CMPS 312 Mobile App Development
QuickMart App
Lab Assignment 4
Deadline – Thursday, November 28, 2024 11:59PM

Objective

The objective of this assignment is for you to practice Floor database library to write, read and query records from the SQLite database.

Preparation

1. Sync the Lab GitHub repo and copy the **Assignment/Assignment4** folder into your repository.
2. Open the project **QuickMart App** in Visual Studio Code. The project has the baseline code of the previous assignment.
3. The code for Providers, Navigation, and Widgets has already been implemented for you. However, the code for the Floor database is not yet completed. Your task is to implement the missing code for the Floor database.

Overview

For this assignment, you are required to implement the Floor Database to achieve functionalities like the baseline application provided. The application should allow users to edit, update
e, delete, filter products, cart items, and favorite items. The app should retain all data even when closed.

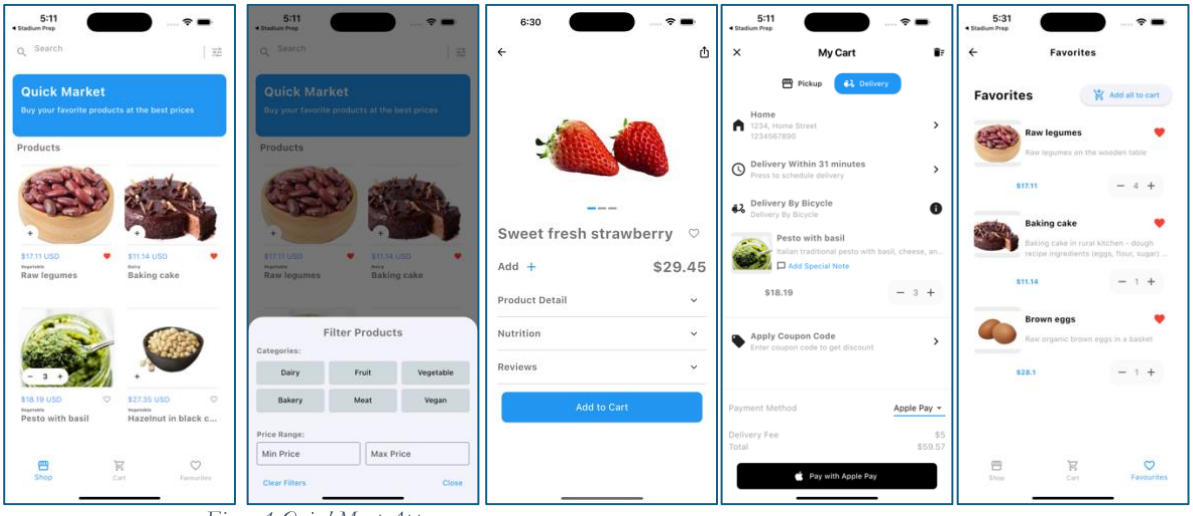

Figure 1 QuixkMart App

Implementation Instructions

1. Create the required Entity classes, **including any missing ones** from the provided base solution, as illustrated in the figure 2 below. Ensure that you implement cascade delete and update functionality while enforcing data integrity through the use of foreign keys.

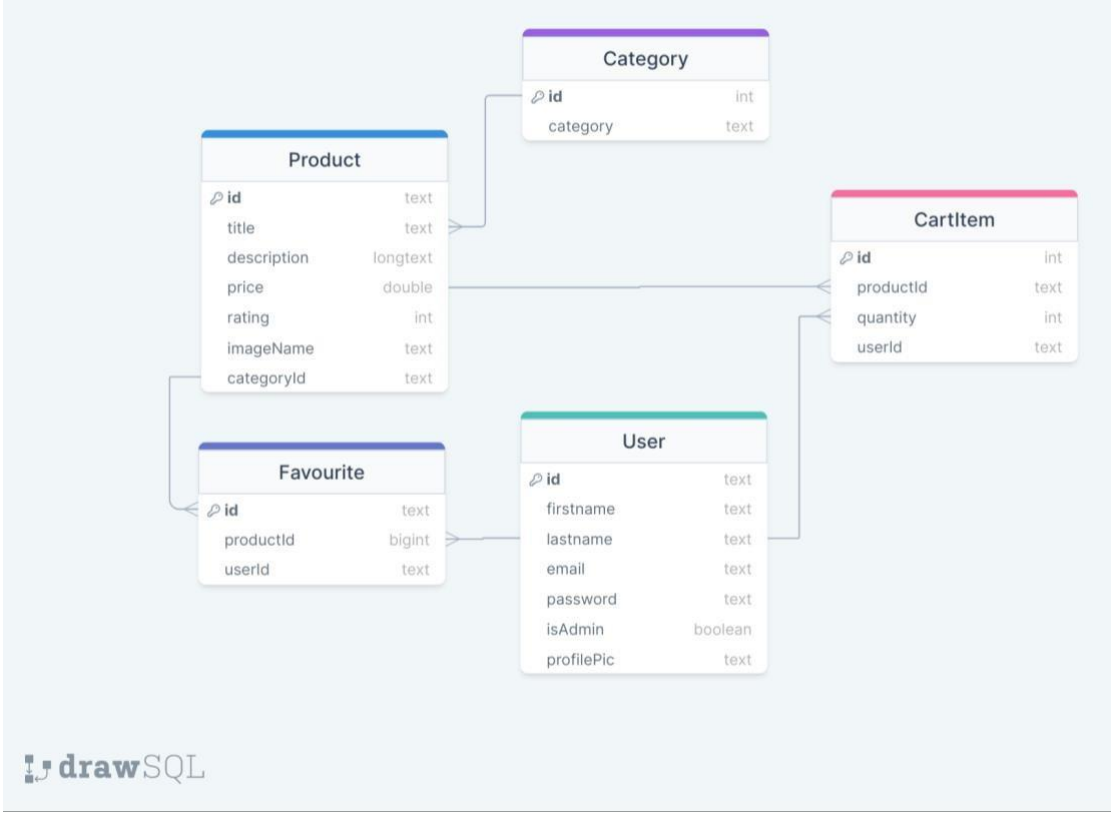

Figure 2 : QMart Entity Relationshop Diagram

2. Create Data Access Objects (DAO) Interface and the Database classes.
3. Inside the DAO Interfaces create **all the necessary functions** and annotations that allow the user to do all the CRUD operations required for the application to work, such as filter products by category , add product to cart item, add product to favorites, delete from favorites, change rating of a product etc..
4. Implement the QuickMart Repository with all necessary methods.
5. Establish a connection between the repository and the view model, ensuring the seamless functionality of the application. Resolve any issues within the providers to integrate your repository implementation effectively.

Important

- Ensure that all queries are implemented inside the database and do not rely on Dart language filters, finds, etc.
- Each student is expected to have a unique app that accomplishes the assigned tasks.
- You are free to use any theme, font, color, or icons of your choice.
- **Feel free to reuse code from your previous assignments**.
- Discussions with colleagues are allowed, **but the use of AI tools and sharing your code** is strictly prohibited. Plagiarism will result in zero grades for all parties involved.

Submit the testing sheet as well as the code under "**your_repository/assignments/assignment4**"