**Diagram, logo

Description automatically generated**

**MEHRAN UNIVERSITY**

**OF ENGINEERING & TECHNOLOGY**

**JAMSHORO, PAKISTAN**

**Project Report**

**Subject: Mobile Application Development**

**Project titled Recipe Book App**

**Roll No:22SW003**

**Section 03**

1. Real World Problem Description:

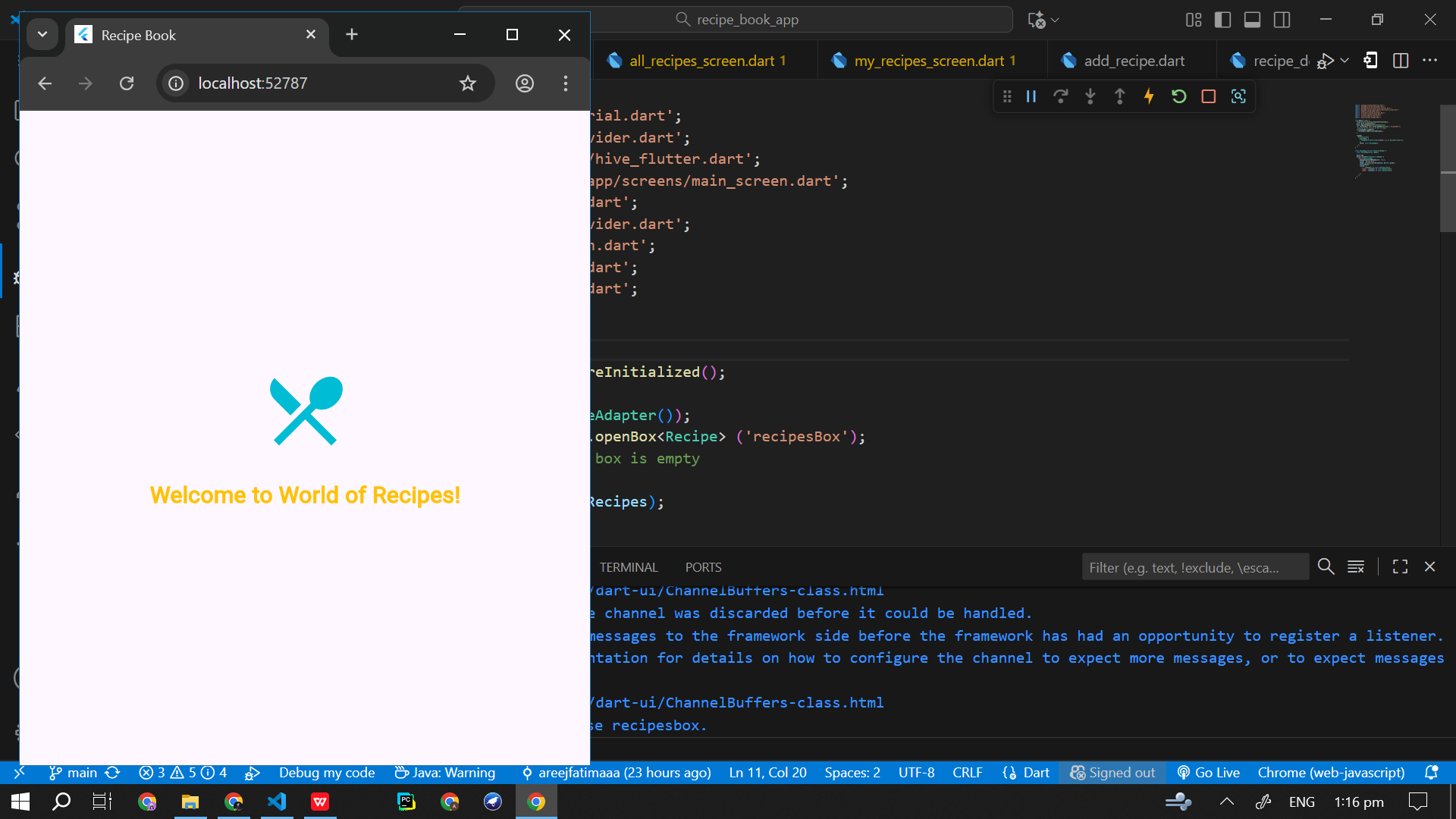
Cooking is a daily activity, yet many people face difficulty in organizing their recipes. Some write them in notebooks, other save screenshots or watch youtube videos, which leads to confusion or recipe not being found easily. Moreover, many apps only offer predefined recipes and do not allow users to store their own creations.

The Recipe Book App is designed to solve this issue by providing user a digital space to save and edit, and view both their own recipes and the ones they might want to try. It helps in managing ingredients, steps, and images in one place, making cooking more organized, personalized, and enjoyable.

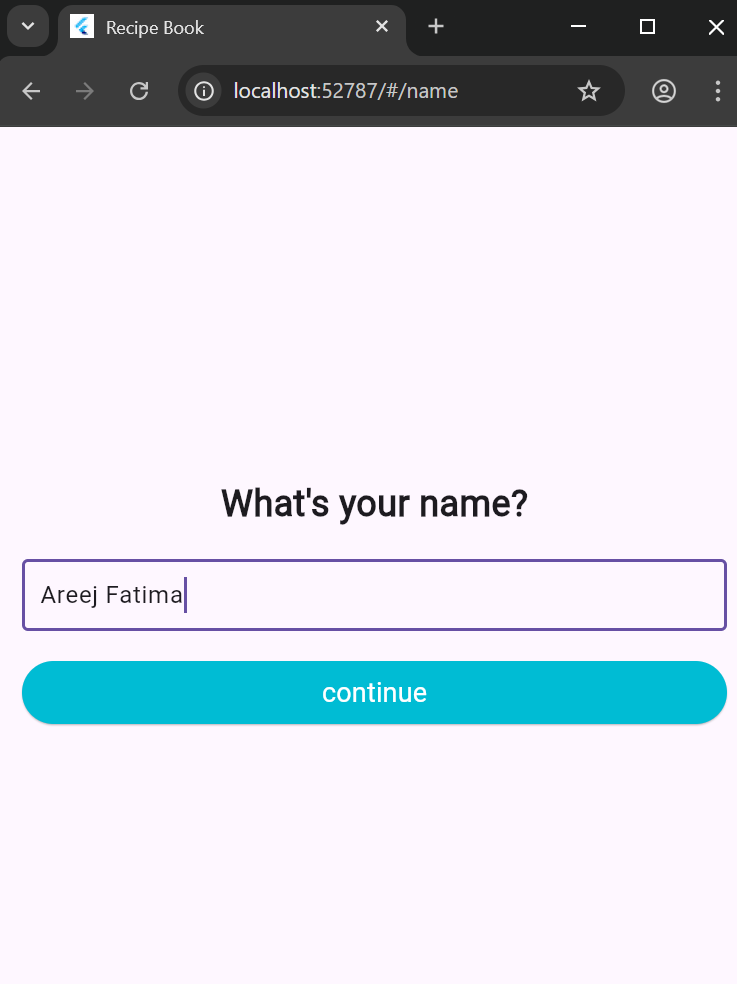
1. **Solution:**

The solution is a **Recipe Book App** developed using **Flutter**. The app allows users to **add, view, and manage recipes** with details such as ingredients, steps, and preparation time. It provides an easy way to store and organize recipes that users want to try or have already made.The main goal is to create a **simple, responsive, and user-friendly interface** where all recipes are stored locally using the **Hive database**, ensuring quick access without an internet connection.

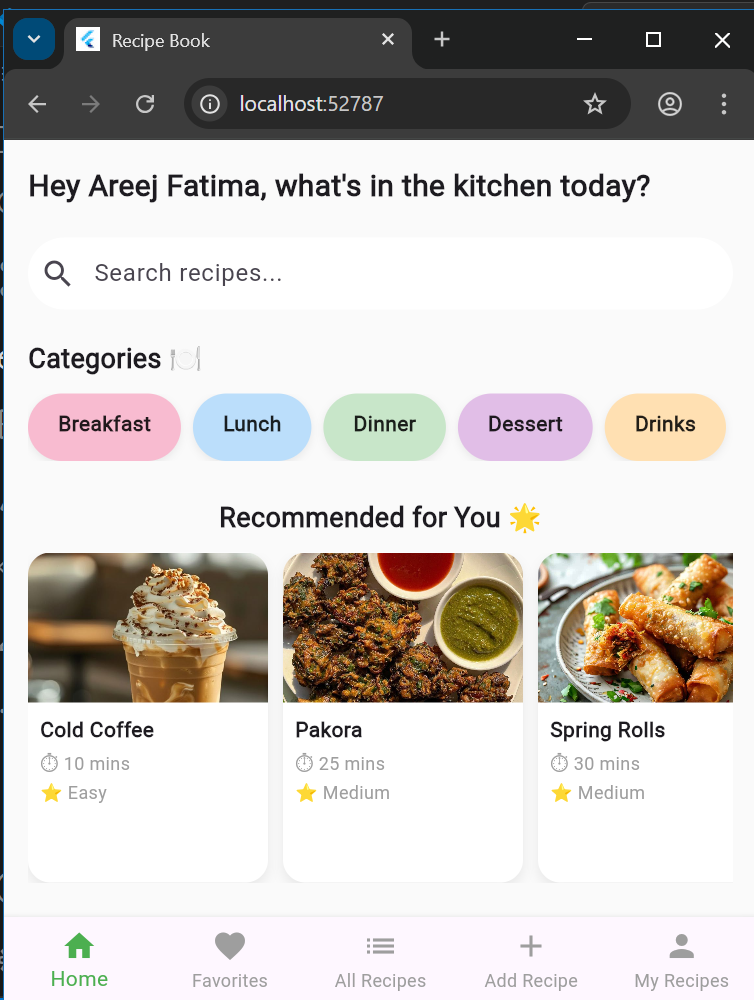
1. **User Interfaces**
2. Splash Screen:

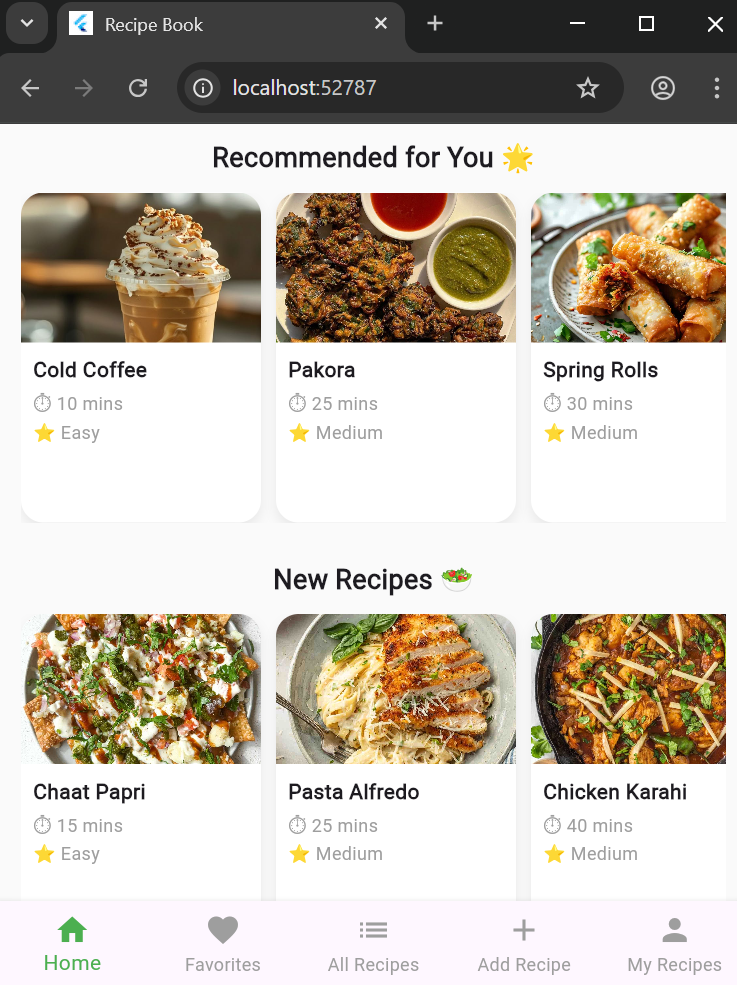


1. Name Screen

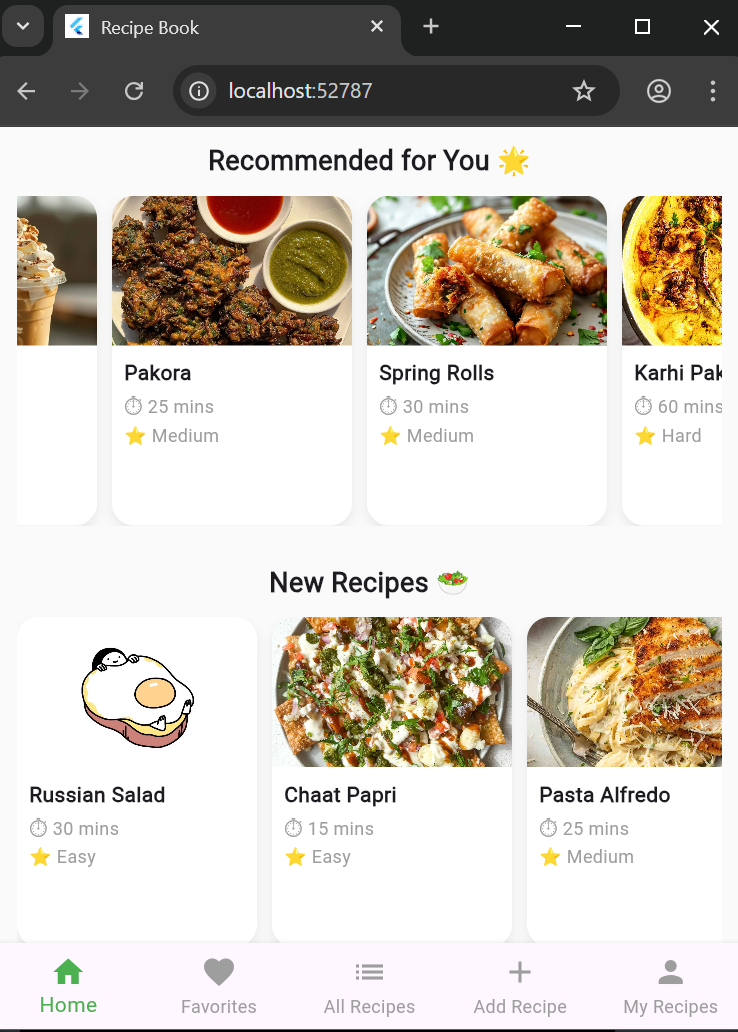


1. Home Page

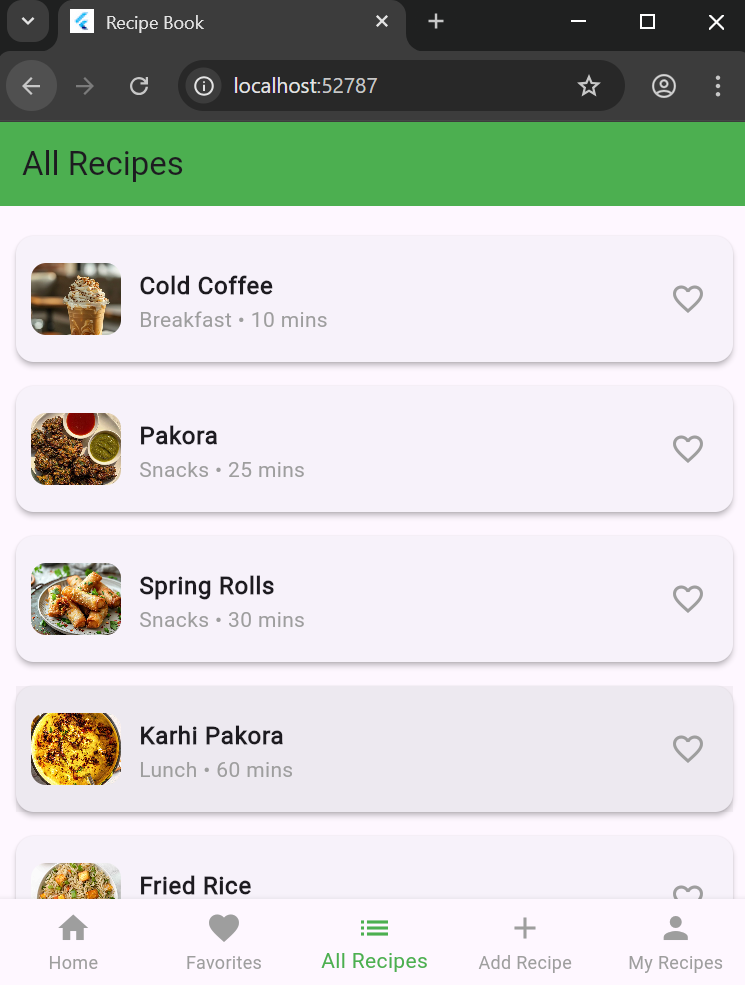


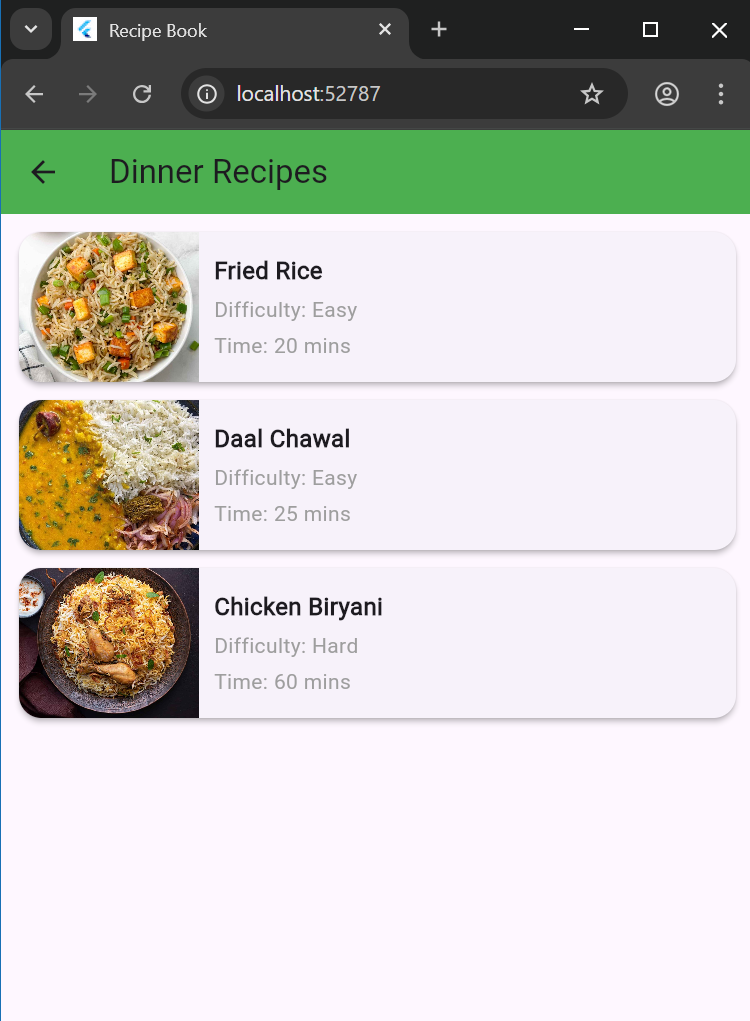


1. New recipe added to homem page looks like this:

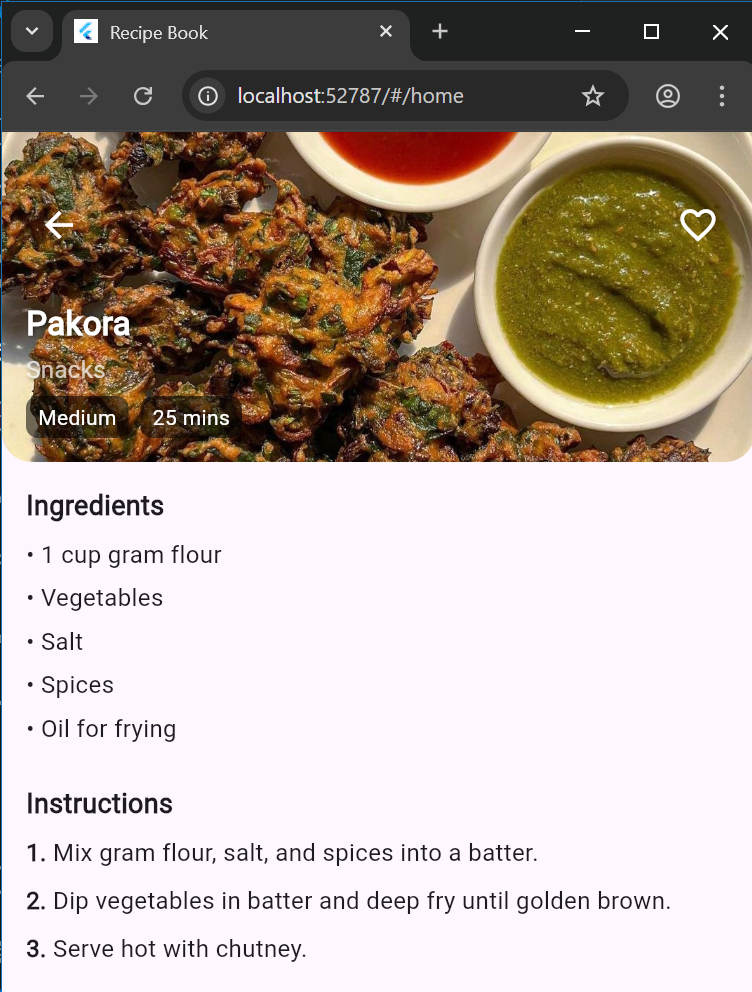


1. All recipes and category screen :

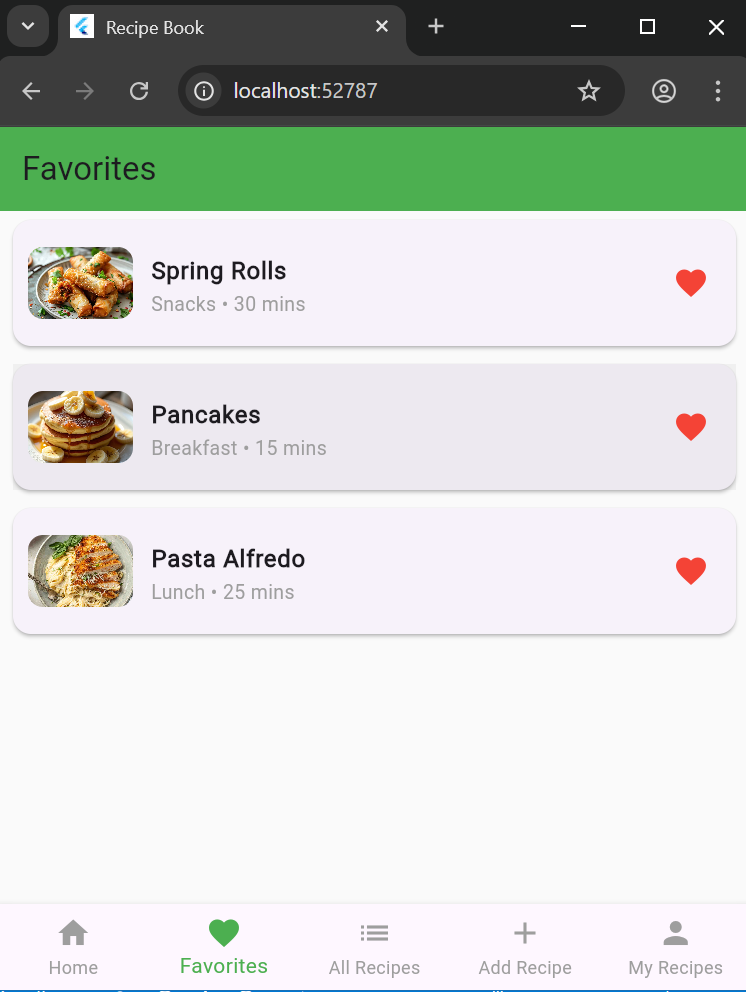




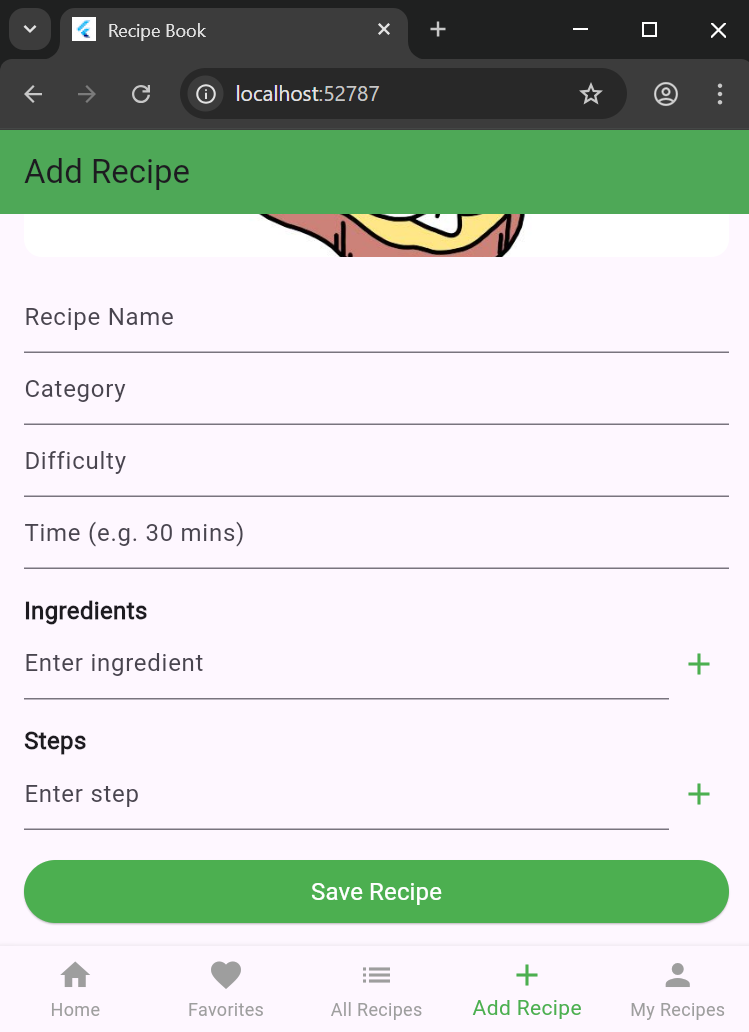
1. Recipe details page

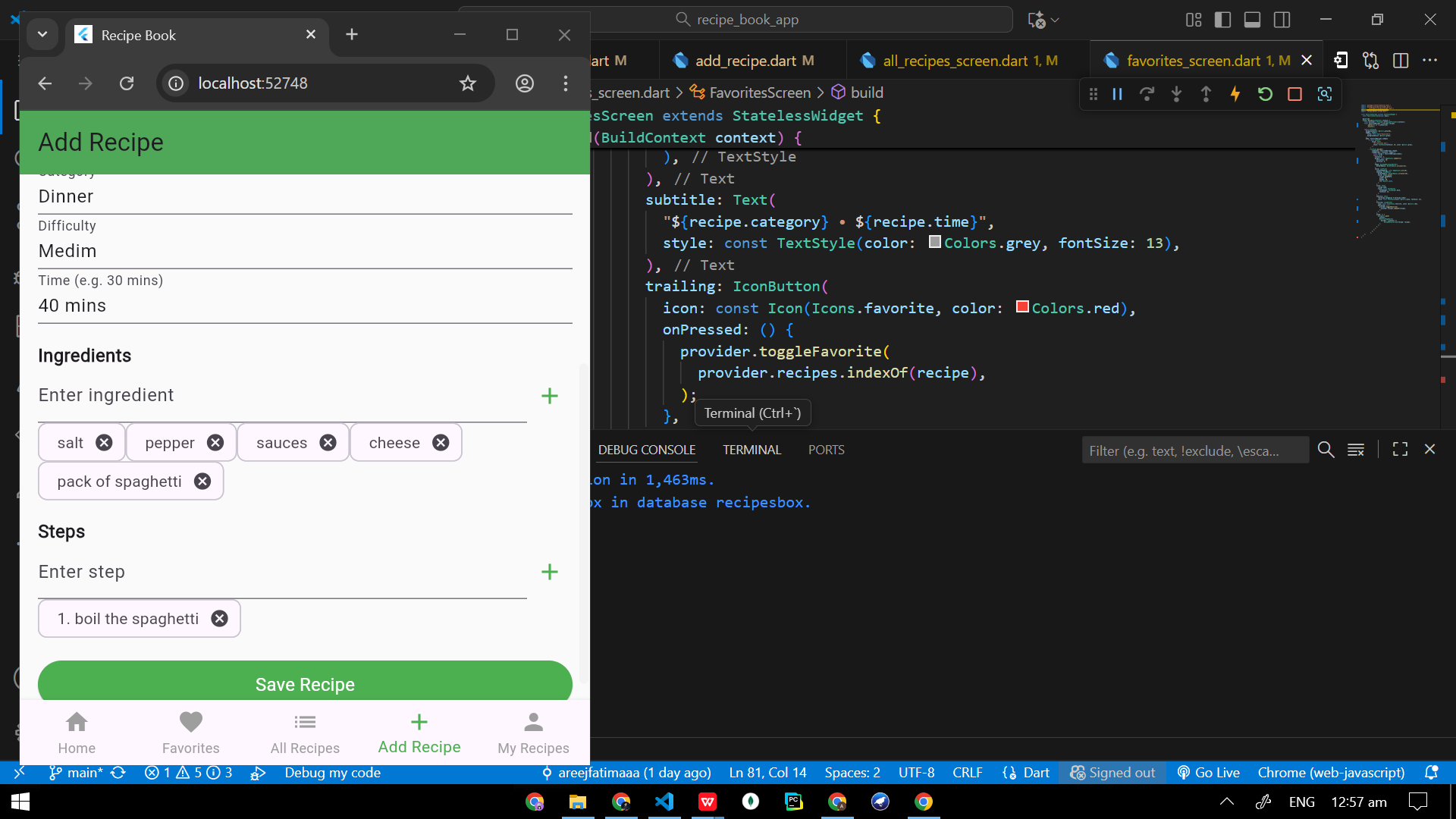


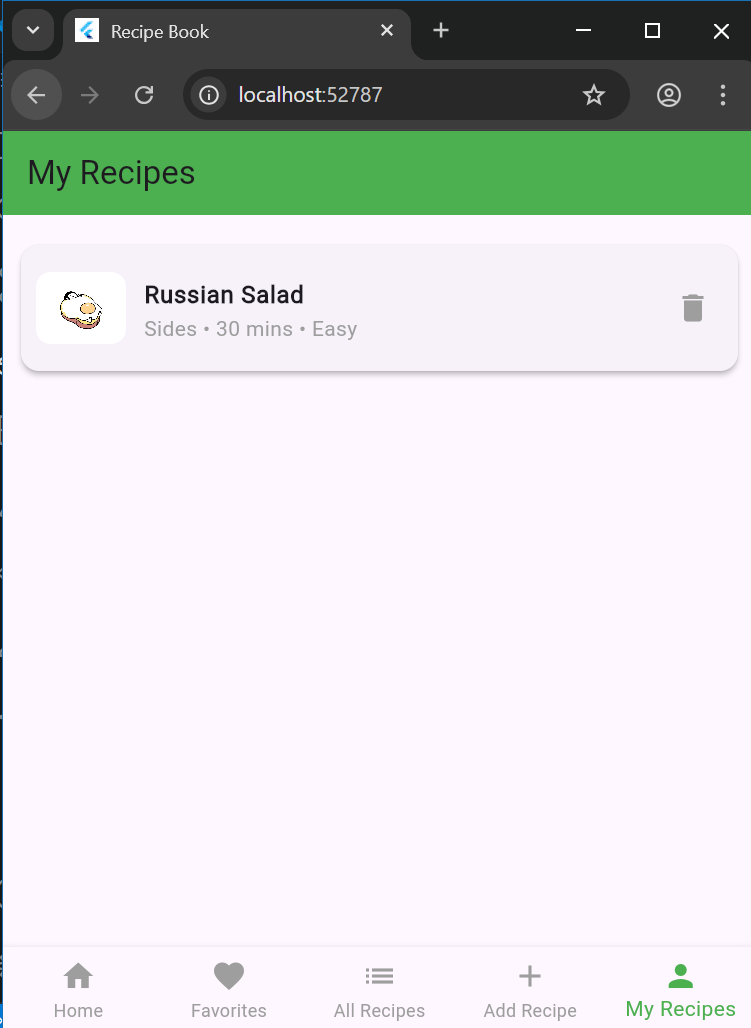
1. Favourites page

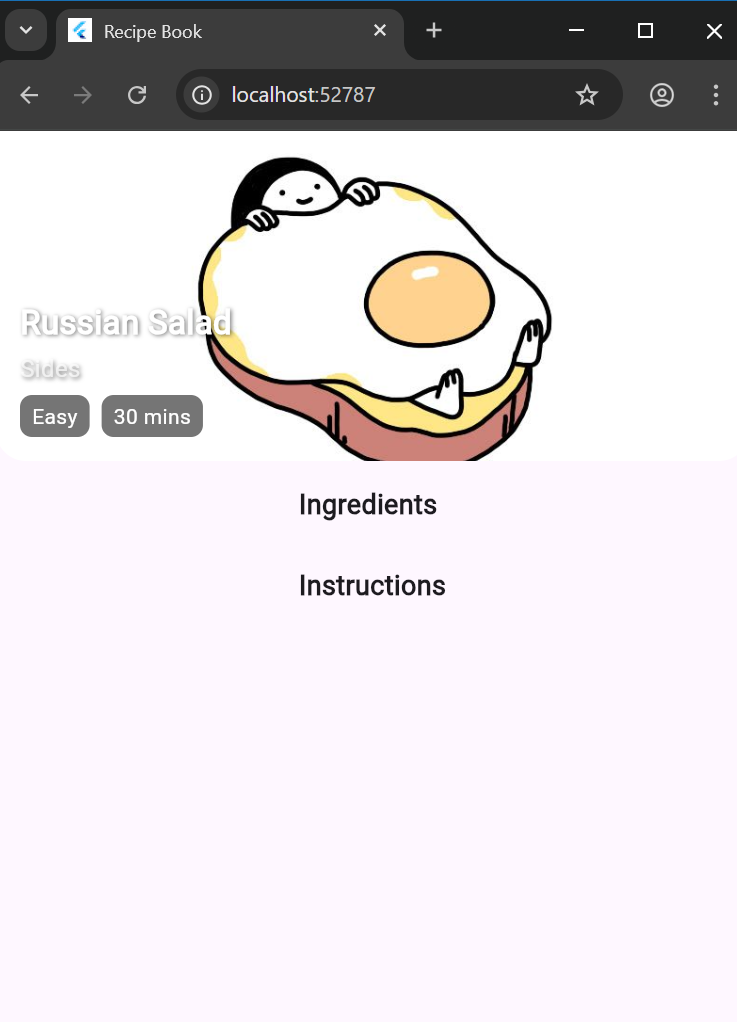


1. Add recipe form and updated my recipe section:









1. Data Storage

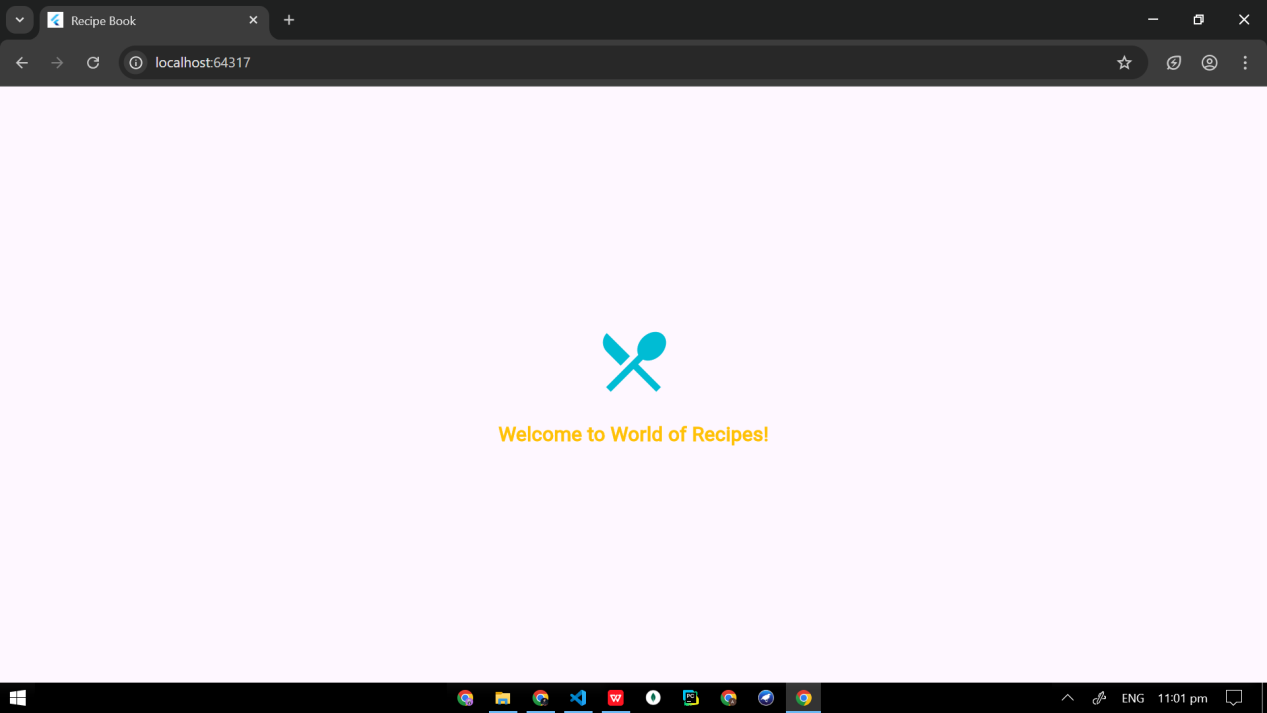
The app uses **Hive,** to store recipe data locally on the user’s device. Hive was chosen for it’s speed, simplicity and offline capability. Each recipe including it’s name, category, difficulty level, preparation time, ingredients, steps and favourite status is stored as a hive object. This allows the app to quickly retrieve and update information without relying on an external server.

1. Errors and Bugs encountered:

* Splash Screen loop after saving a new recipe

Description:

Whenever a new recipe was added using the Add Recipe form instead of returning to the Home screen or updating the recipe list, the app redirected back to the splash screen and stayed there indefinitely.



Cause of the issue:

The naviagtion flow wasn’t properly managed. In initial code, after saving a recipe, it used line of code that tried to pop back to a route that didn’t exist in the current stack, which resulted defaulting back to the splash screen route. Because splash screen was still in the stack from when the app first started.

Solution implemented:

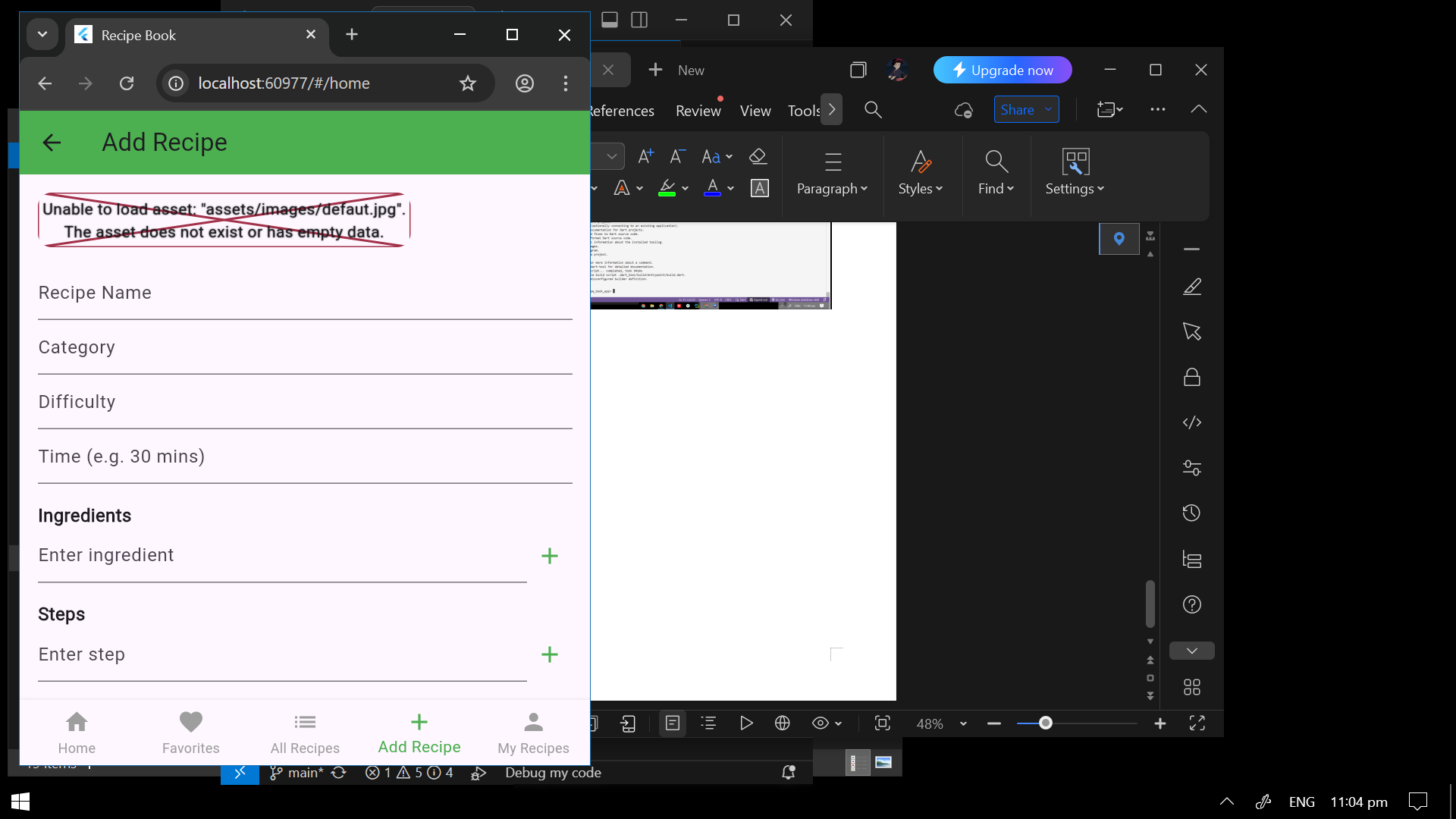
Replaced the navigation method to return directly to Home screen ensuring smooth redirection.

* Image picker not working on desktop

Problem:

When implementing the image picker feature in the Add recipe screen, an error appeared on desktop saying that the asset or file could not be loaded.

The app crashed or displayed a red error screen whenever the user attempted to pick or display an image.



**Cause:**

Flutter image\_picker package doesn’t fully support desktop platforms(especially Windows) the same way it supports Android or iOS.

This caused issues while loading local images from system.

**Solution:**

Since the main goal was to ensure functionality for testing and submission, the image picker feature was temporarily removed.

Instead a default image was assigned automatically to every new recipe.

* **My Recipe screen not updating after adding new recipe.**

**Problem:**

After successfully adding a recipe through the Add recipe form, The My Recipe or All Recipes screen did not automatically update to display the newly added recipe. The new recipe appeared only after restarting the app.

**Cause:**

The issue occurred because the recipe list was not being refreshed when a new recipe was added.

In flutter’s state management, if the provider or state object is not notified properly using notifyListeners(), the UI will not rebuild and reflect the new changes.

**Action Taken:**

The issue was resolved by ensuring that RecipeProvider’s method responsible for adding a recipe calls notifyListeners() after updating the list.

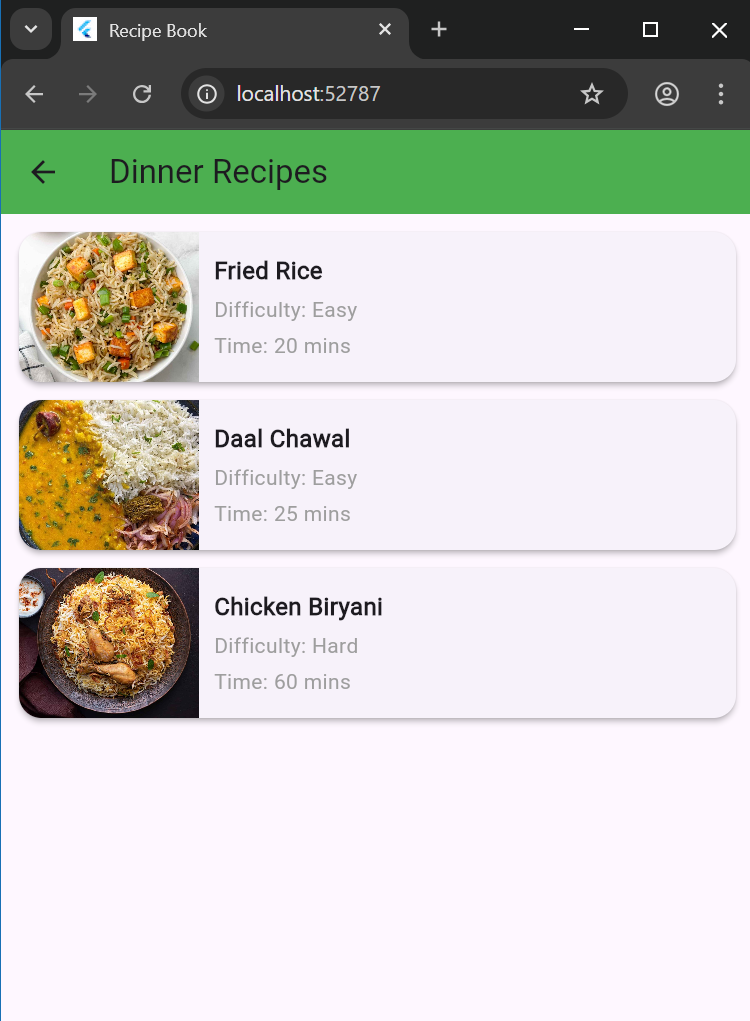
Additionally, the navigation logic was checked to confirm that the screen rebuilds from the provider’s latest state.

* **Bottom Naviagion Bar Disappeared**

**Problem:** The bottom navigation bar disappeared after navigating between screens.

**Cause:** Incorrect screen routing and missing scaffold hierarchy.

**Solution:** Adjusted the widget structure to include the navigation bar consistently across all main screens using a shared layout pattern.



* **Errors in downloading packages.**

