

CMPS 312 Mobile Application Development

Lab 5: Flutter UI Components and Layouts

Lab Overview

In this lab, you will delve into **Flutter's UI development**, focusing on components and layouts. You will build a simple Waiter Tipping and banking app UIs, learning how to use various Flutter widgets to create visually appealing user interfaces.

Lab Objectives

By the end of this lab, you should be able to:

- **Set up and configure a basic Flutter project** using **MaterialApp**.
- **Utilize key Flutter widgets** to build a UI, including:
 - **Scaffold** for structure.
 - **AppBar** for navigation controls and actions.
 - **BottomNavigationBar** for bottom menu navigation.
 - **Center**, **Text**, and **Icon** for content display and alignment.
- **Implement layout widgets** such as:
 - **Row** and **Column** for horizontal and vertical layout.
 - **Expanded** to adjust widget sizing within layouts.
 - **Card** for grouped information display.
 - **Switch** for interactive toggles.
 - **Container** and **Padding** for detailed spacing and alignment.
- **Incorporate images** with **Image.asset** for local assets and **Image.network** for online sources.
- **Enhance UI appearance** with styling, theming (**ThemeData**), and Material Design components.

Lab Overview

This lab is divided into two main parts:

1. **Part A - Waiter Tipping App:** You will develop a simple application to calculate tips for a waiter, learning to utilize basic Flutter's widgets
2. **Part B - Simple Banking App UI:** You will design and implement the user interface for a basic banking application, learning to utilize Flutter's widgets for creating professional UIs.

Part A - Waiter Tipping App

You will develop a simple waiter tipping application's user interface using Flutter. This part focuses on using basic widgets to create a simple UI.

1) Create a New Flutter Project

- Open your terminal or command prompt.
- Run the command to create a new Flutter project named `banking_app` **flutter create waiter-app**.

- 2) **Open the Project** Navigate to the project directory and open it in your preferred IDE (e.g., VS Code, Android Studio).
- 3) **Run the Default App**
 - Ensure you have a simulator/emulator or a physical device connected
 - Run the app to verify that everything is set up correctly.
- 4) **Explore the Project Structure**
 - Familiarize yourself with key files and directories:
 - `lib/main.dart`: Entry point of the application.
 - `pubspec.yaml`: Manages project assets and dependencies.
 - `android`, `ios`: Platform-specific code.
- 5) In `lib/main.dart`, set up a basic Scaffold with an `AppBar` and a body section.
- 6) Implement the following design shown in figure 1.

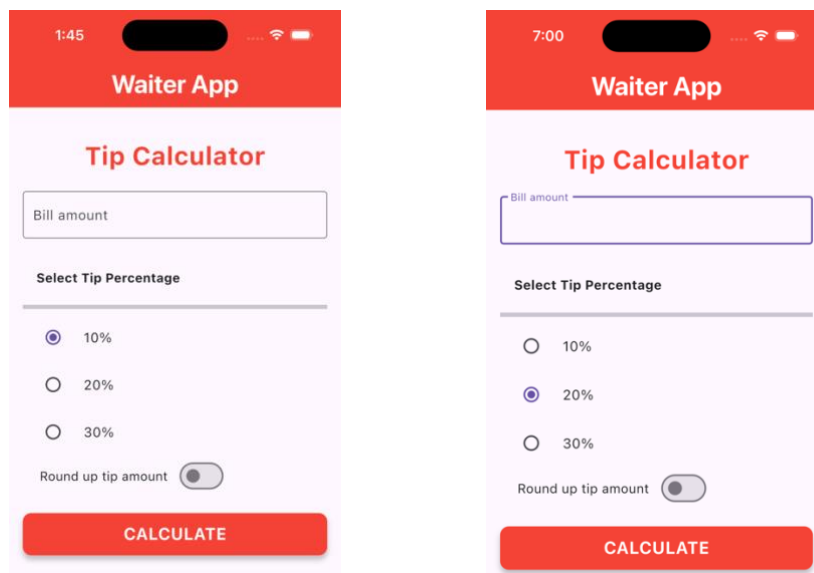


Figure 1 Waiter App

- 7) Run the app on a simulator or physical device to verify functionality.

Part B - Banking App

You are tasked with designing a basic user interface for a banking app using Flutter. The app will showcase customer information and interactive buttons for various actions. In this section of the lab, you will explore Flutter's structure and learn how to build a straightforward UI utilizing fundamental widgets such as `Scaffold`, `Text`, `TextField`, `Column`, `ElevatedButton`, `Image`, and essential layout widgets to organize your content efficiently.

- 1) **Create a New Flutter Project**
 - Open your terminal or command prompt.
 - Run the command to create a new Flutter project named `banking_app` `flutter create banking_app`.

- 2) **Open the Project** Navigate to the project directory and open it in your preferred IDE (e.g., VS Code, Android Studio).
- 3) **Run the Default App**
- 4) In `lib/main.dart`, design the following app

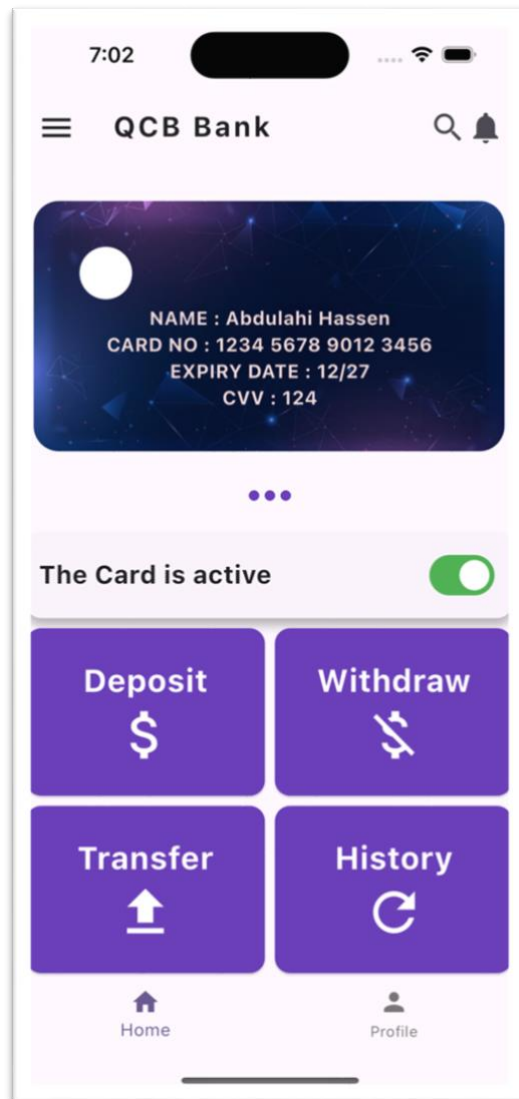


Figure 2 Banking App

- 5) To enhance the modularity and maintainability of your Flutter application, split the code inside the main.dart into several files, each representing a logical component of the application. Except the main.dart put the others inside components folder
 - **main.dart**: Main application entry point.
 - **app_bar.dart**: Contains the **AppBar** Widget.
 - **home_page.dart**: Houses the **HomePage** widget.
 - **credit_card.dart**: Manages the **CreditCard** widget.
 - **bottom_nav.dart**: Defines the **BottomNavigationBar** widget.

- 6) Extract all the constants such as text style into a separate file called constants.
- 7) Test your code
- 8) Push your project containing to your repository.