Albert Le, Maximilian Mok, Hoai An Nguyen, Lam Nguyen, Thu Theresa Nguyen

Panjiyar Krishna, Kevin Pham

Group VI

CMPE 131, Section 3

3 December 2020

Part 1 (Updated)

Everything highlighted in yellow are the updated changes

Project Requirement

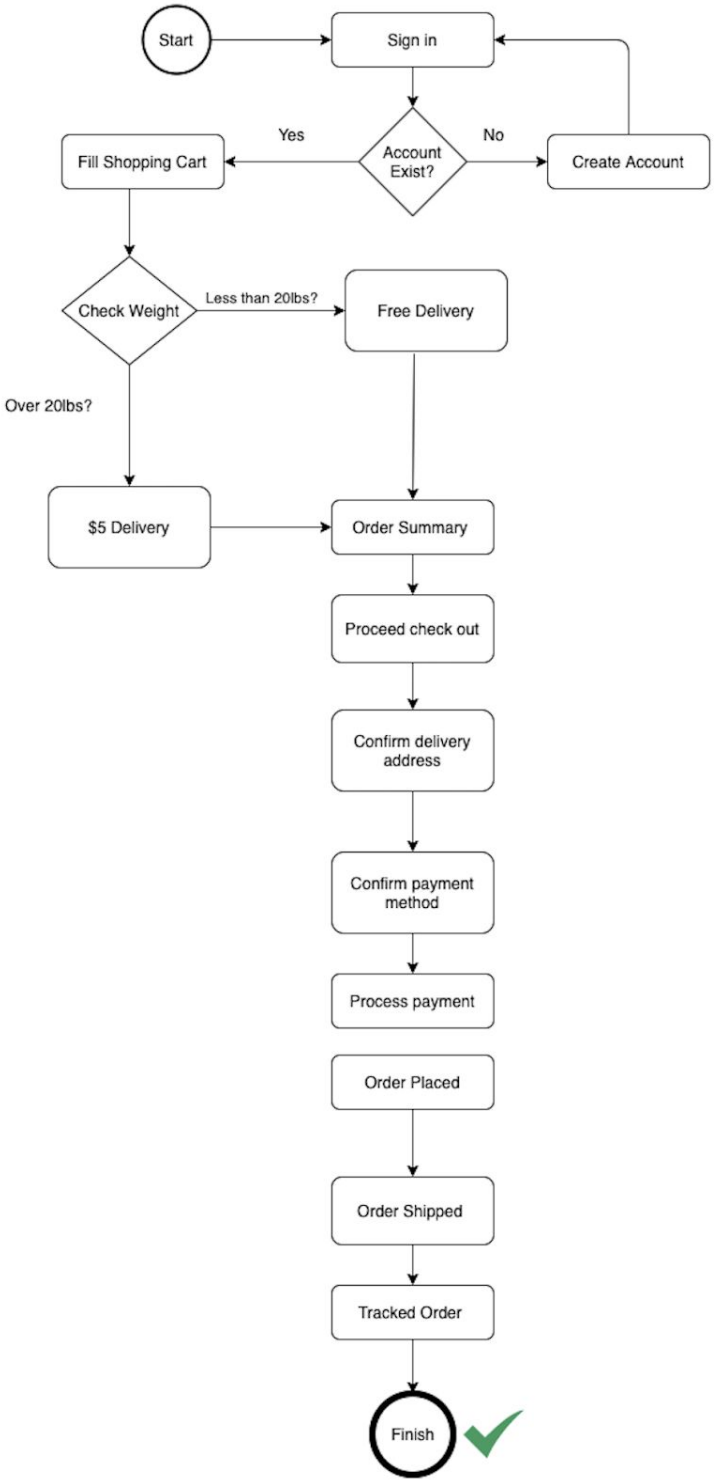
Project Statement

A new online grocery store for retailer food chain, “Organic Food Store”, wants to design and create an online web based store. The main location of the Organic Food Store is based at San Jose State University. The website would support features that allow customers to search inventory, place orders, process payments, and deliver items to the customer’s home. Additionally, the website would allow users to create an account which would store their personal information for future purchases. This type of online grocery shopping and delivery would be ideal for many during the pandemic as it would allow for consumers to receive their grocery while maintaining minimal interaction.

Proposed Solutions

- OFS will allow consumers to purchase organic groceries through an online shopping cart
    - Cart keeps track of customers items as well as the total cart's weight
        - If total weight is less than 20 lbs then delivery is free
        - If weight more than 20 lbs then $5 delivery fee would be added to order
- OFS will include a database table that keeps track of the total inventory of items
    - This database table will include the item's name, weight, ID, category, URL of the product image and price
- OFS will include an additional database tables that keeps track of customers' account information
    - One database table will include the customers' username, password, home address, previous orders
    - Second one includes payment information after the customer updates it in their profile screen.
- OFS will support online payment methods such as Visa and Mastercard THIS IS SIMULATED, WE DO NOT ACTUALLY PULL REAL CARD INFO IN REAL TIME
- OFS will deliver purchased items to the customer's home address after the payment is processed

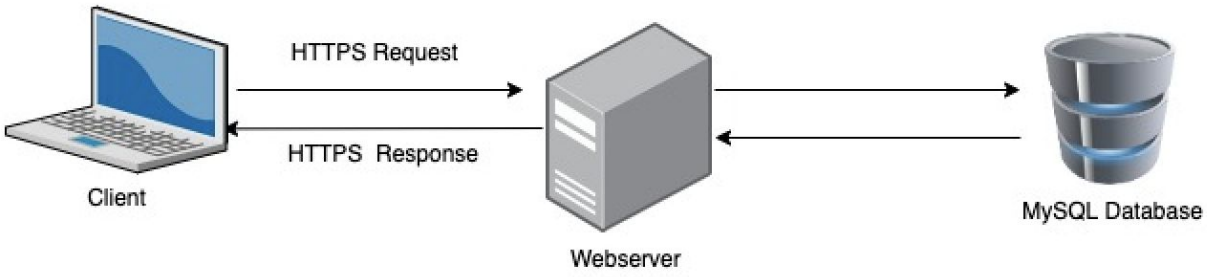
OFS Web Application Flowchart



WE DID NOT IMPLEMENT TRACKED ORDER FROM THE PREVIOUS DIAGRAM

Architecture

OFS Architecture Diagram



OFS Use Cases:

Case 1: Sign In

	Variation	User's Action	System's Action
1	Successful login	User attempts to sign in with email and correct password	System finds the user in the database and logs the user in
2	Wrong Password	User attempts to sign in with email and wrong password	Prompt user to try again or recover the password
3	User is not registered yet	User attempts to sign in with email that system does not recognize	Prompt user to create a new account

Variation #1:

1.1 System checks for any matching emails and checks the password.
1.2 User signs in successfully
1.3 System displays the greeting message and redirects user to "My account"

Variation #2:

1.1 The user is found in the database, but the password is incorrect.
1.2 System displays the error message.

1.3 User is asked to try again.
1.4 User chooses to recover the password.
1.5 System redirects the user to the password recovery page.

Variation #3:
1.1 The email is not found in the database.
1.2 System displays the error message and prompts the user to sign up.
1.5 System redirects the user to the sign up page.

Case 2: Search

	Variation	User's Action	System's Action
1	Product is found	User enters the key word in search engine	System finds a match and displays it
2	Product is not found	User enters the key word in search engine	System shows that the product is not found

Variation #1:
1.1 User enters their email and password.
1.2 System stores email and password into the database.
1.3 System displays a message of successful registration and redirects the user to the login page.

Variation #2:
1.1 User enters their email.
1.2 System checks for any matching email in the database and finds a match.
1.3 System displays an error message and redirects to the sign in page.

Case 3: Sign Up

	Variation	User's Action	System's Action
1	Successful registration	User enters all necessary information	System adds the new user to the database, shows a successful registration message and redirects

			to homepage
2	User already has an account	User attempts to sign in with email that already exist in the database	System shows an error message and prompts user to try again
3	Incorrect Input	User leaves at least one field blank or the format of data is incorrect	System prompts user to fill the empty field or change the data

Variation #1:

1.1 User enters the email and password to create a registration for the order.

1.2 System analyzes the email and password.

1.3 If the email and password is validated, then the user is successfully added to the database.

1.4 System shows the registration successful message.

1.5 System redirects the user back to the homepage.

Variation #2:

1.1 User enters the email and password to create a registration for the order.

1.2 System analyzes the email and password.

1.3 System detects that the email entered is already an email from an existing account and shows an error message.

1.4 System redirects the user back to sign in page.

Variation #3:

1.1 User enters the email, password and address to create a registration for the order.

1.2 System highlights empty fields.

1.3 User fills out empty fields.

1.4 System shows the registration successful message.

1.5 System redirects the user back to the homepage.

Case 4: Make a Payment

	Variation	User's Action	System's Action
1	Successful Payment	Submits card information in Payment Method tab in Account page	Card information record is inserted into a database respective to the Customer record.

2	No Payment Method	Go to Account page, Payment Method tab and fill out cart information	System shows “add payment method” button instead of “place an order” button.

Variation #1:

1.1 User enters their card information in account page, payment method tab

1.2 System stores account number, cardholder name, expiration date into database.

1.3 System will display saved payment information on checkout page and account page

Variation #2:

1.1 User did not enter the payment method before checking out.

1.2 User sees the “add payment method” button and pressed it

1.3 System redirects the user to account page

1.4 User enters their card information in Account Payment Method

1.4 System adds payment information to the database.

1.5 System will display saved payment information on checkout page and account page

Case 5: Checkout

	Variation	User’s Action	System’s Action
1	Total cart weight is less than 20 lbs	User adds items to their shopping cart and presses the checkout button	Total cart weight is calculated and if it is lower than 20lbs, no delivery fee will be applied
2	Total cart weight exceeds 20 lbs	User adds items to their shopping cart and presses the checkout button	Total cart weight is calculated and if it exceeds 20lbs, then a $5 delivery fee will be added to the order total

Variation #1:

1.1 The user adds desired items to their cart.

1.2 User presses checkout button when done shopping.

1.3 System calculates the total weight of the cart by adding the weight of each item together

1.4 If the weight of the cart is less than 20lbs, then no delivery fee or free delivery will be applied.

Variation #2:

1.1 The user adds desired items to their cart.

1.2 User presses checkout button when done shopping.

1.3 System calculates the total weight of the cart by adding the weight of each item together

1.4 If the weight of the cart is more than 20lbs, then a $5 delivery fee will be added to the order total.

Case 6: Admin Privileges

	Variation	Admin's Action	System's Action
1	Manage Order	Alter and update orders from customer	System shows an error message and redirects to sign in page
2	Update Products	Add, drop and update items within the online store	Display the updated changed accordingly from the database

Languages and technologies

Front End	**Back End**	**Host**
html	PHP	XAMPP
css	~~Javascript~~	Firebase
Javascript	MySQL	

Collaboration Tools

Tools	Description
Zoom	Video conferencing and sharing screens
Discord	Instant messaging and live chat
GitHub	Version control and collaboration
Asana	Project organizing, tracking, and managing
Apache	Open-source cross-platform web server software