

Simple Shopper

Sprint 2

CSC 4350 - Fall 2019

Group 4

Amaal Abdi, Diego Gonzalez,

Mohammad Mamun, and Kathy Nguyen

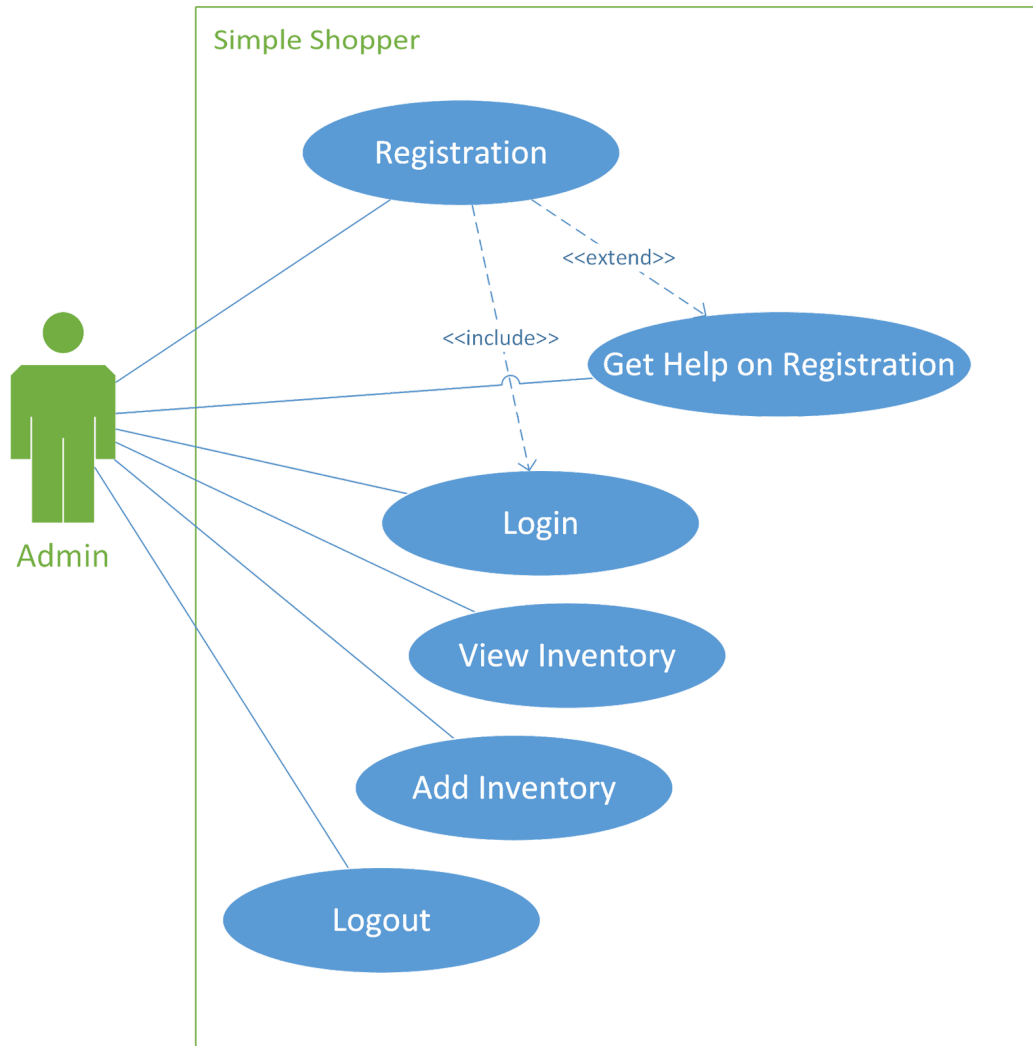
Assignee Name	Email	Task	Duration (hours)	Dependency	Due date	Evaluation
Diego Gonzalez	dgonzalez16@student.gsu.edu	-Use case log in, Log out -Github -Database Specifications	3 hrs	Use cases System specs for database specs.	09/28/19	100%
Kathy Nguyen	knguyen78@student.gsu.edu	- Use cases 6,10,11 - Requirements 2, 6,10,11 - Use Case Diagram	2.5 hrs	Use case for diagram	10/01/19	100%
Mohammad Mamun	mmamun1@student.gsu.edu	- Use cases 1,5,12 -Requirements 1,3,5,12	2 hrs	Use case diagram	10/01/09	100%
Amaal Abdi	aabdi11@student.gsu.edu	-Use cases 4,7,8,9 - Requirements 4,7,8,9 -General formatting	3.5 hrs		10/01/19	100%

Problem Statement

Simple shopping is a project made for people with a restrictive diet. This Product is a way to make grocery shopping simple and convenient for all users. The Product will sort items based on user needs. It will categorize foods and other merchandise into a specific group. For example: Vegans, vegetarians, etc. This Product can solve confusion while navigating through a complex shopping system when looking for specific grocery items. Our main competitors are Thrive, Walmart, Publix, etc., but they mostly sell their branded Product. Simple shopping will try to combine products from all brands and companies and will not require a membership fee. Simple

shopping will be simple to produce and will not require substantial resources. This Product can be built using a simple database and will run on almost all platforms. It will be simple to develop, but the actual huge cost will be labor and the maintenance of the Product.

Use Case Diagram



Use Cases

Use Case #: 1

Name of the use Case: Registration

Actors: Customer, Admin, Database

Description:

- Users will be given an option to register for the website in all the web pages available in our system.
- User clicks on the link to the registration page.
- User inputs their Important information (address, credit card info (optional), name , date of birth, diet, etc.)
- Checks if the information given valid or invalid (credit card / address/email address/ username/ password).
- If the information is given not valid, display error and let them try again.
- Send the users verification message via email.

Exception Path: If the user does not want to register or their information is already in our database.

Alternative Path: There is no need to be registered to shop.

Pre-condition: Needs to click the registration button / link.

Post-condition: The users are registered into the system and their information is saved in our database.

Use Case #: 2

Name of the Use Case: Log In

Actors: Customer, Admin, and Database

Description:

- The user will be given the login page, once accessing our site.
- The user will enter email and password.
- After the inputs are complete, the user will click login.
- If successful, the user will be brought to our inventory page.
- If unsuccessful, the user will be prompted with an error message to try again.
- The information will be validated with database.

Exception Path: If the user does not have an account they will need to Sign Up

Alternate Path: only one path

Pre-condition: Account with valid login credentials

Post-condition: getting to the inventory page.

Use Case #: 3

Name of the Use Case: Log Out

Actors: Customer, Admin

Description:

- At any point of viewing our website the user should be able to navigate to the log out button
- After ordering something the user will be prompted to either log out or continue shopping with us.

Exception Path: internet disconnection which would automatically log out user.

Alternate Path: User can either log out or keep shopping

Pre-condition: Internet, Computer, interfaces to interact with our website(mouse, keyboard)

Post-condition: The logged out page presented after logging out

Use Case #: 4

Name of the Use Case: View Inventory

Actors: Customer, Admin

Description:

- After the user logs in, the page redirects to another page that displays all of the inventory of the grocery website.
- Depending on who the user is, different actions will be available.

Exception Path: none

Alternate Path: If the user chooses to add filtering to the inventory (see use case #5), the inventory will appear differently depending on the level of filtering. If the user is a customer, each item of the inventory will have an “Add to Cart” button associated with it, which will take the user to the “View Cart” page (see user case #7). If the user is an admin, each item will have an “Edit” button next to it, and the page will include an “Add New Inventory” button at the top (see use case #6).

Pre-condition: The user should be logged in as an admin.

Post-condition: Display of inventory items, redirect to either view of cart items (for a customer) or an inventory editing page (for an admin)

Use Case #: 5

Name of the Use Case: Filter Inventory

Actor: Customer

Description:

- Customer have the option to browse shopping items in our site
- Customer clicks on the button/ link to pages.
- Clicks on tag / categories.

- Error if too many tags presents.
- Prompt to remove few tags.
- Inventory is sorted based on the input given by the customer.
- The link displays the page with filtered items.

Exceptional path: user does not click the button.

Alternate path: entering the direct address to the page.

Pre-condition: Needs to be on our site.

Post-condition: Entering filtered page.

Use Case #: 6

Name of the Use Case: Add Inventory

Actors: Admin, Database

Description:

- The admin will identify which inventory item needs updating when new shipment arrives.
- If the item is new and not yet in the system, admin will add inventory by entering the name, quantity, and description.
- The admin will select which item to update or add.
- The admin will enter in a new quantity for the item.
- The admin will repeat this process for all other inventory items that need updating.

Exception Path: If the account is not admin level, show error.

Alternate Path: The admin can cancel editing at any time by returning to the inventory page.

Pre-condition: Account used to login must be admin level.

Post-condition: Confirmation of changes by viewing current inventory levels.

Use Case #: 7

Name of the Use Case: View Cart Items

Actors: Customer

Description:

- The customer clicks on a button that will take them to a list of the items they put into their cart.
- The page will show the total price of all items.
- The page will also have links to purchase items or to return to the inventory.

Exception Path: none

Alternate Path: There are two links, labeled “Continue Shopping” and “Purchase Items”. If the user clicks “Continue Shopping”, the page will redirect to the inventory page (use case 4). If the user clicks “Purchase Items”, the page will redirect to the payment page (use case 9).

Pre-condition: The user must be logged in.

Post-condition:

Use Case #: 8

Name of the Use Case: Add Cart Items

Actors: Customer

Description:

- Each item in the inventory will have an “Add to Cart” Button next to it.
- If the user clicks on one of these buttons, the page will redirect to the “View Cart” page.
- The item that was clicked will be added to the cart, and its price will be added to the total price.

Exception Path: none

Alternate Path: none

Pre-condition: The user must be logged in to add items to the cart.

Post-condition: The page redirects to the “View Cart” page. The item is added to the customer’s cart, and the price of the item is added to the total price.

Use Case #: 9

Name of the Use Case: Pay for Cart Items

Actors: Customer

Description:

- The customer clicks on a button in the “View Cart” page to purchase the items in the cart.
- This takes the user to a payment form, where user must enter card information in order to pay for the items.

Exception Path: If the card information is incorrect, the page will indicate what fields are incorrect. The user will have to enter this information again.

Alternate Path: none

Pre-condition: There must be at least one item in the cart

Post-condition: Confirmation of purchase.

Use Case #: 10

Name of the Use Case: Edit Profile

Actors: Customer

Description:

- Customer will select edit profile to change any information that is displayed on their profile.
- System will display all editable fields where customer will enter in new information.
- System will update when customer saves the new details.

Exception Path: If any fields are left blank, customer will be prompted to enter a value for blank fields.

Alternate Path: none

Pre-condition: Customer must be logged in with a valid account.

Post-condition: Confirmation of changes by viewing profile with new details.

Use Case #: 11

Name of the Use Case: View Profile

Actors: Customer

Description:

- Customer can view profile details after logging in.
- Details to be displayed may include: Name, Age, Diet, and other details that were initially entered during the registration process.

Exception Path: none

Alternate Path: Customer may choose to edit any details by selecting Edit Profile (see use case # 10)

Pre-condition: Customer must be logged in with a valid account.

Post-condition: none

Use Case #: 12

Name of the Use Case: Get Help on Registration

Summary: The users needs help with registration.

Actors: Customer

Description:

- The user needs help on the registration page.
- Give them example for each specific field (username, Password, date of birth, etc.)
- If the user still having problems give them contact information for customer support.

Exceptional path: none

Alternative path: not registering;

Pre-condition: The user is in registering phase.

Post-condition: The user was able to register for the site

System Requirements

Requirement #: 1 Use Case #: 1

Introduction: User registering for the website.

Inputs: Email address, Password, Credit card information and dietary information.

Requirements Description:

- The email address must be valid and unique.
- The user name can have letters and numbers, but no space or special characters
- The username must start with a letter
- Password must be 8-12 characters and must contain upper and lower case letters. No special characters.
- Credit card information must be valid.
- If the data is not valid give them example of format
- If the information is not valid display error and let the user try again.
- Once the valid information is given save all the data to our database

Outputs: Registered user.

Data saved in our database

Requirement #: 2 Use Case #: 2

Introduction: Logging In

Inputs: Username, password, signin button

Requirements Description:

- User will enter username and password.
- If the credentials are correct, the user is logged in and now has access.
- If the credentials were incorrect, an error will show and prompt the user to try again.

Outputs: Error or successful log in

Requirement #: 3 Use Case #: 3

Introduction: logout

Inputs: clicking logout button

Requirements Description:

- User will click the logout button to log out.
- The user should be logged out of the system.
- This will end the users access to cart, inventory, personal information, etc.

Outputs:

The user can no longer access private information from the website, unless logged back in.

Requirement #: 4 Use Case #: 4

Introduction: Display inventory so that the user can browse for items.

Inputs: Button click

Requirements Description:

- User must be logged in
- Items should have either an “Add to Cart” button or an “Edit” button, depending on user type.

- The page should display all inventory items determined selected filter options, if filter is used (see use case #5)

Outputs: List of available inventory items

Requirement #: 5 Use Case #: 5

Introduction: Filtering inventory

Inputs: Tags, option selection

Requirements Description:

- Every item will be labeled with tags that describe what it is safe for (vegetables for vegetarians, dairy product will be labeled with dairy tag. Etc.
- Tags cannot contradict each other(example: meat and vegetarian cannot be selected at the same time).
- Display items with those tags.
- Removing tag will re-filter the list.

Outputs: Filtered list of items.

Requirement #: 6 Use Case #: 6

Introduction: Admin updating inventory.

Inputs: New inventory values (name, quantity, description, diet)

Requirements Description:

- Accepts the new inventory value of items.
- Updates that value in the database.

Outputs: Saves new inventory values and displays on the view inventory page.

Requirement #: 7 Use Case #: 7

Introduction: Viewing cart items for review before purchase or before continuing shopping.

Inputs: Button click

Requirements Description:

- Customer must be logged in

Outputs: Display of current cart items.

Requirement #: 8 Use Case #: 8

Introduction: Adding desired items to customer cart before purchase.

Inputs: Button click

Requirements Description:

- Customer must be logged in

Outputs: Redirect to view cart page, new item entered into customer's cart, addition of total price

Requirement #: 9 Use Case #: 9

Introduction: Entering information to purchase cart items

Inputs: Form input, submit button

Requirements Description:

- Customer must be logged in
- The cart must have at least one item in it
- Card information must be correct
 - All card fields must be filled
 - Card number must match the pattern for common card types (Visa, Mastercard, Discover, etc)
 - User must enter mailing address and billing address information

Outputs: Page showing confirmation of purchase.

Requirement #: 10 Use Case #: 10

Introduction: Customer can edit profile details

Inputs: Customer selects the option to edit profile details

Requirements Description:

- All editable fields will show on edit page.
- Information to edit includes: name, age, dietary restrictions, address, and phone number.
- Customer will enter any new details to replace old information.

Outputs: New details are saved and will display on the view profile page.

Requirement #: 11 Use Case #: 11

Introduction: Customer can view profile details

Inputs: Customer clicks link/button to view profile

Requirements Description:

- Page will show all details previously entered during registration or profile update.
- Information to display includes: name, age, dietary restrictions, address, and phone number.

Outputs: The profile page will show all the details that the customer previously entered.

Requirement #: 12 Use Case #: 12

Introduction: Get help on registration

Inputs: Customer clicks link.

Requirements Description:

- Customer will be taken to a page with example registration form.
- Display customer support info.
- If the customers wants more help. They can contact customer support.

Outputs: Should help with registration.

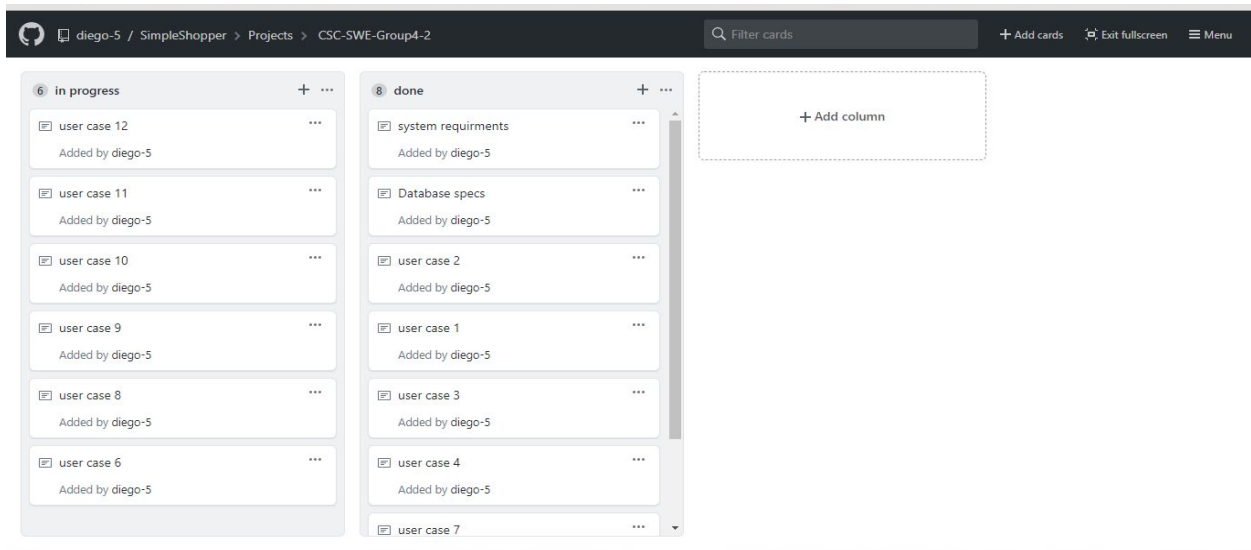
Database Specifications

Database Management System: MySQL

Tables:

- Customers: customerId(Primary Key), customer_first_name, customer_last_name, customer_email_address, customer_password, customer_phone_number, customer_address, customer_city, customer_state, customer_zipcode, customer_dob
- Payments: paymentId(Primary Key), card_Num, card_Exp, Card_Ccv
- Order confirm: order_confirmId(Primary Key), confirm_Num
- Food Inventory: food_id(Primary Key), food_desc, stock, price, Keto_diet, Paleo_diet, Atkins_diet, Vegetarian_diet, pescatarian_diet, vegan_diet, raw_diet, mediterranean_diet
- Customer_orders: orderId(Primary Key), customerId(Foreign Key), order_Date, status

Github Screenshots



Github Link: <https://github.com/diego-5/SimpleShopper>