SW Engineering CSC648/848 Spring 2021

Hermes

Team 03

Milestone 2

March 19, 2021

Roberto Herman - Team Lead (email: rhermanvaldivia@mail.sfsu.edu)
Alex Pena - Backend (email: arpenacoding@gmail.com)
Amit Joshi - GitHub Master (email: ajoshi2@mail.sfsu.edu)
Angela Avina Garcia - Front End
Jacob Sebastian - Backend Lead (email: jsebastian@mail.sfsu.edu)
Jarvis Yu - Front End Lead

History Table

Date Submitted	**Date Revised**
03/19/2021	

Table of contents

1. Executive Summary

As a student, did you ever sit in class trying to listen to what your teacher is trying to say, but can't pay attention to her because you're too distracted by the growls in your stomach? Have you ever forgotten your lunch at home? Did you ever grow tired of the food choices at your campus? With Hermes, those days are over. Hermes is here to provide SFSU students, staff, and faculty with plenty of food options, right at their fingertips, with fast delivery times, straight to their doorstep! Hungry on a dark, rainy night? No problem! Students can stay in their dorm, order food, and finish their homework while their food arrives. No more long, hasty walks to Stonestown Mall's packed food court. No more being late to class because of the long waiting lines at Cafe Rosso. No more dangerous late night MUNI rides just to get some deliciously warm dumplings.

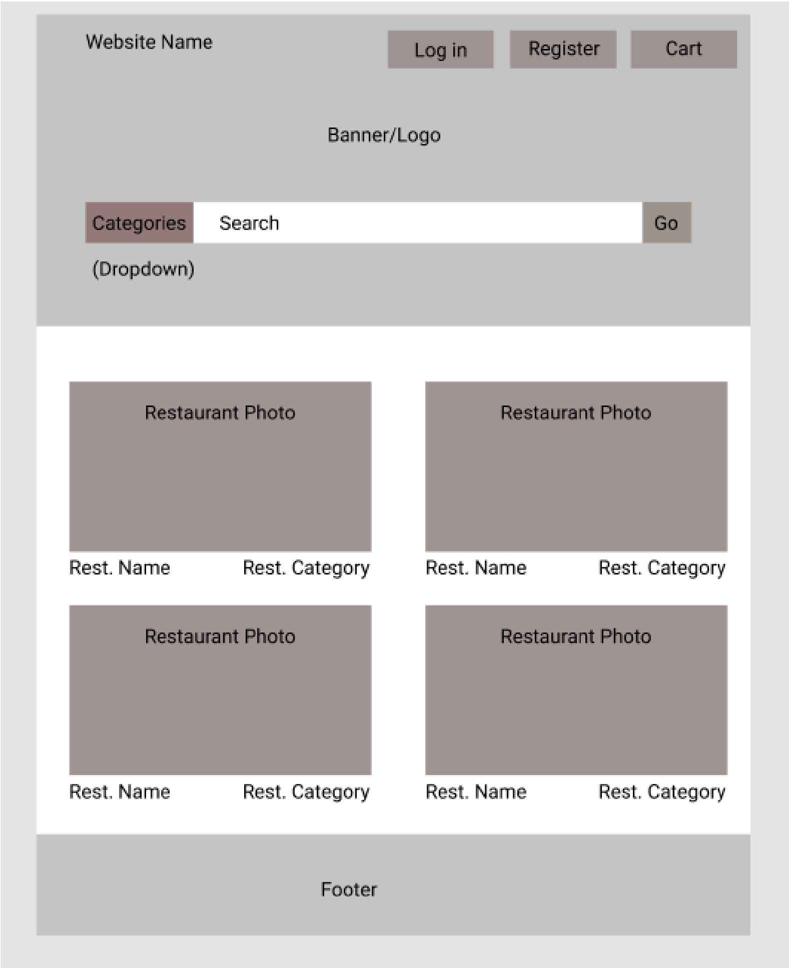
Online food delivery is a billion-dollar industry and there's no doubt SFSU students, staff, and faculty are already using apps such as Door Dash and Uber Eats to get their food delivered. However, none of the popular food delivery apps cater to the needs of SFSU students and faculty like we do. We are here to make their customer experience as enjoyable and convenient as possible. Our goal is to deliver food to any location you can think of within SFSU campus grounds at lightning fast speeds. Because all deliveries made through our app are delivered to campus, delivery drivers can make less round trips, decreasing delivery times and lowering operation costs for restaurants and drivers. Restaurants go through a straight forward sign up process where they're guided step by step through the registration and menu creation process. There's no better way to gain loyal and regular customers than being the go-to restaurant for the campus. A restaurant owner can obtain immediate exposure to the entire student body and faculty staff, and the drivers are equipped with a built-in campus map so they never get lost. It's as easy as that! Once restaurants are signed up, customers can joyfully browse through any nearby restaurant and place their orders.

Hello there, we are a team of six dedicated computer science students with dreams of making campus life more enjoyable for everyone. We hope to learn as much as we can about e-commerce in general, as well as the ever-growing online food delivery industry.

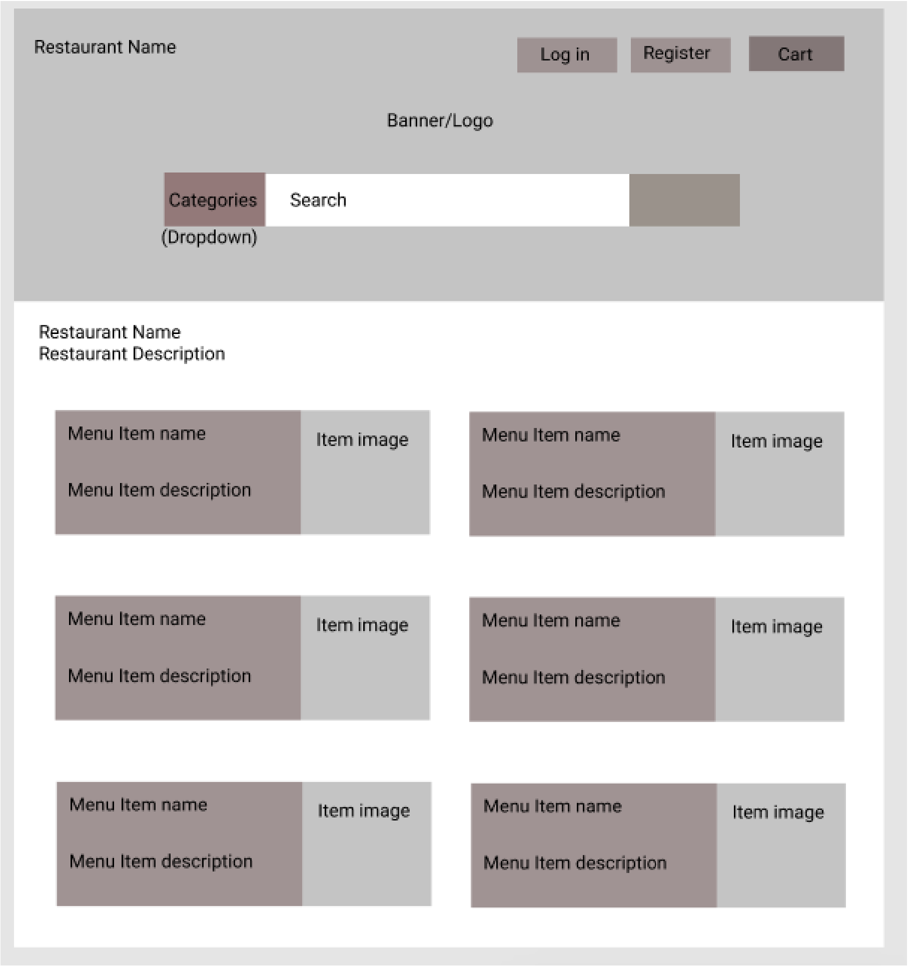
2. Personae and main Use Cases w/ Storyboard

Student : Tony - a full time college student	**Use Case 1:** Hungry, busy student order food at night to their dorm.
 Pain points - Does not drive - Does not like to walk around at night - Is always in a rush - Is on a small budget - Is worried about covid safety Goals - Eat healthy - Succeed in school Skills - Continuously using the phone - Quick at adapting to new technologies - Able to navigate applications and websites	Tony is a hungry student with a craving for pizza. Unfortunately it's dark outside, he doesn't have a car, and he's scared to walk to a restaurant all by himself. Instead, he decides to use our website app to get his food delivered straight to his dormitory's well-lit entrance. He goes on our website app and browses through all the nearby pizza places, checking out their menu items as well as their price. After deciding on his order, he places it, gets an order confirmation, and now waits for the driver to get and deliver his food. Tony realizes that he is even more happy that he decided to get the food delivered because now he can get back to finishing his homework while he waits! When the delivery driver arrives, he gets a notification that they are waiting at the entrance of the dorm. Tony is happy that he was able to find food at this time of night, and that he didn't have to walk out in the dark, cold, foggy San Francisco streets.

Tony goes to the website to order food because he knows that the restaurants deliver to campus. Tony goes to the Home Page where he can search and browse the restaurants.

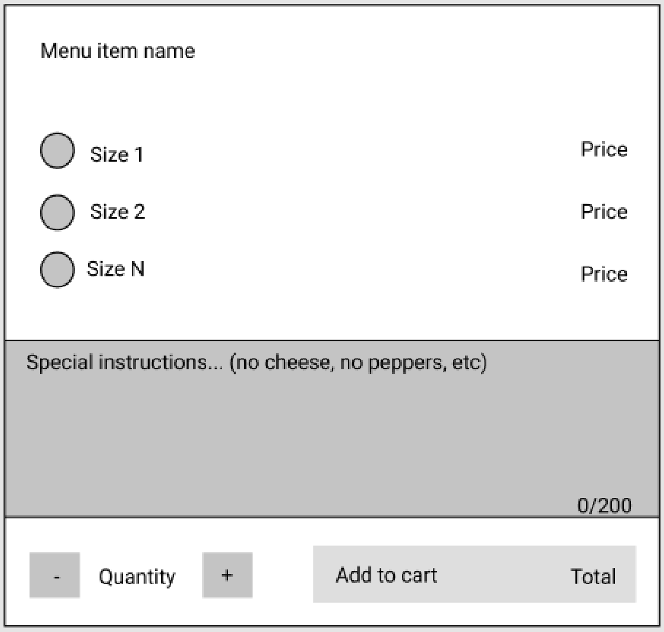


He finds a restaurant he wants to order from and browses their menu.



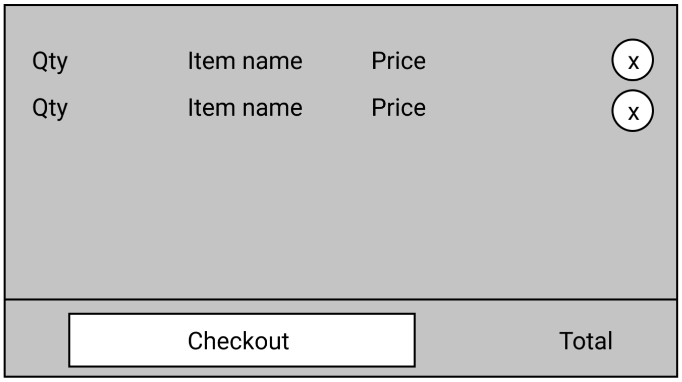
Tony then clicks on a menu item that he is interested in. Where he can see the description of the item, the size, and can select the quantity that he wants for that item and add it to his cart.

Detailed Menu Item Page (Modal). Pops up after clicking on a menu item.



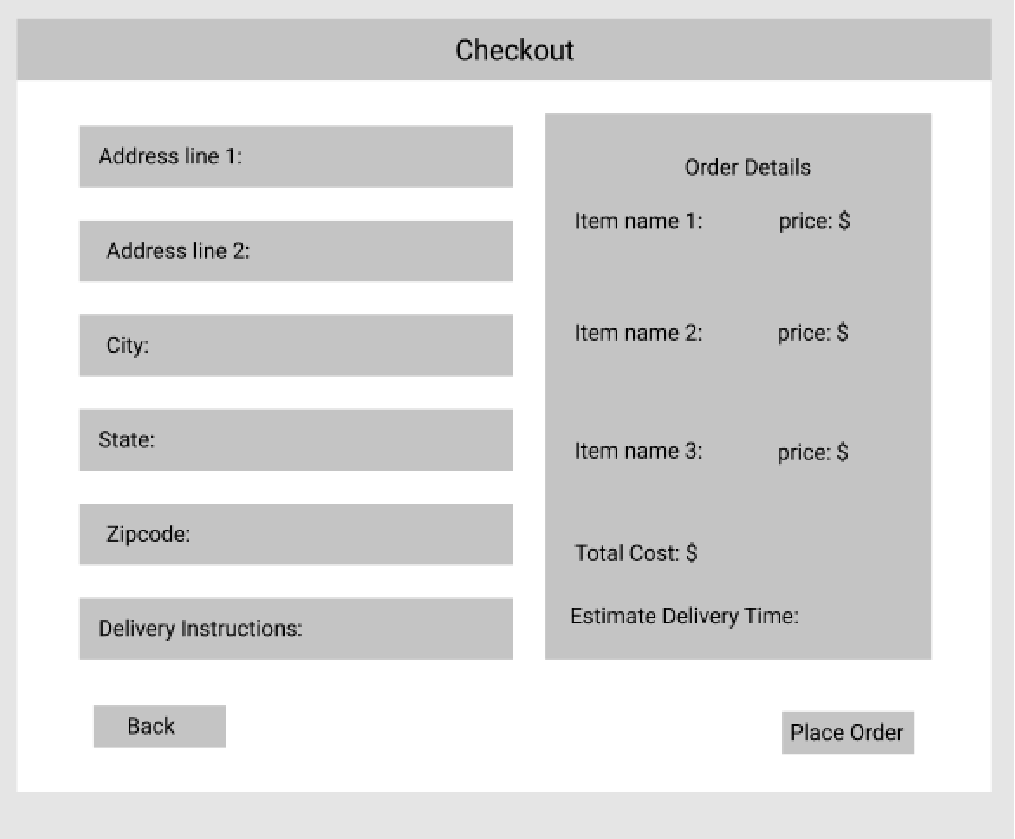
Tony places his order and when he is ready to checkout, he clicks on the cart button at the top of the navigation bar, where he can review his cart items and proceed to checkout.

Shopping Cart modal/dropdown pops up after clicking “Cart” button

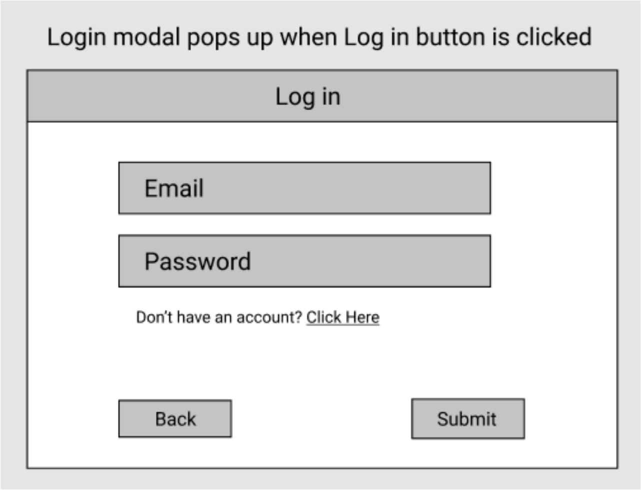


When user clicks checkout and they’re not logged in, the login modal will show up.

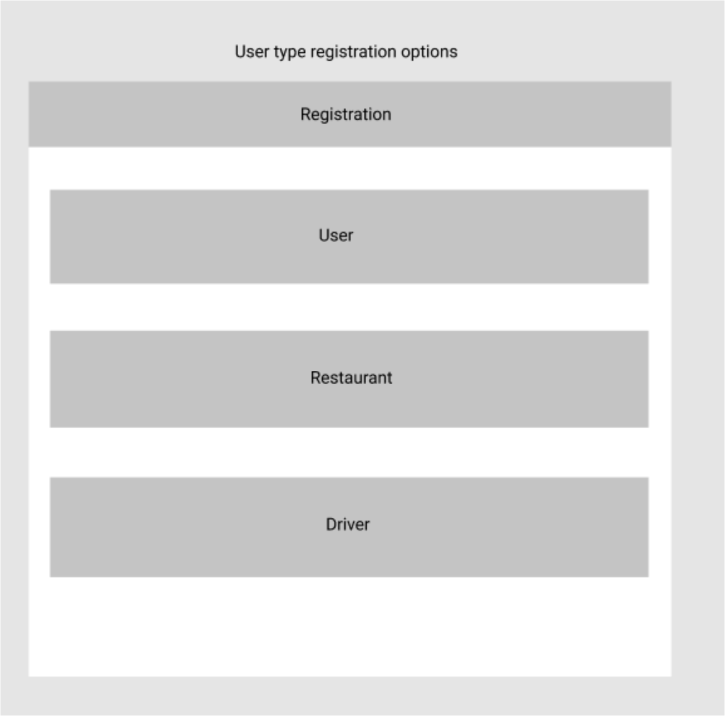
Tony fills out the checkout form and reviews his order. Once he is comfortable with the order Tony clicks on the "Place Order" icon where he expects to get a confirmation notice.



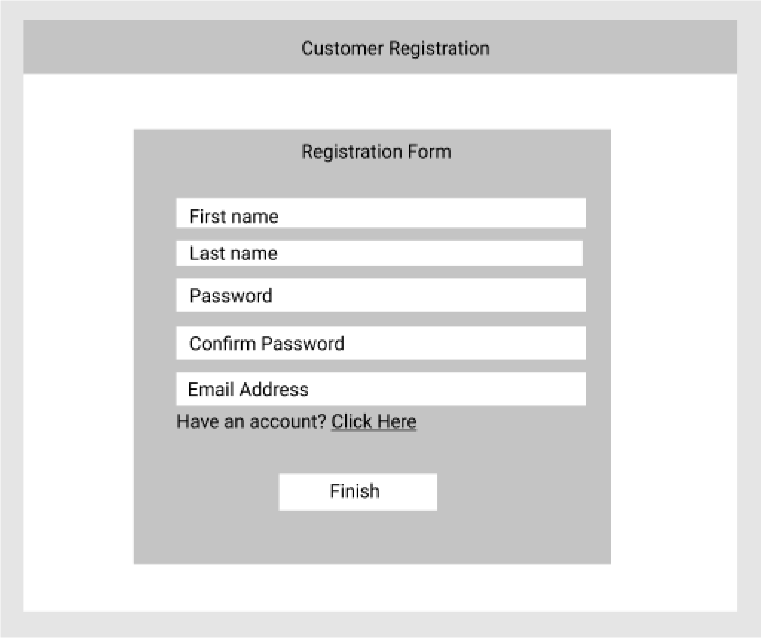
Tony presses place order, but a pop up screen shows up where Tony is supposed to be logged in to place orders. Since Tony does not have an account yet, he clicks on the link to register.


Login modal pops up when Log in button is clicked

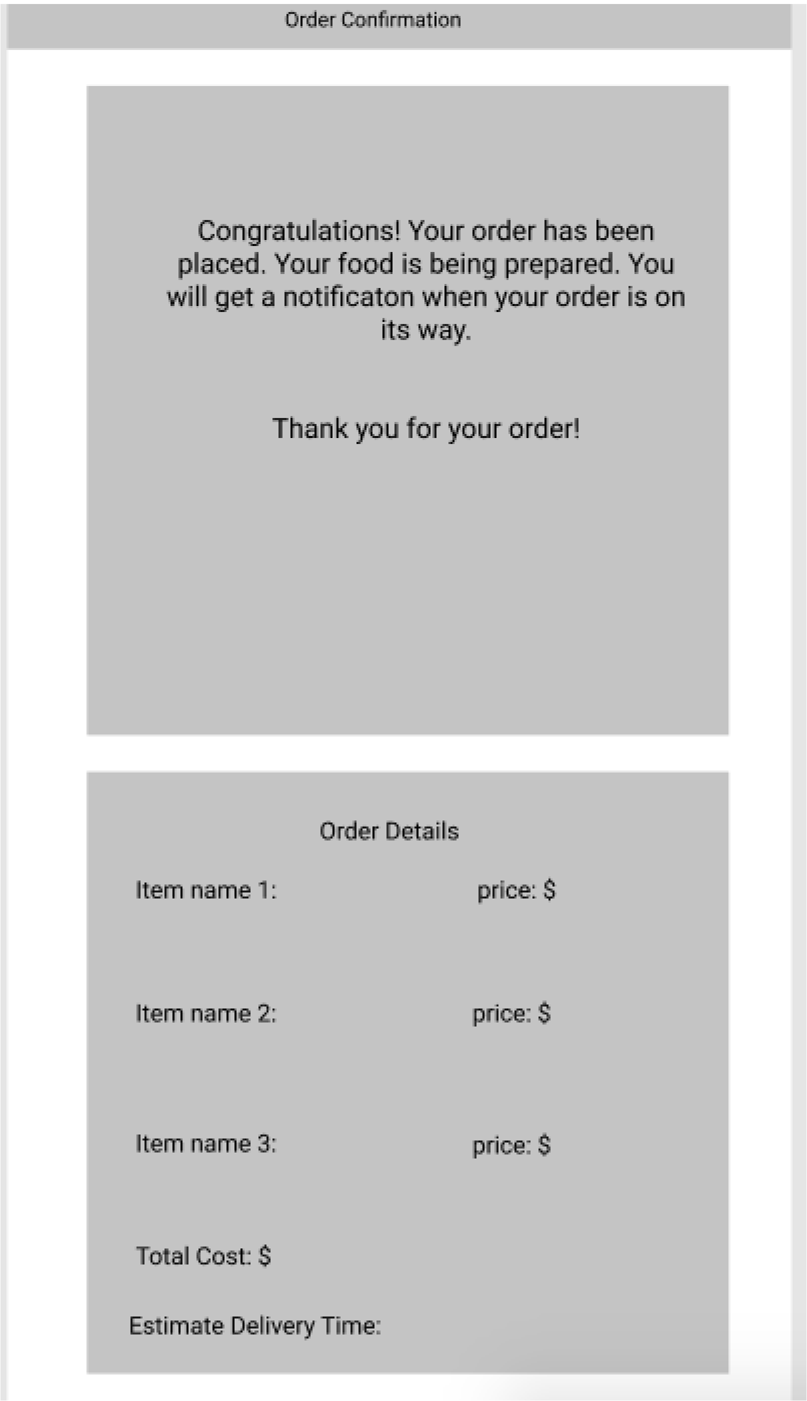
He is redirected to the Registration page and clicks on User Registration link.



He then puts in all his information and creates his account.



Tony then is able to place an order and receives a confirmation and is given an estimated time that the delivery would be made. Now Tony waits for his food to be delivered at the SFSU campus. As Tony waits, he receives notifications about the order's status.



Order Confirmation

Congratulations! Your order has been placed. Your food is being prepared. You will get a notificaton when your order is on its way.

Thank you for your order!

Order Details

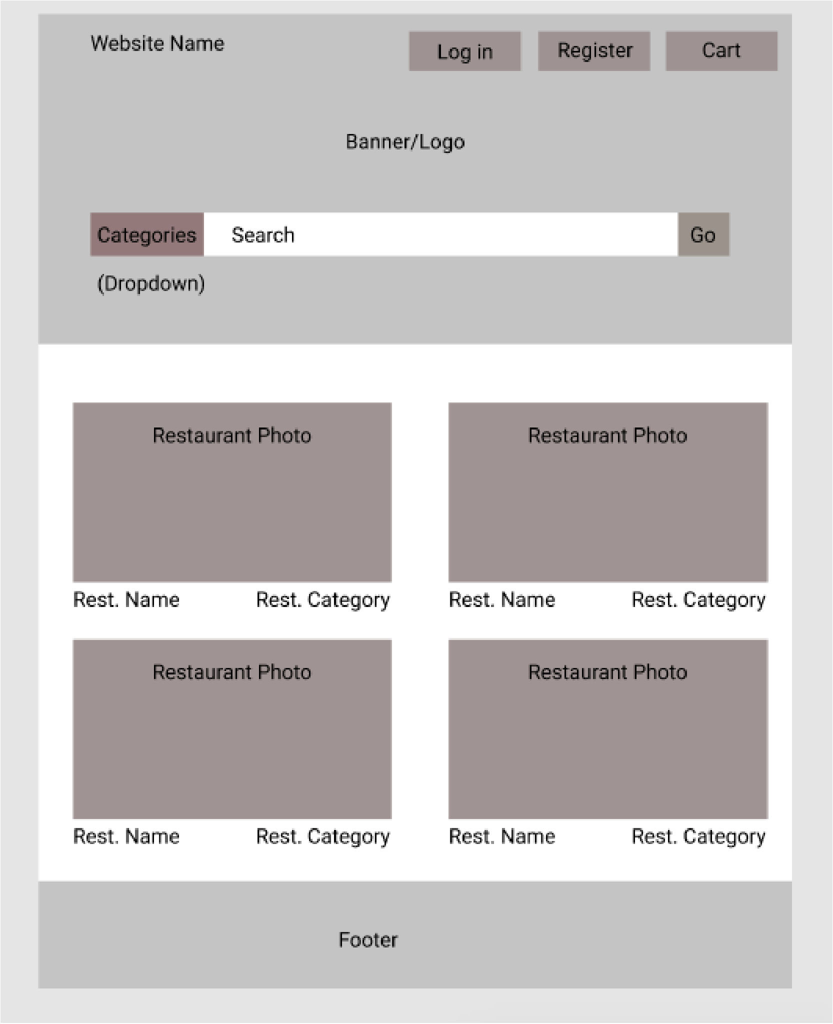
Item name 1:	price: $
Item name 2:	price: $
Item name 3:	price: $

Total Cost: $

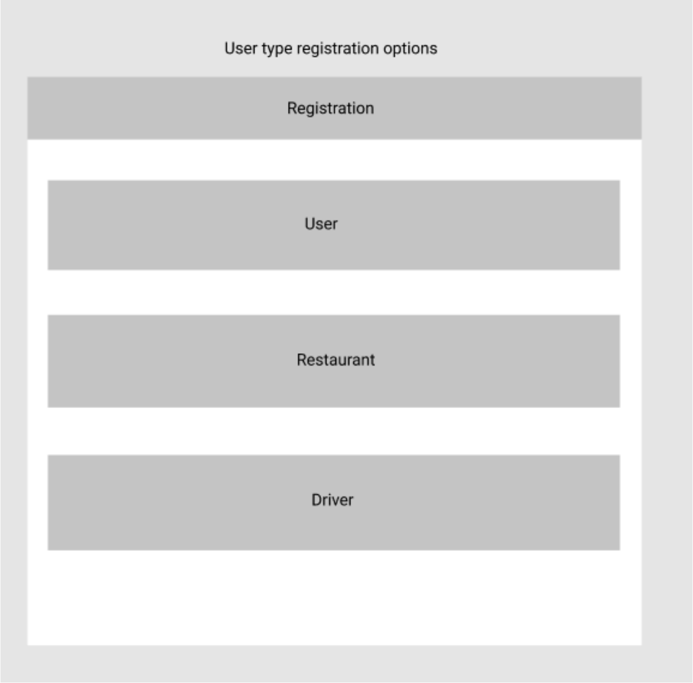
Estimate Delivery Time:

Delivery Person: Marco - main provider for family looking for flexible hours	**Use case 2:** Driver navigates campus and makes deliveries
 Pain point: ● Not knowing the city (where to deliver) ● Not knowing the campus Skills: ● Driving (license) ● Navigation ● Customer service ● Able to use new technology Goals: ● Earn money ● Get to know the city better ● Learn customer service ● Get food discounts	Marco is a delivery driver for a restaurant that is already registered with our website app. After going through the signup process, he is now able to see incoming customer orders to the restaurant. Since the drop off locations are always to the SFSU campus, Marco can take multiple orders with him and deliver them all once he gets to campus instead of having to go back and forth. A map will let Marco know the directions to the drop off location, which is always some place around the SFSU campus. He's never been to the SFSU campus so he's also glad that there's a campus map available for reference as the navigation map doesn't show building numbers. When he gets there, he goes to the building where the delivery needs to be made, the customer comes, pays him, and Marco can now deliver more orders until he runs out and has to go back to the restaurant.

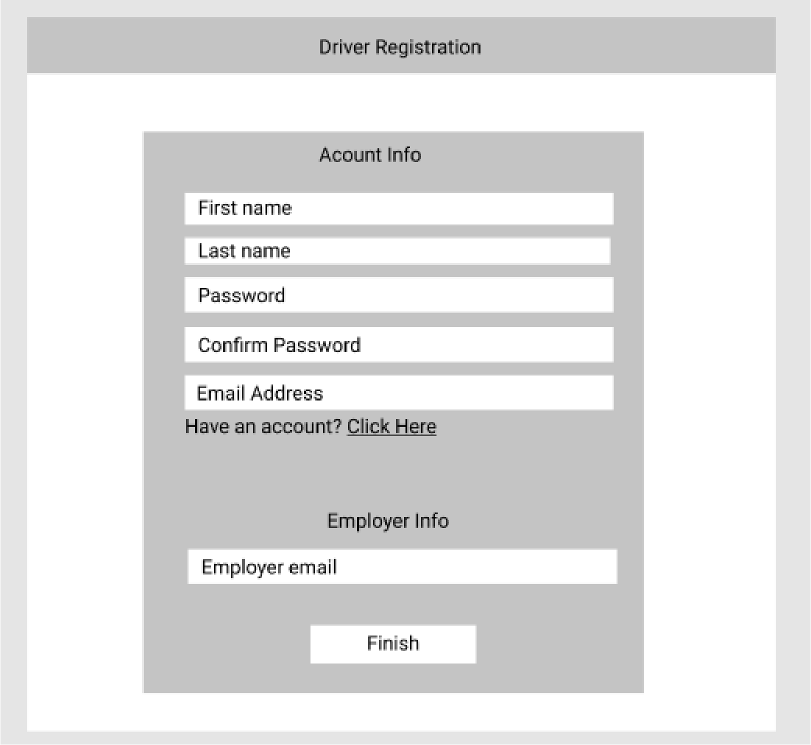
Marco is an employee of a restaurant that is registered on our website. He goes to our website to register as a driver for the restaurant that he works at. Marco goes to the HomePage and selects the “Register” icon.



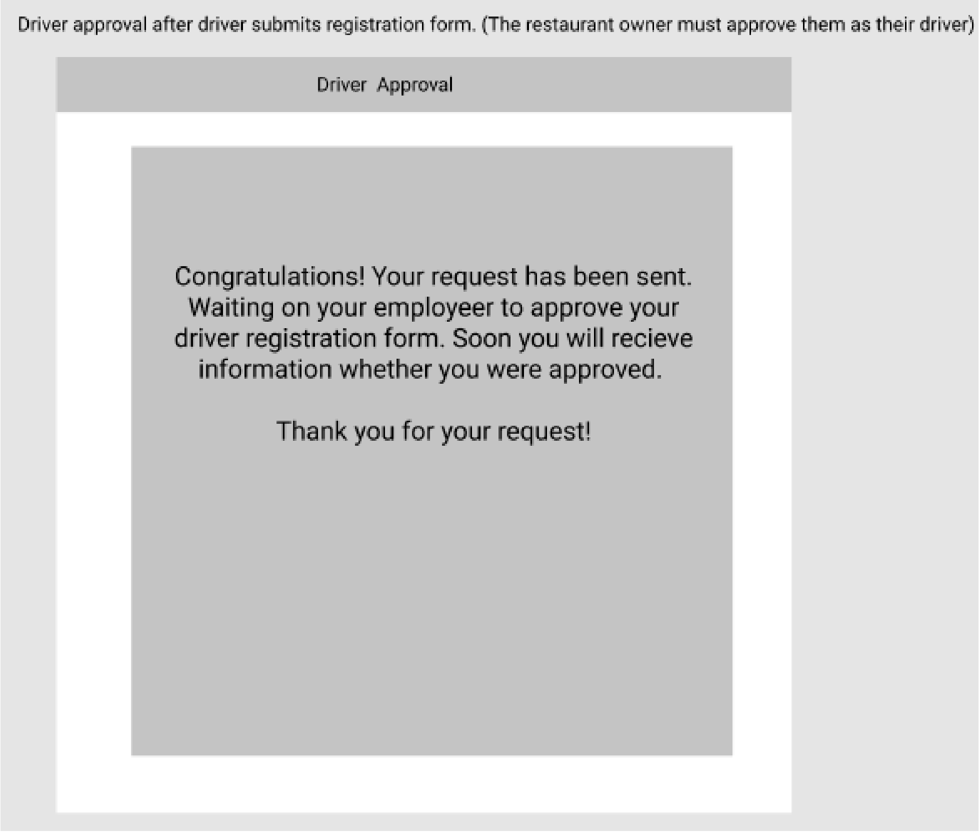
He is then redirected to the Registration page, where he selects Driver Registration to register as a driver for the website.



Marco then gets redirected to the Driver Registration page where he puts all his information to register as a driver.



Marco receives a driver approval form that informs him that his employer needs to approve him.



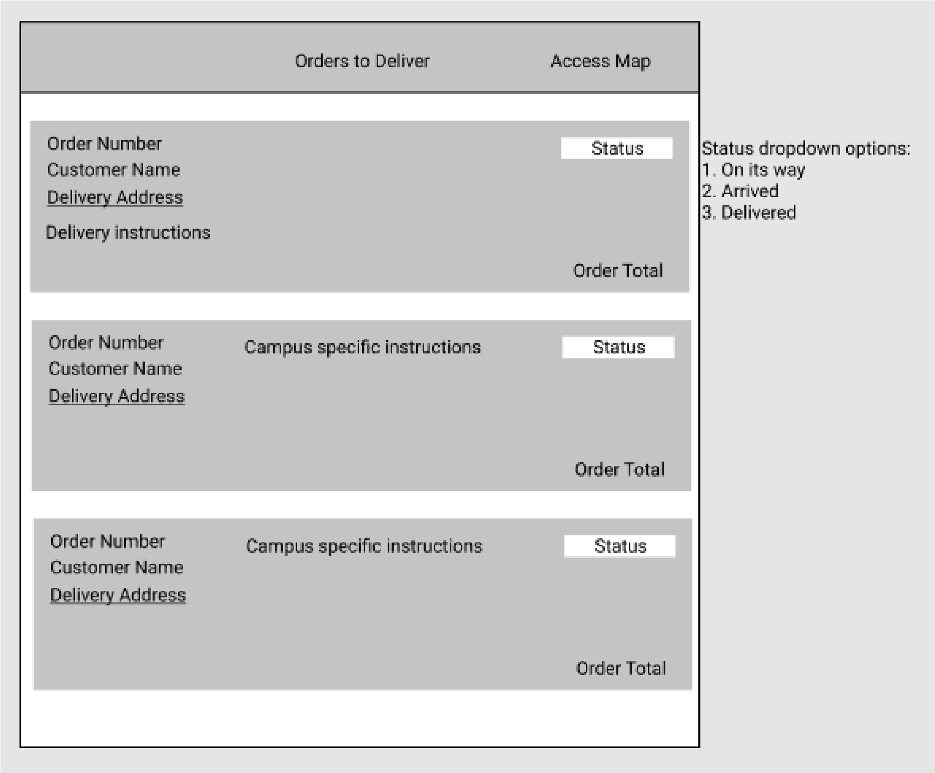
Driver approval after driver submits registration form. (The restaurant owner must approve them as their driver)

Driver Approval

Congratulations! Your request has been sent. Waiting on your employeer to approve your driver registration form. Soon you will recieve information whether you were approved.

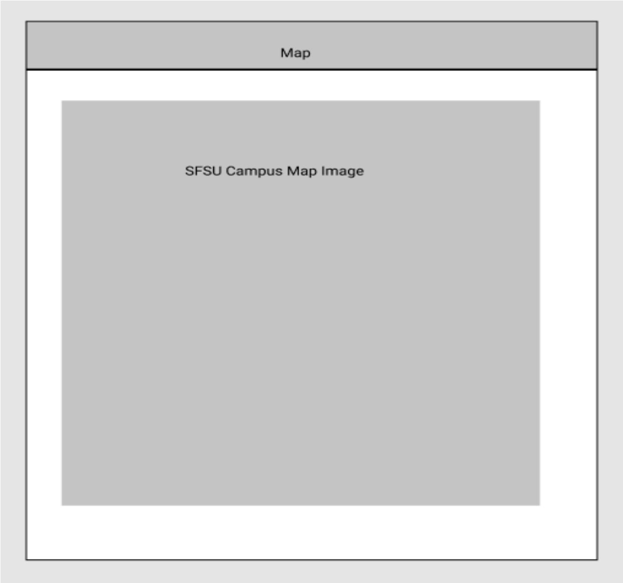
Thank you for your request!

Now that Marco is a registered driver, he now sees a Driver page where he can look at the orders placed and delivers them. He is not quite sure where to go thankfully the website provides a map that will help Marco find the delivery destination.

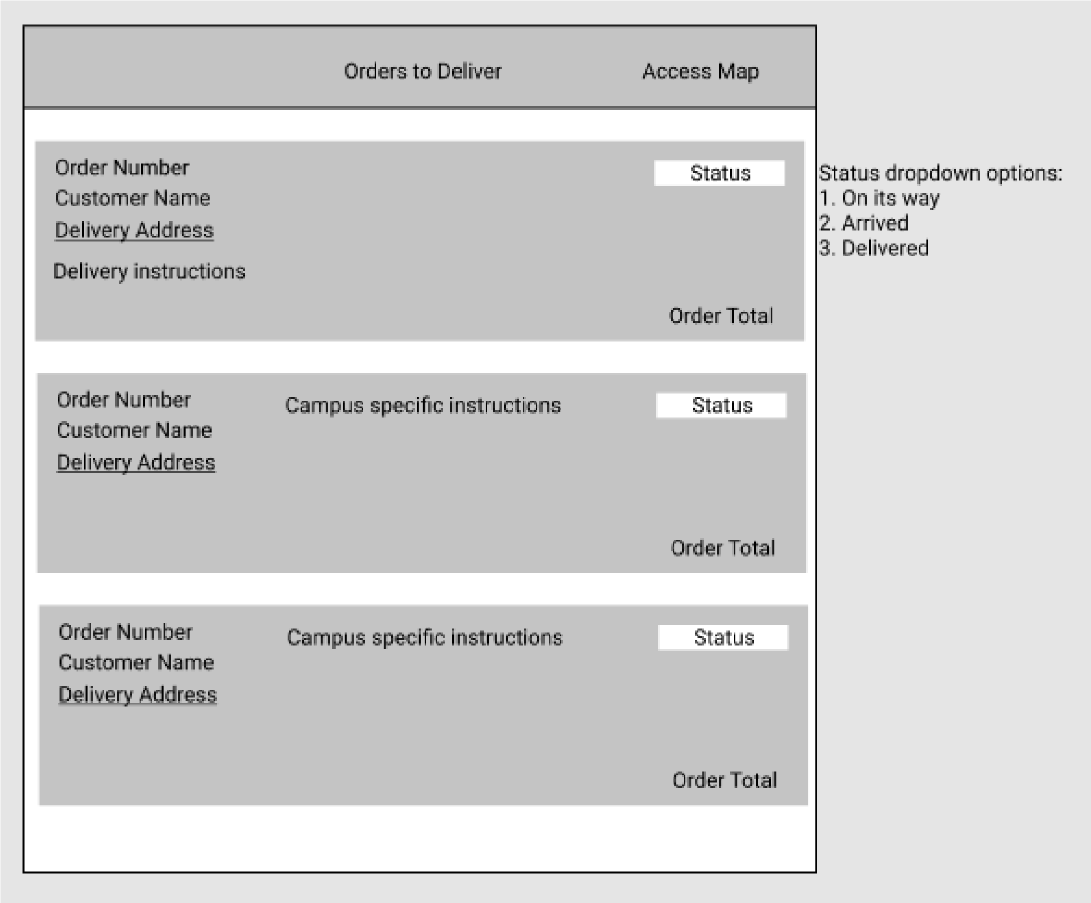


Status dropdown options:
1. On its way
2. Arrived
3. Delivered

Now that Marco is a registered driver, he now sees a Driver page where he can look at the orders placed and delivers them. He is not quite sure where to go thankfully the website provides a map that will help Marco find the delivery destination.



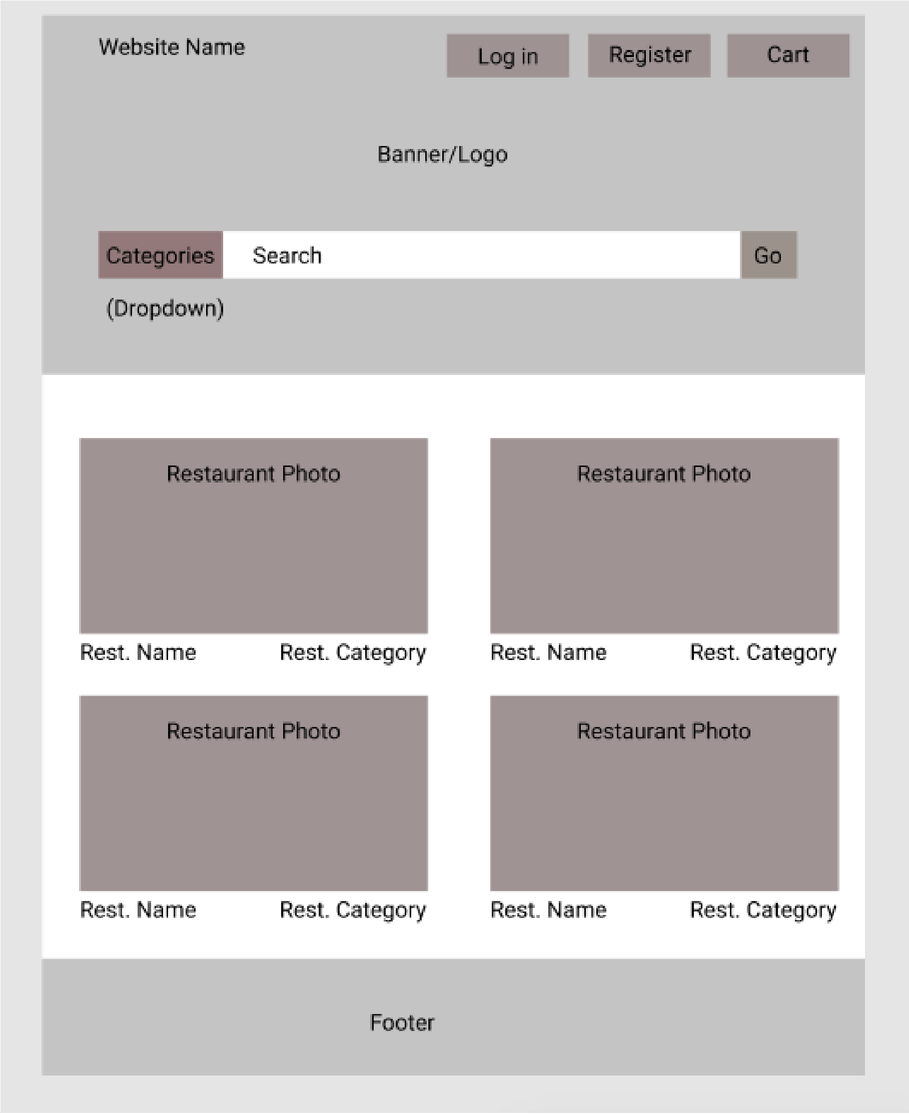
When Marco gets to the customer’s address, he clicks on the status dropdown and selects “arrived”. After customer comes to receive and pay for the order, he can then switch it to “delivered”!



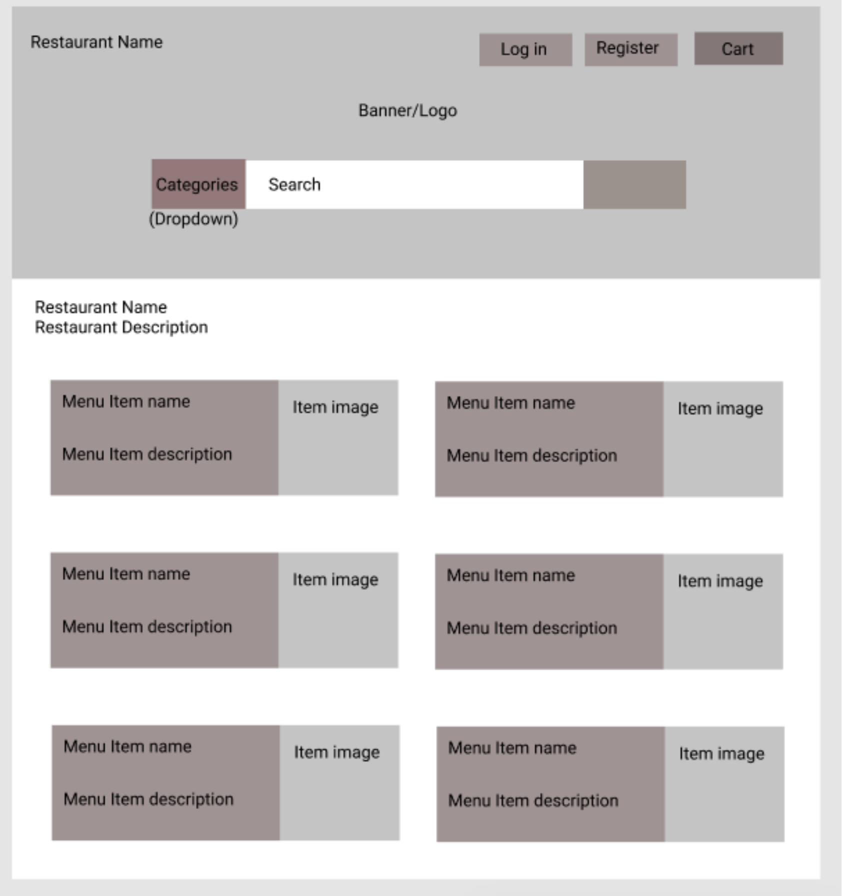
Status dropdown options:
1. On its way
2. Arrived
3. Delivered

School Staff: Danielle - is a full time professor	**Use Case 3:** Professor orders food to meeting room
 Pain point: ● Safety ○ No-contact delivery ● Convenience ○ Time ○ Distance ○ Deliver to classroom (during meetings, etc) ● Budget ○ On a small salary Skills: ● Teaching ● Able to use web applications ● Strong understanding of new technology ● Usually has a computer/ or laptop nearby Goals: ● Help students	Danielle is a full time professor who does not seem to have the time to grab a bite to eat. Danielle has spent the entire morning teaching and helping out her students she forgot to prepare for her staff meeting. She knows that her meeting is at noon and needs to provide lunch for her fellow colleagues. She needs something quick and appetizing to order for herself and colleagues. She goes to our website app and searches for restaurants that can deliver to the campus building where the meeting is being held. She selects the restaurant and chooses what she wants to order for her and her colleagues. She is excited to use the website app and shows her colleagues how easy it is to use it. She gets a notification with the order confirmation as well as the information about the delivery driver. When the delivery driver is in the building, she pays the driver, and gets her food. She is strongly worried about covid safety and was content to see that the delivery driver was respecting her safety. With this positive experience she recommends the website app to her students and colleagues and decides to use it frequently since she is always in a rush.

Danielle is a professor who is in a rush and needs to order food for herself and colleagues. Danielle goes to our website and looks at the featured restaurants at the bottom. Danielle chooses the categories button to browse restaurant types. Danielle selects the category and selects the restaurant that she thinks would be more appealing to her colleagues.

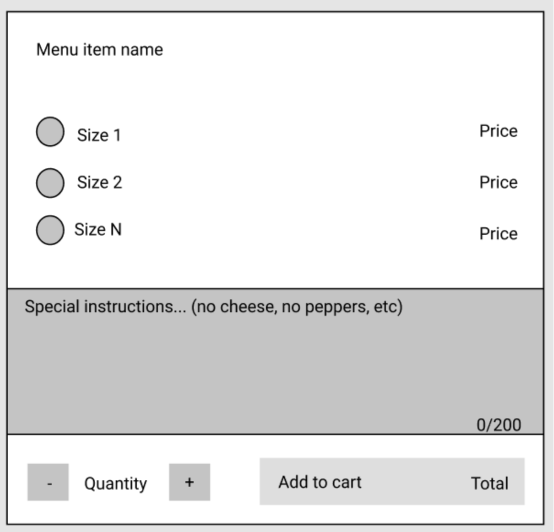


Danielle is able to browse through the restaurant's menu and is able to select the item that she wants to order.



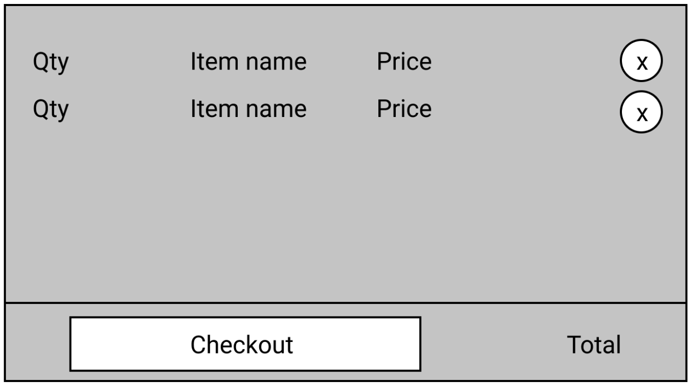
The item is clicked and a modal pops up where Danielle can view the item that she wants to order and gets a brief description of the item. Then she is able to order the items for herself and colleagues.

Detailed Menu Item Page (Modal). Pops up after clicking on a menu item.



Once Danielle is ready to place her order, she clicks on the Cart button and a pop up comes up and she presses Checkout.

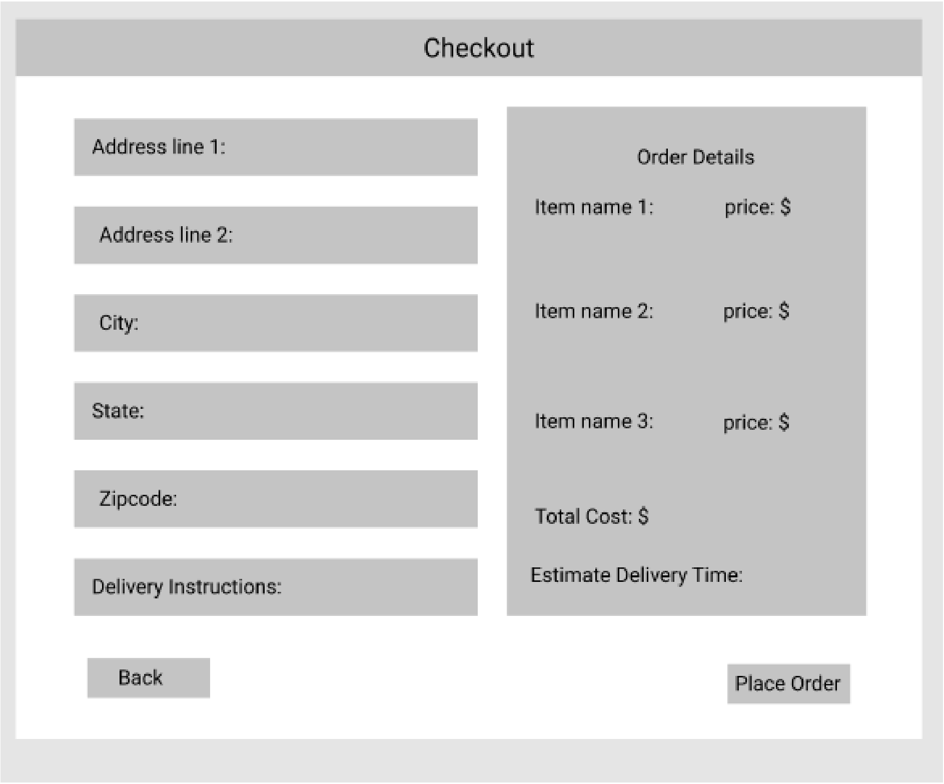
Shopping Cart modal/dropdown pops up after clicking “Cart” button



Qty	Item name	Price	x
Qty	Item name	Price	x

When user clicks checkout and they’re not logged in, the login modal will show up.

Danielle fills out the checkout form and reviews her order. Once she is comfortable with the order Danielle clicks on the “Place Order” icon where she expects to get a confirmation notice.



Checkout

Address line 1:

Address line 2:

City:

State:

Zipcode:

Delivery Instructions:

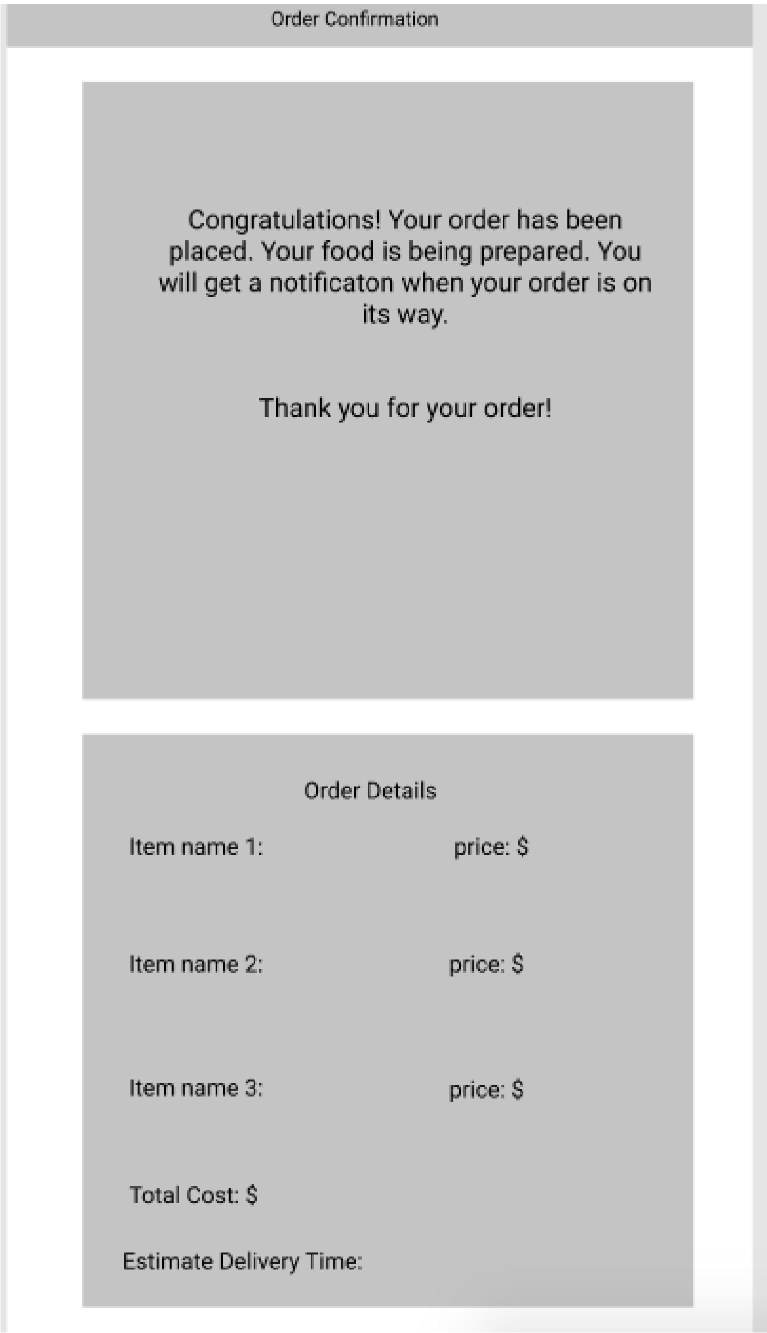
Order Details

Item name 1:	price: $
Item name 2:	price: $
Item name 3:	price: $

Total Cost: $

Estimate Delivery Time:

Danielle receives her confirmation order and is given an estimated time and the name of the delivery driver for her order. Now Danielle waits for her food to be delivered at the SFSU campus.



Order Confirmation

Congratulations! Your order has been placed. Your food is being prepared. You will get a notificaton when your order is on its way.

Thank you for your order!

Order Details

Item name 1: price: $

Item name 2: price: $

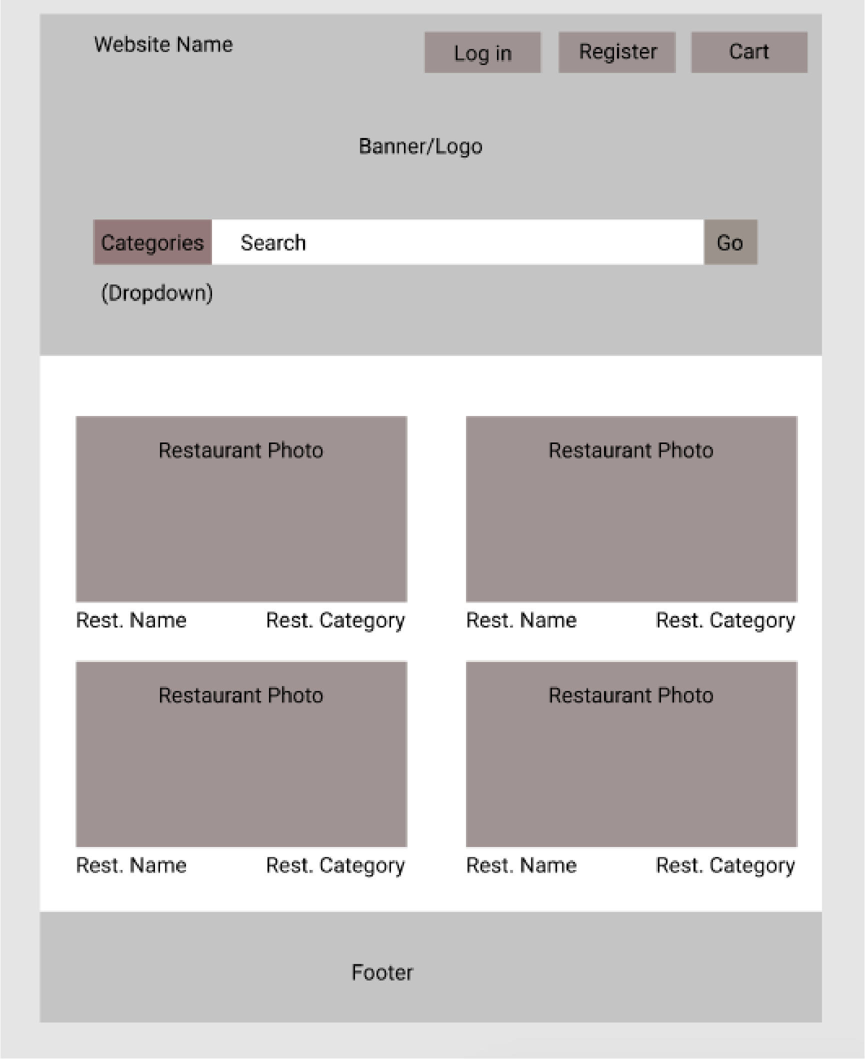
Item name 3: price: $

Total Cost: $

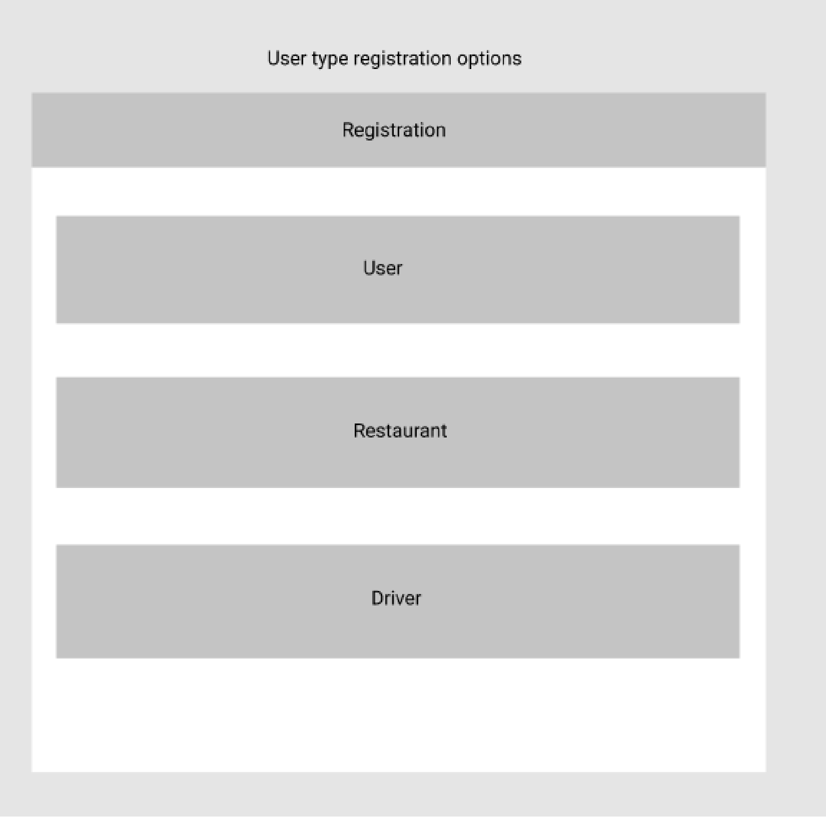
Estimate Delivery Time:

Restaurant Owner: Miguel- brand new to the restaurant industry looking for ways to expand business	**Use Case 4:** Restaurant owner goes through entire registration and advertising process
 Pain point: ● Advertising ● Customer service ● Promotions/deals ● Competition ● Online presence ● Efficiency ● Reviews/Ratings ● Delivery service ● Cost effective ● Popularity with students Skills: ● Providing great food ● Quick service ● Not very technologically diverse ● Has a difficult time using new technology ● Knows how to use a website Goals: ● Expand his business and have more clientele ● Top rate and become popular	Miguel is a brand new restaurant owner who is close to the University. He notices that many of the college students are quickly ordering meals. Miguel started to hear the students talk about a website app where they can order food and have it delivered to campus. He wants to have more exposure to his restaurant and make it approachable for the college students. Miguel decides to register his restaurant on the website app. He is not very good with using applications, but is in great need to bring more business to his restaurant. He goes to our website app and starts to see that multiple small restaurants are being listed and are providing reasonable prices for their food. Thinking about how popular his restaurant has become within the college students he wants to make sure that it is approachable for them. Without much hesitation Miguel registers to have his restaurant listed on the application and is guided with the next steps on how to enroll his restaurant, and add new menu items. When a driver registers on the app, they add the email or the restaurant owner they are going to work for. After the driver registration, the restaurant owner gets a notification asking them to approve or decline the driver. After the restaurant menu gets approved by an administrator, he is excited to see he’s starting to receive orders and his driver is ready to fulfill those deliveries.

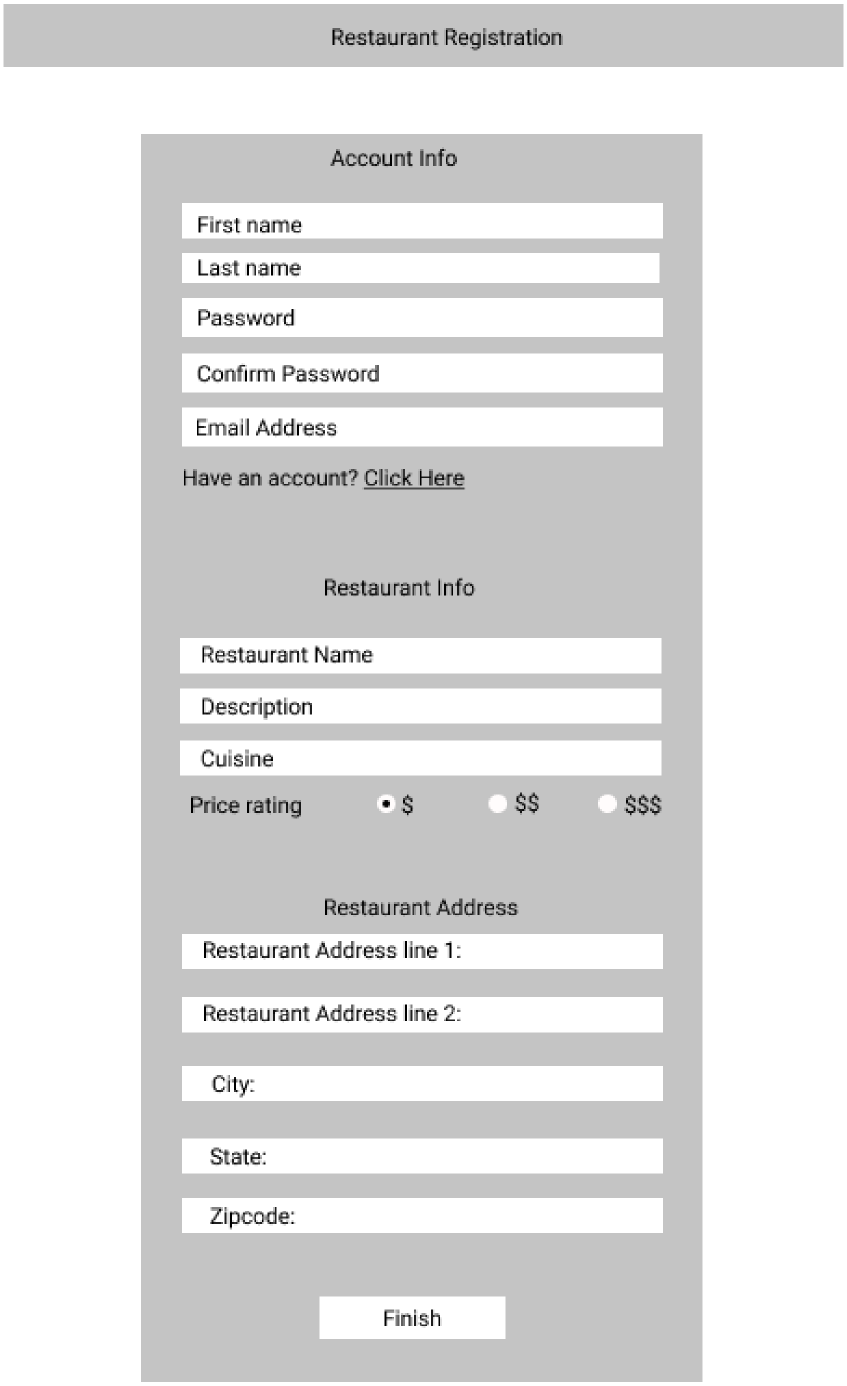
Miguel wants more exposure to his restaurant and make it approachable for the college students. Miguel decides to register his restaurant on the website app. He clicks the "Register" button.



He is then redirected to the registration options, where he selects Restaurant Registration.



He then is redirected to the restaurant registration form where he registers his restaurant.



Restaurant Registration

Account Info

First name

Last name

Password

Confirm Password

Email Address

Have an account? Click Here

Restaurant Info

Restaurant Name

Description

Cuisine

Price rating ● $ ○ $$ ○ $$$

Restaurant Address

Restaurant Address line 1:

Restaurant Address line 2:

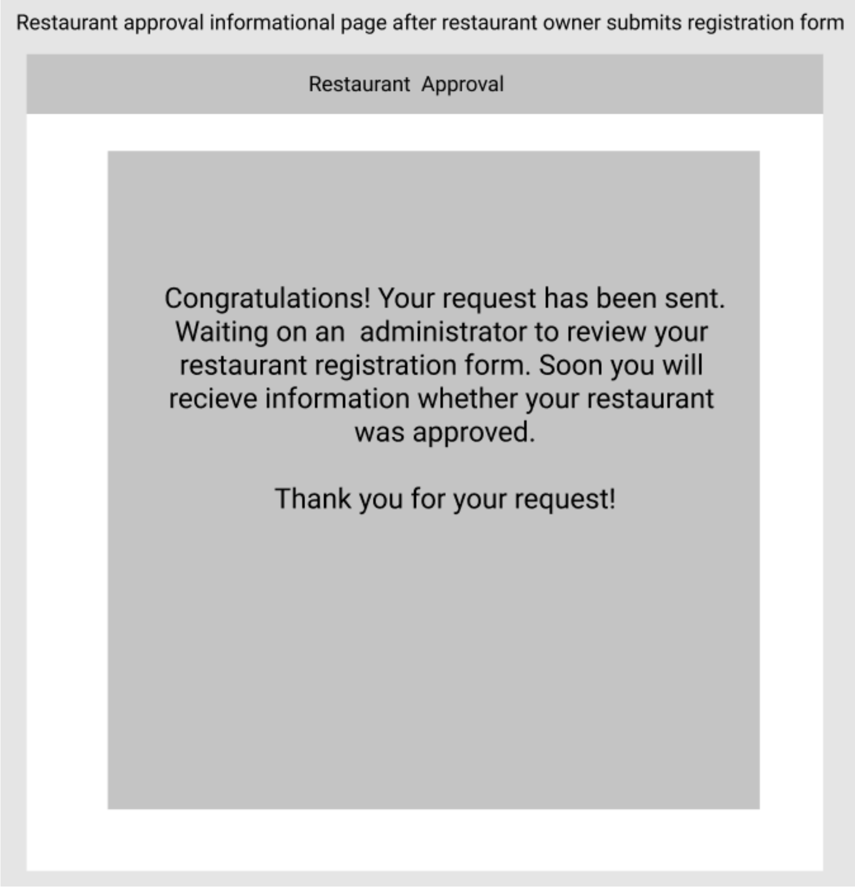
City:

State:

Zipcode:

Finish

After he registers, he sees this message being displayed letting him know his request is being processed.



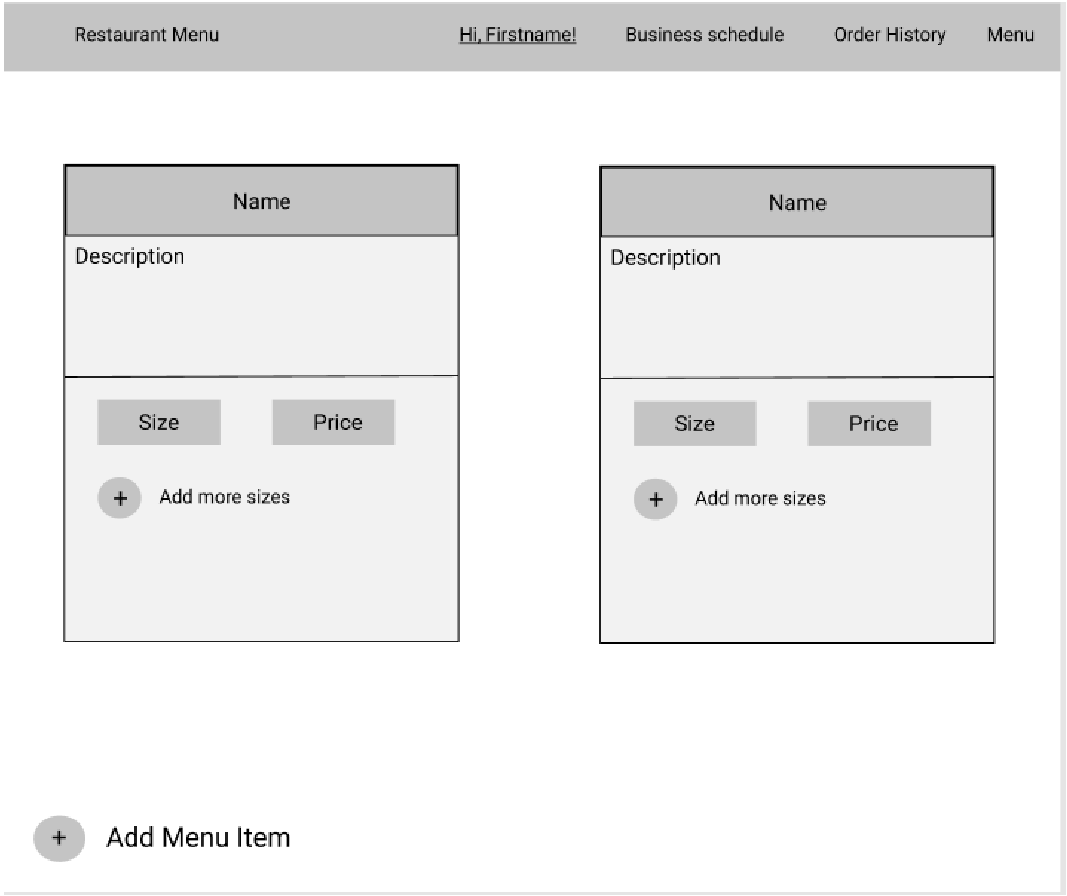
Restaurant approval informational page after restaurant owner submits registration form

Restaurant Approval

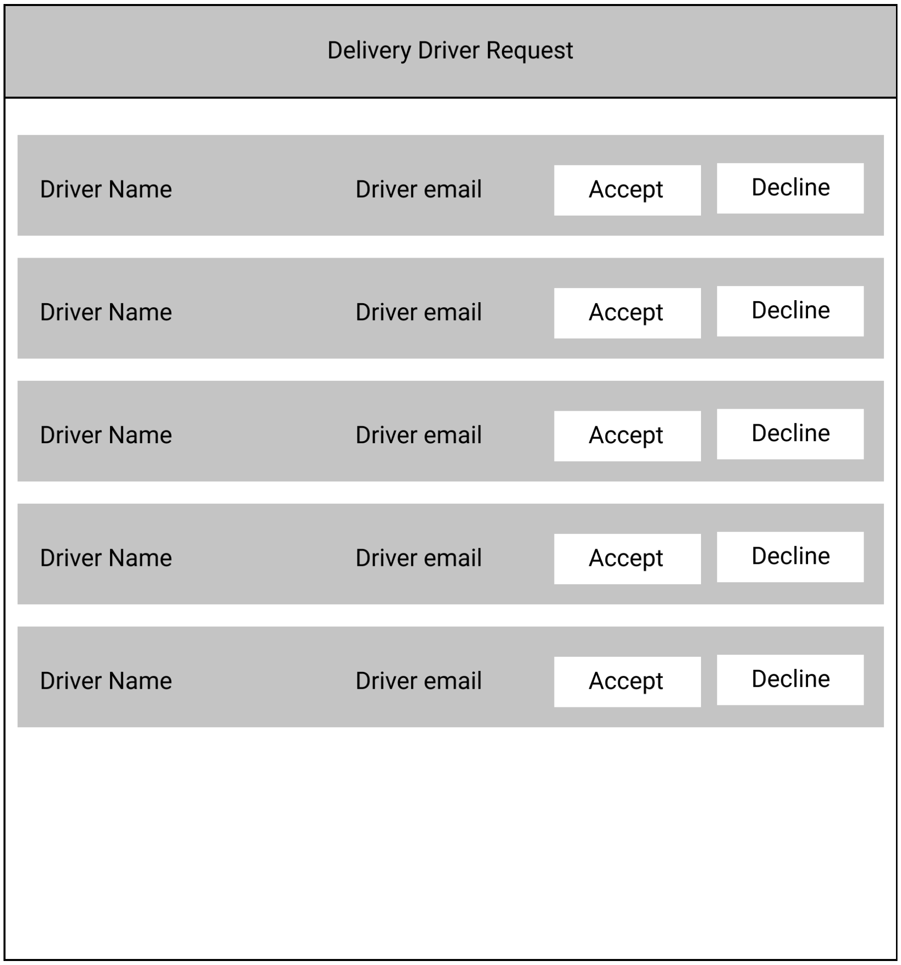
Congratulations! Your request has been sent. Waiting on an administrator to review your restaurant registration form. Soon you will recieve information whether your restaurant was approved.

Thank you for your request!

With his new restaurant registered and approved, he is now able to edit his restaurant page and add menu items.



Now that his account is set up, his restaurant has a menu, the last thing he needs is a delivery driver. The owner goes to “Approve Drivers” where he can approve or decline driver requests.



Delivery Driver Request

Driver Name	Driver email	Accept	Decline
Driver Name	Driver email	Accept	Decline
Driver Name	Driver email	Accept	Decline
Driver Name	Driver email	Accept	Decline
Driver Name	Driver email	Accept	Decline

Admin: Jess- an employee who wants to provide a safe atmosphere for the users and restaurants	**Use Case 5:** Administrator approving and updating the website app
 Pain point: ● Buggy software ● No UI ● Bad customers ● Bad drivers ● Bad restaurants ● Bad reviews Skills: ● Attention to detail ● Integrity and honesty ● Quality ● Has high technological skills ● Very quick learner ● Always has a technological device ● SQL Goals: ● Provide the best service	Jess is an employee who is fascinated with computers and is always on the lookout for new ways to bring a community together. She currently works as the administrator for our website application and she is continuously reviewing restaurant applications, menu items, and reviews. She has to approve all postings that the restaurant posts before they are published to the website. She is happy that she gets to work with the MySQL Workbench, as the UI makes it easy to do her job. Jess is the facilitator and likes to keep everyone in check. Jess communicates with the drivers and restaurants reassuring that they provide safe service. Jess' goal is to filter out all profanity and reach out to restaurants and drivers who receive bad reviews. She wants to make sure that all restaurants are safe and inviting to all users, and that they follow our guidelines and community standards. Jess wants to make sure that the website app is frequently used and that it becomes popular among the SFSU campus as well as restaurant owners and drivers.

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

Main terms

1. **Guest**: A user that has not registered
2. **User**: Registered drivers, restaurant owners, and SFSU students / faculty
3. **Customer**: Faculty and Students who are registered and are ordering food
4. **Restaurant**: A business that prepares and serves food and is registered
5. **Administrator**: The administrator of the database
6. **Driver**: An person who delivers food from restaurants to campus
7. **Order**: An order placed by User
8. **Restaurant Owner**: A user that owns a restaurant and registered
9. **Log in**. Not login or sign in or signin.
10. **Register**. Not signup or sign up.
11. **Cuisine**. Used to refer to food types/categories like Italian, Indian, etc.

Types of Users (In terms of permissions)

1. **Guest (**NOT REGISTERED)
   - 1.1. Has limited access to the website because they are not registered
   - 1.2. Has permission to access as much of the website as possible without having to register (lazy registration).
   - 1.3. Has permission to sign up as a registered user if a student or faculty.
2. **Customer** (REGISTERED student/faculty)
   - 2.1. Has all the permissions as the guest and more
   - 2.2. Has permission to buy and place food orders from approved restaurants.
3. **Administrator**
   - 3.1. Has special permission to the database to perform maintenance and approve restaurants.
4. **Restaurant Owne**r (MUST REGISTER AND LOGIN)
   - 4.1. Has permissions to upload more menu items to their online restaurant menu
   - 4.2. Has permission to remove items from their online restaurant menu.
5. **Delivery Driver** (MUST BE REGISTERED)
   - 5.1. Has permission to know what the customer ordered
   - 5.2. Has permission to know the location of the requested customer order
   - 5.3. Has permission to view unclaimed delivery orders
   - 5.4. Has permission to select an unclaimed delivery and deliver it

Data structures

1. **Guest :** An object for a user that is not currently logged in with the fields:
2. **Time_opened_website**: time the guest opened the website
3. **Device**: device the guest is using
4. **Area**: current area the guest is browsing

5. **Registration_details**: the details a user must submit in order to register, a ‘*’ represents mandatory information
   5.1. Firstname*
   5.2. Lastname*
   5.3. Sfsu email*
   5.4. Nickname
6. **Customer:** An object that has information about a person who is currently using the website that is not necessarily registered with the fields:
   6.1. Time_logged_in - time logged in
   6.2. Account - account name of customer
   6.3. Area - current area the guest is browsing
7. **Menu:** Data structure that holds list of food and drink items a customer can order
8. **Food:** A food object that contains the following fields:
   8.1. Name - name of food
   8.2. Food type - type of food
   8.3. Price
   8.4. Size
   8.5. Description
9. **Driver:** A driver object that contains:
   9.1. Name
   9.2. Distance
   9.3. Order the driver is delivering
   9.4. Employer
10. **Restaurant:** A restaurant object that contains:
    10.1.1. Name
    10.1.2. Address
    10.1.3. Menu
    10.1.4. PriceRating
    10.1.5. Description
11. **Order:** An object that contains details about a specific order such as:
    11.1.1. Total
    11.1.2. Customer
    11.1.3. CreatedAt
    11.1.4. DeliveryAddress
    11.1.5. Items
    11.1.6. RestaurantName
    11.1.7. RestaurantAdd
12. **User:**
    12.1. Name
    12.2. Email
    12.3. Address
    12.4. Password

Main Entities

1. **Account:** The account entity is in charge of keeping track of the entities that are tied to the user's specific account such as current orders and past purchases.
    1.1. **Guest**: A person who is using the website but is not registered. It is Unknown to whether they are a student, faculty, driver, or restaurant owner.
    1.2. **User**: Any student, Driver faculty, restaurant owner who is registered and is logged into their account
    1.3. **Restaurant Owner:** Special account for restaurant owners
    1.4. **Administrator:** Special account that has access to statistical information about the database to provide maintenance.
2. **Restaurant:** Holds the following sub fields,
    2.1. **Hours:** Hours restaurant is open
    2.2. **Location:** Where the restaurant
    2.3. **Restaurant Owner:** The user that is registered and claimed the restaurant
    2.4. **Menu**: List of food and drinks a customer can order
    2.5. **Food**: A food item a customer can order
    2.6. **Drink**: A drink item a customer can order
    2.7. **Order**: Has details about the order such as price, invoice, and what was **purchased** and when it was purchased
    2.8. **Invoice**: Purchase details of an order
3. **Delivery**
    3.1. **Delivery** Driver: an entity which has information pertaining to the delivery driver
    3.2. **Order** : Information of the order
    3.3. **Location**
        3.3.1. Street
        3.3.2. Address
        3.3.3. Zip Code

4. High level Architecture, Database Organization summary only

Search/filter architecture and implementation
To search for terms, we'll use the SQL %LIKE function. Searchable terms will be things such as restaurant names or cuisines. We will display the results alphabetically.

API Endpoints
User
Register: POST /user/register
Update account info: PUT /user/:userId/accountInfo
Delete account: DELETE /user/:userId/account
Get account info: GET /user/:user/accountInfo
Login: GET /user/login

Restaurants
Get list: GET /restaurant/all
Add restaurant: POST /restaurant/addRestaurant
Remove restaurant: DELETE /restaurant/removeRestaurant/:restaurantId
Update restaurant address: PUT /restaurant/updateAddress/:restaurantId

Menu:
Get menu: GET /restaurant/:restaurantId/menu
Add menu item: POST /restaurant/:restaurantId/menu
Remove menu item: DELETE /restaurant/:restaurantId/menu/:menuItemId
Edit menu item: PUT /restaurant/:restaurantId/menu/:menuItemId

Restaurant Owner

Driver
Accept driver: PUT /restaurantOwner/acceptDriver/:driverId

Order
Accept order: PUT /order/acceptOrder/:orderId
Decline order (restaurant owner): PUT /order/declineOrder/:orderId
Create order: POST /order/createOrder
Get order: GET /order/orderId
Update order comment PUT /order/updateComment/:orderId
Cancel order (customer): DELETE /order/cancelOrder/:orderId

Business Schedule
Update business hours PUT /businessSchedule/update/:businessScheduleId

Search
By restaurant name: GET /search/searchRestaurants?restaurnatName=<restaurantName>
By restaurant distance from SFSU campus:
GET /search/searchRestaurants?distanceFrom=<address>

5. Initial list of functional requirements

Priority I

1. Restaurant Owner
   - 1.1. Restaurant owner shall be able to add / remove menu items.
   - 1.2. Restaurant owner shall be able add / remove their restaurant.
   - 1.3. Restaurant owner shall be able accept drivers to work for them.
   - 1.4. Restaurant owner shall be able to view orders.
   - 1.5. Restaurant owners shall be able to set their business own hours.
   - 1.6. Restaurant owners shall be able to set their business schedule.
   - 1.7. Restaurant owners shall be able to set their business' price rating.
2. User
   - 2.1. User shall be able to register for an account.
   - 2.2. User shall be able to edit their account information.
   - 2.3. User shall be able to login to the website.
3. Customer
   - 3.1. Customer shall be able to add / remove products to the shopping cart.
   - 3.2. Customer shall be able to browse restaurants.
   - 3.3. Customer shall be able to set their own location for delivery.
   - 3.4. Customer shall be able to search restaurants by category.
   - 3.5. Customer shall be able to search restaurants by name.
   - 3.6. Customer shall be able to filter search results by restaurant price.
   - 3.7. Customer shall be able to filter search results by distance to SFSU campus.
4. Application
   - 4.1. App shall be able to authenticate users upon login.
   - 4.2. App shall be able to validate a user's email upon registration.
   - 4.3. App shall calculate the cart's total in real time.
5. Driver
   - 5.1. Driver shall be able to access a navigation map.
   - 5.2. Driver shall be able to have access to a campus map.
6. Administrator
   - 6.1. Administrator shall be required to approve restaurants.
   - 6.2. Administrator shall be able to suspend accounts.

Priority II

1. Restaurant Owner
   - 1.1. Restaurant owner shall be able to upload restaurant pictures.
   - 1.2. Restaurant owners shall be able to close for multiple days.
   - 1.3. Restaurant owner shall be able to add menu item size options.
   - 1.4. Restaurant owner shall be able to see their order history.
2. User
3. Customer
   - 3.1. Customer shall be able to receive order status notifications.
   - 3.2. Customer shall be able to see their order history.
4. Application
5. Driver
   - 5.1. Driver shall be able to update the order status.
6. Administrator

Priority III

1. Restaurant Owner
   - 1.1. Restaurant owner shall be able upload menu item pictures.
   - 1.2. Restaurant owner shall be able to add multiple restaurants.
   - 1.3. Restaurant owner shall be able to add / remove / change pictures,logo’s banners.
   - 1.4. Restaurant owner shall be able to add extra topping options to their menu items
2. User
3. Customer
   - 3.1. Customer shall be able to cancel order for free within a penalty-free time limit.
   - 3.2. App shall give users promotions / deals.
4. Application
   - 4.1. App shall update driver’s location for the customer to see.
5. Driver
6. Administrator
   - 6.1. Administrator shall be able to remove accounts.

6. List of non-functional requirements

6.1. 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

6.2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers

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

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

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

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

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

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

6.9. Application shall be very easy to use and intuitive

6.10. Application should follow established architecture patterns

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

6.12. Google analytics shall be used

6.13. No email clients shall be allowed.

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

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

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

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

6.18. 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).

7. Identify *actual* key risks for your project at this time

Skill risks:
Not everyone is familiar or has much experience with the stack we're using. To mitigate this risk, those who have more experience with it will teach others. Also, tasks will be assigned according to their skills. We will match teammates up with those who are more knowledgeable with those who are not.

Schedule risks:
Our schedules are pretty flexible, so we haven't had any trouble setting up meetings and attending them.

Technical risk:
We don't have any technical risks at the moment. We don't foresee any in the future.

Teamwork risks:
Everyone seems to be getting along with each other and everyone contributes.

Legal/content risks:
We are making sure that all images we use are not copyrighted, and we are not asking for any sensitive information.

8. Project management

We will start using Trello. The team lead will assign general tasks to the front end and back end, then the FE and BE leads will break those down and assign specific tasks. With Trello, we can all see who is in charge of what, as well as which things still need to be done or which ones are still in progress. This should give us a better overview of our progress and help us come up with time estimates for accomplishing our milestones as well as get a better feeling of whether we're behind schedule or not.

Also, we will continue having our regular meeting during class on Wednesdays and at 10AM every Friday as a group, while FE and BE teams can set up their individual meetings throughout the week as they see fit. Lastly, we also use Discord and Google Docs to collaborate.

9. Competitive analysis

Feature	Grub Hub	Door Dash	Uber Eats	Hermes
Register for an Account	+	+	+	+
Browse restaurants	+	+	+	+
Shopping Cart	+	+	+	+
Location search	+	+	+	+
Recommended Restaurants	+	+	++	+
Able to choose delivery or pickup	+	+	+	+
Schedule	+	+	+	+
Pay online	+	+	+	-
Specific delivery location and time on campus	-	-	-	++
Campus map for drivers	-	-	-	++

Our advantages lie in the focus of deliveries to SFSU students and faculty with unique functions to make it more convenient to SFSU faculty and students such as a feature where the user can schedule food to be delivered at a specific location at a specific time on campus, allowing the user to pick up their orders when they have free time, ie. leaving after class. With online orders for our app, the main location to have food orders delivered will be around the SFSU campus so delivery men and women will know the general location of where their orders will be delivered to. With the inclusion of basic features of other food delivery apps, SFSU students and faculty will have an easier time having food delivered.

10. High-level system architecture and technologies used

Operating System	Ubuntu Server 20.04 LTS (HVM), SSD Volume Type 64-bit (x86)
Database	MySQL v8.0.23
Web Server	NGINX 1.18.0
Server-Side Language	JavaScript
Web Framework	React, Node (v14.15.5), Express
IDE	Visual Studio Code
Libraries	Bootstrap/React-Bootstrap

11. Team and roles

Alex	Backend
Amit	GitHub Master
Angela	Front End
Jacob	Backend Lead
Jarvis	Front End Lead
Roberto	Team Lead / Document Master

12. Checklist

So far all team members are engaged and attending ZOOM sessions when required.	**OK**
Team found a time slot to meet outside of the class	**OK**
Back end, Front end leads and Github master chosen	**OK**
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	**ON TRACK**

Team lead ensured that all team members read the final M1 and agree/understand it before submission.	**OK**
Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)	**OK**