**MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO**

A logo of a university of engineering and technology

Description automatically generatedA logo of a university of engineering and technology

AI-generated content may be incorrect.

**Assignment**

**Submitted by**

**MUHAMMAD SAIF MEMON (22SW129)**

**FUZAIL HUSSAIN TALPUR (22SW132)**

**Section: II**

**Subject:** Mobile Application Development

**Submitted to: Mam Mariyam**

**Assessment Type: Complex Engineering Problem (CEP)**

**Project Title: CarConnect *– Car Rental & Driver Booking App***

**1. Problem Identification**

In our daily life, people often face difficulty when they need to rent a car or hire a driver. Normally, this process is very time-consuming because users have to visit multiple rental shops, make several phone calls, and negotiate prices in person.  
There is also no easy way for ordinary car owners to rent out their cars safely, and people who want to work as drivers find it hard to connect with potential clients.

The main problems identified are:

* Lack of a single, simple system for car rentals and driver hiring.
* Time and effort wasted in manual searching and communication.
* No verification or approval system to ensure trust and safety.
* Lack of digital record-keeping for bookings and user information.

This shows the need for a convenient mobile application that brings all these services together in one place and helps people connect quickly, safely, and efficiently.

**2. Proposed Solution**

To solve the above problem, we developed a **mobile application named CarConnect** using the **Flutter framework**.  
The app allows people to easily **rent cars**, **hire drivers**, and for **owners to list their vehicles** for others to rent.

The solution also includes an **Admin role**, who reviews and approves all new profiles and car listings before they are visible to users. This helps ensure that only verified cars, drivers, and owners appear in the app, improving reliability and safety.

All the data (users, cars, drivers, and bookings) are stored **locally using SQLite**, which allows the app to work without an internet connection during demonstration. The app also features a responsive interface that works smoothly on different screen sizes.

**Main Objectives**

* To provide a simple, all-in-one mobile platform for car rentals and driver hiring.
* To allow users to book cars with or without a driver.
* To let owners upload car details and rent them out.
* To add an admin verification layer for safer transactions.
* To save all data locally for quick access and offline demonstration.

**3. Project Features**

**For Users:**

* View available cars for rent.
* Search or filter cars by price, city, or type.
* Book a car with or without a driver.
* View and manage their booking history.

**For Car Owners:**

* Add new cars with model, price, and condition details.
* Edit or delete existing listings.
* Submit cars for admin approval before they appear to others.

**For Drivers:**

* Register as a driver with personal and contact details.
* Wait for admin approval before being listed.

**For Admin:**

* View and approve or reject newly registered users, cars, and drivers.
* Remove inappropriate or inactive listings.
* Maintain an overview of bookings.

**Other Features:**

* Simple and modern user interface.
* Responsive layout that adjusts for different mobile devices.
* Local data storage for quick performance.
* Real-time updates on the app whenever a user adds or edits data.

**4. Project Design and Interface**

The app uses Flutter’s built-in widgets for UI design. We kept the interface **clean, simple, and user-friendly**.

**Main Screens:**

**A screenshot of a login form

AI-generated content may be incorrect.A white background with pink and black lines

AI-generated content may be incorrect.A screenshot of a login screen

AI-generated content may be incorrect.A screenshot of a chat

AI-generated content may be incorrect.A white background with black text

AI-generated content may be incorrect.A screenshot of a phone

AI-generated content may be incorrect.A white background with pink and black lines

AI-generated content may be incorrect.A screenshot of a login form

AI-generated content may be incorrect.**

**5. Data Storage (SQLite)**

The app uses **SQLite** as the local database to store all data securely on the user’s device.  
SQLite is a lightweight, reliable, and widely used relational database system that works entirely offline, meaning it does not need an internet connection or an external server to operate. This makes it an ideal choice for mobile applications like ours that focus on smooth, fast, and offline functionality.

**Data Stored in the Database**

| **Table** | **Description** | **Fields** |
| --- | --- | --- |
| **Users** | Stores information of all app users (customers, car owners, and admin) | id, name, email, role (user/owner/admin), approved |
| **Cars** | Contains details of cars listed for rent | id, owner\_id, model, rent\_per\_day, city, available, approved |
| **Drivers** | Stores profiles of all registered drivers | id, name, experience\_years, rate\_per\_hour, approved |
| **Bookings** | Stores all booking records made by users | id, user\_id, car\_id/driver\_id, date, duration, status |

Each record is stored locally and displayed instantly after being created. This means users can add or update information and see the changes immediately without waiting for server responses or internet access.

The app connects with the SQLite database through the **sqflite** package in Flutter, which provides simple functions to insert, update, delete, and fetch data efficiently.

**Justification for Choosing SQLite**

SQLite was chosen for this project due to the following reasons:

1. **Offline Functionality:** It works completely offline, allowing users to access and modify data without an internet connection.
2. **Simplicity:** SQLite is easy to integrate into Flutter and does not require complex setup or backend services.
3. **Speed and Performance:** It is very fast for local data storage and retrieval, making the app smooth and responsive.
4. **Security:** Data is stored locally on the user’s device, reducing the risk of network-related vulnerabilities.
5. **Reliability:** SQLite is a proven, stable database technology used in most mobile systems (including Android and iOS).
6. **No Additional Cost:** Since SQLite comes built-in with most mobile platforms, there are no hosting or server expenses.

**6. Technologies and Tools Used**

| **Tool / Technology** | **Purpose** |
| --- | --- |
| **Flutter** | Main development framework for cross-platform app |
| **Dart** | Programming language used with Flutter |
| **Sqllite** | Local database to store all data offline |
| **Android Studio** | IDE used for writing and testing code |
| **image\_picker package** | Allowing users to upload NIC and car images |

**7. Issues and Bugs Encountered**

| **Issue** | **Description** | **Solution** |
| --- | --- | --- |
| Build errors due to NDK version | Flutter plugins required updated Android NDK | Updated build.gradle with correct ndkVersion |
| Asset directory errors | assets/ folder not found | Created folder and fixed YAML indentation |
| UI overflow on small screens | Some widgets overflowed | Used SingleChildScrollView and flexible widgets |
| Path provider error on web | Plugin not supported for web | Ran app only on Android device |
| State not refreshing after updates | Data didn’t appear instantly | Used setState and Provider for UI refresh |