

# Deck Builder Project Documentation

## Overview

This Flutter application allows users to build, edit, manage, and share decks for a trading card game. It includes user authentication, deck creation tools, a profile page, and public deck browsing.

## Project Structure

```
lib/  
├── data/  
│   ├── model/      # Models (Card, Deck, User)  
│   ├── util/        # API logic, providers, utilities  
│   └── view/         # UI pages (Home, Login, Profile, Deck Builder)  
└── main.dart        # Entry point
```

# Requirements

- Flutter SDK (3.10+ recommended)
- Dart SDK (included with Flutter)
- PocketBase backend
- Internet access for API calls

# Setup Instructions

## 1. Install Flutter

Follow: <https://docs.flutter.dev/get-started/install>

Verify installation:

```
flutter doctor
```

## 2. Install Dependencies

```
flutter pub get
```

### 3. Configure PocketBase

Navigate to PocketBase from main:

```
cd ../deck_builder/pocketbase
```

Start PocketBase:

```
./pocketbase serve
```

Required collections: - `users` - `decks` - `cards`

Make sure `api.dart` points to your PocketBase URL.

## 4. Run the App

Navigate to main file (deck\_builder)

Then run with this command

```
flutter run
```

# Feature Guide

## Home Page

- Displays public decks.
- Tap Plus Button to create new deck → opens Deck Builder.
- Create Deck when logged in.

## Login Page

- Username + password login.
- Session stored via `UserProvider` .

## Sign Up Page

- Email + username + password account creation



## Profile Page

- Shows logged-in user info.
- Displays user's decks.
- Delete decks.
- Logout option.

## Deck Builder

- Create or edit decks.
- Add/remove cards.
- Save as public or private.

## **API System (APIRunner)**

Handles: - Authentication - Fetching decks / cards - CRUD operations for decks - Fetching users

## Running Tests

Tests are located in:

```
test/
```

Run all tests from deck\_builder folder:

```
flutter test
```

# Troubleshooting

## PocketBase connection errors

Check: - Server is running. - API URLs are correct. - CORS allows Flutter requests.

## Tests failing

Make sure dependencies exist:

```
flutter pub add mocktail  
flutter pub add flutter_test
```

## Conclusion

This project provides a full end-to-end deck builder system using Flutter and PocketBase. Extend functionality by modifying UI (in `data/view`) or backend logic (`data/util`).