

A course Project Report on

Smart Healthcare Portal

### Minor Project (22ECSW301)

Submitted by

|  |  |  |
| --- | --- | --- |
| **Team Number: A-09** | | |
| **Name** | **Roll no** | **SRN** |
| Abhishek Angadi | 03 | 02FE22BCS005 |
| Prem V | 23 | 02FE22BCS072 |
| Soukhya Nayak | 47 | 02FE22BCS148 |
| Sharanamma Katti | 36 | 02FE22BCS119 |

Under the guidance of:

Prof. Amey M.

Faculty In-charge:

Dr. Rajashree Khanai

Department of Computer Science and Engineering

KLE Technological University’s

Dr. M. S. Sheshgiri College of Engineering and Technology, Belagavi – 590 008

pg. 1



## DECLARATION

We hereby declare that the matter embodied in this report entitled **“Smart Healthcare Porta”** submitted to **KLE Technological University** for the course completion of **Minor Project in the 6th Semester of Computer Science and Engineering** is the result of the work done by us in the Department of Computer Science and Engineering, **KLE Dr. M. S. Sheshgiri College of Engineering, Belagavi**, under the guidance of **Prof. Amey M.**, Department of Computer Science and Engineering .We further declare that, to the best of our knowledge and belief, the work reported herein doesn’t form part of any other project on the basis of which a course or award was conferred on an earlier occasion by any other student(s). Also, the results of the work are not submitted for the award of any course, degree, or diploma within this or in any other University or Institute. We hereby also confirm that all of the experimental work in this report has been done by us.

Place: Belagavi

Date: June 3 , 2 0 2 5

Abhishek Angadi Prem V

(02FE22BCS005) (02FE22BCS072)

Soukhya Nayak Sharanamma Katti

(02FE22BC148) (02FE22BCS119)



## CERTIFICATE

This is to certify that the project entitled **“Smart Healthcare Portal”**, submitted to **KLE Technological University’s Dr. M. S. Sheshgiri College of Engineering and Technology (Dr. MSSCET), Belagavi**, for the partial fulfilment of the requirements for the course **Mini Project**, is a Bonafide record of the work carried out by **Abhishek A, Prem V, Soukhya N and Sharanamma K**, students in the Department of Computer Science and Engineering, **KLE Technological University’s Dr. MSSCET, Belagavi**, under my supervision. I further certify that the contents of this report, in full or in part, have not been submitted to any other institute or university for the award of any other course or degree.

Place: Belagavi

Date: June 3, 2 0 2 4

Prof. Amey M. Dr. Rajashree Khanai

(Project Guide) (Course Coordinator)

Dr. Rajashri Khanai (Head of Department)

# ABSTRACT

The Smart Healthcare Portal is a web-based system that uses Machine Learning (ML) and Deep Learning (DL) for early detection and personalized management of heart disease and brain stroke. It employs Logistic Regression for risk prediction and CNNs to analyze ECG data. Users can securely store, track, and download medical records, with a cloud-based platform ensuring data privacy and accessibility. The system includes an AI-powered chatbot for instant support. Built with React.js, Node.js, and MongoDB, it offers a responsive interface and scalable backend. The portal empowers patients with real-time insights, promotes preventive care, and supports chronic disease management.

# CONTENTS

|  |  |
| --- | --- |
| LIST OF TABLES | 6 |
| LIST OF FIGURES | 6 |
| 1. INTRODUCTION    1. BACKGROUND    2. PROBLEM STATEMEN       1. OBJECTIVES | 8  8  8  8 |
| 2.LITERATURE SURVEY | 10 |
| 1. DESIGN SPACE    1. UNDERSTANDING THE PROBLEM       1. PROBLEM SPACE       2. STATE OF ART       3. INTERNAL EXTERNAL FACTORS    2. PROBLEM FLOW       1. SOLUTION SPACE       2. FEATURES       3. MODULES       4. RELATIONSHIP BETWEEN MODULES    3. INTERFACE DESIGNING    4. INCLUSIVE DESIGNING       1. SOCIETY       2. END USERS       3. PRIVACY AND SECURITY | 11  11  11  11  12  12  13  13  14  16  17  17  17  17 |
| 1. REQUIREMENTS ENGINEERING    1. ACTORS    2. GENERAL REQUIREMTS       1. INFRASTRUCTURE       2. COMPETITOR    3. FUNCTIONAL AND NON- FUNCTIONAL REQUIREMENTS       1. FUNCTIONAL REQUIREMENTS       2. NON-FUNCTIONAL REQUIREMENTS   4.4USE CASE DIAGRAM | 19  19  20  21  22  23  23  24 |

|  |  |
| --- | --- |
| 1. SYSTEM MODELING    1. UML DIAGRAMS       1. CONTEXT DIAGRAMS       2. ACTIVITY DIAGRAMS    2. MODEL ARCHITECTURE | 26  27  28  28 |
| 1. IMPLEMENTATION    1. ALGORITHM DEVELOPMENT    2. USER INTERFACE DESIGN | 31  32  32 |
| 1. TESTING    1. RESULTS    2. TESTING REPORT    3. TESTING TOOL | 35  35  36  37 |
| RESULTS AND OUTCOMES  CONCLUSION REFERENCES | 39  42  42 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| V. | Fig. 5.1 UML Diagram | 26 |
| V. | Fig. 5.4 Activity Diagram | 28 |
| VI. | Fig. 6.1 User Interface | 34 |
| VII. | Fig. 7.1 Testing | 35 |
|  | Fig 7.2 History |  |
|  | Fig 7.3 Results Of Predictions |  |
|  | Fig 7.4 Prediction Types |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| VII. | Fig. 7.2 Running Python Script | 37 |
| VII. | Fig. 7.3 Checking the Iptables Rules | 37 |
| VII. | Fig. 7.4 Adding the Iptables rules | 38 |
| VII. | Fig. 7.5 Pie chart | 38 |

* 1. **BACKGROUND**

# CHAPTER 1 INTRODUCTION

Rising heart disease and stroke cases demand early diagnosis, but many lack access to checkups and prediction tools. Traditional systems miss real-time analysis and easy report access. Advances in machine learning and cloud technology enable smarter healthcare platforms that empower users to monitor health and make informed decisions.

## PROBLEM STATEMENT

## Early detection of heart disease and brain stroke is crucial for timely and effective treatment, but many patients experience delays due to the unavailability of accessible diagnostic tools and fragmented medical records. Additionally, existing healthcare systems often lack efficient health monitoring, making it difficult for patients to track their medical history, access past reports, and obtain prompt medical advice. This highlights the urgent need for a comprehensive, secure, and intelligent healthcare solution that leverages machine learning to accurately predict diseases, provides personalized treatment recommendations, and offers centralized management of medical records to ensure seamless access and improved patient care

## OBJECTIVES

## To develop a secure and user-friendly web interface that enables users to input symptoms and receive accurate health predictions with personalized treatment suggestions.

## To integrate cloud storage for users to securely view, download, and manage their medical reports, ensuring data privacy and availability.

## To implement a health prediction history system that allows users to track and compare their past and current health predictions for enhanced medical insights.

## To provide an AI-based chat support system that offers instant responses and guidance based on prediction results and user inputs.

1

## CHAPTER 2 LITERATURE SURVEY

[1] A. Reddy et al., 2023 proposed a hybrid model combining Logistic Regression and Decision Tree Ensemble for cardiovascular risk assessment using the Kaggle Heart Disease Dataset. The model achieved 91.2\% accuracy but focused solely on heart disease and lacked cloud integration.

[2] P. Sharma et al., 2022 designed a secure cloud storage system for electronic health records (EHR) using AES encryption combined with blockchain technology. Tested on a custom cloud dataset, it ensured 100\% data integrity and secure access but did not incorporate predictive healthcare features.

[3] N. K. Kumar et al., 2021 presented a machine learning approach for heart disease risk prediction using Random Forest, Support Vector Machine (SVM), Logistic Regression, and K-Nearest Neighbors (KNN) on the UCI Heart Disease Dataset. Random Forest achieved an accuracy of 85.71\%. The approach is limited to static datasets and lacks real-time data integration.

[4] Bharti et al., 2021 proposed a deep learning model combined with Isolation Forest preprocessing for heart disease prediction using the UCI Heart Disease Dataset. The model achieved 94.2\% accuracy but suffers from high computational requirements and lacks real-time monitoring capabilities.

[5] Md. Touhidul Islam et al., 2021 developed an early prediction model for heart disease using Principal Component Analysis (PCA) for feature reduction and a Hybrid Genetic Algorithm combined with k-Means clustering on the UCI Heart Disease Dataset. This model reached 94.06\% accuracy but faces risks of overfitting and is limited to specific datasets.

[6] P. Theerthagiri et al., 2021 employed Recursive Feature Elimination (RFE) with Gradient Boosting Classifier for car \vspace{10cm} diovascular disease prediction using the UCI Heart Disease Dataset. The model achieved 89.7\% accuracy but requires careful feature selection and may have limited generalization ability.

[7] N. Gupta et al., 2021 developed an IoT and machine learning-enabled health monitoring system integrating IoT sensors with Decision Tree models for anomaly detection on real-time sensor data. The system provided timely alerts on abnormal conditions but is constrained by power dependency and limited scalability.

[8] M. Muibideen et al., 2020 introduced a fast heart disease prediction algorithm using a Bayesian Network model on the UCI Heart Disease Dataset, achieving 85\% accuracy. The study lacks comparison with newer machine learning models.

[9] S. Mehta et al., 2020 applied deep learning with Convolutional Neural Networks (CNN) on EEG signals for stroke detection using the TUH EEG Corpus dataset, achieving 85.4\% accuracy. Limitations include a small dataset and high computational demand.

[10] L. Zhang et al., 2019 conducted a study predicting stroke from clinical data using Naive Bayes, KNN, and XGBoost algorithms on the UCI Stroke Dataset. XGBoost achieved an accuracy of 87.3\%, but the system lacked real-time monitoring and personalized care capabilities.

1

# CHAPTER 3 DESIGN SPACE

## UNDESTANDING THE PROBLEM

## The core problem addressed by the Smart Healthcare Portal revolves around the growing incidence of heart disease and brain stroke, which are often diagnosed late due to the lack of accessible and timely diagnostic tools. In many cases, patients do not have access to regular health checkups or centralized digital records, making it difficult to monitor their health conditions over time. Traditional healthcare systems are fragmented, with limited support for real-time disease prediction and medical data management, especially in rural or underserved regions. This results in delayed medical intervention, poor continuity of care, and increased mortality risk. The Smart Healthcare Portal aims to solve this by offering a secure, web-based platform that uses machine learning and deep learning models for early disease detection, provides personalized treatment suggestions, and enables users to manage their medical reports and prediction history through a user-friendly interface.

## PROBLEM SPACE

## The healthcare sector faces significant challenges in the early diagnosis of heart disease and brain stroke, largely due to limited access to routine checkups and advanced diagnostic tools for many individuals. Existing healthcare systems are often fragmented, leading to poor integration of patient data across platforms, which hinders effective diagnosis and treatment. Additionally, there is a lack of real-time health prediction and monitoring capabilities, making it difficult for patients and healthcare providers to respond proactively to emerging health issues. Patients also struggle to track their medical history and access past reports easily, while traditional systems fall short in offering personalized treatment suggestions tailored to individual needs. Accessibility remains a major concern, especially in remote and rural areas where healthcare infrastructure is limited. Moreover, current platforms frequently overlook the benefits of cloud-based secure storage and seamless sharing of medical data. The minimal integration of AI-powered assistants further restricts patient support and engagement. To address these challenges, a smart, unified healthcare solution is essential—one that leverages advanced technology to make healthcare more timely, accessible, predictive, and personalized for all patients.

## STATE OF ART

## The state of the art in smart healthcare integrates machine learning, deep learning, and cloud technologies to improve early disease detection and patient care. Recent models use algorithms like Logistic Regression, Random Forest, and CNNs for accurate prediction of heart disease and brain stroke. CNNs have shown high accuracy in analyzing ECG signals, while logistic regression is effective for symptom-based assessments. Cloud platforms enable secure, scalable storage and remote access to medical records. Some systems also employ blockchain for data integrity. However, most existing solutions focus on a single condition, lack real-time feedback, or do not support personalized treatment. Many also fail to offer user-friendly interfaces or integrated health monitoring. AI chatbots are emerging but are often separate from diagnostic tools. Despite these advances, gaps remain in accessibility, interaction, and data centralization. The Smart Healthcare Portal addresses these issues by unifying prediction models, cloud storage, and an AI chatbot in one responsive platform. It ensures real-time predictions, personalized care, and secure medical data management in a single solution

1

for resource-constrained environments, and AI-driven models that adapt to evolving protocols, ensuring real- time efficiency and accuracy

## INTERNAL EXTERNAL FACTORS

### Internal Factors:

### In the context of the Smart Healthcare Portal project, internal factors refer to the elements within the control of the development team, such as technical skills, choice of technology stack, and project management efficiency. The team’s proficiency in React.js, Node.js, and machine learning played a significant role in building a responsive and intelligent healthcare system. The integration of secure cloud storage, user-friendly design, and AI-based chatbot support also reflect strong internal planning and execution. On the other hand, limitations in dataset diversity, computational resources, and dependence on user-provided data may represent internal challenges that could impact prediction accuracy and scalability.

### External Factors:

External factors include influences from the outside environment that affect the project’s success. These encompass the increasing global demand for digital healthcare solutions, the rise in heart and stroke cases, and advancements in cloud and AI technologies, which create opportunities for the portal's relevance and adoption. However, external challenges such as user data privacy regulations, internet accessibility in rural areas, and the need for medical validation of AI predictions pose potential threats. Both internal and external factors must be continuously addressed to enhance the effectiveness and adoption of the Smart Healthcare Portal.

## PROBLEM FLOW

## The Smart Healthcare Portal addresses the pressing issue of delayed diagnosis and disorganized medical records in current healthcare systems. Many individuals, especially in underserved regions, lack access to routine checkups and real-time disease prediction tools, which leads to late detection of heart disease and brain stroke. The system aims to provide a centralized, intelligent platform that empowers users to monitor their health, receive early predictions, and access medical reports securely. It features a user-friendly web interface where users can input symptoms or ECG data. Machine Learning models, including Logistic Regression and CNN, analyze this data to predict risks and provide personalized treatment recommendations. All health data and prediction history are stored securely on cloud-based databases. Users can track their medical progress through downloadable reports and real-time result displays. The portal is built using React.js for the frontend and Node.js with Express for the backend. MongoDB Atlas is used for data storage, while deployment on cloud services like AWS ensures scalability. An AI-powered chatbot further enhances user experience by offering instant support and health advice.

## SOLUTION SPACE

* To develop a secure and user-friendly web interface that allows users to input symptoms and receive accurate health predictions.
* To offer personalized treatment suggestions based on prediction outcomes using machine learning and CNN models.
* To integrate cloud storage enabling users to securely view, download, and manage their medical reports with data privacy and availability.
* To implement a health prediction history system for users to track and compare past and current predictions, aiding in informed health decisions.
* To provide an AI-based chat support system that offers instant responses, guidance, and assistance based on user inputs and prediction results.

## 3.2.1.1 FEATURES

## 🧠 Real-Time Disease Prediction

## Predicts heart disease and brain stroke using Logistic Regression and CNN models (for ECG data).

## 📄 Medical Report Management

## Users can view, download, and track their prediction history and reports through a secure portal.

## 💬 AI-Powered Chatbot

## Provides instant health advice and guidance, enhancing the overall user experience.

## ☁ Cloud-Based Data Storage

## Secure storage using MongoDB Atlas and deployment on AWS/Firebase, ensuring data availability and privacy.

## 🔐 Secure and User-Friendly Interface

## Developed with React.js and Node.js, offering a responsive design, JWT-based authentication, and HTTPS security.

1

## MODULES

## 1. User Interface (Frontend)

## Purpose: Let users register, input symptoms, upload ECG, and view results.

## How it works: Provides forms, displays predictions & history, and integrates chatbot.

## Significance: Enables easy user interaction and real-time health monitoring.

## 2. Server (Backend)

## Purpose: Manage data flow, authentication, and ML integration.

## How it works: REST APIs with JWT auth; connects frontend with ML services.

## Significance: Securely processes requests and handles prediction logic.

## 3. Machine Learning Module

## Purpose: Predict heart disease and stroke from symptoms and ECG data.

## How it works: Uses Logistic Regression and CNN models for analysis.

## Significance: Automates accurate and early diagnosis.

## 4. Database

## Purpose: Store user data, prediction records, and chatbot logs.

## How it works: MongoDB Atlas saves profiles, reports, and history.

## Significance: Ensures data persistence and easy retrieval.

## 5. Cloud Integration

## Purpose: Host app and securely store medical files.

## How it works: Uses AWS/Firebase for hosting and S3 for file storage with secure access.

## Significance: Provides scalability, security, and reliable access.

## RELATIONSHIP BETWEENMODULES

## Frontend & Backend

## The Frontend collects user input (symptoms, ECG) and sends it to the Backend via REST API calls.

## The Backend processes requests, manages authentication, and sends responses (predictions, treatment info) back to the Frontend for display.

## Backend & Machine Learning Module:

## The Backend communicates with the ML module (Python microservices) to send user data and receive prediction results.

## This integration allows the Backend to act as a bridge between user requests and AI-powered diagnosis.

## Backend & Database:

## The Backend stores and retrieves all user info, predictions, ECG data, and chatbot logs in the Database.

## This ensures data persistence and history tracking for users.

## Frontend & AI Chatbot:

## The AI Chatbot is integrated into the Frontend for real-time user support.

## Chat interactions are logged and managed through the Backend and saved in the Database.

## Cloud Integration & All Modules:

## The Cloud platform hosts the Frontend, Backend, and stores medical reports securely.

## Cloud storage (e.g., AWS S3) manages files uploaded by users, accessible via Backend with secure URLs.

## INCLUSIVE DESIGNING

* + 1. **SOCIETY**

Inclusive design in society ensures that products and services are accessible to everyone, regardless of their age, ability, or background. It promotes equality by considering diverse needs and removing barriers, enabling all members of society to participate fully and benefit equally from technology and services.

## END USERS

Inclusive design focuses on understanding the varied abilities and preferences of end users. It creates flexible and adaptable solutions that accommodate different physical, cognitive, and sensory needs, ensuring a positive and effective user experience for people with disabilities, older adults, and others with unique challenges.

## PRIVACY AND SECURITY

Inclusive design also prioritizes privacy and security by protecting all users' sensitive information, especially vulnerable groups. It implements strong data protection measures and transparent policies to build trust and ensure thateveryone’s personal data is handled respectfully and safely.

## ACTORS

1

# CHAPTER 4 REQUIREMENTS ENGINEERING

### Network Administrators:

### The Smart Healthcare Portal follows a multi-layered client-server architecture with:

### Frontend: React.js for responsive user interaction.

### Backend: Node.js/Express.js for business logic, APIs, and model invocation.

### Database: MongoDB Atlas (NoSQL) for secure data storage.

### Cloud: AWS/Firebase for deployment and storage with HTTPS access

### End Users:

* Individuals seeking early diagnosis for heart disease and stroke.
* Users with access to ECG data or symptom-based concerns.
* Users wanting to track health records securely.

### System Components:

* Frontend (React.js): User interface for login, input, results display, chatbot interaction.
* Backend (Node.js + Express): API management, logic processing, authentication.
* ML/DL Models (Python): Logistic Regression and CNN for prediction.
* Database (MongoDB Atlas): Stores user profiles, reports, history.
* Cloud (AWS/Firebase): Hosting, file storage, security features..

## GENERAL REQUIREMENTS

* + 1. **INFRASTRUCTURE**
* Cloud Hosting: AWS/Firebase for scalable, secure, and accessible deployment.
* Database: MongoDB Atlas for cloud-based document storage.
* Security Protocols: HTTPS, JWT for authentication, signed URLs for file access.

## COMPETITOR

Advantages: Real-time prediction, integrated cloud storage, AI chatbot, responsive frontend.

Limitations in competitors: Lack of cloud integration, static datasets, no chatbot, or missing real-time analysis.

### FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

* + 1. **FUNCTIONAL REQUIREMENTS**
* User login and registration.
* Input of symptoms or ECG data.
* Real-time disease prediction using ML/DL.
* Downloadable medical reports.
* Chatbot support for assistance.
* Viewing and tracking health history.
* Secure storage and retrieval of user data.

### NON-FUNCTIONAL REQUIREMENTS

### Scalability: Supports many users and growing data volume.

### Reliability: Accurate predictions using validated ML models.

### Security: JWT, HTTPS, cloud-secured storage.

### Performance: Real-time responses from ML models.

### Usability: Simple and responsive UI for non-technical users.

### Maintainability: Modular code using React and Node.js makes updates manageable.

## CHAPTER 5 SYSTEM MODELLING

System modeling refers to the process of creating abstract representations of a system to understand, analyze, and improve its performance or functionality. These models help visualize the system's structure, behavior, and interactions to solve problems or make decisions.

## UML DIAGRAMS

## 

Fig 5.1 UML diagram

2

2

1. In fig 5.1 Class Diagram (UML - Object-Oriented Design)

* This diagram shows the data structure and relationships between different entities in the Smart Healthcare Portal:
* Patient: Central user who has attributes like patientId, name, email, and password.
* Can upload medical reports.
* Can view prediction history.
* Can input symptoms for disease prediction.
* MedicalReport: Represents uploaded reports.
* Attributes: reportId, uploadDate, filePath.
* One patient can upload multiple reports.
* Prediction: Represents a disease prediction result.
* Contains: predictionId, date, diseaseType, result.
* Linked to specific symptom data.
* One patient can have multiple predictions.
* SymptomData: Holds input data for prediction.
* Includes both symptoms (key-value pairs) and ecgData (file upload).

## ACTIVITY DIAGRAM

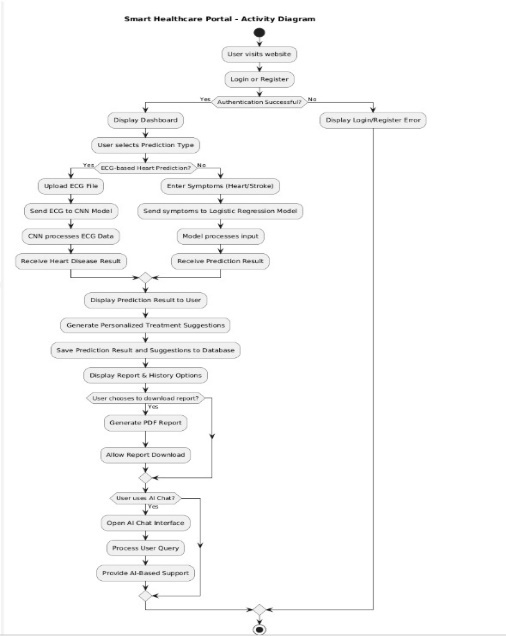


Fig 5.3 Activity Diagram

2

2. In Fig 5.2 Activity Diagram

* User visits portal → Login/Register.
* If successful: Proceed to dashboard.
* Else: Show error.
* Select prediction type:
* ECG-based heart prediction: Upload ECG file → Sent to CNN → Receive result.
* Symptom-based prediction (Heart/Stroke): Enter symptoms → Sent to Logistic Regression model → Receive result.
* System displays prediction result and suggestions.
* Report handling:
* Option to view/download report.
* Generates PDF if chosen.
* AI Chat support:
* If user opts in, chatbot is activated.
* Processes query and offers AI-driven support.

3

## CHAPTER 6 IMPLEMENTATION

The Smart Healthcare Portal is designed to offer a user-friendly interface for patients to manage and monitor their health status, receive disease predictions, view health history, get treatment advice, and access live help or chat support.

1.User Authentication:  
 Users register/login securely using JWT and password encryption.

2.Dashboard:  
Central page providing access to Prediction, Health History, Treatment Advice, and Help Chat.

3.Prediction Module:  
Users input data or upload ECG to predict Heart Disease, Brain Stroke, or ECG-based conditions using ML models.

4.Health History:  
Displays past predictions and allows downloading reports as PDFs.

5.Treatment Advice:  
Provides medical and lifestyle suggestions based on prediction results.

6.Help/Live Chat:  
Offers chatbot or real-time support for user queries

## USER INTERFACE DESIGN

## The Smart Healthcare Portal features a clean, intuitive, and responsive interface designed for ease of use across devices.

## Login/Registration: Simple forms with secure JWT authentication, password encryption, and real-time validation.

## Dashboard: Central hub with quick access cards for Prediction, Health History, Treatment Advice, and

## Prediction Module: User-friendly forms for data input and ECG upload, with clear instructions and instant result display.

## Health History: Organized list of past predictions with options to search, filter, and download reports as PDFs.

## Treatment Advice: Clear, personalized medical and lifestyle suggestions based on prediction results.

## Help/Live Chat: Integrated chatbot and real-time support for user queries, accessible anytime.

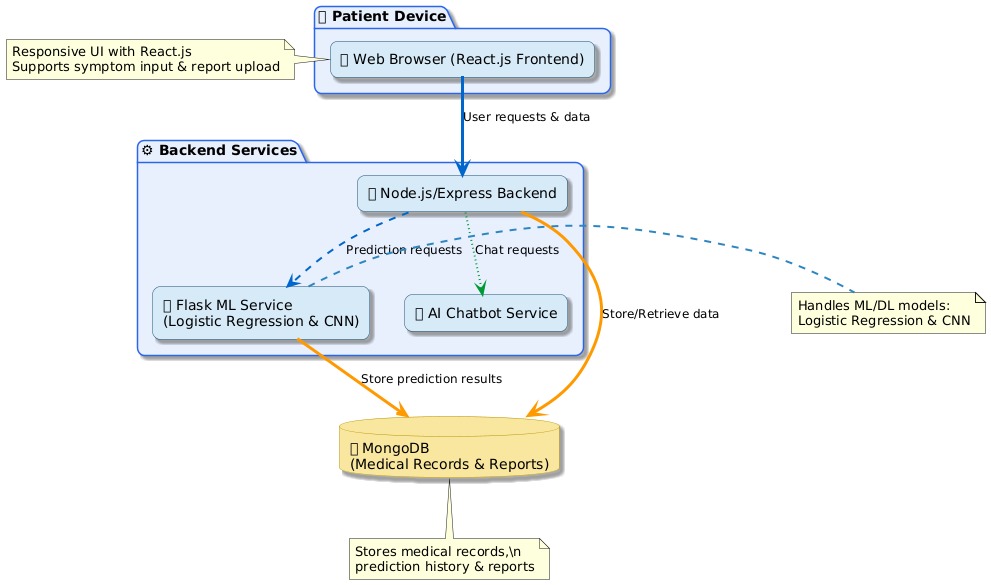


Fig 6.1 User interface design

In Fig. 6.1 The diagram illustrates a UI design for traffic categorization:

Components:

* This diagram shows a medical prediction web app architecture:
* Frontend (React.js) on patient devices lets users input symptoms and upload reports.
* Backend (Node.js/Express) handles requests and routes them to:
* Flask ML Service for predictions using Logistic Regression & CNN.
* AI Chatbot for patient queries.
* MongoDB stores medical records, reports, and prediction results.

## CHAPTER 7 TESTING

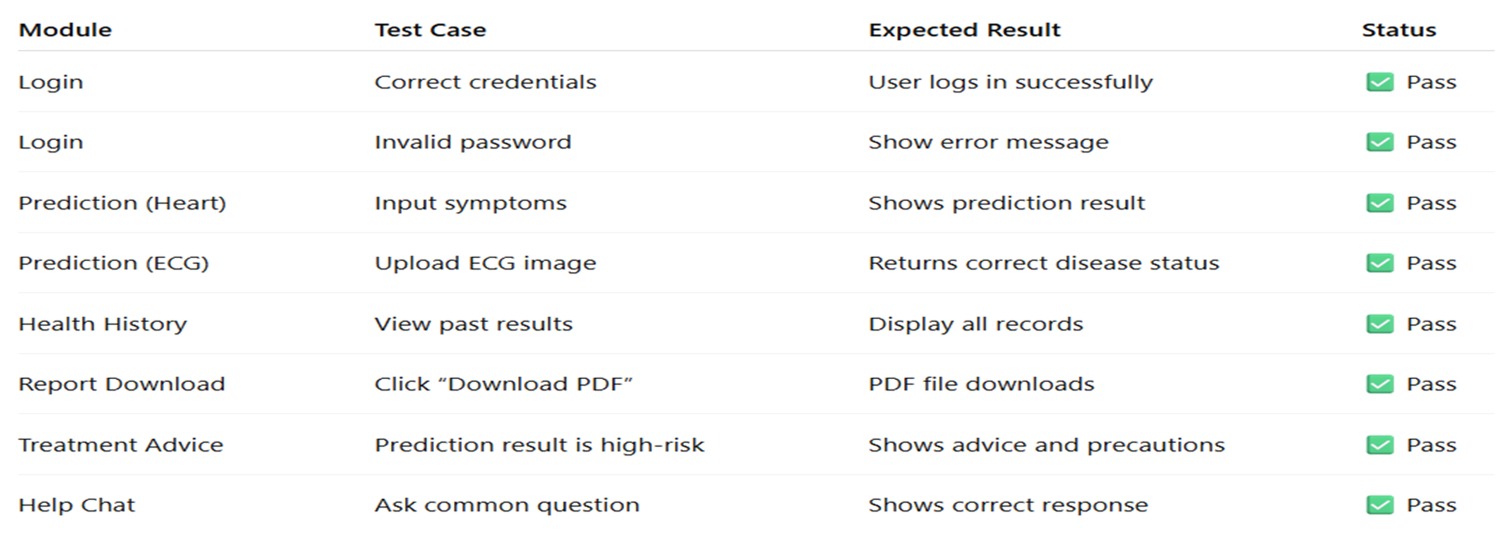


Fig 7.1 Testing

## TESTING TOOL

For this medical prediction web application, the following testing tools can be used:

* + 1. Frontend (React.js)

Jest: For unit and integration testing of React components.

React Testing Library: For simulating user interactions and testing UI behavior.

* + 1. Backend (Node.js/Express)

Mocha + Chai: For unit and API endpoint testing.

Supertest: For HTTP request testing to the Express server.

* + 1. Flask ML Service

PyTest: For unit testing Python functions and ML model outputs.

Postman or cURL: For manually testing API endpoints.

* + 1. Database (MongoDB)

MongoDB Compass: For visual inspection and validation of stored data.

Mockingoose (with Mongoose): For mocking MongoDB in tests.

* + 1. End-to-End Testing

Cypress or Selenium: For full-system testing from frontend to backend including user flows.

## RESULTS AND OUTCOMES

Snapshots of the Project Execution

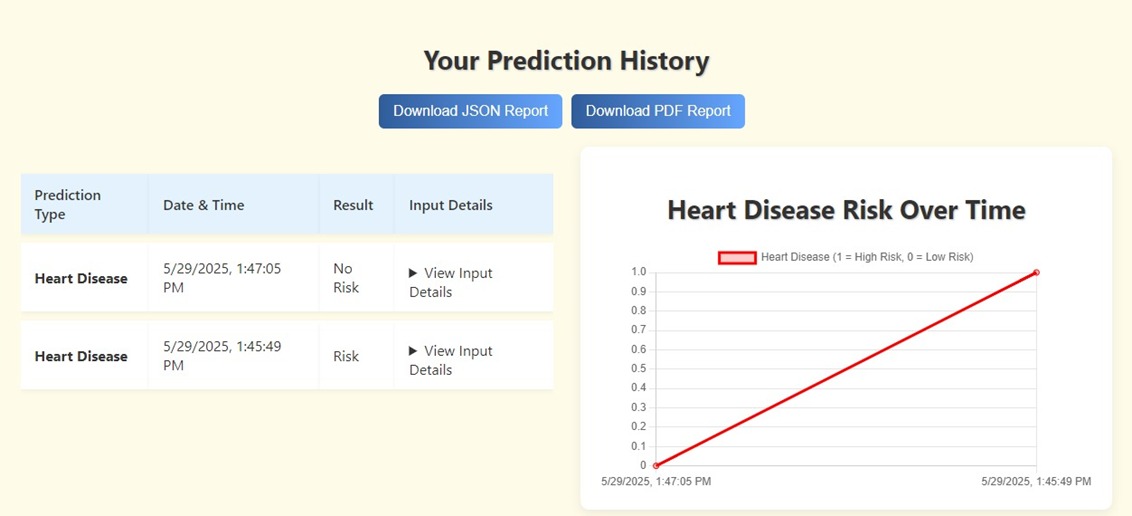


Fig 7.2 History

In Fig. 7.2 Prediction History

* Shows past predictions in a table with:
* Date & Time
* Result (e.g., Risk or No Risk)
* Option to View Input Details
* Includes a graph titled "Heart Disease Risk Over Time" showing risk trends.
* Buttons to Download PDF Report are available.

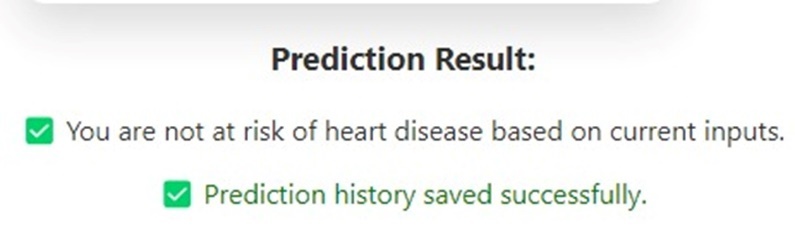


Fig 7.3 Results of prediction

In Fig 7.3 Image 2: Prediction Result

Displays the output of a prediction:

* "You are not at risk of heart disease based on current inputs."
* "Prediction history saved successfully."

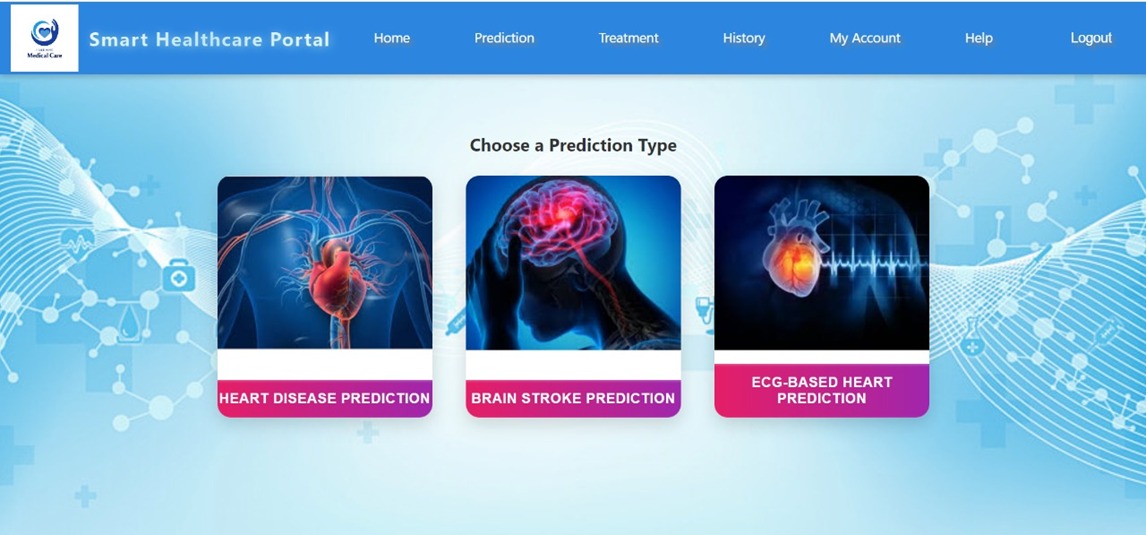


Fig 7.4 Prediction Types

In Fig 7.4 Prediction Type Selection

* This screen shows three prediction options:
* Heart Disease Prediction
* Brain Stroke Prediction
* ECG-Based Heart Prediction
* Users can select one to proceed with a specific health prediction.

## CONCLUSION

The Smart Healthcare Portal effectively combines machine learning models and modern web technologies to provide a reliable platform for early detection and management of heart disease and brain stroke. By utilizing Logistic Regression for risk prediction and a Convolutional Neural Network (CNN) for ECG-based heart disease classification, the system offers accurate, real-time health assessments that can support timely medical intervention.

The integration of a secure cloud-based infrastructure ensures safe storage and easy access to medical records, prediction history, and reports, while the user-friendly React.js frontend enhances accessibility across devices. Additionally, the AI-powered chat support adds an interactive dimension to patient care, offering instant guidance and assistance.

This project addresses critical gaps in healthcare accessibility and monitoring by centralizing diagnostic tools, health data management, and personalized treatment suggestions in a single platform. The Smart Healthcare Portal stands as a scalable and efficient solution that can improve preventive healthcare and empower users to take control oftheirhealthwithconfidence.

4

## REFERENCES

1] A. Reddy et al., “Hybrid Model for Cardiovascular Risk Assessment,”2023.

[2] P. Sharma et al., “Secure Cloud Storage for EHR,” 2022.

[3] N. K. Kumar et al., “Heart Disease Risk Prediction Using ML Classi-fiers,” 2021.

[4] Bharti et al., “Prediction of Heart Disease Using ML and DL,” 2021.

[5] Md. Touhidul Islam et al., “Early Prediction of Heart Disease Using PCA and HGA with k-Means,” 2021.

[6] P. Theerthagiri et al., “Cardiovascular Disease Prediction using RFE and Gradient Boosting,” 2021.

[7] N. Gupta et al., “IoT and ML Enabled Health Monitoring,” 2021.

[8] M. Muibideen et al., “Fast Algorithm for Heart Disease Prediction using Bayesian Network,” 2020.

[9] S. Mehta et al., “EEG-based Stroke Detection,” 2020.

[10] L. Zhang et al., “Predicting Stroke from Clinical Data,” 2019.

[11] World Health Organization. Cardiovascular Diseases (CVDs). 2021. Available at: <https://www.who.int/news-room/fact-> sheets/detail/cardiovascular-diseases-(cvds).

[12] Smith, A., Johnson, T., Miller, K. Gaps in preventive healthcare services. Public Health Reports, 135(2):205–214, 2020.

[13] Johnson, R., Lee, S. Accessibility of healthcare in rural areas: A review.Global Health Journal, 12(1):10–20, 2019.

[14] Chen, L., Wang, H., Zhang, Y. Challenges in healthcare data management. Journal of Medical Informatics, 45(3):123–135, 2018.

[15] Kumar, S., Gupta, R. Logistic regression models for cardiovascular risk prediction. Biomedical Engineering Letters, 10(4):567–574, 2022.