

# SW Engineering CSC648/848 Spring 2021

**Project title:** “Campus Cantina”  
**Milestone 1**  
**Team 04**

Member Name	Member Role
Rajdeep Singh	Team Lead (rsingh12@mail.sfsu.edu)
Rinay Kumar	Backend Lead
Bhavani Goruganthu	Frontend Lead / Document Editor
Frederick White	Github Master
German Perez	Frontend Team
Henzon Zambrano	Backend Team

## **Revisions History:**

Draft Number	Date Created/Revised	Date Submitted	Instructor Comments
Draft 1	February 26 2021	March 05 2021	March 06 2021
Draft 2	Mar 08 2021	Mar 14 2021	

## 1) Executive Summary

*“Life is what happens when you are busy making other plans”*

- John Lennon

As digital technology continues to progress, people are migrating towards online platforms for their various needs such as shopping, entertainment as well as ordering a meal for the day. College campuses represent some of the most concentrated markets for certain types of delivery food and many restaurants find that delivering food items to local college campuses can exponentially increase their order volume when college is in session. In these current times, college students become more occupied from school and work, but they all still need to consume food and that takes up too much time as customers have to drive to restaurants, order and wait for the food. The solution created is “Campus Cantina”. The motivation of Campus Cantina is to be able to provide a convenient food delivery service to SFSU college students, staff and faculty within campus.

Through an online food ordering system, SFSU customers (students, staff and faculty) can easily access a restaurant's menu in a hassle-free manner. As lives get busier, more SFSU customers want to order the taste of the restaurant at home and eat within their comfortable confines. New ordering and food delivery options serve students and faculty who increasingly want their meals when and where it's most convenient for them.

“Campus Cantina” is a unique and user-friendly online food ordering system for exclusive use by SFSU students, staff and faculty. This web-based service is easily accessible from your handy devices i.e. Laptop, Tablet & Mobile. All you need to do is to register and login using your sfsu email id and start hunting for your favorite food from the nearby restaurants.

By creating a web-based service exclusive for SFSU students and faculties, we aim to provide a variety of food cuisines that our users will be able to choose from. Our service will be able to help busy college students that have no time to physically go to their desired restaurants that they want to order in. Our users will be able to see different restaurants on our website as well as weekly special deals. Our service will also open opportunities for restaurant owners/vendors to be able to advertise their menu and services. Our system shall be competitive in terms of providing services from on-campus restaurants.

---

“Campus Cantina” shall have many outstanding features, out of which the ability to order from multiple nearby restaurants to get your meal delivered to any specific location at SFSU campus is the dominant one. Additionally, Campus Cantina shall provide a user-friendly UI experience for SFSU students & faculty to allow ease of ordering food online using a detailed map of SFSU campus and surrounding areas. Our team plans to have a search feature where we shall be simplifying browsing of food menus using category filtering.

Our team is not just a group of individuals but is one strong effort put by six innovative minds. We have different strengths in different areas, but we continuously help and motivate each other as a team. Our goal is not only to finish the project we have but also we hope to learn new things and help each other grow.

## 2) Personae and Main Use Cases:

(All Images used in this section are published for free-use)

### Persona for User Category 1: Kim - Student

About:

- Full time student, active in extracurricular clubs on campus
- Very busy schedule
- Doesn't own a car and currently unemployed
- Tech savvy, uses multiple social media apps

Goals and scenario:

- On her lunch break, she checks CampusCantina for quick and affordable food delivery options.
- After making a selection, she waits for her food to get delivered to her dorm room.



*Persona for User Category 2: Usman – Professor*

About:

- Full time professor at SFSU, busy schedule
- Doesn't know how to cook
- Tech savvy

Goals and scenario:

- He enjoys ordering food from CampusCantina because it's fast and super convenient.
- He can easily get lunch delivered during his office lunch hours.

*Persona for User Category 3: John - Delivery Driver*

About:

- Works a second job for CampusCantina on evenings and weekends
- Saving up for a car
- Physically active
- Basic computer skills

Goals and scenario:

- Checks into work by using the CampusCantina app.
- Receives his first pick up location and heads there by bike. Once picked up, he heads to the customer's location.



*Persona for User Category 4: Nicole - Restaurant Owner*

About:

- Manages her own restaurant in West Portal
- Busy schedule
- Restaurant has low sales during the day
- Basic computer skills

Goals and scenario:

- She registers with CampusCantina and awaits approval from the admin to provide restaurant services.
- She promotes weekly lunch specials in order to increase sales.
- She logs onto CampusCantina and sends an update request with new specials

*Persona for User Category 5: Sam – Admin*

About:

- Works for CampusCantina (regular business hours)
- Very detail oriented and efficient
- Review menu changes each day at a fast pace
- Tech savvy

Goals and scenario:

- Approves restaurant owners before going live.
- Reviews updates ordered by oldest to newest
- Verify that all updates abide by CampusCantina guidelines



### **Use Case 1: Ordering Food**

Kim has a 2-hour break from her college classes and feels hungry. She opens up her CampusCantina app to look for restaurants. She's craving tacos, so she uses the search bar to find Mexican restaurants. After making her selection, she browses the menu. The items are divided into categories. In the specials section, she found a Taco Tuesday deal. She chooses how many tacos she wants and then proceeds to the next window. Kim is prompted to login or register with the app before proceeding. She enters her information and creates an account. Before sending her order, she realizes she forgot to order a drink. After adding a coke, she sends her order to the restaurant. She also specifies the on campus location for delivery. The app reminds her to have a cash or card payment on hand for the delivery driver. She waits in her dorm room for her order to get dropped off. After the driver arrives, she meets them downstairs, pays for her food, and walks away feeling happy.

### **Use Case 2: Registering Restaurant Account**

Nicole is registering a new business account with CampusCantina. She first enters her personal information followed by her restaurant's information. CampusCantina requires her to enter menu items along with pricing. She then has the option to post any marketing announcements on her page. After she finishes, she clicks save but before all these updates go live they must be approved by a CampusCantina admin. Nicole appreciates how quick and easy this task is because she has a demanding work schedule.

### **Use Case 3: Approving Restaurant Postings**

Sam arrives to work at CampusCantina and logs onto his admin account. He can see a list of partnering requests from nearby restaurants. He is responsible for approving/rejecting all restaurant postings. He opens his first update request. Once everything is in order, he approves the request. Only then, the restaurant owner is a partner with CampusCantina. At the end of his shift, he logs out from the admin account. Sam understands that he needs to work at a fast pace in his job while still paying attention to detail in all his requests.

---

**Use Case 4: Pick-Up/Drop Off Orders**

John logs into CampusCantina to start working. He receives his first order with the following information: customer's first name, restaurant's address, and the customer location on campus. He gets on his bike and starts pedaling. He arrives and enters the restaurant. Tells them he's picking up Kim's order. The restaurant staff hands him the food. He heads to the customer's location. Once he's on campus, he uses the app's map feature of SFSU. Once reaching Kim's dorm, John hands her her food in exchange for a cash payment. In this whole process, John never contacts the customer directly unless he is experiencing a longer commute time or has issues with the order.

**3) List of Main Data Items and Entities - Data Glossary/Description:**

*(subject to change)*

**1. Registered User**

- Shall contain user full name, id, address, type (student/faculty/staff), phone, email, password.

**2. Restaurants****2a. Restaurant**

- Shall contain restaurant name, id, address, phone, category, tags, logo pics, banner pics, price level.

**2b. Restaurant Owner Profile**

- Shall contain id, contact name, contact phone, contact email, password.

**2c. Restaurant Menu Items**

- Shall contain id, restaurant id, name, description, price, pictures.

**3. Driver**

- Shall contain driver full name, id, address, phone, email, delivery vehicle, password, current location.

**4. Admin**

- Shall contain username, id, password.

**5. Cart**

- Shall contain id, items, quantity, subtotal

**6. Order**

- Shall contain id, restaurant id, restaurant name, restaurant address, user id, user name, delivery location, cart id, cart contents, tip amount, delivery fee, service fee, total cost, delivery ETA.

**7. Message**

- Shall contain id, message content, sender, receiver, date

**8. Search Result**

- Shall contain id, restaurant id, restaurant name, category, cost.

**4) Initial List of Functional Requirements:**Unregistered User:

1. Unregistered Users shall be able to search for local restaurants and their services.
2. Unregistered Users shall be able to view details of a particular restaurant and browse through their menu.
3. Unregistered Users shall have access to a map of the area surrounding campus with restaurants.
4. Unregistered Users shall be able to enter a chat to request help.
5. Local Favorites shall be displayed on the Home page.
6. Unregistered Users shall register and login with SFSU credentials to order food.

Registered User (SFSU Faculty, Students and Staff):

7. Registered Users shall be able to do everything an Unregistered User can do.
8. SFSU faculty, students and staff shall register for seeking the delivery service of 'Campus Cantina' and shall login with their credentials.
9. Registered Users shall be able to add menu items from multiple restaurants to the cart.
10. Registered Users shall be able to manage the items added to the cart.

- 
11. Registered Users shall either request for delivery in the campus or pick up from the restaurant directly.
  12. Registered Users shall be able to update their profile and delivery location details.  
(Delivery in SFSU campus)
  13. Personal favorites shall be displayed on the Home screen.
  14. Registered Users shall have access to order history.

**Restaurant Owner:**

15. Restaurant Owners shall be able to register to advertise their services.
16. Restaurant Owners shall request approval before their restaurants can go live.
17. Restaurant Owners shall be able to add menu items along with photos and prices.
18. Restaurant Owners shall be able to view and manage the meal orders and update the status accordingly.

**Drivers:**

19. Driver shall be able to login and check the orders assigned to him/her.
20. Driver shall be able to see customer order items.
21. Driver shall be able to see the customer drop off location.
22. Driver shall be able to chat with customers.
23. Drivers shall be able to share their location after signing in so that an order can be assigned to him/her.

**Administrator:**

24. Site Administrator shall approve the restaurants before being searchable by the user.
25. Site Administrator shall be able to delete or remove inappropriate users, drivers, owners.

## 5) List of Non-Functional Requirements:

1. 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
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
3. All or selected application functions must render well on mobile devices (specifcics to be developed in consultation with users e.g. Petkovic)
4. Ordering and delivery of food shall be allowed only for SFSU students, staff and faculty
5. Data shall be stored in the database on the team's deployment cloud server.
6. No more than 50 concurrent users shall be accessing the application at any time
7. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
8. The language used shall be English (no localization needed)
9. Application shall be very easy to use and intuitive
10. Application should follow established architecture patterns
11. Application code and its repository shall be easy to inspect and maintain
12. Google analytics shall be used
13. No email clients shall be allowed.
14. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
15. Site security: basic best practices shall be applied (as covered in the class) for main data items
16. Application shall be media rich (images, maps etc.). Media formats shall be standard as used in the market today
17. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
18. 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:

Features	DoorDash	GrubHub	Postmates	Uber Eats	Campus Cantina
Category Filter	+	+	-	+	+
Cart	+	+	+	+	+
Restaurant Browsing	+	-	+	+	+
Restaurant Map	+	+	+	+	++
Text Searching	+	+	+	+	+
Menu Binary Searching	+	-	+	+	+
On Campus Restaurants	-	-	-	-	+

There are a couple key features that we intend to include that some of our competitors do not have, and some features we plan to do better than our competitors. A map of the restaurants in the surrounding area and the inclusion of on campus restaurants are two features that are minimal or not present in the competition. The UI experience for restaurant browsing is done in a similar way across much of the competitors, except for some and we intend to stick to what would be expected while providing our own experience. We will include menu binary searching, so users can search a menu by different categories, ie drinks.

GrubHub is one competitor that tried to go too far outside the expected and the result is a poor UI experience. Category filtering is another feature offered by all of the competition except Postmates and as such we intend to include it as well. Postmates is another competitor that exclusively does not have a category search which we intend to include.

## 7) High-Level System Architecture and Technologies Used:

Technology Type	Technology	Description
<b>Server Host</b>	AWS Elastic Cloud Computing (EC2)	EC2 - t2.micro - 1 vCPU - 1 GiB RAM
<b>Operating System</b>	Ubuntu Server	Version 18.04
<b>Database</b>	MySQL	Version 8.0.x
<b>Web Server</b>	NGINX	Version 1.18.0
<b>Server Side Language</b>	JavaScript	NodeJS
<b>Web Framework</b>	ExpressJS	Version 4.17.x
<b>Frontend UI Components</b>	ReactJS	Version 17.0.x
<b>Frontend UI Designing</b>	Bootstrap	Version 4
<b>IDE</b>	Visual Studio Code	Version 1.53.2
<b>Supported Browsers</b>	Google Chrome Mozilla Firefox	Most Stable Version

## 8) Team and Roles:

Member Name	Member Role
Rajdeep Singh	Team Lead (rsingh12@mail.sfsu.edu)
Rinay Kumar	Backend Lead
Bhavani Goruganthu	Frontend Lead / Document Editor
Frederick White	Github Master
German Perez	Frontend Member
Henzon Zambrano	Backend Member

## 9) Checklist:

- So far all team members are engaged and attending ZOOM 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**