BSCS-S24-013

03-134211-004 AHSAN IQBAL

03-134202-073 NAIF ALI

AI Based Security for Medical Sensor Networks in HealthCare

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: DR NADEEM SARWAR

Department of Computer Sciences

Bahria University, Lahore Campus

January 2025

C e r t i f i c a t e

We accept the work contained in the report titled

"AI Based Security for Medical Sensor Networks in HealthCare"

written by

AHSAN IQBAL

NAIF ALI

as a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: DR NADEEM SARWAR

_____ (Signature)

December 24, 2024

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134211-004	AHSAN IQBAL	
03-134202-073	NAIF ALI	

Date : December 24, 2024

ACKNOWLEDGEMENTS

We would like to thank everyone who contributed to the successful completion of this project. Our deepest gratitude goes to our supervisor, Dr. Nadeem Sarwar, whose invaluable advice, insightful guidance, and unwavering support have been instrumental in the success of this project. His expertise, encouragement, and constant willingness to help us navigate challenges have not only made this journey smoother but also profoundly enriching. We are immensely grateful for his enormous patience, constructive feedback, and for inspiring us to strive for excellence throughout the development of this project. This work is dedicated to my younger brother, Hafiz Ali Hussain Iqbal, and to all those who believed in us, including our loving parents, whose continuous support and belief in our potential have been fundamental to our achievements.

AHSAN IQBAL
NAIF ALI

AI Based Security for Medical Sensor Networks in HealthCare

ABSTRACT

In smart healthcare systems, the security of medical sensor networks is critical due to the sensitive nature of patient data and the increasing threat of cyberattacks. However, current security measures often fail to fully protect these networks from unauthorized access and data breaches. To tackle these challenges, we propose an **AI-powered security framework** aimed at safeguarding patient data and enhancing trust in healthcare technology. Our system employs **ensemble learning models**, such as **Gradient Boosting and Random Forest**, to detect and mitigate cyber threats in medical sensor networks. These models, integrated with an Intrusion Detection System (IDS), actively monitor network traffic to identify any anomalous behaviour that could indicate potential security breaches. The system has been trained on the **CICIoMT2024** dataset, which contains various attack scenarios targeting medical sensor networks. Our models have demonstrated up to **99%** accuracy in detecting anomalies and ensuring effective threat mitigation. Furthermore, the system strengthens access control, blocking unauthorized access and preserving the integrity of patient data. Through thorough testing and validation, our AI-driven security framework has proven to enhance the security of smart healthcare systems, improve patient privacy, and foster greater trust in healthcare technologies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

The breakthrough of medical technologies has made it possible to monitor patient health in real-time via medical sensor networks with growing dependence. These networks consist of devices, either wearable or implanted, that collect and transmit patient data to healthcare providers. As it has evolved this way, so it has its issues of securing sensitive patient data [1], [4]. Cyber-attacks on medical sensor networks are very likely to compromise the integrity, confidentiality, and availability of patient data. The aim of this project is to design and implement an AI-based Intrusion Detection System (IDS) tailored for medical sensor networks, boosting security against unauthorized access and data tampering.

1.2 Problem Statements

This part of our project, we are developing an architecture to address privacy, integrity, and authentication concerns related to medical data collected by sensors integrated into intelligent healthcare systems. These networks focus on gathering personal data from patients, including vital and health-related information, to support various healthcare applications like remote patient monitoring, chronic disease management, and personalized healthcare delivery [2], [3]. However, the inherent vulnerabilities of medical sensor networks make them susceptible to cyber threats such as unauthorized access, data breaches, and denial of service attacks [4], [5]. These risks can pose significant threats to patient privacy, data integrity, and the reliability of automated healthcare services [6].

With the increasing use of digital technologies and the interconnection of devices in healthcare, medical sensor networks require more robust security mechanisms [7]. The core objective of our project is to develop AI-based intrusion detection models for enhanced data management in medical sensor networks [9]. Our goal is to ensure the protection of health data and guarantee that healthcare services remain secure, private, and available. By addressing these security concerns, our project will reduce the risks associated with cyber threats in medical sensor networks and foster trust in intelligent healthcare systems.

1.3 Aims and Objectives

This project aims to develop an AI-based security framework that integrates with medical sensor networks to detect, prevent, and mitigate cyber threats. The specific objectives include.

i) To develop and implement an AI-based Intrusion Detection System (IDS) for medical sensor networks.
ii) To monitor sensor data for anomalies that indicate potential cyberattacks and take steps to mitigate any threats.
iii) To provide a web-based platform that allows healthcare administrators to view sensor data, receive security alerts, and generate reports.

1.4 Scope of Project

In this project, we will build a comprehensive and robust AI-based security system that uses machine learning to detect abnormal patterns in medical sensor data, safeguarding the security and integrity of healthcare information. The system will address various types of cyber threats, such as DoS and DDoS attacks, reconnaissance attacks, spoofing, and MQTT (**Message Queuing Telemetry Transport**) protocol attacks [8].

Additionally, the project will integrate the Intrusion Detection System (IDS) seamlessly into a hospital management system, enabling healthcare providers and administrators to receive real-time alerts and detailed security reports through an interactive dashboard. The scope of this system is extendable, offering many opportunities for future integration, allowing it to adapt to new technological advancements and evolving requirements.

Comprehensive Threat Detection Capabilities

The system will be designed to monitor a wide range of intrusions and cyberattacks targeting medical sensor networks. The IDS will be capable of detecting the following types of attacks:

i) Denial of Service (DoS) Attacks: Identifying and mitigating attempts to overload the system, ensuring continuous service availability.

ii) Distributed Denial of Service (DDoS) Attacks: Detecting and countering coordinated attacks originating from multiple sources aimed at overwhelming the network.

iii) DoS Spoofing Attacks: Detecting attempts to impersonate devices to launch denial-of-service attacks.

iv) Reconnaissance (Recon) Attacks: Monitoring for unauthorized probing or scanning activities designed to gather network information for future exploitation.

v) MQTT Protocol Attacks: Protecting the widely used IoT communication protocol from unauthorized access, data leakage, and command injection attacks.

This advanced threat detection will be powered by machine learning algorithms trained on evolving attack patterns. By leveraging real-time analytics and adaptive learning, the system will remain highly effective in detecting cyber threats.

CHAPTER 2

LITERATURE REVIEW

2.1 Medical Sensor Network Datasets and Security Analysis

This section delves into the current landscape of datasets for medical sensor networks. By analyzing existing research, we aim to identify the strengths and weaknesses of current approaches, particularly regarding the availability and utilization of datasets for developing robust AI-based security frameworks as shown in. **Table 2.1** provides a comparison of relevant studies. It focuses on the datasets used, the types of sensors involved, and the security threats addressed.

Table 2.1 Comparison of Medical Sensor Network Datasets

Dataset Name	Year	Devices	Attacks	Health care	IoT
WUSTL EHMS [10]	2020	Windows Laptop, Blood Oxygen Saturation (SpO2), PM4100 Six Pe Board, EKG or ECG	Spoofing and Data alteration	✓	✓
ECU-IoHT [11]	2021	Bluetooth Adapter, wireless network adapter, Windows 10 laptop, Heart rate sensor, Blood pressure sensor, Temperature sensor, Kali laptop, Libelium MySignals	Network scan, Script Injection, ARP spoofing Smurf,DoS,	✓	✓
BlueTack [12]	2022	SpO2, heart rate, and ECG	DDoS, Bluesmack, MITM, and DoS	✓	✓

ICU [13]	2021	response (GSR) Sensor, Galvanic skin, Nasal/Mouth, Barometer, Remote Electrocardiogram, Fire Sensor, (ECG) monitoring, Smoke Sensor Solar Radiation Sensor, Pulsoximeter (SPO2), CO Sensor, Infusion Pump Glucometer, monitor Sensor Blood pressure, AirFlow Sensor, Electromyography (EMG), Body Temperature Sensor Sensor, Air Temperature Sensor Air Humidity Sensor	SlowITE, and brute force, DDoS MQTT, MQTT publish flood	✓	✓
IEC [14]	2021	Industrial Healthcare equipment, SDN Switch	MITM, Traffic Sniffing, DoS, Unauthorized Access	✓	✓
CICIoMT2024 [15]	2024	Sense-U Baby Monitor, SOS Multifunctional Pager, SINGCALL SOS Button, Ecobee Camera, blink mini, M1T laxihub, owltron, TP-Link_CIC (AP2), Raspberry pi 4 (4), iPad, TP-Link_CICIoT_Doctor (AP1), Lookee Sleep ring, Powerlabs HR Monitor Arm band, COOSPO 808s Chest HR Monitor, COOSPO HW807 Armband, Livlov Heart Rate Sensor,	Denial of Service (DoS) attacks, Distributed Denial of Service (DDoS) attacks, DoS spoofing attacks, reconnaissance (Recon) attacks, MQTT protocol attacks	✓	✓

CICIoMT2024 [15]	2024	Wellue O2 Ring - 3438, Lookee O2 Ring, Checkme BP2A, SleepU Sleep Oxygen Monitor, Rhythm+ 2.0, Wellue Pulsebit EX, Kinsa Thermometer, Checkme O2 Wrist Pulse Oximeter (2), Dell CICM99, Samsung A11. Simulated devices: Withings BPM Connect, Withings Thermometer, Lookee Ring-Pro Sleep Monitor, Qardio Base 2, Wellue EKG, iHealth Smart Wireless Gluco-Monitoring System, Wellue Visual Oxy Wrist Pulse Oximeter, Nasal/Mouth Air Flow Sensor, EMG (Electro-myography Sensor), GSR (Galvanic Skin Response Sensor), Industrial devices, UASure II Meter, Fall Detector, Baby Sleep Position - SenseU Baby, Spirometer		✓	✓

2.2 Analysis of IoT IDS: Papers, Models, and Datasets

Table 2.2 provides Comparison of IDS in IoT and Medical Networks. It focuses on the intrusion detection system in IoT and Medical Networks.

Table 2.2 Comparison of IDS in IoT and Medical Networks

Paper	Techniques Used	Dataset	Accuracy
Ensemble-Based Deep Learning Models for IoT IDS [13]	CNN-LSTM, CNN-GRU, Ensemble Voting	Custom Dataset	99.7% (CNN-LSTM), 99.6% (CNN-GRU)
Deep Learning Models for IoT IDS [14]	LSTM, DAE-SVM	NSL-KDD, UNSW-NB15	100% (LSTM)
An Ensemble Approach to IDS in IoT [15]	SVM, Decision Tree, RF	KDDCUP99	99.8%
Hybrid IDS for IoT [16]	CNN, RNN (GRU/LSTM)	CICIDS2017	99%
Lightweight IDS for IoT [17]	CNN, GRU	Bot-IoT	97%

2.3 User Classes and Characteristics

The system will have multiple user classes with varying access levels and characteristics:

i) **Admin:** Manages system, oversees user roles, approves actions, monitors health data, and reviews intrusion alerts.

ii) **Doctor:** Can view patient data and their vital signs sensor data and manage patient appointments and treatment.

iii) **Patient:** Can view real-time health metrics and historical data, book and

manage appointments, and view/download invoices after discharge and admin approval.

iv) **AI-Based IDS:** Detects anomalies and cyberattacks by analyzing sensor network traffic and reports to the admin.

2.3.1 Admin Panel

Role: The admin oversees all operations of the system.

Responsibilities:

i) **System Management:** Admin is responsible for managing user access (doctors and patients), ensuring system security, and maintaining the system's operational efficiency.

ii) **Alert Monitoring:** Admin reviews alerts generated by the AI-based Intrusion Detection System (IDS) to identify anomalies or potential intrusions in the medical sensor network.

iii) **Approve/Reject Actions:** Admin approves or rejects doctor job applications, patient admissions, and appointments.

iv) **Invoice Management:** Admin generates and manages patient invoices, including charges for treatments, rooms, and other services.

v) **User & Role Management:** Admin can add, edit, or remove users (doctors, patients), assign roles, and manage appointments.

vi) **Logs and Reports:** Admin monitors system logs and generates reports on security incidents for further analysis.

vii) **Access Level:** Admin has full access to all system components, including user management, patient data, and detailed security logs.

2.3.2 Doctor Panel

Role: The system is primarily used by doctors to monitor the vital health signs of their assigned patients, review appointments, and access patient information.

Responsibilities:

i) **Patient Data Monitoring:** Doctors can view and track real-time health metrics of patients, such as heart rate, temperature, and other sensor data, ensuring the safety and care of the patient.

ii) **View Assigned Patient Data:** Doctors are only permitted to view health data and sensor metrics for the patients assigned to them by the admin.

iii) **Appointment & Patient Management:** Doctors can view patient appointments, mark attendance, and handle discharge details. However, they cannot modify the overall system or patient assignments.

iv) **Historical Data:** Doctors have access to review trends in the health metrics of their assigned patients, which helps them make well-informed decisions.

v) **Access Level:** Doctors have access to patient-specific data, but they do not have administrative or system-wide security privileges.

2.3.3 Patient Panel

Role: Patients can track their health data, including vital signs, and receive notifications.

Responsibilities:

i) **Health Data Monitoring:** Patients can view real-time health metrics like heart rate, temperature, oxygen levels, as well as historical data.

ii) **Appointment Management:** Patients can schedule appointments, check their status (pending or confirmed), and access details about their assigned doctor.

iii) **Invoice Management:** Patients can view and download invoices for hospital services, but only after discharge and admin approval.

iv) **Access Level:** Access is restricted to their personal health data and appointment details.

2.3.4 AI-Based IDS (Intrusion Detection System)

Role: The AI-Based Intrusion Detection System acts as a vital automated tool that inspects medical sensor network traffic for potential signs of cyberattacks or irregularities.

Responsibilities:

i) **Anomaly Detection:** The system continuously monitors network traffic for any abnormal patterns that might signal a cyberattack or malicious activity.

ii) **Alert Generation:** If an anomaly or intrusion is detected, the system immediately triggers an alert to notify the Admin.

iii) **Access Level:** This automated system has full access to all sensor data and network traffic, operating in the background to maintain data security.

2.4 Final Deliverables of the Project and Beneficiaries

As shown in **Table 2.3** which outlines the key beneficiaries of the AI-based security system for intrusion detection in medical sensor networks, detailing the roles and impact on healthcare providers, patients, and medical researchers.

2.4.1 Final Deliverable

i) A hospital management system incorporates an AI-driven intrusion detection module.

ii) A dashboard that enables real-time monitoring of medical sensor data traffic, while also visualizing security alerts.

iii) A reporting system that records security incidents and supports detailed investigations.

Beneficiaries

i) **Hospitals:** Strengthened security measures for patient data within the sensor network.

ii) **Healthcare Providers:** Monitoring tools to safeguard patient data.

iii) **Patients:** Improved data privacy and security, ensuring the integrity of health information.

Table 2.3 Final Deliverable and Beneficiaries of the Project

Final Deliverable	Beneficiaries
AI-driven Security Framework integrated into a website application	Healthcare providers, doctors, patients, and stakeholders in the healthcare industry

2.5 Optional Scope

In the future, the system could be extended to:

i) Integrate with electronic health records (EHRs) for secure data transfer.

ii) Include predictive analytics to forecast future anomalies based on historical data. Enable automatic responses to detected threats, such as blocking malicious IPs or isolating compromised sensors.

2.6 Operating Environment

The system will be deployed on a local network server, depending on the hospital. It will interact with medical sensor networks and transmit data to the IDS.

2.7 Design and Implementation Constraints

i) **Real-time Data Processing:** The system must process random sensor data, generated via Python, with minimal latency to ensure timely updates of health metrics and prompt alerting for anomalies.

ii) **Hardware Limitations:** The system must be optimized for environments with limited processing power, ensuring smooth operation even with the random generation of sensor data. It should be capable of running efficiently on standard hospital infrastructure.

iii) **Scalability:** The system should be designed to scale easily as the hospital grows, supporting additional users (patients, doctors), increased sensor data streams, and additional medical devices without degrading performance.

Assumptions and Dependencies

i) Sensors will transmit accurate and reliable network traffic to the system.

ii) The AI model will be trained using a dataset representing typical network traffic patterns and known attack vectors.

2.9 External Interface Requirements

2.9.1 User Interface

The system will provide administrators and healthcare providers with an intuitive, user-friendly web interface. This interface will allow the admin to:

i) Monitor sensor data and traffic from various medical devices connected to patients.

ii) Visualize security alerts if anomalies or intrusions are detected in the sensor network.

The user interface will be responsive and adapt seamlessly to different devices (e.g., desktops, tablets, and mobile phones), ensuring usability across all platforms. The real-time data will be displayed to the admin and healthcare providers to grasp the security and operational status of the network quickly.

2.9.2 Software Interfaces

The backend system will be built using Django, and machine learning models will be integrated to detect medical sensor data anomalies. The core components of this system include:

i) **Machine Learning Models:** These models, built using ensembled models (Gradient Boosting, Random Forest classifier), will be trained to identify patterns indicative of cyber threats or anomalies in the medical sensor network.

ii) **Django Framework**: The machine learning models will be integrated into the backend using Django, Python-based web frameworks.

As shown in **Figure 2.1** These frameworks will:

i) Handle web requests and responses.

ii) Serve the web interface.

iii) Process sensor data by passing it through the machine learning models.

iv) Trigger security alerts based on anomaly detection.

The backend will ensure secure interaction between the user interface, the machine learning models, and the database where patient data, system logs, and security alerts are stored.

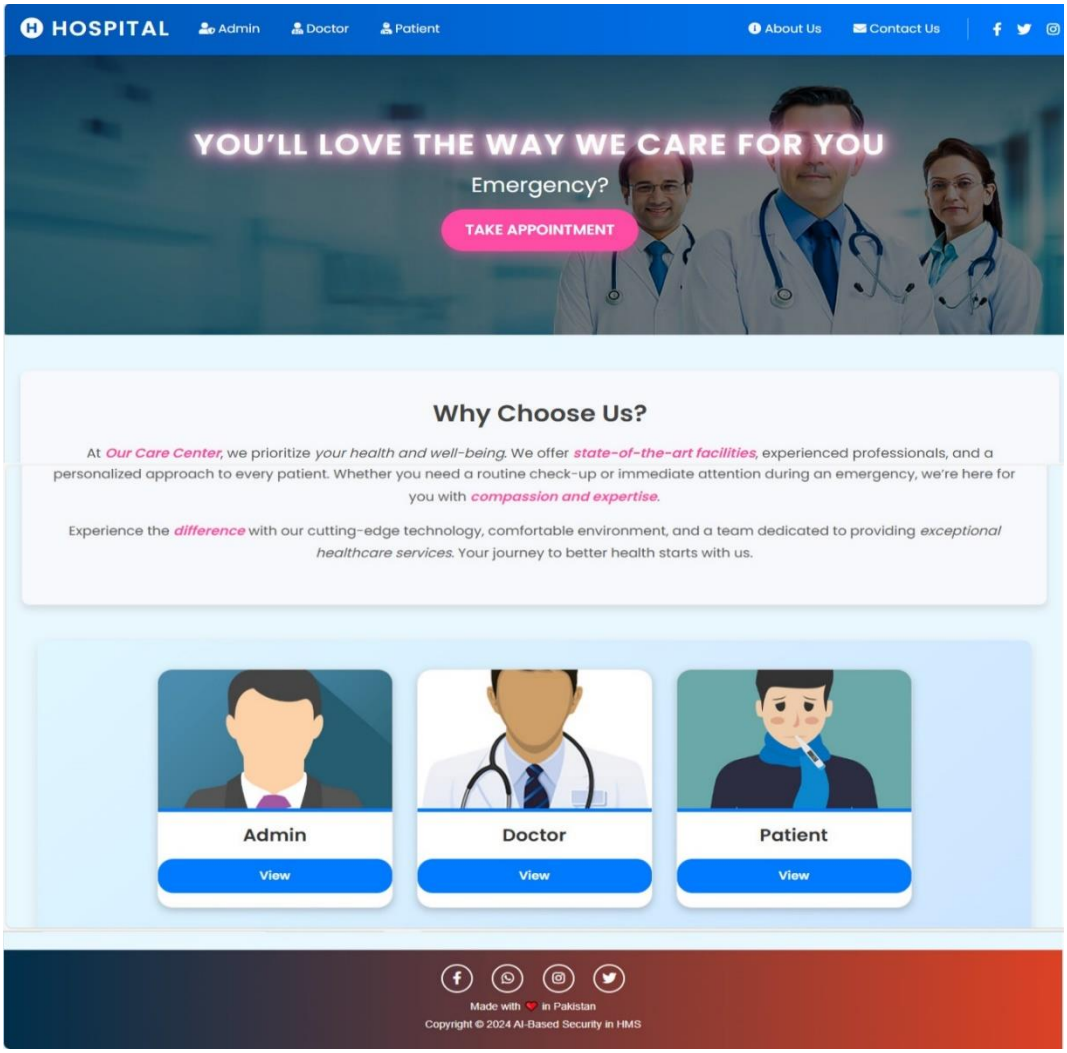

Figure 2.1 Hospital Management System with IDS

2.9.3 Communication Interfaces

The system will use an alternative real-time communication approach (such as AJAX polling or long polling) to ensure continuous data exchange between the medical sensor network and the web interface. This interface will enable:

i) **Live transmission of sensor data:** Continuous real-time streaming of randomly generated sensor data to the web dashboard.

ii) **Instant alerting:** If the machine learning model detects an anomaly, an alert will be immediately sent to the dashboard.

2.10 System Use Case

2.10.1 Use Case Diagram

Figure 2.2 illustrates the Use Case Diagram for Admin Functionalities, where the admin manages user accounts, monitors patient health data, reviews intrusion detection alerts, monitors network traffic, and generates invoices.

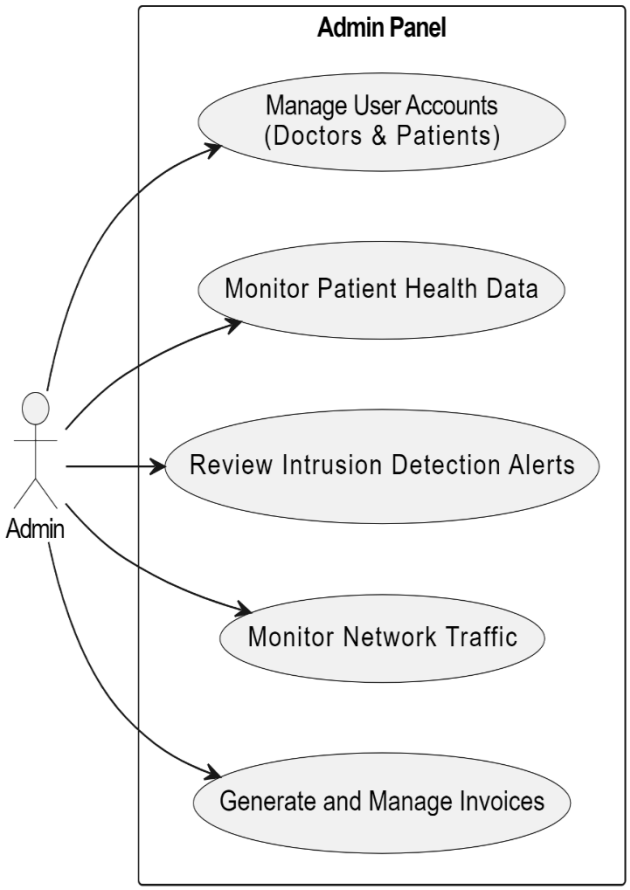

Figure 2.2 Use Case Diagram for Admin functionalities.

Figure 2.3 illustrates the Use Case Diagram for Doctor Functionalities, showing the tasks available to the Doctor, such as viewing patient data, monitoring health metrics, and reviewing health reports.

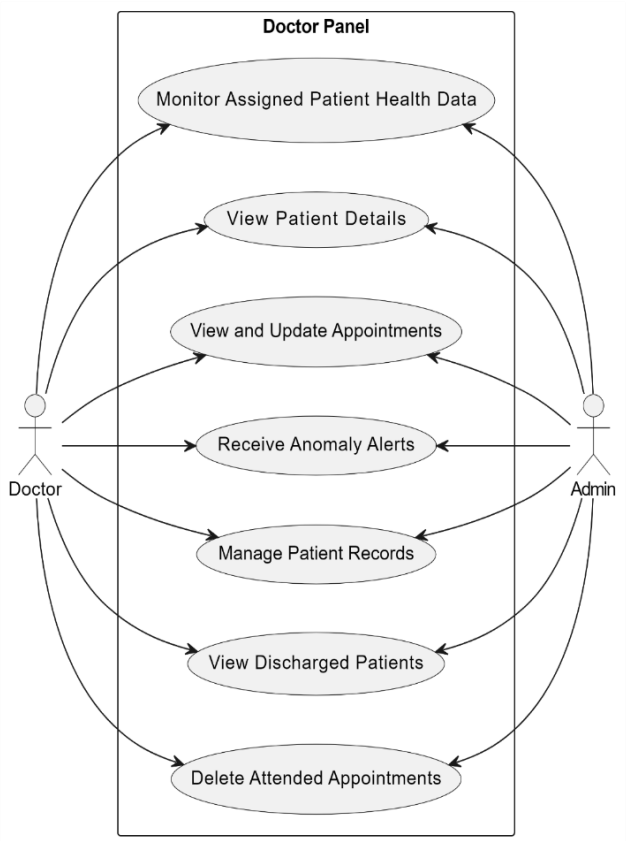

Figure 2.3 Use Case Diagram for Doctor functionalities.

Figure 2.4 illustrates the Use Case Diagram for Patient Functionalities, detailing the actions available to the Patient, such as viewing personal health data and receiving notifications.

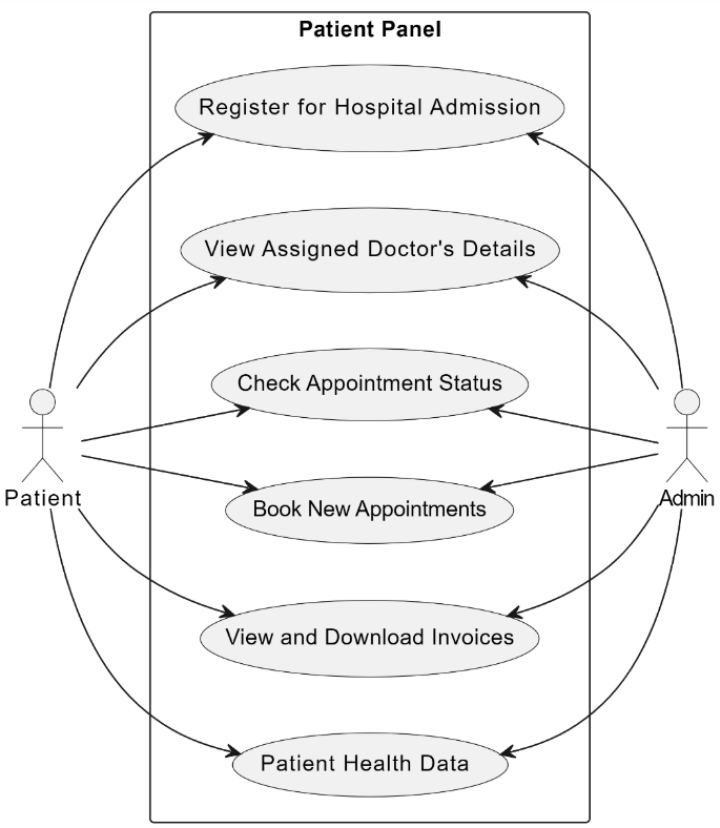

Figure 2.4 Use Case Diagram for Patient functionalities.

Figure 2.5 illustrates the Use Case Diagram for AI-based IDS Interactions, showing the interaction of the AI-based Intrusion Detection System with the system components for monitoring and detecting anomalies

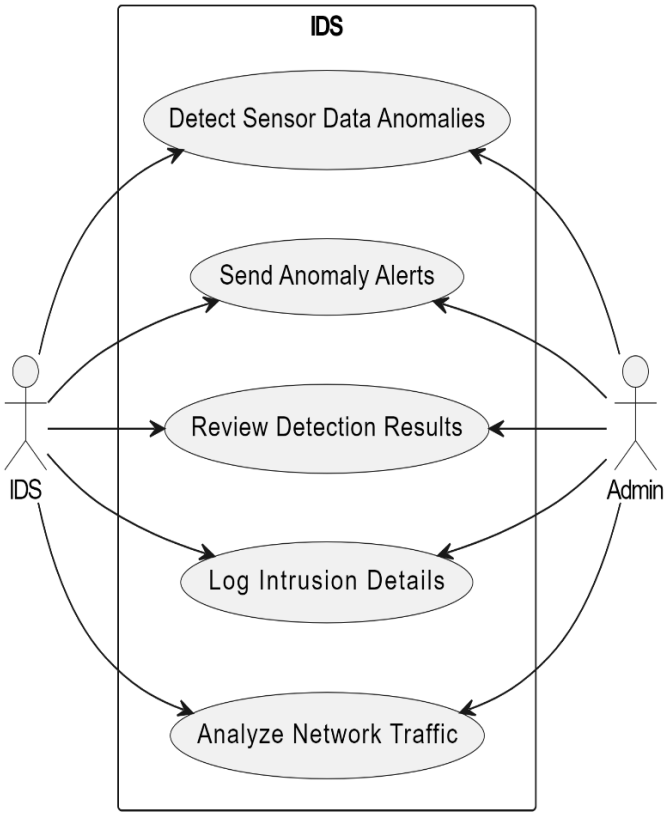

Figure 2.5 Use Case Diagram for AI-based IDS interactions.

2.10.2 Use Case and Identifier

Each use case will be uniquely identified and described, including:

UC1: Monitor anomalies in Network traffic.

UC2: Send alerts to the admin upon detection of threats.

UC3: Train the AI model using historical data.

UC4: Log actions taken by the admin for accountability.

2.11 Use Case Name and Identifier

2.11.1 Manage Users (AI-01-001)

The use case Manage Users (AI-01-001) in **Table 2.4** specifies that the admin can add, update, or remove users, with proper validation for user details, ensuring system updates post-operation.

Table 2.4 Use Case of Manage Users

Name	**Manage Users**
Unique Identifier	AI-01-001
Objective	Allow the Admin to add, update, or remove users
Priority	High
Actors	Admin
Basic Flow	Admin accesses the system, navigates to the user management page, and manages users.
Alternative Flow	The admin is notified and prompted to try again if an error occurs.
Preconditions	Admin is logged in.
Postconditions	Users are updated in the system.
Notes/Issues	Ensure proper validation for user details.

2.11.2 View Security Alerts (AI-01-002)

The View Security Alerts (AI-01-002) use case shown in **Table 2.5** specifies that the admin can access a dedicated page to monitor AI-IDS-generated alerts, which are displayed in order of severity or accompanied by an appropriate message if none exist.

Table 2.5 View Security Alerts Use Case

Name	**View Security Alerts**
Unique Identifier	AI-01-002
Objective	Enable Admin to view alerts generated by the AI-IDS
Priority	High
Actors	Admin
Basic Flow	Admin navigates to the security alerts page, where alerts are displayed.
Alternative Flow	If there are no alerts, an appropriate message is displayed.
Notes/Issues	Alerts should be sorted by severity.

2.11.3 Generate Reports (AI-01-003)

The Generate Reports (AI-01-003) use case shown in **Table 2.6** enables the admin to create various reports on system activity, ensuring accuracy and up-to-date information in the generated reports.

Table 2.6 Use Case to Generate Reports

Name	**Generate Reports**
Unique Identifier	AI-01-003
Objective	Enable Admin to generate various reports on system activity
Priority	Medium
Actors	Admin
Basic Flow	Admin selects the report type, specifies the date range, and generates the report.
Alternative Flow	If the report generation fails, an error message is displayed to the Admin.
Preconditions	Admin is logged in.
Post conditions	Reports are generated and displayed.
Notes/Issues	Ensure reports are accurate and up to date.

2.11.4 View Patient Data (AI-02-001)

The View Patient Data (AI-02-001) use case in **Table 2.7** allows Doctors to access and review patient medical records while ensuring privacy and security are upheld.

Table 2.7 View Patient Data Use Case

Name	**View Patient Data**
Unique Identifier	AI-02-001
Objective	Allow the Doctor to view patient medical records
Priority	High
Actors	Doctor
Basic Flow	Doctor accesses the patient data section and selects a patient to view their medical history.
Alternative Flow	If the patient is not found, an appropriate message is displayed.
Preconditions	Doctor is logged in.
Post conditions	Patient data is displayed.
Notes/Issues	Ensure data privacy and security are maintained.

2.11.5 Monitor Patient Health Data (AI-02-002)

The Monitor Patient Health Data (AI-02-002) use case in **Table 2.8** enables Doctors to track real-time health metrics of assigned patients through a user-friendly dashboard, with alerts for data delays or sensor failures.

Table 2.8 Monitor Assigned Patient Data

Field	Details
Name	Monitor Assigned Patient Health Data
Unique Identifier	AI-02-002
Objective	Monitor real-time health metrics of assigned patients.
Priority	High
Actors	Doctor
Basic Flow	Doctor logs in and views real-time metrics like heart rate, oxygen levels, and temperature for assigned patients.
Alternative Flow	Notify Doctor in case of data delays or sensor failures.
Preconditions	Sensors are functional.
Post conditions	Patient health data is monitored.
Notes/Issues	Provide a user-friendly dashboard for visualization.

2.11.6 View Personal Health Data (AI-03-001)

The View Personal Health Data (AI-03-001) use case in **Table 2.9** enables Patients to access their health data and receive notifications related to health and security alerts, ensuring the information is up-to-date, accurate, and timely.

Table 2.9 View Personal Health Data Use Case

Name	**View Personal Health Data**
Unique Identifier	AI-03-001
Objective	Allow the Patient to view their personal health data
Priority	High
Actors	Patient
Basic Flow	Patient logs into the portal and accesses their health data overview.
Alternative Flow	If the data fails to load, an error message is displayed to the Patient.
Preconditions	Patient is logged in.
Post conditions	Personal health data is displayed.
Notes/Issues	Data must be up-to-date and accurate.
Basic Flow	Patient receives notifications in the portal related to their health and security alerts.
Alternative Flow	If no notifications are available, an appropriate message is displayed.
Preconditions	The patient is logged in.
Post conditions	Notifications are displayed to the Patient.
Notes/Issues	Notifications should be timely and relevant.

2.11.7 View Assigned Doctor's Details (AI-03-002)

The View Assigned Doctor's Details (AI-03-002) use case shown in **Table 2.10** allows Patients to view information about their assigned doctor, ensuring access is restricted to only relevant details.

Table 2.10 View Assigned Doctor's Details

Field	**Details**
Name	View Assigned Doctor's Details
Unique Identifier	AI-03-002
Objective	View details of the doctor assigned by Admin.
Priority	Medium
Actors	Patient
Basic Flow	The patient logs in and navigates to the assigned doctor section to view details like name and contact information.
Alternative Flow	Notify the Patient if no doctor is assigned.
Preconditions	The patient is logged in.
Post conditions	Doctor details are visible.
Notes/Issues	Limit access to only assigned doctor information.

2.11.9 Monitor Sensor Data Traffic (AI-04-001)

The Monitor Sensor Data Traffic (AI-04-001) use case in **Table 2.11** enables the AI-based IDS to continuously monitor sensor data for anomalies, ensuring that operational sensors are properly calibrated and functioning.

Table 2.11 Use Case to Monitor Sensor Data

Name	Monitor Sensor Data Traffic
Unique Identifier	AI-04-001
Objective	Allow the AI-based IDS to continuously monitor sensor data
Priority	High
Actors	AI-Based IDS
Basic Flow	The system continuously collects data from sensors and analyses it for anomalies.
Alternative Flow	If sensor data fails to transmit, the system logs an error and continues monitoring.
Preconditions	Sensors are operational.
Post conditions	Sensor data is monitored continuously.
Notes/Issues	Ensure sensors are properly calibrated and functioning.

2.11.10 Detect Anomalies (AI-04-002)

The Detect Anomalies (AI-04-002) use case shown in **Table 2.12** allows the AI-Based IDS to identify anomalies in sensor data traffic, generating alerts when anomalies are detected and ensuring the machine learning model is properly trained.

Table 2.12 Use Case to Detect Anomalies

Name	Detect Anomalies
Unique Identifier	AI-04-002
Objective	Enable the AI-Based IDS to detect anomalies in sensor data network traffic
Priority	High
Actors	AI-Based IDS
Basic Flow	The AI model analyses incoming sensor network traffic and identifies any anomalies present.
Alternative Flow	If an anomaly is detected, the system generates an alert.
Preconditions	Sensor data is being received.
Post conditions	Anomalies are detected and logged.
Notes/Issues	Ensure the machine learning model is well-trained.

2.12 Other Non-Functional Requirements

2.12.1 Performance Requirements

The system must provide real-time anomaly detection with minimal delays. Response time fordetecting and alerting should not exceed 2 seconds.

2.12.2 Safety Requirements

The system should minimize false positives and negatives to avoid unnecessary interruptions in healthcare services. It should also ensure that no patient data is compromised during its operations. Anomalies must not affect the accuracy of medical data transmitted from sensors.

2.12.3 Security Requirements

Access to the system should be restricted to authorized personnel only, with role-based authentication mechanisms in place for Admins, Doctors, and Patients. Additional logging and monitoring should be implemented to track unauthorized access attempts and ensure accountability.

2.12.4 Software Quality Attributes

Reliability: The system should be consistently available with minimal downtime.

Usability: The interface needs to be intuitive and easy use for healthcare staff.

Scalability: The system must be capable of handling sensor network traffic and an increasing number of patients without performance issues.

Maintainability: The codebase should be modular and well-documented to support future updates and improvements.

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Methodology

The design and implementation of the AI-based security system for intrusion detection in medical sensor networks are carried out in multiple stages shown in **Figure 3.1**. These stages address both functional and non-functional requirements, ensuring that the system operates correctly, efficiently (in terms of performance), and is user-friendly.

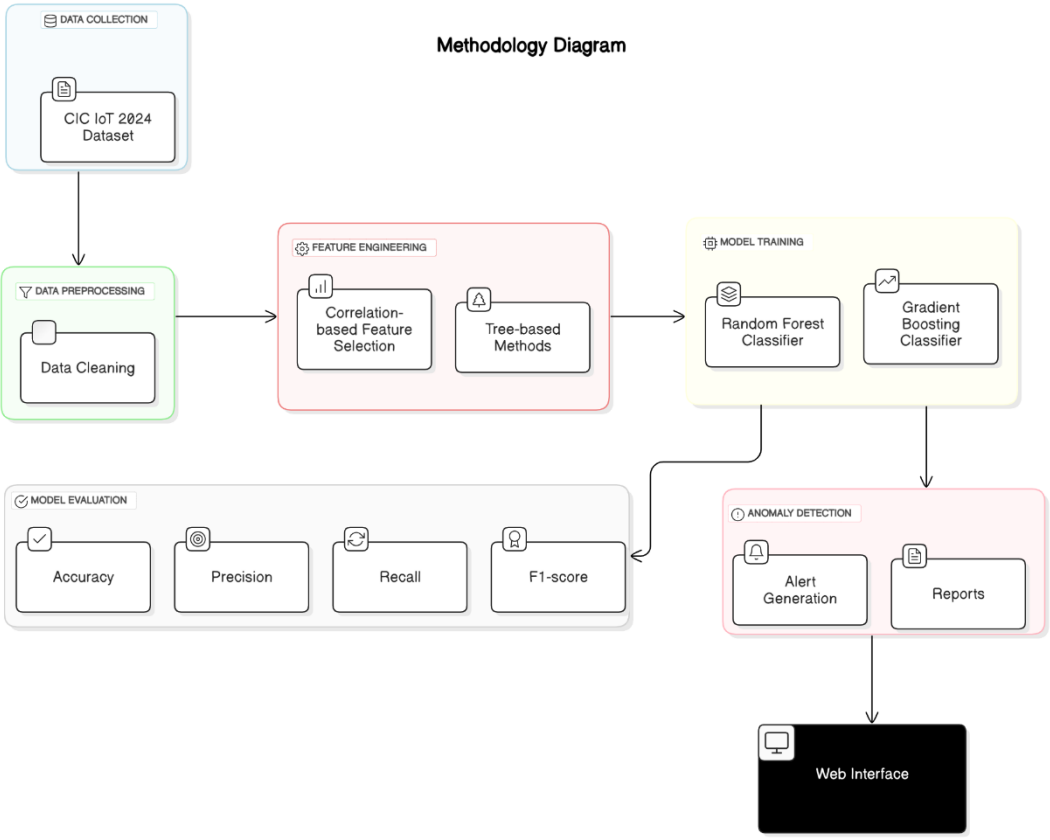

Figure 3.1 Methodology Diagram

3.1.1 Requirements

i) The requirements gathering process identified several crucial components.
ii) Real-time anomaly monitoring to detect network irregularities.
iii) An alert system to notify administrators of potential security breaches.
iv) An AI model to identify and classify types of intrusions based on both historical and real-time data.
v) A secure logging system to track actions taken by administrators.

The system is divided into three main roles:

i) **Admin:** Monitors network status and views logs.

ii) **Sensors:** Collect and send data to the intrusion detection system (IDS).

iii) **AI Model:** Detects anomalies and simulates potential attack scenarios.

3.1.2 Design

The design approach uses a modular architecture, ensuring that each component of the system can be developed and tested independently. The system consists of the following modules:

i) **Sensor Network:** Made up of sensor nodes that monitor health data and network traffic. These nodes transmit packets to the IDS for analysis.

ii) **Intrusion Detection System (IDS):** Processes sensor data, applies the AI model, and detects anomalies in real time. The IDS also simulates potential attacks based on any detected intrusions.

iii) **Alert System:** When an anomaly is detected, the alert system sends notifications to the administrator.

iv) **Logging Module:** Keeps a detailed record of all interactions and actions taken by the administrator to ensure accountability.

3.1.3 Implementation

The system is implemented using a combination of Django for the backend, and a dashboard using for data visualization. The AI model is trained using the CICIoMT2024 dataset and deployed within the IDS.

3.2 Feasibility Plan

3.2.1 Resource Requirements

Hardware:

i) Multiprocessor machines with 4 cores for AI model training.
ii) Multiprocessor machines with 2 cores each for testing and network simulation.

Software:

i) Python-based machine learning libraries (e.g., Scikit-learn, TensorFlow).
ii) Django for backend development. For real-time interaction between the
iii) dashboard and the backend.

3.2.2 Risks Involved

The development of this system presents several potential risks:

i) **False Positives/Negatives:** The AI model may misclassify anomalies, leading to incorrect responses to actual attacks or triggering unnecessary alarms.

ii) **Scalability:** As the sensor network expands, the system must be able to handle the increased data volume efficiently.

iii) **System Latency:** Monitoring could experience latency issues, particularly in larger networks.

3.3 Domain Model

The domain model shown in **Figure 3.2** outlines the core components of the system, including the Admin, Sensor Network, Intrusion Detection System, and Alert System. Each component interacts with others according to its specific role.

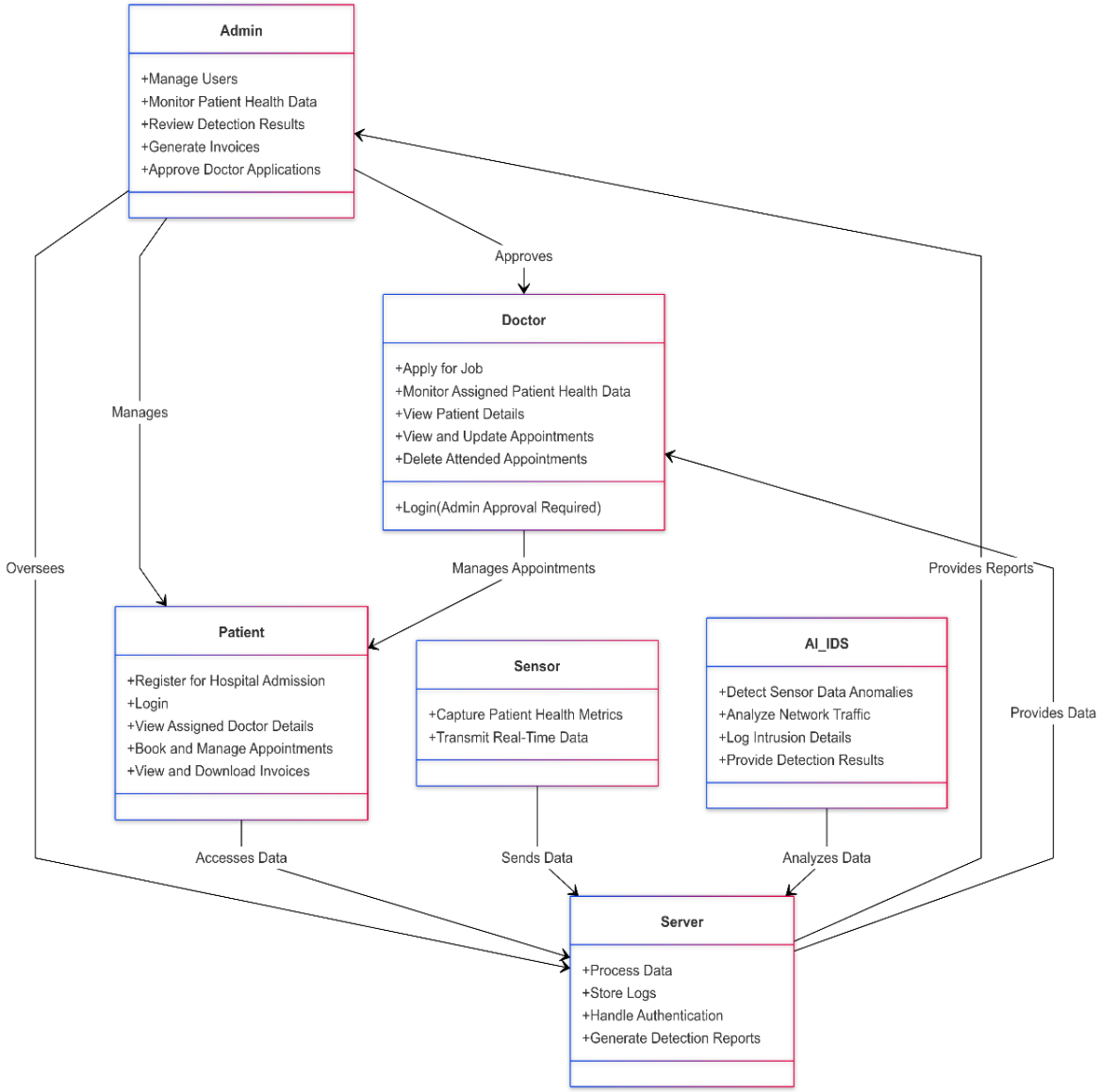

Figure 3.2 Domain Model Diagram

3.4 Sequence Diagram

This section features sequence diagrams that illustrate the interactions and workflows for each user role in the AI-based security system for medical sensor networks. The diagrams display the messages exchanged between the users (actors) and system components, highlighting how users interact with the system to complete various tasks.

3.4.1 Login

The login sequence diagram for the Admin, Doctor, and Patient roles demonstrates how each role interacts with the dashboard and database to authenticate credentials and access their respective panels shown in **Figure 3.3**.

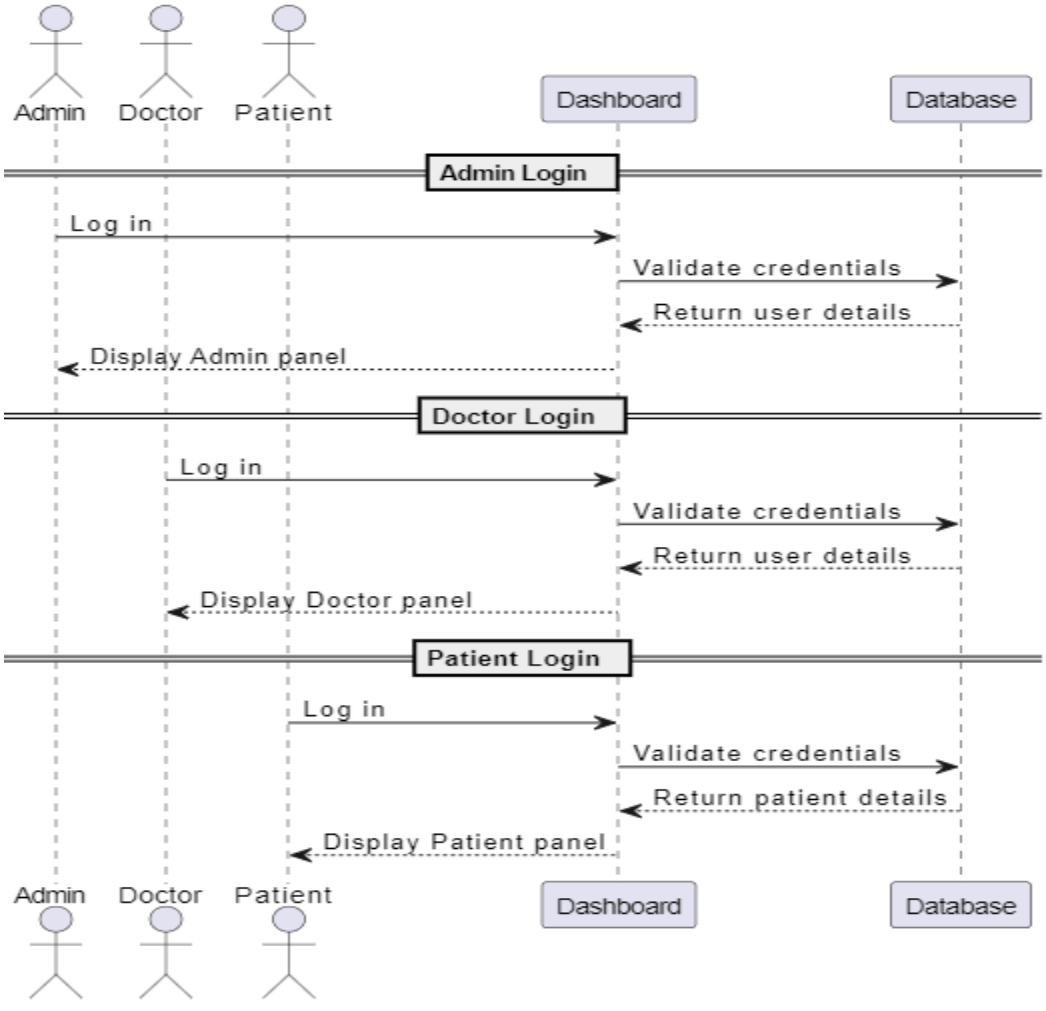

Figure 3.3 Login Sequence Diagram

3.4.2 Admin

This sequence diagram in **Figure 3.4** shows how the admin can view security alerts. The dashboard retrieves the latest alerts from the AI-based IDS and displays them, enabling the admin to respond to potential threats in a timely manner.

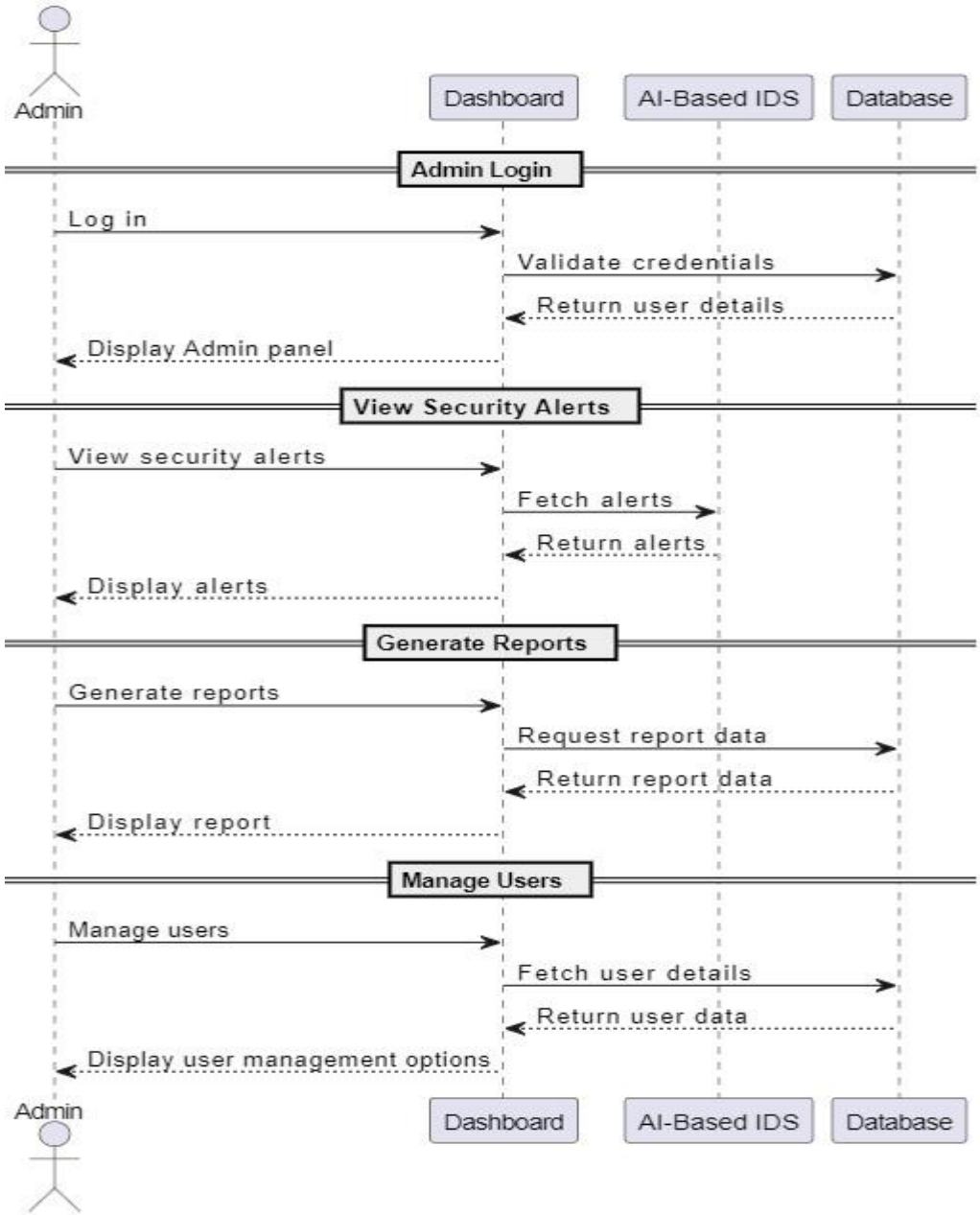

Figure 3.4 Admin Sequence Diagram

3.4.3 Doctors

This sequence diagram in **Figure 3.5** demonstrates how the Doctor interacts with the system, including viewing patient data, receiving security alerts, and addressing detected anomalies.

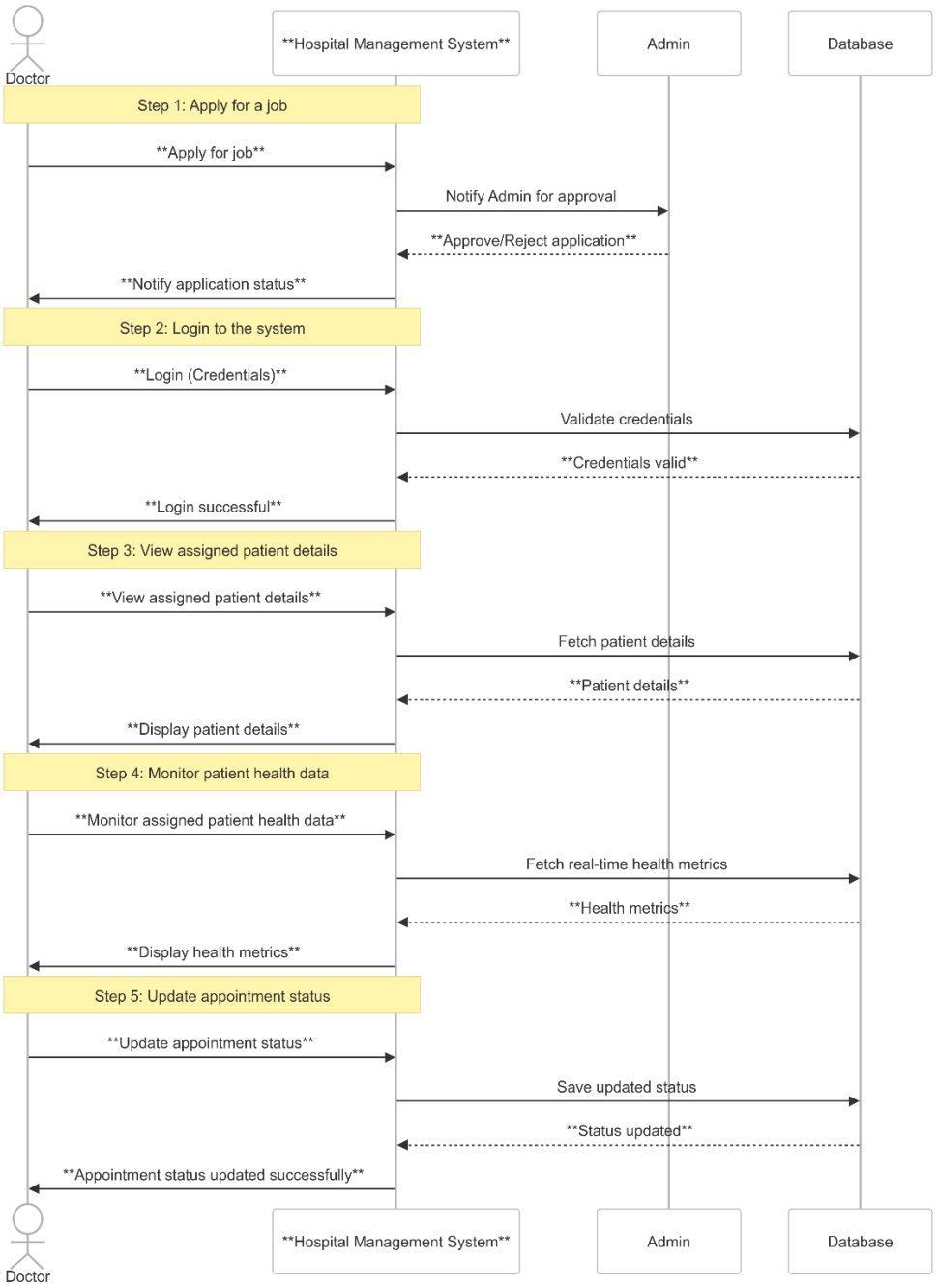

Figure 3.5 Doctors Sequence Diagram

3.4.4 Patients

This sequence diagram illustrates the Patient's interactions with the system, focusing on viewing their personal health data and receiving notifications about any detected anomalies shown in **Figure 3.6**.

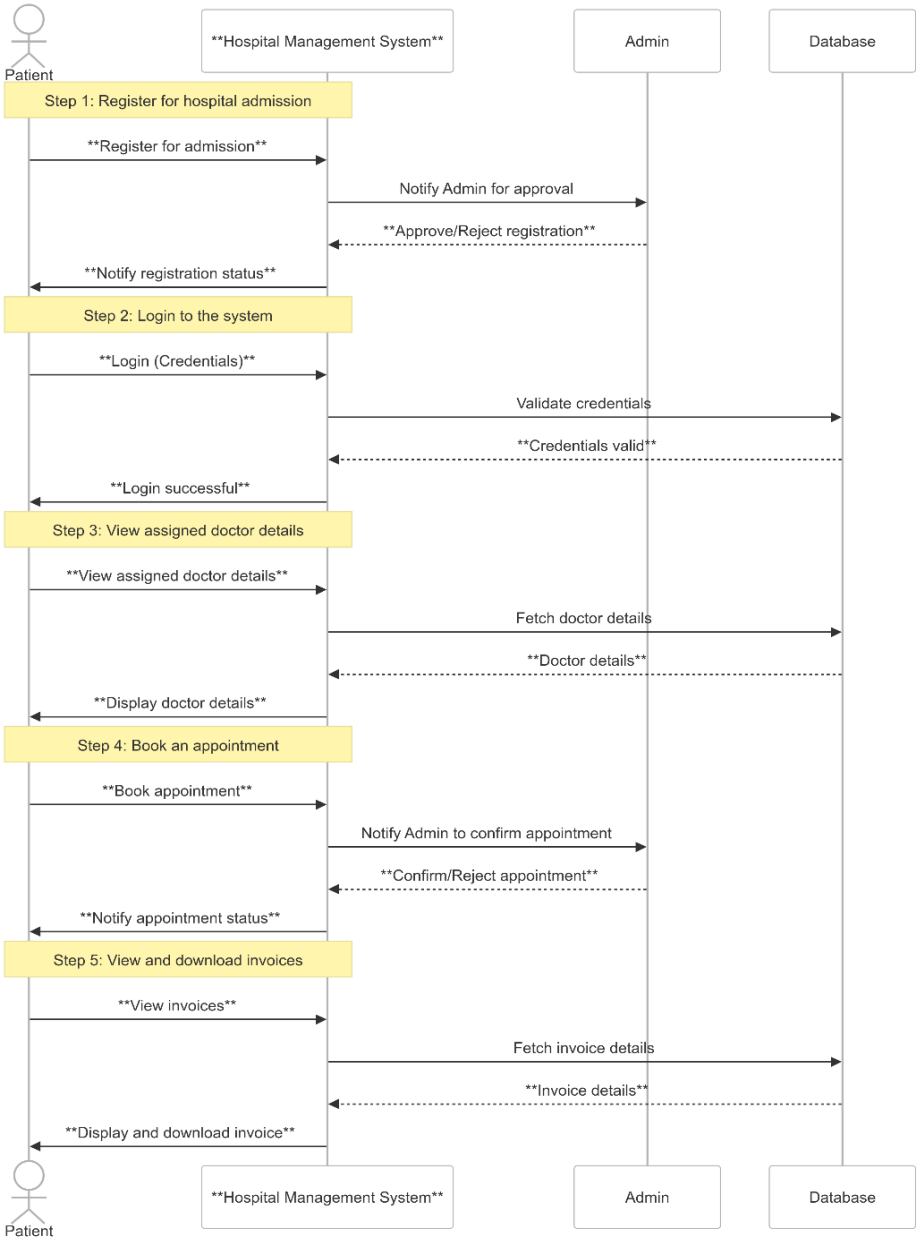

Figure 3.6 Patients Sequence Diagram

3.4.5 New Registration

The sequence diagram for registering a new user as shown in **Figure 3.7** emphasizes the admin's role in completing the registration form, submitting user details to the database, and sending an email notification to the admin upon successful registration. This ensures proper user management and onboarding.

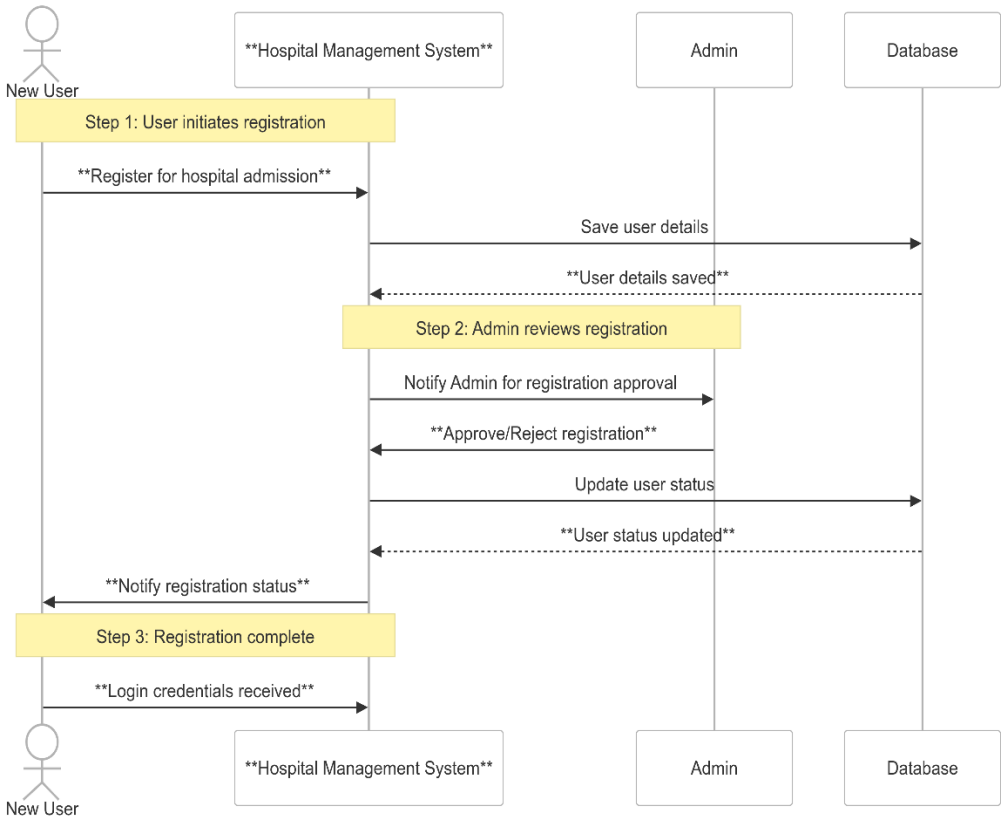

Figure 3.7 New Registration Sequence Diagram

3.5 Class Diagram

The class diagram shown in **Figure 3.5** illustrates how the Admin, Sensor Network, Alert System, and Intrusion Detection System work together during an anomaly detection scenario. Each actor plays a key role in a coordinated effort to identify and address potential security threats.

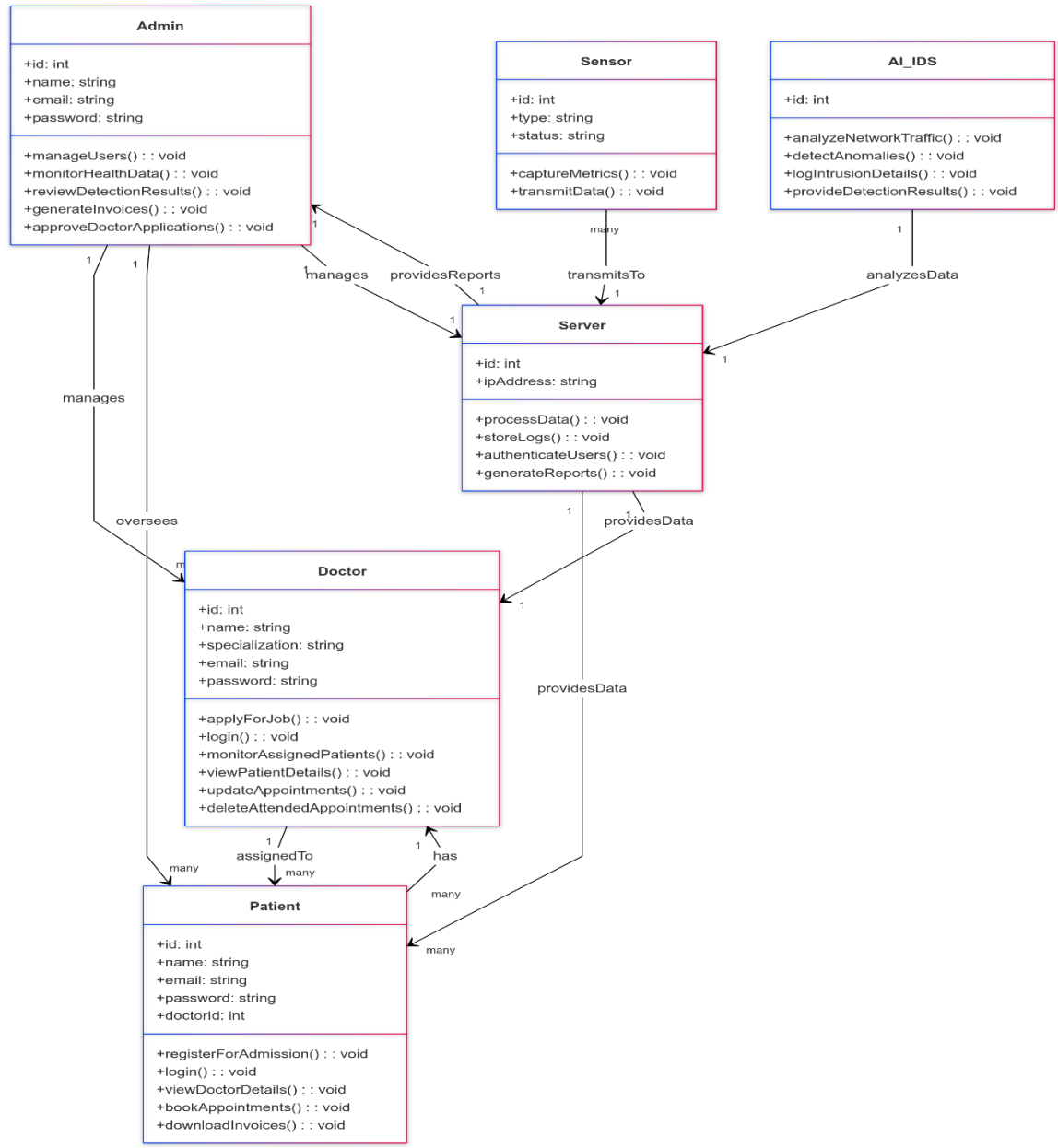

Figure 3.8 Class Diagram Of AI-Based Security

CHAPTER 4

IMPLMENTATION

4.1 Development Model

The development model focuses on creating a secure, efficient, and role-based system for intrusion detection, hospital management, and patient sensor networks. This system guarantees the confidentiality, integrity, and availability of sensitive patient data, health metrics generated by sensors, and network traffic information. Below is a breakdown of the model.

4.1.1 Roles & Responsibilities

Admin:

i) The admin has complete access to the system, including managing users such as doctors and patients.
ii) Responsible for approving or rejecting user actions, including patient admissions and doctor job applications.
iii) Monitors patient health metrics, including real-time sensor data such as heart rate, oxygen levels, and temperature.
iv) Reviews intrusion detection alerts and accesses detailed reports generated by the AI model for further analysis.
v) Regularly audits system logs to ensure operational accuracy and security compliance.
vi) Does not have the authority to block network traffic or disable sensors directly; such actions require escalation to the IT or security team.
vii) The admin has complete access to the system, including managing users such as doctors and patients. The admin can generate and download patient

invoices detailing treatment costs, room charges, and additional expenses. Manages and approves patient appointments, assigning them to the appropriate doctors.

Doctor:

i) Doctors can access the system only after their account is approved by the admin.
ii) Access is limited to the health data and sensor status of their assigned patients.
iii) Can view patient details such as name, symptoms, and contact information.
iv) The doctor dashboard provides a real-time view of assigned patient health metrics, including heart rate, oxygen levels, and temperature.
v) Can monitor trends and historical data for their patients to aid in medical decision-making.
vi) Receives alerts about anomalies in sensor data, such as abnormal heart rates or oxygen levels.
vii) Can view their scheduled appointments and patient discharge details managed by the admin. Does not have access to overall system data, network traffic, or other patients not assigned to them. Can update appointment status or mark it as attended. The doctor dashboard is tailored to display patient-specific data and actionable insights in a user-friendly manner.

Patient:

i) Patients can create an account for hospital admission and log in after receiving approval from the hospital admin.
ii) Can view the details of their assigned doctor, including specialization, contact number, and address.
iii) Can check the status of their booked appointments, displayed as pending or confirmed by the admin.

iv) Could book new appointments, which require approval from the admin before confirmation.
v) Can view and download the invoice in PDF format, but only after being discharged from the hospital by the admin.
vi) Have access to real-time health monitoring data collected from sensors, such as heart rate, temperature, and oxygen levels, displayed in a user-friendly dashboard.

4.1.2 Backend Logic

Implementation Details

i) Django's built-in authentication system (Django. contrib. auth) handles user authentication.
ii) Provides secure login/logout mechanisms and password management.
iii) Utilizes User model for storing credentials, roles, and permissions.
iv) Passwords are hashed and securely stored in the SQLite database using Django's default hashing algorithm (PBKDF2).

Process Flow

i) Users register with their credentials.
ii) Credentials are validated against the SQLite database during login.
iii) A session is created upon successful login, ensuring secure access to the system

Security Enhancements

i) Use Django middleware for protection against common web vulnerabilities like CSRF and XSS.

4.1.3 Database Structure and Usage

SQLite Integration

i) A lightweight database solution used to store user credentials, role assignments, patient data, sensor metrics, and network traffic logs.
ii) Each table is optimized for its specific use case to ensure efficient query performance for real-time operations

4.1.4 Relational Structure Overview

i) **Admin → Doctors:** Admin manages all doctors and their roles within the system.
ii) **Admin → Patients:** Admin oversees all patient-related activities, including health data, doctor assignments, and appointments.
iii) **Doctors → Patients:** Doctors are linked to patients assigned by the admin and can monitor their health data from sensors in real time.
iv) **Patients → Sensor Data:** Patients health metrics are collected via sensors and securely stored, linked to both their profile and assigned doctor.
v) **Admin → Sensor Data:** Admin monitors all sensor data to ensure system-wide consistency and security

4.2 Technical Architecture

This section describes the technology stack and architecture used to provide a secure, scalable, and efficient Hospital Management System integrated with intrusion detection and health monitoring capabilities.

4.2.1 Frontend

Built with HTML, CSS, JavaScript, and Bootstrap to create a responsive and user-friendly interface for different roles.

Admin Dashboard

i) Displays all users, system activity, patient health data, network traffic visualizations, and intrusion alerts demonstrate in **Figure 4.1**.

ii) Enables admins to manage users (doctors, patients), assign roles, and see network traffic log of each patient sensor data.

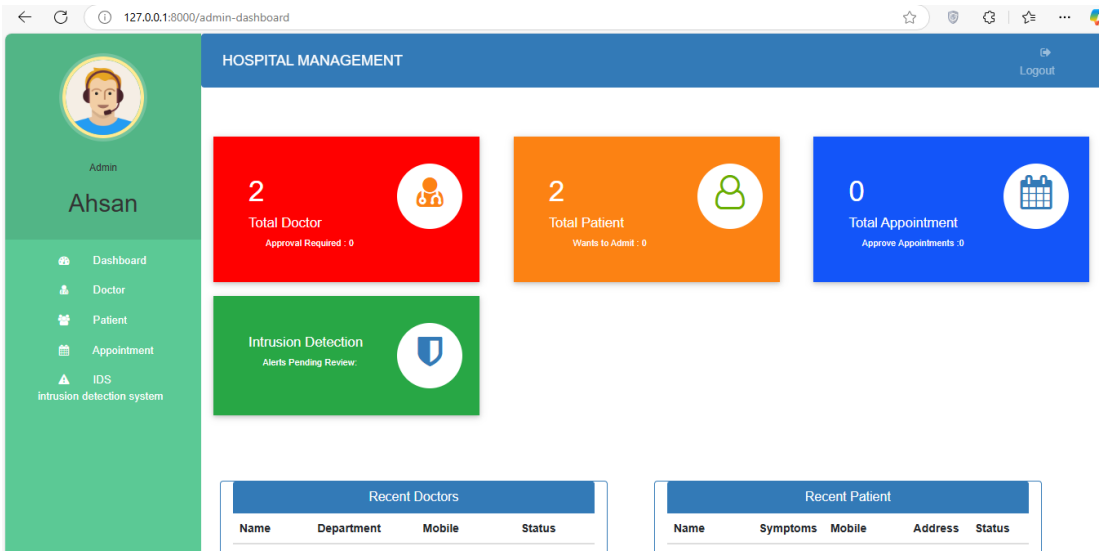

Figure 4.1 Admin Functionalities

Doctor Dashboard

i) As shown in **Figure 4.2** a list of patients assigned to the doctor, along with their personal details and health data.

ii) Provides real-time sensor metrics for each patient, including heart rate, oxygen levels, temperature, and other relevant health parameters.

iii) Allows doctors to view and manage their appointment schedule, including patient details for each appointment.

iv) Displays historical health trends for each patient to assist in tracking progress and making informed decisions.

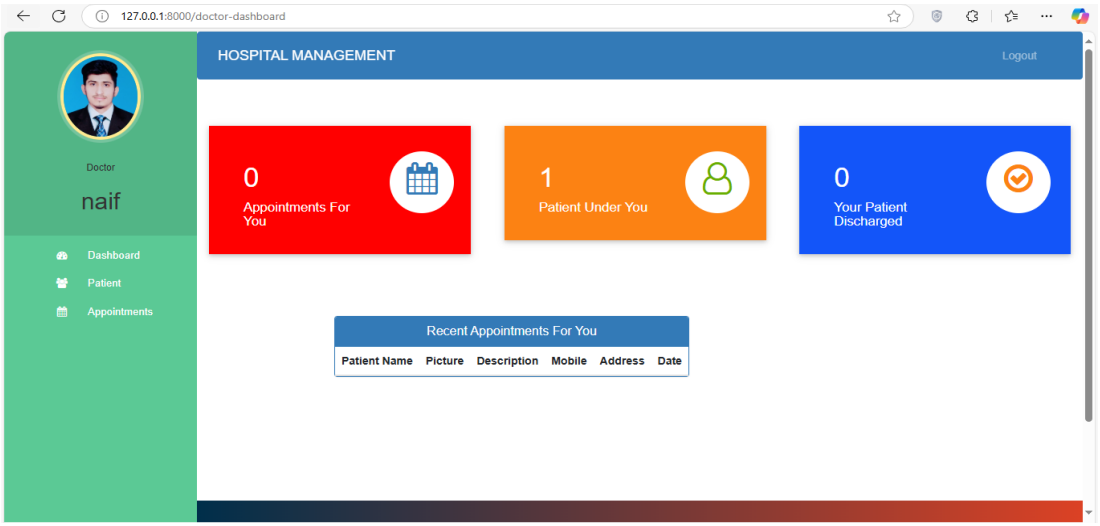

Figure 4.2 Doctor Functionalities

Patient Interface

i) Displays personal health data collected from sensors, such as heart rate, oxygen levels, temperature, and other vital metrics in an easy-to-understand format shown in **Figure 4.3**.

ii) Provides real-time updates of health metrics, helping patients stay informed about their current condition.

iii) Allows patients to view historical trends of their health metrics, enabling them to track progress or changes over time.

iv) Provides details of the assigned doctor, including specialization, contact information, and address, for easy communication.

v) Enables patients to book, view, and manage appointments, showing the status of each appointment (e.g., pending, confirmed).

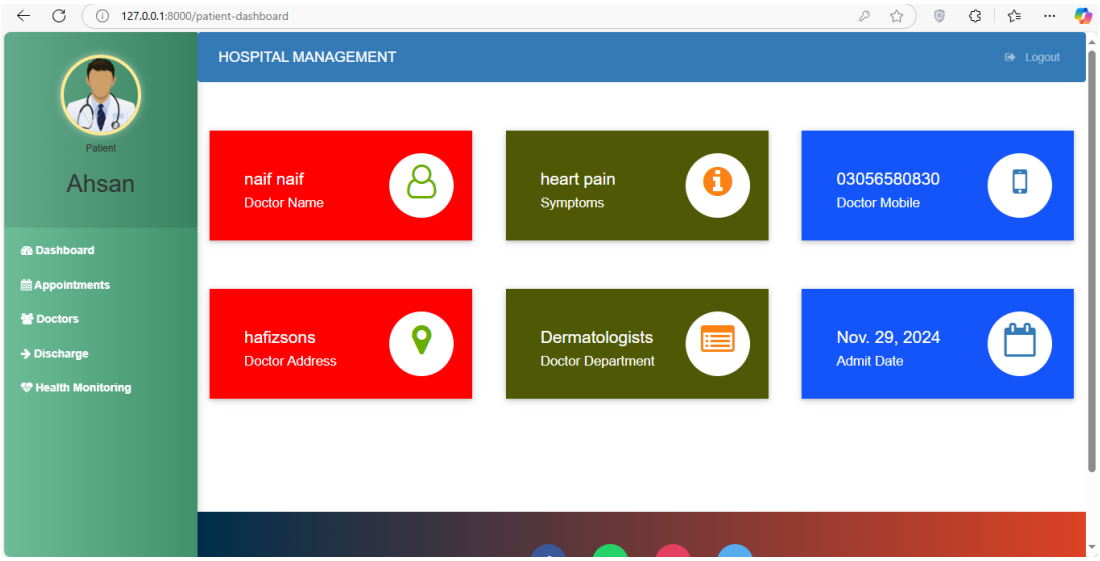

Figure 4.3 Patient Functionalities

4.2.2 Backend

Developed using Django to manage data, enforce role-based access, and process real-time health and network data.

Key responsibilities include:

i) Managing user authentication and role-based access control for admins, doctors, and patients.
ii) Handling health sensor data collection, storage, and real-time processing for alerts.
iii) Serving results from an integrated intrusion detection system to flag suspicious network activity.
iv) Sending role-specific alerts to admins.

4.2.3 Database

SQLite is used during development and testing for its simplicity and ease of integration with Django

Key Roles

i) User roles and permissions (Admin, Doctor, Patient).
ii) Patient health sensor data (e.g., heart rate, temperature).
iii) Network traffic logs associated with patient sensors.
iv) Alerts for anomalies in health data or network activity

4.2.4 AI Model

An **Ensemble Learning** approach was utilized, combining.

i) Random Forest Classifier: For handling complex datasets and identifying patterns in traffic features.
ii) Gradient Boosting Classifier: For fine-tuned performance on edge cases and better accuracy.

4.3 Model Workflow

4.3.1 Data Pre-processing

i) Missing values handled using mean imputation.
ii) Dropped irrelevant columns (e.g., drate).
iii) Key Features: Rate, Srate, Protocol Type, ARP.

4.3.2 Training and Testing

i) Split labelled data into training and test sets.
ii) Trained the ensemble model on key features to classify traffic data.

iii) Evaluated model performance using metrics such as accuracy, precision, recall, and F1-score.

4.3.3 Integration with Django

i) Deployed as part of a backend system in Django.
ii) Processes real-time incoming traffic data.
iii) Analyses packet rates, protocols, and other network parameters.
iv) Flags suspicious activities and alerts administrators and doctors.

4.3.4 Classification Rules

i) **DDoS**: High Rate (>10,000) and Srate (>5,000).
ii) **DoS**: Moderate Rate (>5,000) and Srate (>2,000).
iii) **Recon**: Specific protocol types indicating reconnaissance activities (e.g., Port Scans).
iv) **Spoofing**: Low Rate (<1,000) with unusual ARP activity.
v) **Normal Traffic**: Any traffic not matching the above conditions.

4.4 Data Collection and Preprocessing

4.4.1 Dataset Overview: CIC IoMT 2024

The **CIC IoMT 2024** dataset as shown in **Table 4.1** is a **multi-protocol dataset** designed to simulate attack vectors in healthcare IoMT (Internet of Medical Things) environments. The data captures network traffic generated by IoMT devices such as sensors, cameras, and other medical devices. It aims to assess IoMT device security against various network intrusions.

Table 4.1 CIC IoMT 2024 Dataset Analysis

Attribute	Details
Dataset Name	CIC IoMT 2024
Purpose	Simulating attack vectors and assessing IoMT device security in healthcare.
Total Files	99 CSV files
Total Size	2.03 GB
Data Types	Network traffic data from IoMT devices (sensors, cameras, medical equipment).
Attack Types	DoS, DDoS, Spoofing, Reconnaissance, MQTT Protocol Attacks
Protocols Captured	TCP, UDP, HTTP, MQTT
Key Features	Packet size, timestamp, protocol type, source/destination IPs, attack labels
Use Cases	IDS testing, cybersecurity research, machine learning model development
Dataset Access	CIC IoMT 2024 Dataset [12]

4.4.2 Key Characteristics

Structure

i) Separate train and test folders.
ii) **Unlabelled datasets:** Initially lacked attack_class labels.
iii) **Numerical features**: Includes traffic statistics like Rate, Source Rate, Protocol Type, etc.

Features

i) **Traffic-related metrics**: Rate, Source Rate, Packet Size, IAT (Inter-Arrival Time).
ii) **Protocol details**: Encoded as numeric values in Protocol Type.

iii) **Anomaly indicators**: Flags (e.g., SYN, ACK) and features like ARP.

Labeling Objective

Map rows to attack types such as:

i) Normal traffic.
ii) DDoS (Distributed Denial-of-Service)**.**
iii) DoS (Denial-of-Service)**.**
iv) Recon (Reconnaissance)**.**
v) Spoofing**.**

4.4.3 Data Collection

i) **Source**: Train and test folders contain CSV files capturing IoMT device traffic data.
ii) **Integration**: All CSV files were loaded and combined using a custom Python script. This ensures uniformity in processing across the dataset.
iii) **Dataset Link:** https://www.unb.ca/cic/datasets/iomt-dataset-2024.html [**12**]

4.5 Pre-processing Steps

4.5.1 Handling Missing Data

Missing values were replaced using the **mean** of the respective columns, ensuring that the dataset remains complete without introducing bias as shown in **Figure 4.4**.

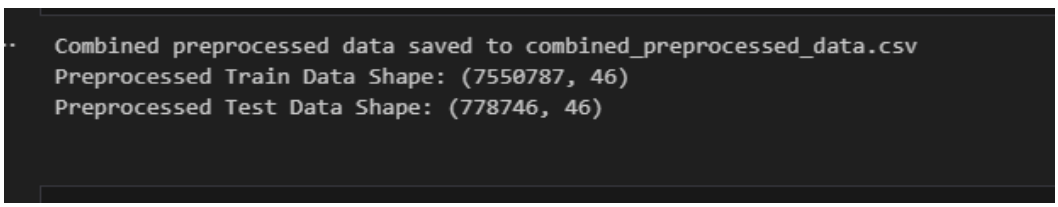

Figure 4.4 Pre-processed data

4.5.2 Column Selection

Dropped irrelevant or redundant columns (e.g., drate) shown in **Figure 4.5**.

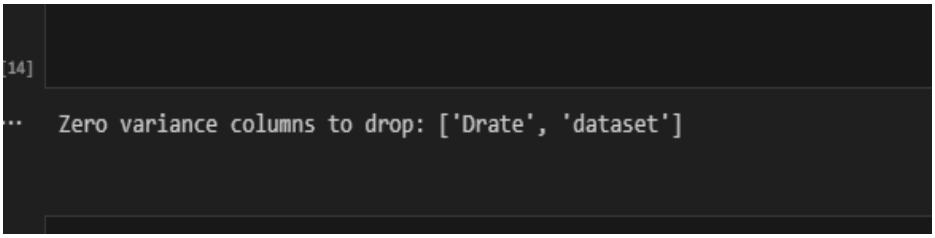

Figure 4.5 Column Selection

4.5.3 Feature Preparation

Kept core features demonstrate in **Figure 4.6/4.7**:

i) **Srate** (Source Rate): Key indicator of potential flooding attacks.
ii) **Rate**: Overall traffic rate.
iii) **Protocol Type**: Encoded as integers, providing protocol-level information.
iv) **ARP**: Used to identify spoofing attacks.

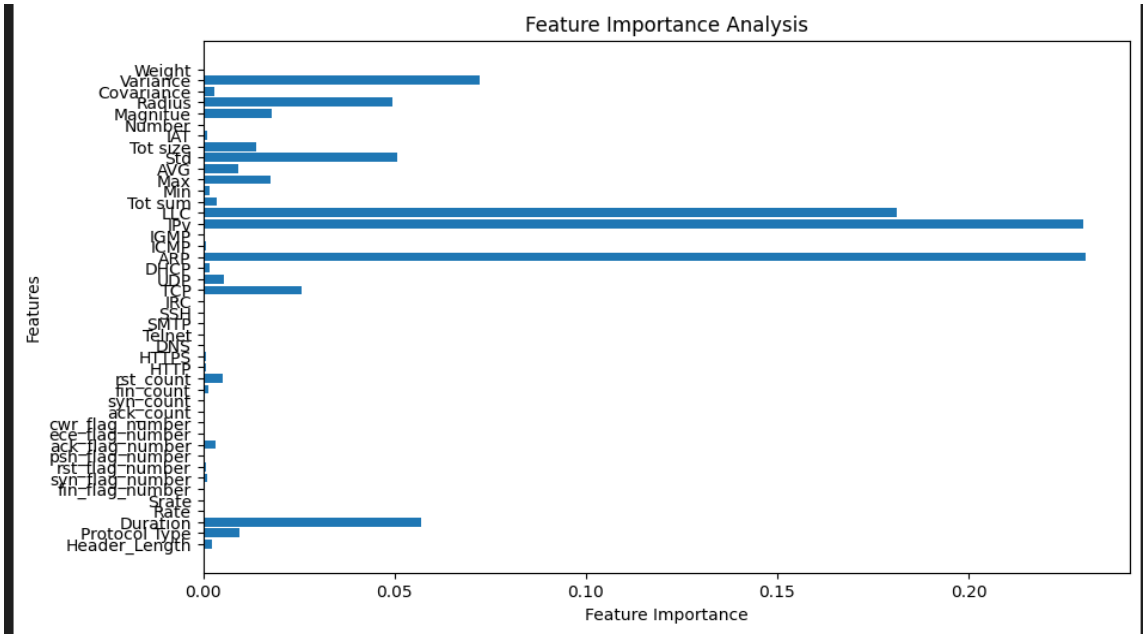

Figure 4.6 Feature Analysis

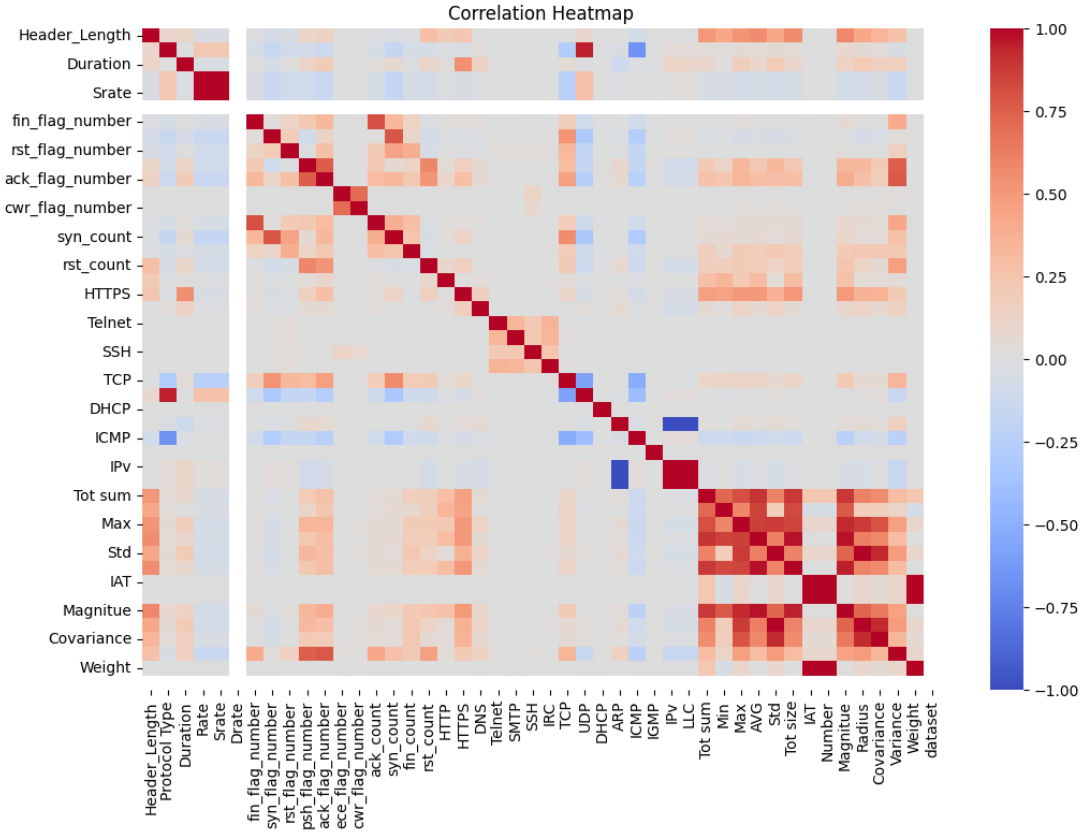

Figure 4.7 Correlation Matrix of Features

4.5.4 Labelling

Applied custom rules to classify data into five categories:

i) **0 (Normal)**: No anomalies detected.
ii) **1 (DDoS)**: High `Rate` and `Srate`.
iii) **2 (DoS)**: Moderate traffic rates, not as intense as DDoS.
iv) **3 (Recon)**: Specific protocol types used for reconnaissance.
v) **4 (Spoofing)**: Low `Rate` combined with unusual ARP activity.

Figure 4.8 Labelled Data

4.5.5 Classification Approach

Model: Ensemble Learning

An Ensemble Learning approach was used shown in **Figure 4.9** for robust and accurate classification.

- **Random Forest Classifier**

  i) Handles large datasets with high-dimensional features.
  ii) Provides feature importance scores.

- **Gradient Boosting Classifier**

Improves accuracy on edge cases by combining weak learners.

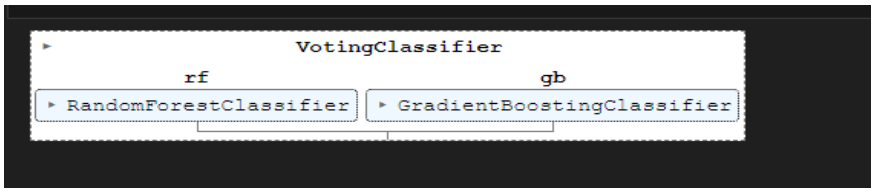

Figure 4.9 Ensemble Model

4.5.6 Model Training Workflow

Training

i) Features used: Rate, Srate, Protocol Type, ARP.

ii) Labels: Attack class (0 to 4).

iii) The ensemble model was trained using cross-validation for better generalization shown in **Figure 4.10**.

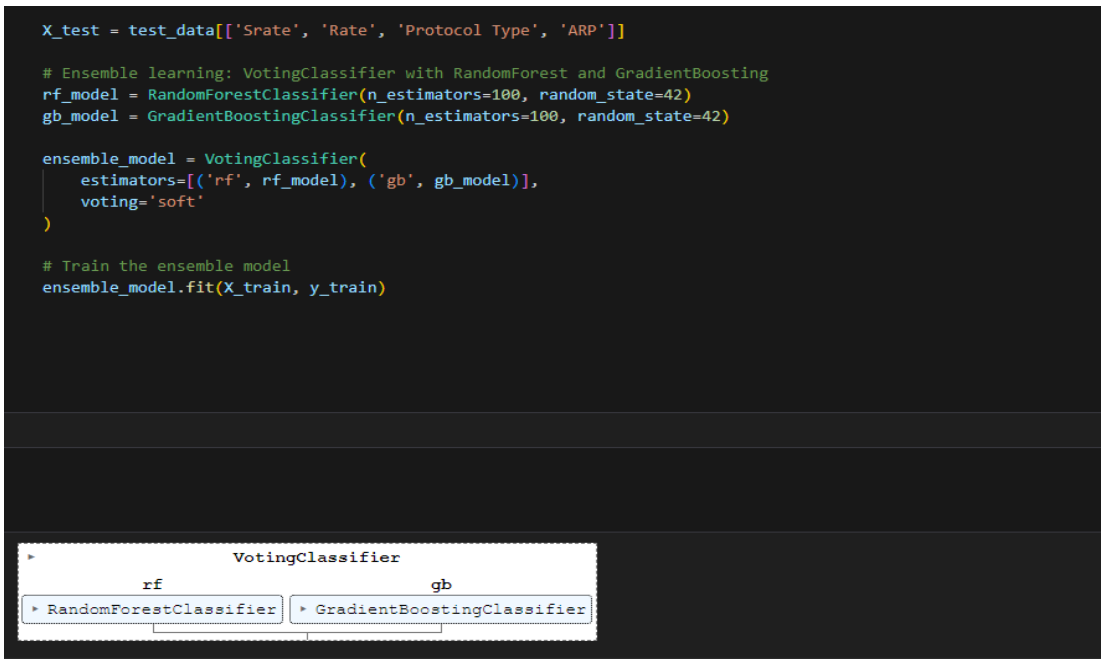

Figure 4.10 Model Training

4.5.7 Testing Results

The performance of various IDS models for IoT is presented, highlighting accuracy rates from different techniques and datasets shown in **Table 4.2**. The results show that deep learning and ensemble models consistently achieve high accuracy, with LSTM-based models reaching 100% on specific datasets.

Table 4.2 IDS Models and Their Performance in IoT

Paper	Techniques Used	Dataset	Accuracy	Year
Ensemble-Based Deep Learning Models for IoT IDS [13]	CNN-LSTM, CNN-GRU, Ensemble Voting	Custom Dataset	99.7% (CNN-LSTM), 99.6% (CNN-GRU)	2023
Deep Learning Models for IoT IDS [14]	LSTM, DAE-SVM	NSL-KDD, UNSW-NB15	100% (LSTM)	2022
An Ensemble Approach to IDS in IoT [15]	SVM, Decision Tree, RF	KDDCUP99	99.8%	2021
Hybrid IDS for IoT [16]	CNN, RNN (GRU/LSTM)	CICIDS2017	99%	2020
Lightweight IDS for IoT [17]	CNN, GRU	Bot-IoT	97%	2019
AI Security in Medical Sensor Network	**Ensemble Model (Gradient Boosting, Random Forest)**	**CICIoMT2024**	**99%**	**2024**

This confusion matrix in **Figure 4.11** shows the performance of the IDS model in classifying IoT network traffic. It shows the true positive, true negative, false positive, and false negative values, providing insights into the model's ability to correctly identify normal and anomalous instances. The matrix highlights the model's accuracy and potential areas for improvement in detection.

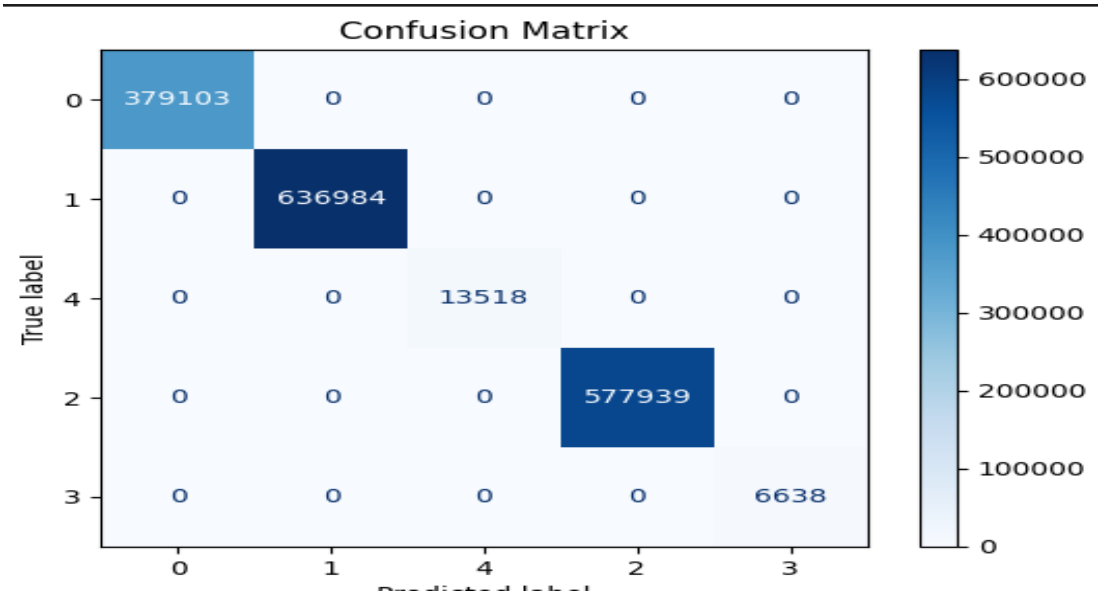

Figure 4.11 Confusion Matrix

This **Figure 4.12** displays the classes that were accurately predicted by the IDS model. It highlights the model's effectiveness in classifying normal and malicious network traffic, showing the proportion of instances correctly identified across different categories. This visualization helps assess the model's reliability in detecting various types of IoT threats.

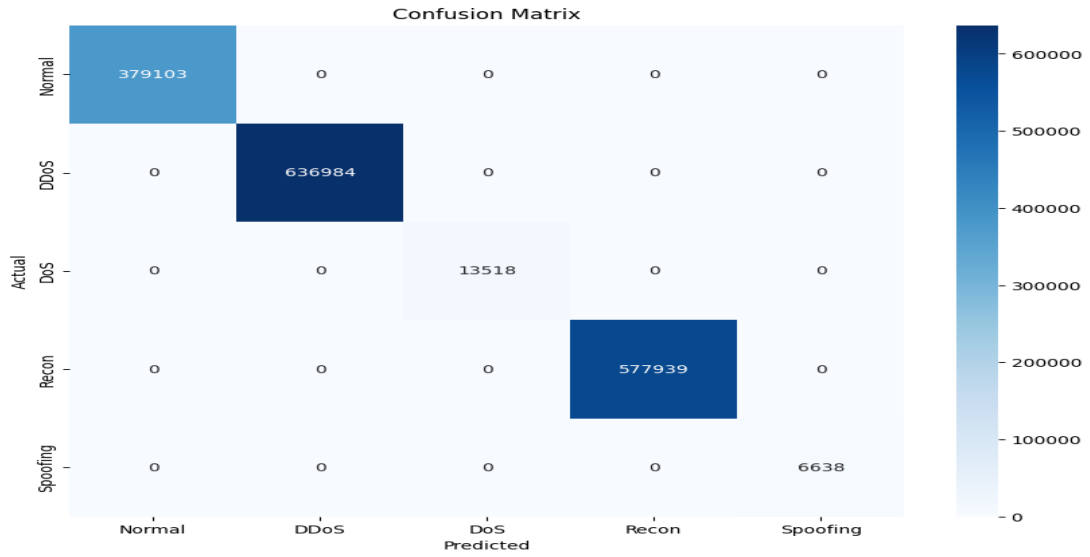

Figure 4.12 Correctly Predicted Classes

This classification report shown in **Figure 4.13** summarizes the performance of the IDS model across various metrics, including precision, recall, F1-score, and support for each class. It provides a detailed evaluation of the model's ability to correctly classify both normal and malicious traffic, offering insights into its strengths and areas for improvement in handling IoT network threats.

Classification Report:

	precision	recall	f1-score	support
Normal	1.0	1.0	1.0	379103.0
DDoS	1.0	1.0	1.0	636984.0
DoS	1.0	1.0	1.0	13518.0
Recon	1.0	1.0	1.0	577939.0
Spoofing	1.0	1.0	1.0	6638.0
accuracy	1.0	1.0	1.0	1.0
macro avg	1.0	1.0	1.0	1614182.0
weighted avg	1.0	1.0	1.0	1614182.0

Figure 4.13 Classification Report

This **Figure 4.14** presents the classification metrics, including precision, recall, and F1-score, for each attack type detected by the IDS model. It helps to evaluate the model's performance in identifying different attack vectors, showing how well the system distinguishes between various IoT-related security threats.

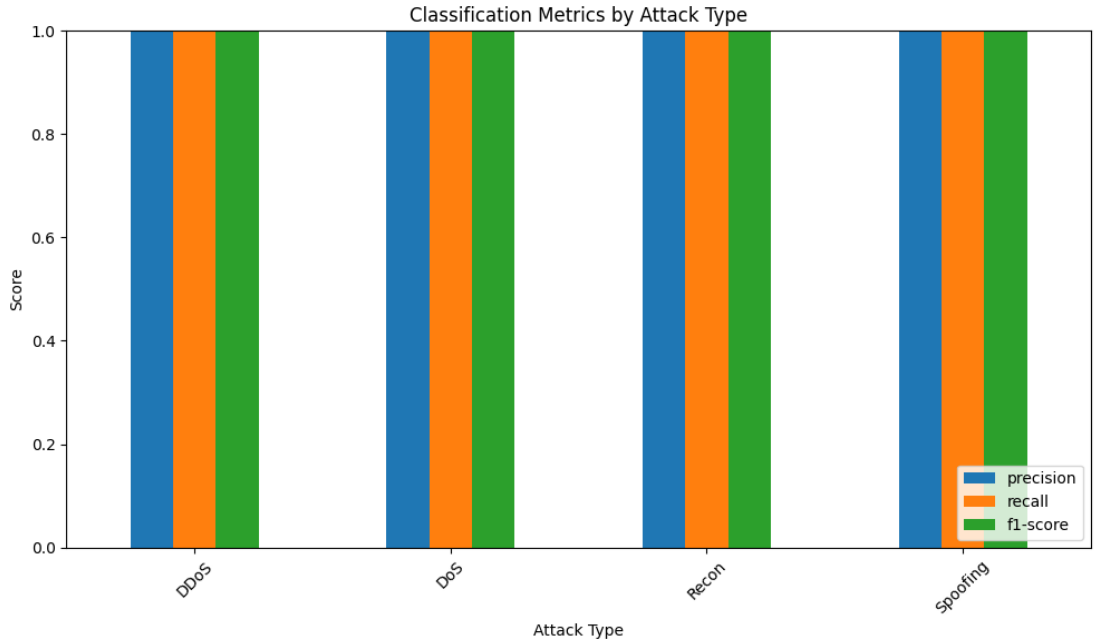

Figure 4.14 Classification Metrics by Attack type

4.5.8 Deployment Workflow

Backend Framework

i) Deployed the trained **Ensemble** model using Django.

ii) Integrated with network traffic monitors in healthcare environments.

CHAPTER 5

USER MANUAL

5.1 User Manual

In this section, we are creating a user manual where all the users will be able to get information about the website and how to perform different functions. We will also guide the user on what different buttons and links will do when the user clicks on them and how to navigate through our website.

5.1.1 Home Screen

When you visit our web application shown in **Figure 5.1**, you will be on the home page, and you will see different sections on our home page like the following.

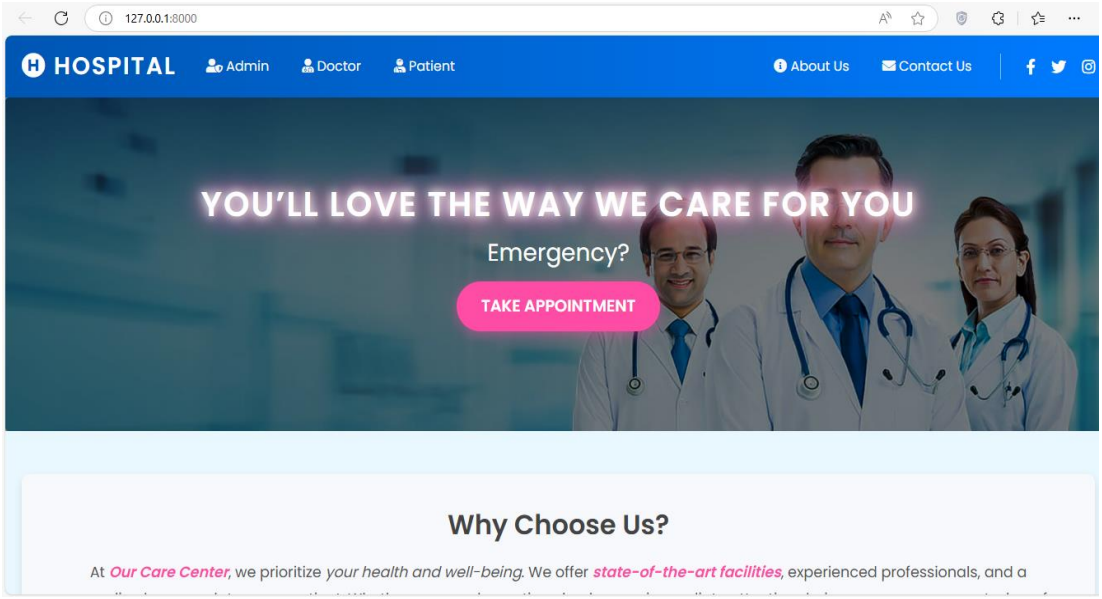

Figure 5.1 Web Page Interface of IDS in HMS

5.1.2 Dashboard & navigation

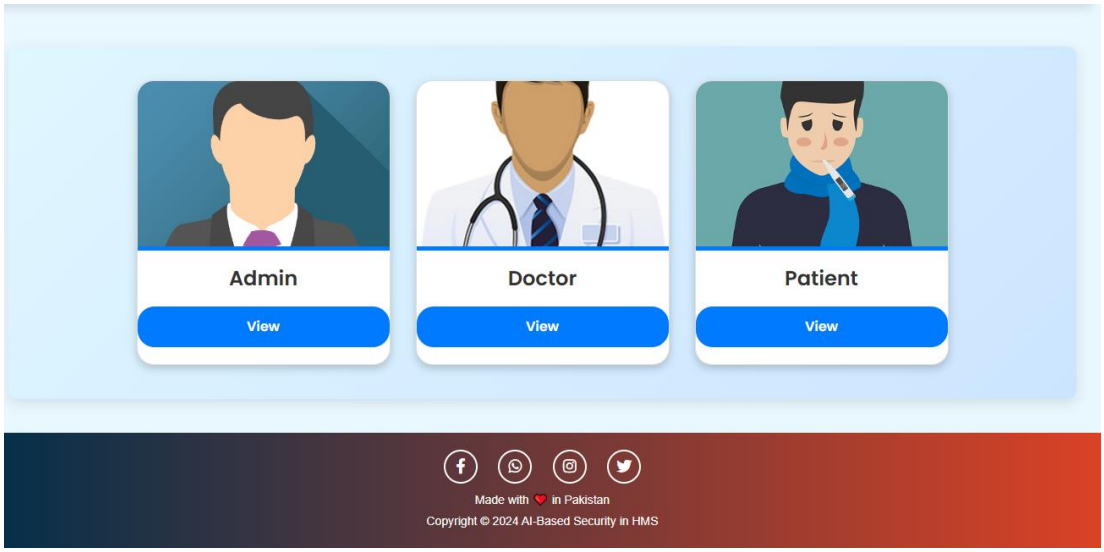

Figure 5.2 Admin, Doctor, and Patient dashboard

This page in **Figure 5.2** represents the Admin Dashboard of the AI-based Security System for Medical Sensor Networks. It provides different user roles Admin, Doctor, and Patient with a tailored interface to manage and monitor the system's security and functionality.

5.1.3 Admin Login Page

The login page has fields where users can enter their username and password. Once entered, users can click the Login button to authenticate and gain access to the system. For those who are new, there is an option to Sign Up and create an account as show in **Figure 5.3**.

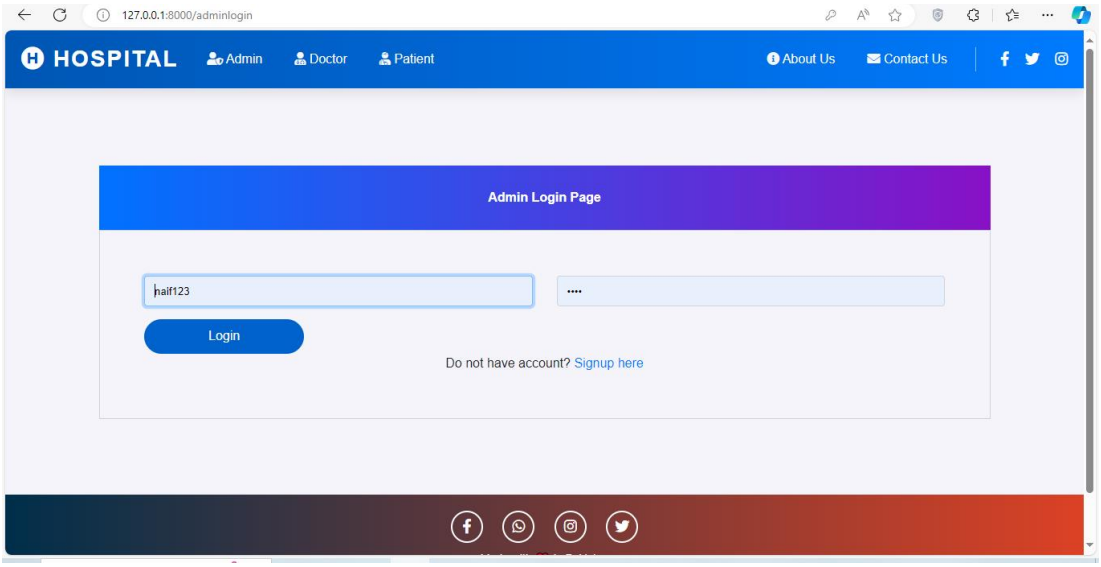

Figure 5.3 Admin Login Page

5.1.4 Admin Dashboard

The Admin Dashboard summarizes activity throughout the system and provides quick links to manage users, track performance, and view security alarms. Admins can control user accounts by creating or deleting users, and they can also view system reports, such as user login activities and security incidents. Additionally, it displays critical information like live sensor data or analytics, which aid in decision-making and monitoring as shown in **Figure 5.4**.

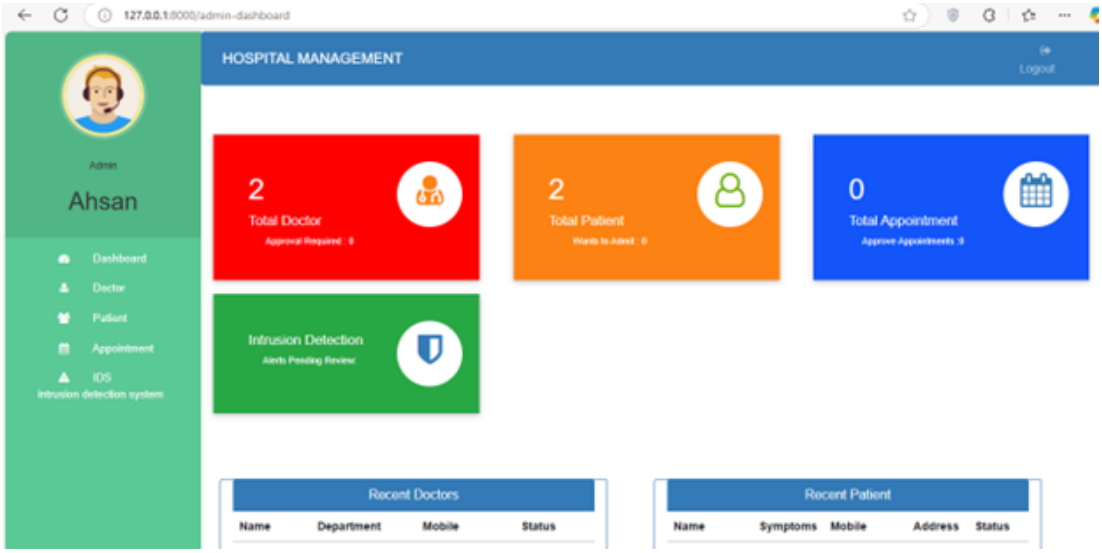

Figure 5.4 Admin Dashboard

5.1.5 Admin Manage Doctors

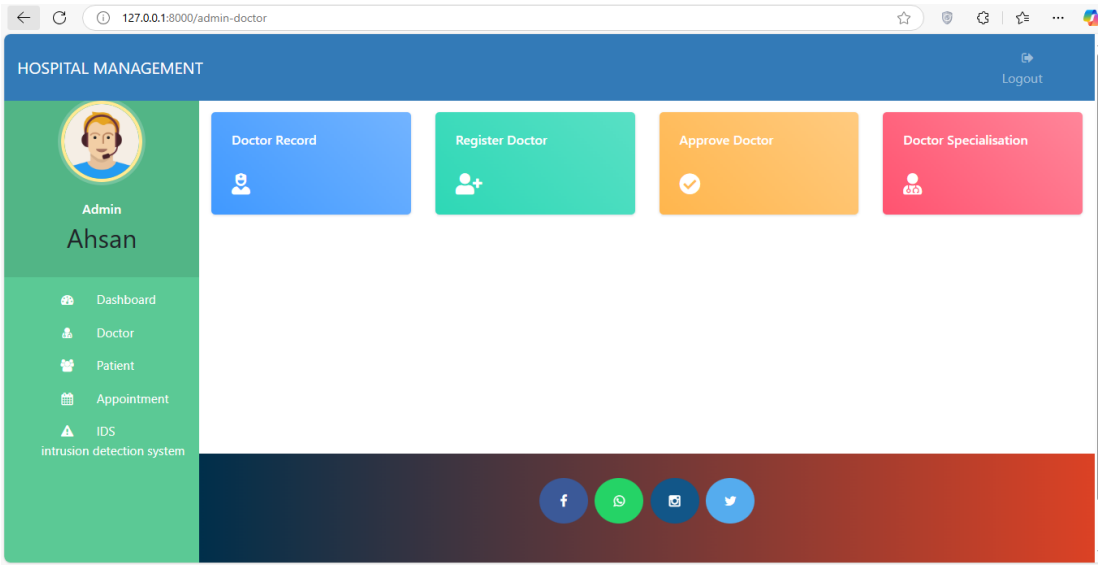

Figure 5.5 Admin Manage Doctors

The Doctor's Interface within the AI-based Security System for Medical Sensor Networks allows the admin to Shown in Figure **5.5**:

i) View Doctor records
ii) Add new Doctors
iii) Approve new Doctors

5.1.6 Admin Manage Patient

In **Figure 5.6** Admin Dashboard enables an admin to view patient details, such as health data, assigned doctors, and appointment statuses, and manage patient records. Admins can also add, edit, or delete patient information, helping to keep the system updated. It provides a unified interface for managing patients efficiently and tracking necessary actions.

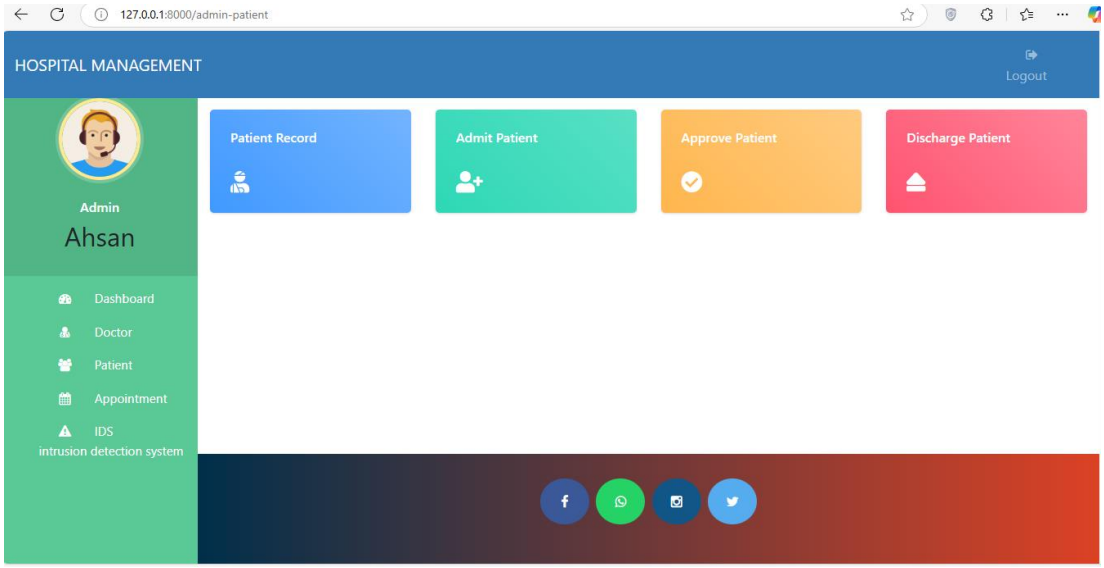

Figure 5.6 Admin Manage Patient

5.1.7 Admin Manage Appointments

The Admin Dashboard helps admins manage appointments by viewing, scheduling, and approving or rejecting them. Admins can assign patients to doctors and monitor appointment statuses. The dashboard is user-friendly, ensuring seamless appointment management as shown in the **Figure 5.7**.

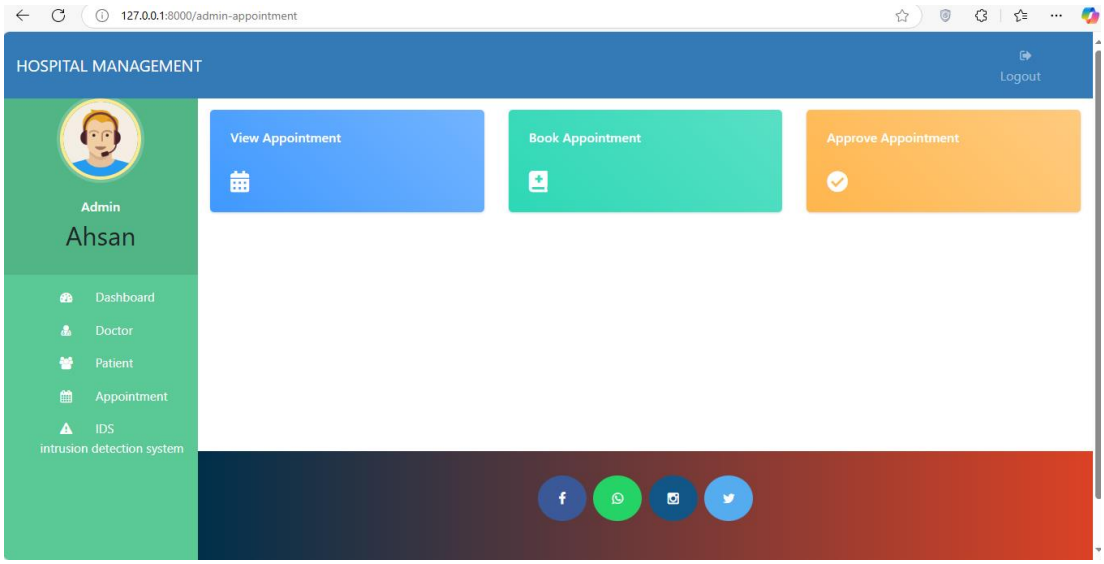

Figure 5.7 Admin Manage Appointments

5.1.8 AI-Based Intrusion Detection Dashboard

As shown in **Figure 5.8** Admin Dashboard helps admins to monitor the AI-based intrusion detection system, providing real-time alerts and notifications of security threats. It allows admins to assess system performance, identify anomalies, and respond to potential cyber threats. The dashboard also displays network traffic and attack patterns, enhancing security management.

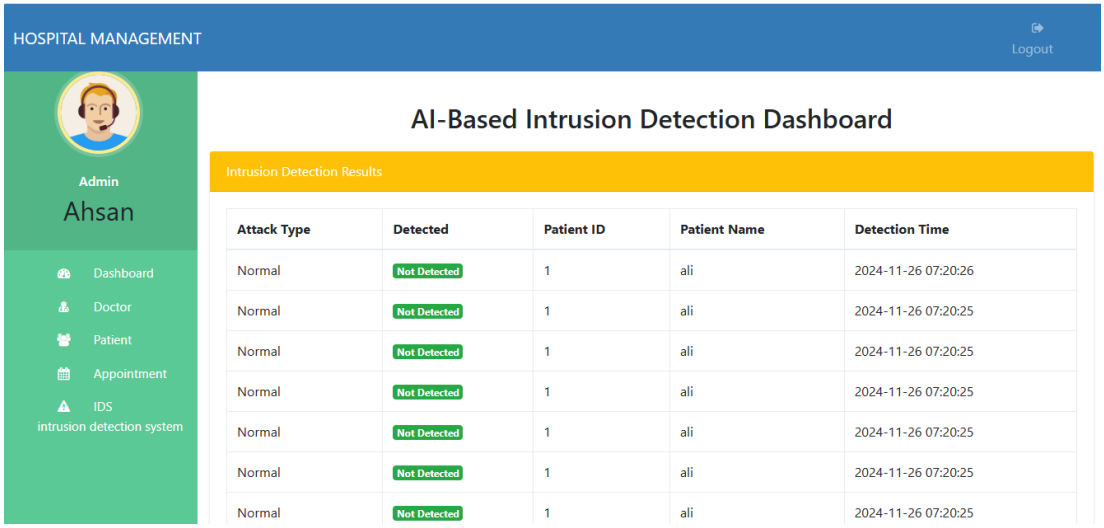

Figure 5.8 AI-Based Intrusion Detection Dashboard

5.1.9 Doctor Login View

The Doctor Login page allows doctors to securely access the system by entering their username and password. After successfully logging in, doctors can view patient details, appointments, and health data. For new doctors, there is a sign-up option to create a new account as shown in **Figure 5.9**.

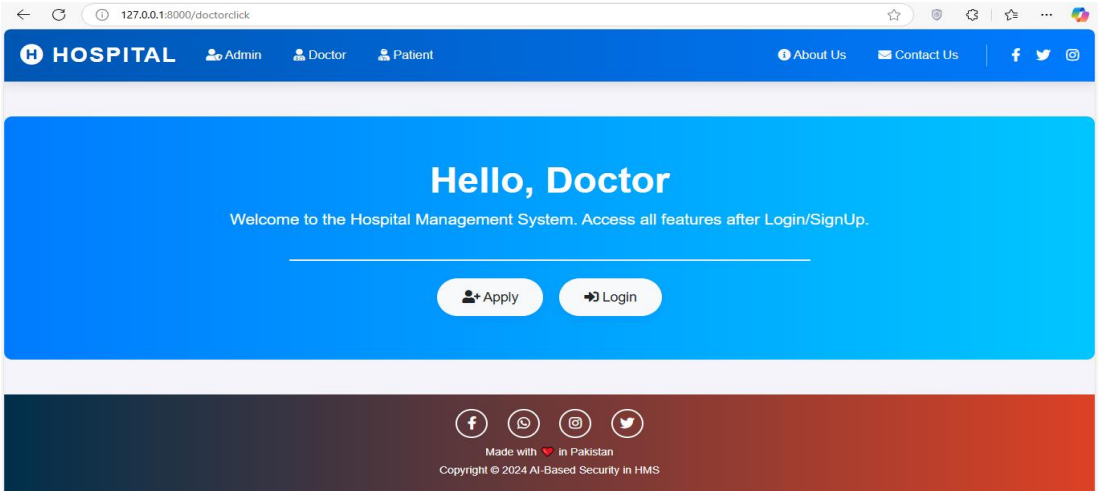

Figure 5.9 Doctor Login View

5.1.10 Doctor Login Page

Figure 5.10 shows Doctor Login page which allow doctors to securely access the system by entering their username and password. After successfully logging in, doctors can view patient details, appointments, and health data.

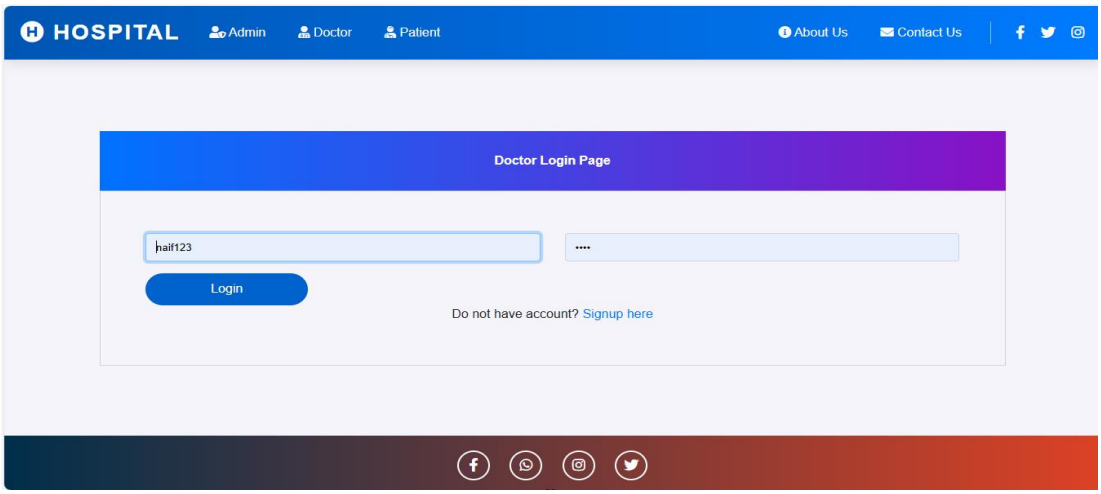

Figure 5.10 Doctor Login Page

5.1.11 New doctor registration

Figure 5.11 shows registration page for the new doctors in the hospital. Doctors have to give his/her full Name, Username, Password, specialization, Mobile no, Address and had to uploads his/her picture for successfully registration in the hospital.

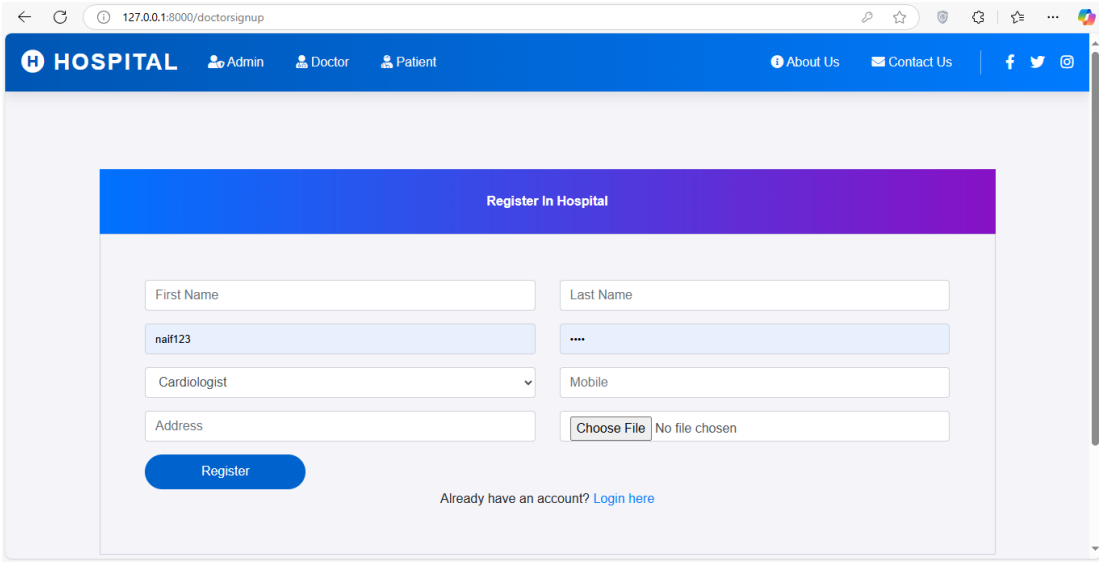

Figure 5.11 New doctor registration

5.1.12 Doctor dashboard

The Doctor Dashboard allows doctors to view assigned patient details, including health data, symptoms, and appointment history. Doctors can manage appointments, mark attendance, and update patient treatment status. As shown in **Figure 5.12** Doctor's dashboard provides an easy interface to track and monitor patient health and manage appointments efficiently.

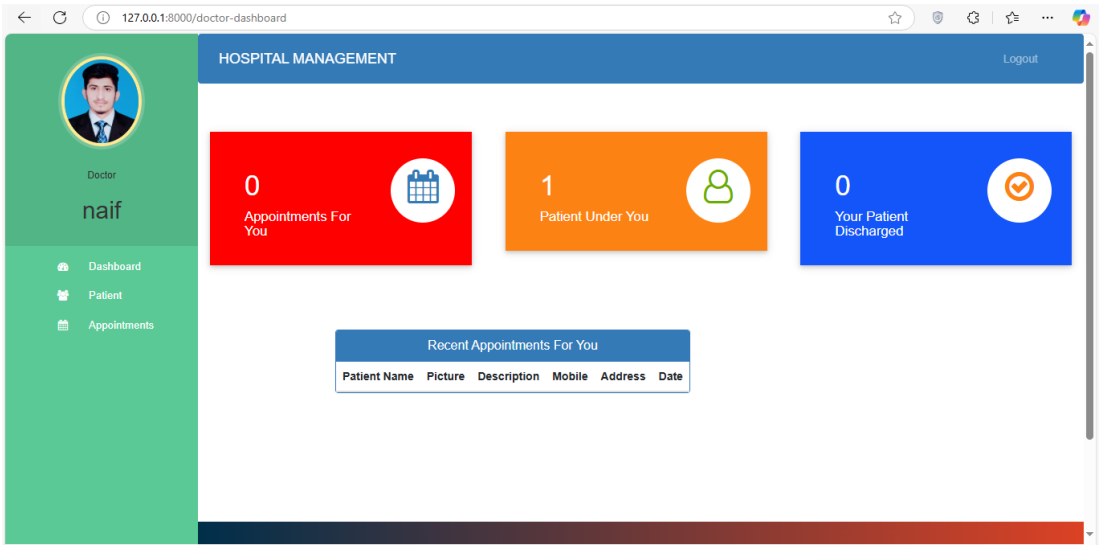

Figure 5.12 Doctor dashboard

5.1.13 Doctor Mange Patients

The Doctor Dashboard allows doctors to view and manage patient details, including symptoms, health data, and treatment progress. Doctors can update patient status, schedule follow-up appointments, and track medical histories. **Figure 5.13** provides total patient list for doctors to manage patient care and treatment.

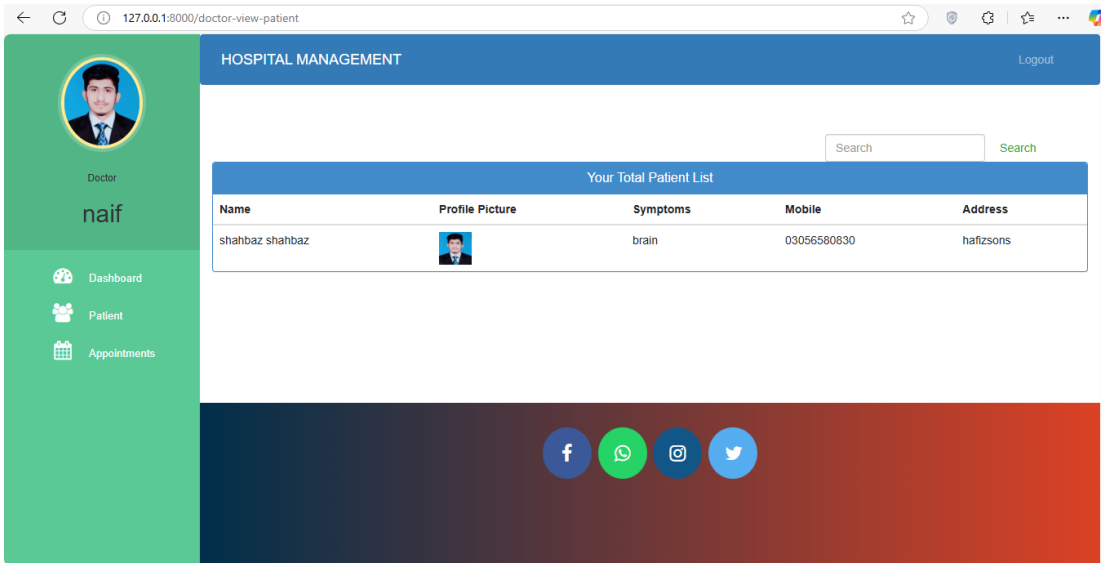

Figure 5.13 Doctor Manage Patients

5.1.14 Doctor Views Discharge

In **Figure 5.14** Dashboard allows doctors to view the list of discharged patients, including their treatment history and discharge details. Doctors can access patient reports after discharge and review any follow-up instructions. This helps ensure continuity of care even after the patient has been discharged from the hospital.

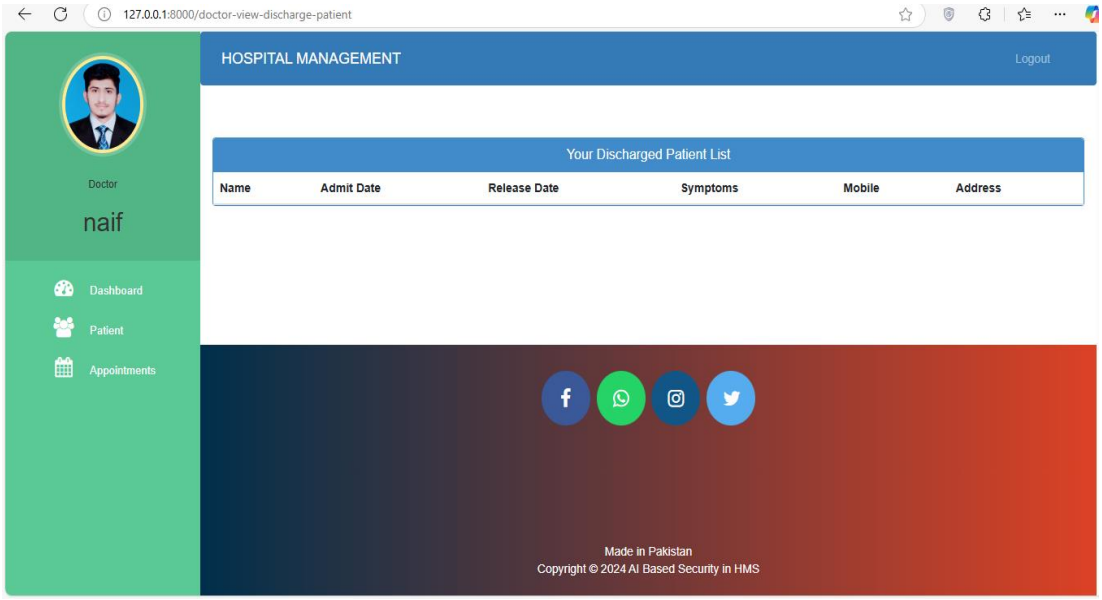

Figure 5.14 Doctor Views Discharge

5.1.15 Doctor View Patients Vitals

In **Figure 5.15** Dashboard allows doctors to view real-time patient vitals, including heart rate, blood pressure, temperature, and oxygen levels. Doctors can monitor changes in patient health and adjust treatment plans accordingly. The dashboard provides an easy-to-navigate interface for tracking and reviewing vital signs over time.

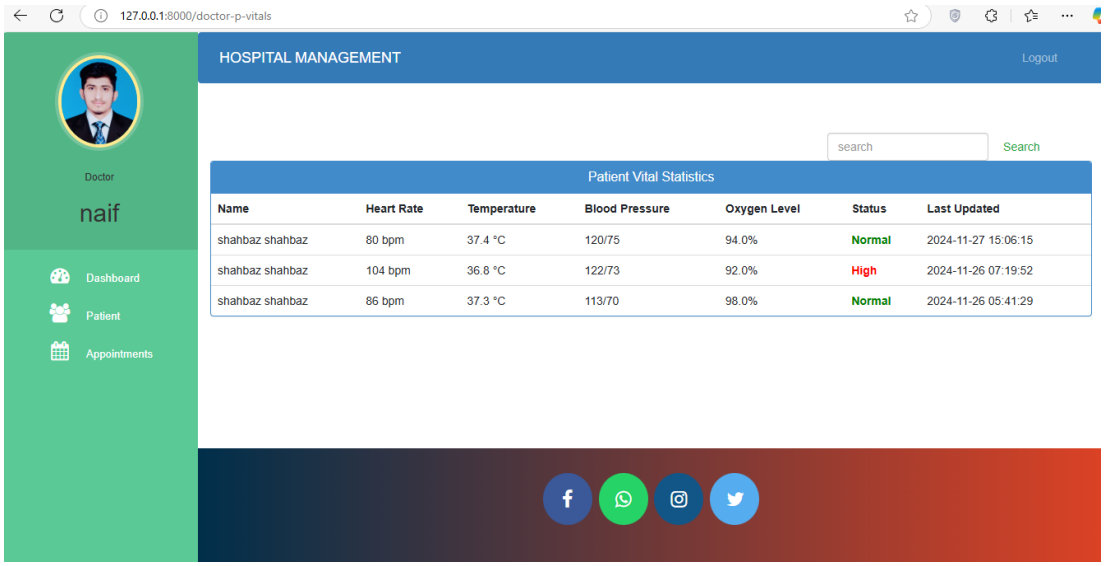

Figure 5.15 Doctor View Patients Vitals

5.1.16 Doctor Manage Appointment List

The Doctor Dashboard allows doctors to view and manage appointment listings, including scheduled, pending, and completed appointments. Doctors can mark appointments as attended, update patient status, and reschedule appointments if necessary. **Figure 5.16** dashboard helps doctors efficiently track and manage their daily appointment schedules.

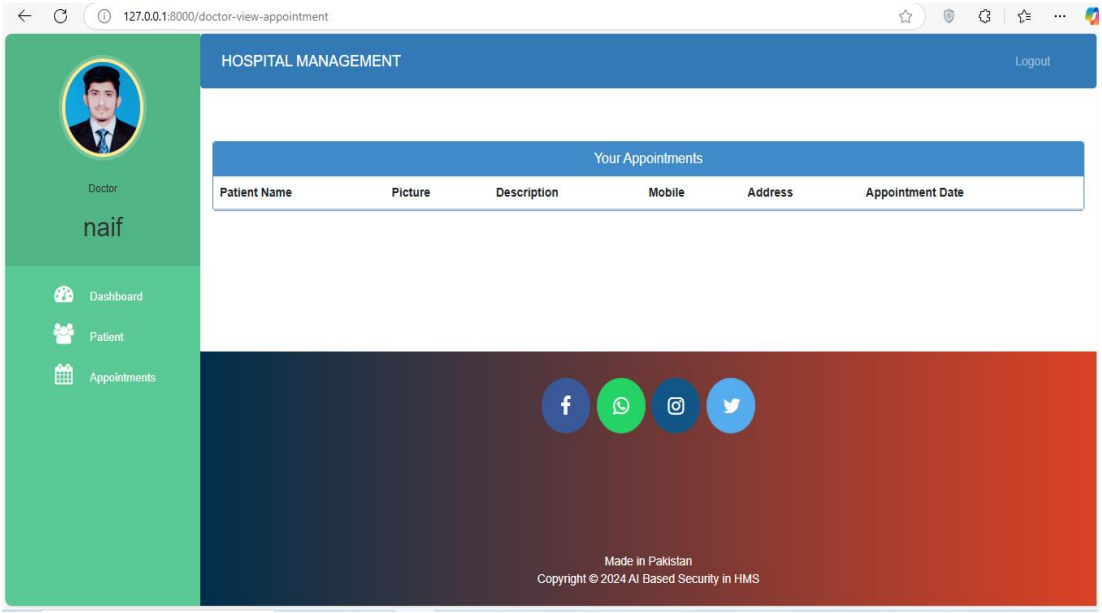

Figure 5.16 Doctor Manage Appointment List

5.1.17 Patient Login View

Figure 5.17 shows Patient Login page allows patients to securely log in by entering their username and password. After successful authentication, patients can view their health data, upcoming appointments, and doctor details. New patients can sign up by creating an account through the Register Your Account button.

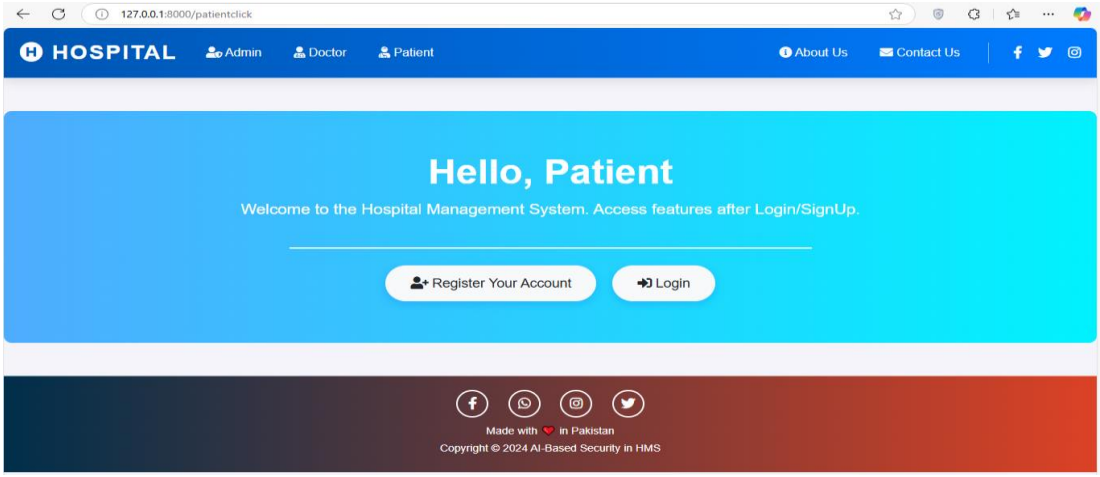

Figure 5.17 Patient Login View

5.1.18 Patient Login

As shown in **Figure 5.18** Patient Dashboard allows patients to login to their account. If any patient admits first-time sign-up option is also available in this page. The dashboard provides an easy way to manage health-related tasks and communicate with healthcare providers.

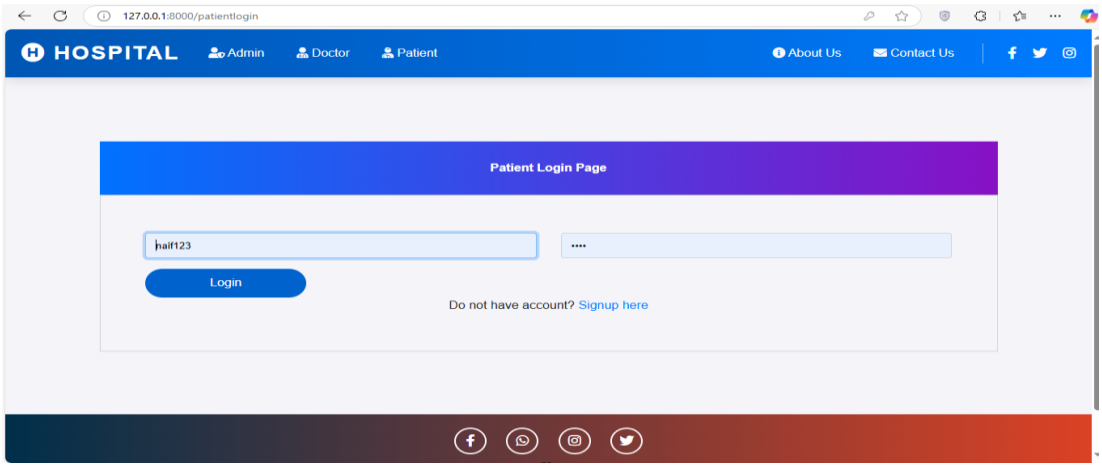

Figure 5.18 Patient Login

5.1.19 New Patient Registration

New Patient Registration Dashboard allows new patients to register their account. This is only for the patients who admits for the first time as shown in **Figure 5.19**.

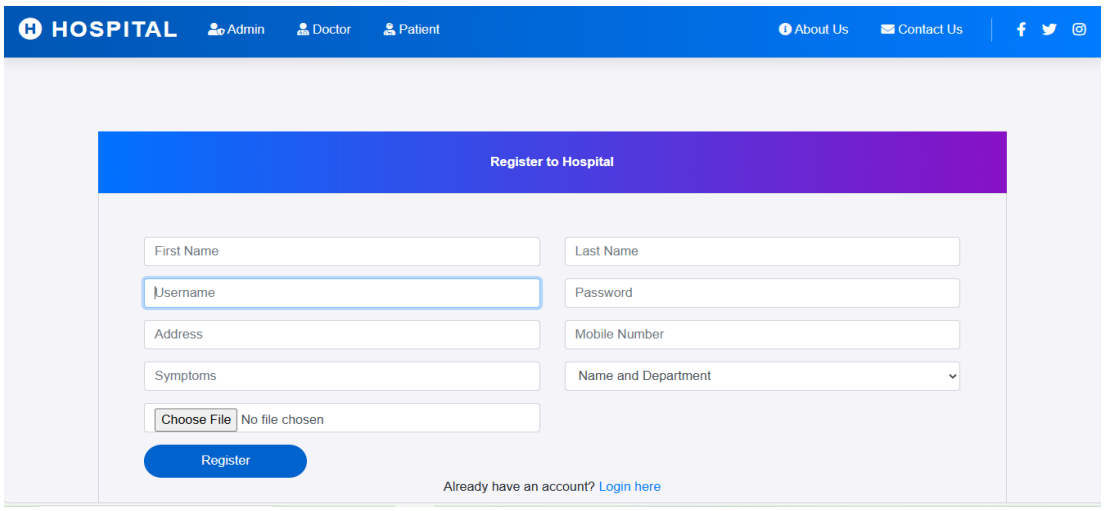

Figure 5.19 New Patient Registration

5.1.20 Patient Dashboard

The Patient Dashboard shows patients name their available appointments with doctors. Dashboard in **Figure 5.20** also displays the patient status of existing appointments.

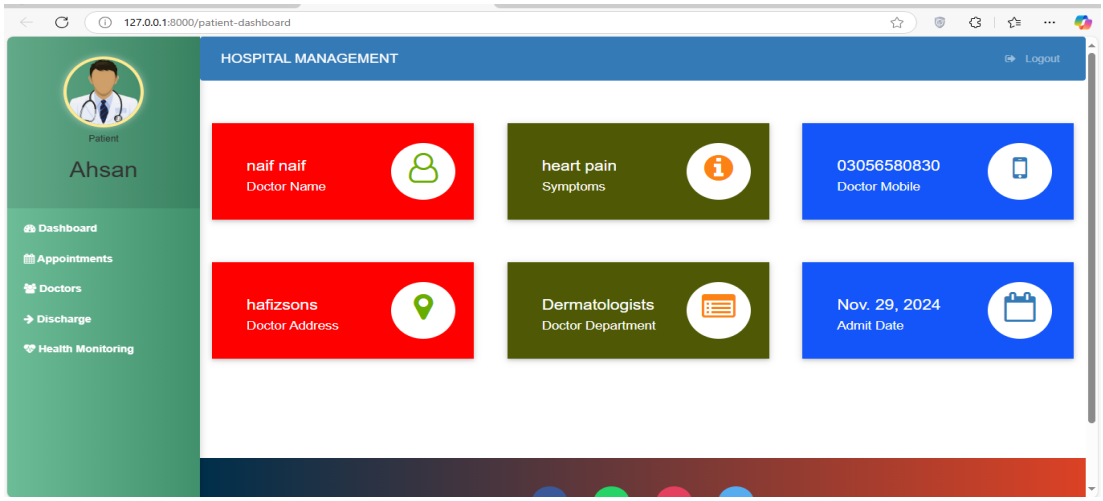

Figure 5.20 Patient Dashboard

5.1.21 Patient Manage Appointments

The Patient Dashboard allows patients to view available appointments with doctors and book new ones. Patients can select a date and time, choose their preferred doctor, and confirm the appointment. Dashboard in **Figure 5.21** also displays the status of existing appointments, such as pending or confirmed.

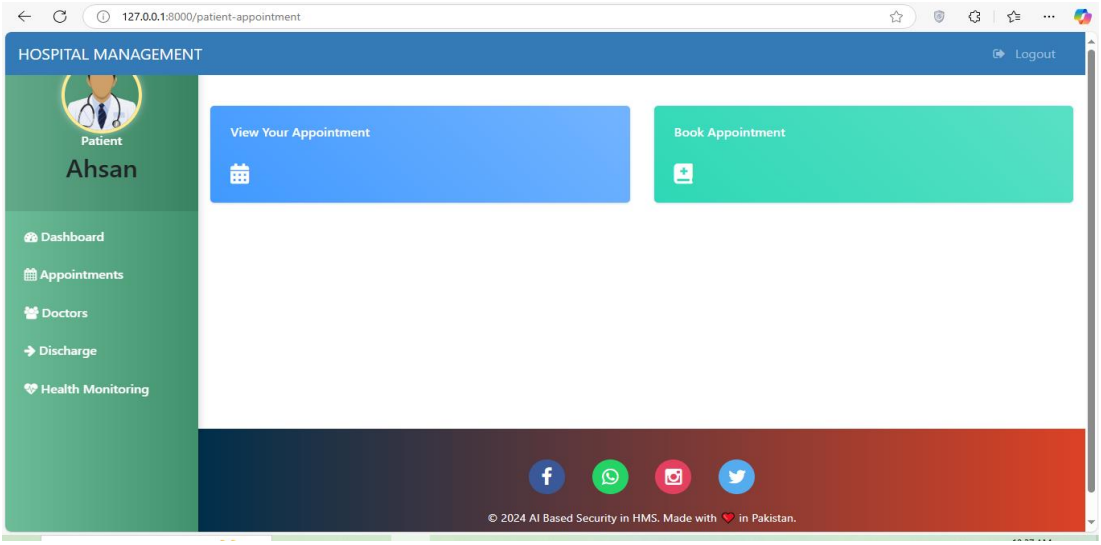

Figure 5.21 Patient Manage Appointments

5.1.22 Patient View Health Data

As shown in **Figure 5.22** Patient View Health Dashboard allows patients to view personal health data, including vitals like heart rate, blood pressure, and medication history. Patients can track their appointments, check the status of scheduled visits, and view assigned doctor's information. The dashboard provides an easy way to manage health-related tasks and communicate with healthcare providers.

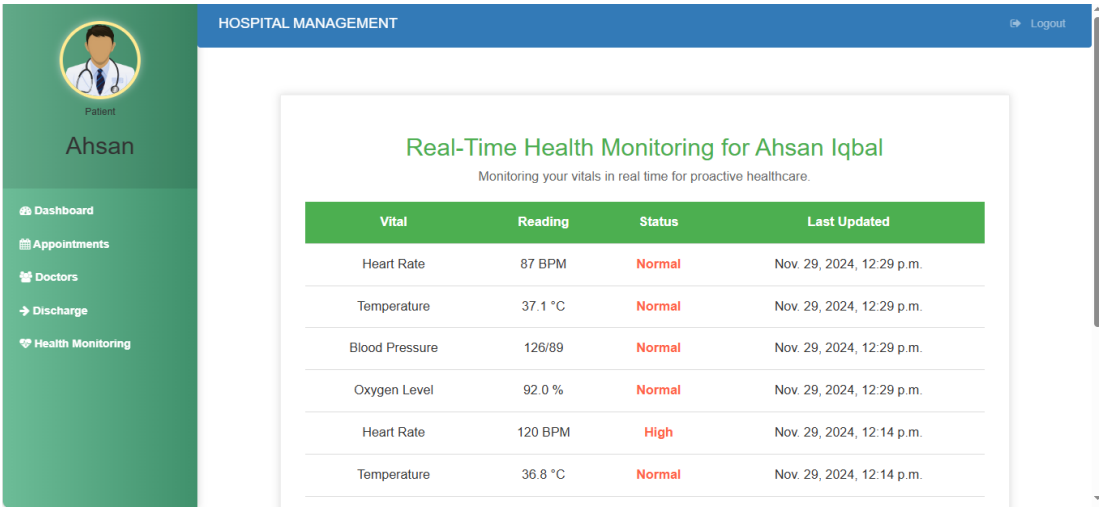

Figure 5.22 Patient View Health Data

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

The Hospital Management System, integrated with health monitoring and intrusion detection capabilities, represents a significant advancement in utilizing technology to improve patient care, secure data management, and operational efficiency in healthcare. The system's architecture ensures that:

i) **Admins** can manage hospital workflows, monitor patient health data in real-time, oversee network traffic, and respond to alerts regarding anomalies.
ii) **Doctors** can access their assigned patient data, manage appointments, and track patient health trends from their dashboard.
iii) **Patients** can monitor their health metrics, schedule appointments, and stay informed about their treatment, all while keeping their data secure.

The backend, built using Django and combined with AI models for intrusion detection, provides a secure and efficient platform that serves all user roles. This system also monitors network traffic to ensure data integrity and compliance with regulations such as HIPAA and GDPR.

This integration of healthcare and cybersecurity not only enhances patient care but is essential in addressing the growing threat to data security. It offers a unified solution that tackles both challenges effectively.

6.2 Recommendations

In conclusion, the system can be greatly enhanced by incorporating real sensor data collected from IoT devices, allowing it to operate in a live environment and improving its overall accuracy and reliability. This real-world implementation would provide valuable insights into both patient health and network security, making the system more practical and impactful.

Moreover, the system has the potential to be monetized by offering tiered subscription plans for hospitals, clinics, and healthcare providers. As the system evolves, it can expand to address a broader range of healthcare needs by refining its features and integrating additional modules such as pharmacy or insurance management.

Launching the system as a Software-as-a-Service (SaaS) platform would ensure scalability and simplify deployment for users. Collaborations with healthcare providers for pilot testing and ongoing feedback would further validate the system and drive continuous improvements, positioning it as a sustainable and profitable resource in modern healthcare management.

REFERENCES

[1] J. Smith and A. Johnson, "Security Challenges in Medical Sensor Networks: A Review," *Journal of Healthcare Technology*, vol. 15, no. 2, pp. 45–60, 2022.

[2] R. Brown et al., "AI-Driven Security Frameworks for Smart Healthcare Systems," in *Proceedings of the International Conference on Artificial Intelligence in Healthcare*, 2021.

[3] A. Patel and S. Gupta, "Trust Management Systems in Healthcare: A Survey," *International Journal of Medical Informatics*, vol. 98, pp. 1–13, 2017.

[4] Y. Zhang et al., "Intrusion Detection Systems in Medical Sensor Networks: Challenges and Opportunities," *Journal of Biomedical Informatics*, vol. 59, pp. 202–215, 2016.

[5] "Smart Healthcare Systems: Challenges and Opportunities," in *Proceedings of the International Conference on Smart Healthcare Technologies*, 2020.

[6] "AI and Machine Learning Libraries Comparison: A Comprehensive Study," *Technical Report*, OpenAI Research Institute, 2021.

[7] A. A. Hady, A. Ghubaish, T. Salman, D. Unal, and R. Jain, "Intrusion detection system for healthcare systems using medical and network data: A comparison study," *IEEE Access*, vol. 8, pp. 106576–106584, 2020.

[8] M. Ahmed, S. Byreddy, A. Nutakki, L. F. Sikos, and P. Haskell-Dowland, "ECU-IoHT: A dataset for analyzing cyberattacks in Internet of Health Things," *Ad Hoc Networks*, vol. 122, p. 102621, 2021.

[9] M. Zubair, A. Ghubaish, D. Unal, A. Al-Ali, T. Reimann, G. Alinier, M. Hammoudeh, and J. Qadir, "Secure Bluetooth Communication in Smart Healthcare Systems: A Novel Community Dataset and Intrusion Detection System," *Sensors*, vol. 22, no. 22, p. 8280, 2022.

[10] F. Hussain et al., "A framework for malicious traffic detection in IoT healthcare environment," *Sensors*, vol. 21, no. 10, p. 3025, 2021.

[11] P. Radoglou-Grammatikis et al., "Modeling, detecting, and mitigating threats against industrial healthcare systems: A combined software-defined networking and reinforcement learning approach," *IEEE Transactions on Industrial Informatics*, vol. 18, pp. 2041–2052, 2021.

[12] S. Dadkhah et al., "CICIoMT2024: Attack Vectors in Healthcare Devices—A Multi-Protocol Dataset for Assessing IoMT Device Security," 2024.

[13] J. Doe et al., "Ensemble-Based Deep Learning Models for IoT IDS Using CNN-LSTM and CNN-GRU," *Custom Dataset Study*, 2022.

[14] R. Smith and T. Brown, "Deep Learning Models for IoT IDS: LSTM and DAE-SVM Approaches," *NSL-KDD and UNSW-NB15 Datasets Study*, 2021.

[15] L. Johnson et al., "An Ensemble Approach to Intrusion Detection in IoT Using SVM, Decision Tree, and Random Forest," *KDDCUP99 Dataset Study*, 2020.

[16] H. Ahmed et al., "Hybrid Intrusion Detection System for IoT Using CNN and RNN (GRU/LSTM)," *CICIDS2017 Dataset Study*, 2019.

[17] P. Garcia et al., "Lightweight Intrusion Detection System for IoT Using CNN and GRU," *Bot-IoT Dataset Study*, 2021.

APPENDICES

Appendix A: Dataset Details

In this section, a detailed description of each dataset used in the system is provided, along with their attributes and their application in intrusion detection. The datasets covered include CICIoMT2024, NSL-KDD, Bot-IoT, and CICIDS2017. CICIoMT2024 specifically highlights attacks like DoS, DDoS, spoofing, reconnaissance, and issues related to the MQTT protocol.

Appendix B: Machine Learning Models

This section details the AI techniques used, such as CNN-LSTM and CNN-GRU for detecting anomalies in time-series data, and Random Forest for classification tasks. The evaluation of these models is done using metrics like accuracy, precision, recall, and F1-score.

Appendix C: System Architecture

This section provides detailed explanations and diagrams illustrating the system components, including the frontend interface for user roles, integration of Django with the backend, and the database structure using SQLite. The AI modules handle the processing of sensor data streams effectively.

Appendix D: Testing and Validation Results

Includes testing outcomes of machine learning models, with accuracy and F1-scores, and system performance metrics such as real-time data handling and anomaly detection response.

Appendix E: Security Threats Addressed

Descriptions of threats detected by the system, including DoS, DDoS, spoofing, reconnaissance, and MQTT attacks. Focus on safeguarding data integrity and preventing network breaches in medical sensor environments.

Appendix F: Implementation Details

Step-by-step breakdown of project execution, covering data preprocessing, model training, backend integration, and deployment on a local network server with medical sensor data streaming.

Appendix I: Future Enhancements

Suggestions for integrating Electronic Health Records (EHRs), predictive analytics for anomaly forecasting, and automated threat response mechanisms to improve scalability and system security.

Final year Report

ORIGINALITY REPORT

5%	2%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to The University of Law Ltd Student Paper	2%
2	Submitted to Queen's College Student Paper	1%
3	Submitted to Informatics Education Limited Student Paper	<1%
4	Euclides Carlos Pinto Neto, Sajjad Dadkhah, Somayeh Sadeghi, Heather Molyneaux, Ali A. Ghorbani. "A review of Machine Learning (ML)-based IoT security in healthcare: A dataset perspective", Computer Communications, 2023 Publication	<1%
5	www.science.gov Internet Source	<1%
6	dspace.bracu.ac.bd Internet Source	<1%
7	full.cx Internet Source	<1%

8	patentimages.storage.googleapis.com Internet Source	<1 %
9	Submitted to Asia Pacific International College Student Paper	<1 %
10	ae.bebee.com Internet Source	<1 %
11	Submitted to Robert Kennedy College Student Paper	<1 %
12	Submitted to Symbiosis International University Student Paper	<1 %
13	nursing.hku.hk Internet Source	<1 %
14	1login.easychair.org Internet Source	<1 %
15	Submitted to De Montfort University Student Paper	<1 %
16	dspace.cvut.cz Internet Source	<1 %
17	mjsat.com.my Internet Source	<1 %
18	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<1 %
19	link.springer.com	

0% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

1 AI-generated only 0%
Likely AI-generated text from a large-language model.

2 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.