



PRESIDENCY UNIVERSITY

Private University Estd. in Karnataka State by Act No. 41 of 2013

School of Computer Science Engineering

**A Report on
Mini-Project
Titled
Interview App**

Course Title: Mobile Application Development

Course Code: CSE2506

Date of Submission: 21/11/2025

Submitted By:

Name	Roll No
MOHAN A	20231CSE0273
MANOJ KUMAR B M	20231CSE0296
DILIP D	20231CSE0263
DEEPAK G	20231CSE0301
KISHORE S V	20231CSE0260

1. Abstract

This project presents an Android-based **Interview Application** built using Java, XML, and SQLite as the local database. The app is designed to help users manage and track their interview practice sessions efficiently, while maintaining offline data accessibility. User authentication is implemented through SharedPreferences, enabling quick login/signup without requiring a remote server.

The system includes a structured **Home Page** where all scheduled or previously created interviews are listed, each with options to *attend* or *delete* the entry. Users can add new interviews through a dedicated **Add Interview** screen, which stores data persistently in SQLite. The **Dashboard** provides analytical insight into the user's performance, including total interviews attended, accuracy, and overall performance metrics. A **bar graph visualization** is integrated to present these statistics in a clear and measurable format. The app also offers a reliable **Logout** feature to maintain user privacy and session control.

Overall, this project demonstrates the practical use of SQLite, SharedPreferences, UI components, and data visualization in building a functional, user-centered Android application aimed at improving interview preparation and performance tracking.

2. Introduction

This Android-based Interview App is created to make that easier by giving users a clean and easy-to-use platform to manage and follow their interview practice sessions.

The application is built using **Java**, **XML**, and **SQLite**, ensuring fast performance and complete offline functionality. User login and signup are implemented with SharedPreferences to maintain a lightweight and efficient authentication mechanism tailored for single-device usage.

Core functionality revolves around the **Home Page**, where users can view, attend, or delete their stored interviews. The **Add Interview** module enables users to create new interview entries, which are securely stored in the SQLite database. To help users analyze their progress, a **Dashboard** displays key metrics including total interviews attended, accuracy levels, and overall performance, along with a **bar graph visualization** for deeper insight.

3. Software and Hardware Requirements

1. Software Requirements

1.1 Operating System

- Windows 10 / Windows 11
- macOS
- Linux (optional)

1.2 Development Tools

- Android Studio (latest stable version)
- Java Development Kit (JDK 8 or above)

1.3 Programming Languages

- Java – for application logic
- XML – for designing user interfaces

1.4 Database

- SQLite (Android's built-in local database)

1.5 Supporting Libraries and Frameworks

- AndroidX libraries
- SharedPreferences for user authentication data

1.6 Android SDK Requirements

- Minimum SDK: API Level 21 (Android 5.0)
- Target SDK: Latest available version

2. Hardware Requirements

2.1 Hardware for Development

- A desktop or laptop computer with:
 - Intel i3 processor or higher
 - Minimum 4 GB RAM (8 GB recommended)
 - At least 10 GB free storage space
 - Screen resolution of 1366×768 or higher
- Android device or emulator for testing:
 - Android 8.0 or above

2.2 Hardware for End Users

- Android smartphone
- Android OS version 8.0 or above
- Minimum 1 GB RAM
- 30–100 MB free storage to install and run the app

4. Modules/Functionalities

This Android Interview Application is divided into several modules. Each module focuses on a specific task to provide a smooth and organized user experience. The main modules and their functionalities are described below.

1. User Authentication Module

This module manages user login and registration.

- Users can create a new account using the signup screen.
- Login credentials are stored and verified using **SharedPreferences**.
- Successful login allows access to the home page.
- Ensures simple and secure session handling.

2. Home Page Module

This is the main screen after login and displays all the interviews created by the user.

Functionalities include:

- Viewing the list of stored interviews
- Option to **attend** an interview
- Option to **delete** an interview
- All data is loaded from the **SQLite database**

3. Add Interview Module

This module allows users to create and store new interview entries.

- User clicks on interview to into Home Page
- Data is validated and saved into the **SQLite database**
- Newly added interviews immediately appear on the Home Page

4. Attend Interview Module

This module handles the interview-taking process.

- User answers questions one by one
- Stores user's correct answers count
- Calculates performance and accuracy
- Saves results for dashboard analysis

5. Logout Module

- Clears saved login data from SharedPreferences
- Redirects the user back to the Login screen

6. Dashboard Module

This module shows the user's overall performance.

Main features include:

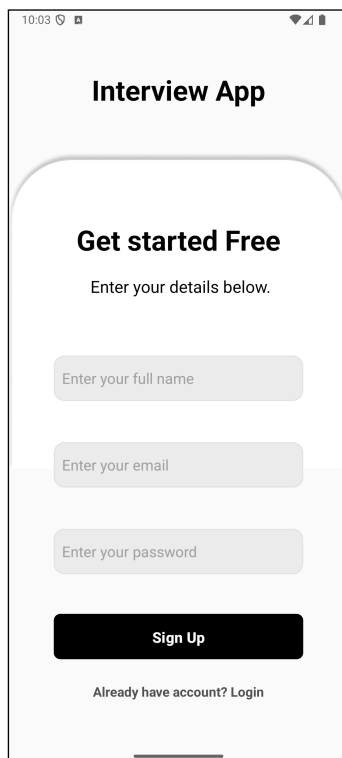
- Total interviews attended
- Accuracy percentage
- Performance score
- A **bar graph** for visual representation of all metrics
- Helps users track improvement over time

5. Results with Screenshots

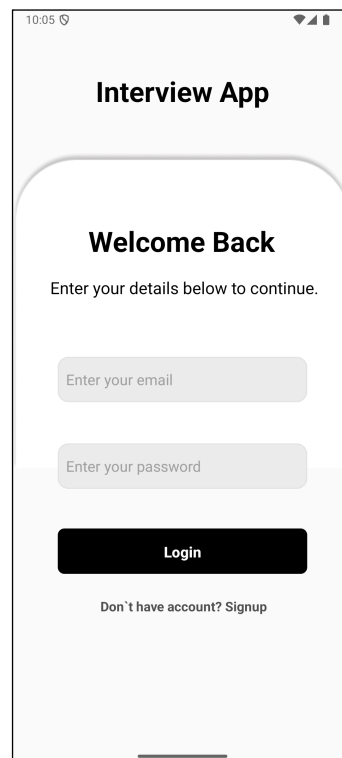
1. Signup Screen and Login Screen

The **Signup Screen** lets new users register and stores their details in SharedPreferences.

The **Login Screen** verifies entered credentials against the stored data and grants access if they match.



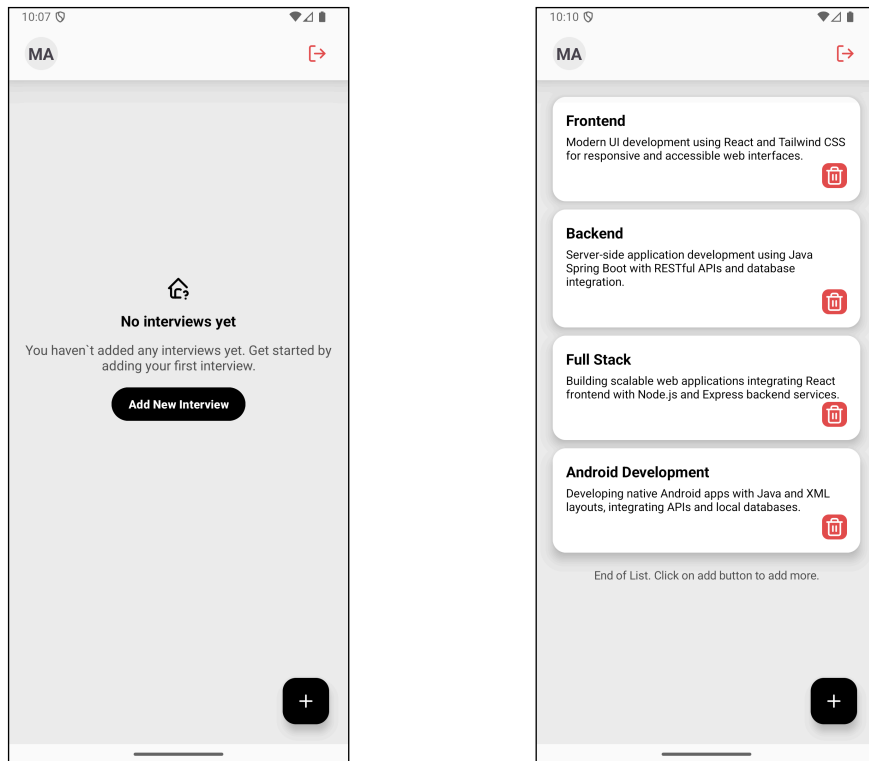
The screenshot shows the 'Interview App' title at the top. Below it is a rounded rectangle containing the text 'Get started Free' and 'Enter your details below.' There are three input fields: 'Enter your full name', 'Enter your email', and 'Enter your password'. At the bottom of the rounded rectangle is a black button labeled 'Sign Up'. Below the rounded rectangle is a link that says 'Already have account? Login'.



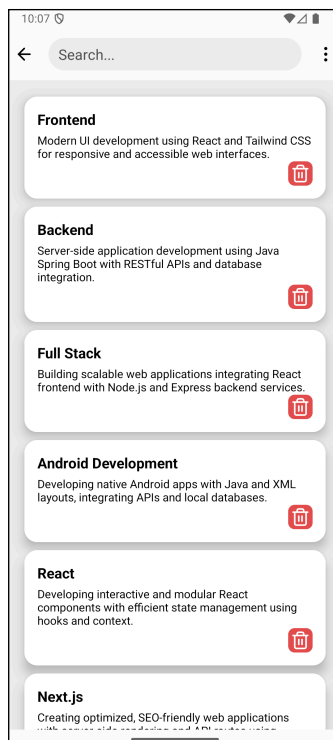
The screenshot shows the 'Interview App' title at the top. Below it is a rounded rectangle containing the text 'Welcome Back' and 'Enter your details below to continue.' There are two input fields: 'Enter your email' and 'Enter your password'. Below the rounded rectangle is a black button labeled 'Login'. At the bottom of the screen is a link that says 'Don't have account? Signup'.

2. Home Page

The Home Page shows all the interviews saved in the SQLite database.



3. Add Interview Screen



4. Attend Interview Screen

When the user attends an interview, questions are displayed one by one. The app calculates the number of correct answers and the performance.

10:08

Frontend
Modern UI development using React and Tailwind CSS for responsive and accessible web interfaces.

1. Which HTML tag is used to include JavaScript code?

- ☒ <script>
- ☐ <js>
- ☐ <javascript>
- ☐ <code>

2. Which CSS property controls text size?

- ☐ font-style
- ☒ font-size
- ☐ text-size
- ☐ text-style

3. Flexbox is used for creating responsive layouts. (True/False)

- ☒ True
- ☐ False

4. HTML stands for _____.
hyper text markup language

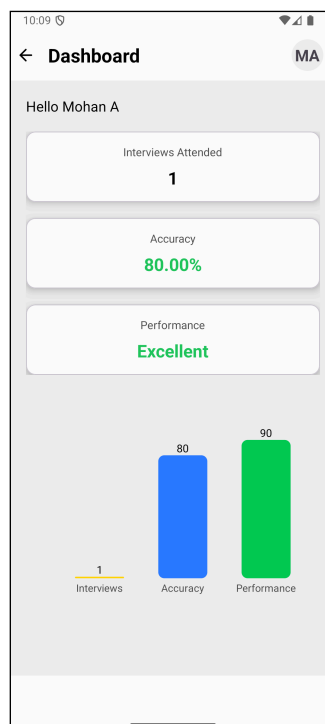
5. CSS stands for _____.
Cascading Style Sheets

SUBMIT

☰

6. Dashboard Screen

The Dashboard shows the user's performance summary



6. Conclusion

This Android Interview App successfully delivers a simple and effective system for managing and tracking interview practice. The app stores all interview data using SQLite, allowing it to work fully offline. User login and signup are handled through SharedPreferences, making authentication fast and lightweight.

The main features—Home Page, Add Interview, Attend Interview, and Dashboard—help users organize their interviews, practice questions, and view their performance. The Dashboard gives a clear summary of total interviews attended, accuracy, and overall performance through both text and bar graphs.

An additional useful feature is the notification system, which sends a notification when the user submits an interview. This improves user feedback and makes the app feel more interactive and responsive.

Overall, the project meets all its goals and shows practical use of Android components such as SQLite, SharedPreferences, Notifications, Activities, Intents, and chart libraries. It demonstrates strong understanding of mobile development, UI design, and data handling.