back4app_demo

Submitted By:

Jolene Melita Saldanha

2024tm93146@wilp.bits-pilani.ac.in

App Architecture & Core Components

App Structure:

- **Frontend:**
    - Built with **Flutter**, providing a responsive, cross-platform UI
    - Manages screens for Login and Home (Student Management)
    - Uses **Parse Server SDK for Flutter** to communicate with backend

- **Backend:**
    - Powered by **Back4App**, a Parse Server-based cloud backend
    - Handles data storage, user management, and authentication securely
    - Exposes data via REST-like APIs consumed by the Flutter app

    - Flutter app calls Back4App for all backend operations (login, CRUD)
    - Parse SDK abstracts network requests, managing sessions and data parsing
    - Data sync and UI update happen seamlessly using Flutter's state management

User Authentication & Database Structure

User Authentication Process:

User provides username & password in Flutter app

- App sends credentials to Back4App via Parse SDK login method

- Back4App verifies user and returns a session token

- Session is maintained by the SDK for secured API requests

- Logout clears session and redirects user to login screen

    Database Structure on Back4App:

- Class: Student
    - Fields:
        - name (String)
        - age (Number)

- Users managed internally by Back4App's User class

- **The solution leverages Flutter for a rich mobile experience and Back4App for scalable backend services, delivering secure authentication and seamless data management.**

Tools Used & Challenges Overcome

1. **Development Tools**

- **Flutter** — UI & app logic

- **Parse Server SDK (Flutter)** — Backend communication

- **IDE** – Visual Studio Code

- **Back4App** — Cloud backend & database

- **Xcode & Android Studio** — For platform-specific testing and builds

2. **Key challenges and how they were overcome**

Challenges	Solution
Managing asynchronous data fetch	Used `Future`, `async/await`, and `setState` properly to keep UI in sync
Handling user authentication flow securely – User token was getting invalidated very soon	Leveraged ParseUser methods; managed sessions with autoSendSessionId enabled
Integrating Parse Server SDK with Flutter – Getting multiple errors on flutter run	Followed official docs carefully; ensured correct SDK version and async handling
Debugging platform-specific issues (iOS build)	Resolved naming and syntax typos; used Flutter doctor & Xcode logs for fixes
Managing editing state and form resets	Used a nullable editingStudent object to track edit mode and clear form correctly