Department of Software Engineering

Mehran University of Engineering and Technology, Jamshoro

Course Title: Mobile Application Development (SW-327)
Instructor: Ms. Mariam Memon
Assignment Type: Complex Engineering Problem

Semester: 6th
Year: 3rd
Submission Deadline: 22-10-2025
Assessment Score: 15 Marks

Submitted By:

Name: Muhammad Yaqoob, Muzammil Hussain
Roll No: 22SW013, 22-21SW160
Department: Software Engineering
Batch: 22SW

Submitted To:

Ms. Mariam Memon

Food Cal Scanner Mobile Application Development Report

Real World Problem Identification

Many people want to keep track of what they eat every day, like calories, protein, carbs, and fats, but doing this manually takes too much time and effort.
Usually, people must enter food items one by one and search for nutrition details online. It's slow, boring, and sometimes they don't even know how much they ate. Because of this, most users stop tracking after a few days.

Main Problems Faced:

- Manually entering every meal takes too much time.
- Hard to estimate part size or nutrition for mixed meals (like biryani, burgers, etc.).
- Nutrition data comes from various sources with different formats.
- People want offline access and privacy — they don't want to send all their data to servers.

So, the real problem is:

How can we make food and calorie tracking faster, easier, and more automatic without losing accuracy?

Proposed Solution

My app, **FoodCal Scanner**, is made to solve this problem in a simple and smart way.
It uses **AI and the phone's camera** to find food and estimate calories automatically. The user just takes a photo — that's it!

Main Features:

- Take a photo (or select one from the gallery) of your meal.

- The AI model (Google Gemini using google_generative_ai package) detects food names and estimates nutrition like calories, protein, carbs, and fats.
- All results are saved locally using SQLite, so users can see their history later, even without internet.
- Clean, simple, and responsive design that works on all screen sizes and both Android & iOS.

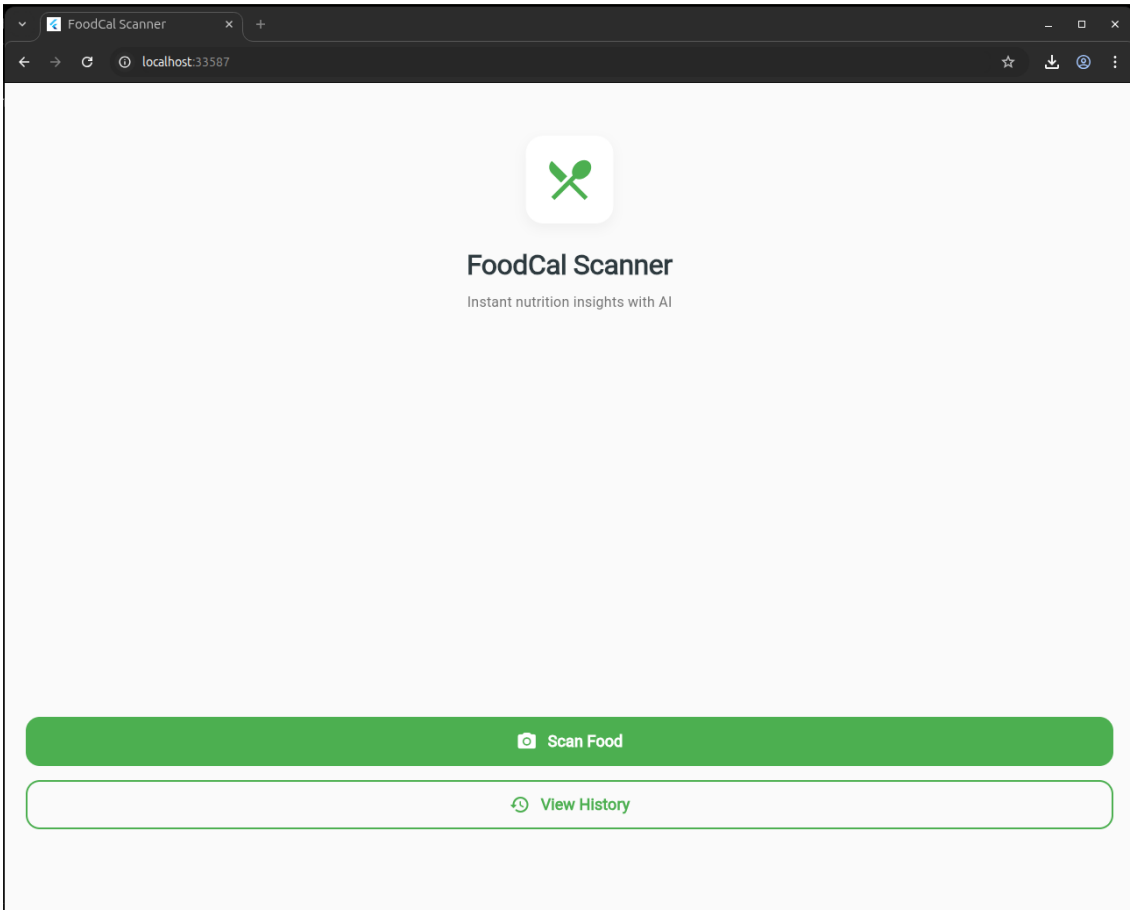
Responsive User Interfaces (Screenshots)

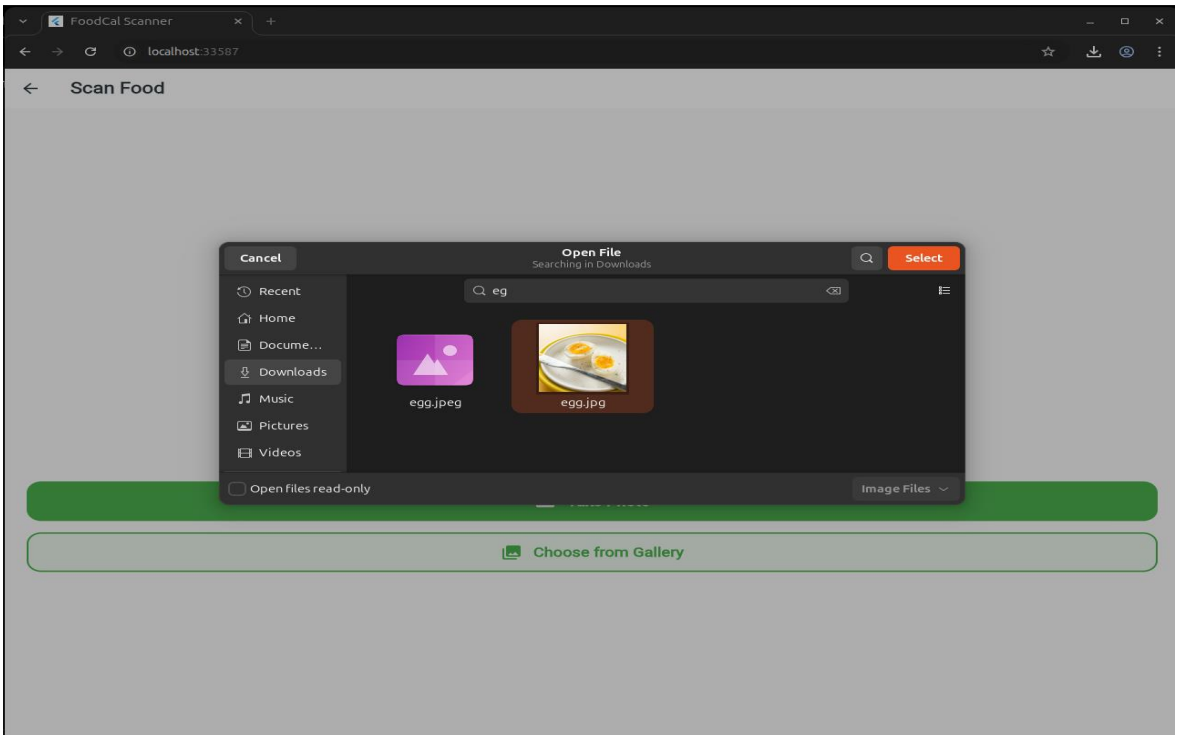
The app includes the following main screens:

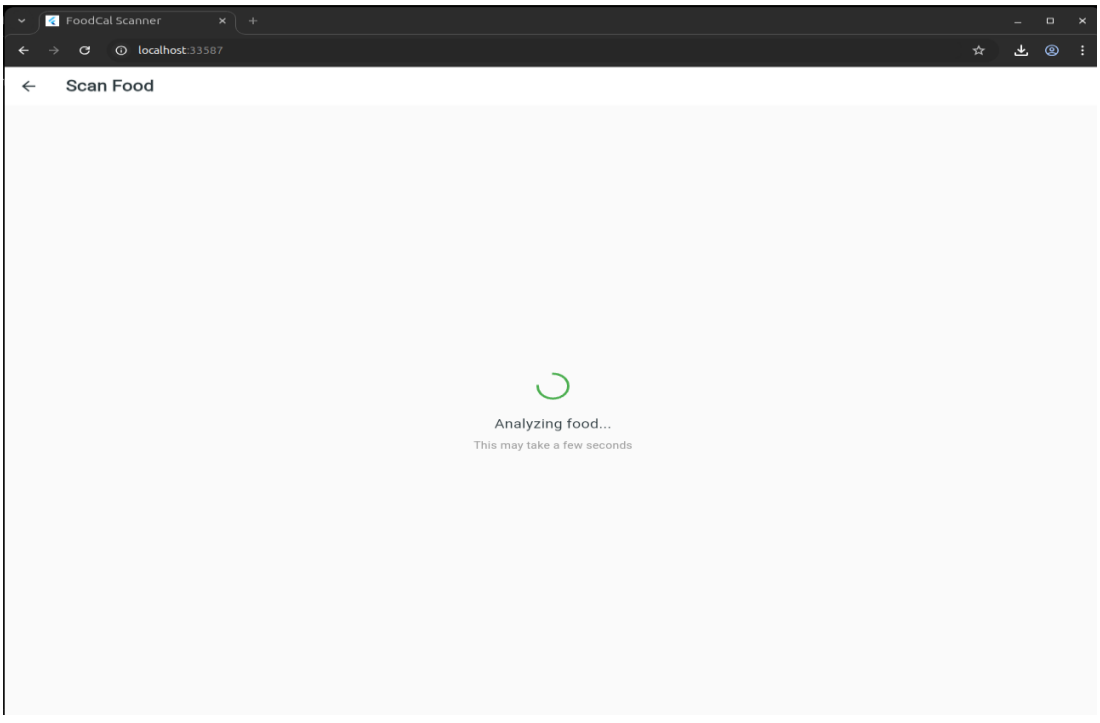
1. **Home Screen** – Has two main buttons: Scan and History.
2. **Scanner Screen** – Lets users take a photo or select from gallery and show progress while analyzing.
3. **Result Dialog** – Displays the image with detected food names, calories, and macronutrients.
4. **History Screen** – Lists all earlier scans (newest first) with date and time.

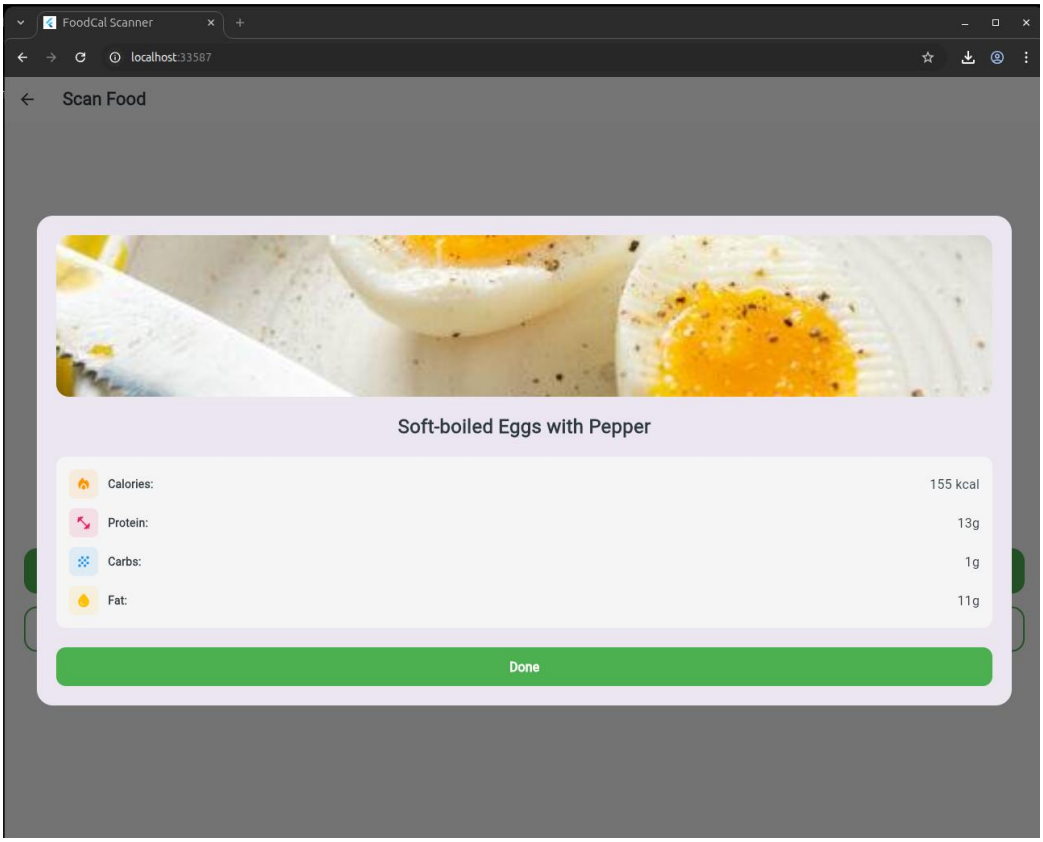
The UI is designed to look clean and easy to use on both **phones and tablets**, adjusting layout automatically based on screen size.

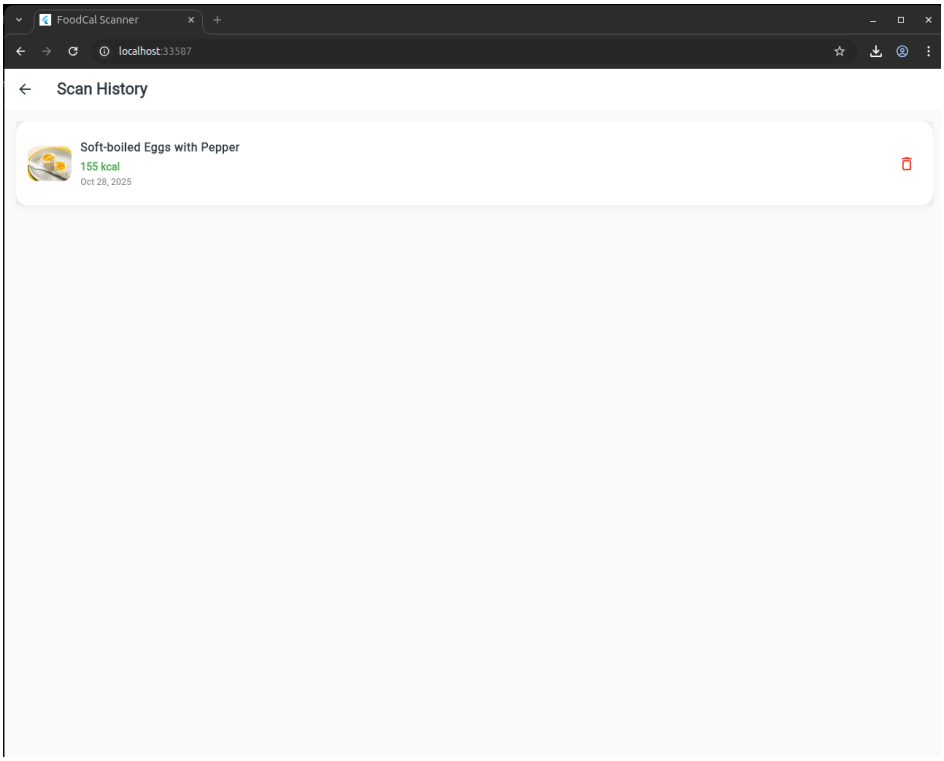
SCREENSHOTS:

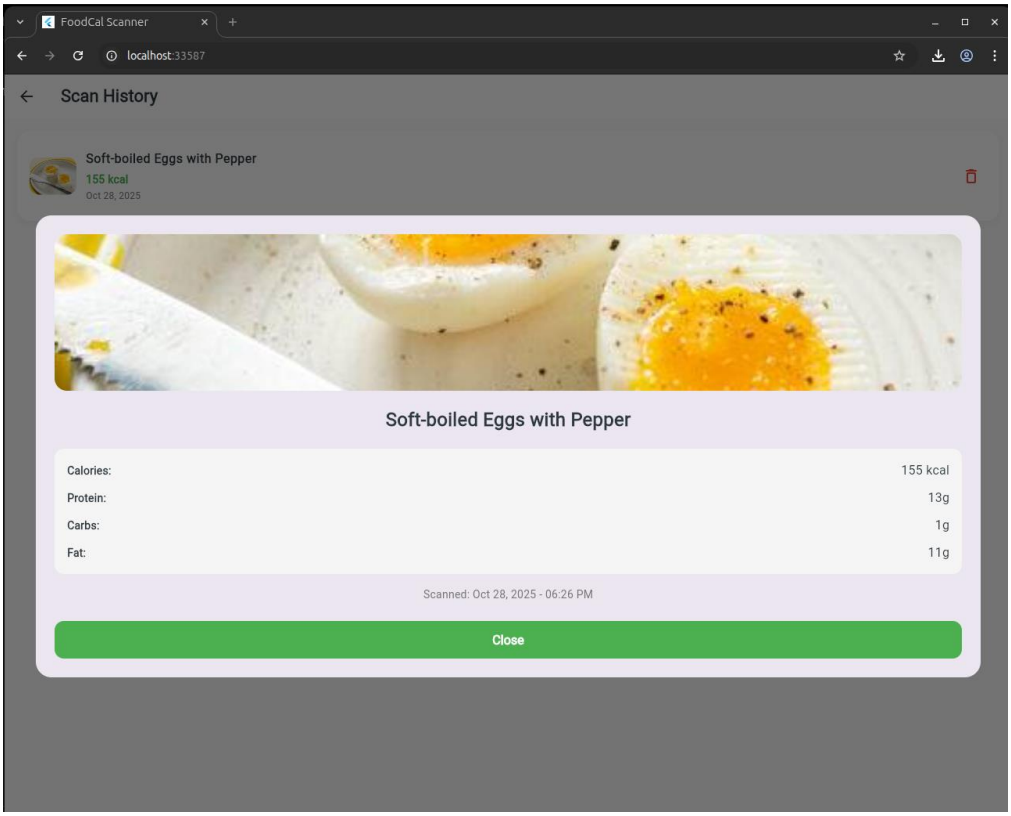

FoodCal Scanner
localhost:33587
FoodCal Scanner
Instant nutrition insights with AI
Scan Food
View History

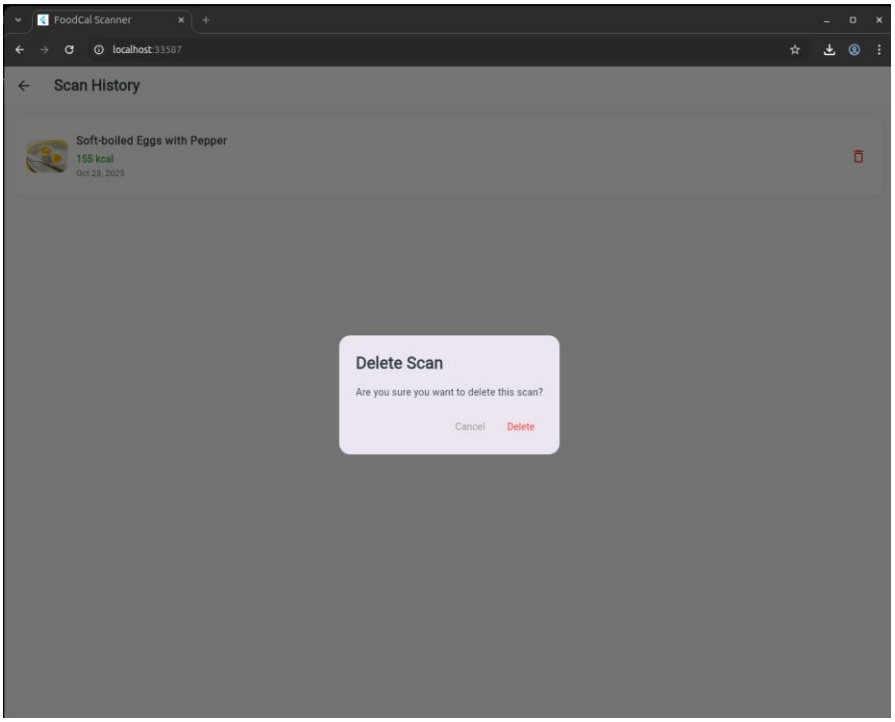

FoodCal Scanner
localhost:33587
Scan Food
Cancel
Open File
Searching in Downloads
Select
Recent
Home
Docume...
Downloads
Music
Pictures
Videos
eg
egg.jpeg
egg.jpg
Open files read-only
Image Files
Choose from Gallery

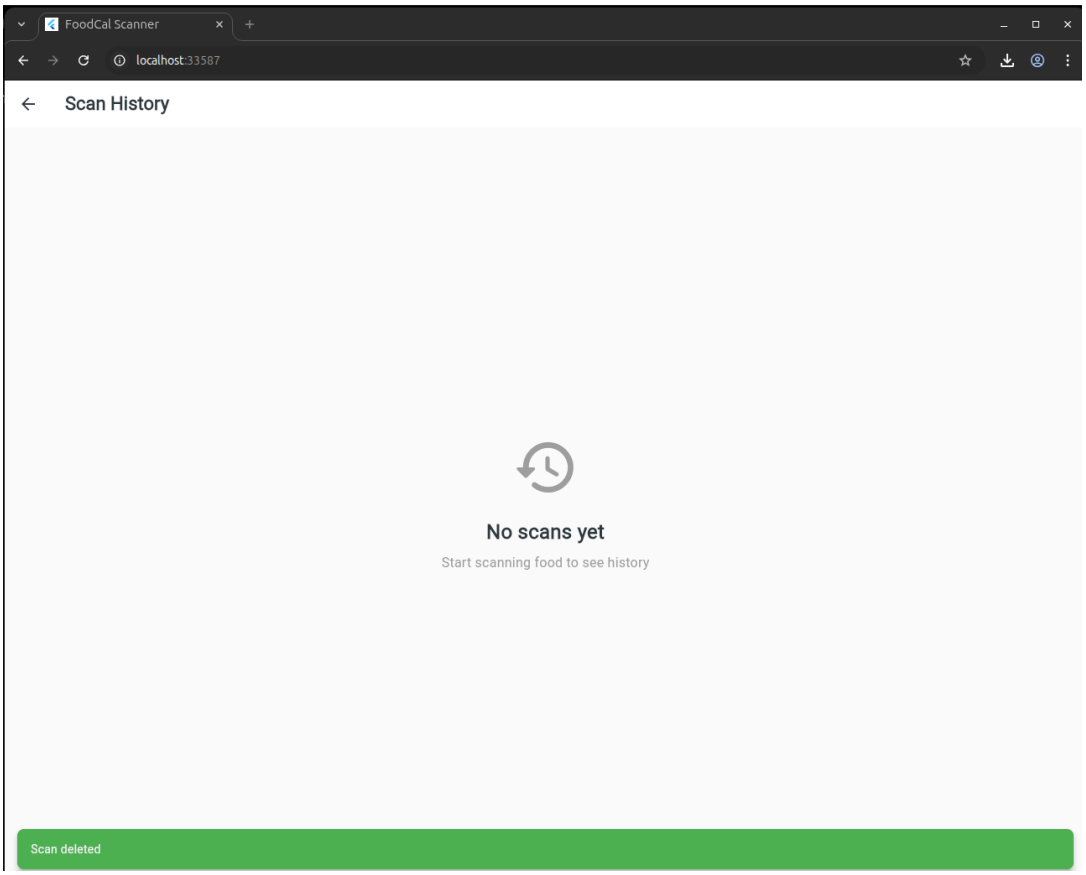

FoodCal Scanner
localhost:33587
Scan Food
Analyzing food...
This may take a few seconds


FoodCal Scanner
localhost:33587
Scan Food
Soft-boiled Eggs with Pepper
Calories:
155 kcal
Protein:
13g
Carbs:
1g
Fat:
11g
Done


FoodCal Scanner
localhost:33587
Scan History
Soft-boiled Eggs with Pepper
155 kcal
Oct 28, 2025


FoodCal Scanner
localhost:33587
Scan History
Soft-boiled Eggs with Pepper
155 kcal
Oct 28, 2025
Soft-boiled Eggs with Pepper
Calories: 155 kcal
Protein: 13g
Carbs: 1g
Fat: 11g
Scanned: Oct 28, 2025 - 06:26 PM
Close


FoodCal Scanner
localhost:33587
Scan History
Soft-boiled Eggs with Pepper
155 kcal
Oct 28, 2025
Delete Scan
Are you sure you want to delete this scan?
Cancel
Delete



Data Storage

Chosen Storage:

SQLite (using the sqflite Flutter package)

Why SQLite?

- Works **offline**, so no internet is needed.
- **Fast and simple** for structured data like our scan history.
- **Cross-platform** — works on both Android and iOS easily.
- **Privacy-friendly** — all data stays on the user's device, not on any server.

Database Details:

- **File name:** food_scans.db
- **Table name:** food_scans

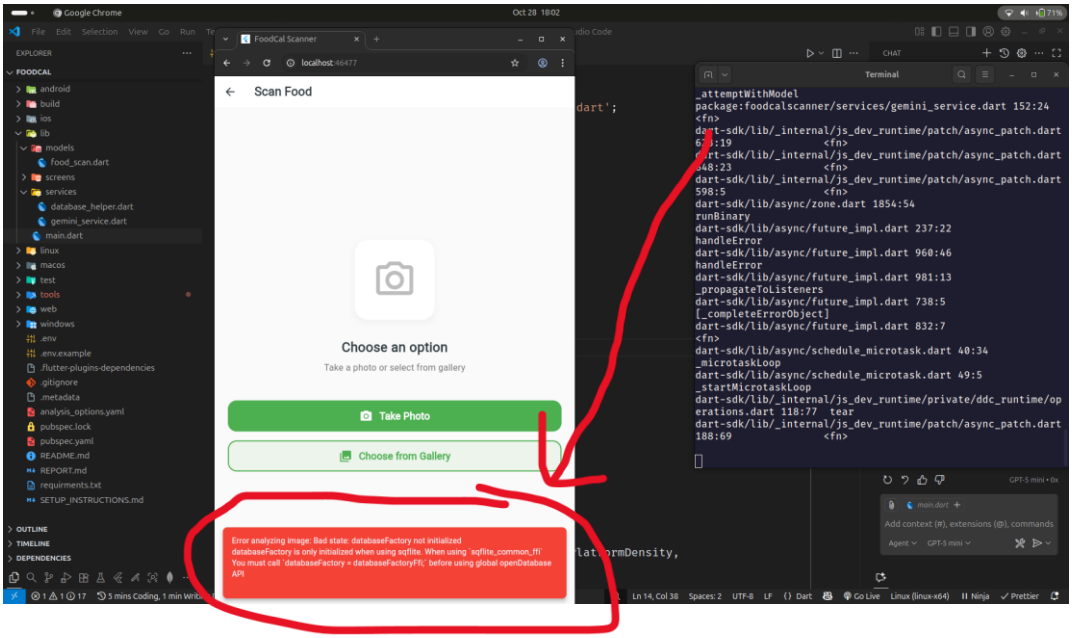
Schema:

Column Name	Type	Description
id	INTEGER (Primary Key)	Auto-incremented record ID
foodName	TEXT	Name of detected food
calories	REAL	Calories (supports decimal values)
protein	TEXT	Protein amount
carbs	TEXT	Carbs amount
fat	TEXT	Fat amount
imagePath	TEXT	Image file location
timestamp	TEXT	Date & time (ISO8601 format)

database Crash on Web (Chrome)

Issue:

The app crashed on the web with the error **“databaseFactory not initialized”** because sqflite doesn’t support web platforms.



Fix:

I added a **web-safe fallback** to database_helper.dart. When running on the web, the app now uses an **in-memory list** to store scans instead of SQLite. On mobile or desktop, it still uses sqflite.

Result:

The app now runs smoothly on Chrome without any database errors, and the scan feature works properly.

I faced other bugs aswell but i didnt read cep before i was just creating app