name	Type <i>indian</i> into the search bar and press enter	Two restaurants should show up in the results	PASS
3	Exact match	Testing exact match in the search field	Type <i>Roti Bistro</i> into the search bar and press enter	One restaurant called Roti Bistro should show up in the results	PASS
4	No matches	Testing no matches using the search field	Type <i>Ethiopian</i> into the search bar and press enter	No restaurants should show up in the results of the search	PASS
5	Long string	Testing excessively long search value	Type a string of 100+ characters or more into the search bar (use lorem ipsum for ease)	No restaurants should show up in the results of the search	PASS

## Testing in Microsoft Edge Browser

#	Title	Description	Input	Expected Result	Result
1	Empty Search	Testing empty search in the search field	Press space and enter or simply press the search button on the right of the search bar	Eight restaurants should appear	PASS
2	% Like	Testing % like in search field for name	Type <i>indian</i> into the search bar and press enter	Two restaurants should show up in the results	PASS
3	Exact match	Testing exact match in the search field	Type <i>Roti Bistro</i> into the search bar and press enter	One restaurant called Roti Bistro should show up in the results	PASS
4	No matches	Testing no matches using the search field	Type <i>Ethiopian</i> into the search bar and press enter	No restaurants should show up in the results of the search	PASS
5	Long string	Testing excessively long search value	Type a string of 100+ characters or more into the search bar (use lorem ipsum for ease)	No restaurants should show up in the results of the search	PASS

 Abbas Mahdavi  
To: Jed Graves



2 attachments (112 KB)  Save all to OneDrive - San Francisco State University  Download all

Hello Jed,

Hope you are doing well.

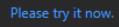
I am reaching out to you to request your assistance in reviewing the search functionality for our app.

Kindly review the attached code snippets attached, from the Browse.jsx file, and provide your valuable feedback.

Should you have any questions or require additional clarification, please don't hesitate to reach out.

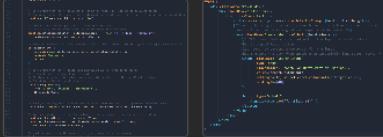
Your contribution is greatly appreciated. Thank you!

Best regards,  
Abbas M,  
Backend Lead

 Completed.  Ok, I will take a look.  Please try it now.

 Reply  Forward

 Jed Graves  
To: Abbas Mahdavi



2 attachments (140 KB)  Save all to OneDrive - San Francisco State University  Download all

Abbas,

I have reviewed the code for the search function and made some comments. Overall, it looks great! Good choices for usability and very readable code.

I did make a couple minor suggestions regarding naming and clarity, but aside from that, it appears to be a well put together function.

I have provided screenshots here. You can also view my comments in Browse.jsx file the development branch of the repository:

<https://github.com/CSC-648-SFSU/csc648-03-sp23-team01/blob/development/application/client/src/components/Browse.jsx>

Your header comment looks good and contains the necessary proprietary information and explanation of functionality, as do your GitHub commit messages.

Please let me know if you have any further questions.

Best,  
Jed G,  
Backend Engineer

 Reply  Forward

```

120    return (
121      <div className='browse-div'>
122        <div className='filter-div'>
123          <div className='filter'>
124            <Filter cuisines={cuisines} handleFilterChange={handleFilterChange} />
125            /* Good job using a form for the search bar. This allows the user to submit the form by
126            pressing Enter, which is a good user experience decision. */
127            <form className='search-bar' onSubmit={handleSearch}>
128              /* It's great that you're controlling the input value with state.
129              This is a good React pattern.
130              Also, good job setting a maxLength for the input.
131              This prevents the user from entering an excessively long search term. */
132              <input className='search-input'
133                type="text"
134                placeholder="Search for Restaurant or Cuisine..."
135                value={searchRestaurants}
136                onChange={(e) => setSearchRestaurants(e.target.value)}
137                maxLength={40}
138              />
139
140              <button type="submit">
141                <PageviewIcon sx={{fontSize: 60}} />
142              </button>
143            </form>
144          </div>
145        </div>
146      </div>
147    );
148  };
149
150  const handleSearch = (event) => {
151    // Good job preventing the default form submission behavior.
152    event.preventDefault();
153
154
155    // It's great that you're trimming the input to remove any leading or trailing spaces.
156    // However, the variable name searchRestaurants is a bit confusing, since it's actually the search term.
157    const searchTerm = searchRestaurants.trim();
158
159    // Nice use of ternary operator to handle the 'All' case.
160    // This is a clean and concise way to create the filter function.
161    const selectedCuisineFilter = selectedCuisine === 'All' ? () => true : (restaurant) =>
162      restaurant.cuisine_name === selectedCuisine;
163
164
165    // Good job handling the case where the search term is empty. This is a good user experience decision.
166    if (!searchTerm) {
167      setRestaurants(allRestaurants.filter(selectedCuisineFilter));
168      navigate('/browse');
169      return;
170    }
171
172    // It's great that you're specifying multiple keys for Fuse.js to search.
173    // This will make the search more robust.
174    // It might be helpful to add a comment explaining what the 'threshold' option does
175    // for those unfamiliar with Fuse.js.
176    const fuseOptions = {
177      keys: ['name', 'cuisine', 'description'],
178      threshold: 0.25,
179    };
180
181
182    // Good job reusing the 'selectedCuisineFilter' from earlier, as it prevents redundant code.
183    const fuse = new Fuse(allRestaurants.filter(selectedCuisineFilter), fuseOptions);
184
185
186    // Nice use of map to extract the actual restaurant objects from the search results.
187    const searchResults = fuse.search(searchTerm);
188    const searchedRestaurants = searchResults.map((result) => result.item);
189    setRestaurants(searchedRestaurants);
190
191
192    // Good job navigating the user to the browse page after the search.
193    // This is a good user experience decision.
194    navigate('/browse');
195  };

```



Abbas Mahdavi

To: Jed Graves



Sat 5/20/2023 7:58 PM

Hello again Jed,

Thanks for the feedback and getting back to me in a timely manner.

I have read your comments, they are very concise  
and practical. I will be sure to make the needed adjustments, based on your suggestions.

Thank you again!

Best,  
Abbas M.  
Backend Lead

Get [Outlook for iOS](#)

...

You are very welcome!

Sounds great, thank you!

Happy to help!

Reply

Forward

## **5) Self-check on best practices for security:**

### **Assets you are protecting:**

- **User Information:**
  - Major Threats: Leaked user email, and password, Users can modify other user's accounts, such as deletion
    - Regular users: Last order location, current order's delivery location
    - For restaurant Owners: Leaked Restaurant details, List of menu Items
    - For Drivers: Accessible known location, last/current accepted order
  - Process of protection: Keep the info in a private and secure database, encrypt the password, pass API requests only through protected routes, make sure important data is only available to the currently logged-in user, using sessions to keep track of the user's activity. Killing their session after a certain time has passed. Making sure the data stored in local storage is cleared when the user is logged out. Overall, keep every valuable information, such as location, email, and other info, in a database that is only accessible to the site admin, and only to the user.
  
- **System Infrastructure**
  - Major Threats: Accessible database information, not protected routes via middleware
  - Process of protecting: Keeping critical information/values in a .env file, that is kept only in Admin's local device. Ensuring the data is private, protected, and not posted anywhere, such as a GitHub repository. Additionally, making sure important routes, such as all admin routes, which have the ability to make major changes to the database, such as deleting users, and restaurants, and viewing the database tables and content. By having multiple layers of security, to routes, APIs, and call requests, both in the front end that denies access directly by browsing to users, but also backend modern and industry approved middleware security, for advanced hackers.

- **Ensuring Passwords are protected/encrypted:**
  - Passwords for ALL users are encrypted and are kept in the database
  - The encryption happens via crypt, which is a one-way encryption system, meaning it cannot be reversed.

<b>id</b>	<b>username</b>	<b>email</b>	<b>password</b>
43	Alex1	Alex1@gmail.com	\$2a\$10\$6oO.cPfDvzuUd6OQBj...

- **Confirming input Data Validation:**

- Search input has a maxLength of 40, allowing it a max and up to 40 characters in the search input:

Code:       <input className='search-input'

                type="text"

                placeholder="Search for Restaurant or Cuisine..."

                value={searchRestaurants}

                onChange={(e) =>

                    setSearchRestaurants(e.target.value)}

**maxLength={40}**

      />

- Ensuring only students with a valid sfsu email can signup as a regular user (This does not apply to drivers or restaurant owners)

Code:

// check if email ends with @sfsu.edu

```
const emailRegex = /^[w-.]+@sfsu.edu$/i;
```

```
if (!emailRegex.test(formData.email)) {
```

```
    setErr("Please enter a valid SFSU email address");
```

```
    return;
```

```
}
```

Other validations:

Validating password is min 5 char, lower and upper case, with a number

```
const passwordRegex = /^(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{5,}$/;
```

The rest are done in a similar fashion:

Confirming passwords match, all users agree to the terms of condition,

Restaurant enrollment validations, such as only \$ chars for price range, the number for a phone number, valid URL for pictures, etc.

## 6) Self-check the adherence to original Non-functional specs:

- Refer to system properties and constraints such as:
  - Reliability: **Done**
  - Response time: **Done**
  - HW and networking requirements: **Done**
  - Usability requirements: **Done**
  - Marketing, legal, licensing: **Done**
  - Media content (formats, size...): **Done**
  - Privacy: what is the data collected, how is the data used: **Done**
  - Compatibility (e.g. which browsers...): **Done**
  - Can refer to:
    - Product (product behavior like speed, reliability): **Done**
    - Organization (e.g. process, standards used): **Done**
    - External factors (e.g. branding, legal disclaimers displayed): **Done**

# Product Screenshots

## Home

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only CopyRight@ SFSU-FoodFeast-2023

 SFSU-FOODFEAST For Restaurants For Drivers About Us   Login Register

All  Search for Restaurant or Cuisine... 

Meals made simple.  
Food delivered anywhere  
on campus.  
Exclusive use for SFSU  
Students, Staff, & Faculty.



Featured Restaurants

  
Roti Bistro  40-45 mins delivery to SFSU

  
Nizaro's Pizza  10-15 mins delivery to SFSU

  
Yuuya Sushi  25-30 mins delivery to SFSU

Favorite Restaurants

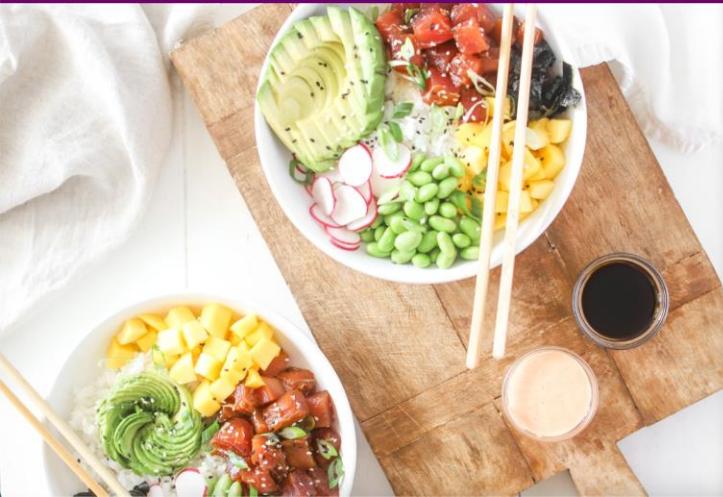
  


## Login

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only CopyRight@ SFSU-FoodFeast-2023

 SFSU-FOODFEAST For Restaurants For Drivers About Us   Login Register

All  Search for Restaurant or Cuisine... 



Sign in

Username \*

Password \*

Remember Me [Forgot Password?](#)



Don't have an account? [Sign Up here](#)

# Register

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST For Restaurants For Drivers About Us [Logout](#) [Login](#) [Register](#)

All



## Sign Up

Username \*

Email Address \*

Password \*

Confirm Password \*

I accept the Terms & Conditions

[Already have an account? Sign in](#)

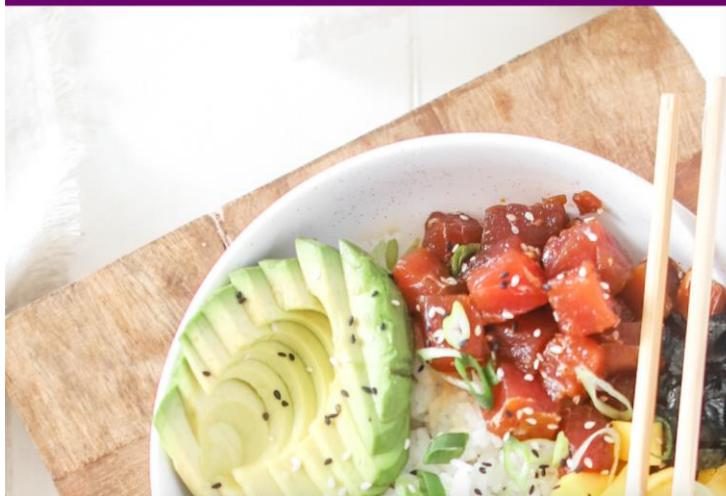
# Restaurant Owner Registration

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST For Restaurants For Drivers About Us [Logout, \(Test10\)](#)

All



## Register Restaurant

Username \*

Email Address \*

Password \*

Confirm Password \*



## Restaurant Owner Registration (Continued)



SFSU-FOODFEAST

For Restaurants For Drivers About Us [Logout, \(Test10\)](#)

Restaurant Address \*

Select Category

Price Range (\$ - \$\$\$\$\$) \*

Current Yelp Star Rating (1-5) \*

Wait Time in Mins (e.g. 20) \*

Store Hours (e.g. 9am-7pm) \*

Store Image URL \*

Description \*

## Driver Registration

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us [Logout](#) [Register](#)

All |  

Sign Up as a Driver

Username \*

Email Address \*

Password \*

Confirm Password \*

I have a valid Drivers License

I accept the Terms & Conditions

# Driver (Unregistered Driver)

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Testio)

All

Search for Restaurant or Cuisine...



Welcome! This page is exclusively for our valued drivers.

We couldn't find a driver account associated with your profile.

Would you like to join our driver community and start earning with us?

If you're ready, simply click the button below to register your account as a driver.

Please note: A valid driver's license is required to proceed.

If you prefer, you can also log out and create a new account when you return to this page.

Feel free to explore other areas of our platform if this doesn't apply to you.

Register your account as a driver today and unlock earning opportunities!

Become a Driver

# Driver Dashboard

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Testio)

All

Search for Restaurant or Cuisine...



Welcome, Test5

Accept an Order, from the list below to Complete!

Campus Map Map Near Me

Order ID: 398

Restaurant Name: Roti Bistro  
Est. Delivery Time: 45 minutes

Restaurant Phone: (615)123-4567

Payout: \$8

Accept

Order ID: 444

Restaurant Name: Clay Oven  
Est. Delivery Time: 30 minutes

Restaurant Phone: (615)123-4567

Payout: \$8

Accept

Accepted Order

Order ID: 446

Restaurant Name: Roti Bistro  
Order Total: 17.19  
Order Placed on: 5/23/2023, 11:19:00 AM

Deliver to: SFSU Campus: hss 455  
Special Instructions:

Expected Delivery Time: Within [ 45 ] minutes

Mark as Picked Up



# Restaurant Dashboard

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023

The screenshot shows a purple header bar with the SFSU-FOODFEAST logo and navigation links for 'For Restaurants', 'For Drivers', 'About Us', and 'Logout'. A search bar is also present. The main content area displays a welcome message 'Welcome, Nizario's Pizza!' and three buttons: 'Orders Page', 'Add Menu Items', and 'Visit your Restaurant Page'. Below these are several input fields containing placeholder text: 'Cuisine: 2', 'Description: Local pizzeria chain serving a menu of classic & creative pies in a low-key setting.', 'Estimated Delivery Time: 15', and 'Address: 393 W Portal Ave, San Francisco, CA'. A photo of a pizza is shown.

## Restaurant Dashboard (Continued)

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023

The screenshot shows a purple header bar with the SFSU-FOODFEAST logo and navigation links for 'For Restaurants', 'For Drivers', 'About Us', and 'Logout'. A search bar is also present. The main content area displays a 'Menu' section. It lists two menu items: 'Cheese Pizza' and 'Combo Pizza'. Each item has its name, price (12.00), description, and a 'Delete Menu Item' button. Photos of the pizzas are shown next to their respective names.

# Orders Page

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023

The screenshot shows the 'Orders Page' with a purple header. The header includes the project name 'SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only' and copyright information 'CopyRight@ SFSU-FoodFeast-2023'. The logo 'SFSU-FOODFEAST' is on the left, and navigation links 'For Restaurants', 'For Drivers', 'About Us', 'Login', and 'Register' are on the right. Below the header is a search bar with dropdown menus for 'All', 'Cuisine', and 'Restaurant', followed by a search icon. A sidebar on the left says 'Unaccepted Orders' with tabs for 'Current Orders' (selected) and 'Completed Orders'. The main content area displays an order summary for Order ID 427, Customer ID 71, and Status: In Progress. It lists items: Total: 53.49, Items in Order: Item id: 419, price: 15.00, special requests: Item id: 420, price: 15.00, special requests: . A green button labeled 'Order Complete' is at the bottom.

# Menu Entry

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023

The screenshot shows the 'Menu Entry' page with a purple header. The header includes the project name 'SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only' and copyright information 'CopyRight@ SFSU-FoodFeast-2023'. The logo 'SFSU-FOODFEAST' is on the left, and navigation links 'For Restaurants', 'For Drivers', 'About Us', 'Login', and 'Register' are on the right. Below the header is a search bar with dropdown menus for 'All', 'Cuisine', and 'Restaurant', followed by a search icon. The main content area features a wooden background image on the left and right. In the center is a white 'Menu Entry' form with fields for 'Item Name\*', 'Price\*', 'Description\*', and 'Item Image URL\*'. It includes two purple buttons: 'ADD ITEM' and 'FINISH & RETURN'.

# Restaurant Page

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Test5)

All



Search for Restaurant or Cuisine...



## Nizaro's Pizza

Local pizzeria chain serving a menu of classic & creative pies in a low-key setting.



## Menu



Cheese Pizza  
\$ 12.00

Add To Cart



Combo Pizza  
\$ 12.00

Add To Cart



Pepperoni Pizza  
\$ 12.00

Add To Cart

# Empty Cart

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Test5)

All



Search for Restaurant or Cuisine...



**Test5, Your Cart is currently empty.**

Keep exploring and add delicious items to continue your food journey!



NO ITEMS IN CART

Shop Now

## Cart with order

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Test10)

All | Search for Restaurant or Cuisine...



Search

### Welcome to Your Cart, Test10



Shrimp Ramen

- 1 +

\$15.00

**Sub-Total:**  
**\$15.00**

**Tax:** **\$1.50**

**Delivery Fee:**  
**\$3.99**

**Total:** **\$20.49**

Add more items

Checkout

## Checkout Page

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Test10)

All | Search for Restaurant or Cuisine...



Search

### Checkout

Please provide your building name (e.g. HSS) & Room Number (e.g. 356)

Building \*

Room \*

Please select your payment method type!

Payment Method \*

Special Instructions

NEXT

## Confirmed Checkout

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023

The screenshot shows a confirmation message box in the center of the screen. The message reads:

Dear Test10, Your order with:  
1 items, for \$20.49 (Payment via Student ID),  
has been placed, and will be delivered to  
SFSU Campus: 1 1  
Please check your email for the order receipt and  
status!  
Thank you for choosing FoodFeast

At the bottom right of the message box is a yellow "Close" button. Below the message box, there is a summary of the order details:

\$15.00
Sub-Total: \$15.00
Tax: \$1.50
Delivery Fee: \$3.99
Total: \$20.49

At the bottom of the screen, there are two yellow buttons: "View Order" and "Place Order".

## Search (All)

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023

The screenshot shows a search results page with the heading "Showing 9 results". There are three restaurant cards displayed:

- Roti Bistro** *Indian*  
Modern, warm-hued Indian eatery turning out traditional, innovative & vegetarian options.  
★★★★★ 40-45 mins
- Clay Oven** *Indian*  
Casual Indian chain branch specializing in tandoori dishes served in a banquet-style space.  
★★★★★ 25-30 mins
- Gold Coast Burger** *American*  
Classic American Retro-Style Burger Joint.  
★★★★★ 15-20 mins

## Search (cuisine)

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us



Logout,  
(Test5)

Japanese | Search for Restaurant or Cuisine...



Showing 2 results



★★★★★ 25-30 mins

★★★★★ 20-25 mins

## Google Maps API

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us

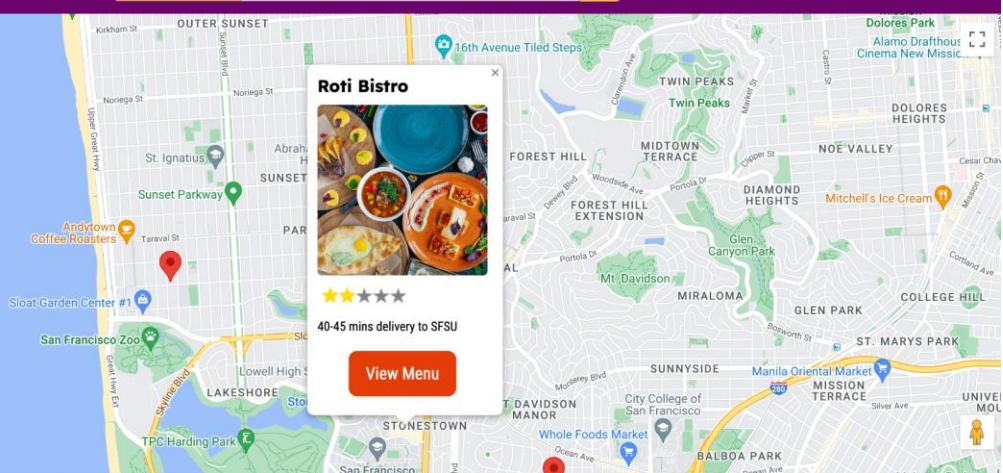


Logout,  
(Testio)

All | Search for Restaurant or Cuisine...



Map Satellite



# About Us

SFSU Software Engineering Project CSC 648-848, Team01, Spring 2023. For Demonstration Only

CopyRight@ SFSU-FoodFeast-2023



SFSU-FOODFEAST

For Restaurants For Drivers About Us

Logout  
(Test10)

All | Search for Restaurant or Cuisine...



Elahe Bashiri  
Team Lead



Abbas Mahdavi  
Backend Lead



Jed Graves  
Developer



Alex Diaz  
Developer

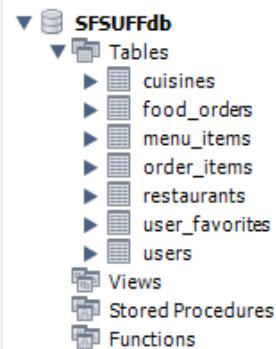


Megan Lew  
Frontend Lead



Nathan Rennacker  
Github Master

## Database Organization



The food\_orders table contains the data necessary to send an order from the cart, to the restaurant, to the driver.

**Table: food\_orders**

**Columns:**

<b>order_id</b>	int(11) AI PK
<b>customer_id</b>	int(11)
<b>restaurant_id</b>	int(11)
<b>order_date</b>	datetime
<b>order_status</b>	enum('Pending','In Progress','Ready for Pickup','Out for Delivery','Delivered','Cancelled')
<b>order_total</b>	decimal(10,2)
<b>delivery_address</b>	varchar(255)
<b>payment_method</b>	varchar(255)
<b>special_instructions</b>	text
<b>order_accepted_by_driver</b>	tinyint(4)

The cuisines table contains an integer key and cuisine name for ease of searching and sorting

**Table: cuisines**

**Columns:**

<b>id</b>	int(11) AI PK
<b>cuisine_name</b>	varchar(255)

The menu\_items table contains the menu items from all the restaurants, linked to each restaurant through the restaurant\_id foreign key.

**Table: menu\_items**

**Columns:**

<b>id</b>	int(11) PK
name	varchar(45)
price	decimal(10,2)
restaurant_id	int(11)
image	varchar(255)
description	text

The order\_items table contains the information about each item in the user's order. This is linked to the food\_orders table through the order\_id foreign key.

**Table: order\_items**

**Columns:**

<b>order_item_id</b>	int(11) AI PK
<b>order_id</b>	int(11)
<b>menu_item_id</b>	int(11)
quantity	int(11)
price	decimal(10,2)
item_total	decimal(10,2)
special_requests	text
item_name	varchar(255)

The Restaurants table contains all the information for each restaurant, including the foreign key to the users table to indicate the owner, and contact information for the driver.

**Table: restaurants**

**Columns:**

<b>id</b>	int(11) AI PK
name	varchar(25)
price	varchar(5)
cuisine	int(11)
description	text
rating	int(11)
est_delivery_time	int(11)
address	varchar(100)
picture	varchar(255)
is_featured	tinyint(1)
phone	varchar(20)
hours	varchar(50)
<b>owner_id</b>	int(11)

The userFavorites table contains the user's favorited restaurants so they will be persisted across login instances.

**Table: userFavorites**

**Columns:**

<b>user_id</b>	int(11) PK
<b>restaurant_id</b>	int(11) PK

The users table contains all of the information about the users, including the flags that indicate authorization level, such as if the user is a driver or restaurant owner.

**Table:** `users`

**Columns:**

<code>id</code>	int(11) AI PK
<code>username</code>	varchar(225)
<code>email</code>	varchar(255)
<code>password</code>	varchar(255)
<code>isAdmin</code>	tinyint(1)
<code>isRestaurantOwner</code>	tinyint(1)
<code>isDriver</code>	tinyint(1)

## Github organization

Our github branches were organized into feature branches, a dev branch, and our main branch. Our backend lead and github master had access to our main github branch.

The screenshot shows a GitHub repository page for the team 'csc648-03-sp23-team01'. The repository is private. At the top, there are links for Pull requests, Issues, Codespaces, Marketplace, and Explore. On the right, there are buttons for Unwatch (2), Fork (0), and Star (2). Below the header, there are tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The 'Code' tab is selected. On the left, there is a sidebar with a 'main' branch dropdown, a '9 branches' dropdown, and a '0 tags' dropdown. To the right of the sidebar, there are buttons for Go to file, Add file, and a dropdown for Code. The main content area shows a list of commits:

Author	Commit Message	Time Ago
AbbasMahdavi021	Fix css overlapping with orderspage and dri...	16 hours ago
Milestones	M0-IT choices	last week
application	Fix css overlapping with orderspage and driver's page	16 hours ago
credentials	Update README.md	2 months ago
LICENSE	Initial commit	3 months ago
README.md	Update README.md	5 days ago

On the right side, there is an 'About' section with the repository name 'csc648-03-sp23-team01' and links for Readme, MIT license, 2 stars, 2 watching, and 0 forks. Below the 'About' section is a 'Releases' section.

## Google analytics stats plot for your WWW site

Google Analytics Report for SFSU FOOD-FEAST

Date: 5/24/23

Website: <http://35.160.127.228/>

### Overview

This is a summary of our website performance via Google Analytics with the main key metrics and insights.

### Audience Overview

Total Sessions: 14

Unique Visitors: 7

The website has not seen much traffic as it's a new deployed site, for a group project!

### Traffic Sources

Direct: 100%

Organic Search: 0%

Referral: 0%

All the Traffic are from direct visits via links/ip addresses. This is infact good because we don't want real users to reach this site as it's for demonstration only!

### Device Category

Desktop: 98%

Mobile: 2%

### Top Pages

Home

Login

Browse

About Us

### User Behavior: Non existence

### Conclusion

Based on the data provided, it is clear that this site is a demonstration only, and does not have real visitors. This was expected, but getting to use Google Analytics was fun and will help us land a job in the near future. I hope.

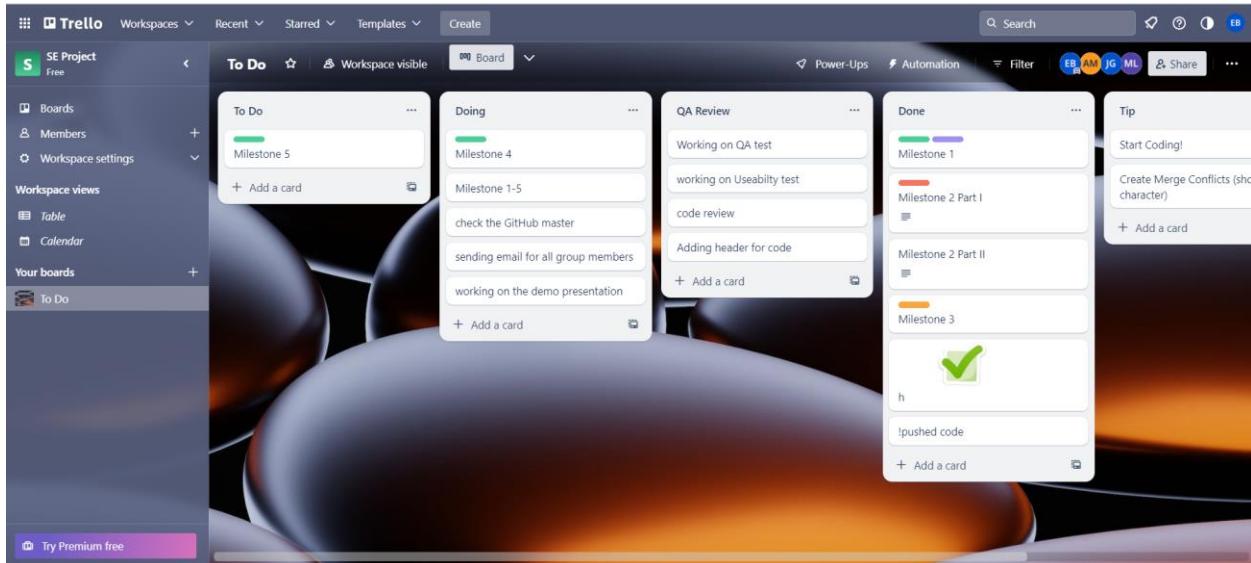
## Project management

As the team lead for our software engineering project, I have held regular team meetings every Friday to discuss project goals, expectations, and deadlines to ensure everyone is on the same page. During these meetings, I assign tasks and encourage team members to share their progress, concerns, and feedback so that we can identify potential issues early on and address them accordingly.

To help keep our team organized and focused, I create a google doc for every milestone that allows members to focus on the tasks, responsibilities, timelines, and resources needed to complete every milestone and future task. This google doc has been shared with all team members to ensure everyone understands their roles and responsibilities and working on them.

To monitor progress and track tasks, I will use a project management tool called Trello. This tool allows me to create boards for each project phase and assign tasks to team members with deadlines and priorities. It also enables me to monitor progress and identify potential delays.

In addition, I am also using communication tools such as Discord and Zoom to facilitate collaboration and communication among team members. These tools allow us to stay connected, share files, and have real-time discussions about the project.



# Elahe Bashiri

---

**From:** Elahe Bashiri  
**Sent:** Sunday, May 21, 2023 9:45 AM  
**To:** Abbas Mahdavi; Jed Graves; Megan Lew; Alexander Adrian Diaz; Nathan Rennacker  
**Subject:** Summary of contributions

Dear All,

I hope this email finds you all well. Here is a summary of my contributions to the project.

**a) My contributions:**

As the team lead for the project, I have significantly contributed to the team's success through my technical skills and effective teamwork. Here is a list of my contributions:

- 1) Project Management: I have successfully managed the team throughout the project lifecycle, ensuring smooth coordination and efficient execution of tasks. This involved creating and maintaining project plans, assigning responsibilities, and monitoring progress to meet deadlines.
- 2) Agile Methodology: I have implemented Scrum practices within the team, facilitating regular stand-up meetings, sprint planning, and retrospective sessions. By adhering to Agile principles, I promoted collaboration, adaptive planning, and continuous improvement among team members.
- 3) Technical Expertise: I actively participated in the technical aspects of the project, including coding and development tasks. I leveraged my skills and knowledge to contribute to the implementation of key features, ensuring high-quality code and adherence to best practices.
- 4) Task Distribution: I took charge of splitting project tasks among team members, considering individual strengths and areas of expertise. By allocating responsibilities effectively, I maximized productivity and ensured a balanced workload for each team member.
- 5) Conflict Resolution: I proactively addressed conflicts and disagreements within the team, promoting open communication and fostering a positive work environment. Through active listening and intermediary, I facilitated the resolution of issues, ensuring that team dynamics remained harmonious.
- 6) Motivation and Support: I consistently provided motivation and support to team members, acknowledging their efforts, and celebrating achievements. I boosted team morale and enhanced overall productivity by fostering a sense of friendship and creating a supportive atmosphere.

**b) GitHub commits: 21.**

**c) Main challenges:**

- 1) Scheduling Difficulties: Coordinating meeting times that work for all team members, particularly as deadlines approached, proved challenging.
- 2) Engaging Introverted Team Members: Encouraging participation from introverted individuals in the early stages of the project posed a challenge.
- 3) Lack of Communication and Updates: Ensuring consistent updates from team members about their progress was challenging.
- 4) Engagement in Group Chats: Some team members faced difficulties engaging actively in group chat discussions.

**d) Improvements:**

Based on what I have learned in the class about software engineering management and processes, there are several areas where I can improve as a team lead in future projects. Firstly, I would work on implementing a more efficient and streamlined scheduling process, considering the availability of team members and potential conflicts closer to project deadlines. Secondly, I would focus on creating a more inclusive and supportive environment from the project's inception, ensuring that all team members, particularly introverted individuals, feel empowered to

contribute. Additionally, I would further develop my conflict resolution skills to address any conflicts that arise promptly and effectively.

Best,  
Elahe Bashiri<Team Lead>

# Elahe Bashiri

---

**From:** Megan Lew  
**Sent:** Monday, May 22, 2023 6:17 PM  
**To:** Elahe Bashiri; Abbas Mahdavi; Jed Graves; Alexander Adrian Diaz; Nathan Rennacker  
**Subject:** Summary of contributions

Dear team,

I hope this email finds you well. Here is a summary of my contributions to the project.

## a. My Contributions

As the front-end lead for the project, I have contributed to the team's success through technical skills and implementing visual elements.

1. Project Management: I have successfully contributed to the team through project lifecycle, designing wireframes/mockups, developing user interfaces, and determining structure of the web application.
2. Agile Methodology: I have actively participated in scrum practices with the team and regularly attended stand up meetings and sprint planning.
3. Technical Expertise: I have actively participated in the technical aspects of the project. I advocated for implementation of key designs and features ensuring smooth flow of our front-end application.
4. User Experience: I implemented UX/UI designs and practical GUI development process with consistent design patterns commonly used in industry. I have successfully validated our application's user experience through a usability test plan.
5. Conflict Resolution: I actively engaged in discussions with the team and resolved conflicts with clarification on my work.
6. Motivation and Support: I supported my team members through acknowledgement of their work and elaborated on their ideas. As a result, our work environment resulted in more creativity and better performance.

## b. Github Commits: 28

## c. Main challenges:

1. Communication: Design Indecision: In the early stages of prototyping, team members wanted to add suggestions to the design. However, most of the design has already been completed prior to receiving their message about the suggestions. Better communication was needed.
2. Scope of Project: Due to the large scope of the project, there were many structure changes. There was a lack of consistent updates on structure changes and progress updates from team members. Communicating progress updates and changes in structure is crucial for effective delegation of front-end tasks. Without consistent updates, it was difficult to ensure that everyone was on the same page.

## d. Improvements:

Based on what I learned about software engineering management and processes, there are several areas where I can improve as a front-end lead. First, I would work on advocating for a better task management system. This would ensure everyone has regular updates on progress and any new changes in project structure. Secondly, I would encourage my team members to provide suggestions prior to working on designs/ implementing features. Additionally, I would further develop in my role as front end lead by finding an efficient solution for delegating front end tasks.

Best,  
Megan Lew  
Front End Lead

## Elahe Bashiri

---

**From:** Jed Graves  
**Sent:** Monday, May 22, 2023 9:28 PM  
**To:** Elahe Bashiri; Abbas Mahdavi; Nathan Rennacker; Megan Lew; Alexander Adrian Diaz  
**Subject:** Summary of Contributions

Hello team,

Below I have included a summary of my contributions to the project.

### a. My Contributions:

1. As a backend developer for this project, I implemented the following functionality:
  - The backend and most of the frontend logic for the driver page
  - The backend and frontend for the restaurant orders management page
  - The backend and most of the frontend logic for the restaurant dashboard
  - The backend for order creation
  - The backend for registering a restaurant
  - The backend for adding a menu item
  - The map, using the Google Maps and Geocoding APIs
  - The context for managing user details across pages and page refreshes
  - Socket.io for updating drivers and restaurants with orders in real time
  - The backend for featured and favorite restaurants
  - Backend logic for managing orders and order statuses
2. I also made various tweaks and adjustments to many other parts of the project to ensure functionality and useability.
3. Stood in for the team lead when she was out due to illness.

### b. Github Commits: 51

### c. Main Challenges:

The biggest challenge that I faced was trying to keep files and branches organized. We were able to get a handle on it toward the end of the project, but there were definitely growing pains. Another challenge was communication. As we all know, developers like to sit in a dark room by themselves and never interact with anyone. Unfortunately, this does not work in the real world, and we experienced first hand why communication is so important in the world of software engineering. Fortunately, we were able to catch this issue early on in development, and were able to establish a continuous line of communication through Discord.

**d. Improvements:**

If I were to start another project from scratch, one of the first things I would do is establish a way for the whole team to track exactly what files they are working on and on what branch, that way there are never any merge conflicts. If a team member needs to modify a file, they would first check to see if any other members have uncommitted changes to that file. Another thing I would do (and to be fair, this doesn't work as well in a learning environment) is to ensure that the leadership hierarchy is observed. The team lead should pass tasks to the back and frontend lead, who pass it to their teams. This is a time-tested way to ensure accountability, and also limits the need for micromanaging from the team lead.

Best,

Jed Graves

Backend Developer

# Elahe Bashiri

---

**From:** Nathan Rennacker  
**Sent:** Monday, May 22, 2023 10:30 PM  
**To:** Jed Graves; Abbas Mahdavi; Elahe Bashiri; Megan Lew; Alexander Adrian Diaz  
**Subject:** Summary of Contributions

Hello everyone, hope you all are ready for the end of the semester.

Here's my summary of contributions:

a. My Contributions:

Constructing the front-end for the 'About Us' pages and laying the foundation of a general React structure for the project.

Designing mock-up UIs using Figma for project milestones and assisting in the design of different parts of the application.

Working on tasked parts of each milestone.

Participating in the creation of the home page for our application, including featuring and favoriting restaurants.

Assisting in the development of the menu pages for our application.

Ensuring thorough documentation (header comments) for every file within our application, enhancing readability and understanding.

b. Github Commits: I made 29 commits to the project's GitHub repository.

c. Main Challenges:

Admittedly, I've struggled with deadlines in the past, mainly due to unclear timelines. This project wasn't an exception and sometimes caused me to shift focus to other classes. I think if deadlines were clearer and more frequently communicated, it would have helped me focus better. Another thing I struggled with was suggesting project changes, as I typically avoid conflict. Moving forward, I understand the need to strengthen my ability to express my viewpoints more assertively, despite the possible discomfort. Despite these challenges, I am satisfied with the overall outcome of the project and look forward to applying these insights in the future.

d. Improvements:

Looking ahead, the first step towards improvement would be to enforce more robust GitHub rules, particularly with regard to branch management and feature branches. I acknowledge that I could have been more proactive in these areas during this project. Additionally, I aim to enhance my ability to voice my opinions assertively, while respecting the decisions of the project leadership, ensuring that my input is acknowledged.

All the best,  
Nathan Rennacker



# Elahe Bashiri

---

**From:** Alexander Adrian Diaz  
**Sent:** Monday, May 22, 2023 10:56 PM  
**To:** Abbas Mahdavi; Elahe Bashiri; Megan Lew; Jed Graves; Nathan Rennacker  
**Subject:** Summary Contributions

Dear All,

Here is a summary of my contributions to the project.

a. **My contributions:**

As the front-end developer I have contributed to the team's success through technical skills.

- The front-end design for the driver page
- Implemented UI design and styling for driver page
- Mockups and storyboards for the driver page
- Create backend to enable regular users to turn their account into a driver account

b. **GitHub commits:** 6(xanderbx and Diaz5-hub)

c. **Main challenges:**

1. Scheduling Difficulties: Meeting times were left towards the end of the week and could meet for an hour max as some of us were managing school, and work.
2. There were a couple of changes in the project that changed the pages layout and would have been better if there had been an agreement on the pages and hadn't changed afterwards.

d. **Improvements:**

I would say is to space out my pushes as each push consists of a lot of code and changes, but I only have a few pushes which gives the idea that not much work was done on my part. I would also like to put more thought into designing the driver page as I wish I would've had a less stressful semester so I could have put more thought and effort into it.

Best,  
Alexander Diaz,  
Front-end Developer

## Elahe Bashiri

---

**From:** Abbas Mahdavi  
**Sent:** Wednesday, May 24, 2023 8:44 AM  
**To:** Jed Graves; Elahe Bashiri; Alexander Adrian Diaz; Megan Lew; Nathan Rennacker  
**Subject:** Summary of Contributions

Dear Team,

I hope this email finds you all well. I wanted to take a moment to share my self-assessment and contributions to our project. Please see the details below:

a. My Contributions:

1. BackEnd Leadership and Teamwork:

- Successfully deployed the core structure of the site using create-react-app, ensuring a clean, modern, and organized folder structure.
  - Developed the backend architecture, including database design, API endpoints, and controllers for frontend queries.
  - Created the database system and established a connection on AWS EC2 instance with the assistance of team member Jed G.
  - Managed the deployment of the site on the instance along with nginx.
  - Took responsibility for writing approximately 70% of the CSS styling, including animations for frontend react components.
  - Implemented frontend API calls using axios.
  - Added authentication middleware to specific routes.
  - Set up express sessions and incorporated other necessary npm modules such as nodemailer (for sending confirmation emails to users) and pm2 for permanent deployment.
  - Configured CORS to ensure smooth communication between frontend and backend.
  - Successfully connected frontend forms, developed by other team members, to backend controllers.
  - Added tables to the MySQL schema.
  - Skillfully mapped API response data into stylish frontend components.
  - Accomplished full-stack features such as the admin dashboard, cart functionality, browse/search/filter options, header navbar, register feature, restaurant context, Home view, and restaurant view.
  - Adapted to client demands and implemented changes to the application effectively.

2. Effective Communication and Git Control:

- Maintained good communication by boosting team morale and ensuring proper Git control. I actively notified the team before pushing code to avoid merge conflicts and sought approval before working on files that were frequently accessed. As a result, I had the second-highest number of messages sent in our group Discord server.

b. Github Commits: 125

c. Main Challenges: Encouraging Team Engagement:

One of the main challenges we faced was the lack of engagement from some team members. It became evident that if any team member did not gain sufficient familiarity with the codebase, their

participation in the project's ongoing development became increasingly difficult. Therefore, I learned the importance of engaging teammates in coding activities as early as possible to prevent confusion and facilitate smooth collaboration. I would like to acknowledge the backend member for actively working with me on the project, which significantly contributed to its success.

d. Areas for Improvement:

In future full-stack applications, I would like to explore and incorporate additional technologies, such as delving deeper into webSocket and utilizing AWS S3 bucket for storing images, which is a common industry solution. Additionally, I aim to acquire a free domain for the site through my student GitHub account (serving the frontend with Amplify) and consider implementing global state management using Redux.

I want to express my appreciation for the opportunity to work with all of you. It has been a great experience collaborating on this project together. Thank you for your contributions and dedication. It was truly a pleasure working with each of you.

Best regards,  
Abbas Mahdavi  
Backend Lead