

Bagel Bar Mobile



Team Members:

- Nur Jahed
- Tyler Kurzawa
- Stephen Magrowski
- Nicholas Razzi

Highlights:

Introduction 2
Analysis & Requirements 3
High Level Design 4
Detailed Design 5
User Interface Design 6
Final Release 7

Introduction

The Bagel Bar Mobile project is a web application designed for the Bagel Bar Café in Kutztown, PA. Now more than ever restaurants are expected to include some form of takeout. The Bagel Bar Mobile app achieves this for customers using an iOS or Android device. React Native was used to create an interactive mobile app for viewing the menu and placing an order for pickup. Other features include creating an account, tracking the status of

an order, and favoriting a menu item. These orders can then be viewed using the employee browser version. The status of an order can be updated, as well as the estimated time until it is ready. Using a separate account, the owner can add or remove menu items from the database. The owner can also manage the standard employee accounts. Both the app and browser components use a central SQL database to function properly.



Team Structure

All four team members had different assigned roles so that a real-world work environment could be simulated. The project leader was Stephen Magrowski, who was tasked with making major project decisions and delegating work. Stephen also held a role as a developer and completed many of the mobile app features.

Tyler Kurzawa was the second developer and created many of the browser version features. The database administrator role was also held by Tyler. Many of the

database tables were optimized and managed by him.

A third developer role was held by Nicholas Razzi who contributed to the browser version. The project manager role was also held by Nick, who scheduled and tracked the progress of the project.

Testing was the responsibility of Nur Jahed, who created test cases and scenarios for the project features. Jahed also used automated testing with Appium to thoroughly check the project requirements.

Development Tools

During this course, the team used many different programs and technologies to create the project. Communication was crucial throughout the entire project. The team made use of Discord and Slack to share ideas and have group discussions. File sharing tools such as GitHub and Google Drive were also used frequently. Whether it was a technical document or the project code, these tools allowed the team to access and edit the same files.

Designing the project was challenging and it was hard to picture how the requirements would be implemented. For this, the team used a program called Balsamiq Wireframes to create low-fidelity mock-ups. This tool assisted the developers by visualizing the layout and theme of the different screens.

Implementation required a wide range of tools, including text editors, programming languages, and frameworks. The primary languages used were JavaScript, PHP, and SQL. A framework and set of libraries called React Native was used for building the mobile app. To run and host the app, a command line tool called Expo-CLI was used. This allowed the app to be viewed and tested on actual devices. The Android Studio emulator was also used to test the app on simulated devices. Visual Studio Code and Notepad++ were used for editing the code.

Analysis & Requirement Gathering

During the analysis phase a comprehensive list of functional requirements was gathered. These requirements specified what functions the system must perform and the features to be included. Each requirement contained a unique ID for traceability. They were also placed into related categories and assigned a priority level. The priority classification for each requirement was based on how crucial the feature is for the project.

There was a total of 33 requirements gathered for the mobile app portion of the project. Only two out of the 33 requirements were not completed. These incomplete features include leaving a rating on menu items and push notifications for the order status. However, all the highest priority requirements were finished. The employee browser version had 22 requirements specified during the analysis phase. All 22 requirements for the browser version were completed.

There were also 6 non-functional requirements specified during the analysis phase. These requirements involve important characteristics of the system, such as availability, capacity, reliability, security, and usability.

The Software Requirements Specification (SRS) document lists all the requirements that were referenced here. It can be viewed in the technical documents folder on D2L. An overview of the requirements for the mobile app is listed below.

Overview of Requirements

Account Management

- Create an account using an email address
- Sign into an existing account using credentials
- Sign out of the account
- Update account information such as the email address, name, and password

Restaurant Information

- View restaurant information
- GPS view of Bagel Bar location in relation to customer

Dashboard View

- View previous orders on the dashboard
- View favorite menu items on the dashboard

Menu

- View and sort the Bagel Bar menu by category
- Select and view a specific menu item – Price, toppings, and description
- Customize and select options for a menu item
- Add a menu item to the cart for checkout

View Cart

- View the cart with summary of order
- Edit or remove an item from the cart
- Save a favorite menu item
- Proceed to checkout for an order

Checkout

- View summary of order – Name, total amount due, and quantity of each item
- Add general notes to an order such as allergy information
- Add promotional codes for an order
- Remove an item from an order within the checkout screen
- Specify order timing – Place immediately or later time of choosing
- Place an order then receive a unique order number associated with the order

Order Placed

- Track the status of an order using the provided order number
- Cancel an order if less than 15 minutes have passed since the order has been placed

High Level Design

One of the first design diagrams created were the System Sequence Diagrams (SSD). These diagrams helped depict a particular use case or scenario for the system. The event triggers or interactions between the actor and the system are included. The events are shown in sequential order to indicate

each step that will be taken by the user to interact with the system. These diagrams were modeled in accordance with Unified Modeling Language (UML) notation 2.0. This means that the symbols and lines used follow a standard modeling language. Figure 1 below shows how a customer will view the checkout screen and their

order summary. Within the checkout screen, the customer will be shown the total amount due, items in their cart, and quantities of each. Here the customer can remove any items that are no longer wanted. The customer can also select the desired time for picking up the order. After the order is placed, a confirmation message is displayed with the order number.

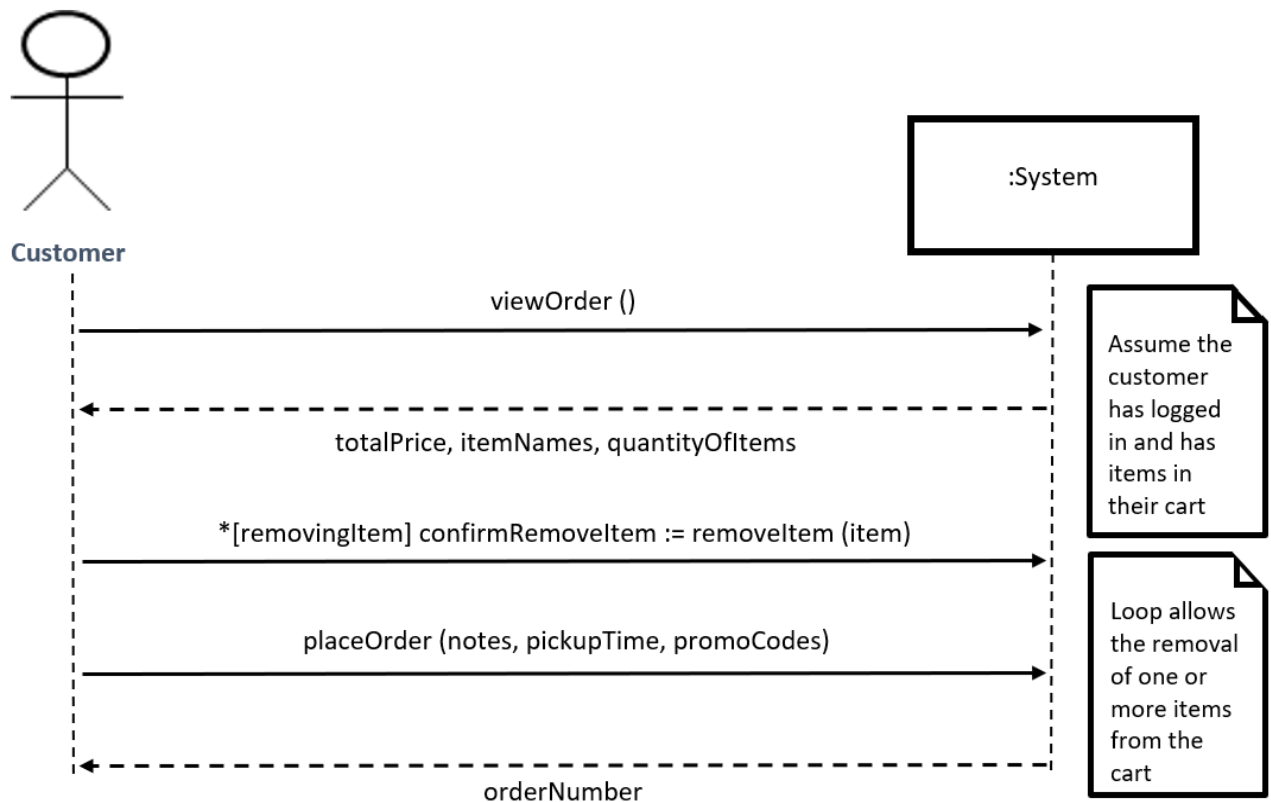


Figure 1: System Sequence Diagram for Checkout

Detailed Design

As the project progressed, more detail was added to the design diagrams. The Design Class Diagram below was especially helpful for organizing the different groups or objects in the system. A Design Class Diagram provides an overview of the target system by describing the objects and classes within the system, as well as the relationships between them.

This object-oriented analysis decomposes the system into real world concepts or objects. This is done by grouping objects into separate classes. The name of the class is shown along with another compartment underneath for the attributes of the class. These describe the object being modeled by specifying the type of data after the attribute name. Accessibility is also shown which specifies whether it is public or private

data. This is useful for ensuring that proper steps are taken to only show information to those who need it or who should have access to it. Showing that there exists some relationship between objects is described as an association, which can be unidirectional or bidirectional. For example, an aggregation relationship shows that the class can still exist without the subclass. Figure 2 below shows the Design Class Diagram for the Bagel Bar Mobile system.

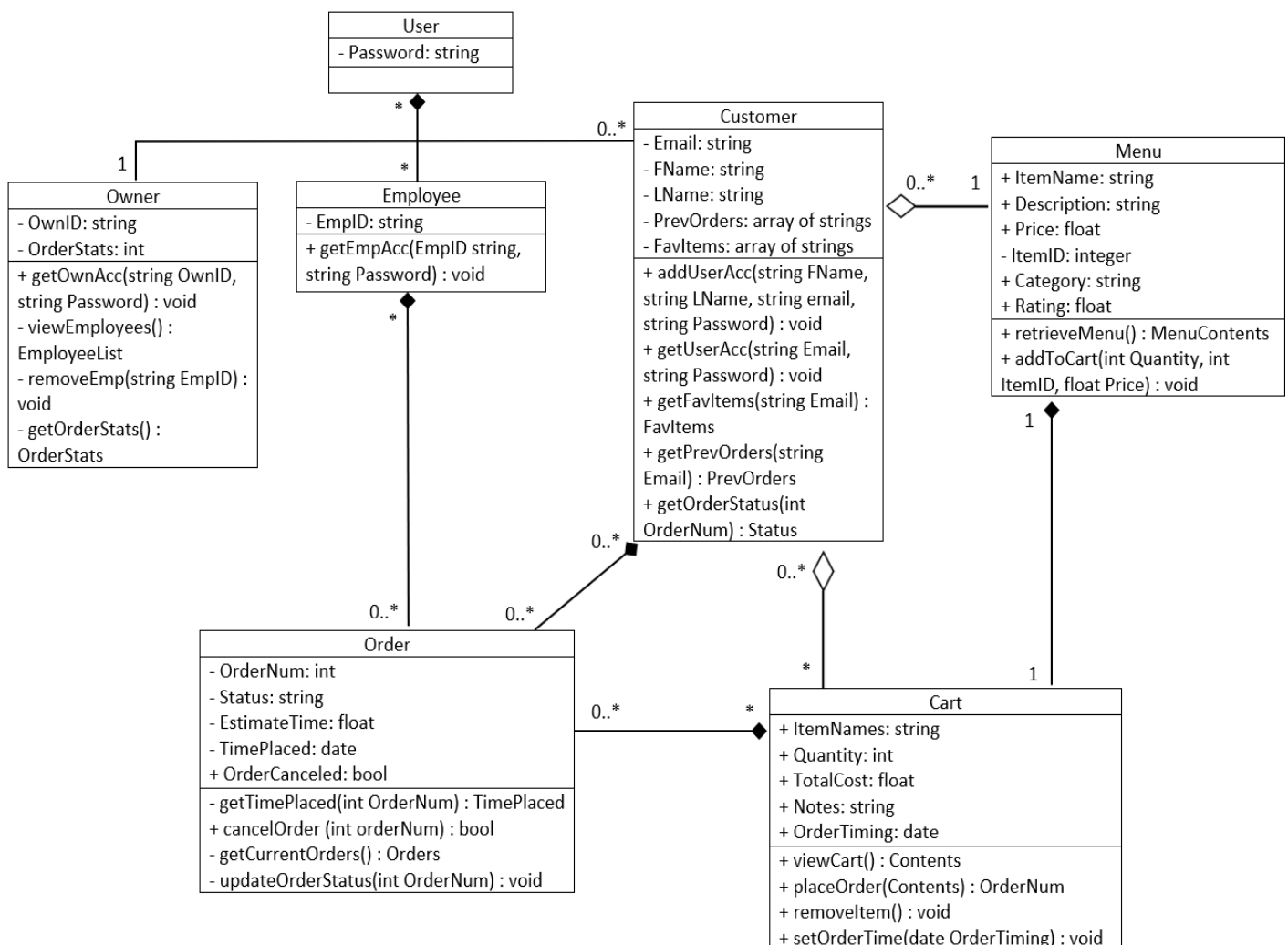


Figure 2: Design Class Diagram for Bagel Bar Mobile

User Interface Design

Another detailed design diagram was the User Interface Design Flowchart. A User Interface Flowchart displays how a user will navigate the user interface of a system. It is also used to show how the various components or screens are connected. The required steps or decisions between the components are shown as well. These decisions

are depicted as diamonds, while the actions or screens are displayed as rectangles.

After logging into their account, the customer will be directed to the home screen. Here, their previous orders and favorite menu items will be displayed if they exist. The customer can also view the menu to select a specific item. After customizing the item, there will be an option

to add it to their cart. Once the customer has placed an item in their cart, there will be an option to checkout. After placing the order, a unique order number will be generated. There will then be the option to check the status of any active orders for the customer. Figure 3 below shows the User Interface Flowchart for the Bagel Bar mobile app.

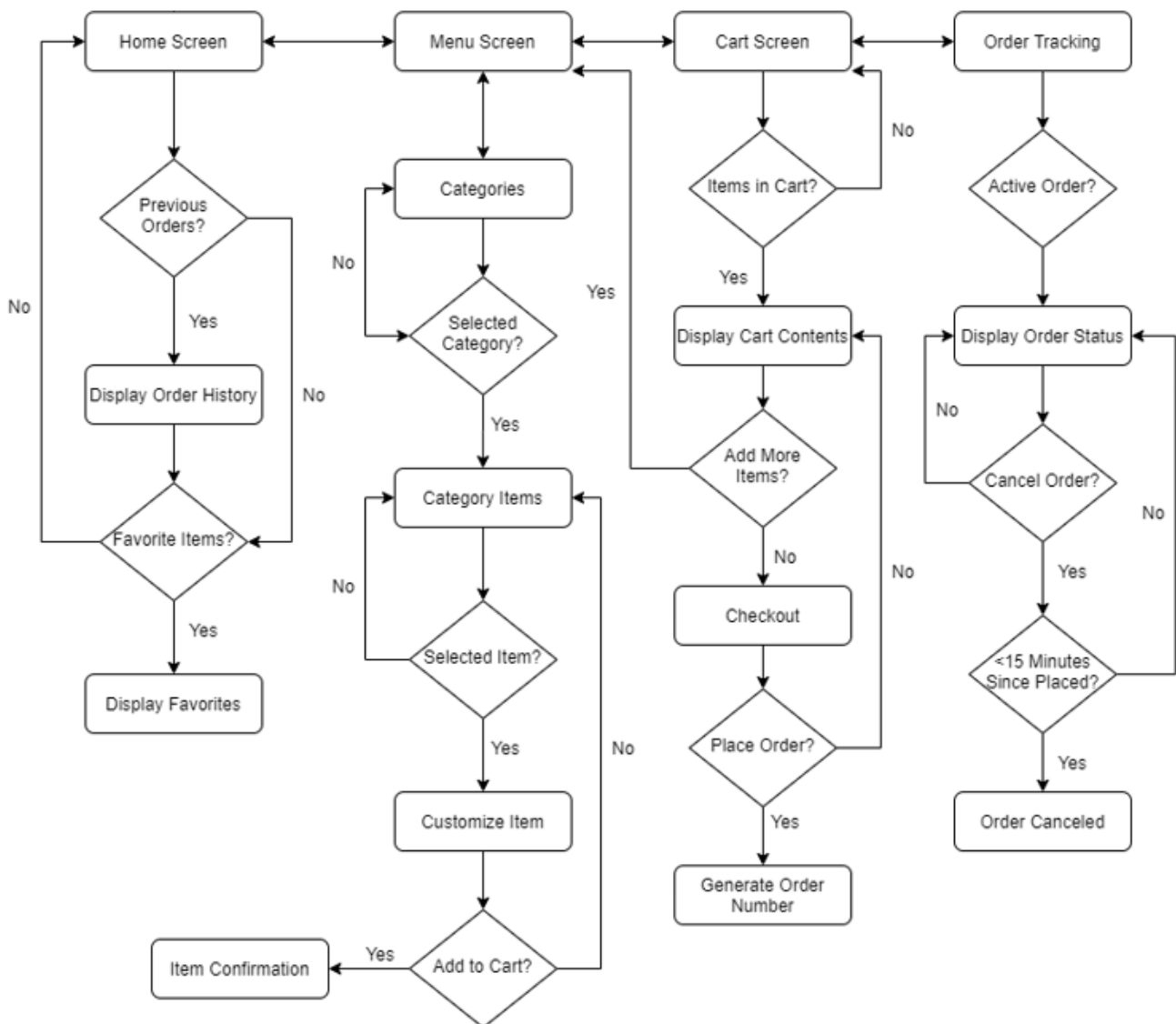


Figure 3: Bagel Bar Mobile App User Interface Flowchart

Final Release

During the Spring semester the Bagel Bar Mobile project was implemented. Using the documents created during the planning and design phases the team had a strong foundation upon which to develop the project. Implementing the project was done in four sprints, which separated the work to be done into two-week intervals. A third

week was allotted after each two-week period for testing and fixing the features that were completed. As time went on, new features were added to the completed portions. The project then developed into a final form that is ready for release to the client. Included below are images of the completed mobile app and browser version for employees.

×

Menu Item

Basic Egg & Cheese

\$3.59

Perfectly cooked eggs with melty cheese between your choice of 24 bagel varieties, croissants, or toast

Quantity

− 1 +

Special instructions

Add special requests here

Select Bagel

Choose one (Required)

Select bagel...

Extras

☐ Extra Cheese

☐ Extra Egg

☐ Add Cream Cheese

Add to Bag

Home

Menu

Bag

Info

×

Checkout

Pickup, ASAP (15-20 mins)

Change

Promo Code

Enter code here

Apply

1 Basic Egg & Cheese

\$3.59

Plain Bagel
Extra Cheese

Remove

Favorite

2 Buffalo Chicken Wrap

\$15.98

Whole Wheat Tortilla

Remove

Favorite

1 Caramel Mocha

\$3.89

Medium

Remove

Favorite

Sales tax:

\$1.41

Items subtotal:

\$23.46

Total:

\$24.87

Home

THE BAGEL BAR CAFE


Welcome, Steve!

Favorites

Active Orders

Previous Orders

Explore the Menu



Home

Menu

Bag

Info

Place Order



[Home](#) [Add Menu Item](#) [Update Employees](#) [View Menu](#) [Change User Info](#) [Log Out](#)

Current Orders

Order	Total Due	Time ordered	Time of Pickup	Date ordered	Status	Update Status	Update Time
Order Number: 96053179 Email: Stev@g.com							
Item Name							
1 Basic Egg & Cheese \$3.59 Plain Bagel Extra Cheese	\$24.87	8:47 pm	9:07 pm	4-25-20	Pending	Update Status	Update Time
2 Buffalo Chicken Wrap \$7.99 Whole Wheat Tortilla							
1 Caramel Mocha \$3.89 Medium							