SW Engineering CSC648/848 Spring 2021

HungryGators-19

Food Delivery and Restaurant Review Service

Available on Mobile and Desktop

Team 01

Branden J. Nguyen – Team Lead – bnguye11@mail.sfsu.edu
Jasmine Jahan – Front End Lead
Yongjian Pan – Back End Lead
Robert Freeman – GitHub Master
Gurjot Singh – Team Member Front End
Tin Thu Zar Aye – Team Member Back End

Milestone 1

March 5, 2021

Date Submitted	Date Revised
March 5, 2021	

Table of Contents

1. Executive Summary:

San Francisco State University is a place of cultural diversity. There are many people including students, faculty, and staff from different parts of the world whose food habits are also different from one another. Though there are a variety of cafeterias and canteens in SFSU, they are not available for online delivery in the campus area. Most of the time students must go to the cafeteria to get food. This can be a hassle especially if students are having food in between classes, in which case, they may not have time to walk to the cafeteria, order the food, wait and be late for class. As well, students must select food at their own risk without reading any reviews or ratings which could be misleading. There are some other applications to get food delivered from larger restaurants, but they are only available outside the SFSU campus area. The process is inconvenient and hard to navigate.

We, Team 01, are building a web-application named HungryGators-19 by which students, faculty and staff can view and leave credible reviews for restaurants, as well as order food to be delivered from these establishments straight to their classroom door or dorm securely and efficiently. With this food delivery service students, faculty and staff will be able to order food by creating an account with their sfsu.edu email and name. If the person does not belong to the university, they cannot order from the app. For the person who is unable to use the app, they can get the phone number of the restaurants from the app and make a pickup order. The minimum viable product of our internet application can be described as a restaurant review and delivery service solely catered to San Francisco State University students and staff. This app will not only solve the problem of convenient food delivery for members of SFSU who all have drastically varying schedules, but it will also save invaluable time which people within the university setting are always short of. Moreover, this web-application is a fantastic platform for new or established restaurants looking to spread their customer base and garner new business opportunities.

The application capitalizes on the vastly growing food delivery industry, as well as the ever-growing population of the SFSU community. The cornerstone of good execution of this business is the great variety of food options to choose from, and the convenient and safe delivery of the food. We would exceed our competitors by not only serving exclusively SFSU students, but also capitalize on the unique niche of visitors of the campus, which are otherwise an untapped resource. With an emphasis on specific delivery drop-off locations, ease-of-use, as well as a great variety of restaurants, this service will appeal greatly to the university community. HungryGators-19 is an application which will revolutionize food delivery service for students at San Francisco State University.

2. Personae and Main Use Cases:

Personae:

John Full-Time staff at SFSU	About John: · He is a busy full-time staff at SFSU · He rarely cooks at home and he loves to eat the food from restaurants. Goals and Scenario: · Since is busy full-time staff at SFSU, he orders the food from the app at least 4 times a week. · Every time he orders the food, he is more concerned about the rating of the restaurant than what kind of meal he wants because he thinks that the more rating the restaurant gets, the tastier the food is. · Therefore, he is always looking for new suggestions about the restaurant.

Diane Full-Time student at SFSU	About Diane: · She is a full-time student at SFSU. · She goes to school all day and she works part time. · She does not have time to cook. · She orders the meal through the app regularly. Goals and Scenario: · Since she is a full-time student and works part-time, she does not have time to cook. · She orders the food through the app regularly, she becomes a premium member of the app. · Sometimes, she gets the coupons to get the discount. · Sometimes, she also gets the delivery fees waive. · Whenever she orders the meal, she is more concerned about the price than the food and nutrition.

Jack Full-Time Faculty at SFSU	About Jack: · He is a full time SFSU's faculty. · He works from 8:00am to 5:00pm · He is always ordering the food from the app for his breakfast and lunch because he does not have time to prepare in the morning. Goals and Scenario: · Since he is a SFSU's faculty, he gets not only a 10% discount, but he also gets free delivery whenever he orders the food from the app. · He becomes the premium member of the app because he uses the app every day.

Leo Visit to SFSU	About Leo: · He is not a SFSU student. · He comes to SFSU to visit his friend. · He tries to order food from the app while waiting for his friend. Goals and Scenario: · He can't order food from the app because he is not a SFSU' student. · He cannot even create the account for the app because the app is only for the SFSU's students and staff. · The app also provides the phone number of the restaurant. · The app also provides the pick-up option for the people who do not belong to SFSU.

Mary Retired faculty Of SFSU	About Mary: · She is a 70 years-old grandmother. · She used to work at SFSU. · She wants to order the pizza for her 7 years old grandchild who is visiting her house. Goals and Scenario: · She rarely orders food through the app, she only has basic skills of using the app for ordering the food. · After she finds the restaurant, she looks through the menu and she wants to add in the cart but she isn't sure how to do it. · She just orders through the phone from the nearest restaurant and picks up the food.

Liam Undergraduate Student at SFSU	About Liam: · He is an undergraduate student at SFSU. · He is a vegetarian. · He is busy with his class schedule for the whole day. Goals and Scenario: · He gets time 1pm-2pm to eat lunch. · Since he is a vegetarian, he searches the vegetarian restaurant from the app to order lunch. · He gets only one hour to eat lunch, so he looks for the nearest vegetarian restaurant through the app. · It takes only 20 mins to get the food.

Main Use-Cases:

John is a busy full-time staff member at SFSU and he likes to order food through the app at least 4-5 times a week. Whenever he orders the food, he always looks at the review section and how many stars the restaurant gets. When he finds the restaurant which has good reviews, he orders right away without looking at the price. Moreover, he also likes to write the review and gives the rating after he tastes the food. He always looks for the new suggestion whenever he orders the food because he does not want to try the same menu every day.

Diane is a full-time undergraduate student at SFSU. Not only does she go to school all day, but she also works part-time. She orders the food through the app regularly because she does not time to cook. The app provides a premium membership for the people who regularly use the app. Sometimes, the premium member gets coupons for the discount or free meal and sometimes they get the fee delivery.

The app is only for SFSU's students, staff, and faculty. If someone who does not belong to the SFSU or someone who does not have the SFSU g-mail account, they cannot create the account for the app to order the food. Moreover, the app also provides the phone number of the restaurant for the people who are not familiar with using the app or who do not know how to order from the app. After they order by phone, they can go and pick up the food from the restaurant. The app provides a 10% discount only for the SFSU's staff and faculty.

Liam is a vegetarian and he is a undergraduate student at SFSU. He loves to use the app because the app provides the option to search which kind of restaurant that the customers are looking for, for example, vegetarian restaurant, Chinese restaurant, Vietnamese restaurant, or Italian restaurant. The customer also can see the estimated arrival time through the app and they also can track the delivery person's location.

3. List of main data items and entities - data glossary/description

a. List of all entities
  1. items
     a. item
        i. An umbrella category for all items in the restaurant
     b. takeout_order
        i. An order which includes several data fields which pertain to the kitchen order such as a listitem_id, item_price,
     c. receipt
        i. An itemized receipt of all items to be charged - visible to cashier, customer, manager, employee, delivery_employee, sfsu_staff and sfsu_student.
     d. cash_register
        i. assigned to a cashier, the cash_register entity has a many-to-many cardinality with a cashier.
     e. review
        i. written for a restaurant by a customer and viewable by employees and restaurant staff
     f. discount
        i. contains discounts such as percentages
  ii. Types of users
     1. List of all actors
        a. registered_user
           i. a user of the application
        b. customer
           i. Shall be a parent entity to all customers, sfsu staff and students alike.
        c. person
           i. Shall be a parent entity to all users, which will hold information about a person non-specific to any role such as gender, age, address, and other related data fields.
        d. delivery_employee
           i. Holds delivery-critical information such as foreign keys to customer_address and name.
        e. sfsu_staff
           i. Holds information about staff members, for some specific field such as staff_benefit_bonus_amount.

        f. sfsu_student
            i. Holds information unique to sfsu_students.

- b. Data structures
    - i. queue
        1. For use in processing entities of type takeout_order
        2. Handles customers orders' FIFO
        3. First person gets served first, and last gets served last
    - ii. map
        1. For itemized receipts with unique key item_id and all relevant fields

4. Initial list of functional requirements:

1. Customers shall be able to register for the web application service.

   · Valid users shall be verified by checking if the email suffix contains '@sfsu.edu' to ensure exclusive use by SFSU students, staff, and faculty.

   · Customers shall be registered as a student/staff/faculty, restaurant owner, driver, or Admin.

   · Registration shall include name, SFSU email, and delivery address to account for estimation of delivery times and restaurant availability via building or apartment number.

2. Customers shall be able to login and logout to and from the web application service using verified email username credentials.

3. Customers shall be able to deregister their account.

4. Customers shall be able to search for restaurants based on different parameters.

   · Location, cuisine type, menu items, etc.

5. Customers shall be able to leave a rating or review of searchable restaurants.

   · Ratings may include a star rating from 0-5 with an optional comment review section included with the rating or photo attachment.

6.   Customers shall be able to place a food order at a restaurant with a delivery or pickup option.

·   Once the order is placed, the order is automatically paid for (not simulated).

·   Unique features shall include the ability of the customer to leave specific delivery instructions.

7.   Customers shall be able to view and make changes to their order via a cart or list.

8.   Customers shall receive notifications about the status of a placed order.

·   Notifications shall be received through the web application.

9.   Customers shall be able to track the status of their placed order via notifications.

10.   Customers shall be able to cancel a placed order.

11.   A web application-based notification system shall be implemented to send notifications about status of delivery to customers.

-     Notifications shall be sent when: an order is placed, a command has been entered to check the order status, a command has been entered to cancel the order, a restaurant is in the process of preparing the order, a restaurant has completed preparing the order, a delivery driver has been assigned to deliver the order, the delivery driver is on their way to pick up the order, the delivery driver has picked up the order, the delivery driver is enroute to the delivery address, the order has been successfully delivered, the restaurant is closed or no longer taking orders.

12.   Customers shall be able to update their account and contact information.

13.   Restaurants shall be able to register for the web application service.

14.   Restaurants shall be able to login and logout of the web application using email credentials.

15.   Restaurants shall be able to create their profile to be published by the Admin.

·   Create and update their name, description, profile image, and advertisement.

16.   Restaurants shall be able to add or remove menu items.

17.   Restaurants shall be able to add or remove or update menu item descriptions.

18. Restaurants shall be able to add or remove pictures from the menu or advertisement.

· Unique feature shall be attaching photos to menu items or a photo carousel of popular menu items.

19. Restaurants shall be able to manage incoming placed orders.

· This includes marking the completion of preparation of an order, order cancellation, and assigning a delivery driver to the order.

20. Restaurants shall be able to stop taking online orders.

· This feature is available if somehow the restaurant is closed during the specific hours or is no longer in service.

21. Restaurants shall be able to de-list their profile if they do not wish to continue service.

22. The Admin shall be required to review and approve Restaurant profiles to be published that meet the terms of service.

23. The Admin shall be required to delete inappropriate items from Restaurant profiles that violate the terms of service.

24. The Admin shall be able to delete users that have violated the terms of service.

25. The Delivery Driver shall be able to access a SFSU campus map UI within the application for delivery directions.

26. The Delivery Driver shall be able to download and view order details to facilitate Restaurant food delivery to Customers.

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

   a. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0. Application delivery shall be from chosen cloud server
   b. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers

c. All or selected application functions must render well on mobile devices (specifics to be developed in consultation with users e.g. Petkovic)

d. Ordering and delivery of food shall be allowed only for SFSU students, staff and faculty .

e. Data shall be stored in the database on the team's deployment cloud server.

f. No more than 50 concurrent users shall be accessing the application at any time

g. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.

h. The language used shall be English (no localization needed)

i. Application shall be very easy to use and intuitive

j. Application should follow established architecture patterns

k. Application code and its repository shall be easy to inspect and maintain

l. Google analytics shall be used

m. No email clients shall be allowed.

n. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.

o. Site security: basic best practices shall be applied (as covered in the class) for main data items

p. Application shall be media rich (images, maps etc.). Media formats shall be standard as used in the market today

q. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development

r. The application UI (WWW and mobile) shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2021 For Demonstration Only" at the top of the WWW page. (Important so as to not confuse this with a real application).

6. **Competitive analysis:**

- There will be student discounts that other competitors do not usually provide.

- Advance search to pick between vegetarian options or nonvegetarian which makes it faster to order than our other competitors.
- Delivery will be only available for SFSU students, and staff.
- Delivery options will be provided based on their time and location.

Feature	Competitor A (DoorDash)	Competitor B (GrubHub)	Our Future product
Delivery address	+	+	++
Food place search	+	+	+
Shopping cart	+	+	+
Delivery Options	+	+	++
Food Menu	+	+	+
Food option(eg: Veg, or nonveg)	+	-	++

+ feature exists; ++ superior; - does not exist

- There are a lot of restaurants which provide delivery service, but we will only be focusing on delivering to students and the staff of SFSU. To order food, you need to have SFSU addresses and you need to have an SFSU account to order which will make it easier and faster for students and staff. You can search nearby places and pick as usual as other restaurants. The shopping cart will be the same feature as others. There will be plenty of options for delivery for the students and staff based on their schedule. They will have the priority to search between different kinds of food options like vegetarian and nonvegetarian which will make it easier and faster to checkout. Most of these options are not provided by other competitors.

7. **High-level system architecture and technologies used:**
   - **Server Host:** Amazon AWS Linux Ubuntu CPU 1 GB RAM
   - **Operating System:** macOS Big Sur 11.2.1

- **Database:** MySQL Workbench80
- **Web Server:** NGINX 1.19.6
- **Server-Side Language:** Python
- **Additional Technologies:**
  - **Web Framework:** Flask
  - **IDE:** PyCharm

8. **Team and roles:**
- **Team Lead:** Branden Nguyen
- **Front End Lead:** Jasmine Jahan
- **Front End:** Gurjot Singh
- **Back End Lead:** Yongjian Pan
- **Back End:** Tin Thu Zar Aye
- **GitHub Master:** Robert Freeman

9. **Checklist:**
   a. So far all team members are engaged and attending ZOOM sessions when required
      i. **OK**
   b. Team found a time slot to meet outside of the class Back end, Front end leads and GitHub master chosen
      i. **DONE**
   c. 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
      i. **DONE**
   d. Team lead ensured that all team members read the final M1 and agree/understand it before submission
      i. **ON TRACK**
   e. GitHub organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)
      i. **DONE**