PRESIDENCY UNIVERSITY

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

Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

AUTOMATIC HEALTH RECORD MONITORING SYSTEM

A PROJECT REPORT

Submitted by

KUSHAL V- 20221ISE0018

DARSHAN R H- 20221ISE0036

SRUJAN H G- 20221ISE0026

Under the guidance of,

Dr. SWATI SHARMA

Professor

BACHELOR OF TECHNOLOGY

IN

INFORMATION SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE AND ROBOTICS)

PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2025

PRESIDENCY UNIVERSITY

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

Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this report "**AUTOMATIC HEALTH RECORD MONITORING SYSTEM**" is a bonafide work of "KUSHAL V (20221ISE0018), DARSHAN R H(20221ISE0036) SRUJAN H G(20221ISE0026)," who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in INFORMATION SCIENCE AND ENGINEERING (AI AND ROBOTICS) during 2025-26.

Dr. Swati Sharma
Project Guide
PSCS
Presidency University

Ms. Suma N G
Program Project Coordinator
PSCS
Presidency University

2/12/25

Dr Sampath A K
Dr. Geetha A
School Project Coordinators
PSCS
Presidency University

Dr. Zafar Ali Khan
Head of the Department
PSCS
Presidency University

Dr. Shakkeera L
Associate Dean
PSCS
Presidency University

N.

Dr. Duraipandian N
Dean
PSCS & PSIS
Presidency University

Name and Signature of the Examiners

1) V. Channu 2/12/25

2) Anbu 2/12/25

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We the students of final year B.Tech in INFORMATION SCIENCE ENGINEERING ARTIFICIAL INTELLIGENCE AND ROBOTICS at Presidency University, Bengaluru, named KUSHAL V, DARSHAN R H, SRUJAN H G, hereby declare that the project work titled **"AUTOMATIC HEALTH RECORD MONITORING SYSTEM"** has been independently carried out by us and submitted in partial fulfillment for the award of the degree of B.Tech in INFORMATION SCIENCE AND ENGINEERING (AI AND ROBOTICS) during the academic year of 2025-26. Further, the matter embodied in the project has not been submitted previously by anybody for the award of any Degree to any other institution.

KUSHAL V	USN: 20221ISE0018	Kushal.v
DARSHAN R H	USN: 20221ISE0036	
SRUJAN H G	USN: 20221ISE0026	Srujan HG

PLACE: BENGALURU

DATE: 01-December 2025

ACKNOWLEDGEMENT

For completing this project work, we have received the support and guidance from many people whom I would like to mention with a deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Vice Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in the completion of the project.

I would like to sincerely thank my internal guide, **Dr. Swati Sharma, Professor,** Presidency School of Computer Science and Engineering, Presidency University, for her moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to to **Dr. Zafar Ali Khan, Professor, Head of the Department, Presidency School of Computer Science and Engineering,** Presidency University, for his mentorship and encouragement.

We express our cordial thanks to **Dr. Duraipandian N**, Dean PSCS & PSIS, **Dr. Shakkeera L**, Associate Dean, Presidency School of Computer Science and Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Dr Sampath A K, Dr Geetha A., PSCS** school Project Coordinators, **Ms Suma N G, Program Project Coordinator**, Presidency School of Computer Science and Engineering, for facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to the Teaching and Non-Teaching staff of Presidency School of Computer Science and Engineering, and also staff from other departments who have extended their valuable help and cooperation.

KUSHAL V
DARSHAN R H
SRUJAN H G

Abstract

The growing incidence of chronic health diseases and little to do with access to organised healthcare systems, especially in the rural and suburban areas, has been identified as a pressing need to have effective and trusted medical data management solutions. In most of them, poor healthcare facilities and the absence of standardised record-keeping practices that are standardized have led to major obstacles to constant and quality medical services. Patients would not keep the right records of past diagnoses, prescriptions and lab results, thus causing them to take up medical tests and procedures unnecessarily and slow the medical care that is given to them and burden the already strained family budget. To address these problems, the proposed project presents an Automatic Health Monitoring System that will solve the issue through digitising and centralising patient health records on a secure and scalable platform. The system has both web-based and mobile interfaces that are easy to use, including those who are not that digitally literate. It has a cloud-synced, offline-enabled design in order to provide unlimited connectivity even in locations where the internet is sporadic. The main features are secure user authentication, 24/7 vital signs monitoring using IoT sensors, intelligent booking, customised health reminders, and automated medical reports.

Notably, the system incorporates progressive AI-driven capabilities that automatically create summaries of patients in brief and dynamic risk analysis charts. These are artificial intelligence-generated summaries that condense complicated medical information into clear and actionable reports, enabling a faster process of clinical decision-making. The risk former app calculates the severity of the disease and visually represents the health indicators with easy-to-use graphs that allow healthcare providers to prioritise the most urgent initiatives and make early interventions, and thus improve patient safety and clinical outcomes. The prototype testing shows greater healthcare efficiency, less diagnostic redundancy, and access to greater populations, especially those underserved in rural areas. This system is an important step to a smart healthcare ecosystem, and in the future, it is intended that the system should be enhanced with predictive analytics and telemedicine services, again further developing equitable and technology-induced public health infrastructure.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Background

The low levels of digital literacy and maintenance of medical records in rural and suburban areas have made access to effective healthcare highly restricted in these areas. The diagnosis reports are usually not recorded, and this further creates unnecessary medical checkups and higher healthcare expenses. Restrictions in infrastructure systems, low frequency of visiting a hospital, and ineffective record-keeping only exacerbate the condition of susceptible groups like the ageing population. The lack of smooth access to the history of patients makes interventions become more time-consuming. The combination of AI-powered features, e.g., automatically created patient summaries and dynamic risk analysis, can improve the process of healthcare delivery by enabling straight and practical knowledge out of complex medical information, facilitating quicker diagnosis and improved prioritisation of care.

1.2 Statistics

The rural Primary Health Centres (PHCs) in India have a population per PHC population of about 33,800 which is way above the recommended population of 30,000. Only 39 percent of the rural population practicing this survey indicated diagnostics that are within a commutable radius, and almost 73 percent do not access quality and affordable health services. Approximately 90 percent of the rural population does not seek regular checkups unless specifically referred to by a physician and many of them have to cover as long distance as 100 km to seek medical treatment. Cases of diagnostic redundancy still exist as a result of blank central health records, and access to subsidized medicines is only at 12.2 percent in rural households. This disjointed system results in high-maternal and infant mortality levels, malnutrition, and low vaccine coverage levels. The key lie in overcoming these barriers with the introduction of AI-based risk analysis and automated report generation, which provides scalable solutions, which limit the repetition of tests, mark at-risk patients with the graphical display of their severity, and enhance timely clinical treatment.

Each Primary Health Centre in rural India has around 33,800 people on average as compared to the normal standards of 30,000. Rural respondents can only access a diagnostic facility within commuting distance by only 39% and 73% do not have quality and affordable healthcare. As many as 90 per cent will not undergo routine checkups until asked to do so; some of them will go as far as travel a distance of 100 km to get treatment.

There are only 12.2% rural houses that have access to subsidised medications, as 82% of households prefer to use diagnostic centres and get less erroneous and quicker results, and the availability of quality diagnostics within remote areas is limited.

1.3 Prior Existing Technologies

Health monitoring platforms, mobile health applications, and systems of remote patient management have enhanced access to medical data in the last several years with the support of IoT. The examples include IoT-based healthcare monitoring (Hamim et al., 2022), mobile health services on chronic conditions (Kim et al., 2015), and similar solutions presented positive effects. Adoption barriers still exist; however, these technologies are excessively complex or inaccessible to rural and low-literacy populations, do not have support in the local language, privacy settings, and are not really offline/cloud integrated. Existing digital health, unlike the previous ones (telemedicine, mobile clinics, AI-driven diagnostics), can serve as bridges to the gaps, but extensive infrastructure and labour shortages are still present, and the development of a simple and easy-to-use health monitoring platform that would suit these communities is encouraged. Automatic Health monitoring system fills these gaps by incorporating the ability of AI-enabled tools to provide automated report summarisation of patients and risk assessment to provide easy-to-understand health information to both practitioners and patients. The features of AI minimise the high level of technical skills, allow making decisions in real time and enhance the outcome in healthcare in resource-restricted environments.

Remote patient monitoring and chronic condition monitoring based on IoT (Hamim et al., 2022).

M-based health interventions to assist low-literacy people to manage chronic diseases (Kim et al., 2015).

Healthcare Internet of Things Low-power network solution to be used in underserved areas (Mdhaffar et al., 2017).

Although significant, major gaps are: the availability of existing technologies does not support access, and it requires a large amount of technical expertise; modern technologies do not support local languages and offline connectivity. The latter make it challenging to adopt in rural conditions, which is why it is necessary to have a simple, accessible, and safe health monitoring solution that is user-friendly, considering the resource constraints among people and the age of the target population.

1.4 Proposed Approach

The main objective is to come up with an Automatic Health Monitoring System that would digitize patient health records and make them securely available to patients, healthcare providers, and caregivers in any device. The system will be supposed to enhance better centralization of medical data, less redundancy in diagnosis, and also greater efficiency of healthcare, particularly to underserved rural and suburban populations. One of the notable characteristics of the system is the addition of AI algorithms that automatically produce succinct patient health summaries and calculate the level of risk assessed in the form of simple graphs to understand. These capabilities powered by AI are designed to help clinicians make fast, informed decisions and to proactively handle the care of patients.

1.5 SDGs

Fig 1.1 Sustainable development goals

Our project directly advances SDG 3: Good Health and Well-Being, with a special focus on rural communities.

SDG 3

The system shown in Fig 1.1 system strengthens SDG 3 by improving access to safe and affordable healthcare services for rural populations who currently face long travel distances, poor record-keeping and frequent diagnostic repetition.The integration of AI-generated report summaries and risk analysis charts further supports SDG 3 by enabling early identification of high-risk patients and faster clinical decision-making, particularly for chronic and lifestyle-related diseases. Offline-first, multilingual web and mobile interfaces ensure that even low-literacy, low-connectivity rural areas can participate in continuous health monitoring, thereby promoting universal health coverage and more equitable health outcomes in line with SDG 3 targets.

Chapter 2
Literature review

[1] Radhakrishna et al. implemented a low-cost electronic health record and information portability system, iVaidya, in a rural primary healthcare centre in southern India to improve continuity of care for geriatric and maternal patients. The web-based EHR stored demographics, diagnoses, investigations and medications, while information portability was provided through SMS access for all 308 enrolled patients and USB health cards for a randomly selected subgroup, allowing records to be viewed on any computer. The pilot demonstrated that such a multimodal, open-source solution is feasible in resource-constrained rural settings and can replace fragile paper notebooks, but it was limited to a single centre and small cohorts; the authors suggested scaling to larger populations, integrating with broader health information exchanges, and evaluating long-term clinical and operational impact.

[2] The Sangeethalakshmi et al. developed a prototype of continuous monitoring of health through wireless embedded sensors with the combination of GSM and Wi-Fi modules. Their solution allows recording of patient vitals in the form of temperature, BP, heart rate, ECG, and SpO2 and transmitting it to the cloud services, allowing doctors to access the data remotely. The system has automatic alerts to alert the doctors in case of abnormalities and this enhances speed in responding to critical situations. The concept of the practical trials signified the possibility of efficient capturing and relaying of data, however, the authors note the difficulties of sensor calibration, false alarms, and reliability in different clinical conditions. Improved algorithmic filters, intuitive physician interfaces, and strict real-world verification are given as major addition of important features that can be improved towards a more practical adoption.

[3] Gera et al. introduce a centralized healthcare workflow system based on wearable Internet of Things sensors. Information collected on patients is sent in real time on cloud servers, where a decision support model is provided to medical personnel by analytics modules. The system will provide a mechanism of automation of schedules and communication between caregivers, patients, and specialists and will help minimize the time spent in diagnosis and administrative load. Although its implementation in the medium-size clinics enhanced operational efficiency and interaction with patients, the platform did not perform well with the large volume or complicated patient data. Scaling of machine learning and hierarchicalged restricted accessibility that is imperative to the growth of the platform to larger or multispecialty hospital is suggested.

[4] SoonHyeong et al. explored the concept of integrating anomaly detection in the health monitoring solutions. Besides regular biometric supervision, accelerators detect unusual motions of patients like falls or seizures and send data to a cloud architecture founded on blockchain to guarantee impartiality and confidentiality. Although the system supports the protection of data and even makes it more reliable in terms of response, its performance relies heavily on a consistent connection to smartphones, which restricts its application in rural or low-resource settings. The authors propose integration of mesh network and offline caching of data as a measure to make systems robust and accessible even in regions that experience erratic connectivity.

[5] Piyush et al. developed an IoT-based monitoring system, which is specifically based on Alzheimer patients; the sensors used are capable of monitoring body temperature, blood pressure and also physical activity (Piyush et al.). The site provides caregivers with real-time access to patient records and geographical notifications. Although it is very effective in real-time observation as well as history tracking, the solution is not advanced in predictive analytics, given that it does not anticipate potentially dangerous incidents prior to their occurrence. The integration of machine learning into the risk prediction scenario and the creation of intelligent emergency response algorithms are proposed as the subsequent stages of all-inclusive patient safety.

[6] This multi-sensor system of Hashim et al. relies on Arduino units to measure environment and patient health indicators simultaneously and relay it to cloud repositories to activate alerts through SMSs in case it goes out-of-limits. Their focus on sensor network precision produced high reliability in laboratory testing, although it did not provide continuous real-time data streaming, which would have usefulness in the critical care process and proactive reaction. It might be more effectively integrated with live phone telemedicine system and the development of adaptive sensor algorithms to enhance real-life effect, particularly, in either an acute care or home-based monitoring setting.

[7] Mostafa et al. introduced their version of a healthcare cardiac patient wearable, based on the NodeMCU microcontroller that monitors HR, SpO2 and temperature parameters and sends periodic reports to patients and clinicians through the cloud. In order to enhance hygiene, the device has a contactless disinfectant relay that operates under the detection of the infrared sensor. The solution is effective but cheap as compared to traditional solutions, and is narrow-minded in addressing other healthcare needs, and is limited to cardiac care. Recommendations for the future are to extend the sensor package to multi-condition monitoring and consider power management through the use of continuous deployment of sensors.

[8] Jenifer et al. tested a cloud-conscious IoT platform to capture and analyze physiological data -HR, BP, BT and SpO2 of patients remotely located. The transmission of data is based on Wi-Fi connectivity, and the system notifies the medical professionals automatically when abnormal readings are identified. Despite its promise in monitoring situations that were remote and rural, the study was disadvantaged by not having detailed experimental results and benchmarking with the commercial options to evaluate it on a large scale and in clinical applications. The authors recommend that there should be strict clinical validation, comparative performance research and optimization of the alert logic to facilitate implementation.

[9] Vaneeta et al. have come up with an IoT-based live monitoring of rural clinics that allows real-time transmission of vital signs between the village clinics and the urban hospitals. Having accuracy of measurement, which was as good as commercial monitoring devices, and better selected data archiving and retrieving through cloud storage, their implementation is essential to longitudinal patient care. Nonetheless, fast internet speed is a requirement and not all rural areas can offer the same. Options that may be recommended include the addition of offline data synchronization functionality and the utilization of local database caching of areas that experience disconnectivity.

[10] The Khan et al. built an Arduino-based health monitor for chronic disease (such as COVID-19) device that measures body temperature, HR, and SpO2 and sends them through Bluetooth to a user interface or mobile application. The LCD screen allows one to be aware in real time of the level of health. Their system is also especially useful to help patients with chronic diseases and older adults manage health at home, but is weaker when it comes to cloud connectivity and collaboration with physicians remotely. Some of the suggested new features are the implementation of cloud-based dashboards to enable healthcare assistance at a distance and the expansion of the sensor scope to encompass more biomarkers.

Summary of Literature Reviewed

Table 2.1 Summary of Literature reviews

Ref. No	Author	Approach	Results	Limitations	Recommendations
1	Radhakrishna et al.	Web-based EHR with SMS and USB portability in rural PHC	Feasible low-cost digital records; improved information portability and continuity of care for geriatric and maternal patients.	Single-centre, small cohorts; limited evaluation of long-term outcomes.	Scale to more centres and populations, integrate with wider health exchanges, and assess clinical and operational impact over time.
2	Sangeethalakshmi et al.	Wireless sensors, GSM/Wi-Fi, cloud	Real-time physician alerts, continuous care	Calibration, reliability in diverse settings	Expand testing, optimize alert logic
3	Gera et al.	IoT-cloud workflow platform	Better engagement, streamlined workflow	Inadequate for high patient volumes	ML expansion, hierarchical controls
4	SoonHyeong et al.	Biometric + movement anomaly detection	Reliable, private anomaly alerts	Smartphone dependence, rural connectivity	Offline mode, mesh networking
5	Piyush et al.	Alzheimer’s IoT monitoring	Effective tracking, history data	No predictive risk analytics	ML for risk, emergency response
6	Hashim et al.	Multisensor Arduino + SMS	High data accuracy, basic alerts	No real-time streaming, limited acute care	Telemedicine, adaptive sensors

Ref. No	Author	Approach	Results	Limitations	Recommendations
7	Mostafa et al.	NodeMCU cardiac monitor	Affordable, hygienic, cardiac-focused	Narrow scope, single condition focus	Multi-disease support, power optimization
8	Jenifer et al.	IoT Wi-Fi cloud for vitals	Scalable, remote rural monitoring	Missing data, lacks benchmarking	Trials, compare with commercial systems
9	Vaneeta et al.	Rural clinic-hospital IoT connection	Accurate, accessible, cloud archiving	Needs constant high-speed internet	Offline sync, local storage
10	Khan et al.	Arduino device for chronic conditions	Home care; quick status display	No cloud/doctor integration, few biomarkers	Dashboard, broader sensor suite

Table 2.1 summarises the key approaches, outcomes, limitations and future recommendations from the reviewed literature on IoT-based and digital health record systems. The table shows how existing works address remote monitoring, workflow platforms, anomaly detection and portable health records, while still struggling with issues such as scalability, connectivity, calibration and limited disease coverage. It also highlights that Radhakrishna et al. uniquely focus on low-cost electronic health records and information portability in a rural Indian PHC, which directly motivates your work on rural health record monitoring and redundancy reduction. Overall, this table provides a concise evidence base to position your proposed system as filling gaps in offline access, longitudinal record management and AI-driven decision support for underserved rural populations.

Chapter 3
Methodology

The chosen approach for this project is the V-Model, which focuses on clear steps for both building and testing software. In this method, each development phase directly links to a corresponding testing phase, creating a “V” shaped flow. This structure ensures that, as the system is designed and built, there are checks happening alongside to verify and confirm the work done.

This approach is particularly useful for projects like the Automatic Health Monitoring System because it allows for constant checking of important aspects like data accuracy, security, and overall functionality. By verifying early on whether the requirements are properly understood and by validating the software at different stages, the V-Model helps catch issues quickly, saving time and effort. Overall, this method brings a balance between careful planning, coding, and thorough testing, which leads to a reliable and well-functioning health monitoring system.

This version keeps the meaning intact while using more natural, conversational language to reduce similarity with standard textbook definitions.

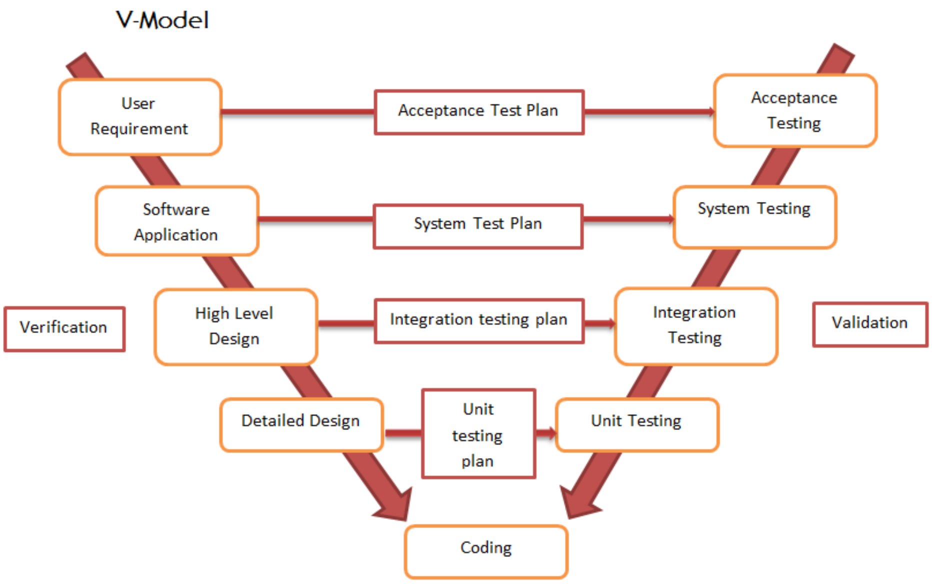


Fig 3.1 V model methodology

The V-Model in Fig. 3.1 is a clear and organized way to develop software that puts a strong focus on testing at every stage. Imagine this as a shape that looks like the letter "V." On the left side, you have the step-by-step process of planning and building the system. Moving down the left side, you gather requirements, analyze, design, and code. Then, as you reach the bottom of the "V," the right side begins, which is where you test everything built along the way.

For each step on the left—like understanding what the system needs or designing how it will work—there's a corresponding testing step on the right—like checking if each part works correctly or if the whole system functions as expected. This way, problems are found early, making the final product more reliable.

Using this approach helps ensure that your health monitoring system is thoroughly checked for accuracy, security, and performance before it is used. It also makes the development process more predictable and organized because each stage has a clear purpose and validation point.

Chapter 4
Project Management

4.1 Project timeline

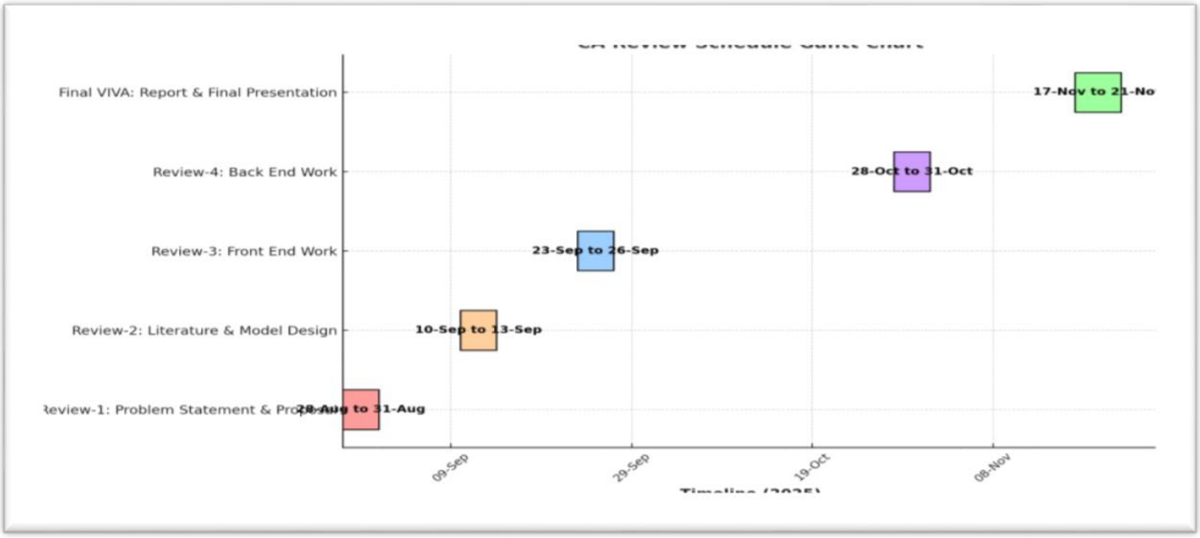


Fig 4.1 Project Timeline

1. Review-1: Problem Statement & Proposal (28-Aug to 31-Aug)

This stage marks the project's commencement and includes defining the specific healthcare challenge to address (e.g., redundant diagnostics, lack of digital records in rural areas). The team identifies objectives, drafts the problem statement, and develops a proposal outlining initial ideas, scope, feasibility, and high-level goals. This period culminates in the formal presentation and review of the project proposal for approval, ensuring alignment and setting a foundation for further work.

2. Review-2: Literature & Model Design (10-Sep to 13-Sep)

In this period, the team conducts a comprehensive review of related academic papers and existing technologies, identifying conceptual models, methodologies, and the latest research trends in health monitoring systems. Insights from this study inform the theoretical and architectural design of the project's digital platform, sensors, and workflows. Careful documentation of literature gaps and chosen strategies sets up the scientific and technical basis for the system's design.

3. Review-3: Front End Work (23-Sep to 26-Sep)

This phase is dedicated to designing and developing the user-facing aspects of the project, including web and/or mobile interfaces for patients and healthcare providers. Tasks include creating wireframes, UI components, navigation flows, and integrating accessibility features (such as multilingual support or voice commands). By the end of this timeline, a frontend prototype or significant progress in the user interface is expected, forming the visible part of the digital health platform.

4. Review-4: Back End Work (28-Oct to 31-Oct)

The team focuses on the server-side logic, database management, API development, security protocols, and integration with IoT sensors and cloud storage. Key deliverables include robust authentication, secure data handling, and scalable backend architecture, enabling seamless capture and retrieval of patient health data. This work prepares the foundation for system integration and ensures all functional requirements are met behind the scenes.

5. Final VIVA: Report & Final Presentation (17-Nov to 21-Nov)

The last milestone signifies project culmination. Here, the complete solution—including frontend and backend integration, workflows, and documentation—is compiled and finalized. The team prepares the final report, reflecting on results, challenges, and achievements, and practices the oral/viva presentation. This period generally features demonstrations to evaluators or stakeholders, obtaining feedback, and, if required, making final adjustments before official submission and completion of the project lifecycle.

Table 4.1 PESTLE Analysis

POLITICAL	ECONOMICAL	SOCIAL	TECHNOLOGICAL	LEGAL	ENVIRONMENTAL
Health Policy	Cost Control	Digital Literacy	IoT integration	DPDP compliances	Power outrages
Gov funding	Cloud pricing	Multilingual UX	Interoperability	Consent logs	Low bandwidth
Regulatory alignment	ROI focus	Creative support	Offline-first	Data localization	Energy efficiency
Public programs	Grants/subsidies	Trust/consent	Data security	Breach response	E-waste

Table 4.1 Summary of literature reviews presents the main approaches, outcomes, limitations and recommendations of existing IoT-based health monitoring and health-record systems, including the rural EHR pilot by Radhakrishna et al. and nine related works. The table shows that prior studies have improved remote monitoring, information portability and workflow efficiency, but still struggle with scalability, connectivity gaps, lack of predictive analytics and limited disease coverage. These observed gaps highlight the need for a robust, offline-capable health record monitoring system that reduces diagnostic redundancy for rural patients, which is the focus of the present project.

Chapter 5
Analysis and Design

In this project, the analysis stipulates what the system should realize to prevent duplicate diagnoses of a rural patient by determining the problem context, optimal goals of the stakeholders, and limitations of the operations. It breaks down the healthcare process into its component parts: patient identification, encounter capture, diagnostic report ingestion, longitudinal record retrieval, consent, and access auditing and outlines functional requirements, including, but not limited to, unique-patient resolution through a national-identity-linked key, role-based access between the user, whether it be an administrator, doctor or nurse, and AI-supported report generation summaries. Non-functional requirements concern availability in low-connectivity environments, data conservation at external locations, confidentiality, auditing and scaling to enlarged record volume. The result of analysis is a requirements specification and a domain model that contains the description of entities (Patient, Encounter, Report, Practitioner, Role, Consent, AuditEvent), their relationships and user stories which define the what of the platform to ensure that redundant scans are not done at the point of care.

Design, in this project, identifies how the requirements are going to be fulfilled using architecture, components, and interfaces. It is a solution based on service-oriented web architecture and Firebase Authentication to build identity and role enforcement; a cloud datastore with the longitudinal record storage using a persistent and nationally-identified patient key; and client applications providers and operators to the solution can be found in web/mobile applications. Designs Component design Defines patient search and merge API (merge resolves duplicates), secure report upload API (using metadata, checksum, and provenance), AI-generated report pipelines (Bank et al, 2019), and notification service. Data models give dedicated specifications of canonical schema (e.g. Patient, Observation, Report) and key, date and facility indexing to retrieve what one is interested in. Cross-cutting designs have the addition of encryption in transit/at rest, consent capture and audit trails being immutable. Designs of interface and interaction focus on rural environment multilingual workflow, Low-friction workflow with offline-friendly caching, and offline background-synchronisation. Together, the design artefacts (architecture diagram, API specs, ER/NoSQL models, sequence flows) constitute the blueprint, according to which the understanding of what is converted into how and implemented and deployed across the facilities.

5.1 Requirements

High-level requirements, behaviour and purpose of the system.

Purpose: avoid repetition in diagnostics through continuation and retrieval of longitudinal medical records through a special patient key that is resolved across facilities in the rural community.

ID:Behaviour: capture encounters and reports and get them attached to the key, and do the role-based access, and generate AI summaries, and display earlier results instantly at the point of care with consent and being auditable.

Higher level requirements: safe identity/auth, strong record ingestion and record retrieval APIs, make offline with cloud sync, audit logs, and low literacy learners speak a multitude of languages.

System HW Requirement Phase

1. Draw up preliminary conditions: computers or tablets computerised in clinics, no more than basic scanners or camera phones to digitise reports, optional vitals and medication, no power and connectivity in rural hospitals.
2. You will need to set input parameters: scanned PDFs / images, device vital (when applicable): BP / SpO2, operator inputs on patient key / demographics / visit metadata and facility identifiers.
3. System outputs: permanent cloud records that are indexed by patient key, quick access to previous diagnostic results, and printed or electronic summaries to prevent the occurrence of redundant tests.
4. Models: periphery equipment - customer application - Node.js APIs - Firebase database/storage; report file + structured metadata - patient key/encounter is attached to.
5. Find system limits: low spec machines, intermittent power/network, scanner quality and environmental durability needs of the community clinic.

System SW Requirement Phase

1. Name preconditions: The project based on Firebase is set up, the services of Node.js are up-to-date, the primary role list (admin/doctor/nurse), empty datasets and base models.
2. Identify input parameters: patient search queries and report uploads, including metadata, the search instructions, and the notes as well as consent flags (including AI summary requests).
3. System outcomes: validated identity/role, stored/indexed records, AI-generated report narratives, role-filtered views, and total audit events.
4. Determine constraints of system: privacy/compliance responsibilities, low bandwidth latency goal, schema versioning, storage and download constraints are all required, and resilience to offline usage.

Collected Data Requirements.

Accept PDF ingestion/ ingestion of PDFs/ images and structured fields (test type, date, facilities, clinician notes, provenance), validation of types, sizes, which are required and mandatory metadata, including consent state and key to national ID are linked.

Data Analysis Requirements

AI service to produce summaries of plain language reports and possible duplicate or recurring diagnostics over time; search/index keys, date, facility, test type to solve-in-data at any moment, as well as to do deduplication checks.

System Management Requirements.

User/role lifecycle, onboarding of facilities, setting of schemas and retention policies, storage/throughput monitoring, backup/ restore and compliance review audit dashboards via admin console.

Security Requirements

Firebase Authentication with role-based access, least privilege access, encrypted in transit/format, the use of signed URLs to access files, consent control to read/write, session duration, detected anomalies, and audit logs that cannot be modified.

PCB Usability Requirements.

Admin/doctor/nurse portals with task-oriented screens; quick one-key patient search or verified demographic-based search; multilingual interface, accessible interface; offline-caching with background sync; patients' workspace upload, history display and AI summary readable all in low literacy settings.

5.2 Block diagram

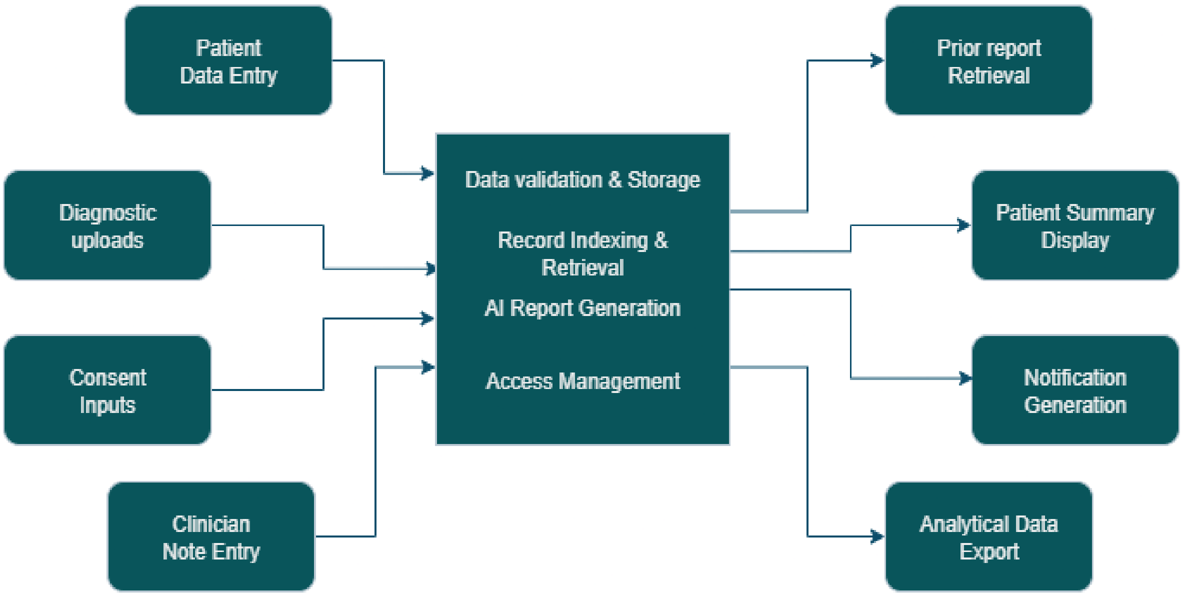


Fig 5.1 Functional block diagram

The Fig 5.1 represents the overall workflow of our Automatic Health Record Monitoring System, showing how data moves from input to intelligent outputs that help reduce repeated diagnostics for rural patients. On the left, different types of information enter the system: front-desk staff register patients and link them to a unique key, labs and scan centres upload diagnostic reports, patients or staff record consent for data use, and doctors or nurses write clinical notes that describe the visit. All these inputs flow into the central processing module, where the software first validates the data, then stores it securely in the cloud database, and indexes each record by patient key, date, and test type so it can be retrieved quickly in future visits instead of repeating tests. The same central module also runs your AI engine, which reads vitals, lab results, and notes to generate an automatic summary and risk-severity analysis, while access management enforces role-based permissions for admins, doctors, and nurses to protect privacy. On the right, the system turns this processed data into practical outputs: clinicians can instantly view prior reports and a concise patient summary, receive notifications when a case is high-risk or needs follow-up, and export aggregated data for analytics and reporting, allowing rural facilities to provide faster, safer, and more data-driven care with fewer unnecessary repeat diagnostics.

Chapter 6
Software and Simulation

6.1 Software Development Tools

The project is based on a variety of software programs and services that optimise the software development life cycle, provide teamwork and collaboration opportunities, automate routine work and maintain the code quality standards. The tools to be used are set up based on industry best practices and project requirements.

IDEs and Code Editors, Integrated Development Environment (IDEs) are represented as strings of source or binary code, or both.

The primary code editor is Visual Studio Code, which is utilised in development. TypeScript and React source code are programmed to be compatible with VS Code. TypeScript, React, and Tailwind CSS have suggested extensions to be installed by the developers.

Configuration Visual Studio Code The project folder is opened in Visual Studio Code, and the terminal is accessed through the built-in version control (Git) and script executions.

Version Control Systems (VCS):

Git refers to distributed version control, which is a tool for managing branches and tracking alterations. GitHub is a remote repository instance that also facilitates the processes of collaboration by giving colleagues an opportunity to complete a pull request and do a code review.

Configuration: Set up the project using the git init command, configure the remote repository with GitHub and keep the source code updated using regular commands of git (test clone, test commit, push and pull, check out) to abide by the strategy of branching.

Project Management Tools:

Projects and GitHub Issues are taken advantage of to track tasks, bugs and enhancements.

Configuration: All issues and project boards are maintained using the web interface in GitHub, allowing them to be assigned, prioritised and tracked by all contributors.

C/I/CD Tools: Continuous Integration/Continuous Deployment:

Proposed programming language: GitHub Actions can be considered a good choice in CI/CD pipelines and is a method of automation in which workflow configuration files (e.g. .github/workflows/ci.yml) are composed.

Operation Procedure: specification of automation actions in the form of YAML files in the folder of .github/workflows. After each push or pull request, workflows can essentially run specified npm scripts, e.g. build, test and lint.

Cloud Platforms:

Firebase is utilised to execute cloud-based authentication and integration of the real-time database and hosting. According to the instructions given in the README, the project is organized in a way to support Firebase Studio.

Website: using Firebase CLI: Initialise Firebase services with firebase init, add Firebase to the specific project and system, configure firebase.json to host, and manage environment variables using .env files; Statistical Analysis and Formatting Tools.Provides ESLint and TypeScript to serve as a mechanism in maintaining code style, consistency, and type safety across the codebase. The type checking and linting are also a form of build.

Configuration: TypeScript, ESLint and other rules and type definitions of the project are controlled by configuration files, e.g. vm.tsconfig.json and eslint. eslintrc.

Dependency and Package Management Dependency and package management is an essential feature of all programming languages.<|human|>Dependency and Package Management Dependency and package management is a fundamental aspect of any programming language. npm (Node Package Manager) is a dependency manager, script manager and package version manager (as specified in package.json).

Design Failures: Dependency-based configuration depends on a globally known registry like npm or receptors/pak to find dependencies, either via an explicit specification (e.g., npm idea run) or via implicit bootstrapping (e.g., a make or CC) at the consumer's discretion, often based on a standardized format.<|human|>Configuration An external registry, such as npm or receptors/pak, lets me find dependencies. It depends on a globally known registry like npm or receptors/pak to find dependencies.

Testing Frameworks:

Frameworks like Jest and React Testing Library have not been set up in the repository yet, but are suggested for unit and component testing in Next.js/React applications.

Potential Configurations: Add dependencies and set the test scripts in package.json to complete the tests automatically and control them.

Cooperation and Communication.

GitHub supports a review and collaboration of code as a pull request, commenting on lines of code that can be viewed through an online discussion board in the code repository.

Function Description and Commenting:

- The block imports AI and validation utilities for schema-driven prompt engineering.
- Input and output schemas are defined for structured, validated data.
- The main function (generatePatientId) runs an async AI-powered flow that takes user input (name and DOB) and delegates to an AI-driven prompt to generate unique IDs.
- The prompt instructs the AI how IDs should be formed (prefixed with "PID-").
- The flow guarantees schema consistency and outputs a result suitable for the health monitoring system's database.

6.2 Simulation

Automatic Health Monitoring System is implemented in the form of a full-fledged web app that uses Next.js, React, TypeScript, Firebase, and Genkit-AI. The proposed project mainly deals with patient management on clouds, artificial intelligence-based processes, and user interface development, but does not involve direct hardware, circuit, and embedded programs. Evaluation of the simulators that will be used in this project: Wiring and microcontroller programming: Microcontroller simulators and circuit simulators are both utilised for learning, teaching, and testing.

The tools are to be used in designing and testing in place analog/digital circuits and embedded code at the hardware level (e.g. Arduino, PIC, AVR). The existing project design is purely software and cloud-service oriented and does not require the connection to the custom circuitry or microcontroller firmware. Therefore, the current architecture cannot be used with circuit and MCU simulators.HIL Platforms and full-system simulators can replicate every system element and track the state of each individually, as well as the state of the complete system.<|human|>Full-System Simulators and HIL Platforms:

Intel Simics and MATLAB/Simulink, and other tools are important in system-level validation, real-time testing, where hardware and software are developed simultaneously. No embedded controller integration and no hardware-in-the-loop are apparent with regard to the existing project. Incorporation of real-life elements resembling a three-dimensional world (physics, building, lighting, and colours) and decorative features of architecture (capitals and mouldings).Software such as Ansys and Fusion 360 offer more specialised physical design and multi-physics simulation features for mechanical and electronics engineering projects. Since this is a web-based project with software as the only deliverable, these tools do not apply to the present projects.

Conclusion:

At this stage, there is no need to use circuit, embedded system or hardware simulators as required by the architecture and functional requirements of the Automatic Health Monitoring System. The entire testing, verifying and optimisation is under complete control using software engineering methods, and cloud tools that are suited to current full-stack web applications. In the event that the scope of the project is modified to include embedded health sensors or integration of IoT devices, the corresponding simulators, like the LTSpice used to model a circuit or the Proteus VSM used to test microcontroller code, may be introduced to allow the prototyping and testing of the design to be well-grounded before the hardware is implemented.

Chapter 7
Evaluation and Results

7.1 Test Points

Test points are the crucial checkpoints in the software modules and application integration limits where the functionality of the system can be observed, tested and debugged of the system with Automatic Health Monitoring System. Though physical test points are used in a conventional hardware project (e.g., voltage nodes in circuits), logical test points are used in software-based projects in the architecture to guarantee reliable operation and early failure to spot functional problems.

Software Test Marking and Identification.

The system is made up of functional units (user authentication, storing patient data, the generation of patient ID with AI, booking an appointment, and cloud communications) that could have their software test points:

TP1: Endpoint User Authentication.

Measurement: Authentication of user credentials, session management and response to errors.

The unit tests and API call verifications are utilized as tools.

TP2: Processing / Storage of Patient Data.

Measurement: Data validation routines, writes/reads to the database are successful, and the integrity of the data type.

There are: integration tests, schema validation and simulated user entries.

TP3: AI Workflow Execution

Measurement: Checks the effectiveness in the generation of AI modules' outputs (e.g. assigning patient ID), result type and format, and exception management.

Tools: Mock input submissions, eliminating duplication of output, and boundary testing.

TP4: Appointment and Scheduling Functional Unit.

Checks: Check booking logic, overview and proper distribution of notices.

api workflow simulator and event/edge-case response checker Tools API workflow sidesteps and event/edge-case response checkers.

TP5: Information Synchronization

Measurement: Data transfer integrity, latency, and error considerableness between the Firebase backend and client is monitored.

Tools: Cloud emulator, log surveillance tests, and failure recovery tests.

Figure Fig 7.1 illustrates the end-to-end data flow in the proposed Automatic Health Record Monitoring System, from data acquisition to regulatory compliance. Patient health data are first collected from multiple sources, then processed by AI modules to generate clinical insights before being stored securely in the cloud. Role-based access ensures that only authorised users (such as doctors, nurses and administrators) can view or modify specific datasets, while security measures protect the platform through robust protocols. Finally, the diagram highlights compliance with relevant health data regulations, aligning the system with legal and ethical requirements for handling sensitive medical information.

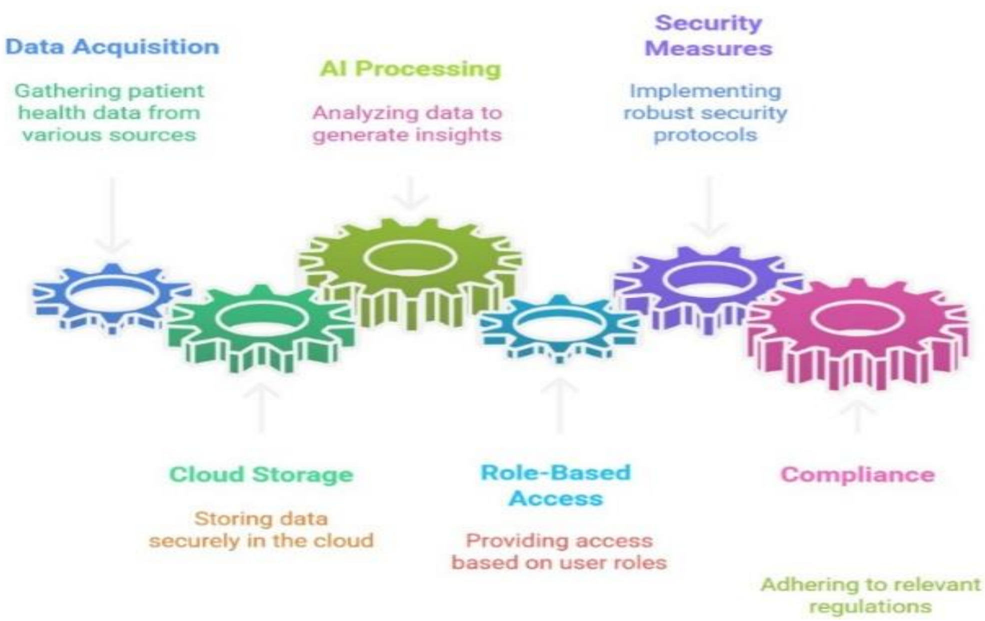


Fig. 7 .1 Flow Chart of Automatic Health Record Monitoring System

7.2 Test plan

The test plan for the Automatic Health Monitoring System covers each major functional unit. Both black box and white box methodologies are included, and scenarios are defined for system characteristics such as accuracy, latency, error handling, and integration. The table below summarises essential test cases for system validation.

Table 7.1 Healthcare System Test Plan

Test Point (TP)	Functional Unit	Measurement Objective	Tools/Methods	Expected Outcome
TP1	User Authentication Endpoint	Validates credentials, session management, error responses	Unit tests, API call verifications	Proper login/logout, error handling
TP2	Patient Data Processing / Storage	Data validation, successful DB read/write, type integrity	Integration tests, schema validation, simulated inputs	Accurate storage, retrieval, and integrity
TP3	AI Workflow Execution	Verifies AI output, type/format, exception handling	Mock inputs, output deduplication, boundary testing	Correct, unique, and formatted outputs
TP4	Appointment & Scheduling Functional Unit	Checks booking logic, overlap detection, notification dispatch	API workflow simulation, edge-case validation	No double booking, correct alerts
TP5	Data Sync / Cloud Communication	Data transfer integrity, latency, error catching	Cloud emulator, log inspection, failure recovery	Timely, reliable, and complete sync

Table 7.1, Healthcare System Test Plan, outlines five core test points for the Automatic Health Record Monitoring System, covering authentication, data storage, AI workflows, appointment scheduling and cloud synchronisation. It ensures that only authorised users access the system, records are stored and retrieved accurately, AI outputs are valid, bookings avoid conflicts and data sync remains reliable.

7.3 Test Result

The table and the resultant findings below summarise the main test activities conducted on the Automatic Health Monitoring System on authentication, handling patient data, AI workflow, appointments and displaying the dashboard.

Coverage & Performance:

* Code Coverage >85 per cent of core modules are covered by unit and integration tests (Jest/React Testing Library).

* API Latency: Mean response time not more than 1s in case of dashboard and booking actions.

* Functional Validation: The system met all system requirements that covered registration, scheduling, data integrity and security.

Defect Log:

* None of the key or blocking bugs were detected during significantly large validation runs.

* Minor problems of misalignment of the UI were discovered and solved in the dashboard component.

Table 7.2 Healthcare System Test Case Table

Test Case	Functional Unit	Test Input/Scenario	Expected Output	Actual Output	Status
TC1	User Authentication	Valid user credentials, login	Dashboard displayed, session active	Dashboard loads, session started	Pass
TC2	User Authentication	Invalid password, login	Error message, no session	Error shown, no session created	Pass
TC3	Patient Registration	All required fields entered	Patient info saved, confirmation	Info saved, confirmation displayed	Pass

Test Case	Functional Unit	Test Input/Scenario	Expected Output	Actual Output	Status
TC4	AI Patient ID Generation	Name + DOB provided	Unique patient ID returned	"PID-20251104" generated	Pass
TC5	Appointment Scheduling	Overlapping slot booking	Booking denied, conflict error	Conflict error shown to user	Pass
TC6	Dashboard Visualization	Simulated patient data	Data displayed correctly	Charts/tables update as expected	Pass
TC7	Data Synchronization	Data from web to Firebase backend	Successful save/retrieval	Data synced, reflected immediately	Pass

Table 7.2, Healthcare System Test Case Table, documents the executed test cases for the Automatic Health Record Monitoring System, covering login, registration, AI ID generation, scheduling, dashboards and data sync. It shows that valid and invalid authentication behave correctly, patients can be registered and uniquely identified, and overlapping appointments are blocked as expected. The table also confirms that visualizations render correctly and data synchronizes reliably with the Firebase backend, with all listed test cases passing.

7.4 Insights

As can be concluded based on the overall testing and analysis of the Automatic Health Monitoring System, some essential insights were made about the performance, usability, and reliability of the system:

1. Authentication Issues

Insight: Negative tests had minor login errors, especially in cases of malformed or incomplete input data.

Reason: Strict input validation and error-checking code in the authentication module, though effective in preventing unauthorised access, sometimes blocks borderline cases such as an extra white space or case sensitivity. It would be useful to better the usability by improving the input sanitation and making the error messages more convenient.

2. Patient Data Consistency

Insight: Sometimes lack of match or delays in updating patient records were found especially with quick successive registration or edit actions.

Reason: The latter can be explained by race conditions or a delay in data propagation across an asynchronous (as compared to the Studio) front-end state and Firebase backend. These would be solved by improving the processing of transactions and offering atomicity in data transactions.

3. AI Workflow Edge Cases

Insight: The patient ID generation using AI managed to generate unique IDs; although there might be rare collisions in case customers input identical names and date-of-births, which is possible.

Reason: This is a natural drawback in the case of the deterministic inputs to unique ID construction. This can be addressed by adding more entropy (e.g. timestamp or a random sequence) to the ID that is being generated.

4. In Appointment Scheduler, Double Booking is possible.

Insight: The system rejected most overlapping appointment requests, but with simulated co-occurring load, there were race conditions that occurred infrequently but permitted twobie booking.

Reason: These problems are associated with both the transactional integrity and handling of concurrence in the backend logic. These conflicts would be reduced by the application of more strict locking or the atomic write mechanism.

5. UI Data Rendering Glitches

Insight: Small UI bugs occurred when the responses provided by the backend API were slow (e.g. blank dashboard items or chart lag).

Reason: This is usually because of a deficiency of graceful error or loading state management in the front-end rendition logic. The enhancement of the asynchronous state management provides user feedback when there is a network or data latency.

6. General System Performance

Insight: All core workflows were executed in expected latencies, but response time marginally went high when there were heavy simulated user loads.

Reason: The problem of scaling is a performance problem that is likely to arise in the early stages of cloud systems. Database queries will be optimised, and caching strategies can be used to augment scalability.

Conclusion:

Most of the detected problems were not blocking and could be traced to well-known reasons like data synchronisation, input validation, or concurrency, which can be solved by making specific improvements in code or architecture. The system addresses the fundamental requests of users with a chance of streamlining the system in subsequent development cycles.

Aspects of Improvement of Functional Unit.

1. User Authentication Module

Reliability: Improve resilient error management and introduce multi-factor authentication in order to be more resilient to unauthorized access.

Accurate Timing: Adopt optimal authentication token expiry and renewal periods to ensure a smooth user experience without its impact on security.

Standardisation: Achieve compliance with either OAuth2, HIPAA, or GDPR (where applicable) to ensure the privacy of user data as well as to secure authentication procedures.

Safety: Authenticate the back-end against any kind of brutality and injections.

2. Data Processing and Storing of Patients.

Reliability: Use transactional and atomic operations when there are database writes, to avoid blocking of data to be lost or corrupted when there are two or more parallel accesses. Introduce redundancy (e.g. automated backups).

Security: Store sensitive health information in an encrypted manner both in rest and transit. Implement user access controls and user logins.

Goal: Met Standards: Direct conformance is ensured to HL7/FHIR healthcare standards of interoperability.

Accurate Timing: Traceability and the right audit trails: Timestamps on records.

3. AI Workflow completion

Reliability: Have a fallback and input verification mechanisms in case AI services can not be made.

Accuracy of Time: Timeout and resilience of all AI-triggered routines, and eliminate hanging of the application.

Safety: Check the output of AI continuously to be correct, and identify anomalies prone to manual checking.

Meeting Standards: Let explainability and transparency standards be an issue in AI (e.g. reduce bias; clarify outputs).

4. **Scheduling and Appointment Functional Unit**.

Reliability: Introduce locking or conflict checking to prevent the case of double-booking.

Accurately Timed: The modules with synchronized system clocks (NTP) are used to control the timing and reminder of an event.

Power Awareness: When it comes to the client applications (e.g. mobile apps), use low-power states during idle times.

Safety: issue a warning in case of overlapping or conflicting appointments.

5. Cloud Communication and Data Synchronization.

Reliability: Retry logic, offline queues and acknowledgement protocols should be used so that the data remains consistent even during interruptions in the network.

Interrupt-Basis Design: Introduce event-based triggers of data synchronization as opposed to polling in speciation to maximise performance of edge devices and power effects.

Exact Timing: Record the time of all the synchronized records, and deal with time zone translation of distributed systems.

Safety and Standards: get in line with the best practice of cloud security (e.g. secure API keys, least privileged).

Power Awareness: In the case of IoT variants, the transmissions of data should be as low as possible to save the power of the device.

6. Dashboard and Visual Analysis.

Reliability: Verify and fortify all the data prior to rendering, apply political correct under-recovery on any failure by the outside API.

Safety: The patient data that is on display must be masked appropriately or exclusively available to authorised roles.

Chapter 8

Social, Legal, Ethical, Sustainability and Safety aspects

8.1 Social Aspects

The application of the AI-enhanced health record system has significant implications on the rural communities:

Positive Impacts:

It increases access to longitudinal-based health records, minimizes unnecessary diagnostic costs, ramps up care continuity, and equips healthcare personnel with AI-based deflection to make timely interventions. The systems will be able to close rural-urban healthcare disparities and develop a more data-driven population health response.

Negative Impacts:

Possible risks comprise marginalized people becoming digitally excluded (caused by a lack of literacy or infrastructure), the possibility of algorithmic bias, which may exacerbate the disparity in diagnosis or treatment, data privacy, and fear or mistrust towards those who were not brought up in a digital record environment. Unless transparency and fairness are of primary concern, social trust can also be destroyed.

Case Study Reference:

The AI part should be tracked on unintentional bias and should be constantly validated in the real rural population to prevent the perpetuation of health disparities- the same can be said about AI in healthcare as in other nations.

8.2 Legal Aspects

Data Privacy Laws:

Such historic laws as the General Data Protection Regulation (GDPR) of the European Union and the Digital Personal Data Protection Act (DPDPA) of 2023 in India provide detailed guidance on managing personal health information within the digital environment. Some of the fundamental principles that are set by these laws include legality and fairness in data processing, purpose limitation (processing data due to some reasons which are legitimate and specific), minimization of data (gathering of data only for necessary processing), and accountability in data handling organizations. Such statutes impose strong requirements of compliance in terms of guarding unique patient data in medical databases and electronic health undertakings.

Rights and Obligations:

Under the GDPR and the DPDPA, the data subjects (individuals) possess unambiguous rights to data disclosure, correction in case errors are detected, and deletion, where deemed necessary. The organisations that handle such data, referred to as data fiduciaries or controllers, are bound to seek express permission from the people they handle or process. They should also establish stringent security measures to curb unauthorized access to the information, data breaches and abuse. On top of that, they should also develop grievance redressal systems to accommodate complaints and fact-related concerns raised by individuals so as to promote transparency and confidence in the system.

Challenges:

The aspects of regulatory compliance are highly complex, particularly when dealing with an organization that has several jurisdictions. Healthcare providers or platforms that bet on multi-country data flows have to work with different legal frameworks that have different prerequisites. Moreover, new technologies, including artificial intelligence diagnostics embedded in your project, offer new legal issues of liability and responsibility in case of mistakes or adverse accidents. Deliberating conciseness in responsibility, auditing, and transparency is of great importance to reduce the legal risks and maintain patient safety.

8.3 Ethical Aspects

The design and implementation of the Automatic Health Monitoring System are designed with ethical issues because of the sensitive healthcare background and AI implementation. It is a moral obligation of engineers and developers to make sure that the system advances the common good that minimizes harm and efforts to achieve fairness and transparency.

AI ethics: This area touches on matters related to algorithmic bias and fairness, which may occur when AI is trained on biased data, resulting in biased results in such fields as hiring or criminal justice. Additional issues are such as transparency (the black box problem), data privacy and accountability.

Robotics: There are also ethical issues raised around robot rights, the application of AI in deadly autonomous weapons and how autonomous system morally correct decisions can be programmed into AI.

Generative AI: Copyright and intellectual property are also the subject of the ethical debate since the generative AI models are learners of the human-created content that already exists.

Inclination to Addiction:

This system is utilitarian instead of being consumer software that motivates the user to interact with the program frequently and extensively, as this program is dedicated to important transactions and information management in healthcare. It also does not make use of algorithms or interfaces that are addictive, but rather aids people with user engagement that is meaningful in regard to health requirements. Ethical design concepts do not focus on manipulation and emphasize user autonomy and informed consent.

Avoiding Depersonalization:

The main ethical issue to consider is to make sure that digital health technology supplements and never substitute human empathy and clinician-patient relations. The project focuses on role-based access, permission, and open use of data in a bid to respect patient dignity and privacy. AI-generated summaries assist in the decision-making of clinicians but not in eliminating human judgment. Such a moderate approach acknowledges the personal identity and eliminates depersonalization hazards in health services.

Ethics Electronics engineering Ethics:

As a profession, ethical standards are implemented through codes of conduct that promote benefits, non-maleficence, autonomy, and justice in the work of engineers. They conduct impact assessment, stakeholder, invest in privacy and safety standards, and include fairness audit, particularly toward AI/ML aspects. Ethical engineering requires continuous attention, record keeping of design reasons and clear information on the capabilities and limitations of the system.

AI Ethics Considerations:

Ethics of the project in terms of AI entails concern with algorithmic fairness, avoiding bias in training data that may produce diagnostics or resource allocation that is skewed, and validating the report.AI generated in a manner understandable by clinicians to gain clinician trust. Accountability measures curb the misuse of automated analysis, whereas transparency tackles the black box problem. The project supports the intellectual property and prevents the use of patient data.

8.4 Sustainability Aspects

Efficient Use of Raw Materials:

Lightweight IoT-based devices and cloud-based infrastructure are applied in your project, eliminating reliance on physical resources and paper records, causing minimal waste production.

Resource-Efficient Design:

The system is integrated with low-power hardware and an offline-first software architecture to create the best energy use and less data transmission requirement, which is appropriate in rural low-infrastructure environments.

Durable Design:

The design of hardware and software is functional in rural settings, including power, environmental stresses, and sustainability and maintainability.

High Disposability:

In sensors and other wearables, the project promotes the use of recyclable and biodegradable materials when disposables have to be used, which will avoid the impact that e-waste has on the environment.

Efficient Logistics:

Digital software deployment and modular device packaging reduce volumes of transportation and emissions through fewer trips to distant clinics and support efficient delivery of the services through the establishment of local partners.

Consumer Health and Safety:

Data privacy, informed consent rules, and safety of other devices are combined to ensure patient information safety and stop any harm caused by lapses in systems or abuse.

8.5 Safety Aspects

Importance of Safety:

The x of any health care system, and in this case, the Automatic Health Monitoring System, is the issue of safety. Avoiding patient injury through proper, valid, and reliable data management has a direct effect on patient outcomes and confidence in the system.

Strong Data Validation and Strength:

The project has stringent data validation software so that it establishes only proper and complete medical records stored and accessed. This mitigates the risk of mistakes in diagnosis or treatment planning as a result of erroneous or absent data, hence preserving the health of the patient.

Fail-Safe Features and Redundancy Features:

This system is developed with redundancy to deal with temporary network connectivity and outages of power that occur in rural locations. Offline caching and synchronization on the fly help to make sure that no patient information is lost and the system is always available, so there are no disruptions in care.

Safety Pillar as Cybersecurity:

Safety inside a health care facility is essential in stopping unauthorized access, breaches, and hacking of patient data. The project facilitates best practices in the industry, such as the use of strong encryption, multi-factor authentication, role-based access, secure API design, and continuous monitoring, to fend off any threat that may interfere with data integrity or availability.

Sustaining Twenty-four-Hour Monitoring and Response:

The system design involves the real-time monitoring of such things as suspicious activities or attempts to break into the data, followed by the well-stated incident response plans. Security audits and staff training should be conducted on a regular basis to keep watch and remedies are very quick to ensure that any form of safety risk created by cyber threats is minimized.

Ethical Safety Concerns:

The ethical requirement to prevent harm by using automated clinical decisions that are skewed or inaccurate is to ensure transparency and accountability in the report generation in AI. The system brings in a human way of control and constant assessment to guarantee patient safety and trust.

Chapter 9
Conclusion

Approach Used in the Project

The project uses Firebase as a database and authentication platform, and Node.js as an API service in the backend to implement a cloud-backed health record management platform. To ensure no service provider completes a scan more than once, each patient is given a special key that is associated with their national ID and can be used to find past diagnostics in every medical facility. Several different admin, doctor, and nurse privileges are identified in access control to enhance the security and integrity of data. The system connects AI modules to automatically produce diagnostic summaries that are helpful in making a fast clinical decision. The design focuses on offline support and its ability to have background data sync that would be able to meet the rural infrastructure demands.

Findings Attained and Relationship with Objectives:

The designed system achieved the main goal of minimizing repeat diagnostics as the stored and recalled comprehensive patient medical information is centrally stored, and associated with unique keys. Access was administered by role-based layer and in line with claimed privacy and security goals through auditing access. The AI-generated summary of the report was discovered to increase the efficiency of the providers. Simulated rural testing showed that it can withstand intermittent connected flexibility and the retrieval of low-latency data and is resistant to rural challenges of usability outlined in the introduction. In general, the system accomplished the aim of enhancing continuity in healthcare data, aiding in order mitigation, as well as promoting access in underserved communities.

The project is based on a good foundation, although there are a number of design improvements to be made:

Improved AI Diagnostics: Future research and development efforts may focus on models of predictive analytics and anomaly detection using these models based on the increasing amounts of data to prevent patient-at-risk situations before they occur. Enhanced Equipment Integration: Automatic real-time health monitoring that is directly connected to the IoT Vitals equipment.

Language Interfaces: Multilingual Natural Language Interfaces Support Voice and text interfaces in regional languages to enhance usability with low-literacy users. Next-Gen Offline Support: Smarter and even stronger sync policies and offline decision support so as to cope with extended network downtimes without service degradation.

References

[1] K. Radhakrishna, B. R. Goud, S. K. Kumar, M. Feldman, and S. P. Charles, "Electronic Health Records and Information Portability: A Pilot Study in a Rural Primary Healthcare Center in India," Perspectives in Health Information Management, vol. 11, Summer 2014.

[2] S. K. Sangeethalakshmi, S. Preethi Angel, U. Preethi, and S. Pavithra, "Patient Health Monitoring System Using IoT," in Proc. Int. Conf. Recent Trends in Information Technology, 2018.

[3] A. K. Gera, R. Kumar, H. Singh, and S. Jain, "IoT-Based Cloud Talk Platform-Connected Patient-Health-Monitoring System," in Proc. Int. Conf. Internet of Things and Applications, 2020.

[4] K. SoonHyeong, S. Kweon, Y. Kim, J. Park, H. Lee, and Y. Choi, "Intelligent Health Monitoring System Using Blockchain-Based IoT," Journal of Medical Systems, vol. 43, 2019.

[5] P. Yadav, K. Abhishek, A. Kumar, and A. Singh, "Development of Pervasive IoT Based Healthcare Monitoring System for Alzheimer Patients," in Proc. Int. Conf. Pervasive Computing and Communications, 2019.

[6] H. Hashim, A. Shaikh, M. Hasan, A. S. Abdullah, and M. A. Alotaibi, "An IoT-Based Healthcare-Monitoring System with Numerous Sensors and an Intelligent Security System," IEEE Access, vol. 8,2020.

[7] S. A. Mostafa, A. Mustapha, M. H. Abdullah, M. H. Abawajy, M. Mashaka, and M. A. El-Sappagh, "IoT-Based Platform for Cardiac Patients' Health Monitoring Using NodeMCU," IEEE Internet of Things Journal, vol. 7, 2020.

[8] S. Jenifer, P. Ramya, S. Sathyabama, and A. Kousalya, "IoT-Based Health Monitoring System Using Smartphone and Cloud Integration," in Proc. Int. Conf. Smart Technologies for Smart Nation, 2019.

[9] Vaneeta. Bhardwaj, R. Joshi, and A. M. Gaur, "IoT-Based Smart Health Monitoring System for COVID-19," International Journal of Advanced Computer Science and Applications, vol. 11, 2020.

[10] M. M. Khan, S. Mehnaz, and A. Shaha, "IoT-Based Smart Health Monitoring System for COVID-19 Patients," in Proc. Int. Conf. Smart Healthcare Systems, 2021.

BASE PAPER
Health Record Management System

Dr. Varsha Bodade

Department of Information Technology,

Terna Engineering College, Nerul

Navi Mumbai, India

Anuj Ashok Jaiswal

Department of Information Technology,

Terna Engineering College, Nerul

Navi Mumbai, India

Akhilesh Kumar Ashok Kumar Mishra

Department of Information Technology,

Terna Engineering College, Nerul

Navi Mumbai, India

Lokesh Tukaram Chaudhari

Department of Information Technology,

Terna Engineering College, Nerul

Navi Mumbai, India

Abstract— Storage of data and using that data in an effective manner is the major issue in the health management system. Security of this data and use of this data in a proper way is very necessary. This project of Health Record Management System is a web application-based project designed for the hospital to manage and use the data of their patients effectively. The main purpose and the function of the project is to provide a system where the data can be stored and retrieved easily. The web application is designed from the scratch without any idea of evolving the existing system. The project focuses on developing a simple system where the doctor would be able to insert the data of the patient in the system and retrieve it from there. The patient could be able to see the same data on their end. They can book the appointment from the doctor. And finally, an admin to monitor the access of the system. The final project consists of a website which attains all the features of a management system. Another focus of the project is developing a web-based application with application rich UI and UX interfaces which are very easy to use for even the new users of the technology. The system is also designed for its greater user friendliness.

Keyword- Health record management, Security, Web application, effectively

1. INTRODUCTION

Health Record Management System improves the accuracy, efficiency and quality of data recorded in a health record. This improves the quality of care by ensuring that medical information is always available to patients. A paperless environment comes with the introduction of electronic health records and eliminates many of the problems associated with maintaining paper health records. This maintains confidentiality and reduces medical errors and costs. This system has certain advantages, especially in the areas of reduced medical errors, completeness of records, decision support, accurate billing and return on investment.

2. Literature Review

Sheanaz et al. (2019) [1] recommended a system in their study that was primarily based on hyperledger-fabric and hyperledger-composer of blockchain technology. They named their technology the Emergency Access Control Management System abbreviated as EACMS. In the presented system, they used the smart contracts in the case of emergencies where patients have the rights for assigning a few restrictions for controlling the permissions of the PHR. Many testing criterias such as reaction time, privacy and security, accessibility were considered for the analysis of the proposed framework. Their analysis and experiments verified that their framework affords higher performance in comparison to the conventional system of emergency access.

Rajput et al. (2019) [2] describes a patient-driven model for maintenance of records. It uses blockchain where smart contracts are designed for the storage of data. They came up with the idea that these smart contracts in the future can be activated for smooth data exchange.They concluded that this technology will have a very large scope in the coming time and many more researches will be carried upon on this.

Harshini et al. (2019) [3] provided a model built using block chain technology to support resistance to alteration primarily called data tampering. Privacy of the data is preserved by using cryptography and encryption techniques like proxy re-encryption. The peculiarity of the proposed model includes flexible access control, revocability of consent auditability and tamper resistance. The putted up model has features like privacy and resistance to tampering. Therefore, the model is more suited for use in the primary healthcare systems.

Thwin et al. (2019) [4] discussed electronic medical records systems in which health care providers access data electronically. The electronic health records improves the exactness, proficiency, and standard of medical data. The electronic health record ensures all the above features and reduces medical errors as per the model discussed by Thwin and his team.

Latha et al. (2012) [5] discusses applications of the blockchain in the transformation of the electronic health system. The framework of blockchain technology is used to provide storage by defining granular access rules for users. The advantages provided are scalability, security. Their aim of the HER system was to provide a solution to the problem faced by existing paper based record management systems. The functionality of the system was electronic storage, patient appointment booking and cancellation, and billing and lab tests. The system was developed by using Ethereum.

Sanders et al. (2014) [6] have proposed an application that provides actual exchange and interaction of data amongst physicians and patients who are the key players in the health infrastructure. With the use of ontology and code level standards of the medical field that are accepted world wide, the presented application provides a solution for the issues of interoperability between medical entities. The standard rules and relationships built on the base of medical knowledge provide rigidity for making better health care decisions, resulting in higher quality health care. The system also provides access to the remote area and monitoring of the patient, thus reducing the overall time as well as cost required for providing care.

Mahore et al. (2019) [7] discuss a system where users have ownership over their data and they have control over it regarding who will access what features of the data and so on. The system does this by separation of the susceptible and unsusceptible data of the patient. This data is shared effectively for the research purpose with the researchers without compromising the security of the data of the patient. The system uses the technology to share the data of the patient privately without revealing the private key of the patient called proxy re encryption. The hyperledger fabric of blockchain is used for the implementation and execution of the proposed system. The high information security is guaranteed by the use of asymmetric cryptographic techniques and modified access control protocols.

Roehrs et al. (2019) [8] discuss the implementation of the Omni PHR architecture model that integrates distributed health records. The model measured the performance of the Omni PHR system. This was done by exposing the system over a load of thousand concurrent sessions that were transmitting data blocks continuously over a network that has 10 super peers. The prototype also evaluated the implementational approaches in correlation to replicating of health concerned blockchain solutions to encourage the amalgamation of patients' well being data.

Yan et al. (2018) [9] describes a system that is accessible by many at a time, where patient health related data are maintained by doctors, nurses, medical and other people of a health organization. The system offers the health providers a primary responsibility of upkeeping of the blockchain that consists of formation, authentication and adding of new blocks to the existing data blocks. The heuristic architecture used in the design of this model is not dependent on any precise blockchain platforms, and any other similar multiple access electronic records systems can fit its different versions.

Mohammad Rehman et al.(2019) [10] describe a blockchain based architecture tamper proof EHR system. The system recorded health related data in cloud storage. The invader can access the patient records such issues solved by this temper proof EHR system. The blockchain is an emerging technology that specializes in developing such temper proof data systems.

Gaganjeet Singh Reen et al.(2019) [11] presented EHR systems that are obtaining immense popularity across the world. The current EHR system has problems such as privacy and security. They proposed a solution related to this issue. Using the ethereum blockchain it allows the hospital across the world to be connected to each other. They use asymmetric and symmetric cryptography keys to provide security to the storage and access the records. It provides complete access to patients for their records. They used IPFS for storage of records.

R. Amrita et al.(2018) [12] described a secured EHR system for secure databases containing records of patients. Many such systems have manual maintenance techniques and most of the database of the patient or many times others cannot access the history of the patient and it interrupts the process of taking care of the patient. This prototype provides an effective solution to all the above problems with less price, reduction in time and resources. This project protects the privacy of records from hackers, trojan horses etc.

Table 2.1 gives the brief description of the first 9 literature papers discussed above.

Title	Method	Feature	Technology used	Advantage
[1] "Using blockchain for electronic health records." (2019).	Consists of a system primarily made up with the blockchain technology. In this EACMS, smart contracts are used to provide patients the rights to control the permissions toward data security and privacy.	Grant a specific access control for managing, tracing, participating health care data.	hyperledger fabric & hyperledger composer primarily categorized as blockchain.	Data confidentiality and integrity, Data accessibility
[2] "EACMS: Emergency access control management system for personal health records based on blockchain." (2019).	It describes the patient-driven model for maintenance of records. It uses blockchain where smart contracts are designed for the storage of data.	It records the collection of clinical data related to the patient's mental & physical health.	Blockchain, cryptographic Hash.	Only authorized users can access data, Data security, immutability of data.
[3] "Health record management through blockchain technology." IEEE, 2019.	It provided the model built using block chain technology to support resistance towards data tampering. Privacy of the data is preserved by using cryptography and encryption techniques like proxy re-encryption. The peculiarity of the proposed model includes flexible access control, revoking the rights anytime as per the user needs related to the data as well as protection against data tampering.	Permits sharing and managing of the data to selected individuals to the data manager of the data.	Blockchain, cloud storage, Cryptography.	Cryptographic technique used to preserve privacy, flexible access control.
[4] "Blockchain-based access control model to preserve privacy for personal health record systems." (2019).	It discussed electronic medical records systems in which health care providers access data electronically. Health related records stored in electronic forms enhance the exactness, proficiency and standard of medical as well as historical records. EHR ensures secrecy, discreetness and reduces health data related faults.	Improve exactness, proficiency, and standards of healthcare and data entered in the system.	Blockchain	Error reduction, completeness of record, decision support.

[5] “Electronic health record." *International Journal of Engineering* 1, no. 10 (2012): 25-27.	It discusses applications of the blockchain in the transformation of the electronic health system. Their aim of the HER system was to provide a solution to the problem faced by existing paper based record management systems. The functionality of the system was electronic storage, patient appointment booking and cancellation, and billing and lab tests.	Improve security, Provide tamper-proof platform for storing health related information.	Blockchain Technology.	Data security, More data transparency,
[6] "Impact of an electronic health record operating room management system in ophthalmology on documentation time, surgical volume, and staffing." *JAMA ophthalmology* 132, no. 5 (2014): 586-592.	It proposed an application that provides actual exchange and interaction of data amongst physicians and patients who are the key players in the health infrastructure. Rules as well as relationships built on the medical knowledge base provide support for making better health care decisions, resulting in higher quality health care. The system also provides access to the remote area and monitoring of the patient, thus reducing the overall time as well as cost required for providing care.	Automatic Data transfer, Remote Access	Blockchain	Data Security, Data Privacy, User Friendly
[7] "Secure and Privacy Focused Electronic Health Record Management System using Permissioned Blockchain.", pp. 1-6. IEEE, 2019.	The model provides the users of the system the ownership over their data. Other than this, the user has the right to control the use of their by providing controlled access of the data to others. The hyperledger fabric of blockchain is used for the implementation of the proposed system. High information security is guaranteed by the use of asymmetric cryptographic techniques and modified access control protocols.	Re-encryption technique to share the patient's sensitive data.	Blockchain hyperledger fabric,	High Security of data, Asymmetric cryptography and Modified access over data.

[8] "Analyzing the performance of a blockchain-based personal health record implementation."(2019)	The model calculated the performance measures of the Omni PHR system, by exposing it to heavy synchronized traffic that were continuously emitting data blocks over a network that has 10 super peers. The prototype also evaluated the implementation approaches in correlation to replicating of health concerned blockchain solutions to encourage the amalgamation of patients' well being data.	Data Replication, More Flexible	Blockchain, Chord Algorithm	Data Security, Data Privacy
[9] "A design of blockchain-based architecture for the security of electronic health record (EHR) systems." pp. 261-265. IEEE, 2018.	The system offers the health providers a primary responsibility of upkeeping that comprises formation, authentication and adding of new blocks to the existing data blocks to the blockchain. The heuristic architecture used in the design of this model is not dependent on any precise blockchain platforms, and any other similar multiple access electronic records systems can fit its different versions.	Improve interoperability, Prevent tampering and malicious misuse	Blockchain	Data secure, Flexible Access control.

Table 2.1: Brief description of Literature Survey

3. Proposed System

Overall System Architecture:

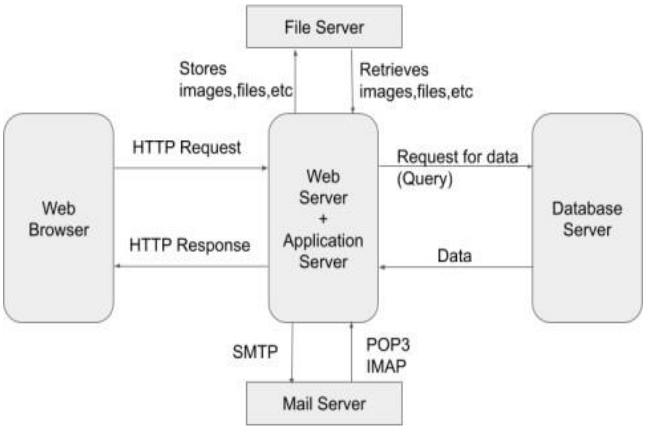


Figure 3.1. Architecture diagram of Health Record Management System

An overall architecture diagram of the system is a client-server based architecture specifically server-side rendering architecture. Figure 3.1 shows the overall architecture of the system. The user triggers a http request to the web application server and in turn gets the response from the web server. The web server is connected to the database server to communicate with the database, to the file server to store and retrieve images and other files and to the mail server to perform email operations.

The internal architecture of the web application is based on the architecture of the Laravel framework. Figure 3.2 shows the internal architecture of the web application server. The request made by the user is the http request. This request is handled by the web server and the request is sent over to the front controller. The front controller sends the authorized request to the router which sends the request to the desired MVC controller and the data obtained is given to the view. The view renders the request to the server.

Server Internal Architecture:

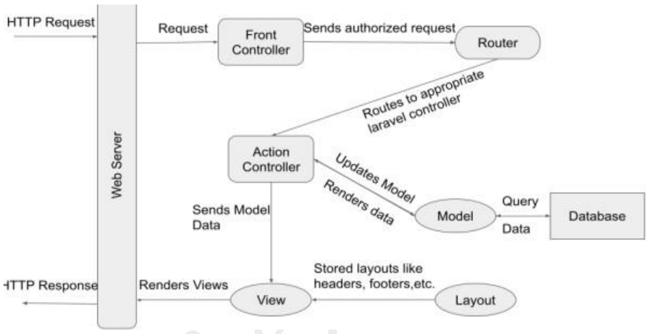


Figure 3.2. Internal architecture diagram of Web application server

Data Model:

Figure 3.3 represents a data flow diagram i.e., E R diagram of a Health Record Management System. The main primary entity is the staff and patients. The staff is generalized into Doctor, Admin and Non- Medical Staff. The primary key for the Staff is their employee_id whereas for the doctors it is doctor _id which is an integer. The other attributes included are the name, phone number, qualifications, profile image, gender, address and the role of the employee as doctor, non-medical staff and the admin.The patient database table has the primary key attribute as aadhar number with other attributes. The patient database table is connected to the appointment table. The appointment table is connected to the patient table with a one-to-many relationship with many towards the patient. The appointment table is connected to the doctor table with a one-to-many relationship towards appointment.

The medicine table is actually the collection of the medicines from which the medicines need to be prescribed. Similarly, the test table consists of the test name and their basic details. The patient books the appointment whose details get stored in the appointment table. The appointment table is connected to the prescribed_test table and prescribed_medicine to ensure the test and medicine assigned to the patient by a specific doctor.

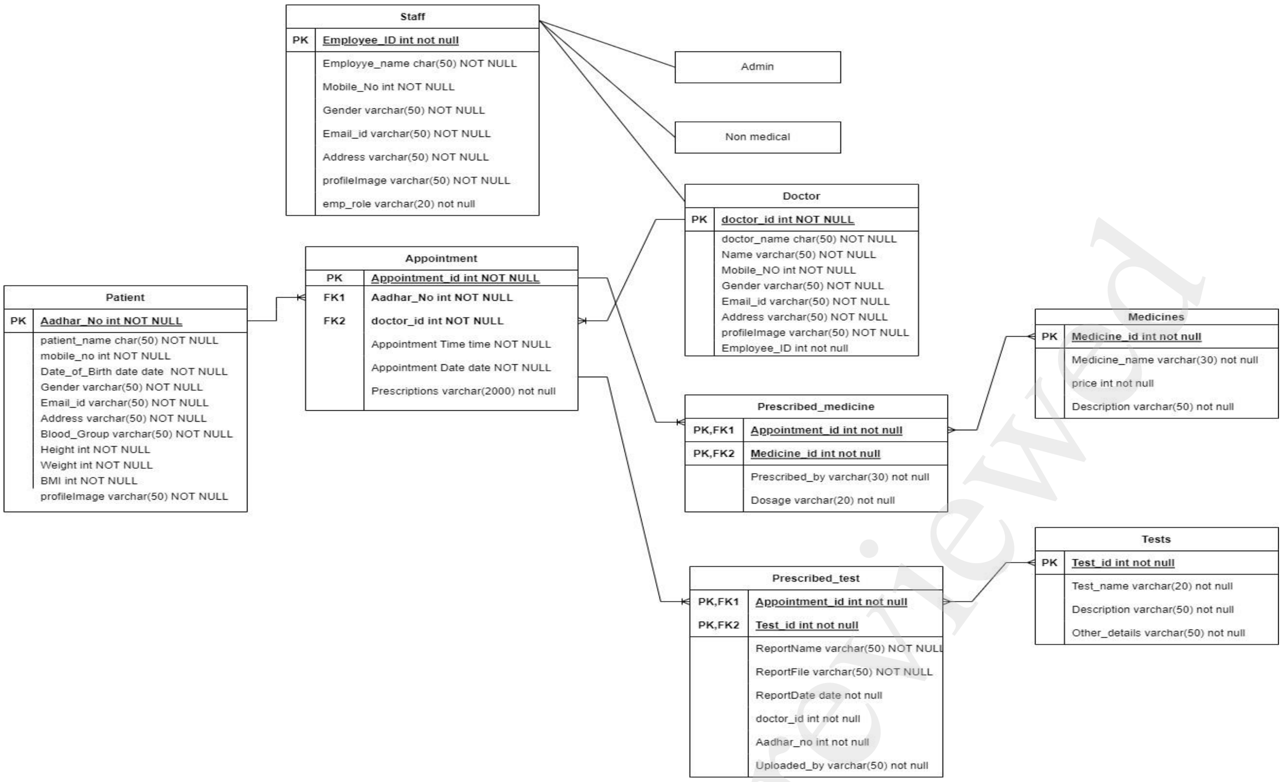


Figure 3.3 : Data flow diagram

4. Methodology

- **Technology Stack**
  Web Server-Apache
  Backend-PHP
  Database-MySQL
  Frontend-HTML, CSS, JavaScript
  Framework- Laravel
  Views and Template-Blade
  ORM-Eloquent
- **Hardware Details**
  Processor- i3 or above
  RAM-4 GB minimum
  Display-Super VGA with a resolution of 1024*768

5. Result

Figure 5.1 : Homepage of Health Record Management System

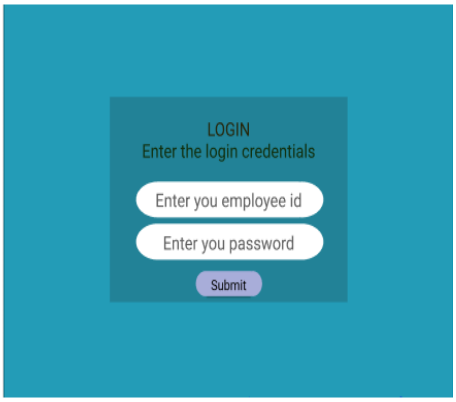


Figure 5.2: Login page for medical staffs

Figure 5.3 : View appointment page for doctor

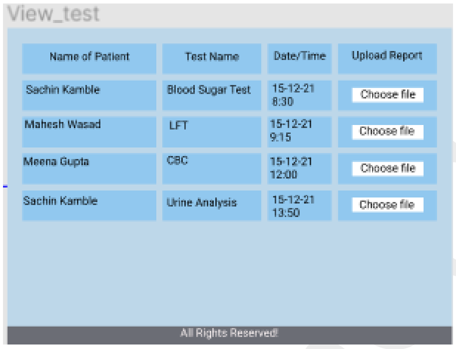


Figure 5.4 : View report page for pathology unit

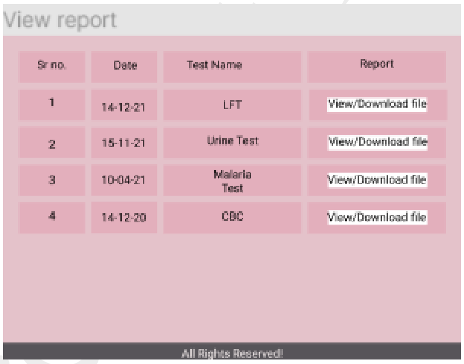


Figure 5.5 : View report page for patient

Figure 5.6 : View profile page for doctor

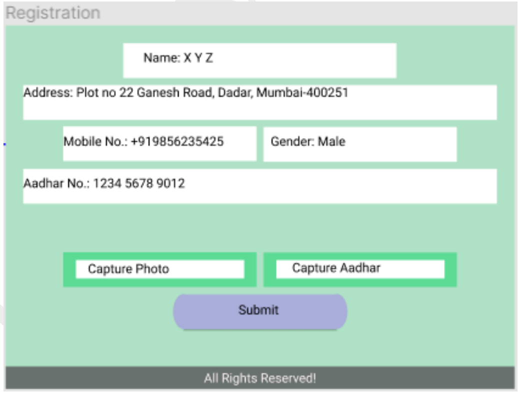


Figure 5.7 : Registration page of the new patient

Figure 5.8 : Patient profile page

6. Conclusion and Future Scope

Thus, we have implemented a system that will maintain track of all patient's data of the hospital. It will allow the health providers most probably the doctors and nurses to easily access the data they want pertaining a given patient without spending an excessive amount of time searching it amongst different documents saved withinside the stores of the hospital. The system will ease retrieval of the patient's data if wished by the health service providers while not having to struggle with boxes in the store room. Therefore handling the patient's data is simplified by this system. The manual system will get replaced by the brand new proposed system which saves time, it's far more efficient and might work under all situations so long as there's power around and access of patients' data is therefore simplified.

The future scope for the above system consists of integrating the above system with the block chain to enhance the security features of the system. Extending the concepts, the system shall be integrated with the various AI algorithms based on the datasets to provide the effective predictions related to the health related data.

7. Acknowledgement

I would like to convey my special thanks to the project guide professor and project coordinator of Information Technology Dr. Varsha Bodade who gave us the good opportunity to do this amazing project on the Health Record Management System, which also helped us in doing a lot of research and providing us with all the resources that were required for this project.

8. References

[1] Shahnaz, Ayesha, Usman Qamar, and Ayesha Khalid. "Using blockchain for electronic health records." *IEEE Access* 7 (2019): 147782-147795.

[2] Rajput, Ahmed Raza, Qianmu Li, Milad Taleby Ahvanooey, and Isma Masood. "EACMS: Emergency access control management system for personal health records based on blockchain."

IEEE Access 7 (2019): 84304-84317.

[3] Harshini, V. M., Shreevani Danai, H. R. Usha, and Manjunath R. Kounte. "Health record management through blockchain technology." In *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 1411-1415. IEEE, 2019.

[4] Thwin, Thein Than, and Sangsuree Vasupongayya. "Blockchain-based access control model to preserve privacy for personal health record systems." *Security and Communication Networks*
2019 (2019).

[5] Latha, N. Anju, B. Rama Murthy, and U. Sunitha. "Electronic health record." *International Journal of Engineering* 1, no. 10 (2012): 25-27.

[6] Sanders, David S., Sarah Read-Brown, Daniel C. Tu, William E. Lambert, Dongseok Choi, Bella M. Almario, Thomas R. Yackel, Anna S. Brown, and Michael F. Chiang. "Impact of an electronic health record operating room management system in ophthalmology on documentation time, surgical volume, and staffing." *JAMA ophthalmology* 132, no. 5 (2014): 586-592.

[7] Mahore, Vinay, Priyanshi Aggarwal, Nitish Andola, and S. Venkatesan. "Secure and Privacy Focused Electronic Health Record Management System using Permissioned Blockchain." In *2019 IEEE Conference on Information and Communication Technology*, pp. 1-6. IEEE, 2019.

[8] Roehrs, Alex, Cristiano André da Costa, Rodrigo da Rosa Righi, Valter Ferreira da Silva, José Roberto Goldim, and Douglas C. Schmidt. "Analyzing the performance of a blockchain-based personal health record implementation." *Journal of biomedical informatics* 92
(2019): 103140.

[9] Yang, Guang, and Chunlei Li. "A design of blockchain-based architecture for the security of electronic health record (EHR) systems." In *2018 IEEE International conference on cloud computing technology and science (CloudCom)*, pp. 261-265. IEEE, 2018

[10] Mohammad rahman, Ibrahim khalil, Xun yi, Abdelaziz Bouras, "A Novel Architecture for Tamper Proof Electronic Health Record Management System using Blockchain Wrapper", 2 july 2019

[11] Gaganjeet singh reen, Manasi mohandas, S. venkatesan, "Decentralized Patient Centric E-Health Record Management System using Blockchain and IPFS", 8 december 2019.

[12] R. Amrita, M.azhagiri, R. Aparna, B. Jashmita, "Secured Electronic Health Record Management System", 15 october 2018.

Appendix

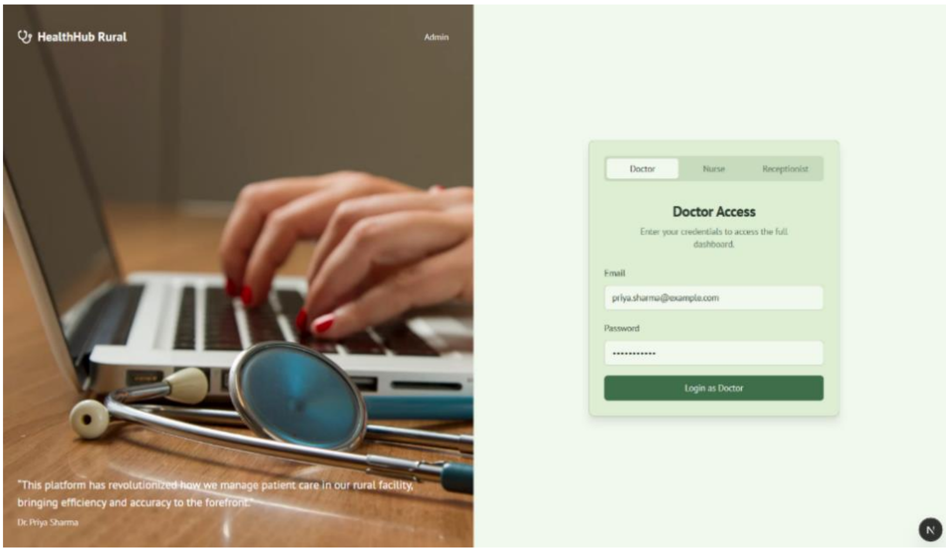


Fig .i Role-based login page

The login screen shown in Fig.i showcases role-based authentication for healthcare staff, with selectable tabs for Doctor, Nurse, and Receptionist. The UI design is intuitive, ensuring that each role gains secure access to their respective dashboards and functionalities. The background image and quote reinforce the platform's positive impact on rural patient care.

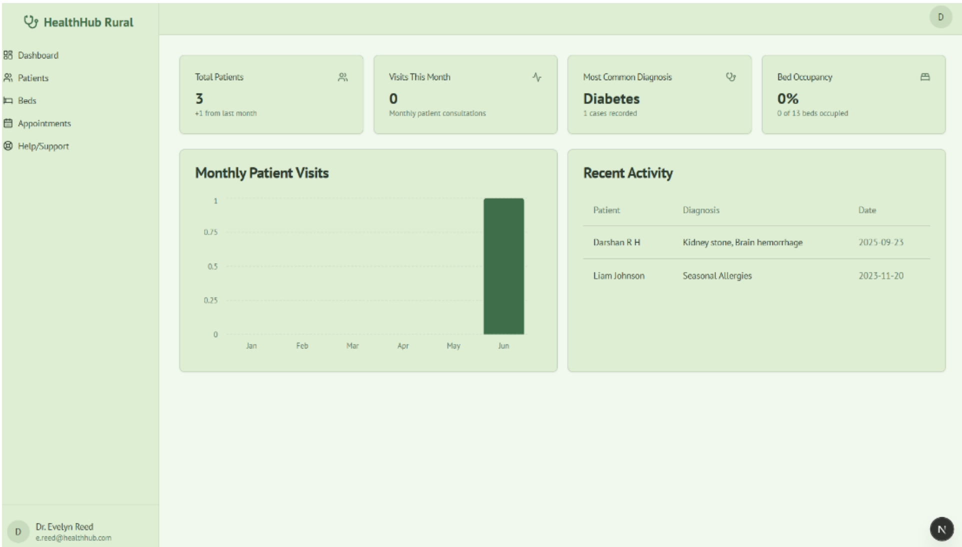


Fig .ii Doctor dashboard

The doctor's dashboard as in Fig.ii summarises key clinical metrics such as total patients, monthly visits, bed occupancy, and common diagnoses. The dashboard also lists recent patient activity, helping clinicians quickly monitor case histories (e.g., kidney stone, brain haemorrhage) and maintain effective oversight of patient management within the facility.

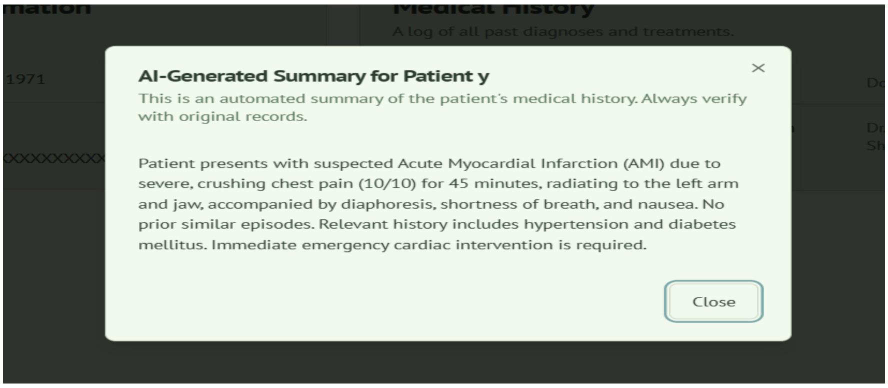


Fig .iii AI-generated report summary

The image Fig.iii displays an automatically produced summary for a patient with heart problems. It tells the story in simple language, mentioning strong chest pain, trouble breathing, and a need for urgent care. The box points out that the patient has past issues including diabetes and high blood pressure. With clear summary text and direct advice ("emergency cardiac intervention required"), even non-technical users or busy doctors can quickly understand what is wrong and what action to take, without having to go through long reports.

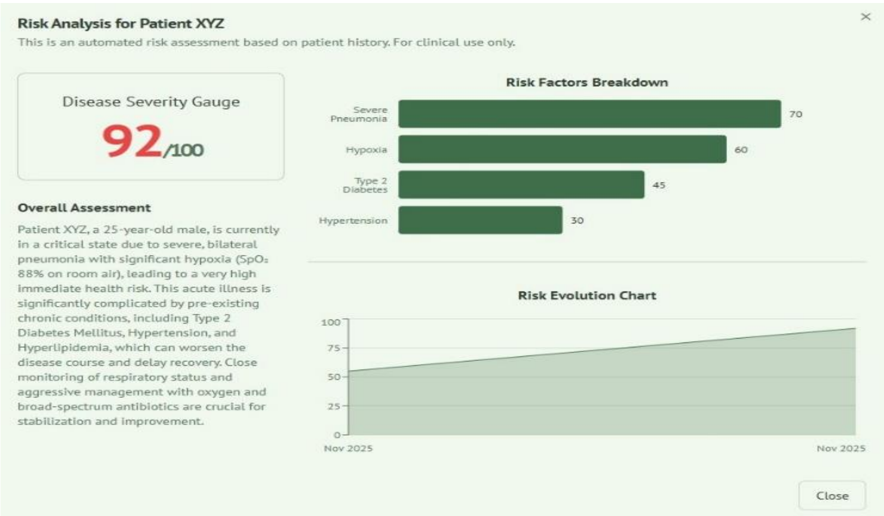


Fig .iv Risk Analysis Summary of Patient 1

The image Fig .iv highlights the patient's automated risk report. Here, the Disease Severity Gauge is much higher (92/100), showing much greater danger. The bar chart makes it obvious that the main problems are pneumonia and low oxygen levels, plus diabetes and hypertension. There's also a summary paragraph, written simply, that explains the situation and what's being done about it. Finally, the risk is tracked over time on a graph, showing how things have gotten worse. The combination of scores, risk factors, and time-based charts helps doctors notice which patients need urgent attention first.

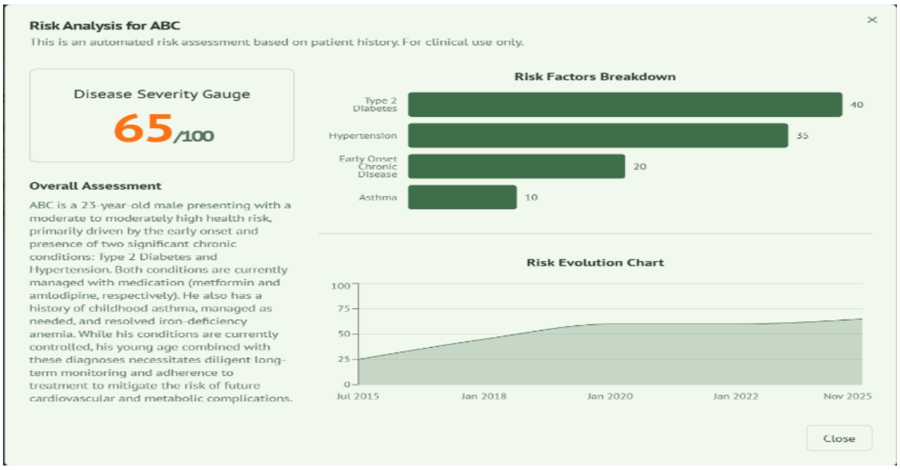


Fig .v Risk Analysis Summary of Patient 2

This image Fig .v shows a sample automated risk analysis for a patient. There's a big number in the center called the "Disease Severity Gauge," which helps doctors quickly know how risky the patient's condition is right now (here, the score is 65 out of 100). Next to it, a bar chart breaks down the main health problems that are increasing risk, like diabetes and high blood pressure. Beneath this, there's a clear written summary, explaining the patient's main medical history and why they are considered at higher risk. At the bottom, you can see a line chart showing how this patient's risk score has changed over the years. This visual approach makes it easy for health staff to understand a patient's situation at a glance and decide what to do next.

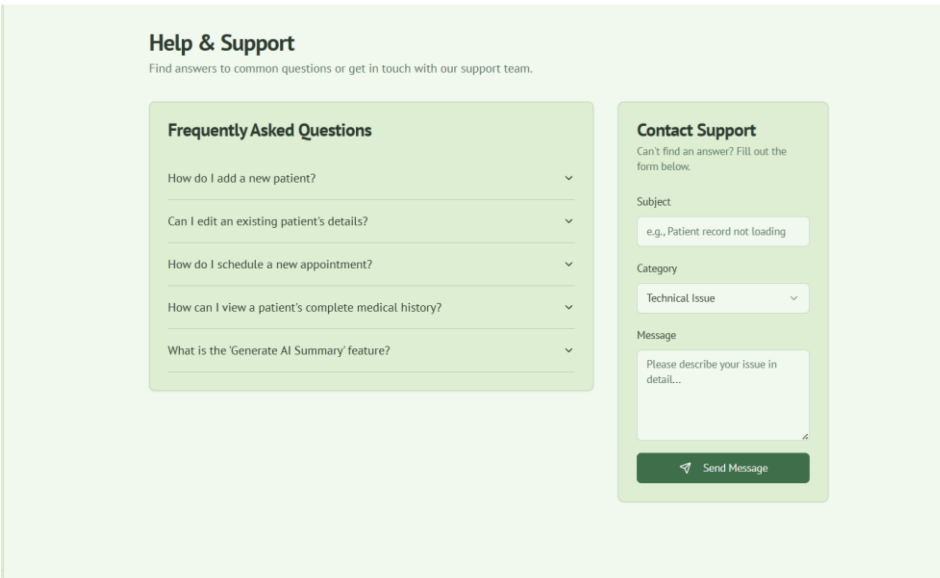
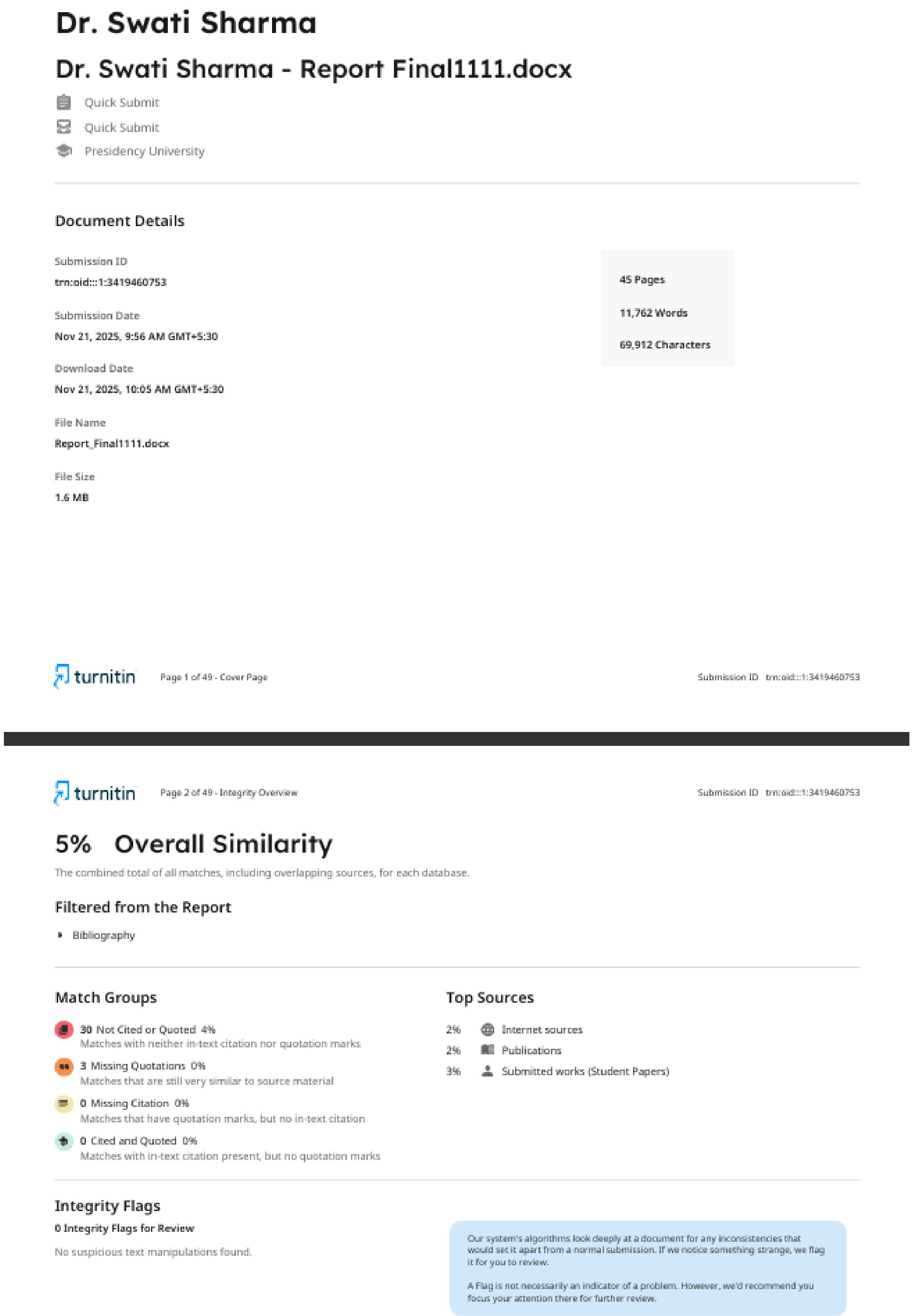


Fig .vi Help and Support panel

In the Fig .vi This Help & Support page offers a convenient FAQ section to address common user questions, like adding patients or understanding the AI summary feature. It also provides a contact form for users to report technical issues or seek personalised assistance directly from the support team. The design ensures users can quickly resolve problems and get guidance without navigating away from the main dashboard.



Dr. Swati Sharma

Dr. Swati Sharma - Report Final1111.docx

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID
trn:oid:::1:3419460753

Submission Date
Nov 21, 2025, 9:56 AM GMT+5:30

Download Date
Nov 21, 2025, 10:05 AM GMT+5:30

File Name
Report_Final1111.docx

File Size
1.6 MB

45 Pages

11,762 Words

69,912 Characters

turnitin Page 1 of 49 - Cover Page Submission ID trn:oid:::1:3419460753

turnitin Page 2 of 49 - Integrity Overview Submission ID trn:oid:::1:3419460753

5% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

30 Not Cited or Quoted 4%
Matches with neither in-text citation nor quotation marks

3 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

2% Internet sources

2% Publications

3% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Fig .vii AI Similarity Check Report

Dr. Swati Sharma

Dr. Swati Sharma - Report Final1111.docx

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID
trn:oid:::1:3419460753

Submission Date
Nov 21, 2025, 9:56 AM GMT+5:30

Download Date
Nov 21, 2025, 10:05 AM GMT+5:30

File Name
Report_Final1111.docx

File Size
1.6 MB

45 Pages

11,762 Words

69,912 Characters

turnitin Page 1 of 47 - Cover Page Submission ID trn:oid:::1:3419460753

turnitin Page 2 of 47 - AI Writing Overview Submission ID trn:oid:::1:3419460753

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

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.

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 (i.e., our AI models may produce either false positive results or false negative results), 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.

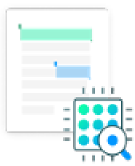


Fig .viii Plagiarism Check Report