NEWS READER APP

FLUTTER ASSIGNMENT

Sri Pratheep S

OVERVIEW

- **User Authentication & Personalization –** Users log in with a username and password, ensuring secure access. Firebase integration helps manage authentication and data storage.
- **Real-Time News Fetching & Submission –** The home screen displays news fetched from an API, and users can contribute by adding news articles with a title and content.
- **Weather Integration for City Updates** – A weather API fetches and displays real-time weather information for different cities, enhancing the user experience.
- **Multi-Language Support with Localization –** The app includes localization, allowing users to switch between different languages for a globally accessible experience.
- **Modern UI with Smooth Navigation & Gestures –** The app utilizes Flutter's styling, theming, and navigation features, along with interactive gestures, to ensure a sleek and user-friendly interface.

FEATURES IMPLEMENTED

- ✓ User Authentication (Login with Username & Password)
- ✓ Home Screen Fetches and Displays News from an API
- ✓ User Can Add News (Title & Content Submission)
- ✓ Weather API Integration for City Weather Updates
- ✓ Multi-Language Support (Localization)
- ✓ Database & Firebase Integration for Storing News
- ✓ Smooth Navigation & User-Friendly UI

CONCEPTS USED

- User Authentication (Firebase Login with Username & Password)
- API Calling (Fetching News & Weather Data)
- Form Handling (User Adding News - Title & Content Submission)
- Database & Firebase Integration (Storing News & User Data)
- Navigation Between Screens (Smooth Transitions & Routing)
- Localization (Multi-Language Support for Global Accessibility)
- Widgets (Stateless & Stateful for UI Management)

FIREBASE INTEGRATION

- **User Authentication** – Firebase Authentication is used for secure login, allowing users to sign in with a username and password.
- **Real-Time Database Storage** – User-submitted news articles are stored in Firebase Firestore, ensuring persistent and synchronized data across devices.
- **Cloud Storage for Media** – Firebase Cloud Storage can be used to store images if users upload news with media attachments.
- **Seamless Data Fetching & Updating** – News content is dynamically retrieved from Firebase, ensuring real-time updates without manual refreshes.
- **Security & Access Control** – Firebase rules manage user access, ensuring only authorized users can post or modify news while preventing unauthorized data changes.

WIDGETS & LAYOUTS

- **Stateless & Stateful Widgets –** The app uses **StatelessWidgets** for static UI elements like headers and buttons, while **StatefulWidgets** handle dynamic content like the news list and weather updates.
- **Flexible Layouts with Rows & Columns –** The app organizes content using **Row**, **Column**, and **Stack** widgets to structure the news feed, weather info, and user input fields efficiently.
- **ListView for Dynamic News Display –** The news feed is implemented using **ListView.builder**, allowing efficient, scrollable, and dynamically loaded news articles.
- **Material Design & Theming –** The app follows **Material Design principles**, using widgets like **AppBar, Card, and FloatingActionButton** for a modern, interactive UI.
- **Adaptive & Responsive UI –** The app ensures responsiveness across devices using **MediaQuery** and **Expanded/Flexible** widgets, making it work seamlessly on different screen sizes.

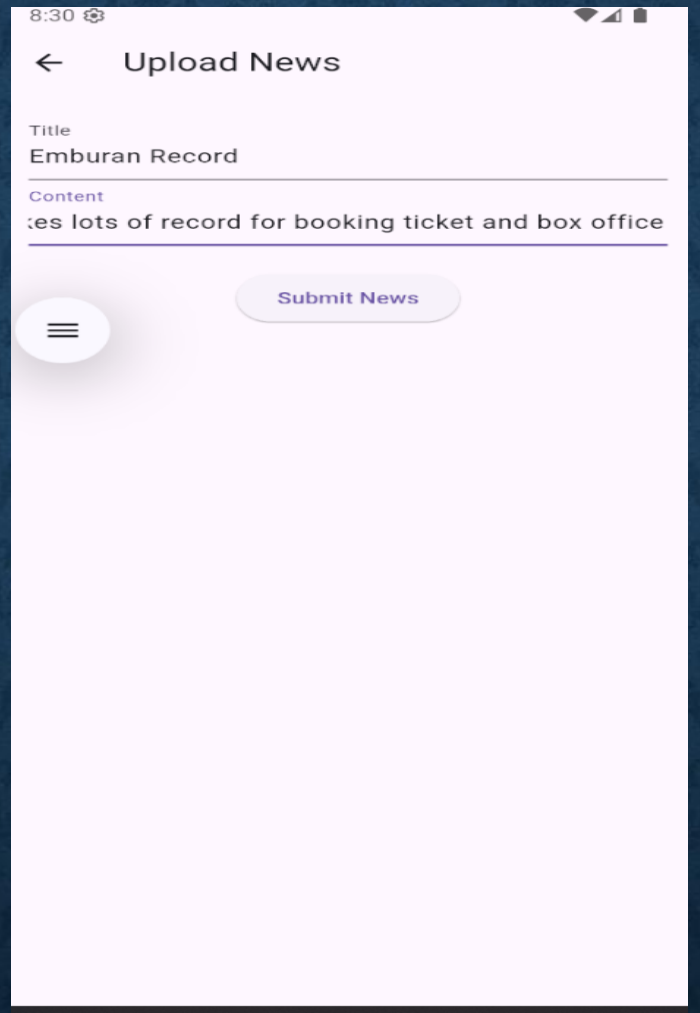
NAVIGATION

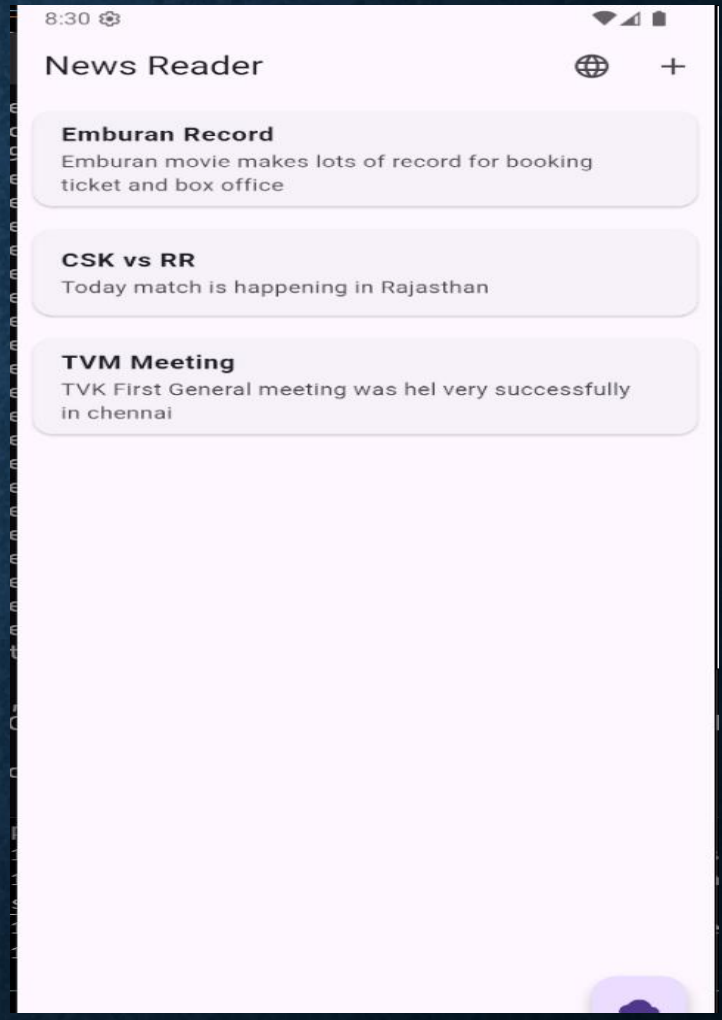
- **Screen Transitions with Navigator** – The app uses Navigator.push() and Navigator.pop() for smooth transitions between screens, such as moving from login to the home screen and opening the news submission page.
- **Named Routes for Organized Navigation –** The app defines and manages routes using **named routes**, making navigation structured and easy to maintain.
- **Bottom Navigation Bar or Drawer** – A **BottomNavigationBar** or **Drawer** can be used for easy access to different sections like Home, Add News, and Settings.
- **Passing Data Between Screens –** News details are passed between screens using **Route parameters**, allowing users to view full articles when clicking on a news item.
- **Back Navigation & State Handling –** The app ensures proper state management when navigating back, preserving user input and news data using providers or stateful widgets.

LOCALIZATION & API CALLING

- **Multi-Language Support with Localization –** The app uses **Flutter Intl** and **JSON/ARB files** to support multiple languages, allowing users to switch languages easily.
- **Fetching News & Weather from APIs –** The home screen loads news from an external API, while a weather API fetches real-time weather updates for user-entered cities.
- **Simple API Requests –** The app connects to the internet to get news and weather data and send user-added news to the server.
- **User-Submitted News Stored via API –** When users add news by entering a title and content, the data is sent to an API and stored for future access.
- **Manual Language Selection –** Users can manually select their preferred language from the settings, and the app updates the UI accordingly.


8:29
News Reader
CSK vs RR
Today match is happening in Rajasthan
TVM Meeting
TVK First General meeting was hel very successfully in chennai


8:30
Upload News
Title
Emburan Record
Content
es lots of record for booking ticket and box office
Submit News


8:30
News Reader
Emburan Record
Emburan movie makes lots of record for booking ticket and box office
CSK vs RR
Today match is happening in Rajasthan
TVM Meeting
TVK First General meeting was hel very successfully in chennai