NTU Ride Pilot

24-FYP-204

Session 2021-2025

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

SUBMITTED BY

Abubakar Nadeem	21-NTU-CS-1291
Imran Ali Niaz	21-NTU-CS-1321
Huzaifa Ahmad	21-NTU-CS-1318

SUPERVISED BY

Supervisor: Zahid Javed

Co-Supervisor: Muhammad Naeem

Department of Computer Science

National Textile University, Faisalabad

Certification

This is to certify that this project titled "**NTU Ride Pilot**" was found to satisfy the requirement for the award of a "**Bachelor of Sciences in Software Engineering**" degree by the Department of Computer Science, National Textile University.

Supervisor
Zahid Javed

Signature	Date

Co-Supervisor
Muhammad Naeem

Signature	Date

External Examiner

Signature	Date

Internal Examiner

Signature	Date

FYP-Convener

Signature	Date

Head of Department
Dr.

Signature	Date

Declaration

We hereby declared that this document is completely written by us, and it is totally our effort and none of anyone from outside of our group has copied it. This Report is purely written technically in accordance with our project.

Group Members:

Abubakar Nadeem _____ _____
21-NTU-CS-1291 Signature Date
E-Mail: abubakaranjum066@gmail.com

Imran Ali Niaz _____ _____
21-NTU-CS-1321 Signature Date
E-Mail: imranaliniaz786@gmail.com

Huzaifa Ahmad _____ _____
21-NTU-CS-1318 Signature Date
E-Mail: huzaifa.ahmad486@gmail.com

Acknowledgement

First of all, In the Name of Allah, the Most Beneficent, the Most Merciful. All the praises and thanks be to Allah. A lot of love for our beloved Holy Prophet MUHAMMAD (S.A.W), his guidance always helps us to get the right path. After that, the preparation of this Software Requirements Specification (SRS) document was a collaborative effort involving Abubakar, Imran, and Huzaifa. The author would like to acknowledge and express gratitude to all individuals who directly or indirectly contributed to the completion of this document. Special thanks are extended to our Supervisor Zahid Javed and our Co-Supervisor Muhammad Naeem, who provided invaluable guidance and support throughout the SRS preparation process. The author would also like to acknowledge the invaluable support and suggestions provided by Zahid Javed. This SRS document would not have been possible without the support and cooperation of all involved individuals. The author is confident that this document will serve as a comprehensive guide for the development and implementation of the NTU Ride Pilot.

Signed by:

Abubakar Nadeem,

Imran Ali Niaz,

Huzaifa Ahmad,

Date:

00/00/2025

Abstract

Efficient and secure transportation is vital for educational institutions, where thousands of students rely on daily commutes. However, current systems face significant challenges, including inefficiencies, security vulnerabilities, and a lack of real-time tracking and communication. Issues such as overcrowding, unauthorized access, and poor resource utilization arise when students board buses without proper verification. Additionally, the absence of real-time bus monitoring and effective communication channels leads to confusion, delays, and suboptimal transport management.

The *NTU Ride Pilot* addresses these challenges by integrating advanced technologies like RFID-based ID verification, live GPS tracking, and automated communication tools. The system ensures only authorized students access transport services, provides real-time visibility of bus locations for students, parents, and administrators, and enables timely updates regarding delays, route changes, or emergencies. It also tracks driver performance, monitors bus occupancy, and generates insights for optimizing route planning and capacity utilization.

By modernizing transport operations, the *NTU Ride Pilot* enhances safety, boosts operational efficiency, and delivers a reliable and user-friendly commuting experience. This solution is an ideal choice for educational institutions aiming to transform their transportation infrastructure with secure and efficient technology.

Table of Contents

List of Figures

Description **Page #**

List of Tables

Description **Page#**

List of Abbreviations

NTURP	NTU Ride Pilot
RFID	Radio Frequency Identification
NTURP system	Admin Panel, Driver/Student App

CHAPTER 1

1 Introduction

Proper transportation is a requirement in school systems of all types, where many thousands of students face daily transportation needs. But many institutions have problems such as ineffective workflow, potential and noted security threats, no real-time control and messaging systems. Children board buses without identification hence self-aggregation, trespass and misuse of transport means are common as are overcrowding. Further, there is the lack of real time tracking in buses as well as the performance of the drivers, and poor communication channels leads to confusion and poor control and co-ordination of transport operations.

To counter these challenges, we are optimistic about the implementation of the proposed NTURP system in an educational setting. This system incorporates high technologies such as RFID used for the identification process, real-time tracking using GPS, and technology-reliant alerting mechanisms for better transport logistics management. How it helps: It ensures that only approved students gain access to students' transportation services, gives real-time mapping of the bus locations to students, parents, and administrative staff and assists in timely notification on any … delays, route alterations, or emergent situations. Further, it measures driving behaviour, controls the passenger load factor, and provides notifications for route efficiency and capacity management.

This solution does not address objectives present in the current status quo; it revolutionizes transport management turning it into a safe, optimised, and friendly environment for users. With the help of new technologies, the NTURP system increases security, optimizes the work of transport and ensures its safe functioning, which makes it a perfect choice for any school with the desire to create modern transport infrastructure.

1.1 NTU Ride Pilot

NTURP is new approach for improving transportation services and gradually making transport secure and safe for education institutions. It encompasses up-to-date solutions such as RFID for persons' identification, real-time GPS navigation, and an enhanced communication system to build an easy-to-navigate transportation environment. This system enables authorised institutions to finally determine the best routes for bus transport, the best capacity to be provided for each transport at a given period as well as the performance of the transporters in ensuring that students, parents and other members of the institution get secure and efficient means of transport.

1.2 Reason to Develop

The development of the NTURP system is driven by several compelling factors, despite the existence of other transportation solutions:

1.2.1 Addressing Unique Institutional Needs

Schools are bound to face certain problems such as intrusion, traffic congestion and improper signalling. These specific requirements cannot be served efficiently by generic transport management systems, and thus the need to develop a transport management solution for this environment specifically.

1.2.2 Enhancing Safety and Security

This is important to avoid insecurity or that some students who are not using transport facilities benefit as others who deserve it are locked out. RFID approach encompasses accurate and almost instantaneous means of reducing misuse and increasing security for both students and drivers.

1.2.3 Bridging Communication Gaps

Failure to share information with other transport departments, with parents and students results to disorganization. This presented system entails use of instantaneous notification for news such as delay, change of route and often emergencies to promote timely and precise communication.

1.2.4 Optimizing Resource Management

Overcrowding or underutilization of buses is a common issue. By integrating occupancy monitoring and route optimization, this system helps administrators allocate resources more efficiently, reducing costs and enhancing comfort.

1.2.5 Scalability and Adaptability

The system can suit the requirements necessary to smoothly operate with fleet in any type of educational institution regardless of the scale of transportation system present there. It is also scalable where the institutions will be able to increase its benefiting as it responds to the changing transport needs.

This is much more than a technology enhancement project; it is a transformational project across the transport system that takes the future of transport safety and efficiency into consideration in an educational environment.

1.3 Problem Statement

Public and private learning institutions are among the organizations that experience high levels of challenges when it comes to transportation management. Some of the widespread problems are overcrowded or, on the contrary, underfilled buses, theft, poor communication with the bus drivers, and absence of the possibility of tracking a bus's location. Such issues thus complicate movement, organization, and functioning, and pose risk to the learners, their parents, and school management. Currently available solutions do not always have the architecture and flexibility to meet these specific institutional requirements.

1.4 Purpose

The primary objective of the developed NTURP system is to become a one-stop for all the transport management needs of educational establishments. Through secure access verification, GPS tracking while the bus is in operation, and other communication features, the developers of the system intend to enhance the performance, safety, and overall communication into the student, parents, school, and bus company.

1.5 Project Goals

- Implement secure ID verification to prevent unauthorized access.
- Provide live bus location tracking for real-time visibility and improved coordination.
- Enable real-time notifications for updates such as delays, route changes, or emergencies.
- Track key metrics like speed, stop intervals, and adherence to schedules.
- Streamline bus scheduling, capacity management, and route planning.
- Ensure an intuitive interface for administrators, parents, and students.

1.6 Objectives

Objectives of the project are as follows:

- Integrate RFID-based ID verification to ensure only authorized users board the buses.
- Provide GPS-enabled tracking for buses accessible to students, parents, and administrators.
- Enable alerts for overcrowding or underutilization to optimize bus capacity.

- Monitor driver behaviour to ensure adherence to safety and efficiency standards.
- Develop a mobile app for notifications and updates to keep all stakeholders informed.

1.7 Project Scope

Even though this system is created for educational institution it can be expanded at other domains including corporate transport, public streamlined or private bus lines. The architecture of FMGO enables flexibility and expansion depending on the transportation needs and its functionality and organization.

1.8 Proposed Solution

The system provides efficient and well-structured functional solutions for transportation management. The solution offers RFID for ID check during entry, GPS tracking for buses in real time, alert on occupancy, checking of driver performance, and mobile applications. Through this system, there will be highly improved efficiency in the running of institutions, minimized wastage of resources, and improved safety and reliability of transport for all users

1.9 Project Scheduling

Below is the Gantt chart that has been developed for the NTURP project. This chart is intended to illustrate the project's schedule: the time when each activity was planned/started and the time when it was planned/ended. They give a broad plan of how the various tasks in the project are expected to be done, and when thus help in keeping track of the project. The time plan of the project is illustrated in the Gantt chart in Figure 1.1 below.

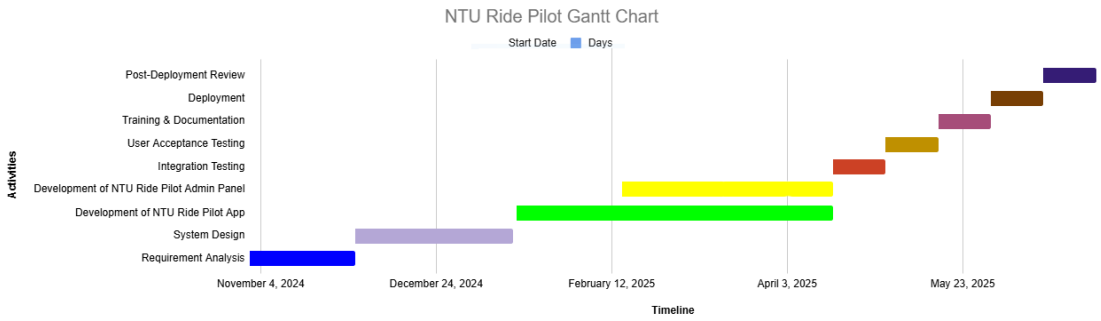


Figure 1.1 Gantt Chart

CHAPTER 2

2 Literature Review

The NTURP aims to enhance transportation services for students and parents by integrating technologies such as RFID scanning, GPS tracking, real-time data analytics, and mapping services. This chapter reviews existing literature and technologies pertinent to the project, including RFID technology in transportation systems, GPS-based bus tracking, load management in public transportation, predictive arrival systems, and the utilization of mapping APIs like Google Maps and Map box. Additionally, it examines existing student transportation management systems to identify current solutions and gaps.

2.1 Related Work

Several transportation management systems cater to student transit needs, offering features like GPS tracking, route optimization, and parent communication. Notable examples include:

2.1.1 Tyler Technologies' Student Transportation Software

Provides integrated solutions for bus routing, fleet maintenance, and parent communication, connecting various aspects of transportation management.
Cons:

- High implementation and licensing costs for smaller institutions.
- Steep learning curve for administrators unfamiliar with the software.
- Limited customization options for unique institutional requirements.

2.1.2 Edulog

Combines school bus routing, GPS fleet tracking, student journeyship management, and parent communication apps into a single platform, aiming to streamline transportation operations.
Cons:

- Complex setup and configuration for multi-campus institutions.
- Frequent updates are causing temporary compatibility issues.
- Limited integration with non-standard hardware or legacy systems.

2.1.3 Loqqat

Provides a smart real-time school bus tracker and management software, enabling route scheduling and live tracking to ensure student safety.
Cons:

- Narrow focus on live tracking, lacking advanced features like fleet maintenance.
- Limited scalability for large institutions with extensive fleets.
- Higher costs for adding additional features beyond basic tracking.

2.2 RFID Technology in Transportation Systems

RFID is widely used in transportation for access control and validation of the user. The usage of an RFID tag within student cards enables easy barcode scanning and validation strengthening the parameter of user credibility. The literature review also focuses on RFID advantages concerning reliability, fast processing, and accuracy in real situations for passenger identification, which would help in determining eligibility of passenger in bus systems.

RFID technology is also used in NTURP to authenticate fee payment and obtain time-based scanning to reduce misapplication of the system. The integration of RFID with time constraints means that the delivery will be partial and will ensure compliance and consequently trust among the stakeholders.

2.3 GPS-Based Bus Tracking

With GPS, social transport has been revolutionized through tracking vehicles in the road networks. In doing so, passengers and bus administrators can track bus positions thus increasing the buses' operational visibility and service delivery. GPS tracking and the provision of estimated arrival time are found to increase the user satisfaction since the device continually and accurately indicates the location.

In NTURP, GPS tracking helps parents and students in planning their travel effectively. When partnered with predictive algorithms, GPS data eliminates the prospects of early or late estimates and keeps drivers on their toes. The same also contains important information about the best route choice and the most suitable time in the context of transportation by bus.

2.4 Load Management in Public Transportation

It is very important to strike a balance between the number of passengers and the carriage capacity in a public transport system because an excess or a shortage of passengers causes problems. Methods like weight sensors and real-time passenger count are found efficient for measuring the bus holding capacity.

In addition to real time load analysis for safety of passengers and bus fleet, NTURP also includes efficient bus operation. Prepare for the load according to

the existing pattern and data and use the pattern to estimate the demand for various products or services.

2.5 Predictive Arrival Systems

The estimates of arrival are derived from the GPS, traffic and past records making them reliable for PSA. Research shows that such systems improve user trust and satisfaction. Use of enhanced features and the integration of more advanced machine learning algorithms can of course enhance the accuracy of the predictions.

In NTURP, arrival features that predict help to cut down waiting time and thus the result is convenient to both students and parents. These features, implemented in simple presentations with user-friendly interfaces, are helpful for a user.

2.6 Mapping APIs: Google Maps and Map box

Mapping services are integral to transportation management systems, providing visualization and geolocation functionalities.

2.6.1 Google Maps API

Google Maps API is a full service solution for map integration which allows for real time traffic data, route and time estimations. The versatile Log parser and its reliability is a clear reason why developers choose to work with the company. To work in NTURP, Google Maps API is quite helpful in providing real time location of buses and their estimated time of arrival, thus making the general user interface more effective.

2.6.2 Map box

Map box is an interactive map tool which operating system can be adapted according to the preferences of the developers. It provides dynamic app theme support, working offline maps, and improved integration options. In certain niches of mapping specifically designed solutions, Map box is flexible and fast. For NTURP, Map box gives an opportunity to design compelling, user-friendly front ends for users.

It means that the decision to choose, for example, Google Maps API instead of Map box, will be based on the specifications, the cost, and the expectations

from users. Both solutions are fundamental in the establishment of safe transport networks.

CHAPTER 3

3 System Requirements

In this bankruptcy, all of the useful requirements of the NTURP and the overall requirement of the stockholders are documented as it's an important a part of a mission or product that allows to satisfy stakeholder's necessities. Now, we can speak system necessities, practical necessities, software program development, and present and selected methodology with the purpose of technique. These sections describe software program methodologies which are present and decided on for this assignment with the glide of machine and alertness detail depicted.

3.1 Functional Requirements

3.1.1 User Authentication and Authorization:

There should be user roles supported in the system (Super Admin, Admin, Driver, Conductor, Student) and only allow access after a proper authentication. It must have a module for creating the user base, the role they should have, and the permission of the role to open some of the modules that a user should open.

3.1.2 Bus and Route Management

The Admin should also be able to handle bus detail and want to set territories or routes for buses. The Driver should have the facility to record/update the driving routes for individual buses. Also there exists the scenario where the Admin needs to view a list of available buses and their corresponding routes.

3.1.3 Ride Management

The Driver must be able to initiate or end a ride, with the system tracking the live location of buses during active rides. The Driver is responsible for authenticating student cards when they board the bus, and the system must record ride data, including the bus number, route, boarded students, and the driver. Both Admin and Students must have access to the bus's live location, while Admin also be able to view the complete ride history for all buses.

3.1.4 Bus Card Management

The admin must be able to assign bus cards to students, as well as revoke or enable student bus cards as needed. The system must also verify student bus cards during boarding to ensure proper access.

3.1.5 Session and Student Management

The system must allow the Admin to create and end user sessions, as well as set their expiry dates. It should automatically disable student cards when a session expires or is deleted. Additionally, the system must generate app credentials for students upon their addition to the system.

3.1.6 Bus Staff Management

Admin must be able to add and manage Drivers and Conductors within the system. Upon registration, the system must generate app credentials for these staff members to enable secure access and management of their duties.

3.1.7 Complaint Management

Students and Drivers must have the ability to submit complaints through their apps. Admin should have a module to view, address, and resolve these complaints, and the system must maintain a record of all complaints along with their current statuses.

3.1.8 Announcement Management

Admin must be able to create and manage announcements within the system. Announcements must be delivered as notifications to Drivers and Students through their apps to ensure timely updates.

3.1.9 Notification and Alerts

The system must send notifications to Students and Drivers regarding announcements, route updates, and other relevant information. Additionally, the system must alert Admin if a bus deviates from its assigned route or leaves its designated area.

3.1.10 Live Location Tracking

The system must track and display the real-time location of buses during active rides. Both Admin and Students should be able to access this live location data via their apps, and the system must store location data for ride history and analysis purposes.

3.2 Non-Functional Requirements

3.2.1 Security:

Apply strict checks for the users' authorization and authenticity.

3.2.2 Performance

The system should be able to answer user actions on the application quickly.

3.2.3 Availability:

The system should be online all the time.

3.2.4 Scalability:

Also, the system must be capable providing its services to a rising number of users and data.

3.2.5 Usability:

Web based system should be easy to use and navigable with an aim of reaching the users of the product.

3.2.6 Maintainability:

It should be easy for the system to be maintained and upgraded from time without lots of time being consumed.

3.2.7 Efficiency

While functioning, the system should effectively manage the amount and kinds of resources used and reduce response time.

3.3 Use Case Diagram

In respect to showing graphic representations of actor communication with the components of the systems the best method therefore is to identify and draw Use Case diagrams that illustrate which actor is capable of performing or accessing what function or component of the systems under consideration.

3.3.1 *Use Case of Sign In*

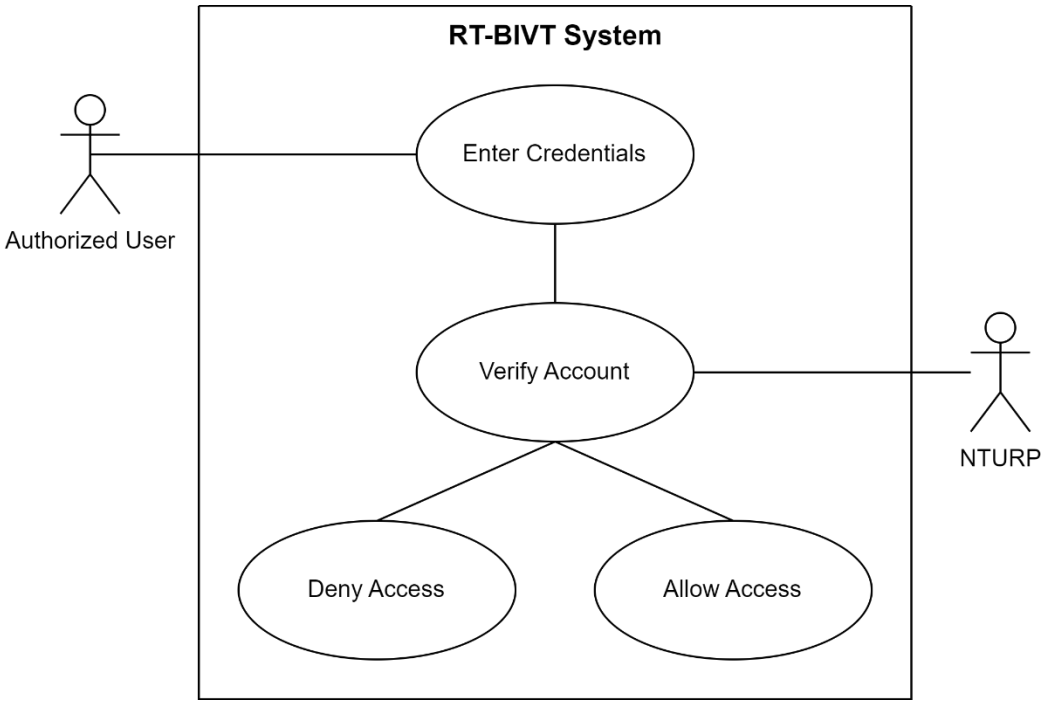


Figure 3.3.1 Use Case of Sign In

3.3.2 *Use Case of Sign Up*

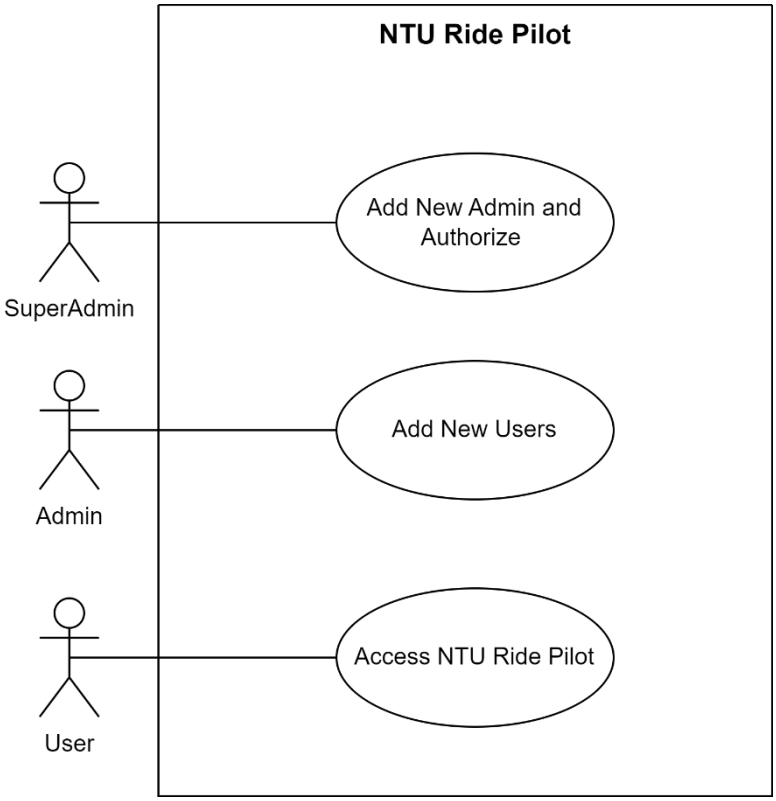


Figure 3.3.2 Use Case of Sign Up

3.3.3 Use Case of Bus and Route Management

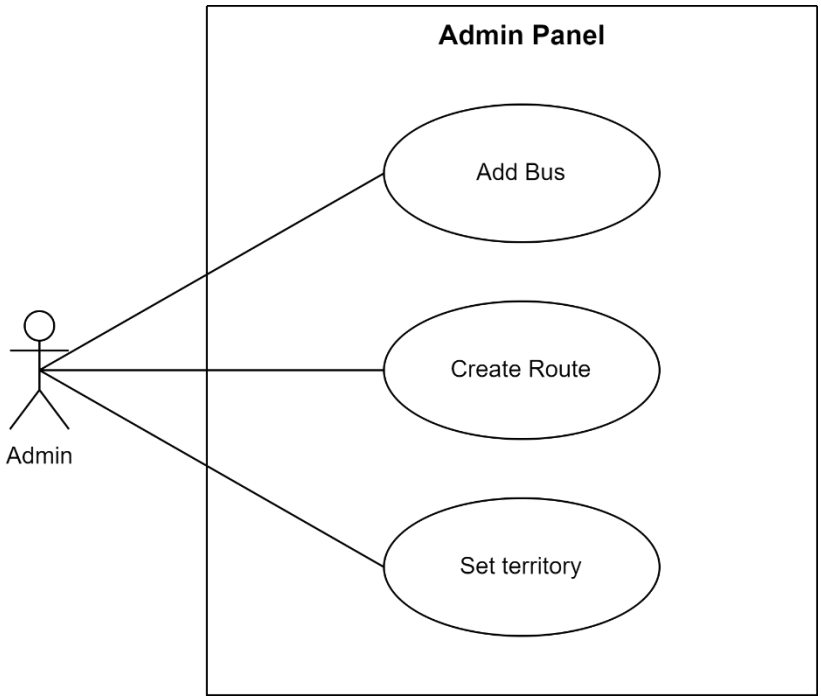


Figure 3.3.3 Use Case of Bus and Route Management

3.3.4 Use Case of Ride Management

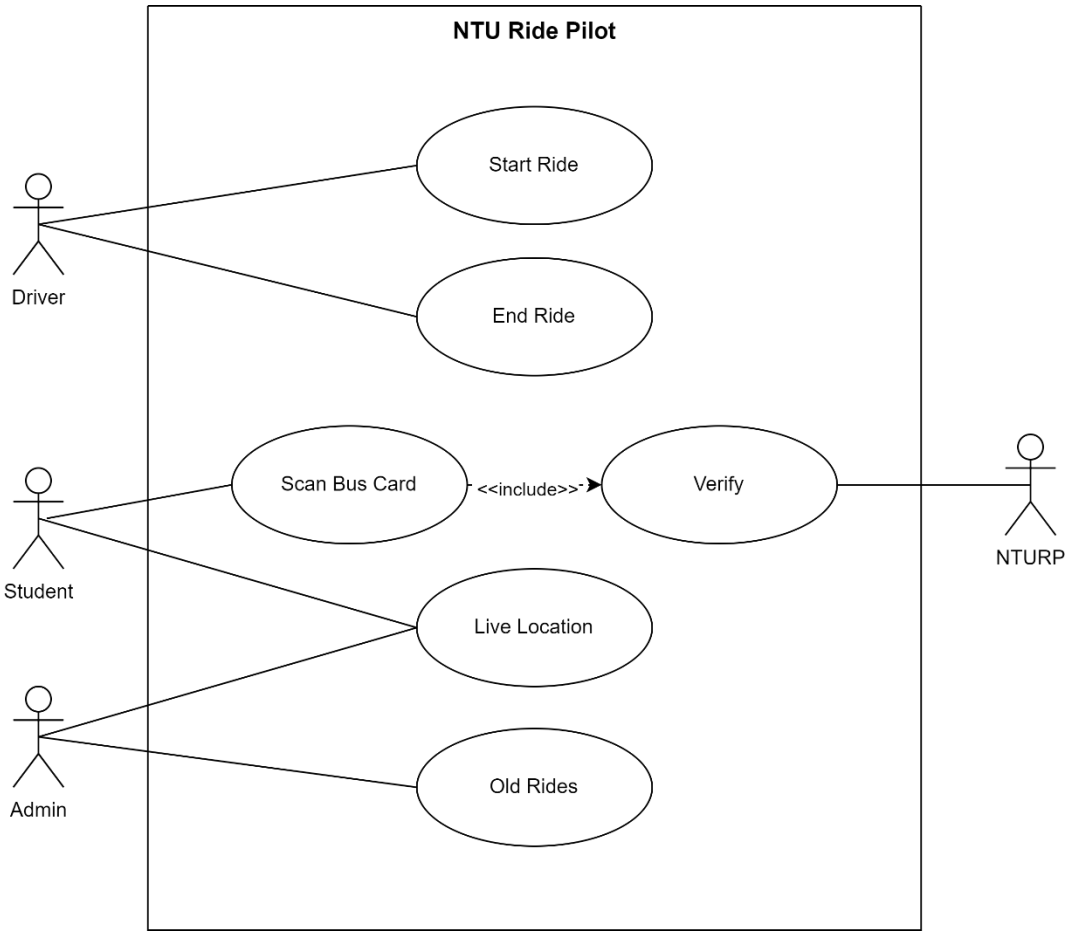


Figure 3.3.4 Use Case of Ride Management

3.3.5 Use Case of Bus card Management

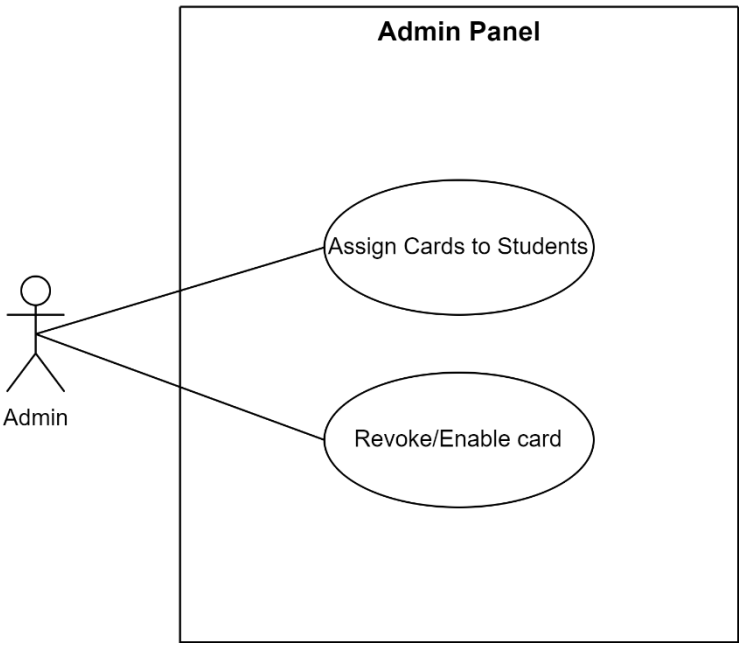


Figure 3.3.5 Use Case of Bus Card Management

3.3.6 Use Case of Student & Session Management

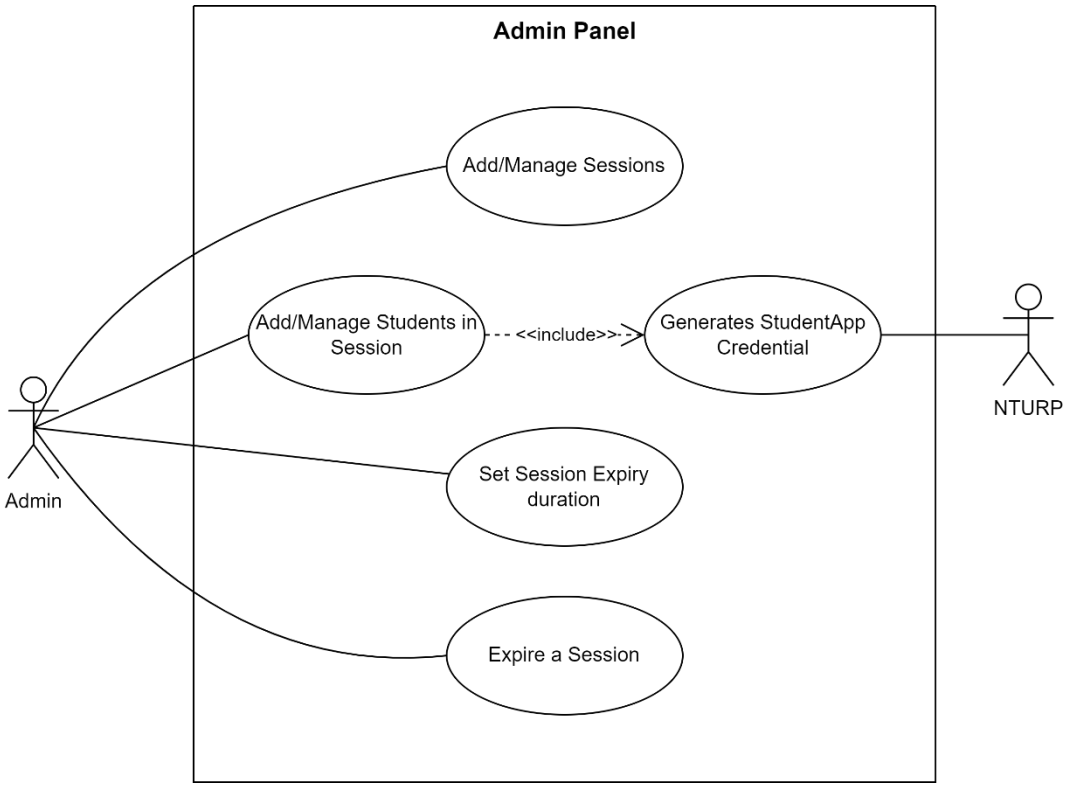


Figure 3.3.6 Use Case of Student & Session Management

3.3.7 Use Case of Bus Staff Management

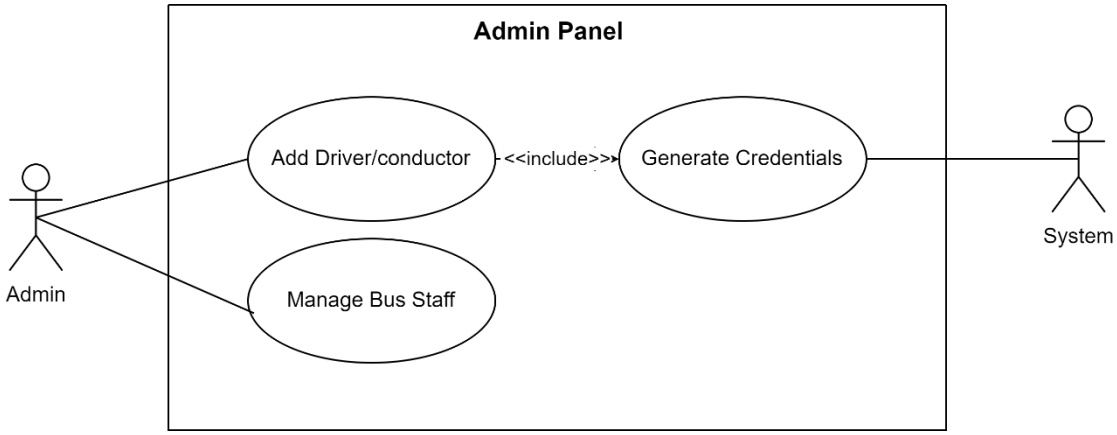


Figure 3.3.7 Use Case of Bus Staff Management

3.3.8 Use Case of Complaint Management

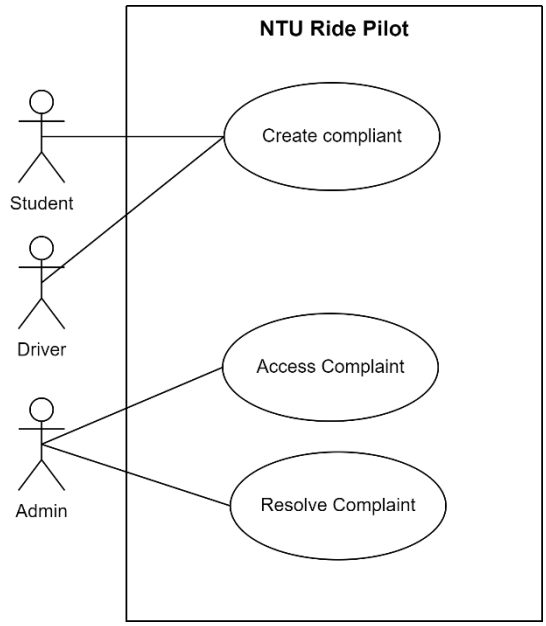


Figure 3.3.8 Use Case of Complaint Management

3.3.9 Use Case of Announcement Management

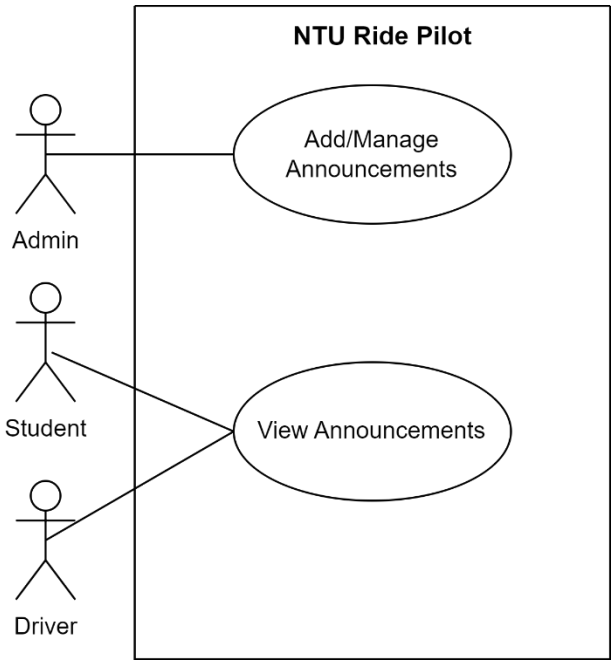


Figure 3.3.9 Use Case of Announcement Management

3.3.10 Use Case of General Functionalities

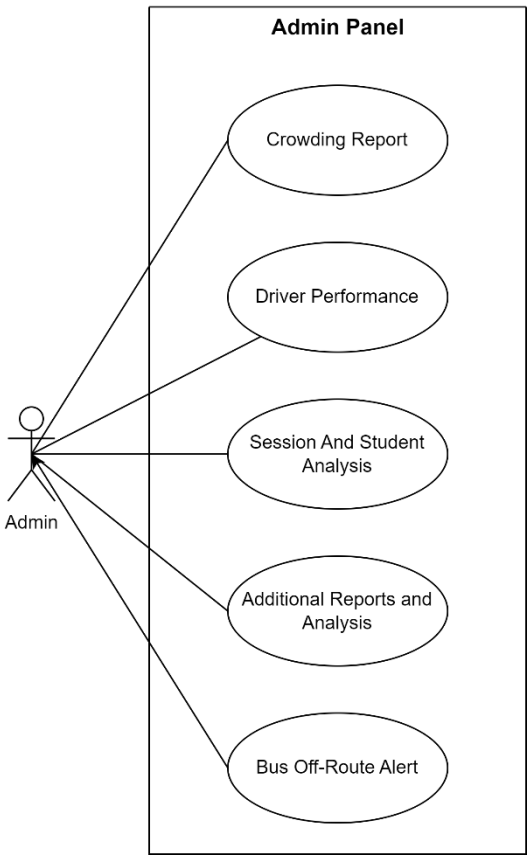


Figure 3.3.10 Use Case of General Functionalities

3.3.11 Use Case of Complete System

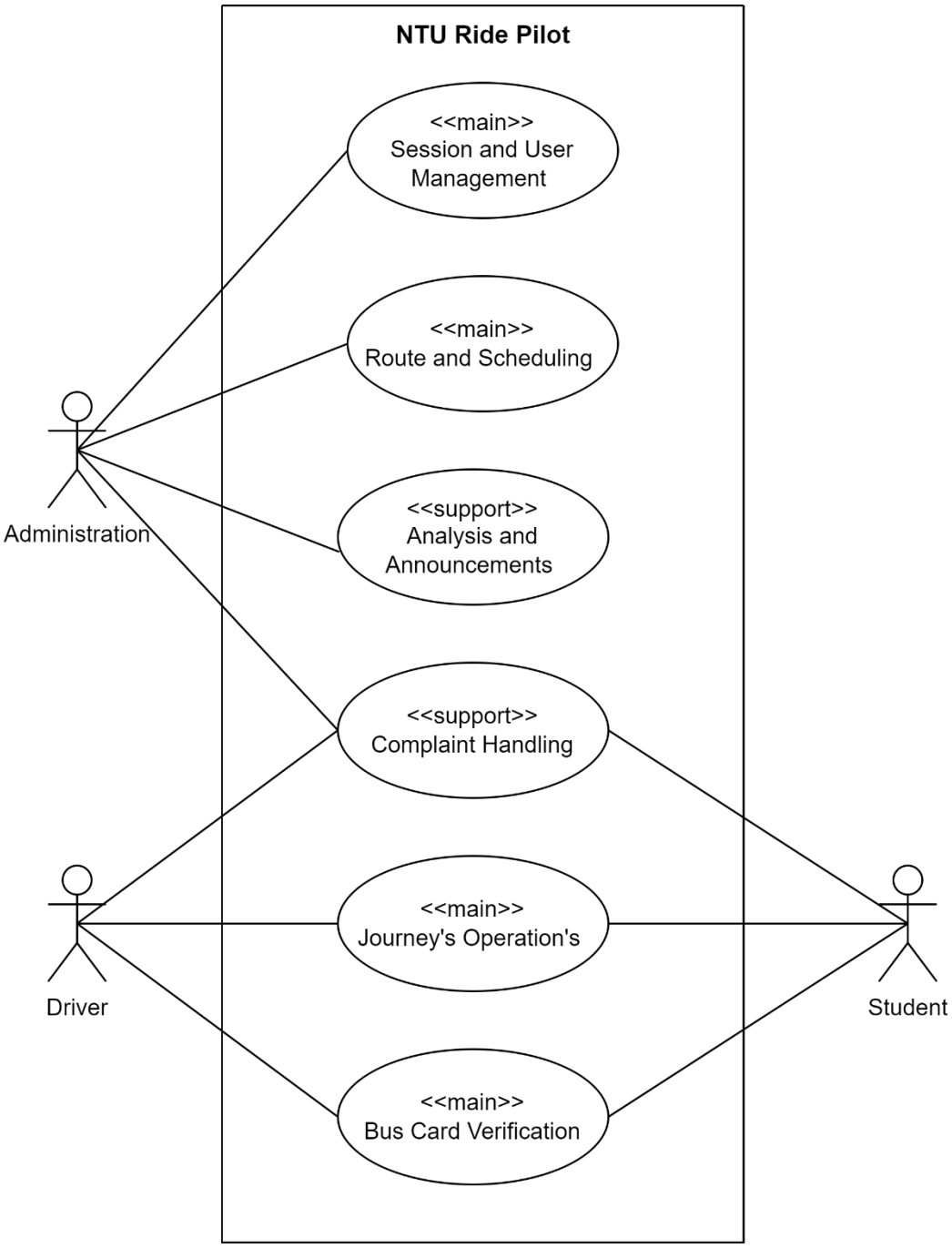


Figure 3.3.11 Use Case of Complete System

3.4 Use Case Description.

Use case description contains every piece of information (use case id, use case name, description, pre- and post-conditions) of each use case.

3.4.1 Description of Sign In:

Table: 3.4.1 Description of Sign In

Field	Details
Use Case Id	01
Use Case Name	Sign In
Actors	Authorized User
Description	Authorized users log in to the NTURP via provided applications. Access is denied if the user does not have an account or proper credentials.
Pre-condition	User must have an account and credentials.
Post-condition	User logs into the NTURP system or is denied access.

3.4.2 Description of Sign Up:

Table: 3.4.2 Description of Sign Up

Field	Details
Use Case Id	02
Use Case Name	Sign Up
Actors	Super Admin, Authorized User
Description	A Super Admin has the ability to create admins and assign specific roles and permissions to each. These admins are tasked with managing the creation of user accounts, including those for bus staff and students.
Pre-condition	User must have an account, enough permissions and credentials to access system.
Post-condition	A new user is created with specific roles and permissions to operate with in NTURP.

3.4.3 Description of Bus and Route Management:

Table: 3.4.3 Description of Bus and Route Management

Field	Details
Use Case Id	03
Use Case Name	Bus and Route Management
Actors	Admin
Description	The admin is responsible for adding buses, defining routes, and assigning territories.
Pre-condition	Admin must be authenticated and have enough permissions to manage buses and routes.
Post-condition	The admin effectively oversees the management of buses, routes, and territories.

3.4.4 Description of Ride Management:

Table: 3.4.4 Description of Ride Management

Field	Details
Use Case Id	04
Use Case Name	Ride Management
Actors	Admin, Driver, Student
Description	The driver starts and ends rides while verifying student bus cards during the trip. The system logs ride-related data and tracks the bus's live location, enabling administration, students, and parents to access ride details and monitor the bus's real-time location.
Pre-condition	The driver must be authenticated and have specified the route and the bus they are assigned to. The student must possess a valid and active bus card.
Post-condition	The driver successfully operates the Rides, while the system tracks and updates the ride details and live bus location. This allows both administrators and students/parents to access the ride

	information and the live location of the bus.

3.4.5 Description of Bus Card Management:

Table: 3.4.5 Description of Bus Card Management

Field	Details
Use Case Id	05
Use Case Name	Bus Card Management
Actors	Admin
Description	The admin is responsible for assigning and managing student bus cards.
Pre-condition	The admin must be authenticated and have necessary permissions to manage bus cards. Students are required to be registered in the system.
Post-condition	The admin can successfully assign bus cards to students, revoke the cards to prevent further use, and re-enable them when needed.

3.4.6 Description of Session and Student Management:

Table: 3.4.6 Description of Session and Student Management

Field	Details
Use Case Id	06
Use Case Name	Session and Student Management
Actors	Admin, System
Description	The admin can create and terminate sessions, set session expiry, and manage students within the system. When a student is added, the system automatically generates app credentials for them. Expiring or deleting a session will deactivate all bus cards associated with that session.
Pre-condition	The admin must be authenticated and have necessary permissions to manage

	bus cards. Student data must be accurate and complete for the generation of credentials.
Post-condition	The admin effectively oversees both sessions and student management.

3.4.7 Description of Bus Staff Management:

Table: 3.4.7 Description of Bus Staff Management

Field	Details
Use Case Id	07
Use Case Name	Bus Staff Management
Actors	Admin, System
Description	The admin oversees drivers and conductors. When a new staff member is added, the system automatically generates app credentials for them.
Pre-condition	The admin must be authenticated and have necessary permissions to manage bus staff.
Post-condition	The admin effectively manages bus staff.

3.4.8 Description of Complaint Management:

Field	Details
Use Case Id	08
Use Case Name	Complaint Management
Actors	Student, Driver, Admin

Description	Students and drivers are allowed to submit complaints about transportation services and related concerns.
Pre-condition	All users are required to be authenticated. Administrators must also be authenticated and possess the necessary permissions to manage complaints.
Post-condition	Complaints submitted by students and drivers get reviewed and resolved by the authorities.

Table: 3.4.8 Description of Complaint Management

3.4.9 Description of Announcement Management:

Table: 3.4.9 Description of Announcement Management

Field	Details
Use Case Id	09
Use Case Name	Announcement Management
Actors	Admin, Driver, Student
Description	The admin shares announcements related to transportation.
Pre-condition	All users are required to be authenticated.
Post-condition	The administrator successfully posts announcements.

3.4.10 Description of General Functionalities:

Table: 3.4.10 Description of General Functionalities

Field	Details
Use Case Id	10
Use Case Name	General Functionalities
Actors	Users
Description	The admin can generate reports and analyse valuable information extracted from the data currently stored.

Pre-condition	The admin must be authenticated and have necessary permissions.
Post-condition	Administration makes better decisions and enhances services through the analysis of current transportation data.

Chapter 4

4.1 Agile Software Development Methodology

Agile methodology is an iterative and incremental approach to software development that emphasizes flexibility, collaboration, and customer feedback. Unlike traditional methodologies like the Waterfall model, Agile allows for adaptive planning, evolutionary development, and continual improvement, enabling rapid and flexible responses to change. It is particularly effective in managing the complexity and unpredictability of software projects.

4.2 Selected Methodology: Agile

A software development methodology is a way to improve development work with the help of dividing the development process into distinct phases to make a system with better productivity. It also helps to structure and control the whole system. It involves different methodologies, also called the Software Development Life

Cycle, that are stages for software development with a certain set of rules. Generically, we categorized the methodologies into Rapid application development and planned-driven. Waterfall, spiral is planned driven while agile is Rad based.

4.3 Reasons for Selecting Agile Methodology

1. **Flexibility and Adaptability**: Agile allows the project to adapt to changes in requirements and technology swiftly.
2. **Customer Collaboration**: Regular feedback from users ensures that the development aligns with the user's needs and expectations.
3. **Incremental Delivery**: Agile facilitates the delivery of small, workable segments of the project, ensuring a faster time-to-market and continuous improvement.
4. **Risk Management**: Regular reviews and iterations help in early identification and resolution of issues, reducing the overall risk.

4.4 Agile Project Planning and Execution for NTURP

Agile project planning and execution involve the division of the project into sprints, with each sprint aimed at delivering a potentially shippable product increment. The key phases include:

1. **Product Backlog Creation**: Gather and prioritize necessities for the NTURP undertaking, growing a product backlog.
2. **Sprint Planning**: At the start of every dash, pick a hard and fast of capabilities from the product backlog and plan their delivery.
3. **Daily Stand-ups**: Conduct day by day meetings to speak about development, demanding situations, and plan the day's paintings.
4. **Sprint Execution**: Develop, take a look at, and combine features inside the dash.
5. **Sprint Review**: At the quilt of every sprint, reveal the finished paintings to stakeholders and collect comments.
6. **Sprint Retrospective**: Reflect at the sprint to identify successes and regions for development.
7. **Release Planning**: Plan releases based totally at the undertaking progress, stakeholder comments, and marketplace situations.

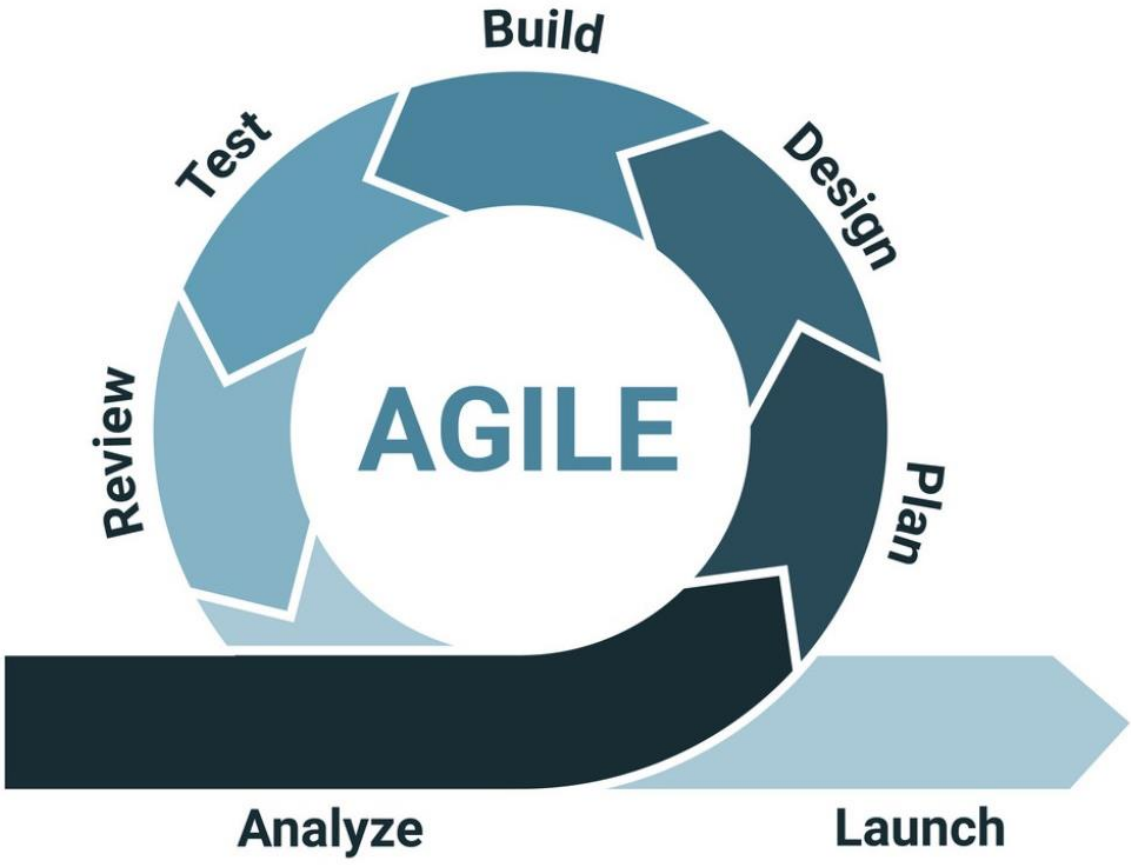


Figure 4.1 Agile Model

Chapter 5

5 System Architecture

5.1 Database Design

The database for this NTURP is implemented using Google Fire store, a NoSQL cloud database that provides real-time data synchronization and scalability. Fire store's document-based structure is well-suited for handling dynamic and hierarchical data while maintaining efficiency in read and write operations. The schema is designed to ensure optimal performance, scalability, and maintainability.

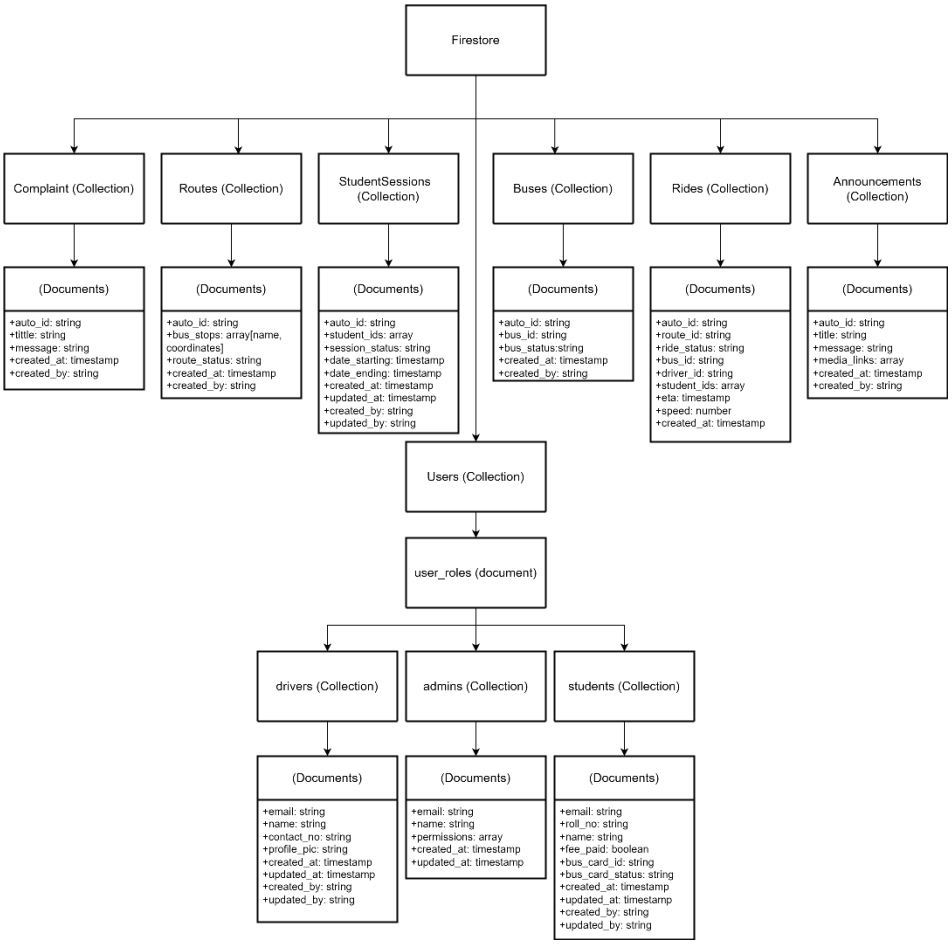


Figure 5.1.1 Database Design

5.2 Sequence Diagram

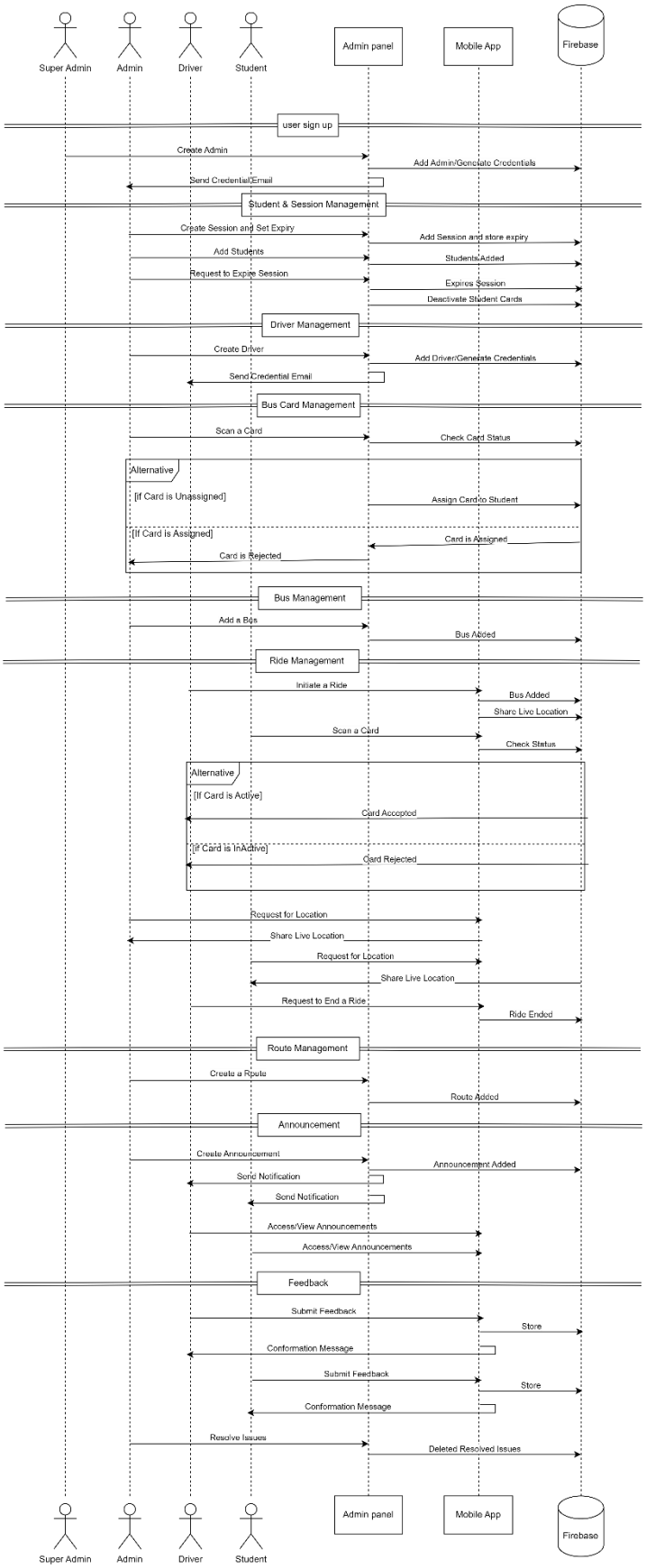


Figure 5.2.1 Sequence of whole system

Chapter 6

6 AI-Model and Algorithms

6.1 General Disease Detection:

We integrate the disease detection application that is primarily based at the system learning algorithm that is decision tree. We integrate this matching software for the assist of docs. Doctor will ask the signs and symptoms from patients and supply to that application a good way to are expecting the sickness from the given dataset.

A Decision Tree is a famous system gaining knowledge of set of rules used for classification and regression duties. Here is an in depth clarification of the way a selection tree works, including the important thing standards, steps, and attributes concerned in constructing and the use of a choice tree.

6.1.1 Key Concepts of Decision Tree:

1. **Node**: Each point of decision in the tree.
    - **Root Node**: The topmost node, representing the entire dataset.
    - **Internal Node**: Nodes that represent decisions based on attributes.
    - **Leaf Node**: Terminal nodes that represent the output of the decision.
2. **Edge**: Connects nodes, representing the outcome of a decision.
3. **Attribute/Feature**: A characteristic of the data used to split the nodes.
4. **Gini Index / Entropy**: Measures used to determine the quality of a split. Lower values indicate a better split.
    - **Gini Index**: Measures the impurity of a dataset. Used in CART (Classification and Regression Trees).
    - **Entropy**: Used in ID3 and C4.5 algorithms to measure the disorder in the dataset.

6.1.2 Steps in Building a Decision Tree

1. **Selecting the Best Attribute**:
    - The process begins at the root node. The algorithm evaluates all possible features and selects the one that best splits the data into subsets with the most homogeneous target variable (class label).
    - This selection is based on the impurity measures such as Gini Index, Entropy, or Information Gain.
2. **Splitting the Dataset**:
    - Once the best attribute is selected, the dataset is split into subsets. Each subset corresponds to a unique value or range of values of the selected attribute.

3. **Creating Sub-nodes**:
    - Each subset created from the split becomes a new node in the tree. This node will be further split using the same process recursively.
4. **Stopping Criteria**:
    - The splitting process stops when one of the following criteria is met:
        - All the data in the node belongs to a single class.
        - There are no remaining attributes to split the data.
        - A predefined stopping condition such as maximum depth of the tree or minimum number of samples per node is reached.
5. **Pruning**:
    - Pruning is done to improve the tree's generalization ability by reducing its size. It removes branches that have little importance and can lead to overfitting.
    - **Pre-pruning**: Stops the tree growth early.
    - **Post-pruning**: Removes branches from a fully grown tree.

6.1.3 Training Epochs

During an epoch, the learning algorithms updates the model’s parameters based on data. It took approximately 80 epochs to train.

6.1.4 Architecture

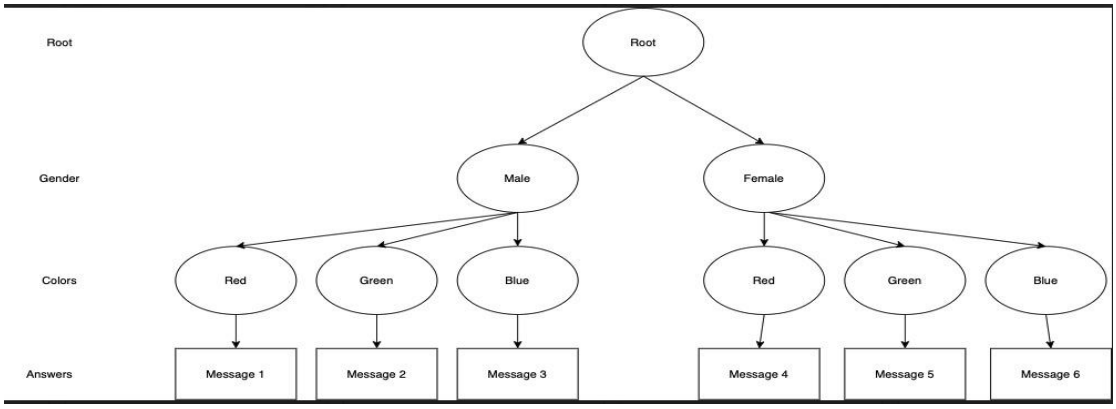


Figure 6.1.4.1 Architecture Diagram of Decision Tree

6.1.5 Confusion Matrix

Figure 6.1.5.1 Confusion Matrix of G.D.D

	Predicted Positive	Predicted Negative
Actual Positive	700	40
Actual Negative	60	635

Accuracy = (700 + 635) / (700 + 635 + 60 + 40)

Accuracy = 0.89

Accuracy = 0.89 x 100

Accuracy = 89 %

6.1.6 Accuracy

Model accuracy is 89 %.

6.1.7 Activity diagram of program

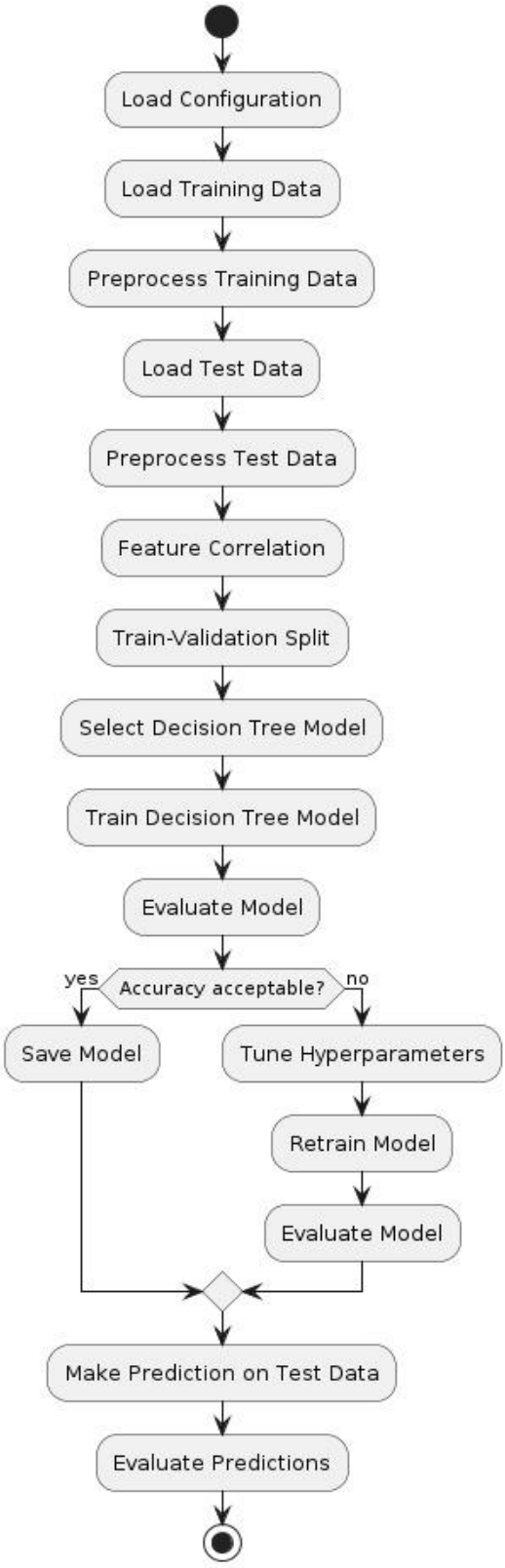


Figure 6.1.1 Activity diagram of program

6.2 Fractured Bone Detection:

We have additionally integrate AI model for detection of fractured bones of wrists. This is done with the help of YOLO set of rules this is used for the detection of objects. Doctor will give the X-Ray image as enter to the model then model will predict, it's far fractured or now not as output.

6.2.1 Dataset

The GRAZPEDWRI-DX is an open dataset containing 20327 annotated pediatric trauma wrist radiograph images of 6091 patients, treated at the Department for Pediatric Surgery of the University Hospital Graz between 2008 and 2018. Several pediatric radiologists annotated the images by placing bounding boxes to mark 9 different classes:

- boneanomaly (276 boxes),
- bonelesion (45 boxes),
- foreignbody (8 boxes),
- fracture (18090 boxes),
- metal (818 boxes),
- periostealreaction (3453 boxes),
- pronatorsign (567 boxes),
- softtissue (464 boxes),
- text (23722 boxes).

6.2.2 YOLO Working and Explanation

YOLO (You Only Look Once) is an object detection algorithm that has gained significant popularity in the field of computer vision. It is both an algorithm and a model architecture designed for real-time object detection tasks. YOLO stands out for its speed and accuracy in detecting objects in images and videos.

1. **Algorithm/Model**: YOLO is both an algorithm and a model architecture. The YOLO algorithm applies a deep convolutional neural network (CNN) model to perform object detection tasks.
2. **Real-Time Object Detection**: YOLO is designed for actual-time item detection, that means it could system pictures or video frames swiftly and hit upon gadgets within them with excessive accuracy.
3. **Single Pass**: Unlike conventional object detection techniques that apply a sliding window approach or vicinity notion networks, YOLO takes a single

bypass thru the enter photograph or frame and without delay predicts bounding packing containers and class probabilities for all items detected.

YOLO Working

1. **Grid-based Detection**:
    - YOLO divides the input image into a grid of cells. Each cellular is chargeable for predicting bounding boxes and sophistication possibilities for objects that fall inside it.
2. **Bounding Box Prediction**:
    - Each grid cell predicts multiple bounding containers (with predefined sizes and styles) along with self belief scores that imply the chance of each box containing an object.
3. **Class Prediction**:
    - Each bounding field predicts elegance chances for the items it incorporates. YOLO makes use of softmax activation to are expecting the opportunity distribution across more than one training.
4. **Non-Maximum Suppression (NMS)**:
    - YOLO applies non-maximum suppression to remove redundant or overlapping bounding boxes with decrease self belief rankings, retaining only the maximum assured detections.

6.2.3 Training Epochs

During an epoch, the learning algorithms updates the model's parameters based on data. It took approximately 120 epochs to train.

6.2.4 Architecture Diagram:

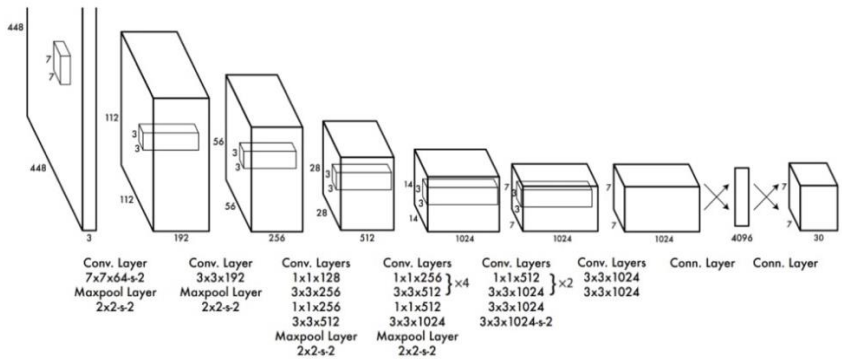


Figure 6.2.1 Architecture Diagram

6.2.5 Confusion Matrix:

Figure 6.2.5.1 Confusion Matrix of Yolo Model

	Predicted Positive	Predicted Negative
Actual Positive	2800	200
Actual Negative	300	2060

Accuracy = (2800 + 2060) / (2800 + 2060 + 300 + 200)

Accuracy = 0.81

Accuracy = 0.81 x 100

Accuracy = 81 %

6.2.1 Accuracy

Model Accuracy is 81%.

6.2.2 Activity Diagrams

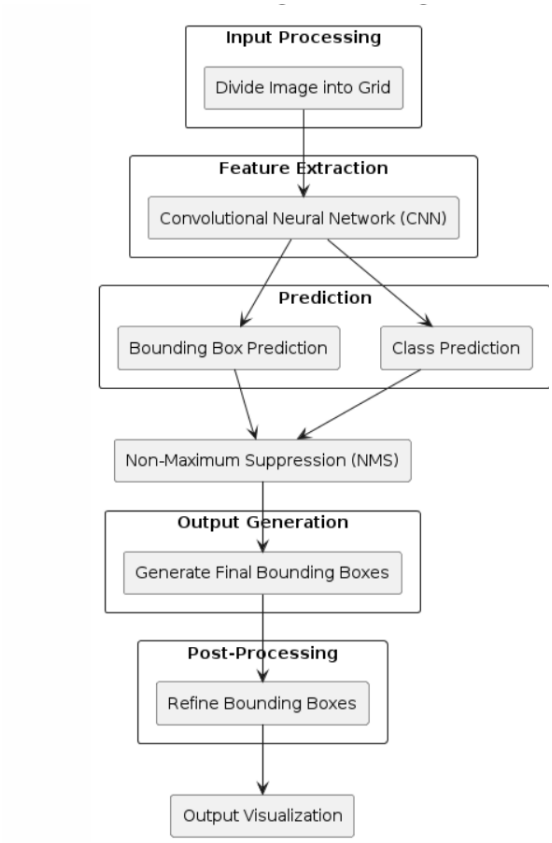


Figure 6.2.2 Activity diagram of working of YOLO

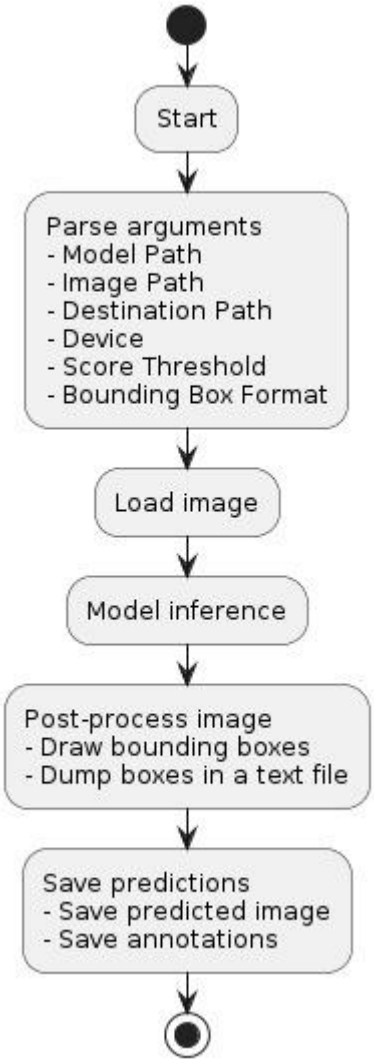


Figure 6.2.3 Activity diagram of program

Chapter 7: System Implementation

After successfully completing the initial phases of development, including requirement analysis, design, and planning, we now proceed to the final stage: the actual implementation of the hospital management system. This chapter provides a comprehensive overview of the tools, technologies, and processes involved in the development of the system. We will detail each module and component, as well as the integration of an AI model for disease detection.

7.1 System Tools and Technology

The tools and technologies used in this project are as follows:

- **Django**: A high-level Python web framework that encourages rapid development and clean, pragmatic design.
- **React**: A JavaScript library for building user interfaces, particularly single-page applications.
- **MySQL**: An open-source relational database management system.
- **VS Code**: A source-code editor made by Microsoft with support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.
- **AI Model (Decision Tree)**: Used for disease detection based on user input.

7.1.1 Django

Django is a excessive-stage Python web framework that lets in fast improvement and clean, pragmatic layout. It takes care of much of the problem of net improvement, so developers can cognizance on writing their programs while not having to reinvent the wheel.

- **Routing**: Django's URL dispatcher allows clean, readable URLs.
- **Templating**: The Django template engine facilitates the separation of presentation and business logic.
- **ORM**: Django's Object-Relational Mapping (ORM) system makes it easy to interact with the database.
- **Security**: Django provides built-in protection against many security threats, including SQL injection, cross-site scripting, and cross-site request forgery.

7.1.2 React

React is a JavaScript library for building user interfaces. It enables developers to build complex UIs from small, isolated pieces of code called "components."

- **Component-Based**: React promotes the development of reusable components.
- **Virtual DOM**: React uses a virtual DOM to optimize rendering and improve performance.
- **JSX**: A syntax extension for JavaScript that looks similar to HTML, making it easier to write and understand components.

7.1.3 MySQL

MySQL is a widely used relational database management system. It is known for its reliability, high performance, and ease of use.

- **Data Storage**: MySQL stores data in tables, which are efficient for storing structured data.
- **Query Language**: MySQL uses Structured Query Language (SQL) for accessing and managing data.
- **Transactions**: MySQL supports transactions, ensuring data integrity and consistency.

7.1.4 VS Code

Visual Studio Code, commonly known as VS Code, is a source-code editor developed by Microsoft. It includes support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

- **Extensions**: VS Code has a rich ecosystem of extensions that enhance its functionality.
- **Terminal**: The terminal allows running commands and scripts directly within the editor.
- **IntelliSense**: Provides smart completions based on variable types, function definitions, and imported modules.

7.2 Module Implementation

7.2.1 Dashboard

The Dashboard module provides an overview of the hospital's operations. It includes various sub-modules for different departments and functionalities.

7.2.1.1 Front-Desk

- Allows searching for patient records.
- Manages patient appointments.
- Displays the history of patient encounters.
- Allows uploading patient-related documents.
- Tracks the number of patient encounters.
- Generates reports on patient registrations.
- Provides reports on ward activities.
- Generates reports on patient encounters.

7.2.1.2 Nursing

- Records patient vital signs.
- Manages injection administration.
- Handles dressing and medical procedures.
- Manages immunization records.
- Manages antenatal clinic activities.
- Handles requisitions for medical supplies.
- Manages nursing inventory.
- Manages patient admissions.
- Generates reports on antenatal care.
- Provides reports on pregnancy cases.

7.2.1.3 Doctors / Department

- Manages outpatient department clinics.
- Handles patient admissions.

- Generates reports on patient encounters.
- Provides reports on patient diagnoses.

7.2.1.4 Laboratory

- Manages pending lab requests.
- Records lab results.
- Manages parameters for test reporting.
- Handles administration of lab tests.
- Manages categories of lab tests.
- Handles requisition of consumables.
- Manages lab inventory.
- Tracks the count of lab tests.
- Records lab test results.

7.2.1.5 Radiology

- Manages pending X-ray requests.
- Manages pending CT scan requests.
- Manages pending ultrasound requests.
- Manages pending ECG requests.
- Records radiology test results.
- Manages parameters for radiology test reporting.
- Handles administration of radiology tests.
- Tracks the count of radiology tests.

7.2.1.6 Dispensary

- Manages outpatient dispensary.
- Manages inpatient dispensary.
- Handles purchase orders.
- Manages requisition for drugs and supplies.
- Manages dispensary inventory.
- Generates reports on dispensary activities.
- Manages the pharmacy store.

- Handles management of drugs.
- Manages consumables inventory.
- Manages store operations.
- Manages vendor information.
- Handles drug purchases.
- Confirms receipt of purchased drugs.
- Generates reports on drug expiry.
- Manages overall inventory.
- Manages consumables inventory.
- Confirms purchases.

7.2.1.7 HMO Authorization

- Manages pending authorizations.
- Generates approval reports.
- Provides summaries of bills.
- Generates payment reports.
- Manages patient bills.
- Manages company-related information.
- Reviews patient bills.
- Generates patient bill reports.
- Manages admissions and discharges.
- Manages hospital services.

7.2.1.8 Cashier

- Manages cashier activities.
- Handles deposits.
- Manages expenses.
- Generates daily cashier reports.
- Provides revenue reports.
- Generates payment reports.

7.2.1.9 Reports

- Generates lab result reports.
- Provides appointment reports.
- Generates diagnosis reports.
- Manages drug inventory reports.
- Manages consumable inventory reports.
- Generates reports on admissions and discharges.
- Tracks patient SMS history.
- Provides payment reports.
- Generates encounter reports.
- Provides dispensary reports.
- Generates revenue reports.
- Manages ward reports.
- Provides antenatal care reports.
- Generates pregnancy reports.
- Tracks birth reports.
- Manages immunization reports.
- Tracks patient encounter count.
- Tracks laboratory test count.
- Tracks radiology test count.
- Provides registration reports.
- Manages expense reports.
- Confirms purchase reports.
- Tracks user login logs.

7.2.1.10 User Management

- Manages vaccine information.
- Handles ward management.
- Manages bed availability.
- Manages user accounts.
- Controls user access permissions.
- Configures patient information.

- Manages patient discharge.
- Handles admissions and discharges.
- Manages admissions.
- Handles database backups.

7.2.1.11 SMS

- Composes SMS messages.
- Tracks SMS history.
- Manages SMS settings.
- Generates SMS reports.

7.2.1.12 Administration

- Manages administrative activities.
- Configures encounter settings.
- Handles database backups.
- Manages antenatal care settings.

7.3 System Integration and Deployment

The health center management system is built the usage of Django for the backend, React for the frontend, and MySQL for the database. The integration of those technology ensures a seamless and efficient workflow.

7.3.1 Frontend-Backend Communication

React additives speak with the Django backend thru RESTful APIs. These APIs cope with diverse requests inclusive of fetching patient data, updating records, and processing bills.

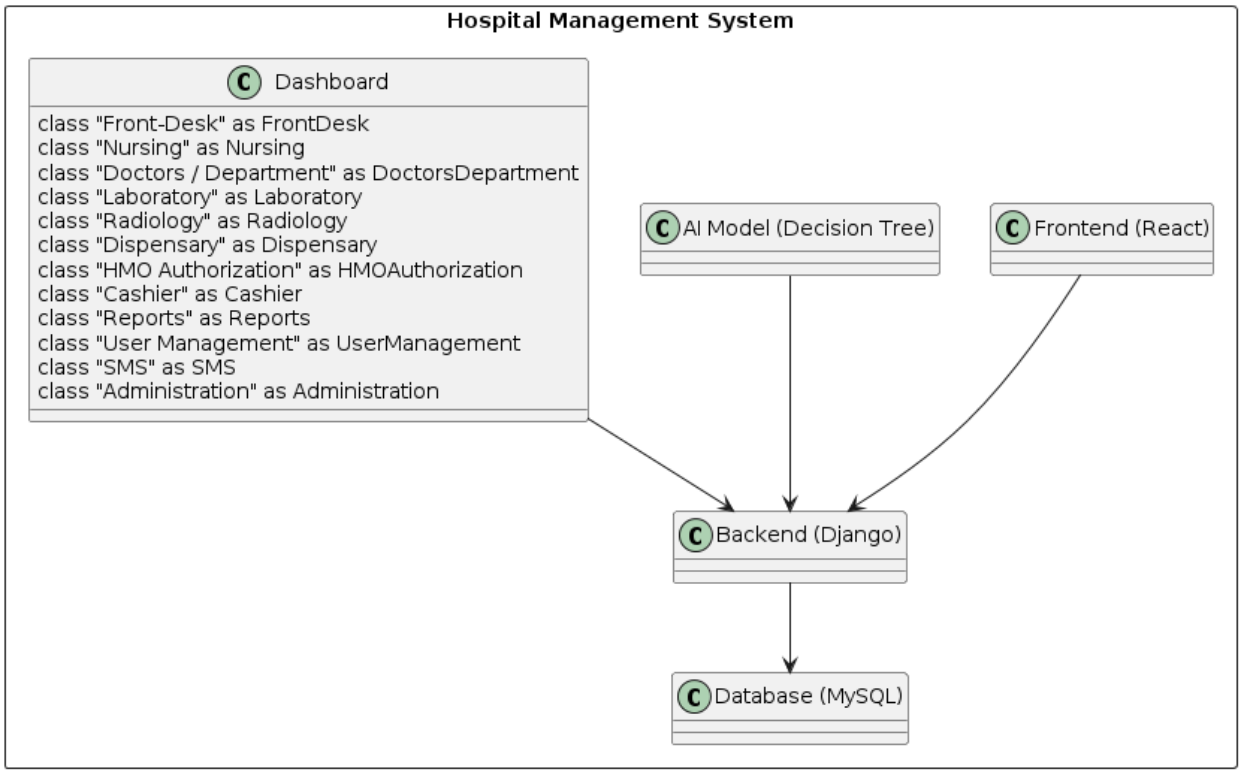


Figure 7.3.1.1 Class Diagram of Hospital Management System

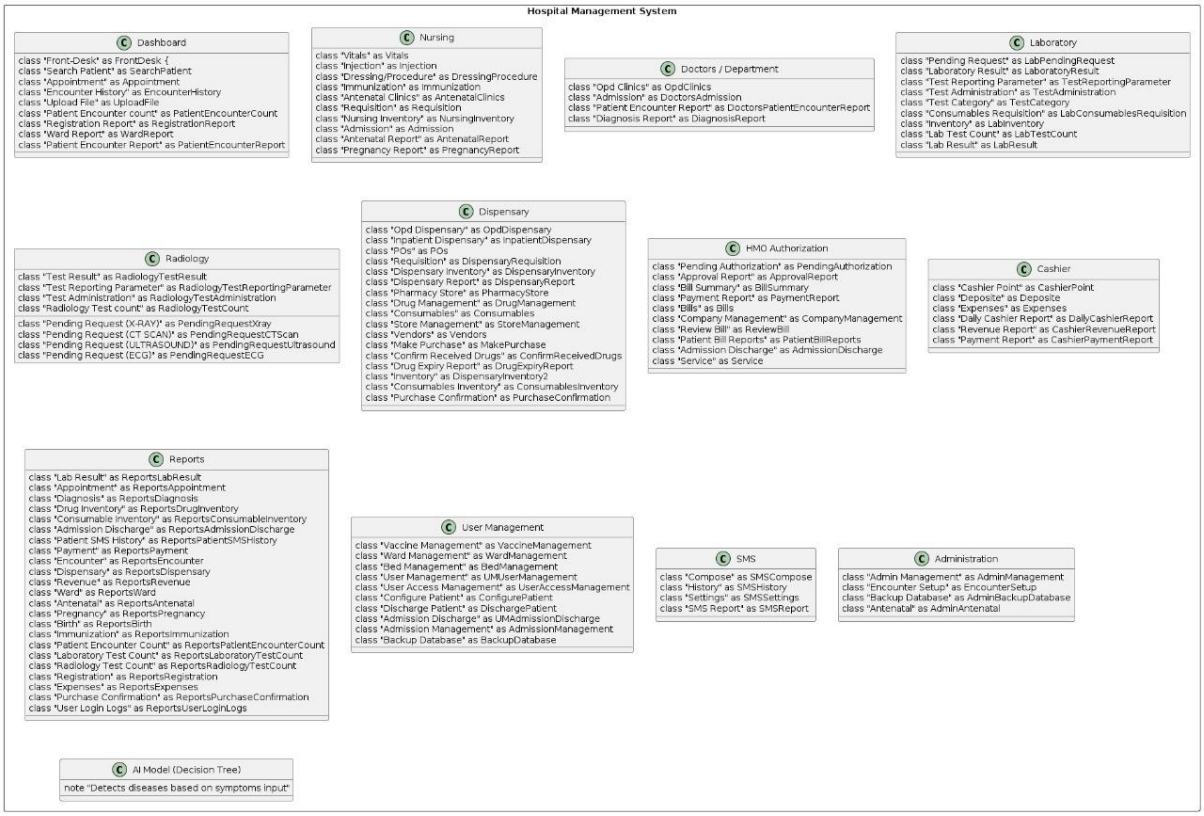


Figure 7.3.1.2 Component Diagram of System

7.3.2 Database Management

MySQL is used to save all the records related to the health facility's operations. Django's ORM makes it smooth to have interaction with the database, ensuring information integrity and consistency.

7.3.3 Deployment

The system is deployed on a cloud platform, making sure excessive availability and scalability. Regular backups are taken to prevent statistics loss, and security features are carried out to shield touchy data.

Conclusion

The implementation of the medical institution management machine entails the integration of numerous tools and technologies to create a complete answer for managing hospital operations. Each module and component is carefully designed and advanced to ensure a seamless and efficient workflow. The integration of an AI model for sickness detection adds a precious function to the device, supplying accurate and timely diagnoses primarily based on affected person input. This chapter provides an in depth assessment of the implementation method, highlighting the important thing components and capabilities of the machine.

Chapter 8

8 Screenshots

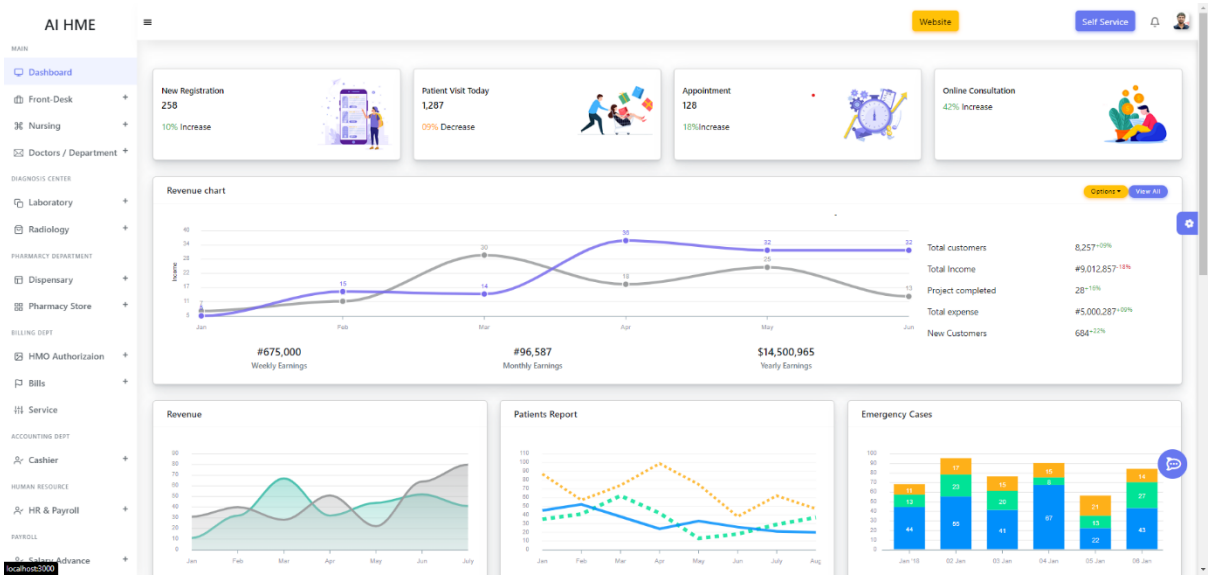


Figure 8.1 Main Dashboard

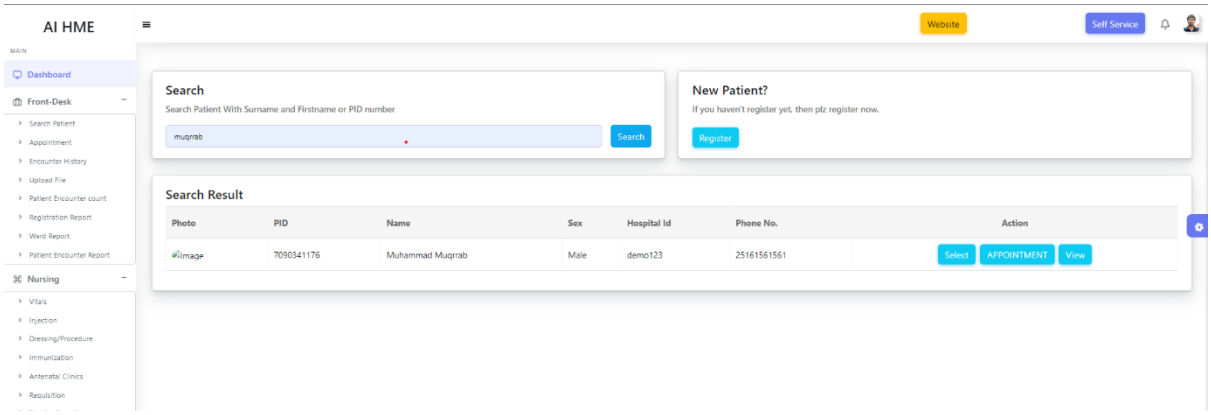


Figure 8.2 Patient Dashboard

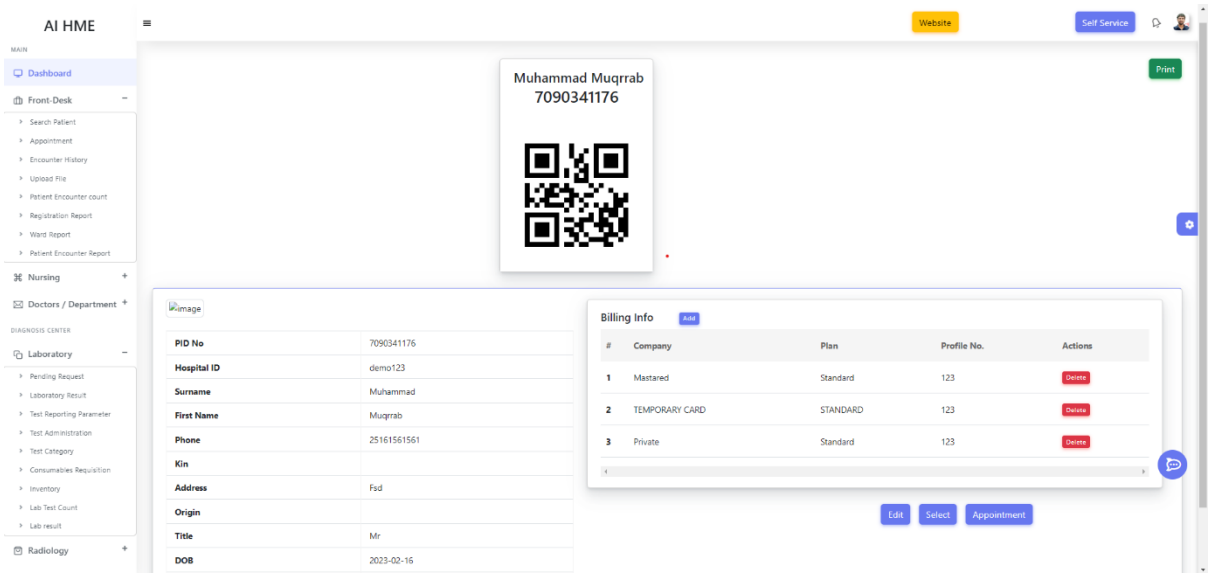


Figure 8.3 Patient Detail

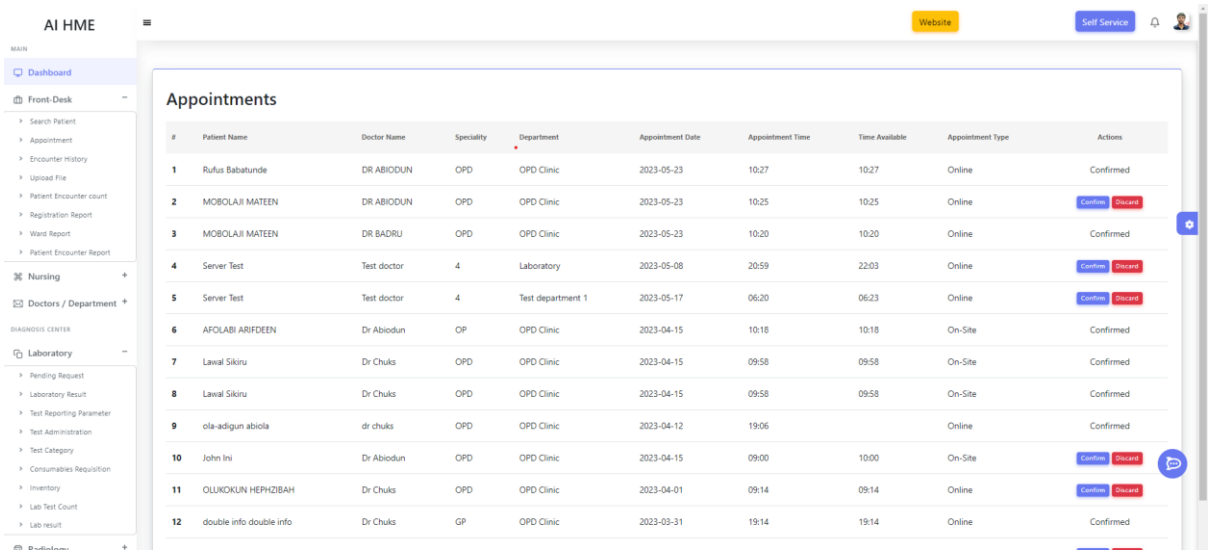


Figure 8.4 Patient Appointment

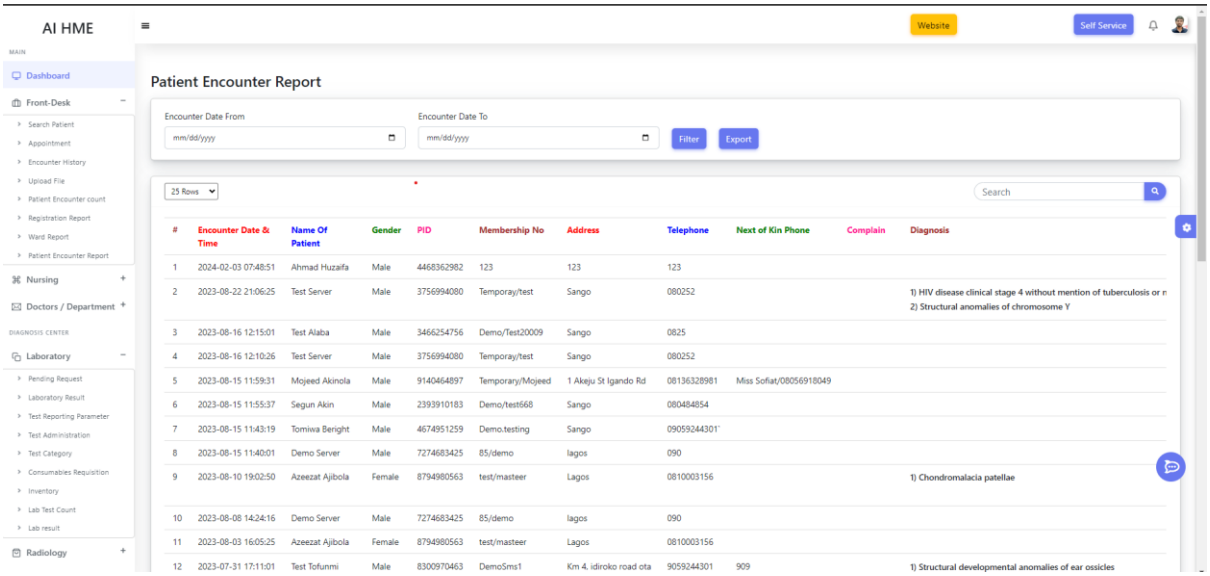


Figure 8.5 Patient Report

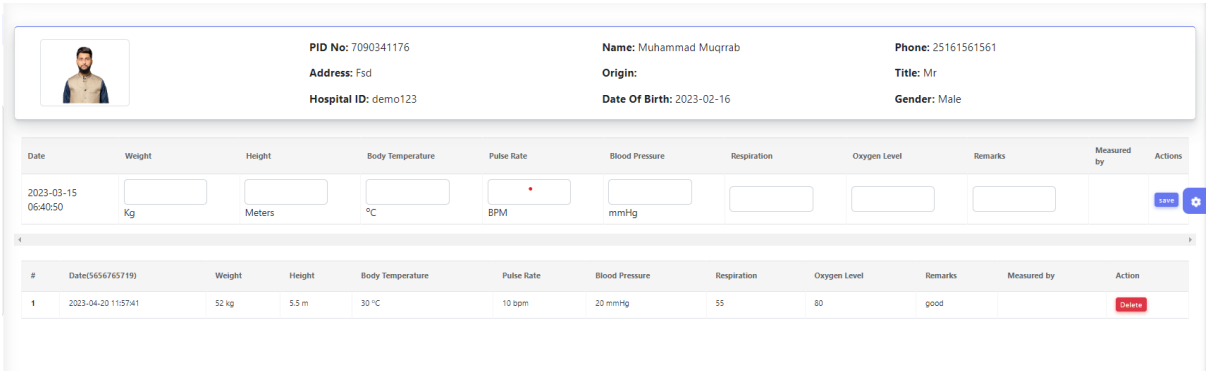


Figure 8.6 Nurse Vital

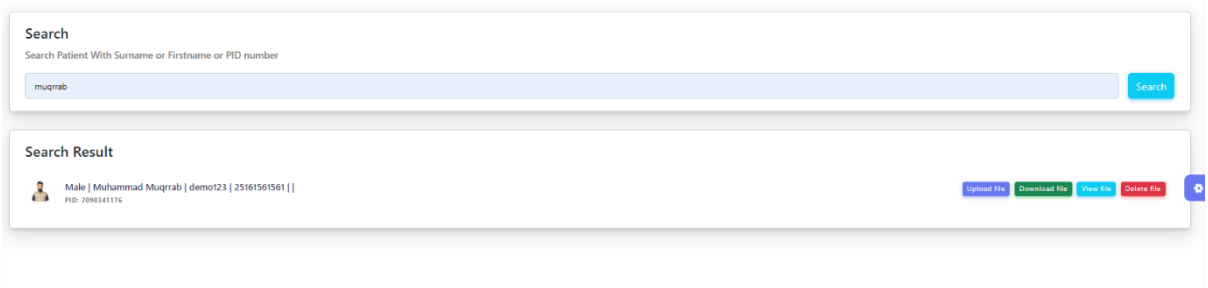
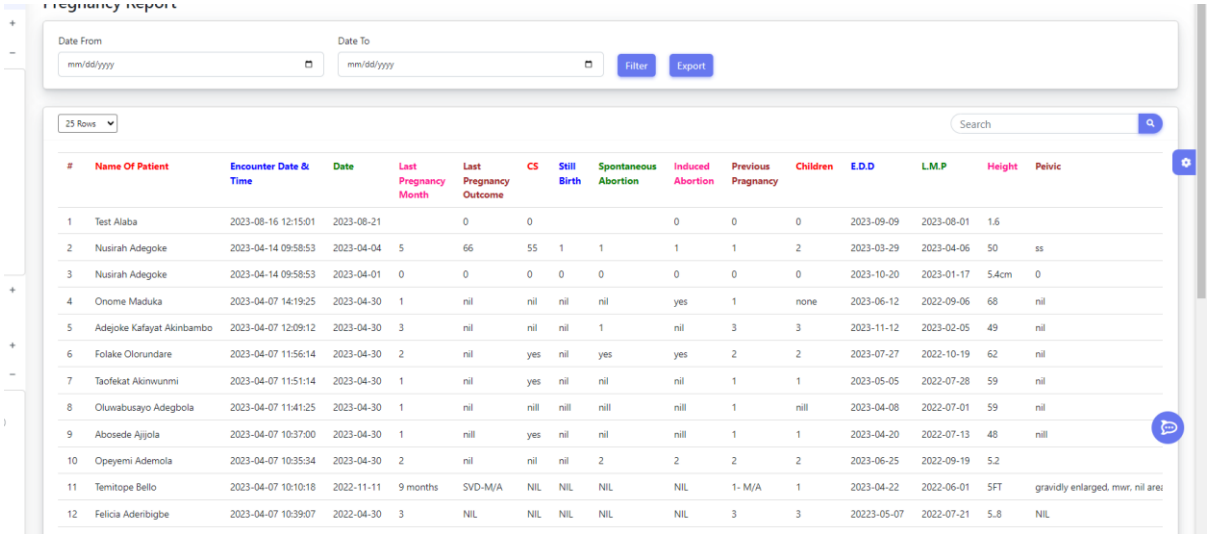
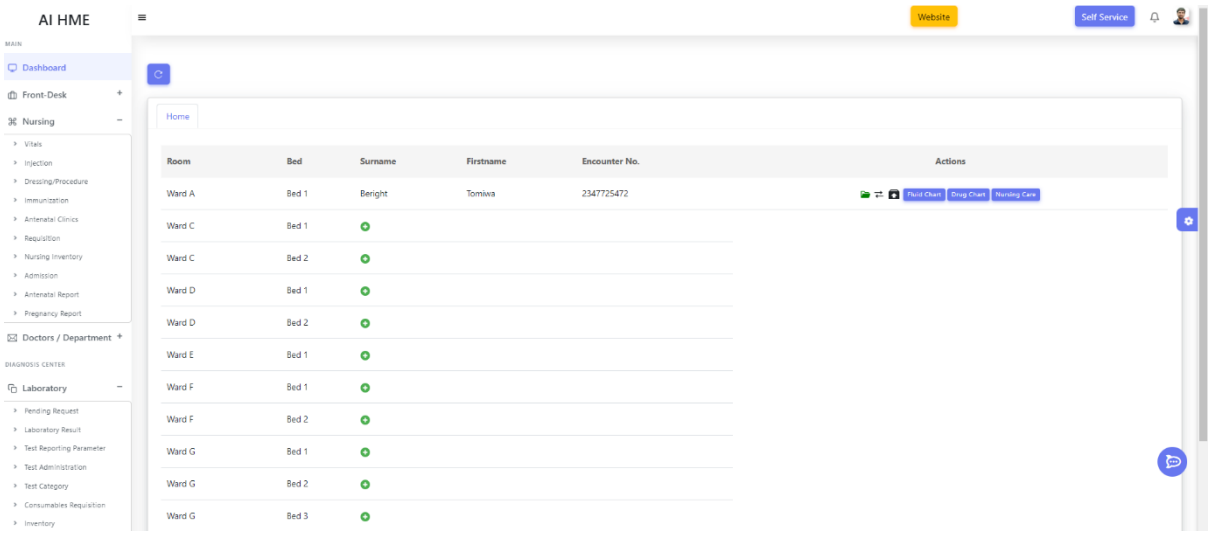


Figure 8.7 Patient Data Update



#	Name Of Patient	Encounter Date & Time	Date	Last Pregnancy Month	Last Pregnancy Outcome	CS	Still Birth	Spontaneous Abortion	Induced Abortion	Previous Pragnancy	Children	E.D.D	L.M.P	Height	Peivic
1	Test Alaba	2023-08-16 12:15:01	2023-08-21		0	0			0	0	0	2023-09-09	2023-08-01	1.6	
2	Nusirah Adegoke	2023-04-14 09:58:53	2023-04-04	5	66	55	1	1	1	1	2	2023-03-29	2023-04-06	50	ss
3	Nusirah Adegoke	2023-04-14 09:58:53	2023-04-01	0	0	0	0	0	0	0	0	2023-10-20	2023-01-17	5.4cm	0
4	Onome Maduka	2023-04-07 14:19:25	2023-04-30	1	nil	nil	nil	nil	yes	1	none	2023-06-12	2022-09-06	68	nil
5	Adejoke Kafayat Akinbambo	2023-04-07 12:09:12	2023-04-30	3	nil	nil	nil	1	nil	3	3	2023-11-12	2023-02-05	49	nil
6	Folake Olorundare	2023-04-07 11:56:14	2023-04-30	2	nil	yes	nil	yes	yes	2	2	2023-07-27	2022-10-19	62	nil
7	Taofekat Akinwunmi	2023-04-07 11:51:14	2023-04-30	1	nil	yes	nil	nil	nil	1	1	2023-05-05	2022-07-28	59	nil
8	Oluwabusayo Adegbola	2023-04-07 11:41:25	2023-04-30	1	nil	nill	nill	nill	nill	1	nill	2023-04-08	2022-07-01	59	nil
9	Abosede Ajijola	2023-04-07 10:37:00	2023-04-30	1	nill	yes	nil	nil	nill	1	1	2023-04-20	2022-07-13	48	nill
10	Opeyemi Ademola	2023-04-07 10:35:34	2023-04-30	2	nil	nil	nil	2	2	2	2	2023-06-25	2022-09-19	5.2	
11	Temitope Bello	2023-04-07 10:10:18	2022-11-11	9 months	SVD-M/A	NIL	NIL	NIL	NIL	1- M/A	1	2023-04-22	2022-06-01	5FT	gravidly enlarged, mwr, nil are
12	Felicia Aderibigbe	2023-04-07 10:39:07	2022-04-30	3	NIL	NIL	NIL	NIL	NIL	3	3	20223-05-07	2022-07-21	5..8	NIL

Figure 8.8 Pregnancy Reports



Room	Bed	Surname	Firstname	Encounter No.	Actions
Ward A	Bed 1	Beright	Tomiwa	2347725472	Fluid Chart, Drug Chart, Nursing Care
Ward C	Bed 1				
Ward C	Bed 2				
Ward D	Bed 1				
Ward D	Bed 2				
Ward E	Bed 1				
Ward F	Bed 1				
Ward F	Bed 2				
Ward G	Bed 1				
Ward G	Bed 2				
Ward G	Bed 3				

Figure 8.9 Rooms and Bed Allocation

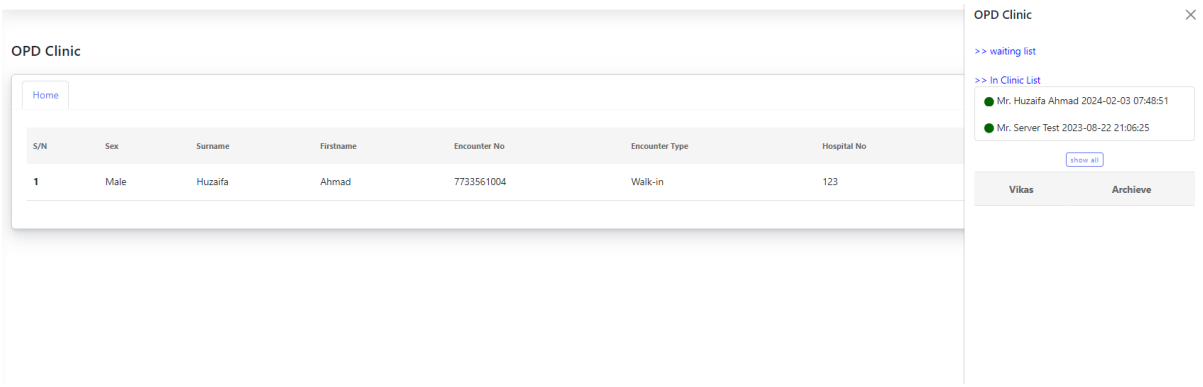


Figure 8.10 OPD Clinic

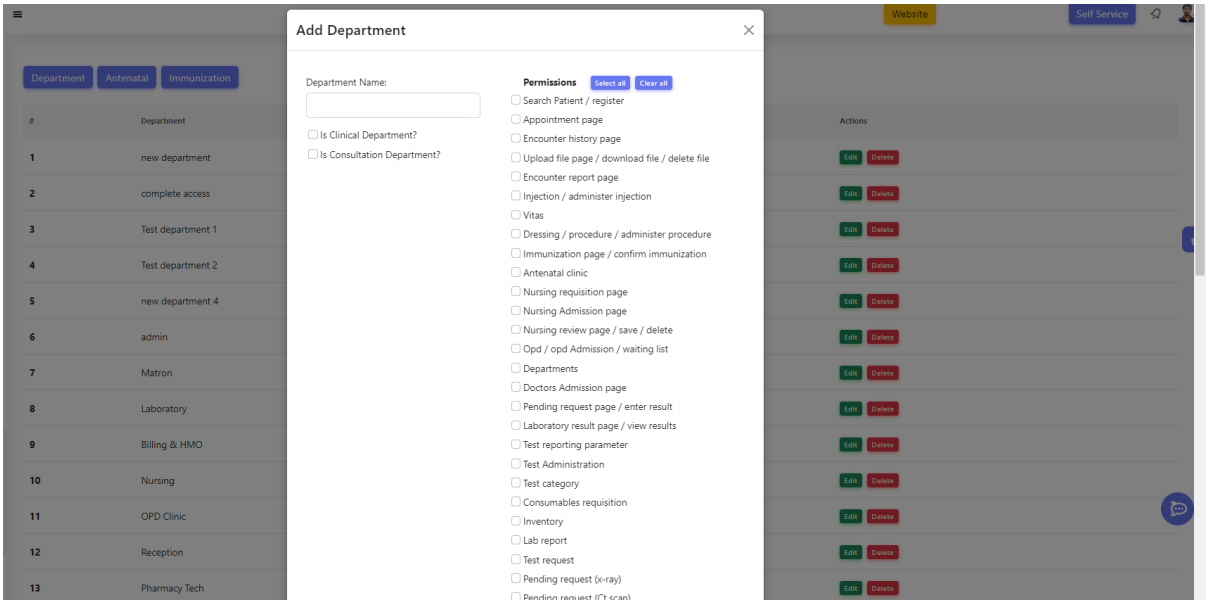


Figure 8.11 Authorization

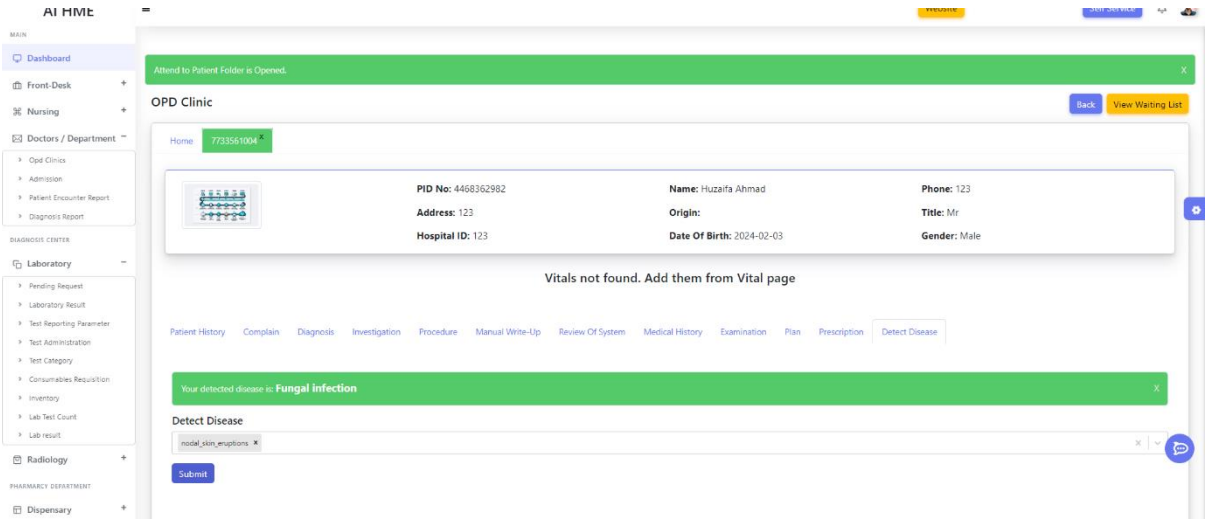


Figure 8.12 General Disease AI Model

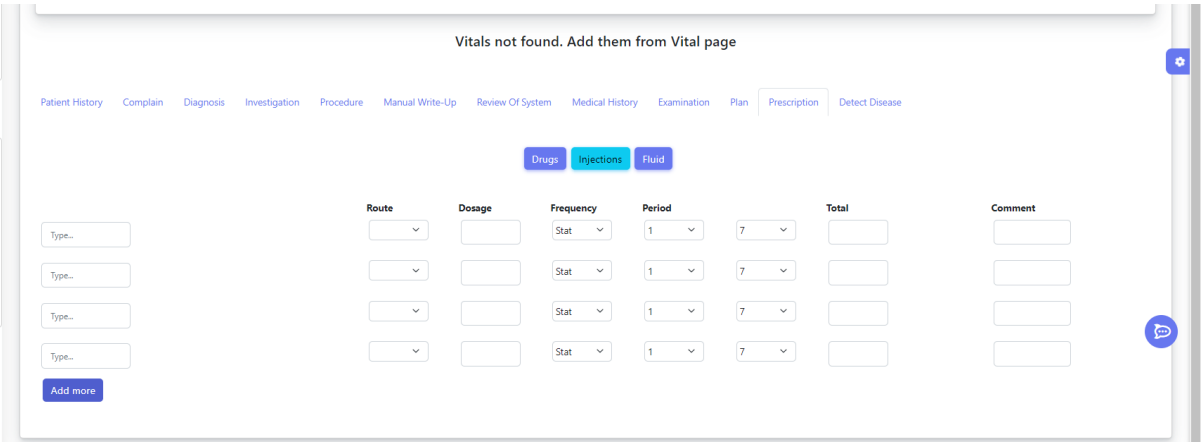


Figure 8.13 Patient Vitals Details

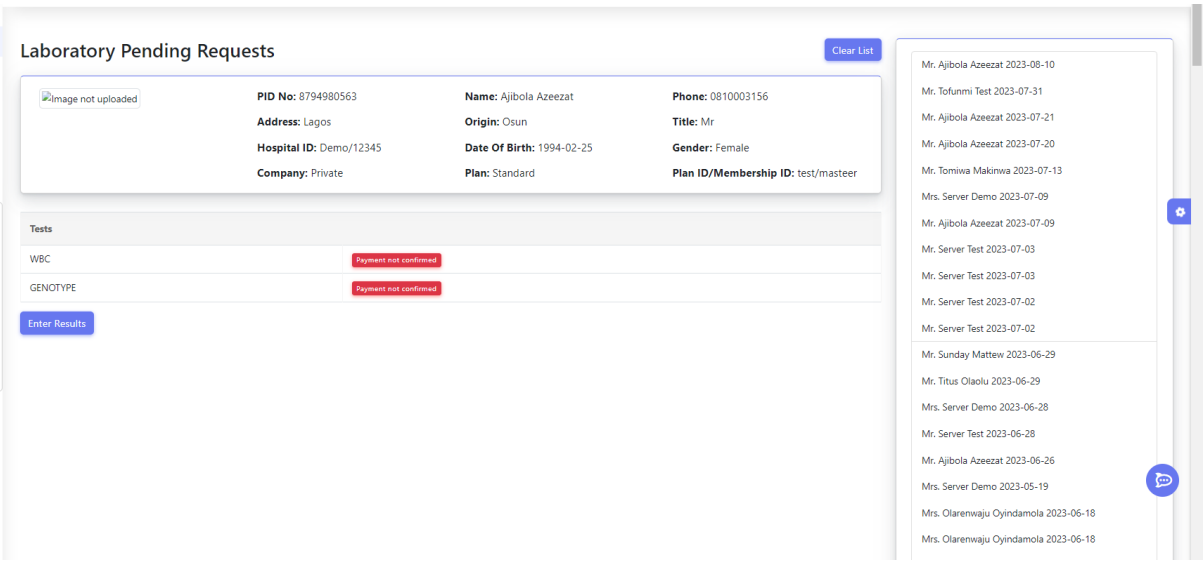


Figure 8.14 Laboratory

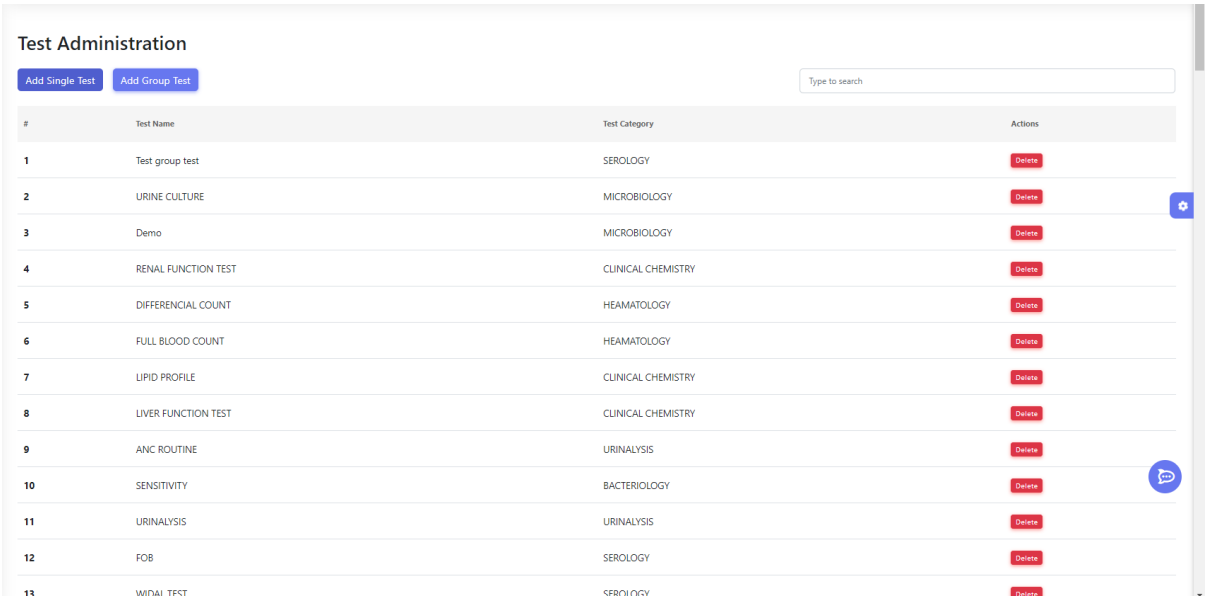


Figure 8.15 Laboratory Tests

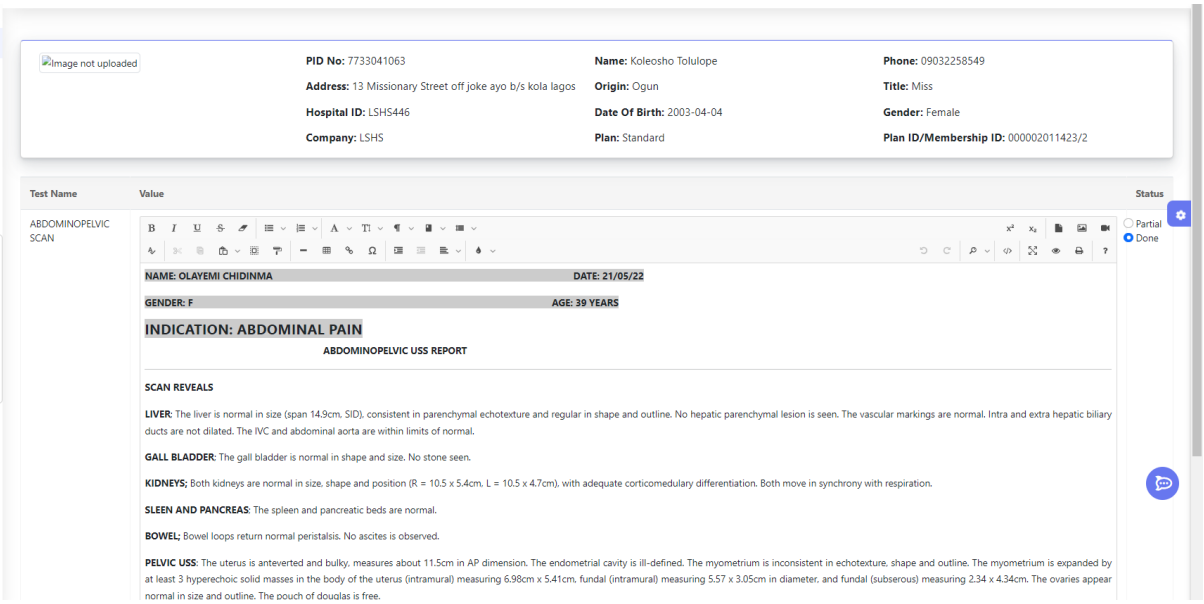


Figure 8.16 Laboratory Test Reports

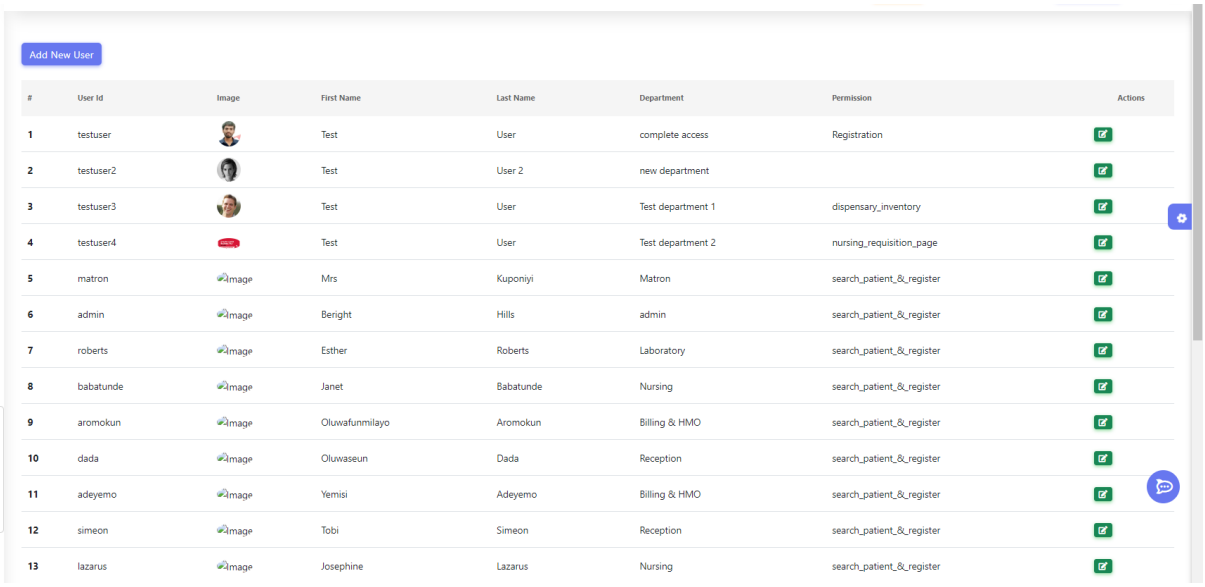


Figure 8.17 User Management

References:

1. Davenport, T., & Kalakota, R. (2019). The potential for artificial intelligence in healthcare. Future healthcare journal, 6(2), 94.
2. Jiang, F., Jiang, Y., Zhi, H., Dong, Y., Li, H., Ma, S., ... & Wang, Y. (2017). Artificial intelligence in healthcare: past, present and future. Stroke and vascular neurology, 2(4), 230-243.
3. Patel, V. L., Arocha, J. F., & Amador, P. (2001). Cognitive models of medical problem-solving: Role of illness scripts. Studies in Health Technology and Informatics, 84, 751-755.
4. Ramesh, A. N., Kambhampati, C., Monson, J. R. T., & Drew, P. J. (2004). Artificial intelligence in medicine. Annals of the Royal College of Surgeons of England, 86(5), 334.
5. Topol, E. (2019). Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again. Basic Books.
6. A. C. &. G. S. Müller, Introduction to Machine Learning with Python: A Guide for Data Scientists, Sebastopol, CA: O'Reilly Media, 2016.
7. Davenport, T., & Kalakota, R. (2019). The potential for artificial intelligence in healthcare. Future healthcare journal, 6(2), 94.
8. Jiang, F., Jiang, Y., Zhi, H., Dong, Y., Li, H., Ma, S., ... & Wang, Y. (2017). Artificial intelligence in healthcare: past, present and future. Stroke and vascular neurology, 2(4), 230-243.
9. Patel, V. L., Arocha, J. F., & Amador, P. (2001). Cognitive models of medical problem-solving: Role of illness scripts. Studies in Health Technology and Informatics, 84, 751-755.
10. Ramesh, A. N., Kambhampati, C., Monson, J. R. T., & Drew, P. J. (2004). Artificial intelligence in medicine. Annals of the Royal College of Surgeons of England, 86(5), 334.
11. Topol, E. (2019). Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again. Basic Books.
12. A. C. &. G. S. Müller, Introduction to Machine Learning with Python: A Guide for Data Scientists, Sebastopol, CA: O'Reilly Media, 2016.
13. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Retrieved from https://arxiv.org/abs/1506.02640
14. Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Retrieved from https://arxiv.org/abs/1612.08242
15. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. Retrieved from https://arxiv.org/abs/1804.02767

16. Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. Retrieved from https://arxiv.org/abs/2004.10934
17. Ultralytics. (2020). YOLOv5. Retrieved from https://github.com/ultralytics/yolov5
18. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems (NIPS)*. Retrieved from https://arxiv.org/abs/1506.01497
19. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. *Proceedings of the European Conference on Computer Vision (ECCV)*. Retrieved from https://arxiv.org/abs/1512.02325
20. Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). Focal Loss for Dense Object Detection. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Retrieved from https://arxiv.org/abs/1708.02002
21. https://reactjs.org/docs/getting-started.html
22. https://reactjs.org/docs/getting-started.html
23. https://stackoverflow.com/
24. https://www.youtube.com/watch?v=8Q_QQVQ1HZA&list=PLfFghEzKVmjuhQwKhYXvdU94GSU-6Jcjr
25. https://www.youtube.com/watch?v=8kv_njhOeZ0&pp=ygUaQUkgYmFzZSBkaXNlYXNlIHByZWRpY3Rpb24%3D
26. www.kaggle.com
27. www.chat.openai.com
28. www.youtube.com
29. Collaborations with Doctors and related staffs