CS 361 Software Engineering I

Requirements

Meal planner to promote healthy meals with low waste

10/20/2019

Group 24

John Casey III
Raymond Lieu
Todd Radin
NianJun Shi
Brian Shim

Requirements Definition

Functional Requirement

- The system shall authenticate the user's credentials before use.
- The system shall allow users to create an account.
- The system shall allow users to subscribe to the service.
- The system shall allow users to cancel an order and subscription.
- The system shall display curated meals and snacks that are healthy, inexpensive, and easy to prepare.
- The system shall display the ingredients to meals and snacks.
- The system shall display the cooking instructions for meals and snacks.
- The system shall display the nutritional information for meals and snacks.
- The system shall report any allergens associated with a meal or snack.
- The system shall allow users to order ingredients from the curated meals and snacks library.
- The system shall allow users to add ingredients from meals and snacks to a shopping list.
- The system shall allow users to provide payment for their order or subscription.
- The system shall allow users to track their order.
- The system shall allow users to view their order history.
- The system shall allow users to add meals and snacks to their favorites list.
- The system shall allow users to review purchased meals and snacks.
- The system shall allow users to filter meals and snacks such as by cooking difficulty, types of cuisine (Italian, Mexican, etc.), serving size, allergens, and rating, calorie range, total order number from all users.
- The system shall send notifications to users reminding them of healthy options, order status, etc.
- The system shall allow users to opt-in to receive a return label to send compostable items back to the distribution center.
- The system shall track food waste and carbon footprint associated with users' purchases and compost returns.
- The system shall prioritize displaying meals and snacks that utilize local, seasonal and low carbon footprint ingredients whenever possible.

Non-Functional Requirement

- The system shall authenticate the user within 10 seconds.
- The system shall allow users to create an account within 7 clicks.
- The system shall allow users to subscribe to the service within 120 seconds.

- The system shall allow users to cancel a subscription within five clicks.
- The system shall display curated meals and snacks within 15 seconds.
- The system shall display 9 items per page and load more items as the user scrolls down.
- The system shall display the information about each meal or snack within 10 seconds.
- The system shall allow users to add or remove an item to their cart within 10 seconds and 1 click.
- The system shall allow users to complete their order within 60 seconds and 5 clicks.
- The system shall allow users to cancel an order within 60 seconds and 5 clicks.
- The system shall display tracking information for a user's order within 30 seconds.
- The system shall display a user's order history within 30 seconds.
- The system shall display a user's order history going back to 1 year.
- The system shall allow users to add an item to their favorites list within 5 seconds.
- The system shall complete a search filter for food items within 15 seconds.
- The system shall send users notifications to browse and/or purchase meals and snacks on a weekly basis.

Use Cases

Use Case 1

Viewing and scheduling curated meals and snacks

Actors

- User

Preconditions

- User has an account and is logged in
- User has an active subscription to the service

Postconditions

- Meals and snacks are scheduled.
- Ingredients are either added to the shopping list or to the order for delivery.

Flow of Events

- User browses through curated meals and selects the one they would like to schedule.
- User selects quantity of servings they want to schedule.
- User picks the date to schedule the meal.
- User is prompted whether to add the ingredients to the shopping list or add to order for delivery.

Use Case 2

Ordering ingredients to prepare scheduled meals

Actors

- User
- Third-party delivery service

Preconditions

- User has an account and is logged in
- User has an active subscription to the service
- User has scheduled meals and added meals to order

Postconditions

- Third-party delivery service delivers order to user

Flow of Events

- User selects delivery dates for each meal in order
- User enters shipping information
- User enter billing and payment information
- User submits order
- Payment information is sent to third-party for processing
- Once payment has been approved, successful response is recorded and completion screen is displayed to user
- Order is added to order history for tracking
- Order information is sent to third-party distribution center
- Order is shipped from distribution center to user
- Tracking update notifications are provided to the user
- Order arrives at user-specified address

Use Case 3

Track supply chain waste index and carbon footprint

Actors

- User

Preconditions

- User has an account and is logged in
- User has an active subscription to the service
- User has scheduled meals through the service

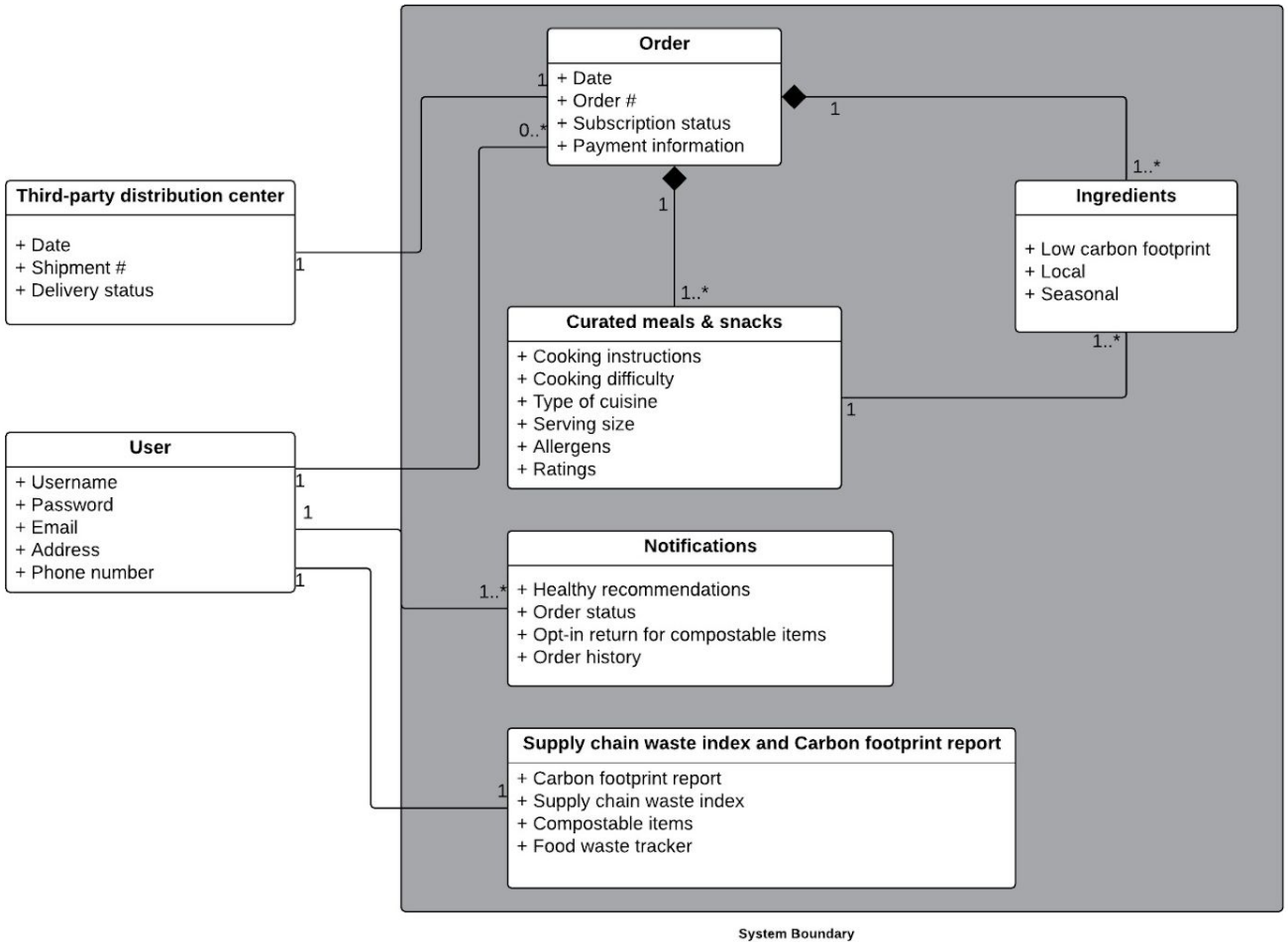
Postconditions

- User receives a summary of total supply chain waste index and carbon footprint

Flow of Events

- With each order completion, the system tracks and calculates the user's total supply chain index and carbon footprint from values associated with the ingredients
- User selects to view their personal waste and carbon footprint report
- User specifies the duration from which to generate the report
- Report is displayed on the screen

UML Class Diagram



Requirements Specification

Functional Requirement

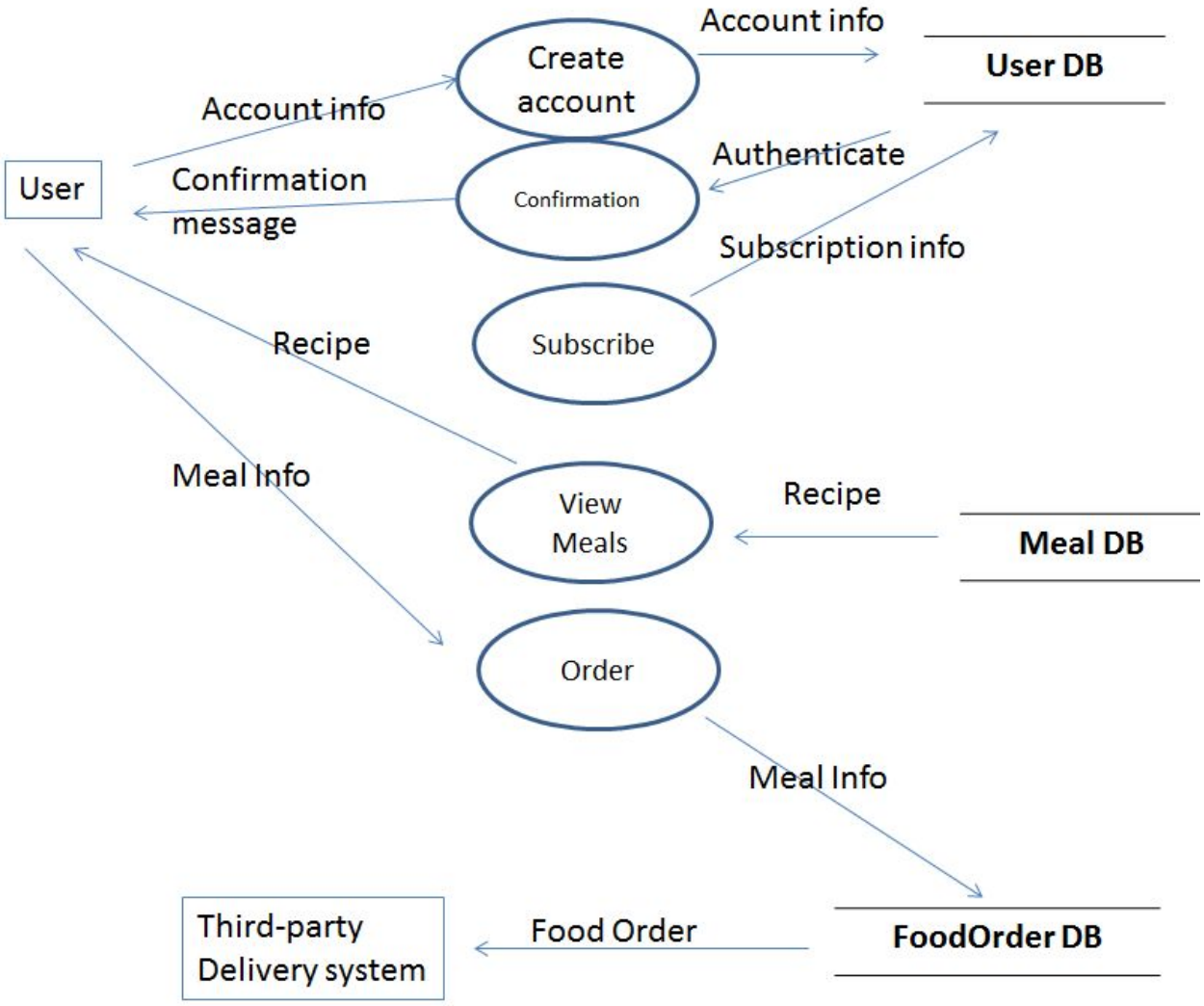
- The system will validate user credentials against credentials stored in the database.
- The system will validate new user accounts against current users in the database to check if user already exists.
- The system will validate users against database and prompt them to subscribe if they are new users.
- The system will save and store sessions to reduce the number of logins by the same user.
- After login, the system will display recommended meals to users based on previous purchases.
- Upon selecting a meal or snack, the system will gather information about item from database and send that information to display to the user.
- Upon adding an item to a user's cart, item information will be sent to the system to keep track of.
- During payment, the system will send information to third-party to process payment.
- Upon successful payment, the database will record order information.
- Upon canceling an order, information will be sent to the system and the database will be updated and refund issued via third-party.
- After adding item to favorites list, the client will send item and user information to system's database to record.
- Upon searching with filter, the client will send a request to the database and the system will display result to user.
- The system will maintain carbon footprint and supply chain waste index data in the database for all ingredients to be used in tracking food waste and carbon footprint associated with users' purchases and compost returns.
- Upon order completion, the system will send shipment information to third-party and track delivery status from third-party database.
- Upon selecting track order status, the client will send the request to the system, which in turn, retrieves information from third-party database.
- Upon scrolling through meal and snack list, the client will make additional asynchronous requests to the database to retrieve additional meals and snacks and the system will display results to user.

Non-Functional Requirement

- The system shall query the database for user authentication and return result within 10 seconds.

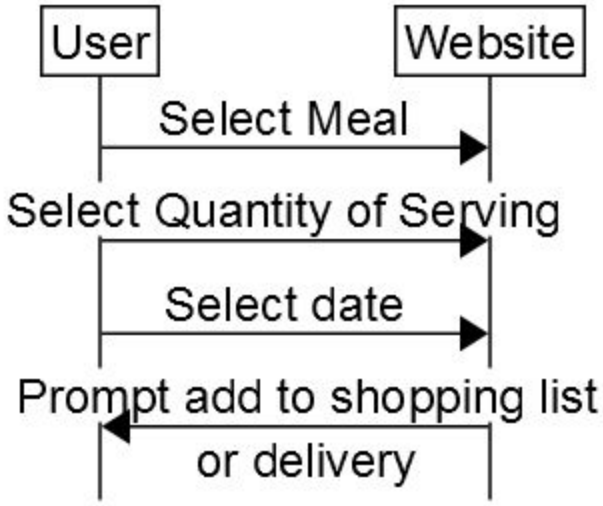
- The system shall query the database for account creation and return result within 60 seconds.
- The system shall update the database when a user subscribes to the service and return result within 120 seconds.
- The system shall query the database for curated meals and return result within 15 seconds.
- The system shall query the database for information about meals or snacks and return result within 10 seconds.
- The system shall allow users to add an item to their cart within 10 seconds.
- The system shall query the database for order completion and payment processing and return result within 60 seconds.
- The system shall query the database for order history and return result within 30 seconds.
- The system shall allow users to add an item to their favorites list within 5 seconds.
- The system shall query the database for information about user favorites and return result within 5 seconds.
- The system shall query the database for filtered meals or snacks and return result within 15 seconds.
- The system shall query third-party database for shipment status and return result within 15 seconds upon receipt of tracking information from third-party.

Data Flow Diagram

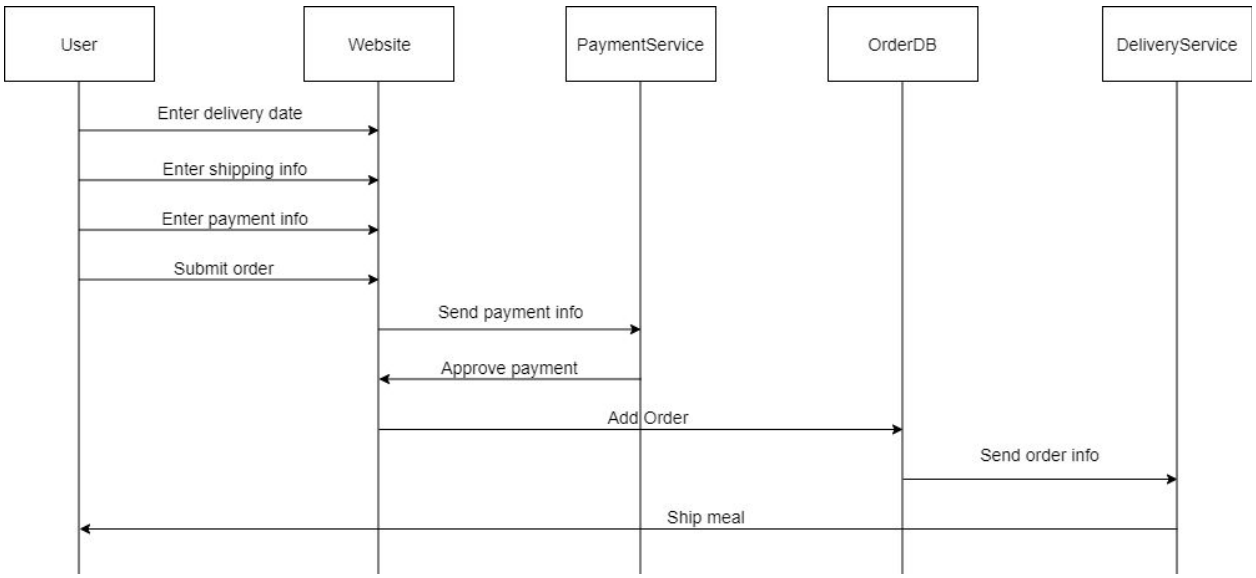


Message Sequence Charts

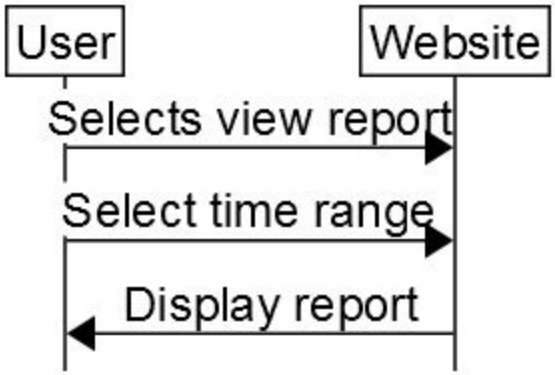
Use Case 1



Use Case 2



Use Case 3



Customer Meeting Summary

We weren't able to set up a face-to-face meeting, but were able to get a response back from the customer on Thursday and spoke briefly about requirements on Friday. We've invited him to our team Slack channel so we can collaborate easily going forward.

Team Member Contributions

Here is a list of team member contributions:
All - Communicated on Slack throughout the week and added contributions to Google doc
Brian Shim - Requirements, use case scenarios, UML diagram
Raymond Lieu - Dataflow diagram and all three use case message sequence charts
John, NianJun, Raymond and Todd - Finalized requirements and use cases text during group meeting