­­­­­



Scrum Master

Binumon Joseph

Assistant Professor  
Department of Computer Applications

INSTHELPER

20INMCA509 - Mini Project 2

Ajul K Jose

AJC20MCA-I004

INMCA2020-25 S9

<https://github.com/ajulkjose246/InstHelper>

[mail.ajulkjose@gmail.com](mailto:mail.ajulkjose@gmail.com)

ABSTRCT

InstHelper is an advanced Flutter Android app aimed at revolutionizing college campus operations through automation and innovative technology integration. The app empowers users with seamless access to a range of functionalities, including Smart Parking, Smart Inventory, Smart Library, Room Navigation, Guest Room Booking, Stock Information, and College Vehicle Management, directly from their smartphones. Utilizing QR code technology, InstHelper enables efficient management of parking spaces, inventory items, navigation, and vehicle details, providing real-time updates to enhance user convenience. One of the standout features of InstHelper is the incorporation of Mixed Reality (MR) within the Smart Library module. This feature leverages augmented reality (AR) to provide immersive navigation experiences, guiding users to locate specific sections and books effortlessly. The MR option enriches the traditional library visit, making it interactive and engaging. For administrators, InstHelper offers a robust web interface to monitor and analyze various metrics. Through this dedicated website, administrators can view graphical representations of data, such as parking usage, inventory status, room bookings, and vehicle information, enabling data-driven decision-making and efficient campus management. Additionally, the College Vehicle Management module allows drivers to add and manage details of college-owned vehicles, including insurance, test dates, and maintenance schedules, while administrators can view and monitor this information. This ensures the efficient management and upkeep of college vehicles, enhancing overall operational efficiency.

MODULES:

1. Authentication: Manages user registration, login, and role-based access control.
2. Smart Parking: Provides QR code-based parking space allocation and real-time availability updates.
3. Smart Inventory: Tracks inventory items like room keys using QR codes and real-time status updates
4. Smart Library: Offers mixed reality navigation to help users locate sections and books within the library
5. Room Navigation: Assists users in finding specific rooms on campus using indoor maps and QR codes
6. Guest Room Booking: Enables users to view and book available guest rooms within the college
7. Stock Information: Displays real-time information on available stock items in the college bookstall
8. Admin Dashboard: Provides graphical data representation and real-time updates for administrators

USERS:

1. Students: Utilize features like smart parking, inventory management, library navigation, room navigation, guest room booking, and viewing stock information
2. Faculty: Access smart parking, inventory management, library navigation, room navigation, and stock information for efficient campus management
3. Administrators: Oversee all system modules, manage resources, analyze data through the admin dashboard, and ensure overall system security
4. Visitors (Parents, Guests) : Use Smart parking, room navigation and guest room booking features

REQUIREMENT GATHERING

1. **Project Overview**:
   * InstHelper aims to revolutionize college campus operations through automation and advanced technology integration. The project addresses common administrative and logistical challenges faced by college campuses such as managing parking spaces, inventory, library navigation, room navigation, guest room booking, stock information and vehicle management. The main objectives are to increase efficiency, improve user convenience and provide real-time updates.
2. **System Scope**:
   * The system is proposed as a full-scale implementation for college campuses. It will integrate multiple functionalities into a single platform accessible via a mobile application and an admin web interface
3. **Target Audience**:
   * Students
   * Faculty
   * Administrators
   * Visitors
   * Parents
   * Drivers
4. **Modules**:
   * Smart Parking: Locate available parking spaces using QR codes.
   * Smart Inventory: Manage and track inventory items using QR codes.
   * Smart Library: Provide mixed reality navigation for easy access to library sections and books.
   * Room Navigation: Assist users in locating classrooms and other facilities.
   * Guest Room Booking: Enable booking of guest rooms for visitors.
   * Stock Information: Display available stock information in the college book stall.
   * College Vehicle Management: Manage vehicle details, insurance, and test dates.
5. **User Roles**:
   * Students: Access information on parking, library navigation, room navigation, stock information, and guest room booking
   * Faculty: Similar access as students, with additional capabilities for managing inventory
   * Administrators: Full access to all functionalities, including vehicle management and admin web interface for data monitoring.
   * Visitors: Access to guest room booking and room navigation
   * Parents: Access to guest room booking and room navigation.
   * Drivers: Manage and update vehicle details.
6. **System Ownership**:
   * The system is owned by the academic institution (college/university)
7. **Industry/Domain**:
   * Education
8. **Data Collection Contacts**:
   * Amal K Jose

* Role: Assistant Professor
* Mail: [amalkjose324@gmail.com](mailto:amalkjose@amaljyothi.ac.in)
* Contact: 94964 40324
  + Nisha E C
* Role: AES Software Developer
* Contact: 82813 41319
  + Aleena Joseph
* Role: AES Software Developer
* Contact: 8086025320
  + Jaison Joseph
* Role: Driver cum Office Assistant
* Contact: 9495313766

1. **Questionnaire for Data Collection**:
   * What are the current challenges faced in managing parking spaces on campus?
   * How is the inventory currently managed, and what improvements are needed?
   * What features would you like to see in a smart library navigation system?
   * How do visitors currently navigate the campus, and what difficulties do they encounter?
   * What is the current process for booking guest rooms, and what issues arise?
   * How is stock information in the book stall managed, and what improvements are needed?
   * How are college-owned vehicles currently managed, and what details are tracked?
   * What specific vehicle information should be included in the management system (e.g., insurance, test dates)?
   * How do you currently notify users of upcoming expiries or service due dates for vehicles?
   * What type of graphical information would be most useful for administrators in the web interface?

FEASIBILITY STUDY

Feasibility study is an important phase in software development process. It enables developers to have a clear picture of the product being developed in terms of outcomes of the product, operational requirements for implementing it, etc. A feasibility study is conducted to determine whether the project will, upon completion, fulfil the objectives of the organization in relation to the work, effort, and time invested in it. As a result, a new application often undergoes a feasibility assessment before approved for development. A feasibility study enables the developer to predict the projects usefulness and potential future. An evaluation of a system proposals viability takes into account its impact on the organization, capacity to satisfy user needs, and efficient use of resources.

Various feasibility studies are:

* Technical Feasibility
* Operational Feasibility
* Economic Feasibility

**Technical Feasibility**

The Vehicle Management System (VMS) project is technically feasible due to the college's adequate IT infrastructure, which includes robust servers and network capabilities to support the application. The development team possesses the necessary expertise in Flutter, backend technologies, and database management. Utilizing Flutter for cross-platform development ensures broad compatibility and benefits from a large, active support community. This stable and well-supported technology stack will facilitate the successful implementation of the VMS.

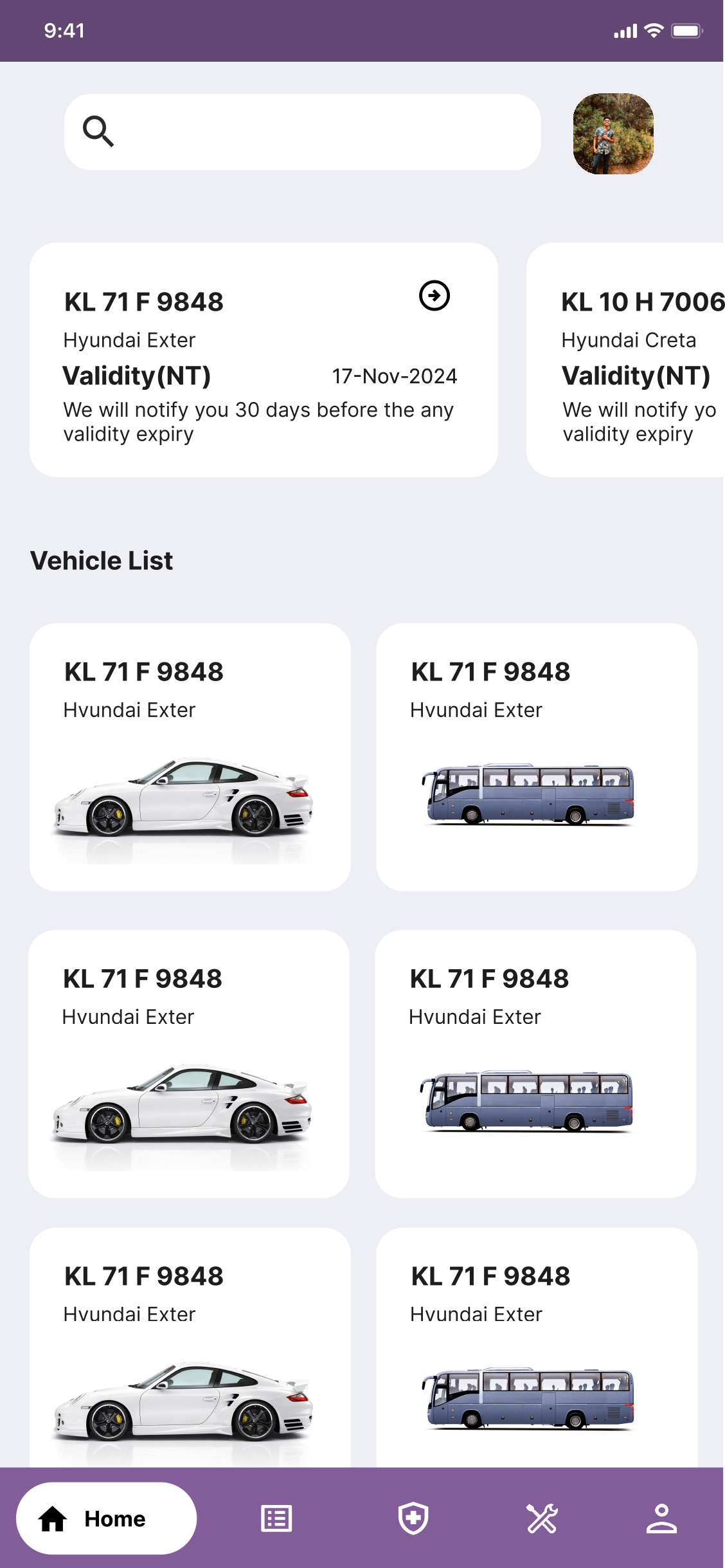
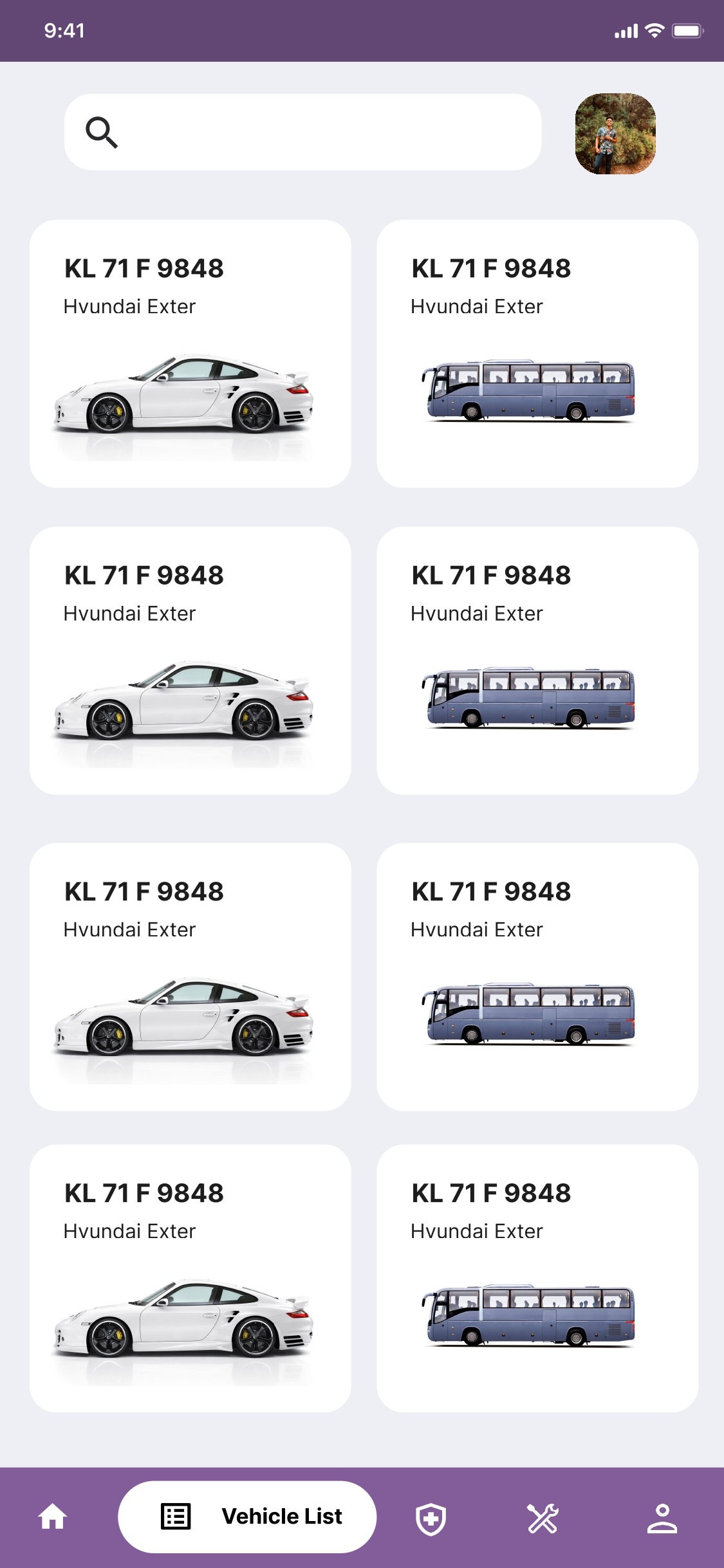
**Operational Feasibility**

The VMS is feasible and poised for smooth adoption within the college. Initial feedback from stakeholders shows high acceptance and enthusiasm for the system. To ensure smooth integration and usage, training sessions will be conducted for users. The system will seamlessly integrate with existing campus databases and infrastructure, minimizing disruptions while leveraging current resources.

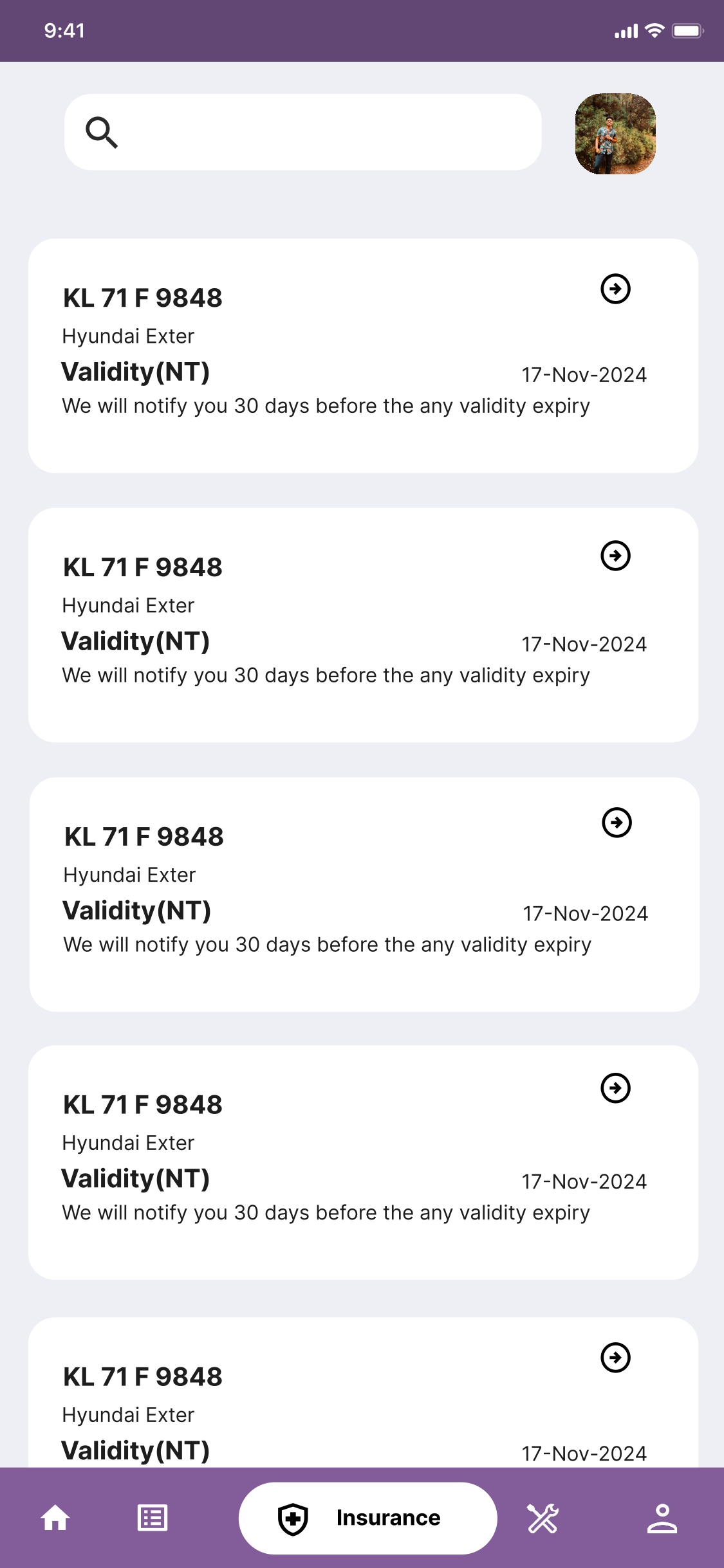
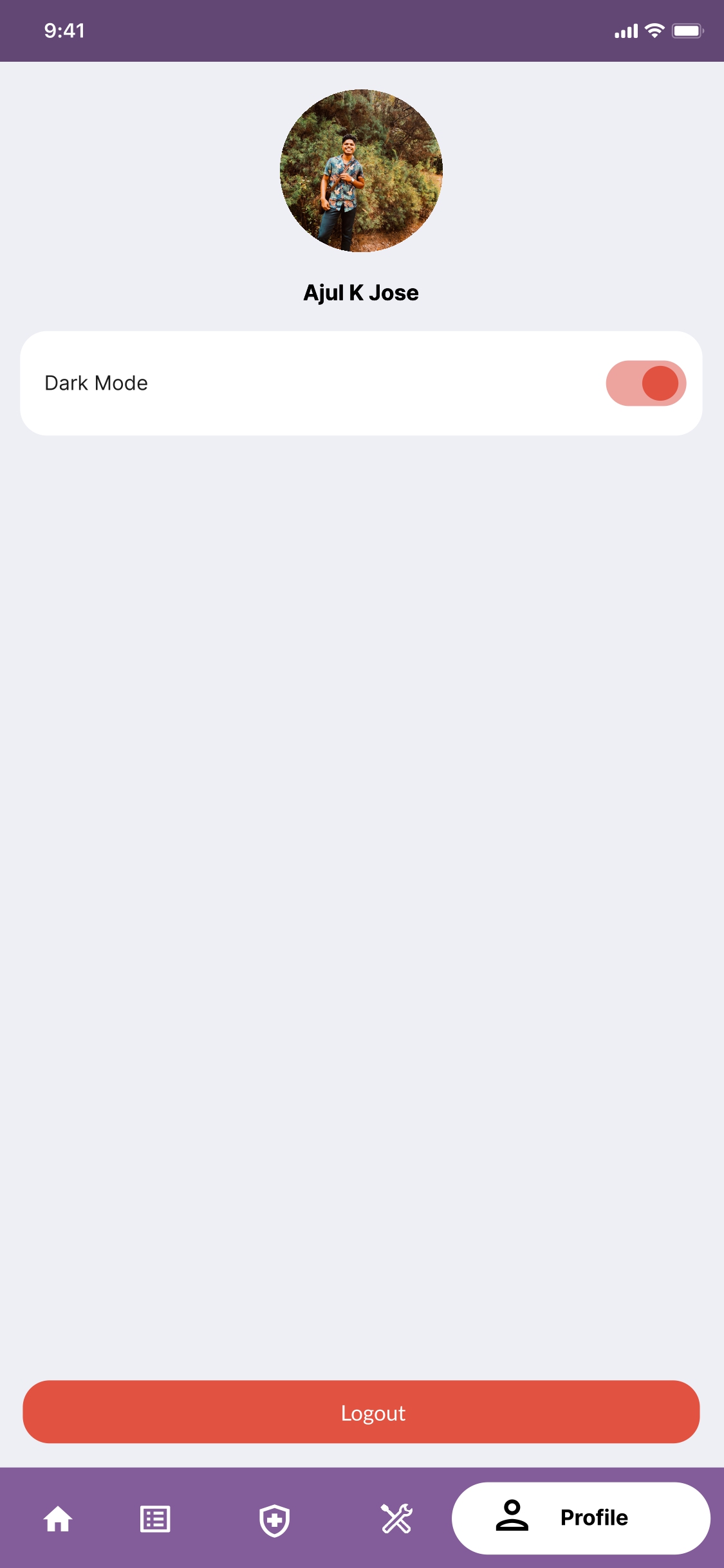
**Economic Feasibility**

The Vehicle Management System (VMS) project presents a viable and cost-effective investment. The total estimated cost for development and deployment, including necessary hardware and software purchases, is modest, ensuring that it falls within a low budget. The anticipated benefits are substantial: increased operational efficiency, reduced administrative workload, and enhanced user satisfaction. These improvements are expected to lead to long-term savings and better resource utilization, effectively offsetting the initial costs.

USER INTERFACE DESIGN USING FIGMA

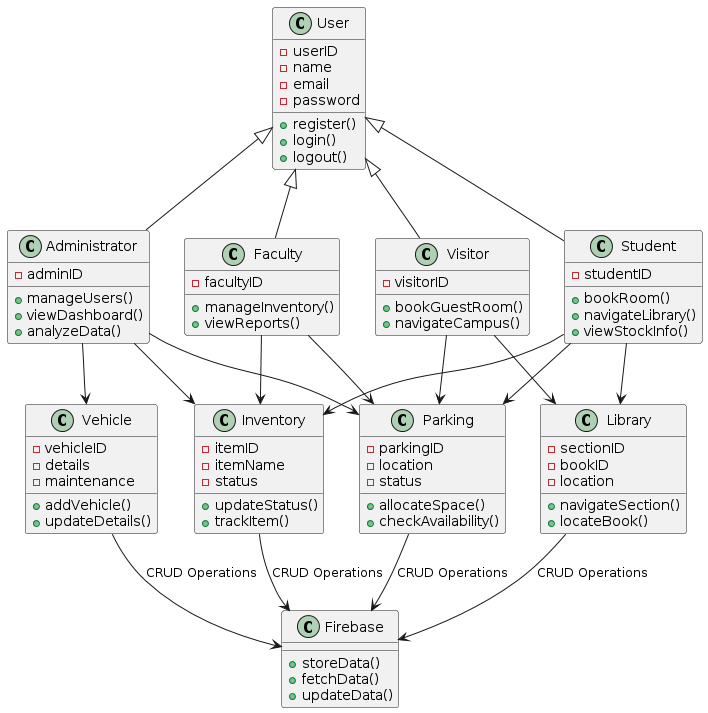
Home Vehicle List

­ 

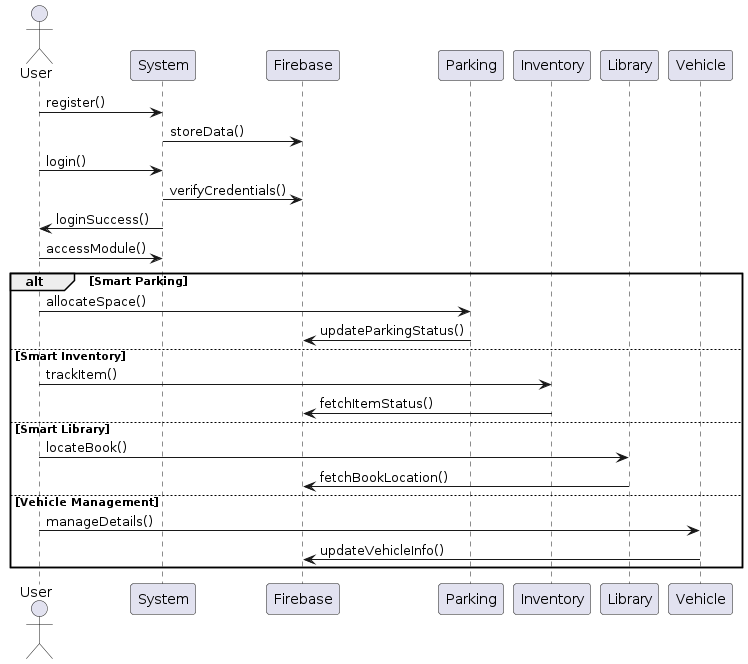
Insurance Profile

UML DIAGRAM

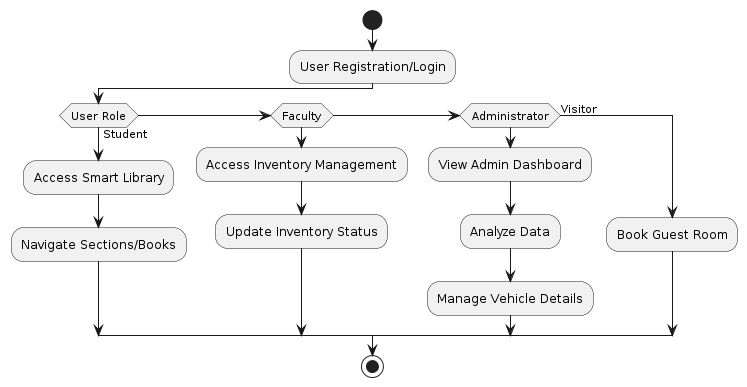
**Class Diagram**

****

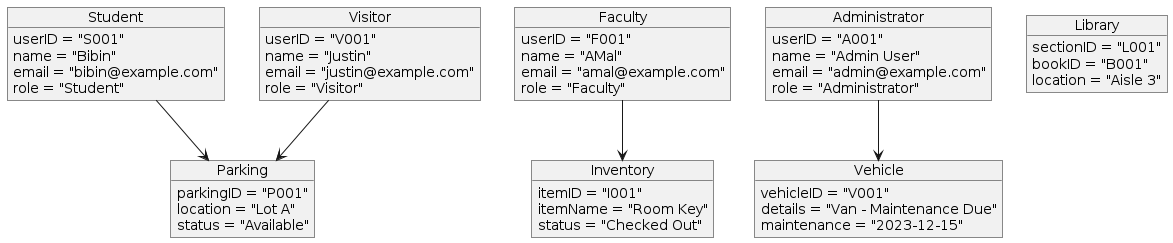
**Sequence Diagram**

****

**Activity Diagram**

****

**Object Diagram**

****

**Use Case Diagram**

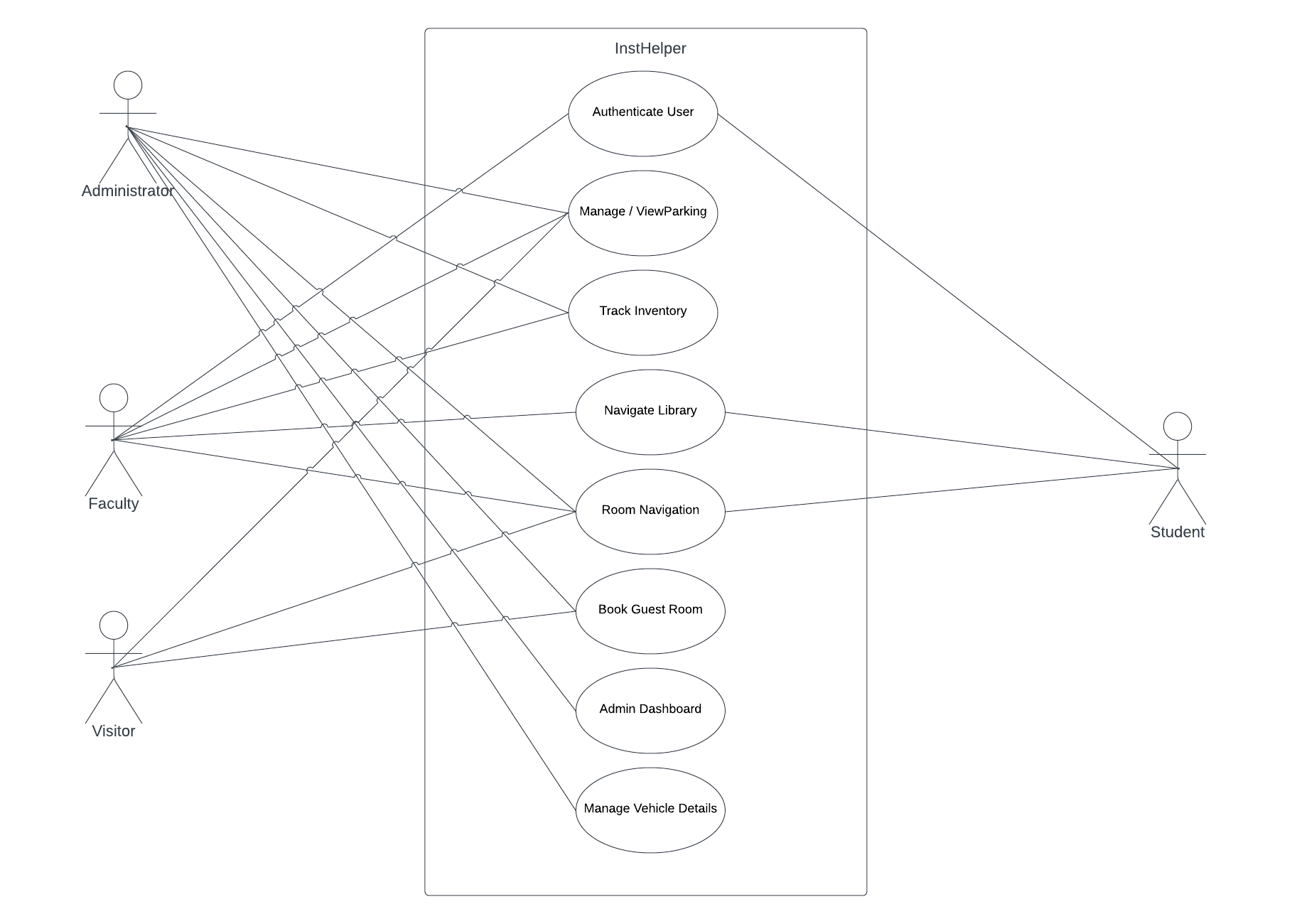
****

TABLE DESIGN

**1.tbl\_login**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Fieldname | Datatype | Key | Constraints | Description |
| 1 | log\_id | int(11) | Primary key | Primary key | Primary key in the table |
| 2 | usr\_id | int(11) | Foreign key | References usr\_id in tbl\_register | References user registration |
| 3 | usr\_email | varchar(30) |  | Not null | User Email ID for login |
| 4 | usr\_password | varchar(100) |  | Not null | User Password for login |
| 5 | usr\_status | int(11) |  | Not null | Role of the User |

### **2.tbl\_register**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Fieldname | Datatype | Key | Constraints | Description |
| 1 | usr\_id | int(11) | Primary key | Primary key | User ID |
| 2 | usr\_name | varchar(50) |  | Not null | User Name |
| 3 | usr\_contact | varchar(15) |  | Not null | Contact Number |
| 4 | usr\_address | varchar(255) |  |  | Address of the User |

### **3. tbl\_vehicle**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Fieldname | Datatype | Key | Constraints | Description |
| 1 | vehicle\_id | int(11) | Primary key | Primary key | Vehicle ID |
| 2 | reg\_number | varchar(20) | Unique | Not null | Vehicle Registration Number |
| 3 | owner\_id | int(11) | Foreign key | References usr\_id in tbl\_register | Owner's User ID |
| 4 | vehicle\_type | varchar(20) |  | Not null | Type of Vehicle |
| 5 | model | varchar(50) |  | Not null | Model of the Vehicle |
| 6 | chassis\_no | varchar(50) |  |  | Chassis Number |
| 7 | engine\_no | varchar(50) |  |  | Engine Number |
| 8 | fuel\_type | varchar(10) |  |  | Type of Fuel |
| 9 | current\_mileage | int(11) |  |  | Current Mileage |
| 10 | registration\_date | date |  |  | Date of Registration |
| 11 | fitness\_upto | date |  |  | Fitness Certificate Expiry Date |
| 12 | insurance\_upto | date |  |  | Insurance Expiry Date |
| 13 | pollution\_upto | date |  |  | Pollution Certificate Expiry Date |
| 14 | emergency\_contact | varchar(15) |  |  | Emergency Contact Number |
| 15 | assigned\_driver | varchar(50) |  |  | Assigned Driver |
| 16 | ownership | varchar(20) |  |  | Ownership Type (Individual/Company) |
| 17 | purpose\_of\_use | varchar(50) |  |  | Purpose of Use |

### **4. tbl\_rto**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Fieldname** | **Datatype** | **Key** | **Constraints** | **Description** |
| 1 | rto\_code | varchar(10) | Primary key | Primary key | RTO Code |
| 2 | rto\_name | varchar(50) |  | Not null | RTO Name |

### **5. tbl\_models**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Fieldname** | **Datatype** | **Key** | **Constraints** | **Description** |
| 1 | model\_id | int(10) | Primary key | Primary key | Vehicle Model Id |
| 2 | model\_name | varchar(50) |  | Not null | Vehicle Model Name |
| 3 | Model\_image | varchar(50) |  | Not null | Vehicle Model Image |

### **6. tbl\_fuelType**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Fieldname** | **Datatype** | **Key** | **Constraints** | **Description** |
| 1 | fuel\_id | int(10) | Primary key | Primary key | Vehicle fuel |
| 2 | fuel\_name | varchar(50) |  | Not null | Vehicle fuel Name |

### **7. tbl\_drivers**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Fieldname** | **Datatype** | **Key** | **Constraints** | **Description** |
| 1 | driver\_id | int(10) | Primary key | Primary key | Driver Id |
| 2 | driver\_name | varchar(50) |  | Not null | Vehicle driver Name |
| 3 | driver\_contact | varchar(50) |  | Not null | Vehicle driver Contact |
| 4 | driver\_license | varchar(50) |  | Not null | Vehicle driver licence No |