UNIVERSITY APP

-Developed By **Haries Palaniappan**

[Problem Statement III: Mobile Application:

Develop a Mobile Application using Flutter or Any Mobile App development framework. The application should have Login page, after successful login navigate to Home Page. Display the list of universities in list view with search feature. On clicking a list item - Show the respective Item details in a new page i.e., University Details Page.

Note:

- For Login and other storage use Firebase.
- For Universities List APIs use:

  GET -

  http://universities.hipolabs.com/search?country=India **]**

TABLE OF CONTENTS

1. Introduction

This documentation provides a comprehensive guide for developing a mobile application using Flutter or any mobile app development framework. The application will have a Login page, and upon successful login, it will navigate to the Home page. The Home page will display a list of universities in a list view, along with a search feature. Clicking on a university list item will open a new page, the University Details page, displaying the specific details of the selected university.

2. Technology Stack

- **Flutter:** a cross-platform UI toolkit for building natively compiled applications for mobile, web, and desktop.

- **Firebase:** a mobile and web application development platform that provides authentication, real-time database, storage, and more.

- **API :** API stands for Application Programming Interface. In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications.

3. Prerequisites

Before starting the development process, ensure that you have the following prerequisites installed and set up:

- **Flutter SDK:** Visit the Flutter website (https://flutter.dev/) and follow the installation instructions for your operating system.

- **Android Studio (for Android development) or Xcode (for iOS development):** Install the IDE suitable for your target platform.

- **Flutter and Dart plugins:** Install the Flutter and Dart plugins in your IDE.

- **Firebase account:** Create a Firebase account in the firebase official website "(https://firebase.google.com/) and create a new project.

4. Installation and Setup

- Clone or create a new Flutter project using the Flutter CLI.
- Open the project in your preferred IDE.
- Configure the Flutter SDK path in your IDE.
- Ensure that your project is set up correctly by running the command `flutter doctor` in your project directory.

5. Firebase Configuration

- Create a new Firebase project in the Firebase Console.
- Enable the Firebase Authentication service and choose the authentication methods you want to support (e.g., Email/Password, Google Sign-In, etc.).
- Set up the Firebase project for Android and/or iOS by following the provided instructions.
- Download the `google-services.json` file for Android or `GoogleService-Info.plist` file for iOS and place them in the respective project directories.
- Install the necessary Firebase packages in your Flutter project by adding the Firebase dependencies to the `pubspec.yaml` file and running `flutter pub get`.

6. Building the Login Page

- Create a new Flutter widget for the Login page.
- Design the user interface for the Login page using Flutter widgets.
- Implement the necessary form fields and validation for the login credentials.
- Set up the form submission logic to authenticate the user using Firebase Authentication.
- **Test Login Account :**
    - **Email Address: test@gmail.com**
    - **Password : test@123**

7. Authentication with Firebase

- Integrate the Firebase Authentication package into your Flutter project.
- Implement the authentication logic for the Login page, allowing users to register, log in, and reset their passwords.
- Use Firebase APIs to handle user authentication requests and responses. Ensure that the login process is secure and handles errors gracefully.

8. Navigating to the Home Page

- Create a new Flutter widget for the Home page.
- Set up the navigation logic to move from the Login page to the Home page upon successful login.
- Use Flutter's navigation features, such as `Navigator.push()`, to navigate between screens.

9. Retrieving University Data

- Create a new Flutter service or repository for handling API requests.
- Import the necessary HTTP package for making API calls in Flutter.
- Define a function to fetch the university data from the provided API endpoint.
- Use the HTTP package to make a GET request to the API endpoint (http://universities.hipolabs.com/search?country=India).
- Parse the response data and extract the required information, such as the list of universities.
- Create a model class to represent the university data and deserialize the API response into instances of this model.

10. Implementing the University List View

- In the Home page widget, create a ListView widget to display the list of universities.
- Retrieve the university data using the previously created service or repository.
- Map the list of universities to a list of ListTile widgets, providing the necessary information for each list item.
- Display the list of ListTile widgets within the ListView widget.

11. Adding Search Functionality

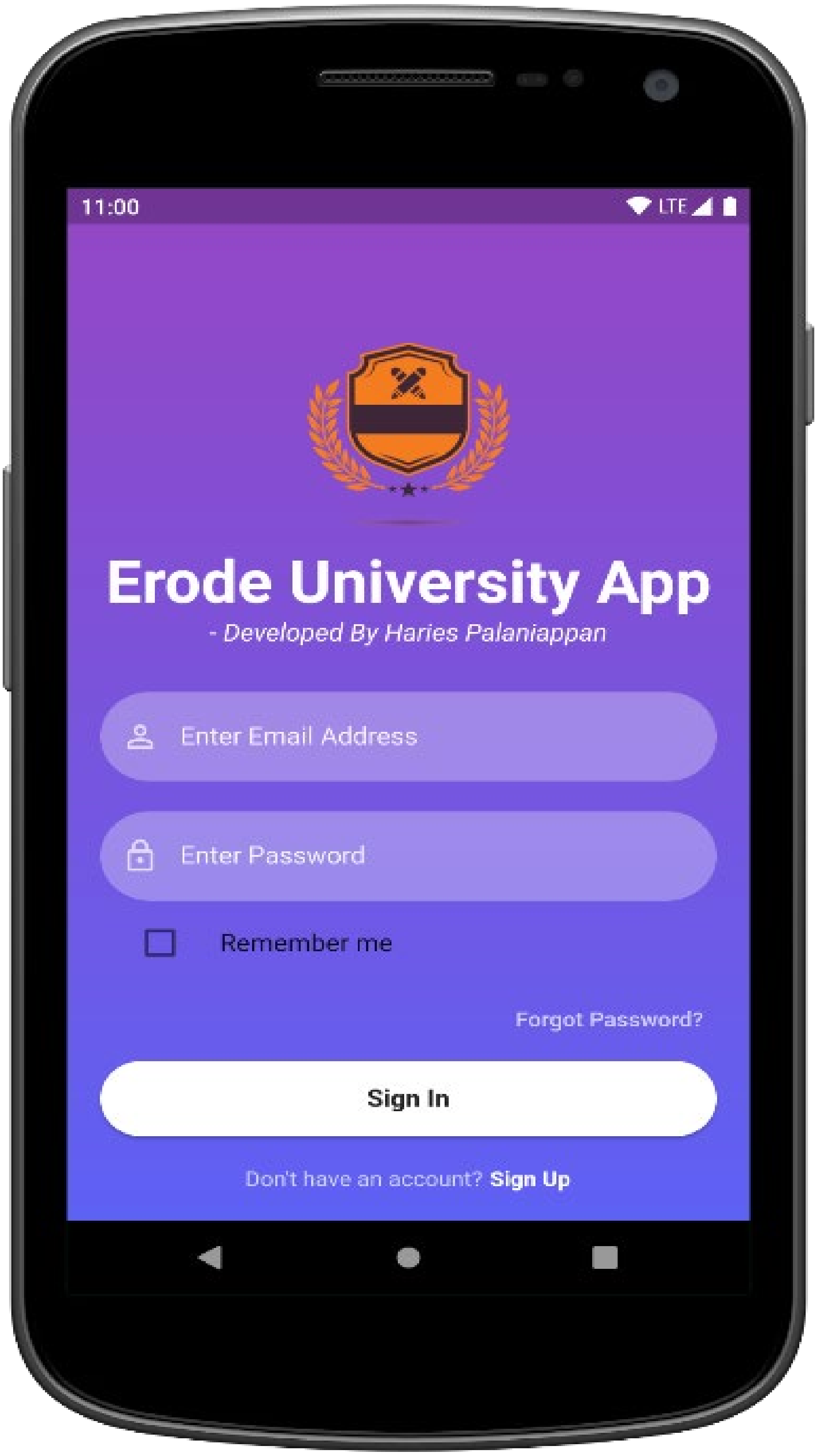
- Create a search bar widget in the Home page for users to enter search queries.
- Implement the necessary logic to filter the list of universities based on the user's search query.
- Update the ListView widget to display the filtered list of universities when a search query is entered.
- Refresh the list as the user types, providing real-time search results.

12. University Details Page

- Create a new Flutter widget for the University Details page.
- Set up the navigation logic to open the University Details page when a university list item is clicked.
- Pass the selected university's data to the University Details page.
- Design the user interface for the University Details page, displaying the specific details of the selected university.
- Use Flutter widgets to present the university details in an organized and visually appealing manner.

13. Screenshots (University App)

13.1 Login Page:



11:00

Erode University App

- Developed By Haries Palaniappan

Enter Email Address

Enter Password

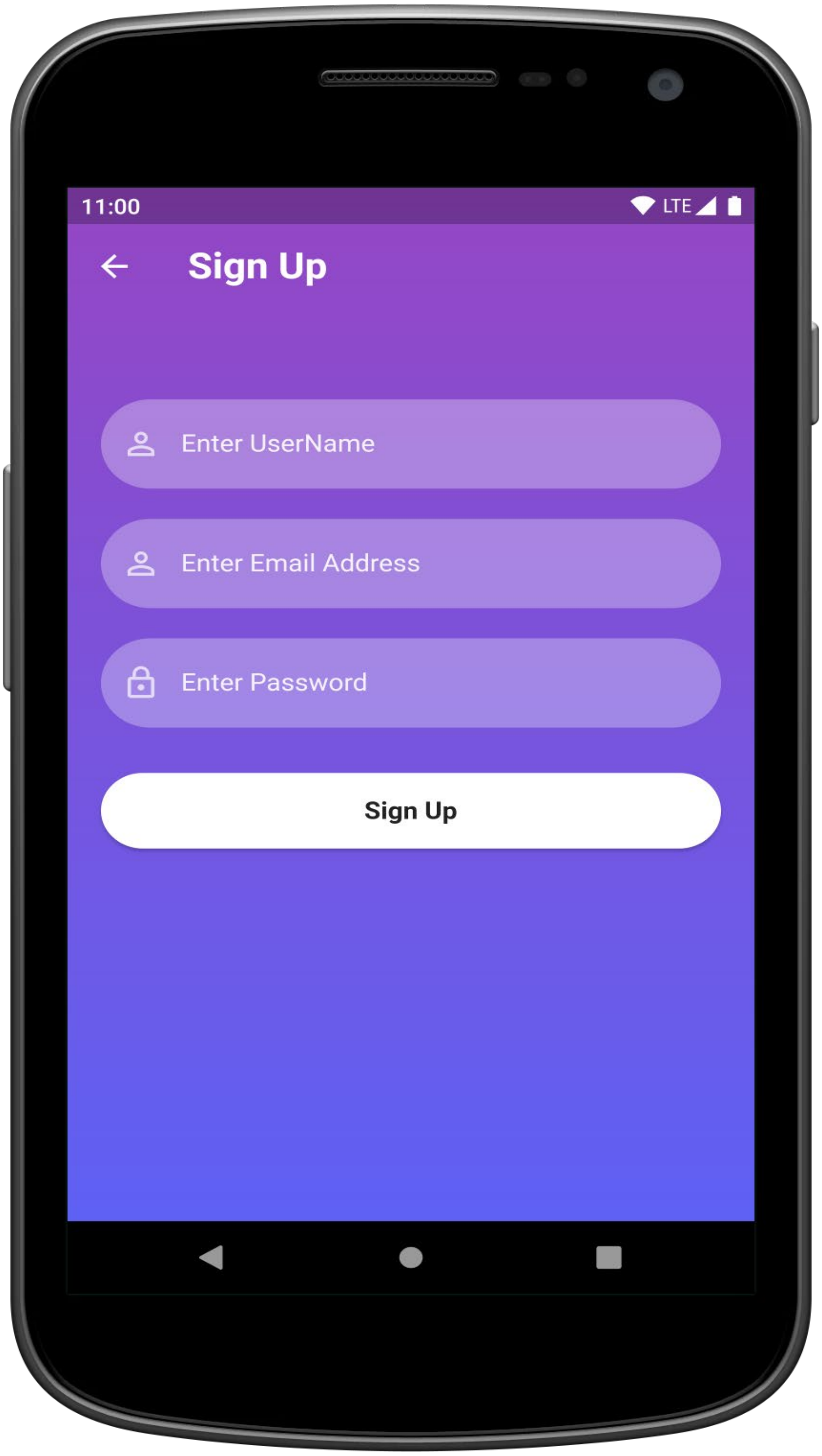
□ Remember me

Forgot Password?

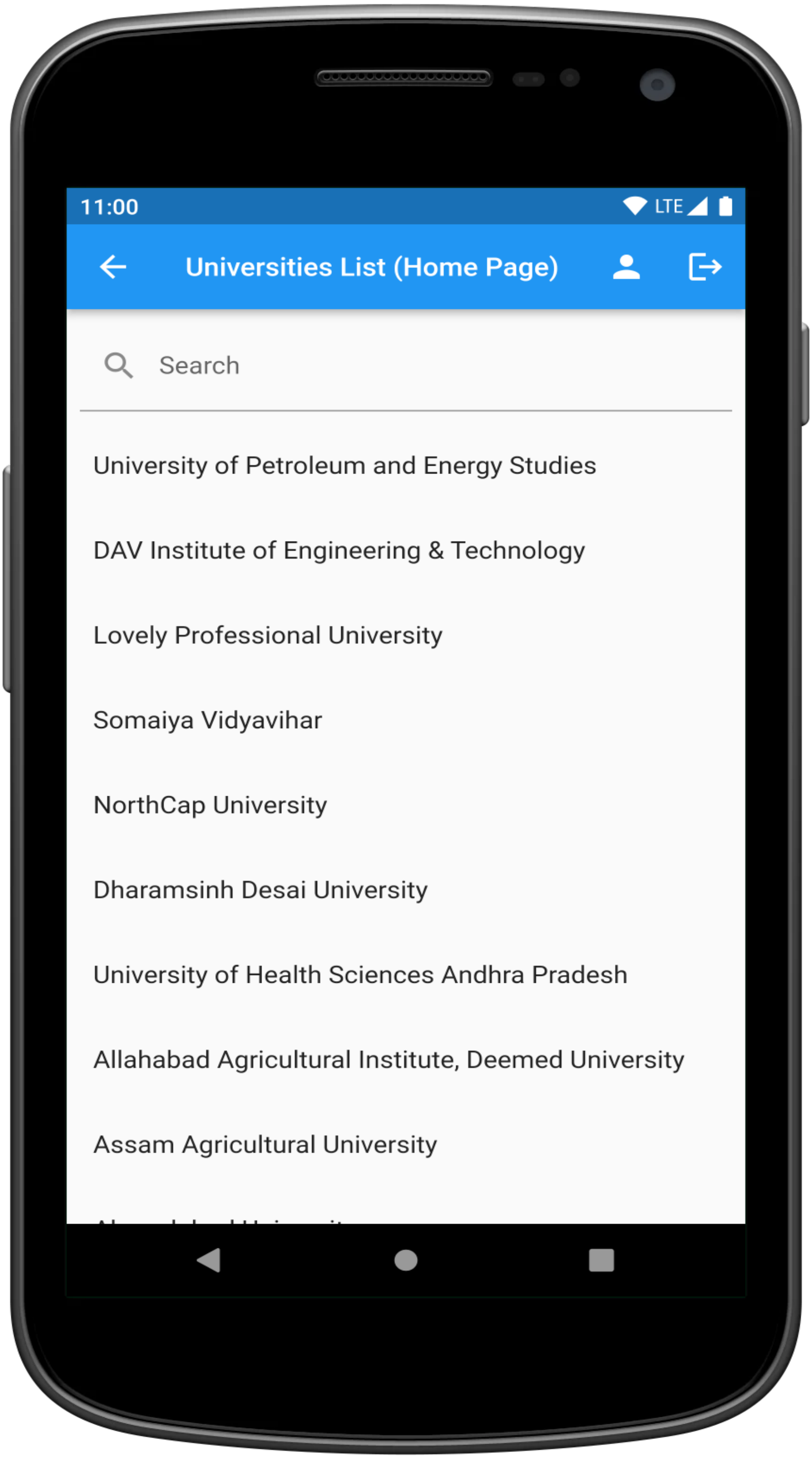
Sign In

Don't have an account? Sign Up

13.2 Sign up Page:



13.3 Home Page:



11:00

Universities List (Home Page)

Search

University of Petroleum and Energy Studies

DAV Institute of Engineering & Technology

Lovely Professional University

Somaiya Vidyavihar

NorthCap University

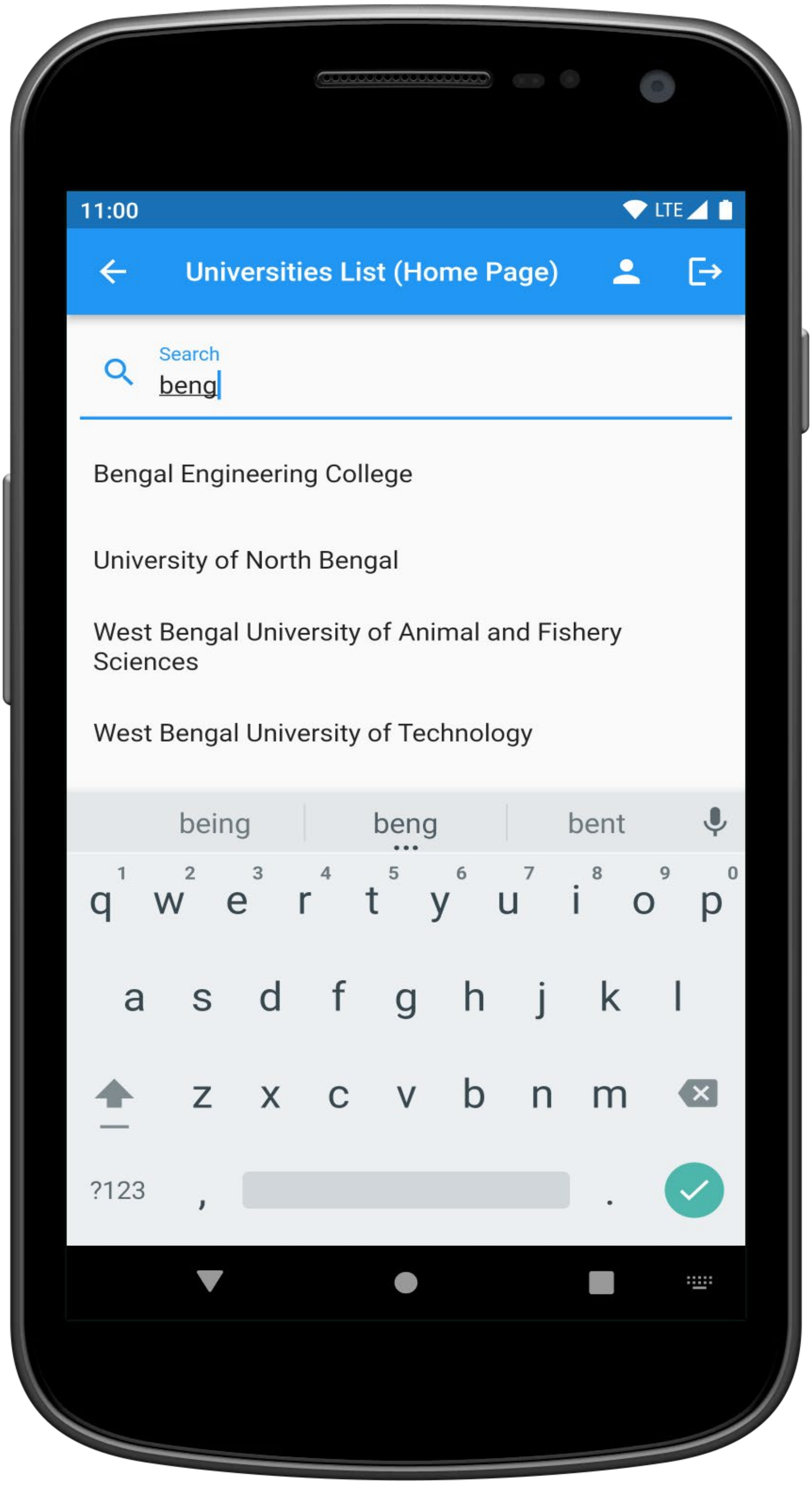
Dharamsinh Desai University

University of Health Sciences Andhra Pradesh

Allahabad Agricultural Institute, Deemed University

Assam Agricultural University

13.4 Search Functionality:



beng

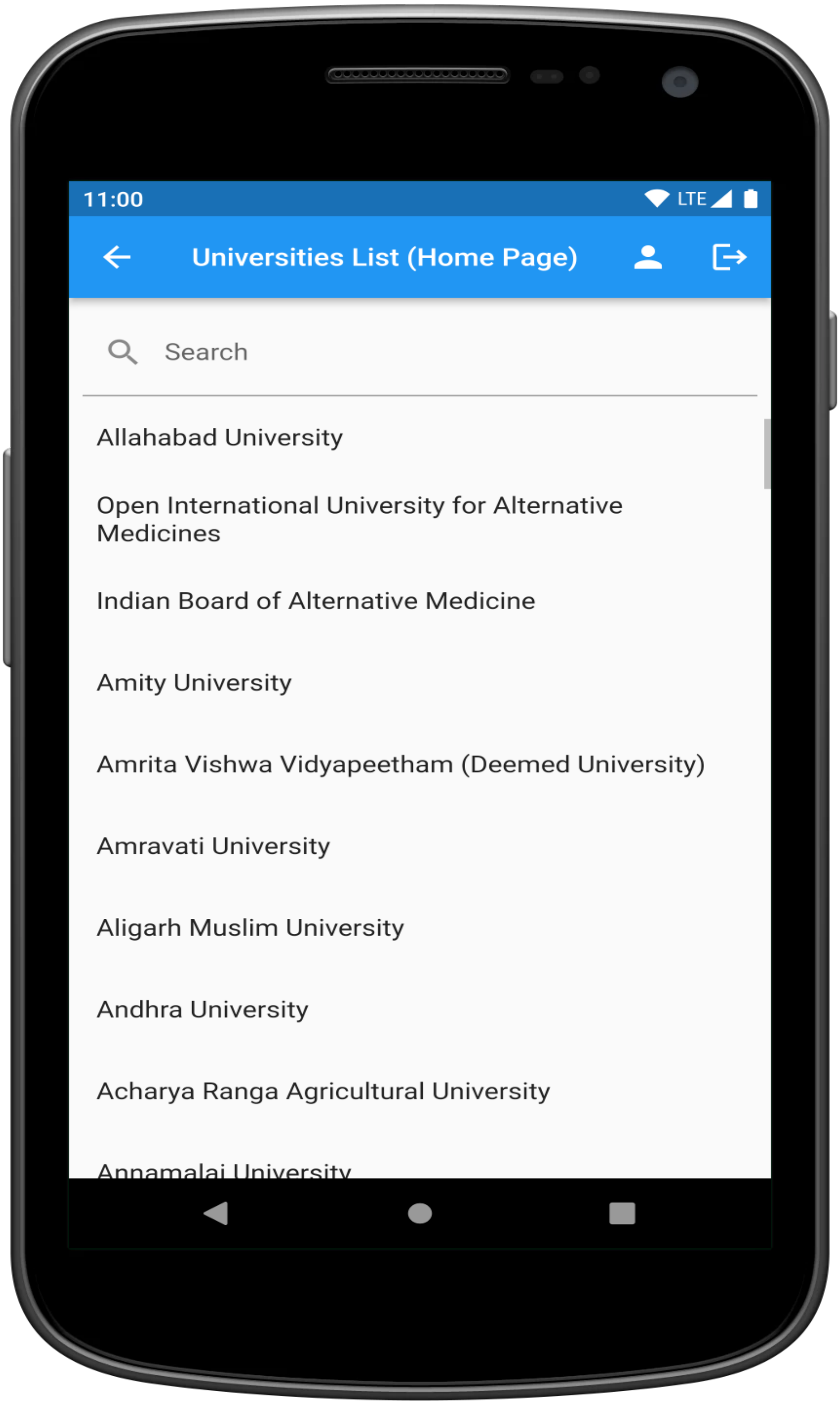
Bengal Engineering College

University of North Bengal

West Bengal University of Animal and Fishery Sciences

West Bengal University of Technology

13.5 Scroll Bar Functionality:



Allahabad University

Open International University for Alternative Medicines

Indian Board of Alternative Medicine

Amity University

Amrita Vishwa Vidyapeetham (Deemed University)

Amravati University

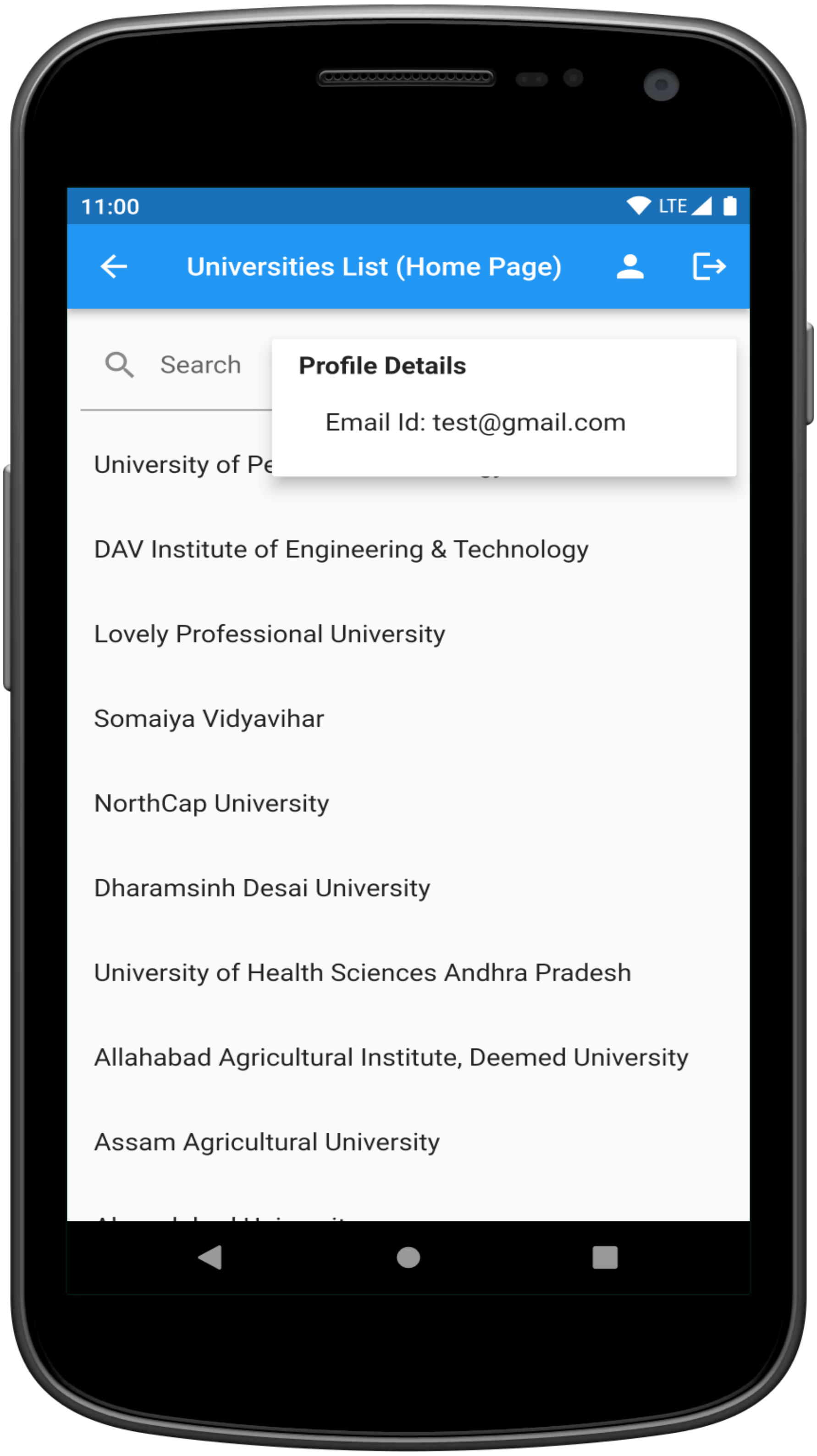
Aligarh Muslim University

Andhra University

Acharya Ranga Agricultural University

Annamalai University

13.6 Profile Details:



11:00

Universities List (Home Page)

Search

Profile Details

Email Id: test@gmail.com

University of Pe

DAV Institute of Engineering & Technology

Lovely Professional University

Somaiya Vidyavihar

NorthCap University

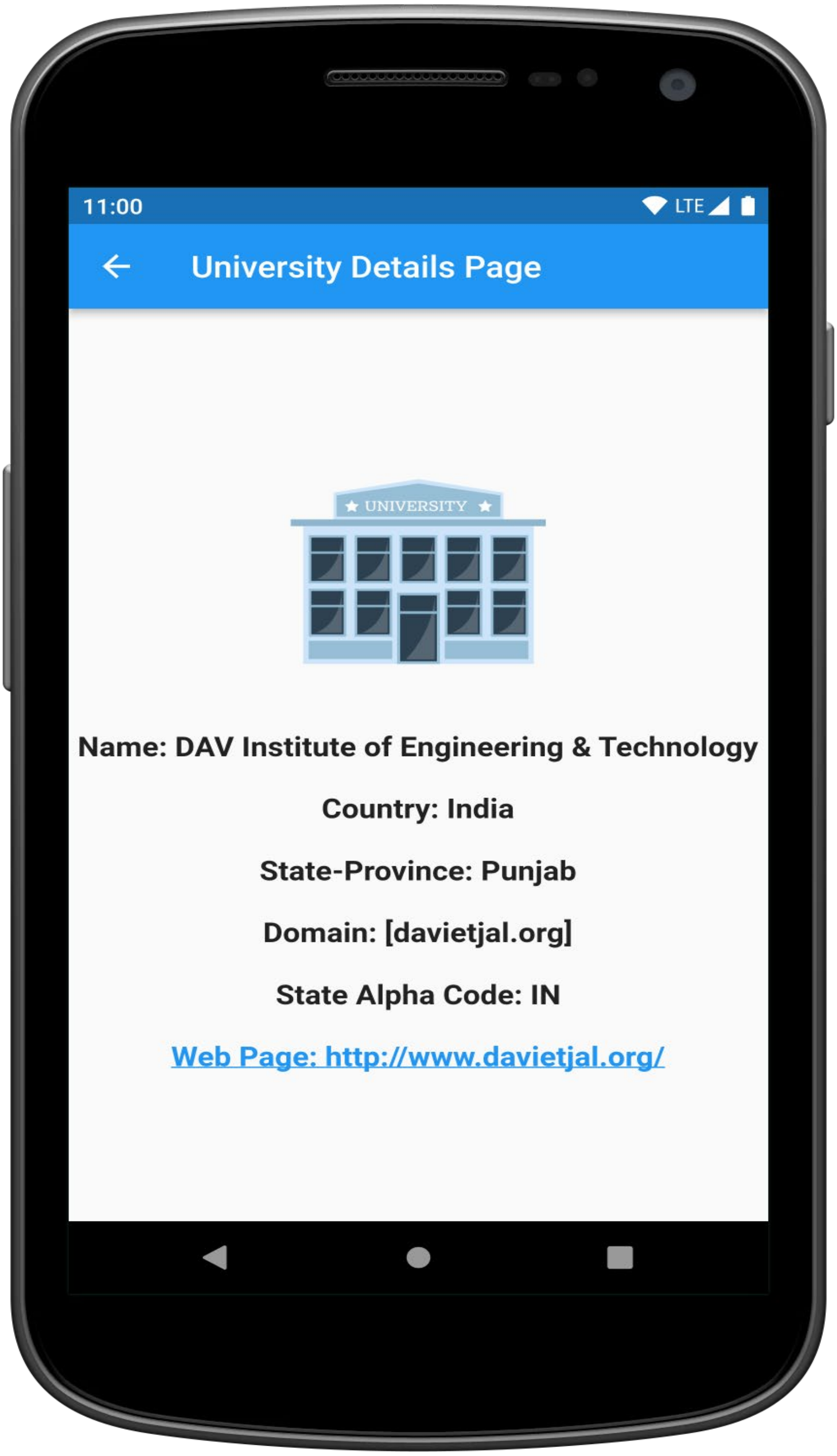
Dharamsinh Desai University

University of Health Sciences Andhra Pradesh

Allahabad Agricultural Institute, Deemed University

Assam Agricultural University

13.6 University Details Page:



University Details Page

Name: DAV Institute of Engineering & Technology

Country: India

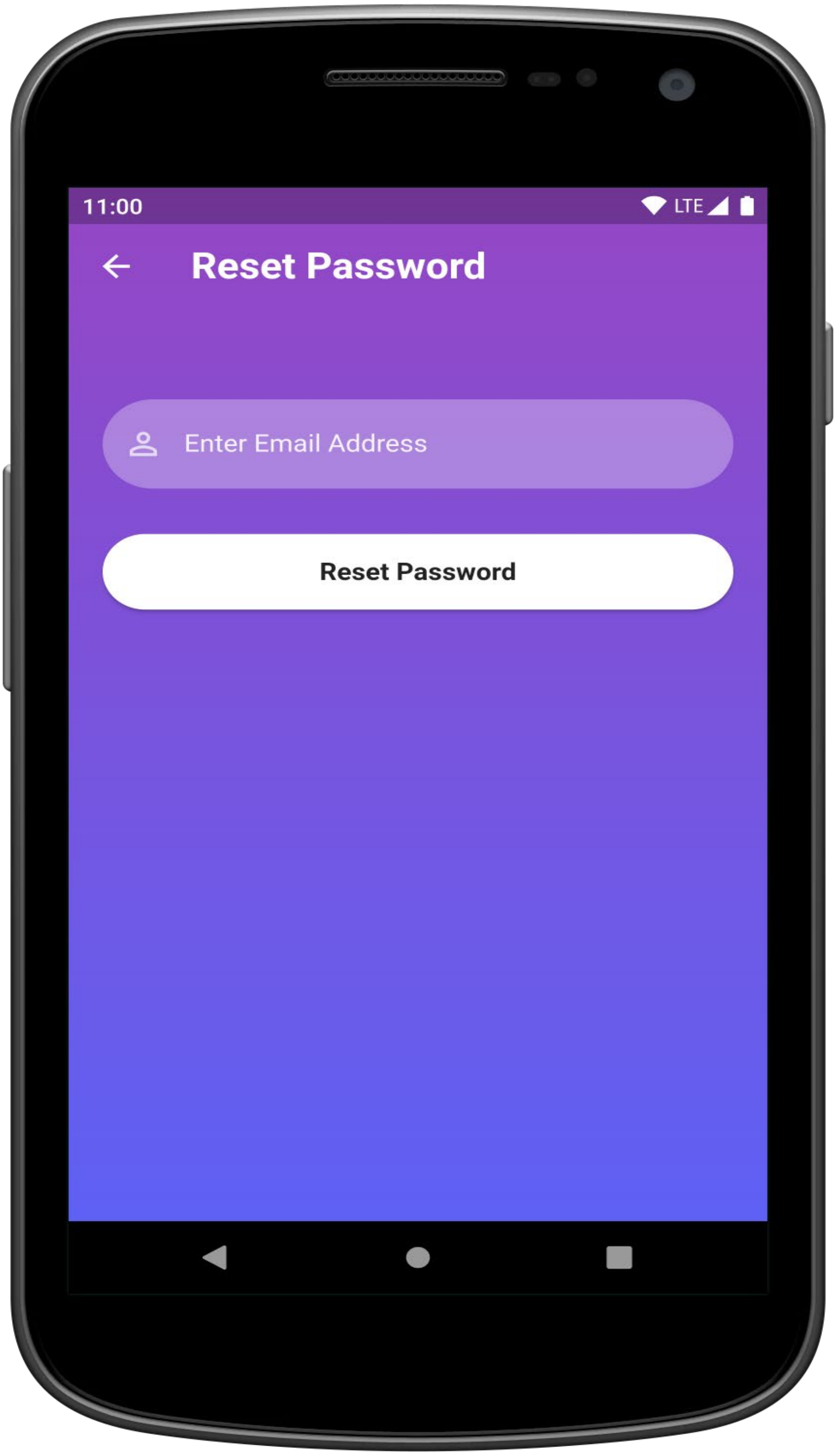
State-Province: Punjab

Domain: [davietjal.org]

State Alpha Code: IN

Web Page: http://www.davietjal.org/

13.7 Forgot Password Page:



14. Conclusion

The application includes a Login page, Home page with a list view of universities, search functionality, and a University Details page. Firebase was used for user authentication and storage. You can further enhance the application by adding additional features and improving the user interface.

15. Resources

Here are some resources you may find helpful while developing your Flutter application:

- **Flutter Documentation:** https://flutter.dev/docs
- **Firebase Documentation:** https://firebase.google.com/docs
- **Flutter HTTP package:** https://pub.dev/packages/http
- **Flutter ListView widget:** https://api.flutter.dev/flutter/widgets/ListView-class.html
- **Flutter Navigation and Routing:** https://flutter.dev/docs/development/ui/navigation
- **Flutter Cookbook:** https://flutter.dev/docs/cookbook