EECS 2311 - TEAM 7 - PlatePlan

Iteration 2 Wiki

Team Members

Name | Student ID
Farah Madkour | 219913219
Meem Morshed | 219476142
Ricky Nguyen | 219461201
Andrew Nong | 219661537
Pouya Sameni | 216491623

ITR2 User Stories

This is where we talk about what we each worked on, give an estimate of the effort, when it started and when it finished along with a one paragraph summary of why we did it.

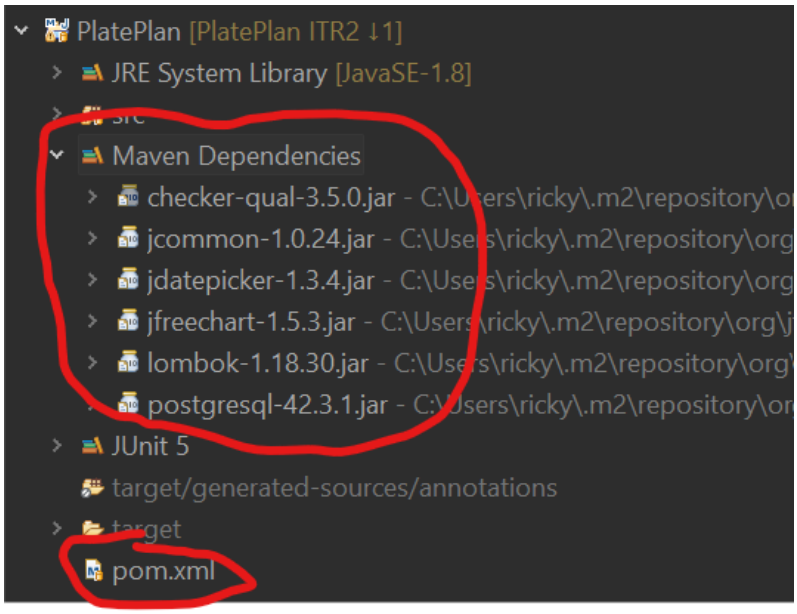
Story 1 (Ricky Nguyen) - This story enabled the viewing of feedback left by customers and deleting unwanted feedback. This story is estimated to take 3 days to be fully implemented. The story began its' development on March 1, 2024, and was finalized on March 3, 2024. As stated by our customer, they wanted an efficient way to be able to communicate with customers to apply possible changes to the business. In addition to this, business users can manage feedback for other users by simply deleting it. This will decrease the congestion of feedback and be helpful towards the business user.

Story 2 (Meem Morshed) - This feature enhances the efficiency of table management for restaurant owners. With the ability to combine tables, owners can optimize seating arrangements for larger groups or special events, maximizing the use of available space. Additionally, the ability to edit current tables provides flexibility in adjusting seating capacities or layouts based on changing needs. This feature streamlines the table management process, allowing owners to quickly and easily make adjustments to accommodate varying party sizes and improve overall operational efficiency.

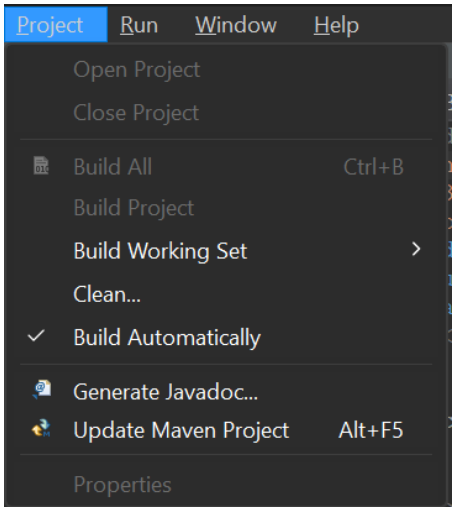
Technical Setup

Project Setup

To set up the project, we first want to import the maven project "PlatePlan". Then, you want to make sure all dependencies are available

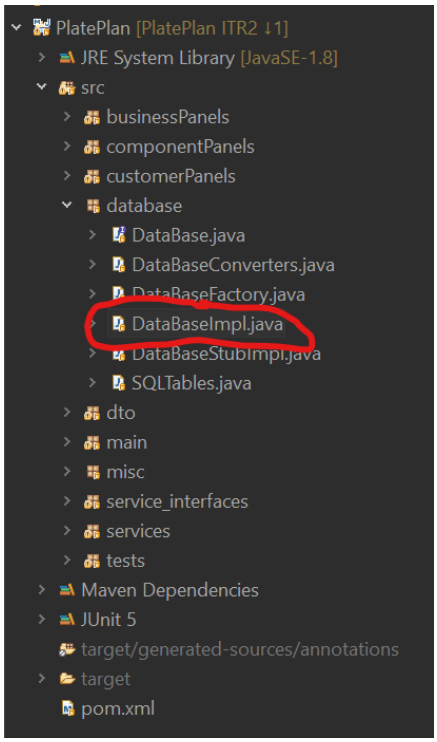


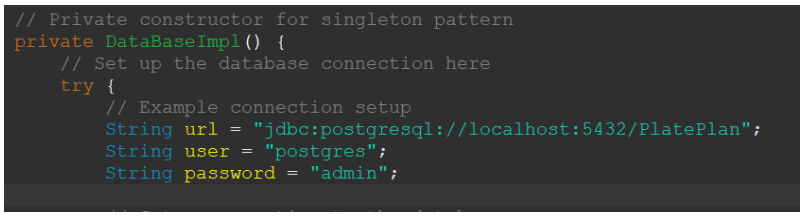
This can be done by simply adding the .jar files manually. An alternative would be to build the project through Maven and do the following:
First, clone the repository. Then, once the files are in your developing tool clean the project.



Database Setup

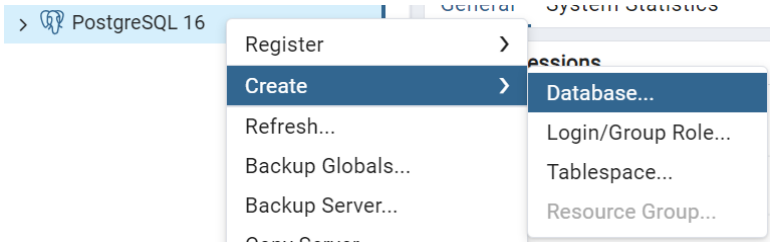
When setting up the database, please note that the username and password of the database can be found in DataBaseImpl.java (/PlatePlan/src/database/DataBaseImpl.java).

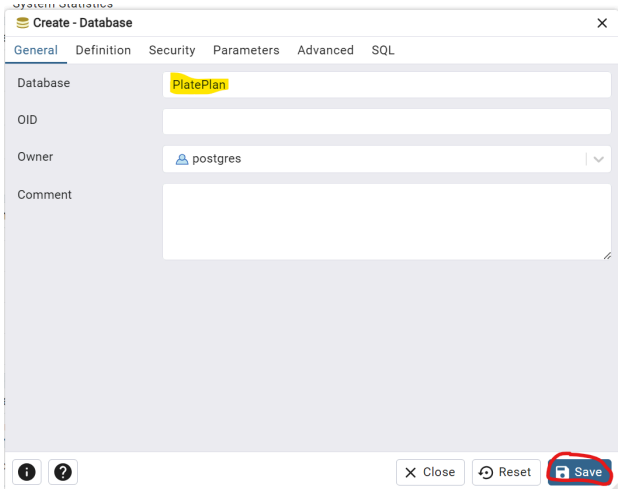




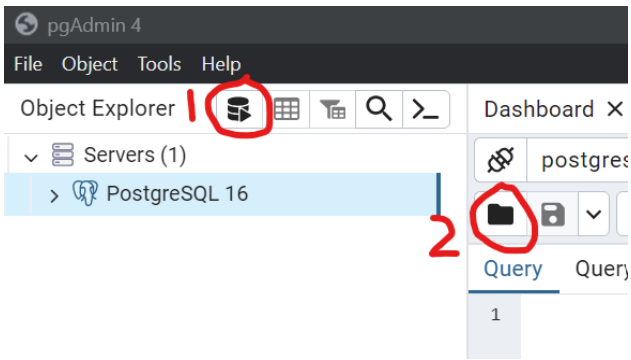
```
// Private constructor for singleton pattern
private DataBaseImpl() {
    // Set up the database connection here
    try {
        // Example connection setup
        String url = "jdbc:postgresql://localhost:5432/PlatePlan";
        String user = "postgres";
        String password = "admin";
```

When adding the database, right-click on PostgreSQL and create a new database. Please note that the database should be named “PlatePlan”.

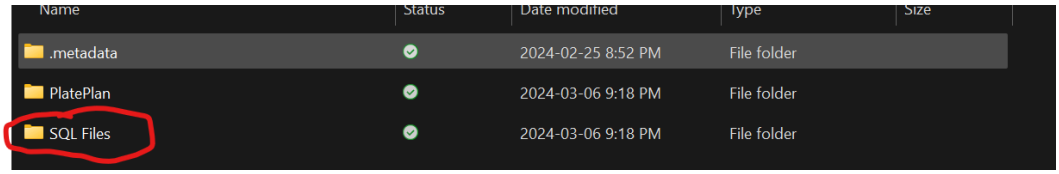




Click the Query Tool button. This will open up a window for you to upload the PlatePlanBackup.sql file.



Can’t find the PlatePlanBackup.sql file? It will be located under the address (/PlatePlan/SQL Files/PlatePlanBackup.sql).

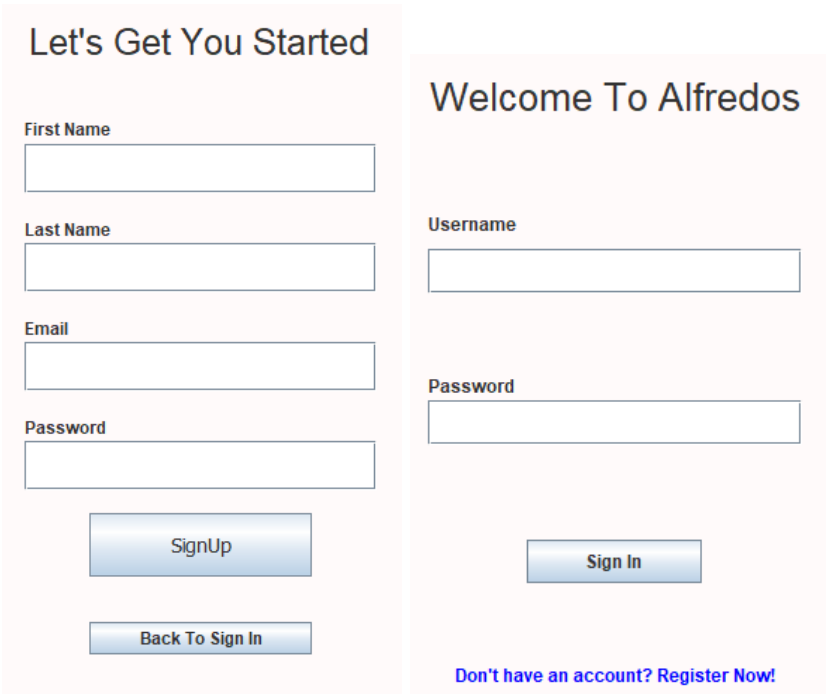


Finally, if required, update the username and password for Postgres as needed in the DataBaseImpl.java previously mentioned.

Accounts

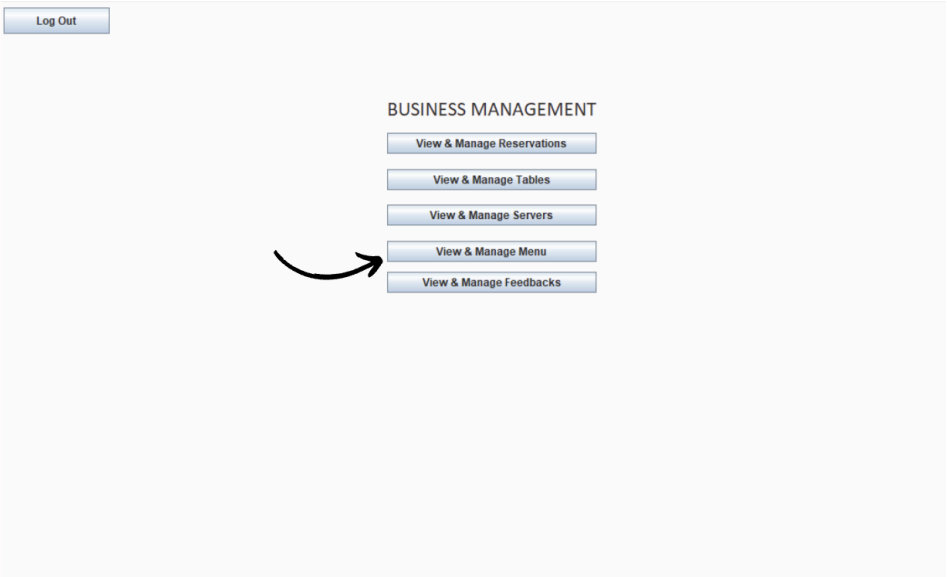
There are two types of accounts one can log in with. On the business side, you can log in utilizing the business username and password. They are *alfredo* and *password* respectively.

For customers, they can register through the registering panel. Once finished, they can enter their username and password to sign in.

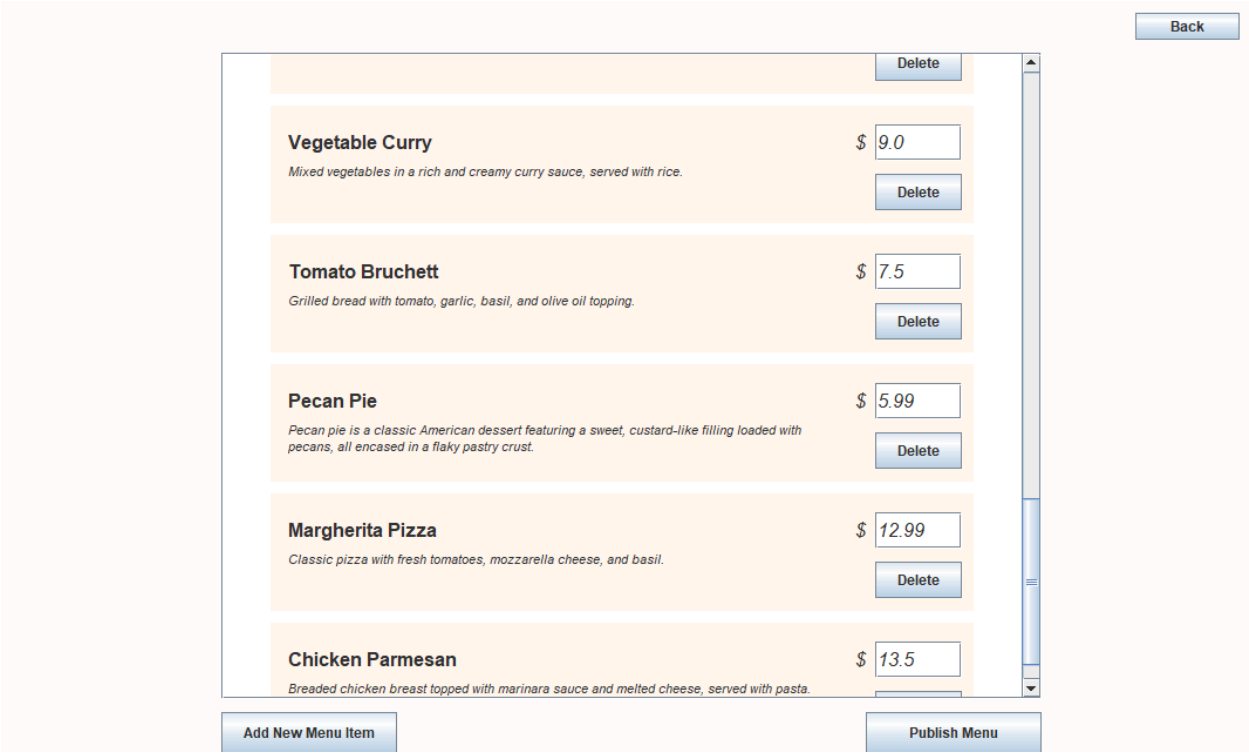


Use Cases

Using the menu as business

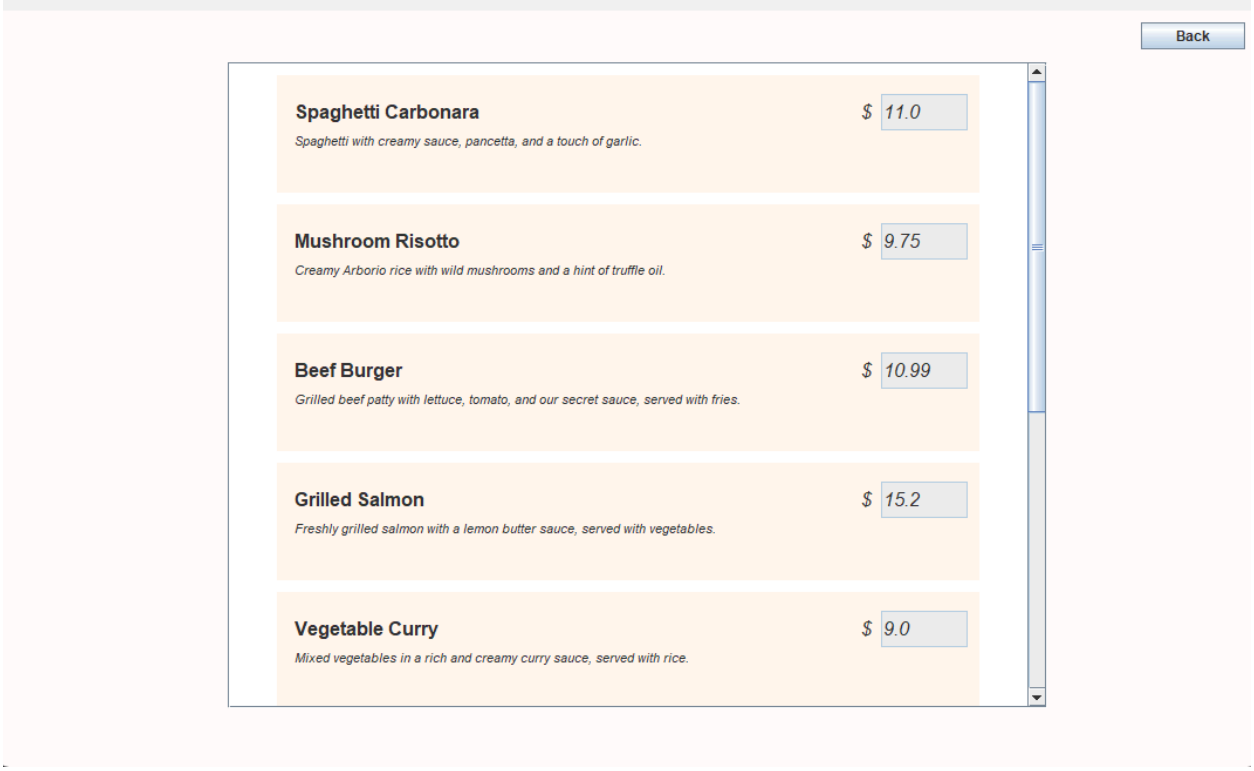


After logging in with the business account details, you have the option to use the menu from the business side.



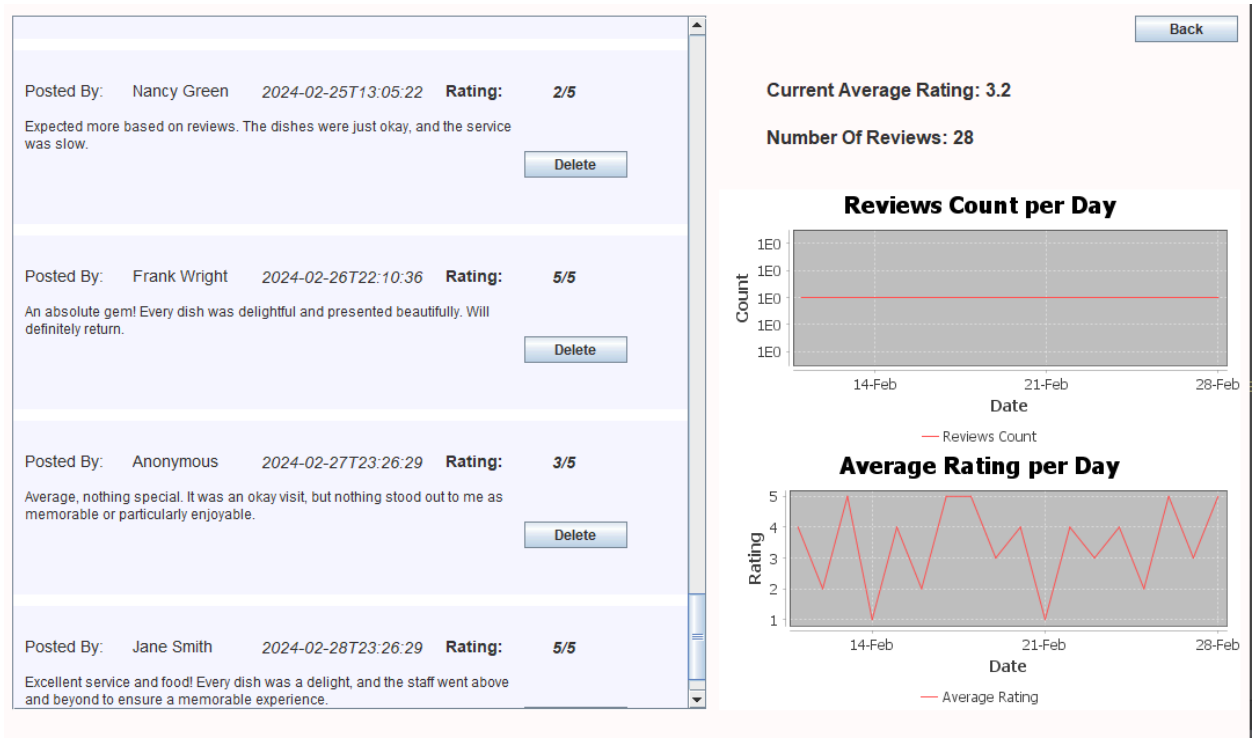
From here, you have the option to delete current menu items, or change the price of them. There are also buttons that allow the business to add new menu items. Once you're done, publish the menu.

Using the menu as customer



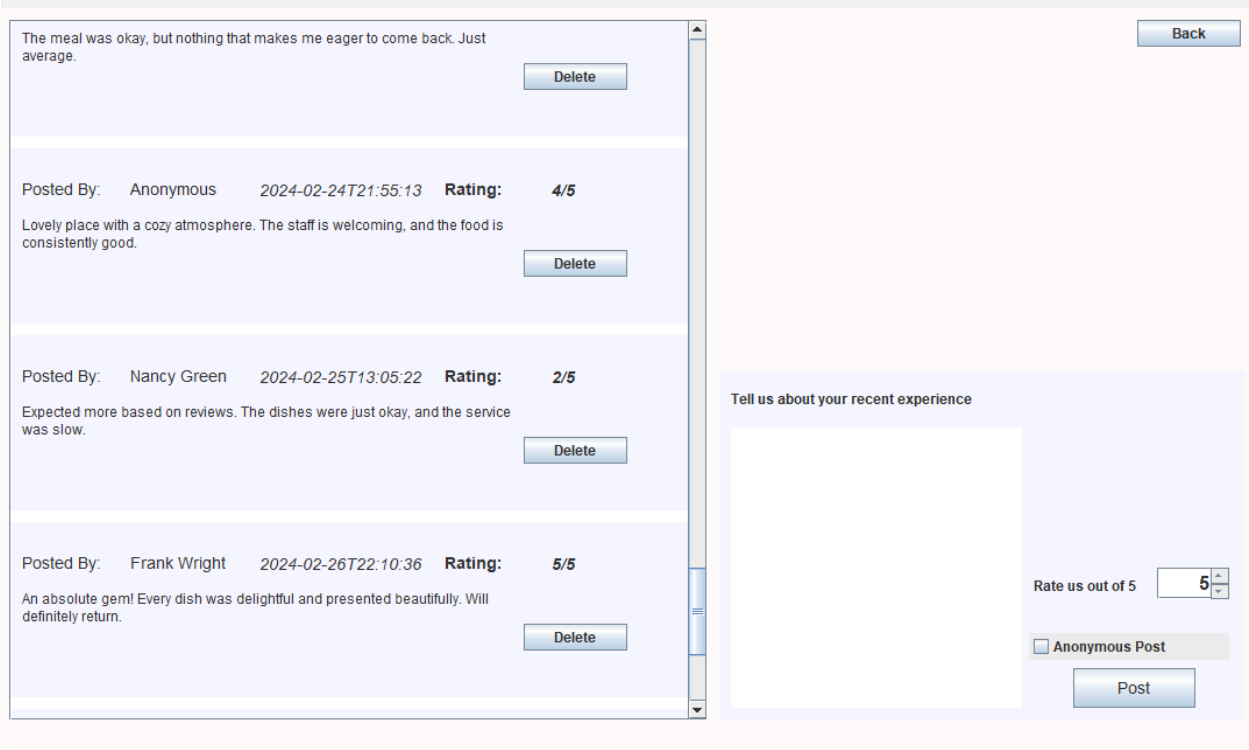
When using the menu as a customer, the user can view all of the available options, their descriptions, and their prices.

Using feedback as business



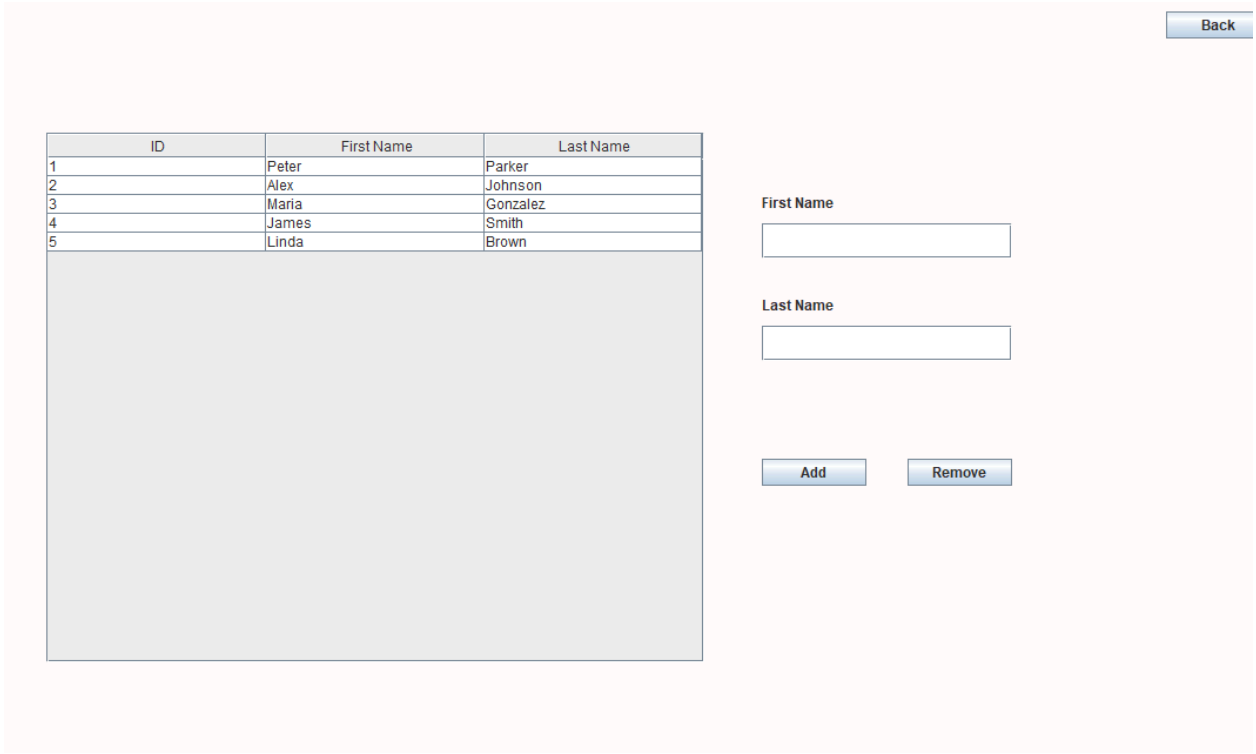
When using feedback as the business, you can view all of the customer reviews and ratings of their experience at the restaurant, with the option to delete any comments. On the right of the screen, there are two charts, one showing the review count per day, and the other showing the average rating per day.

Using feedback as customer



When using the feedback feature as a user, you have the option to view other peoples comments and ratings about the restaurant, as well as post your own review with a rating out of 5, with the option to post anonymously as well.

Using Server screen as business



When using the server screen from the business POV, you can view all of the current servers along with their IDs. Apart from that you can also add new servers, and remove current servers.

Using tables screen as business

Unit Testing

In our quality assurance process, we implemented unit tests to scrutinize both the services and business logic layers of our application. This testing approach utilized stub databases, facilitated by a toggle within our database factory settings. By adjusting this toggle to a "development" mode, our tests bypassed direct connections to PostgreSQL, instead leveraging these stub databases to simulate database interactions. This method enabled us to achieve an impressive 95% average coverage across all pertinent files, thoroughly examining both standard functionalities and potential edge cases to confirm the reliability and correctness of our codebase.

For integration testing, the database factory setting was switched to "production" mode. This adjustment allowed for focused testing on the actual database implementation methods. The primary objective of these integration tests was to verify seamless interaction between our application and the PostgreSQL database, ensuring no discrepancies in database behavior or issues with data integrity. Particular attention was paid to the functionality of converters responsible for transforming PostgreSQL query results into Java objects, validating their accuracy and effectiveness in a production-like environment.

Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
PlatePlan	34.0 %	5,230	10,144	15,374
src	34.0 %	5,230	10,144	15,374
businessPanels	0.0 %	0	3,975	3,975
customerPanels	0.0 %	0	2,708	2,708
componentPanels	0.0 %	0	2,123	2,123
dto	52.1 %	974	895	1,869
database	84.9 %	1,850	329	2,179
DataBaseImpl.java	76.3 %	721	224	945
DataBaseStubImpl.java	91.0 %	537	53	590
DataBaseConverters.java	86.5 %	294	46	340
DataBaseFactory.java	87.0 %	20	3	23
SQLTables.java	0.0 %	0	3	3
StubDataBaseRecords.java	100.0 %	278	0	278
main	46.5 %	67	77	144
tests	98.7 %	1,497	20	1,517
services	98.0 %	842	17	859
ReservationServiceImpl.java	95.1 %	135	7	142
FeedbackServiceImpl.java	96.9 %	127	4	131
MenuServiceImpl.java	96.7 %	59	2	61
ServerServiceImpl.java	97.6 %	83	2	85
TablesServiceImpl.java	99.3 %	296	2	298
AccountsServiceImpl.java	100.0 %	142	0	142

Core components and all features currently available

Login and Registration Account System

The program PlatePlan has an account system that can store account information including passwords, emails, names, and usernames. The system will then only allow users with the right credentials to login to access the rest of the user features. If the user does not have an account setup then they can easily register by inputting a first name, last name, email, and password.

Reservation System

The system allows the customer to set a reservation, to reserve a table the following information is needed. The Date of reservation, number of people, the available timeslot and any special notes and/or allergies. On the business side the business can see any upcoming reservations which will show the date and time slot of the day, but also automatically assign a server and table to the customer.

Rating and Analytics

The customer receives the average rating of the restaurant, and can provide ratings as a message and stars and can post the rating anonymously as a public rating, or displaying their name. On the business side of the reviews and rating system, the business can see all the ratings that have been posted and read the messages. There are also graphs to display information such as average rating per day as a graph and a value to see past and current trends, reviews per day to see how many reviews are posted in a graph to see past and current trends, and displays the total amount of reviews of all time.

Menu

For the customer the menu displays the items the restaurant serves, alongside with a description of the food item, and the pricing of said item. On the business side the menu can be edited such as removing food items, changing the price of the items and adding new items.

Table and Server System

The table and server system is used by the business side and allows the business to define tables that will be taken by a server. The table is defined by the capacity, ID and the waitstaff that will be attending the table. Tables can be added, removed and combined to allow for different seating arrangements. Servers are also defined in the system with first name and last name. Servers can also be removed and added as seen fit.