CEREAL KILLER

Maria Gloria Obono Ondo, Anay Rajesh Pampatwar, Brian Marcel Patrao, Vivek Murugan, Divya Shree

Software Engineering Systems, Northeastern University
{obonoondo.m, pampatwar.a, patrao.b, murugan.v, shree.d}@northeastern.edu

***Abstract*—**There is currently a great need for a tool to search cooking recipes based on ingredients. Current search engines do not provide this feature. Most of the recipe search results in current websites are not efficiently clustered based on relevance or categories resulting in a user getting lost in the huge search results presented. They also do not provide a video preparation of a recipe. Our project aims to combine the features like search based one main ingredient, video suggestions about the recipes, images of the recipes, and a weekly planner for a user to organize his/her meals based on the favorite recipes he/she has selected.

I. Problem Description

Cooking daily three times a day is almost equivalent to a full time job and it's hard for most of the people around the globe, especially students and working professionals to cook for themselves. Without any proper plan often students and working professionals cook for a day or two and then end up starving and eating fast food from outside daily which is not good for our health.

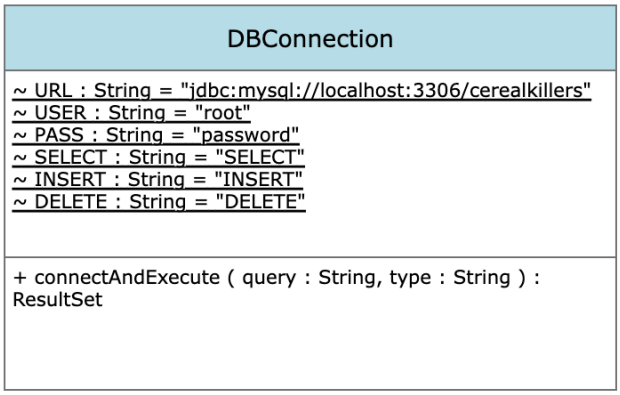
We usually also end up buying ingredients which we forget to use completely and it sits in the pantry until it gets spoiled.

II. Analysis (Related Work)

In our research we found that there is YouTube that has tons of recipes to make any dish. Then there are apps to plan your diet and there also exists apps like Healthify that gives us the nutritional equivalents for each food item. But there is no app designed specifically for students and working professionals to resolve their issues like maintaining a pantry, suggested recipes which can be filtered based on preparation time, that also gives you the option to add a list of your favorite recipes and curate a shopping list for you. That's why we decided to create an application that is a one stop for all these use cases.

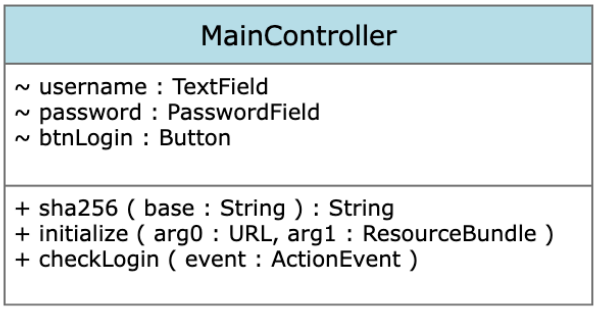
III. System Design

The system is designed according to the three user problems that we target to solve. These are pantry management, recipe suggestion and weekly planner. There is also a login/signup scene where the flow of code starts. So, each of these four scenes has individual scene controllers. The SQL connection class, which has the get Connection method is declared globally and used whenever there is a need for the connection string. JDBC Driver Jar file is included in the class path.



- Login Screen

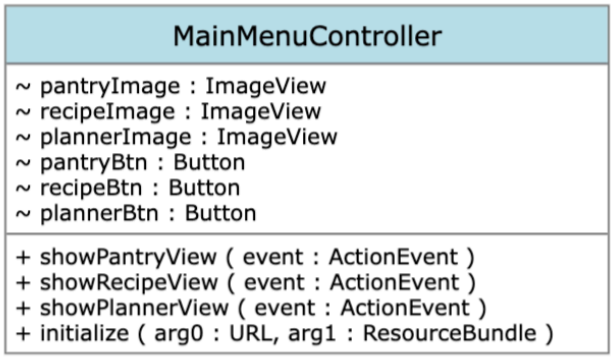
The login screen controller makes a get request to the database to authenticate the user. Once the user is authenticated, he is shown the navigation menu. Below is the UML class diagram.

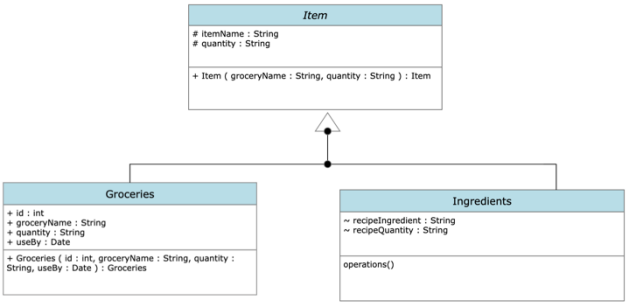


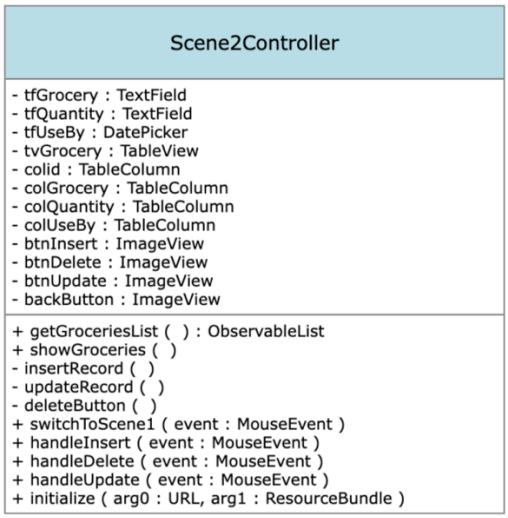
- Pantry view screen

Pantry screen allows the user to manipulate his groceries. CRUD operations are incorporated on the database using

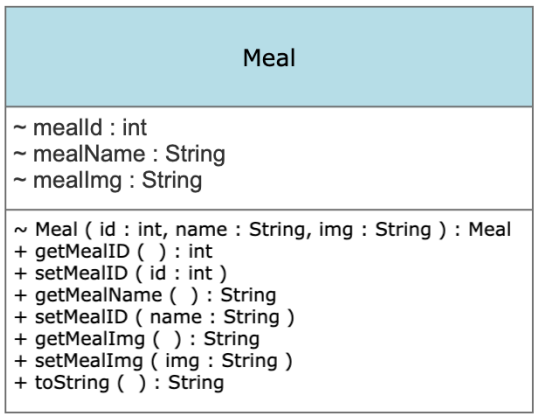
scene controller class. connectAndExecute() method is called from the global DBConnection class to get the connection string to connect to the database. Below is the UML class diagram. Event handlers are attached to the add, update, and delete buttons. Below is the UML class diagram.

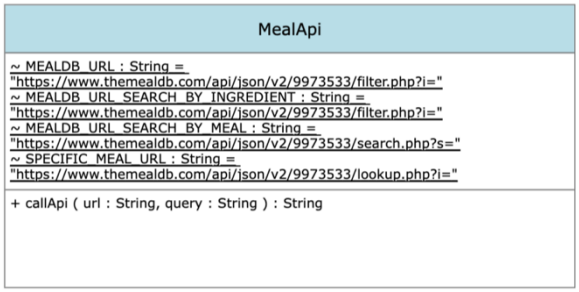


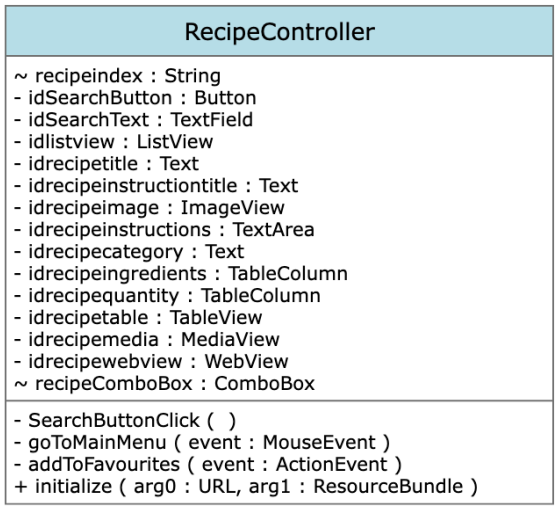




the Recipe screen controller with the help of JSON parser. Data retrieved from the API call is then stored in an array of recipe type objects. Below is the class diagram of recipe type. To show the video, media view and web view are used in the scene builder.







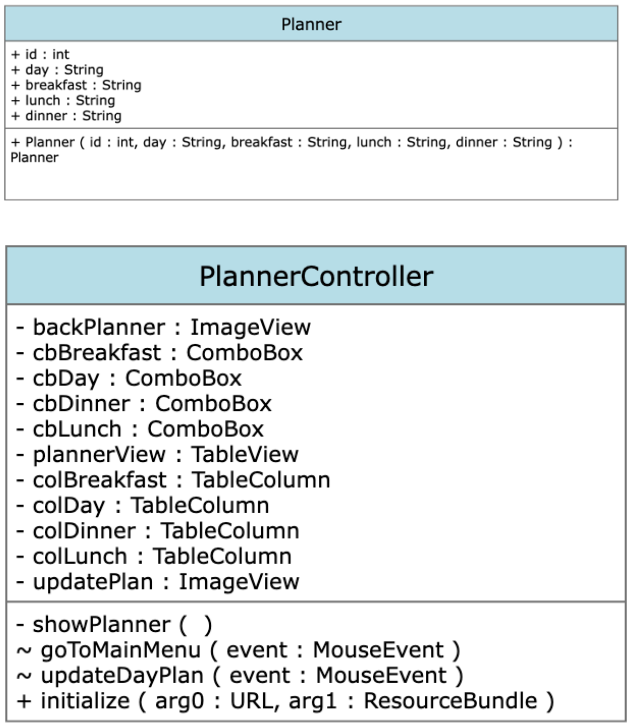
- Recipe suggestion screen

Recipe suggestion controller makes an API call to MealDB. Http request is used for the same. The JSON data is parsed in

- Weekly planner view

Table view is used to show the week plan of the user. Each day has three meals. The database is stored in seven rows

corresponding to seven days of week and has three columns to store breakfast, lunch, and dinner meals. Below is the UML diagram.



IV. Implementation

A. JavaFX Library

Because the JavaFX library is purely written as a Java API, JavaFX applications can reference any API from the Java library. That is, any application done in JavaFX can use any of the Java API libraries to access and communicate with native system capabilities. The JavaFX APIs are readily available as fully integrated JRE and JDK. They do not need separate installation but will need activation in the IDE to get started with developing JavaFX applications.

B. TheMealDB API

We used TheMealDB API to get our recipes. TheMealDB API is an open-source database of Recipes from around the world with great photos. The TheMealDB API is an open online database of food meal recipes that integrates into applications, web etc. This is a simple JSON API with features like high quality transparent PNG ingredients, meal image uploads, browsing meals in sequence and more. TheMealDB provides a free data source API for recipes online, that originated on the Kodi forums, as a cool way to browse recipes on your TV.

C. JDBC API

The interface for accessing relational databases from Java is *Java Database Connectivity (JDBC)*. Via JDBC you can create a connection to the database, issue database queries and update as well as receive the results. JDBC is one of the standard APIs for database connectivity, using it we can easily run our query, statement, and fetch data from the database.

D. MySQL Workbench

We used MySQL Workbench to get data in our application. MySQL Workbench is a unified cross-platform, open-source relational database design tool that adds functionality and ease to MySQL and SQL development work. MySQL Workbench provides data modeling, SQL development, and various administration tools for configuration. It also offers a graphical interface to work with the databases in a structured way.

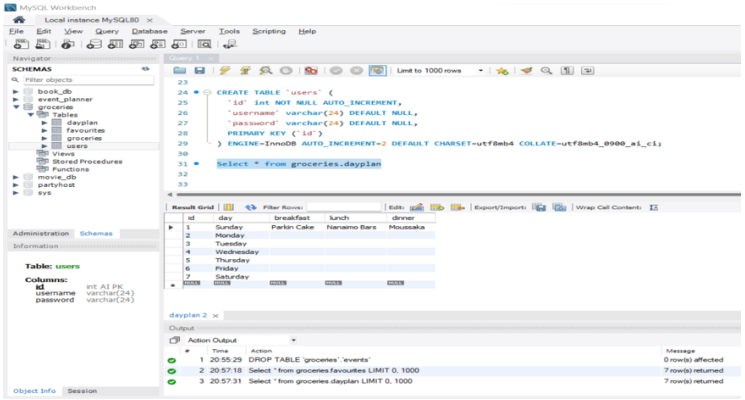
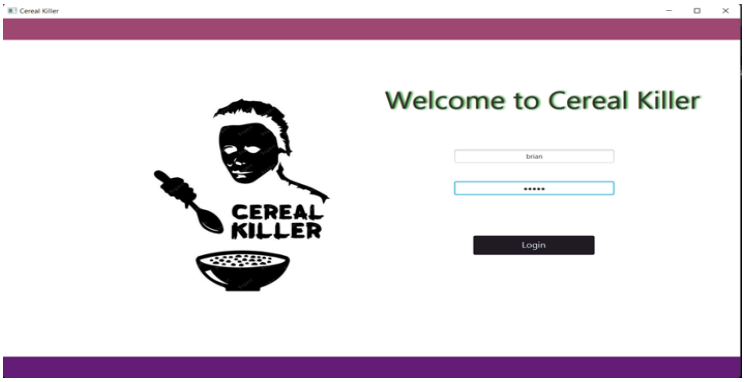


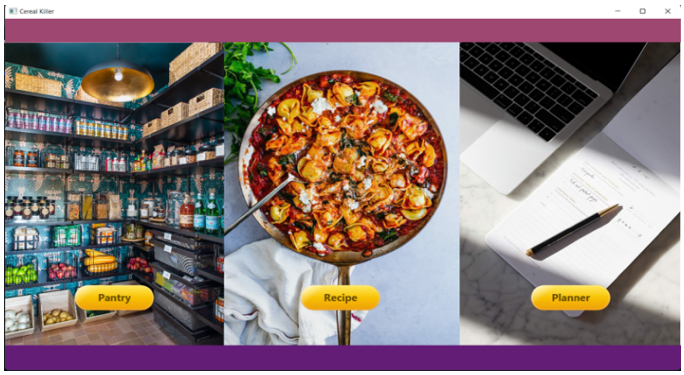
Figure 6. MySQL Workbench

V. Evaluation

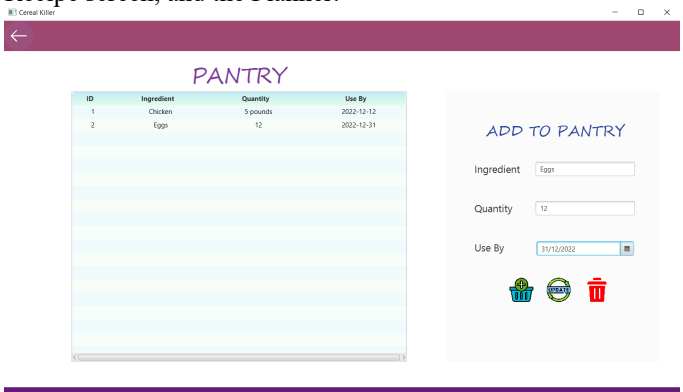
The Cereal Killers project aims to help users plan their meals in advance, potentially helping them to eat healthier, save money, and reduce food waste.



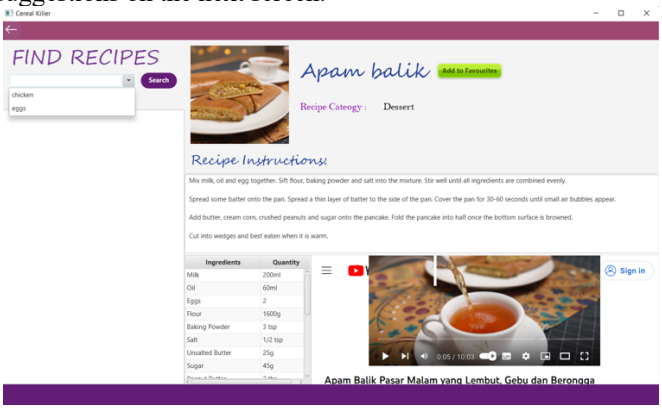
It begins with a login screen for users to store their information. The passwords are saved using a sha256 hashing algorithm to secure their data.



There are 3 main screens, namely the Pantry screen, the Recipe screen, and the Planner.

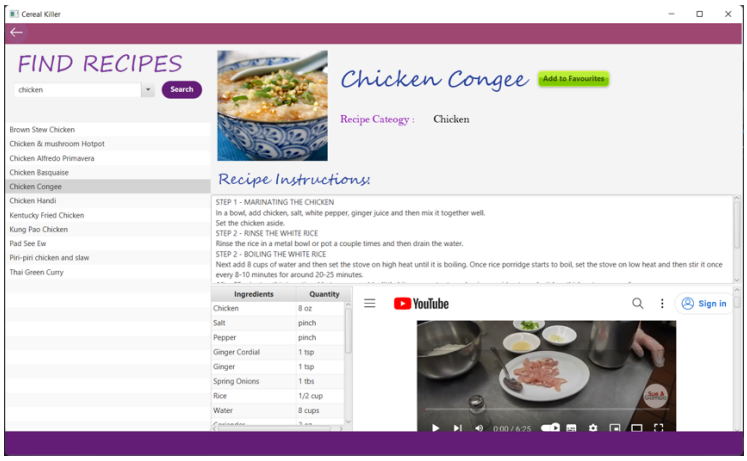


The pantry view stores the items currently in one's pantry. It has 3 fields namely the ingredient, quantity of each ingredient, and its expiration (use by) date. This is used to provide recipe suggestions on the next screen.

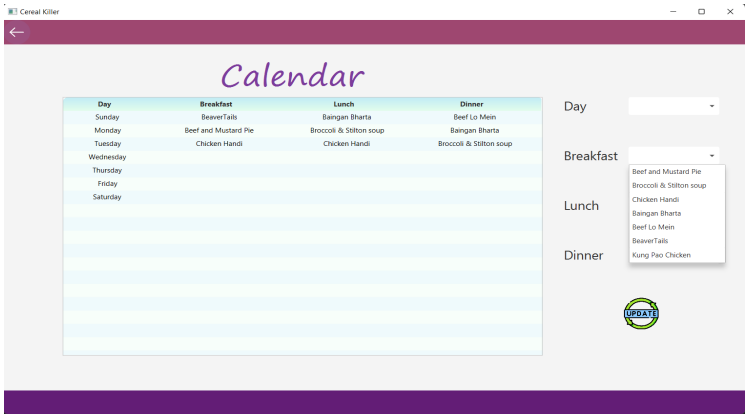


The recipe screen allows you to select items in your pantry as the main ingredient and search for recipes accordingly. It provides a default recipe on page load.

We can also search for any other recipe by typing the main ingredient in the search bar and clicking the search button as seen above



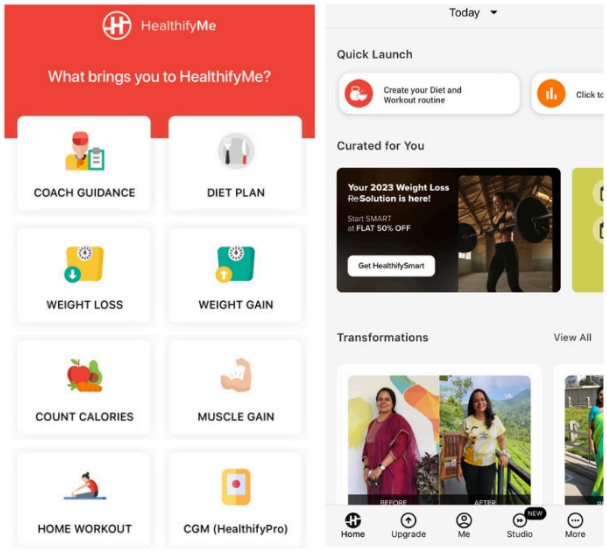
We can choose a recipe based on the suggestions. On choosing one's preferred meal, it shows us the instructions to prepare that meal. It also provides information about the ingredients required and the quantity needed. In case of any difficulty, a YouTube video is also provided so that the user can watch and see exactly how the meal is prepared. We can then add these meals to our favorites. This is used to plan our meals for the week in the next screen.



The last view is the calendar/planner view where we can add meals for the weak and prepare in advance. It shows you options based on the recipes added to favorites.

VII. Discussion (Reflection)

The main goal of our project is to create an application that helps its user to consume healthier home-prepared food. Other similar apps that are there in the market usually have several ads to even access this basic need and, in many cases, these ads are focused on people who want to lose or gain weight. These ads also ask people for subscription and contact dietitians. Most of the audience does not need that, they just need an organizer that helps them to plan well so after using this software stops using it because of irrelevant ads and notifications.



Our application does not have any scope of ads and notifications would be helpful to the user like setting up a weekly reminder to set up their calendar or in future scope give notifications about where they could shop from to get the best deals.

VIII. Conclusions and Future Work

We completed the first version of our product with a basic architecture.

- A user can set up this application and start using it as a planner and recipe searcher and maintain their pantry record.
- During our project demo professor suggested us to think more about the environment in which this app will be user so we have thought of adding a text to speech converter that will read out the instructions for food prep.
- Wherever users are adding form input there we will try to add audio option to it so that users have both options of typing as well as adding audio input.
- If our team had more time, we would have added an automatic shopping list that would get created based on missing items in the pantry and the meals planned for the week. We would have also added list of auto generated recipes based on the ingredients in the pantry and added filters on various factors like preparation time, cuisine, calories etc.
- Add more API sources so that the user can tailor the recipes to his own liking.

IX. Job Assignment

- Anay Pampatwar: UML class diagram for main, Scene controllers for pantry and recipe suggestion screen, database design for recipe, connecting all the scenes together.
- Divya Shree: ImageView implementation, UI design, databases, login and signup page layout design and controller class.
- Vivek Murugan: Recipe suggestion scene controller, web view and media view, database design for recipes.
- Maria Gloria Obono Ondo: Calendar View FXML design, created Scene controller class and helped to collaborate and schedule team calls and meets.
- Brian Marcel Patrao: Parsing the JSON data from API response, rendering the data to the recipe scene, Recipe suggestion scene controller, managing Trello board and setting up GitHub, code cleanup and created UML class diagrams.

References

[1] GmbH, V. 2. 2. V. L.-. (n.d.). *MySQL and Java JDBC - Tutorial*.https://www.vogella.com/tutorials/MySQLJava/article.html

[2] *Free Meal API | TheMealDB.com*. (n.d.). https://www.themealdb.com/api.php

[3] *MySQL Connection GUI Design using JavaFX and Scene Builder*. (n.d.). Engineering Education (EngEd) Program | Section.https://www.section.io/engineering-education/mysql-connection-graphical-user-interface-design-using-javafx-and-scene-builder/