علوم الحاسب ونظم المعلومات
Info Sys & Comp Science

جامعة ٦ أكتوبر
October 6 University

Graduation Project

Predicting Cerebrovascular Stroke Disease Related to Heart Disease

Prepared By:

No.	Student Name	Department
1	Ahmed Yasser Abdullah El Sharkawy	AI
2	Ibrahim Ehab Ibrahim Hassan	AI
3	Ahmed Mohamed Yousef Gad Al Mawla	AI
4	Ahmed Ali Ali Abd Elshafy	AI
5	Ahmed Amr Afte Deghedy	AI
6	Arwa Mohamed Hassan Hassan	CS
7	Sohila Wessam Hamed Refaie	CS
8	Afnan Fekry Mahmoud Hammad	IS

Supervisor:

Assoc. Prof. Hussam Elbehiery

CO-Supervisor:

Eng. Ahmed AbdelMoneim

2025

Acknowledgment

Acknowledgment

Both scientifically and practically, we have put a lot of effort into our project. None of this would have been possible without the support of Allah, our dedicated team of doctors, assistants, and fellow team members.

We would like to express our sincere gratitude to **Prof. Dr. Nabila Mohamed Hassan** for her support throughout the semester.

We extend our sincere thanks to Assoc. **Prof. Dr. Hussam Elbehiery** for his insightful comments, unwavering support, and continuous guidance that played a key role in achieving these results. We are also grateful for the facilities and resources he generously provided throughout the work.

We would also like to acknowledge the time and effort of Assoc. **Prof. Hussam Elbehiery** has invested in reviewing and providing feedback on our work. His constructive criticism and valuable suggestions have significantly contributed to improving the quality and effectiveness of the project.

Special thanks to **Eng. Ahmed AbdelMoneim**, our supervisor in this project, for supporting us. He was very patient with us and acted like a real mentor when we needed him. We have greatly benefited from his knowledge.

Special appreciation goes to the specialists in this topic, **Dr. Hossam Sakr**, for the great help he offered and the huge assistance in guiding us through our project.

List of Contents

List of Contents

List of Abbreviations

List of Abbreviations

Abbreviation	Full Form
AI	Artificial Intelligence
ECG	Electrocardiogram
SPO2	Saturation of Peripheral Oxygen
NLP	Natural Language Processing
ML	Machine Learning
BP	Blood Pressure
CNN	Convolutional Neural Network
LLM	Large Language Model
OLED	Organic Light-Emitting Diode
BLE	Bluetooth Low Energy
ADC	Analog-to-Digital Converter
IoT	Internet Of Things
UART	Universal Asynchronous Receiver/Transmitter

List of Figures

List of Figures

Abstract

Abstract

Healthcare technology plays a crucial role in addressing the critical medical challenges in the modern world. Predicting a stroke and preventing heart disease is becoming very dire globally, considering the effect on mortality rates and a high standard of living. This work applies machine learning in an incorporated mobile application and embedded systems for developing a predictive model to identify high-risk individuals so necessary interventions can be undertaken to prevent diseases.

It combines the latest data analysis with real-time health monitoring; this work links technology with medicine to deliver actionable insights in support of improved health outcomes. The health of the world could be improved with the early diagnosis, prevention, and management of diseases by empowering individuals and enabling healthcare professionals to make sound decisions. The innovative method shows the ability of technologies to change the face of life-threatening ailments.

Chapter one

Introduction

Chapter One: Introduction

1.1 Introduction to Stroke Disease and its relation to Heart Disease

A cerebrovascular stroke, generally called a stroke, is a medical condition where there impedes the flow of blood towards the brain, leading to the death or damage of brain cells. It is among the leading causes of mortality and a cause of prolonged disability in the world, hence deeply affecting individuals and health care. On the other hand, heart diseases are a set of cardiovascular disorders, including coronary artery disease, heart failure, and arrhythmias. These diseases are very much prevalent as the leading cause of death all over the world. In addition, their association with cerebrovascular stroke usually appears together.

Their connection to each other is due to the common risk factors and physiological interactions. Atrial fibrillation is considered a common heart rhythm disorder that increases the danger of ischemic stroke due to clots in blood circulation that travel to the brain. Heart failure and other cardiovascular disorders further worsen the chances of cerebrovascular complications. Understanding such interrelations will be important to enhance preventive strategies and improvements in health outcomes.

1.2 Problem Definition

What is the challenge with stroke prediction today?

Even as medical science has reached far, the prediction and prevention of stroke continues to present a huge challenge. People who are prone to heart diseases remain at a very high risk, but this often goes unnoticed until all is lost.

Why are existing methods incomplete?

The currently available tools for predicting risk mainly work within a narrow circle of clinical factors involving blood pressure and cholesterol while fully excluding the entangled complications of lifestyle and genetics with comorbid conditions. The result of this minimalist approach is that the identification of cases becomes incomplete.

What are the consequences of these gaps?

Poor prediction of stroke risks could result in delayed medical intervention, failure in detection, and even inappropriate treatments, all at the cost of patient harm and overheads in healthcare.

1.3 Existing Solutions for Prediction Stroke Disease and Detection

The role of the different existing solutions and technologies that support the detection, prediction, or self-assessment of a stroke is presented. Some such facilities are discussed here:

1. **Human Interpreters**:

Consulting doctors is the conventional approach. Doctors review patient history, analyze symptoms, and perform diagnostic tests such as MRI or CT scans to estimate stroke possibilities.

2. **Online Platforms**:

Websites or telemedicine platforms allow communication between the patient and the doctor. Users can upload their medical history or test reports for examinations and get advice on the possibility of stroke through remote communication.

3. **Mobile Applications**:

Applications provide easy tools to monitor health parameters such as blood pressure and pulse rate, allowing preliminary estimates of stroke possibilities based on the data entered by the user.

4. **Wearable Devices**:

Smart watches and fitness trackers monitor vital signs constantly and can detect abnormalities such as abnormal heartbeats or blood pressure that may result in a stroke.

1.3.1 Overview of Current Applications and Websites

The following are some of the examples of existing mobile apps and websites regarding Concern Stoke Disease:

Stroke Riskometer (Mobile App):

The application used in determining the chances of stroke conditions of an individual with the use of personal health data: age, gender, and medical history. The purpose of the application is for initial risk detection and precautions against stroke.

Stroke Management (Mobile App):

A supportive application made by Dr. Yahia Lodi involves monitoring of stroke risk factors, educating the patient about stroke prevention, and recommending changes in lifestyle. It is well known in the Middle East.

Stroke Awareness Foundation (Mobile App):

It includes information about stroke prevention, its identification, and rehabilitation. Applications can be used for the assessment of stroke risk and self-monitoring of key parameters of health. The application is available in many countries and is supported by the community.

Stroke Connect (Mobile App):

Stroke Connect helps patients and healthcare professionals connect and manage stroke recovery. This app contains a stroke risk assessment tool and monitors the recovery process. Geographically, this app serves India, mainly, and its surroundings.

Brainomix (Website):

For diagnosis and monitoring purposes, Brainomix offers advanced AI-enabled software solutions for stroke patients. It helps analyze brain imaging scans for early intervention decisions. Currently, the solution of Brainomix is used clinically in many varied countries around the world.

NIH Strokenet (Website):

The National Institutes of Health-designed NIH Strokenet encompasses a well-rounded knowledge base providing data, tools, and research in stroke prevention, treatment, and recovery. It has been used in the US for research and for clinical care.

1.3.2 Strengths and Weaknesses

Let’s review some of the strengths and weaknesses for each:

- **Stroke Riskometer (Mobile App):**

Pros:

- Friendly interface
- Personalised risk based on individual health parameters

Cons:

- Narrow scope-altlumately focuses on basic risk factors
- User information-based, so it may be subject to user bias

- **Stroke Management (Dr. Yahia Lodi) (Mobile App)**
  Pros:
  - Personal stroke risk monitoring
  - Education about prevention

  Cons:
  - Not as well-known out of the Middle East
  - Little interaction with caregivers
- **Stroke Awareness Foundation (Mobile App)**

  Pros:
  - Extensive information and resources on stroke
  - Health indicator monitors

  Cons:
  - No advanced predictive algorithms
  - Only general health advice is available, no risk calculation
- **Stroke Connect (Dr. Bindu Menon) (Mobile App)**

  Pros:
  - Direct communication between patients and doctors
  - Recovery tracking and stroke prevention tips

  Cons:

  - Availability is majorly rested in India
  - It is not fully functional without the internet.

- **Brainomix (Website)**

  Pros:

  - AI diagnosis and monitoring of stroke
  - Handy in clinical practice for detailed analysis of brain images

  Cons:

  - Very costly for an individual
  - Requires medical equipment for integration

- **NIH Strokenet (Website)**

  Pros:

  - An all-in-one platform for stroke research and clinical practice
  - Backed by a reputed organization, NIH

  Cons:

  - Purely academic usage; not designed for individual usage
  - Not available to the public; only through institutions

Chapter Two

Literature Review

Chapter Two: Literature Review

2.1 Suggested Solution

Stroke and heart disease remain leading causes of death and disability worldwide. Early detection and continuous monitoring are essential to **improving** outcomes, yet many current systems lack integration, accessibility, and real-time response. Our proposed solution aims to address this by combining Artificial Intelligence, Embedded Systems, and a Mobile Application into one unified platform.

From the patient's perspective, the solution enables continuous health monitoring through wearable sensors and a user-friendly mobile app. This allows individuals to track key indicators such as heart rate, oxygen saturation, and ECG in real time. AI-driven predictions provide personalized insights and early warnings, empowering users to take preventive action and manage their condition proactively.

From the medical and clinical side, the system supports healthcare professionals by providing structured, real-time health data and predictive insights. This assists in early diagnosis, improves treatment planning, and reduces diagnostic delays. By filling the gap between patient data and clinical decisions, it strengthens the overall healthcare infrastructure and enhances patient outcomes—especially for those with heart-related stroke risks.

2.2 External Expert Consultation

The experience and support of Dr. Hossam Sakr, a neurologist and psychiatrist at Tanta University Hospital, have contributed much to this study. In fact, his comments on the identification of the most influencing factors on stroke incidence, especially in cardiac-related illnesses, have created a great deal of value. The contribution of Dr. Sakr has been very important in shaping the form and structure of the predictive model so that clinically relevant data could model real-life scenarios.

Long-time experience in the diagnostics and treatment of stroke cases by Dr. Hossam Sakr was also of great value while determining the most contributing health parameters to the risk of stroke. His contribution helped further tune the approach of the project in the collection of data and the analysis of the predictions, so that the solution would be scientifically sound but clinically applicable. We are grateful for all his support and collaboration extended to us while developing this project.

2.3 Suggested Project

2.3.1 Introduction about the project

The concept is essentially based on creating a mobile app that will be integrated with a box sensor for the purpose of real-time monitoring of crucial vital health parameters along with stroke risk assessment. It will be further embedded with a chatbot to guide users through questions and answers with information regarding user health, app features, and measures for preventing strokes. This will ensure smooth access to support, making it more engaging and user-friendly.

The application acquires data from various sensors implanted in a box device, like heart rate, blood pressure, SPO2, and ECG signals while executing its processing via various machine learning models. It presents personal health insights, predictive analytics, and suggestions that chatbots can interactively act upon. AI-powered with real-time data monitoring on the health of people suffering from heart disease foretells the possible risk of a stroke so that the users can avoid such situations.

2.3.2 Properties of the Project

- **Real-Time Health Monitoring:**
  The application integrates with various sensors to capture vital signs such as heart rate, blood pressure, and oxygen saturation, providing continuous updates on the user's health status.

- **AI-Powered Stroke Risk Prediction:**
  Sensor-based health data is processed by machine learning models to estimate the probability of a stroke based on the user's medical history and real-time metrics.

- **CNN-Based Stroke Detection Model:**
  A Convolutional Neural Network (CNN) analyzes brain CT scans to classify and detect stroke conditions, assisting in early diagnosis and medical decision-making.

- **Interactive Chatbot:**
  A built-in chatbot assists users by answering questions, offering stroke prevention tips, and guiding them through app functionalities for a smoother experience.

- **User-Friendly Interface:**
  Designed for accessibility, the app offers an intuitive interface that accommodates users of all technical backgrounds.

- **Personalized Recommendations:**
  Based on the user's health data and risk profile, the app delivers customized advice on lifestyle changes, preventive actions, and when to seek medical consultation.

2.3.3 Objectives of the Project

In an era defined by technological advancements and digital innovation, this project aims to stroke risk prediction and prevention. By integrating AI, box device, and user-friendly applications, it aspires to empower users and healthcare providers alike.

Key Objectives are:

1. **Predictive Healthcare**:

   The app is supposed to predict the risk of stroke among heart patients and thus provide an opportunity to take necessary steps in preventing stroke-related complications.

2. **Interactive Assistance**:

   A chatbot in the application will ensure easy availability of information and will promptly assist the users.

3. **Continuous Health Monitoring**:

   Application interaction with sensors will ensure regular data gathering, mapping real-time glimpses of the condition of the user.

4. **Seamless Integration with device**:

   Application integration with a box device works in tandem because the health data collected from these devices is done so efficiently to process and provide timely alerts and predictions.

5. **Support for Healthcare Professionals**:

   It should provide critical data to the healthcare professional for decisive action in the management of patient care for improving clinical performance.

2.4 Technologies and Tools

Several technologies and tools, such as embedded systems, machine learning, and chatbots, work out in an integrated, smooth, and continuous system in a stroke risk prediction, health monitoring, and interactive assistant project that encompasses:

- **Embedded Systems**:

  The system is based on the integration of a variety of sensors, heart rate monitors, ECG sensors, SPO2 sensors, and BP devices into box device systems. These sensors will collect the health data from the user and feed it to the application.

- **Machine Learning (AI)**:

  Machine Learning acts as the backbone of the predictive model, analyzing a great number of different kinds of datasets available in multi-dimensionalities. Their contribution towards algorithms guarantees that the detection of the hidden pattern in stroke-related data has been identified, thus providing fine-tuned predictions. The design and training will be done using Scikit-learn.

- **Chatbot Technology**:

  Chatbots integrated with the app using NLP will be used to offer personalized services, answering customer queries and guiding users in navigating through features on the app.

- **Mobile Application**:

  It designs the user interface for health monitoring, predictive analytics, a chatbot, and recommendations on health via a mobile application. Cross-platform frameworks, like Flutter, can be applied in the development of the same to support devices on the Android platform.

- **Cloud Services**:

  Data from sensors and chatbot user interactions are managed and stored on cloud platforms, such as Google Firebase, for easy access and scalability. Cloud storage also secures health data of users, accessible remotely by health personnel.

This is an integrated solution that uses embedded systems, machine learning, chatbot technology, and mobile app integration for the realization of real-time health monitoring, prediction of possible stroke risks, and personalized assistance to enable users to act before serious complications can set in.

Chapter Three
Design and Methodology

Chapter Three: Design and Methodology

3.1 Use-Case Diagram

The use case diagram showcases the interactions between the primary user (the patient) and the healthcare application, alongside its integration with external systems. Patients can perform essential tasks such as registration, login, medical data entry, and interacting with a chatbot for guidance. The system enables users to input health metrics, such as blood pressure, which can be collected manually or through hardware devices. Data is securely transferred to Firebase (database) or cloud services for storage and processing. The chatbot further assists users by offering support and resolving queries.

Key external integrations include Firebase for data management, cloud services for scalability and processing, and hardware devices for health data collection. The system also includes optional features like sharing results on social media platforms. Relationships between use cases use UML standards such as include and extend relationships to ensure error handling and verification of data. This patient-centric system is all about security, accuracy, and seamless integration for effective healthcare management.

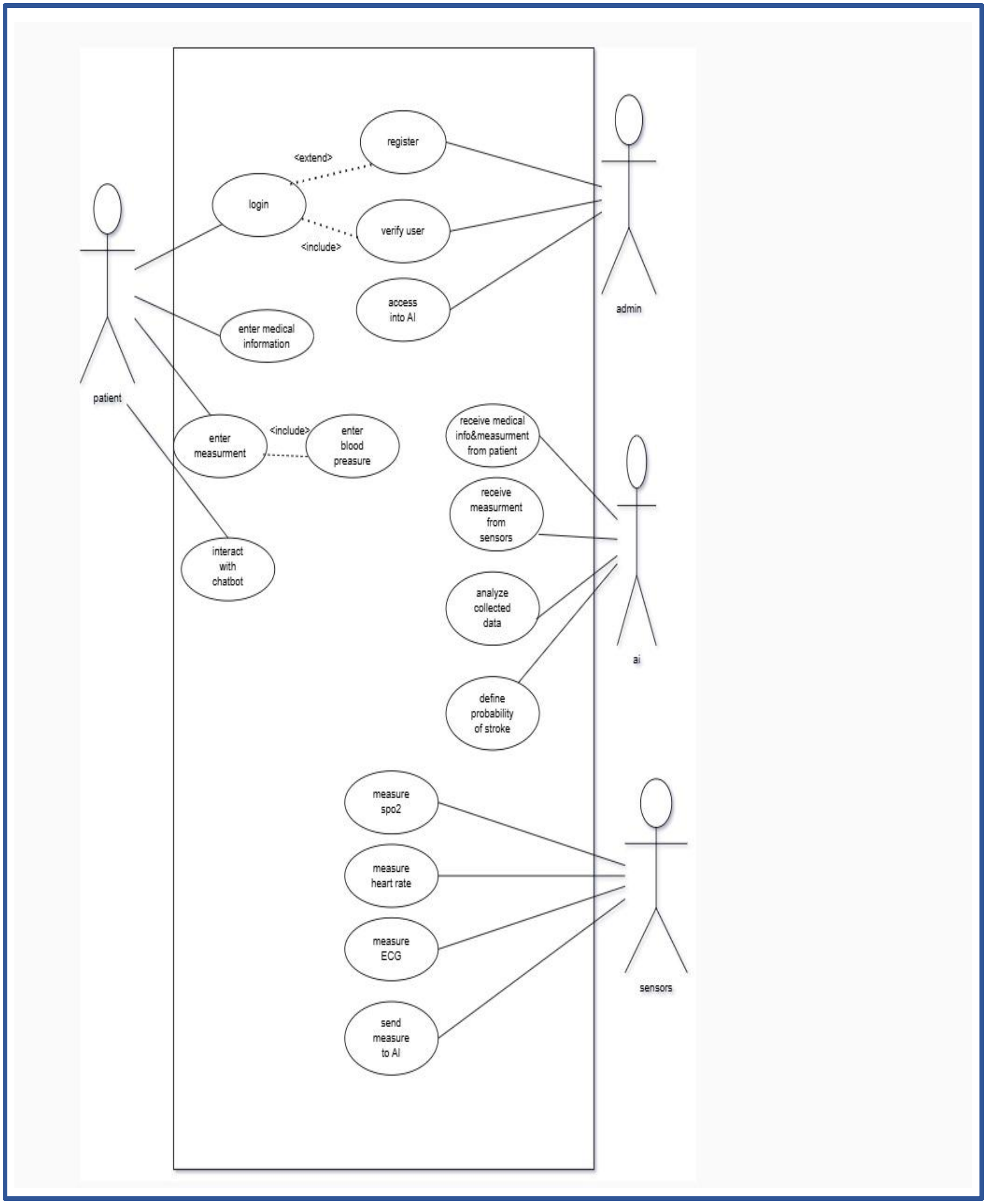


Figure 3.1 Use-Case Diagram

3.2 Sequence Diagram

It gives an overview of the series of workflows that this healthcare application is intended to use Artificial Intelligence in stroke risk detection and guide the user thereafter. This sequence diagram will show the interaction of a user, mobile application, box device, cloud service, and chatbot in performing its functions and logical structure step by step.

The process is initiated with a registration or logging into the mobile application by a user. Afterwards, a user fills in the medical data either manually or using a box device. The cloud saves data to perform further analysis using the AI model. Then it will present to the system to what degree a patient has the possibility to develop stroke, with several recommendations. Also, the chatbot feature provides more facilities to ask certain questions over any topic.

This diagram is important for system design because it aids the developers and stakeholders in understanding the workflow, pinpointing issues that might arise, and thus optimizing the processes. It will also help designers, developers, and testers align on how the system will behave, thus guaranteeing efficiency and friendliness in the health care solution.

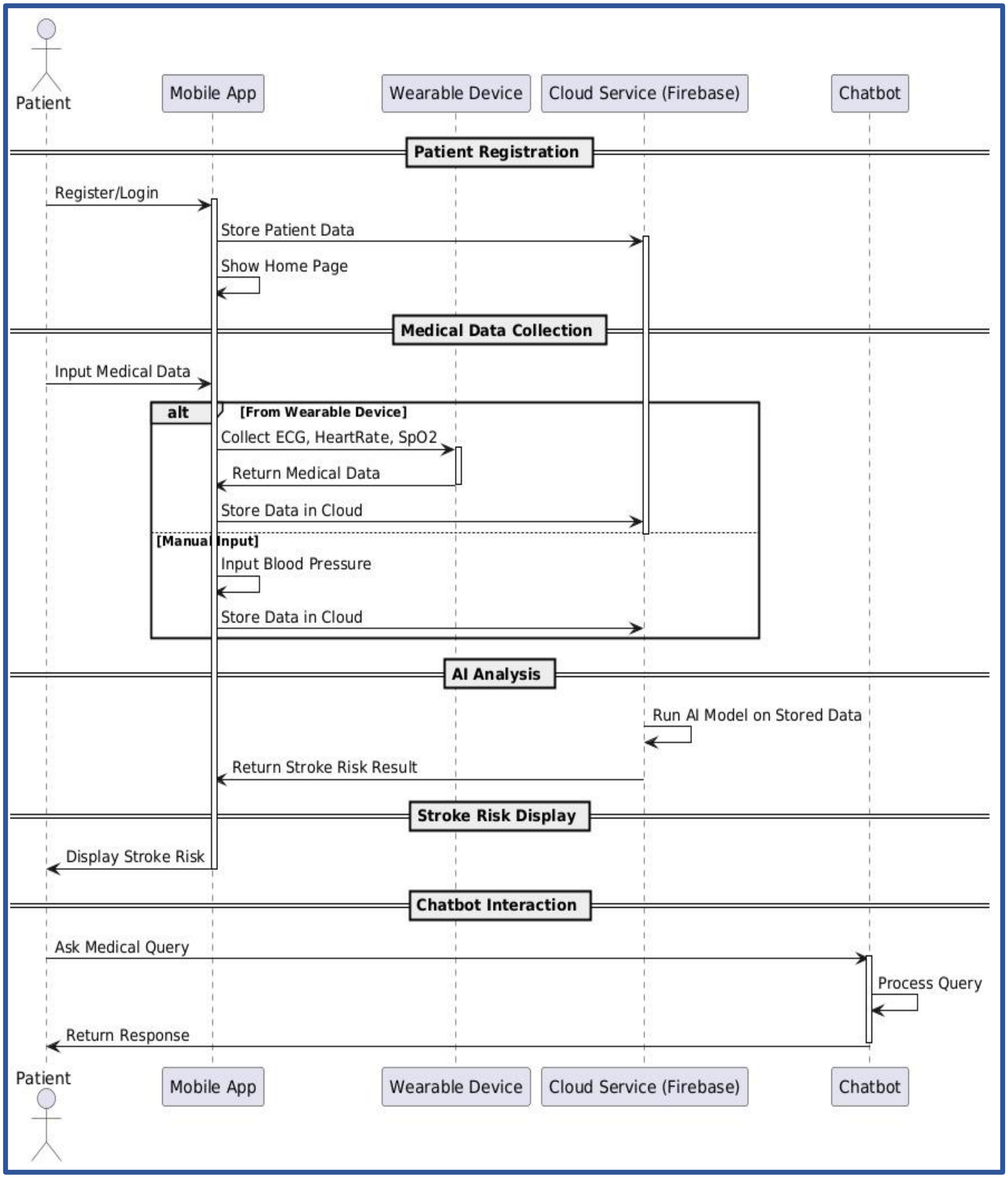


Figure 3.2 Sequence Diagram

3.3 Data Flow Diagram (DFD)

3.3.1 Context Diagram

This diagram shows the interaction between the central process and the entities interacting from outside.

The input from the patient for some medical information or measurement in the form of main process will be "Predict Cerebrovascular Stroke Disease." Sensors will provide SPO2, heart rate, and ECG.

The processing will be performed by AI itself and the predicted stroke probabilities. Outputs to be given with the patient and AI systems to efficiently perform diagnosis and predictions.

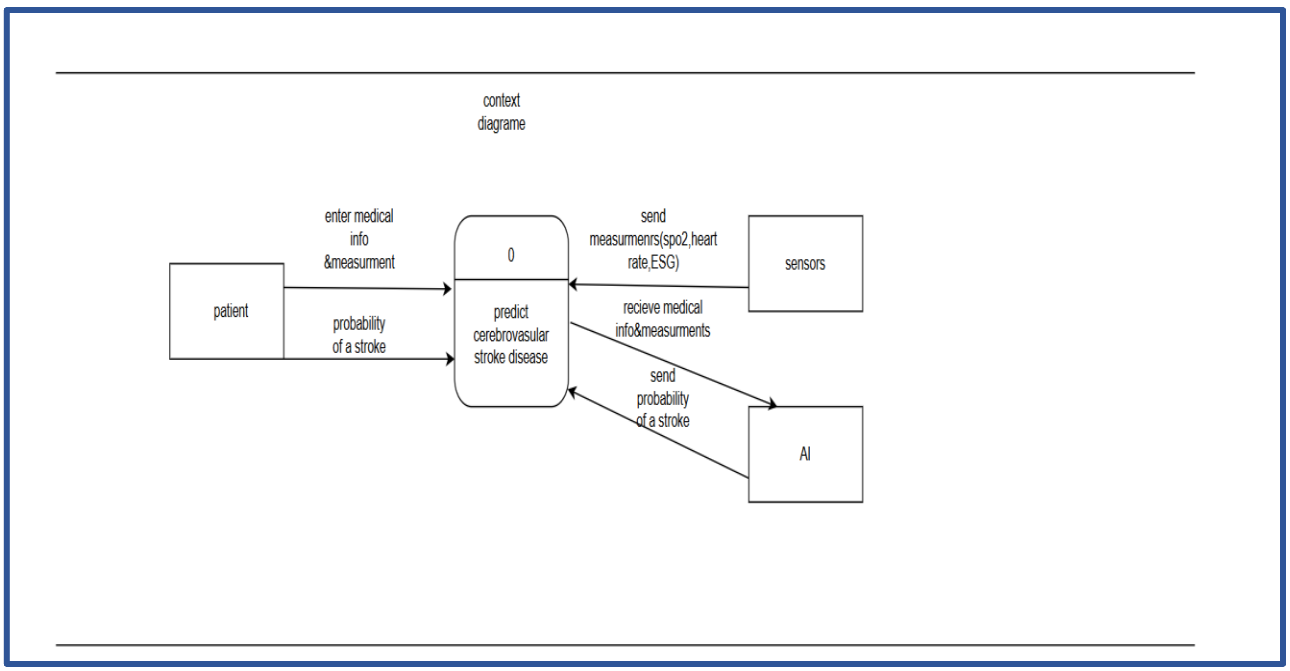


Figure 3.3 Context Diagram

3.3.2 Level 0

This diagram decomposes the system into three major processes: data analysis received, which is Process 1.0; calculating the measurements upon the receipt of data, which is Process 2.0; and calculation of stroke probability, which is Process 3.0.

It also identifies the interactions with external entities, such as the patient and sensors. Data flows concerned are medical information, measurements, and probability results.

AI, storage of data, D1 for the proper processing to rightly calculate the stroke probability.

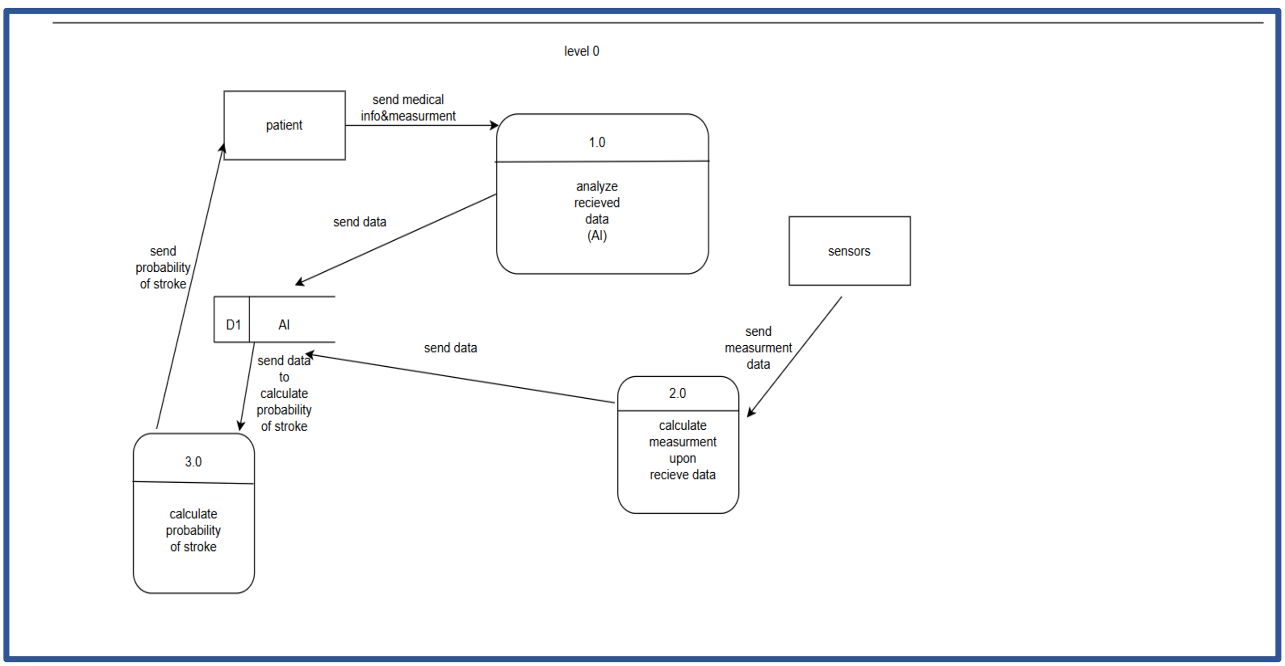


Figure 3.4 Data Flow Diagram (Level 0)

3.3.3 Level 1

This diagram breaks down the major process into three sub-processes: analysis of the received data, Process 1.1; assessment of the patient's condition, Process 1.2; and data transmission to AI for further analysis, Process 1.3. Received medical data and measurements are processed. Each subprocess finds out the accuracy of the assessment; therefore, output from the analyzed data or condition evaluation will flow down further where AI-based predictions are made or needs further processing. This will add to efficiency in decision-making.

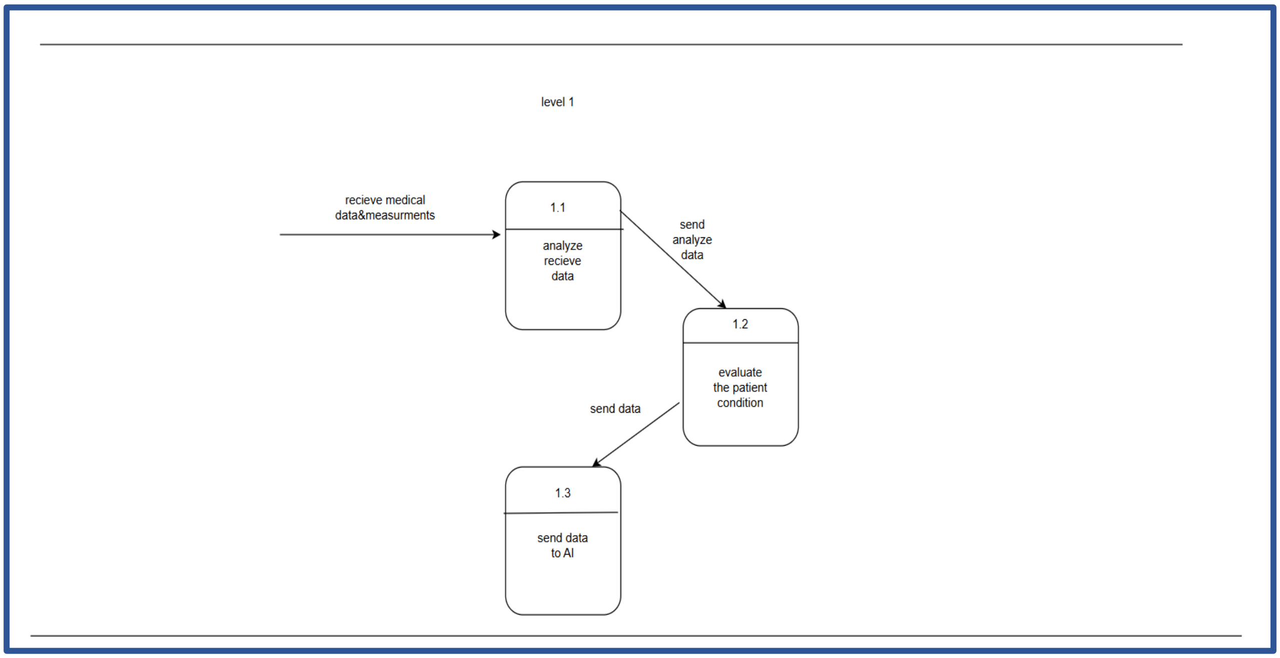


Figure 3.5 Data Flow Diagram (Level 1)

3.3.4 Level 2

More in detail, the measurement processes are decomposed into four concrete subprocesses: measuring SpO2 (Process 2.1), heart rate (Process 2.2), and ECG (Process 2.3).

Measured data in each subprocess goes through sequential outputs to ensure completeness and accuracy of vital signs monitoring. Finally, all measurements gathered shall be transmitted to an AI system (Process 2.4) for integration and analysis. This structure enhances precision in stroke prediction and diagnosis.

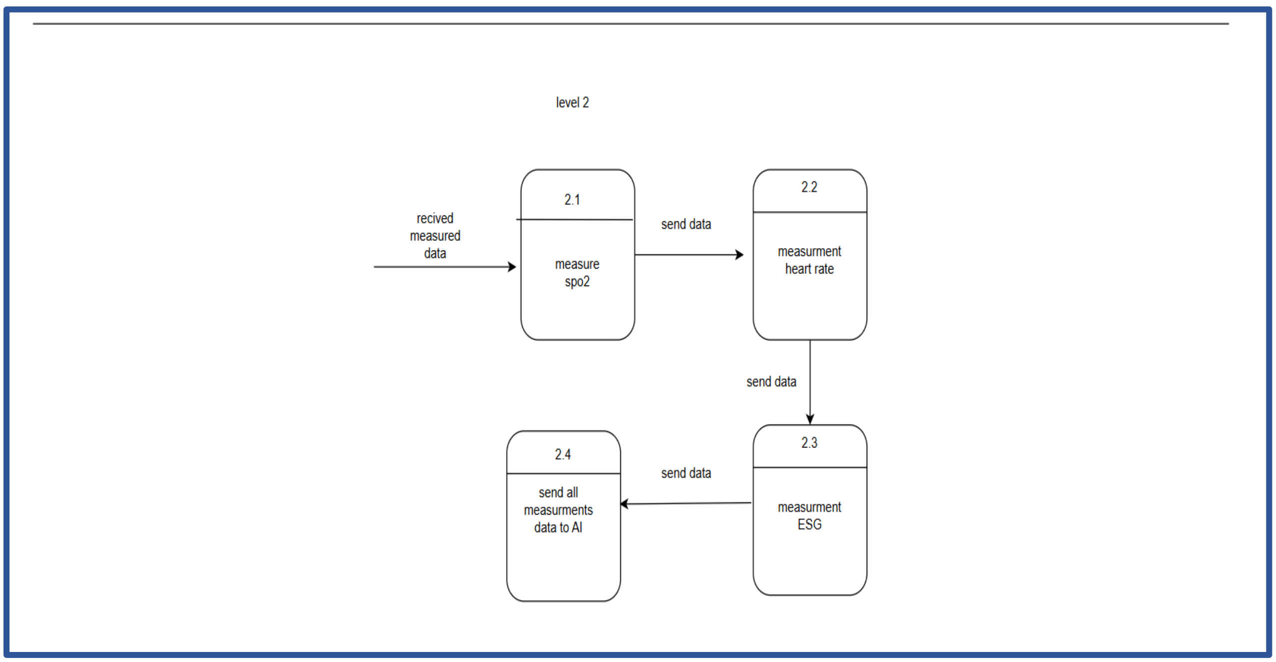


Figure 3.6 Data Flow Diagram (Level 2)

3.4 Entity Relationship Diagram (ERD)

This diagram of a healthcare system's architecture of data. The entities of the diagram are Patient, MedicalData, Sensors, Device, Admin, AIAnalysis, and Chatbot. The interrelations between the entities present the patients conversing with the admins, recording data from sensors as well as getting the AI tools analyses. Admins manage the operations; devices are connected to patients for their data monitoring, and the chatbot handles queries for better communication. The significant data points are the timestamps, such as those that when analyzed output probabilities, and identifiers that allow different functions and tracking within the system. The attributes in which these will assist involve writing software for continuous health monitoring and decision-making purposes.

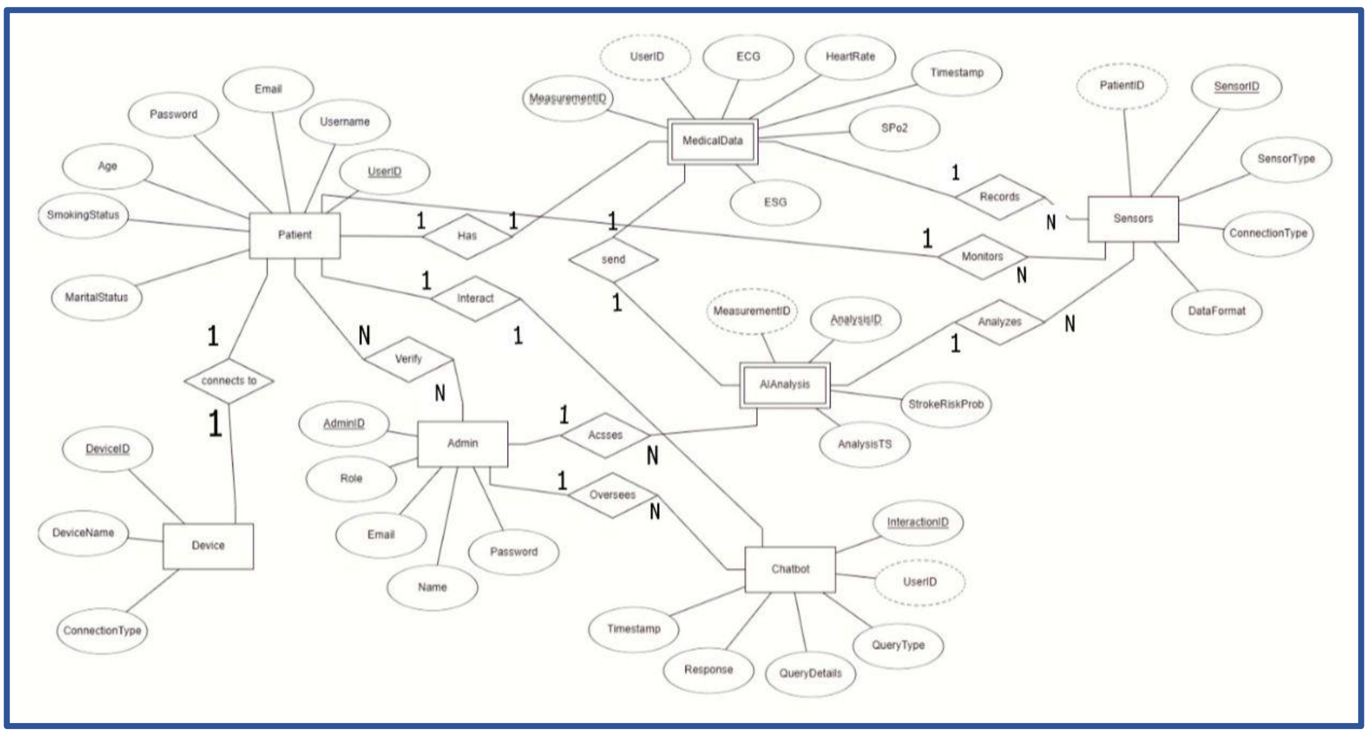


Figure 3.7 ERD Diagram

3.5 Activity Diagram

3.5.1 Patient Activity Diagram

This flowchart depicts the flow a patient would take while interacting with the healthcare application. This starts when the application is opened and checks whether the user is logged in or not. If he is, the patient will go straight to the home page of the application; otherwise, register or log in, saving this information locally.

Here, the patients engage with the application either through getting measurements using a box connected device that captures most of the parameters about the diseases of ECG and heart rate or take direct measurements without the use of a connected device.

This information will then be displayed, analyzed, and forwarded for artificial intelligence diagnosis about stroke, among other diseases in Firebase. Having done all these processes, Firebase will analyze the AI-powered diagnosis in search of risks for stroke disease.

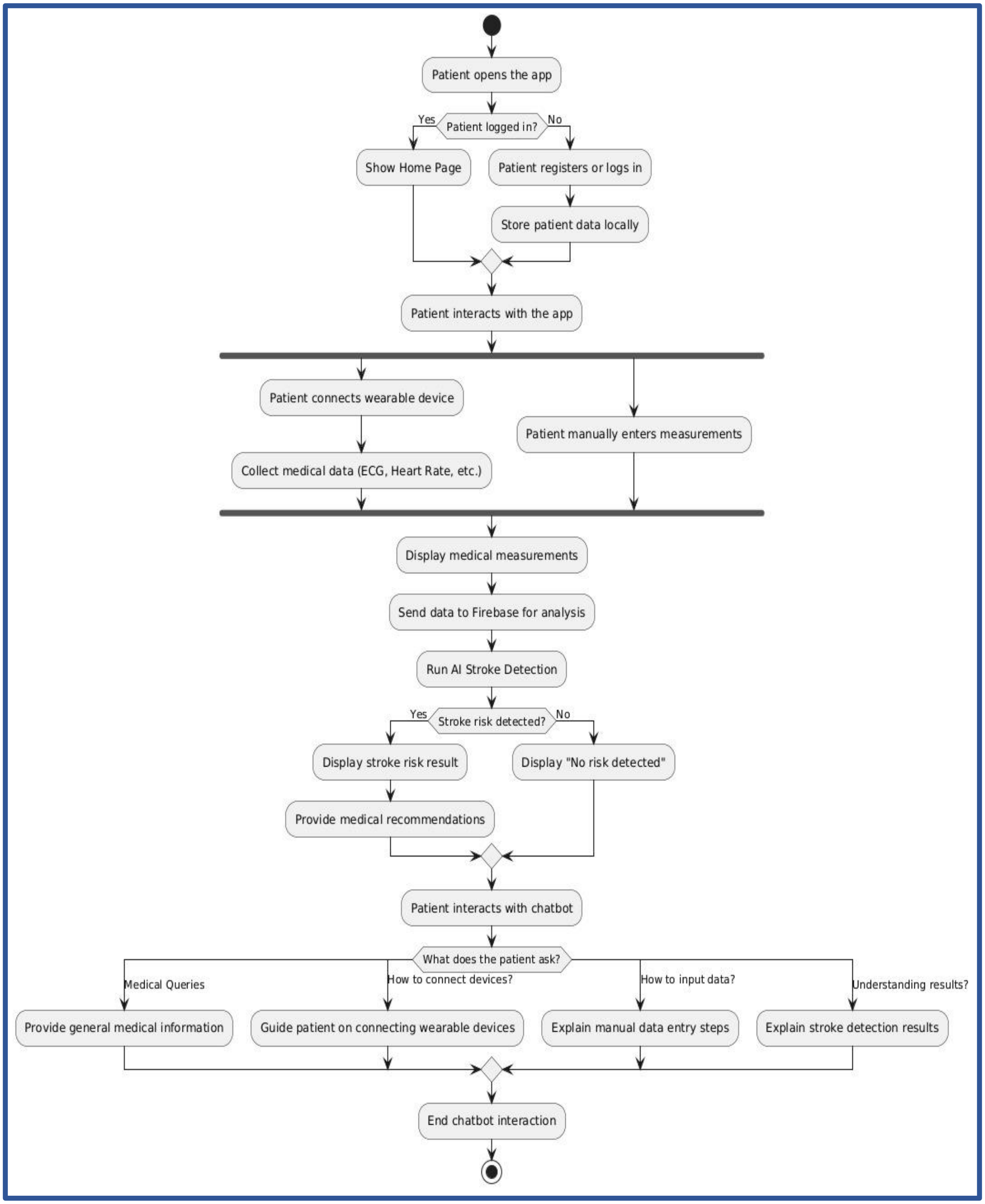


Figure 3.8 Patient Activity Diagram

3.5.2 Admin Activity Diagram

This flowchart outlines the admin activity workflow in a healthcare system. The process begins with the admin logging into the admin panel. Upon access, the admin performs key tasks such as verifying patient accounts, reviewing AI analysis logs, and monitoring chatbot interactions.

Based on these tasks, the admin determines if any action is required. If account management is needed, they update or verify patient accounts. For troubleshooting, the admin resolves system issues or assists patients. Additionally, they may generate, and review system reports to ensure proper functioning.

After completing all necessary actions, the admin logs out of the system, concluding their session. This workflow ensures efficient system management and user support.

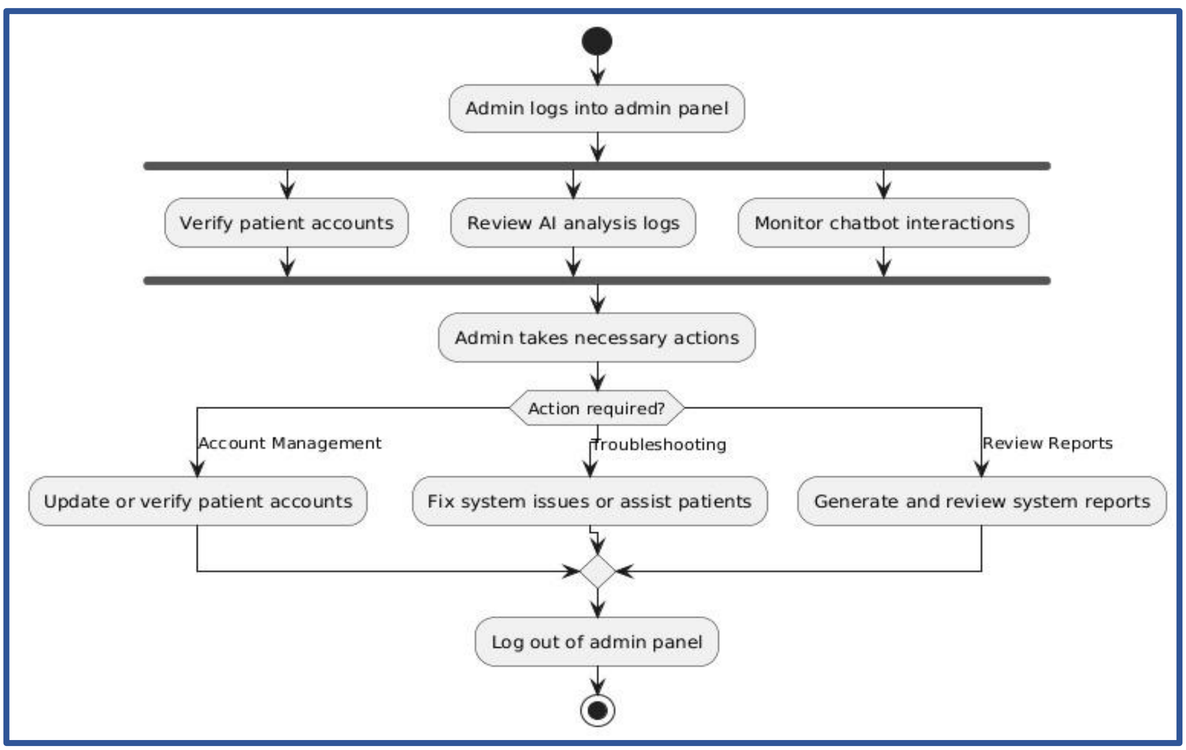


Figure 3.9 Admin Activity Diagram

Chapter Four

Implementation Work

Chapter Four: Implementation Work

4.1 Application Description

The application focuses on predicting stroke and heart disease using artificial intelligence to support early detection and prevention. It aims to assist individuals in monitoring their health by analyzing medical indicators and providing real-time feedback through a mobile interface.

It uses a user-friendly design, making it easy for patients to interact with the system. The interface is simple, clear, and responsive across devices, enhancing the overall user experience.

"Sahha" integrates with box device sensors that collect real health data such as heart rate, blood pressure, and ECG signals. These data are sent to a cloud-based AI model hosted on Google Cloud Platform for prediction.

The application also includes a chatbot for user interaction, a self-check feature, and a request system for medical devices. All user data and actions are securely managed through Firebase services to ensure safety and scalability.

4.2 Firebase Integration

Firebase serves as the backbone of “Sahha” backend infrastructure, providing essential services that enable secure data handling, user management, and real-time interaction.

4.2.1 Firebase Authentication

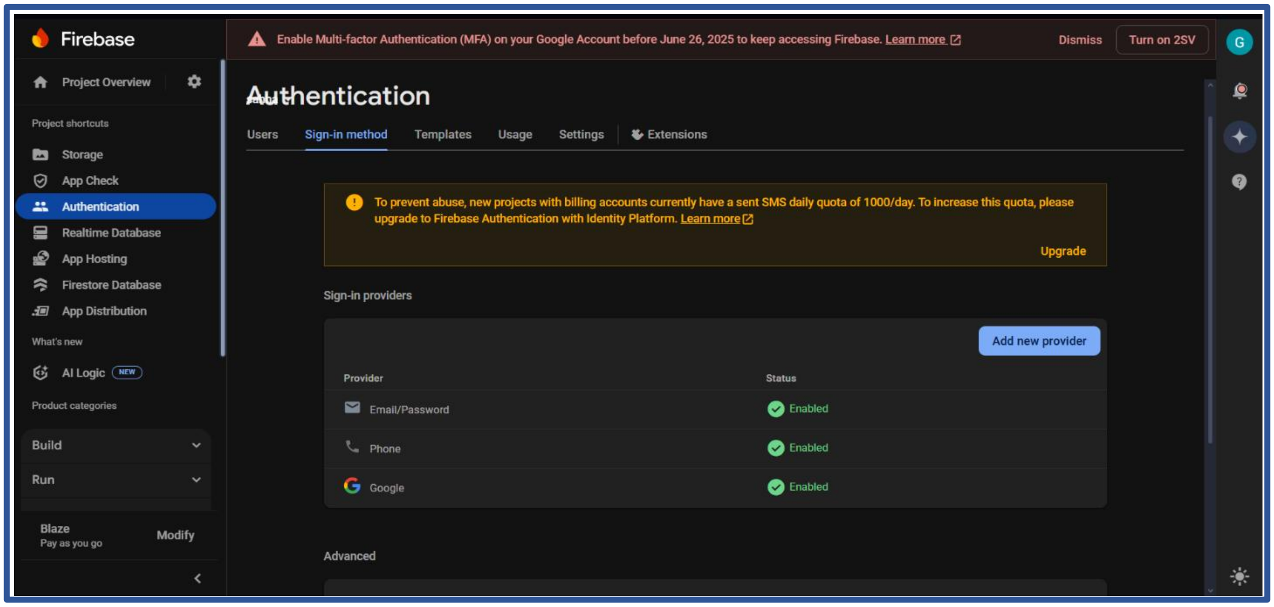


Figure 4.1 Firebase Authentication

The application uses Firebase Authentication to manage user sign-in through both email/password and Google accounts. This allows for smooth onboarding experience while maintaining a high level of security for all users.

4.2.2 Cloud Firestore

All user-related data, including personal health profiles, device requests, and interaction history is stored in Cloud Firestore, a flexible NoSQL cloud database. Firestore enables real-time data syncing between the mobile app and the cloud, ensuring that user data is always up to date.

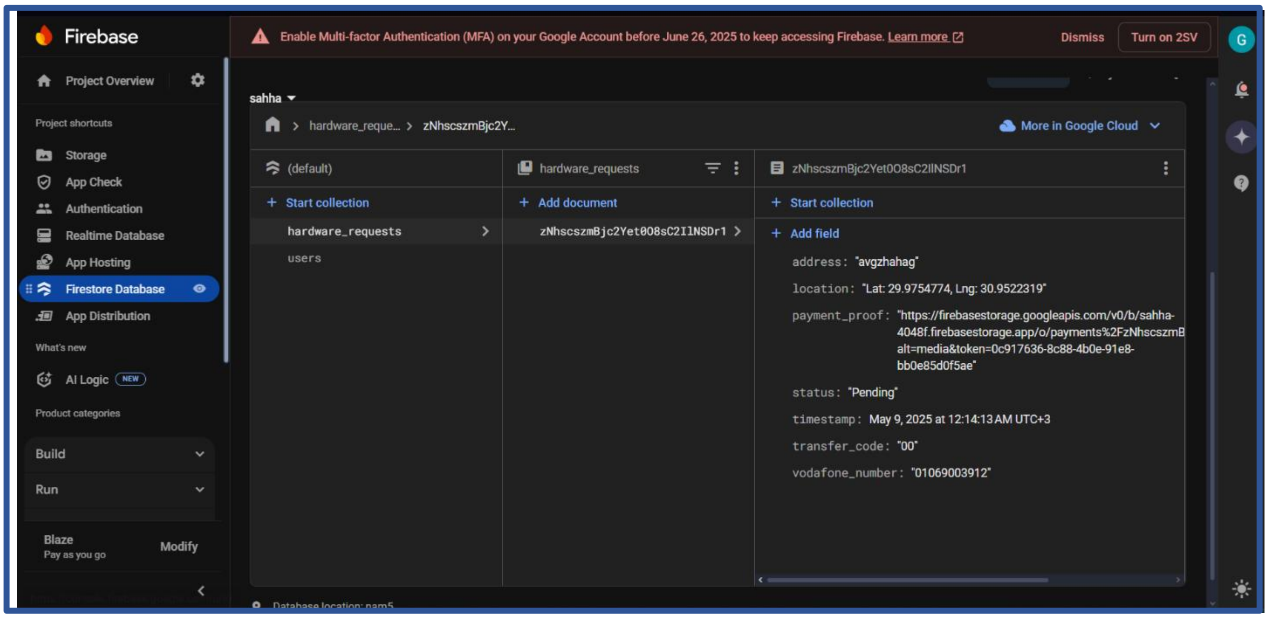


Figure 4.2 Firestore Database

4.2.2 Firebase Storage

In addition, Firebase Storage is used to handle file uploads, especially payment proof images submitted during device requests. These files are stored securely and linked to the corresponding Firestore records.

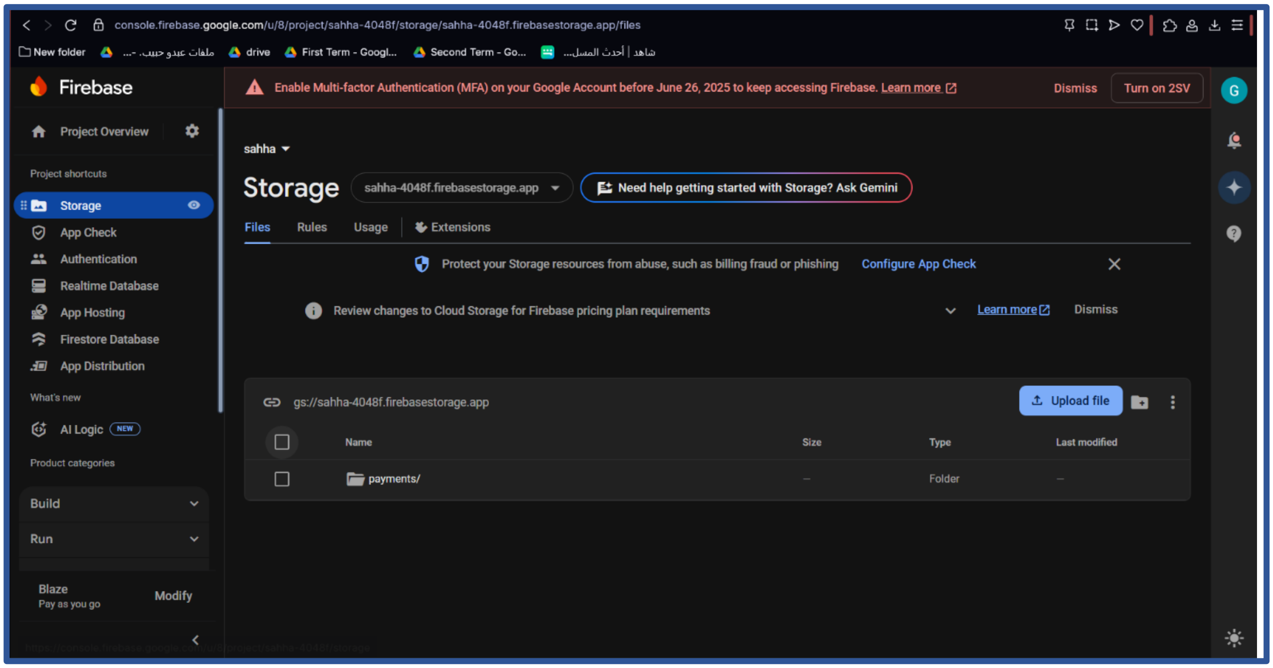


Figure 4.3 Firebase Storage

This integration of Authentication, Firestore, and Storage within Firebase provides a scalable, reliable, and secure ecosystem that supports “Sahha health monitoring features and enhances user experience.

4.3 AI & Google Cloud Platform Integration

The core intelligence of the “Sahha” application lies in a cloud-hosted AI system developed using Flask and deployed on Google Cloud Platform (GCP). This system is responsible for handling all prediction-related operations.

The backend API receives input data from the app either manually entered by the user or collected from sensors and passes it through two machine learning models:

- One for predicting heart disease
- The other for detecting stroke risk

Once both models finish processing, the API returns a single, combined result to the application in real-time. This allows the user to receive immediate health insights with no delay.

Hosting the system on GCP ensures:

- High availability
- Scalability
- Low latency even under heavy traffic, making it reliable for continuous medical use.

By offloading the prediction process to the cloud, the app remains lightweight and responsive, while benefiting from the computational power of Google Cloud’s infrastructure.

4.3.1 AI Module Implementation

The AI module implemented in the system plays a vital role in analyzing health data to predict the risk of cerebrovascular stroke, especially in patients with underlying heart conditions.

Model Architecture

The architecture was designed to include a nested model, where a heart disease prediction model is run first, and its result is then passed to the stroke prediction model. This structured flow ensures that the stroke risk prediction considers the user's cardiac condition as a contributing factor, thus increasing the reliability of the results.

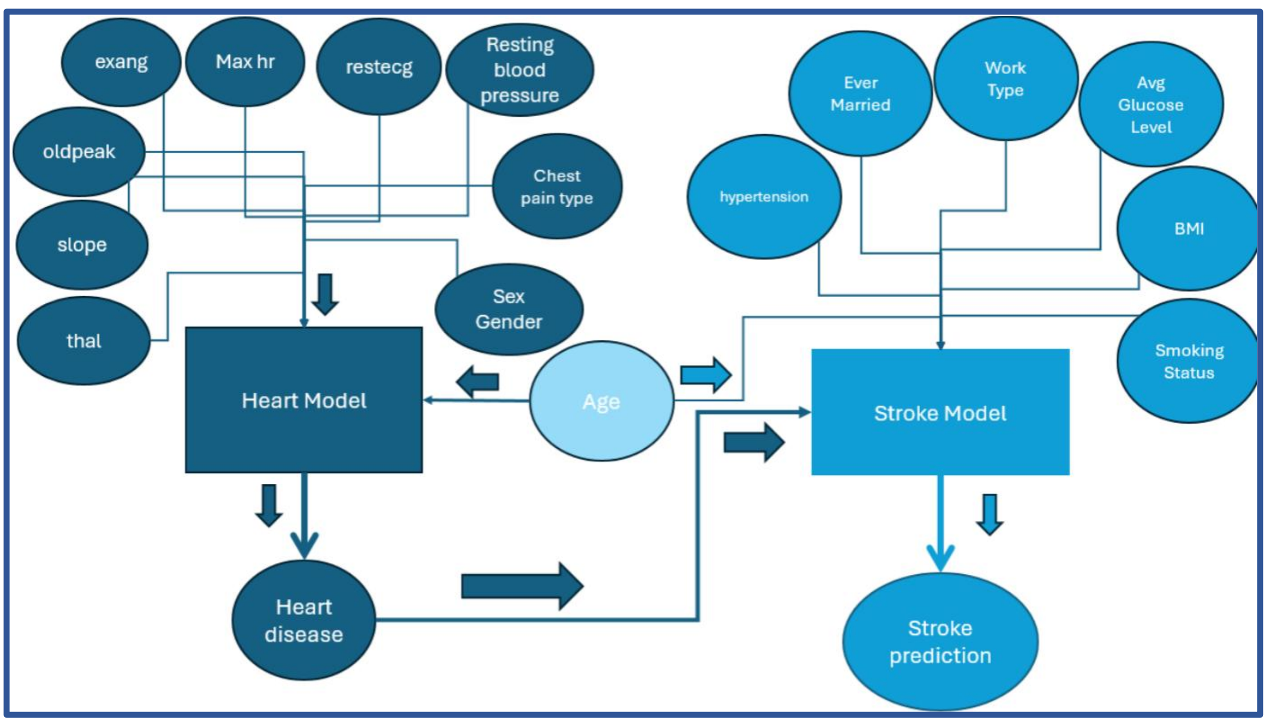


Figure 4.4 Model Architecture

Datasets Used: The models were trained using publicly available stroke and heart disease datasets. The stroke dataset was obtained from Kaggle's "Stroke Prediction Dataset"

(https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset), Dataset Features (id, gender, age, hypertension, heart_disease, ever_married, work_type, residence_type, avg_glucose_level, bmi, smoking_status, stroke)

while the heart disease dataset came from the UCI Machine Learning Repository.

Dataset Features (age, sex, chest_pain_type, resting_bp, cholesterol, fasting_blood_sugar, restecg, max_hr, exang, oldpeak, slope, num_major_vessels, thal, target)

(https://www.kaggle.com/datasets/abhishek14398/heart-disease-classification). These datasets provided a diverse range of features necessary for model training and evaluation.

Deployment Using Flask and GCP: Once trained, the AI models were deployed using a Flask-based web server. Flask was selected due to its lightweight and flexible nature for API deployment. The application programming interface (API) receives data from the mobile application, processes it through the AI models, and returns a probability score indicating stroke risk. This API is hosted on Google Cloud Platform (GCP), which ensures scalability, real-time performance, and secure access.

API Response Flow: The mobile application collects user health data via sensors or manual input, then sends this data to the API. The backend processes the heart disease prediction first; if positive, the result is appended to the stroke model input. The final stroke prediction score is then returned to the mobile app and presented visually to the user, enabling proactive health management.

This module is a core component of the entire system, bridging real-time health data with advanced AI decision-making to provide timely, personalized stroke risk assessment.

4.3.2 Chatbot Implementation

We fine-tuned **Meta-Llama-3.1-8B** to create **Sahha**, a medical chatbot specialized in stroke-related information.

The training data was sourced from well-known medical textbooks on stroke and restructured into instructional Q&A format using large models like **ChatGPT** and **Gemini**.

These models helped clean, summarize, and format the content to suit fine-tuning objectives.

Link-dataset: https://huggingface.co/datasets/Ahmed-El-Sharkawy/Stroke

Link-Model: https://huggingface.co/Ahmed-El-Sharkawy/Stroke-medical-model-finetuned

Fine-tuning was performed using **instruction-based prompts**, with 4-bit quantization applied to optimize memory usage. The resulting model can accurately answer stroke-related medical questions, offering a reliable tool for students, clinicians, and patient education. By focusing on a

single domain and using expert-validated content, **Sahha** ensures higher factual accuracy and interpretability than general LLMs.

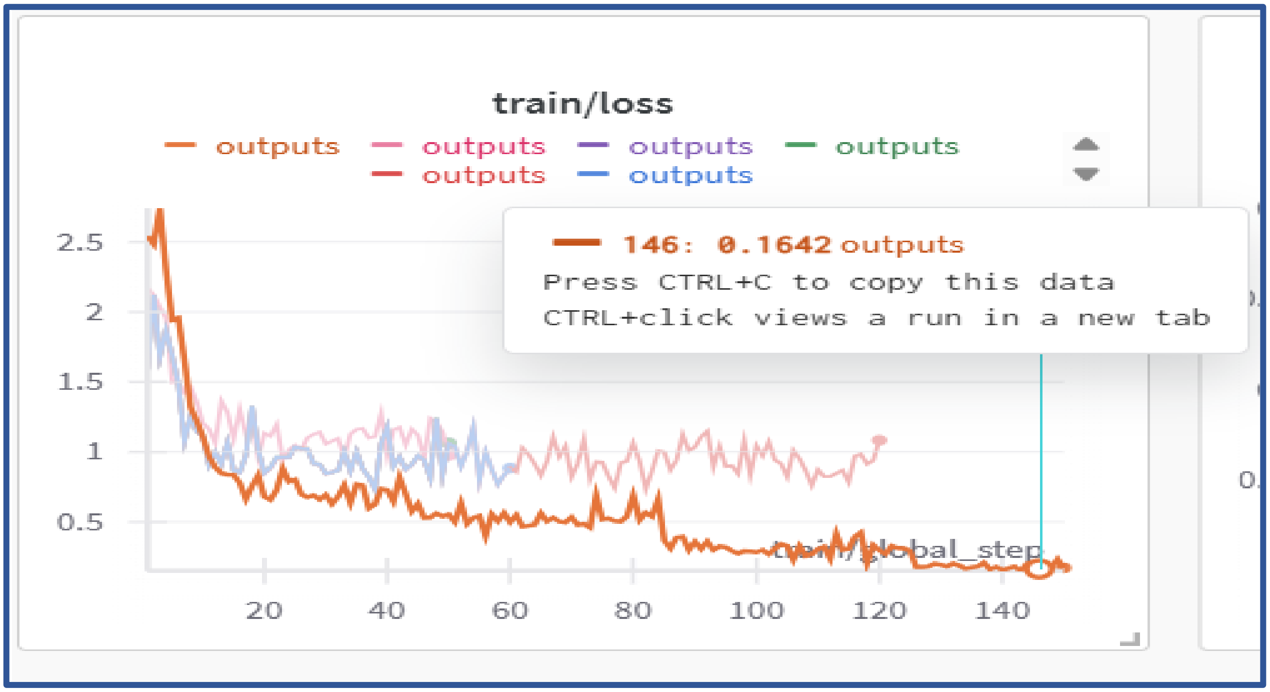


Figure 4.5 Train/Loss CNN Model

4.3.3 Computer Vision Implementation

A deep learning system for automated stroke detection from brain CT scans using a custom Convolutional Neural Network (CNN) architecture. The model performs binary classification to determine whether a CT scan is *Normal* or shows signs of *Stroke*.

4.3.3.1 CNN Architecture

The stroke classification model uses a custom Convolutional Neural Network (CNN) to perform binary classification of brain CT scans as either “Normal” or “Stroke.”

It includes convolutional and pooling layers for feature extraction, followed by fully connected layers for prediction.

Trained from scratch, the model is lightweight and suitable for real-time clinical deployment with input size 224×224×3.

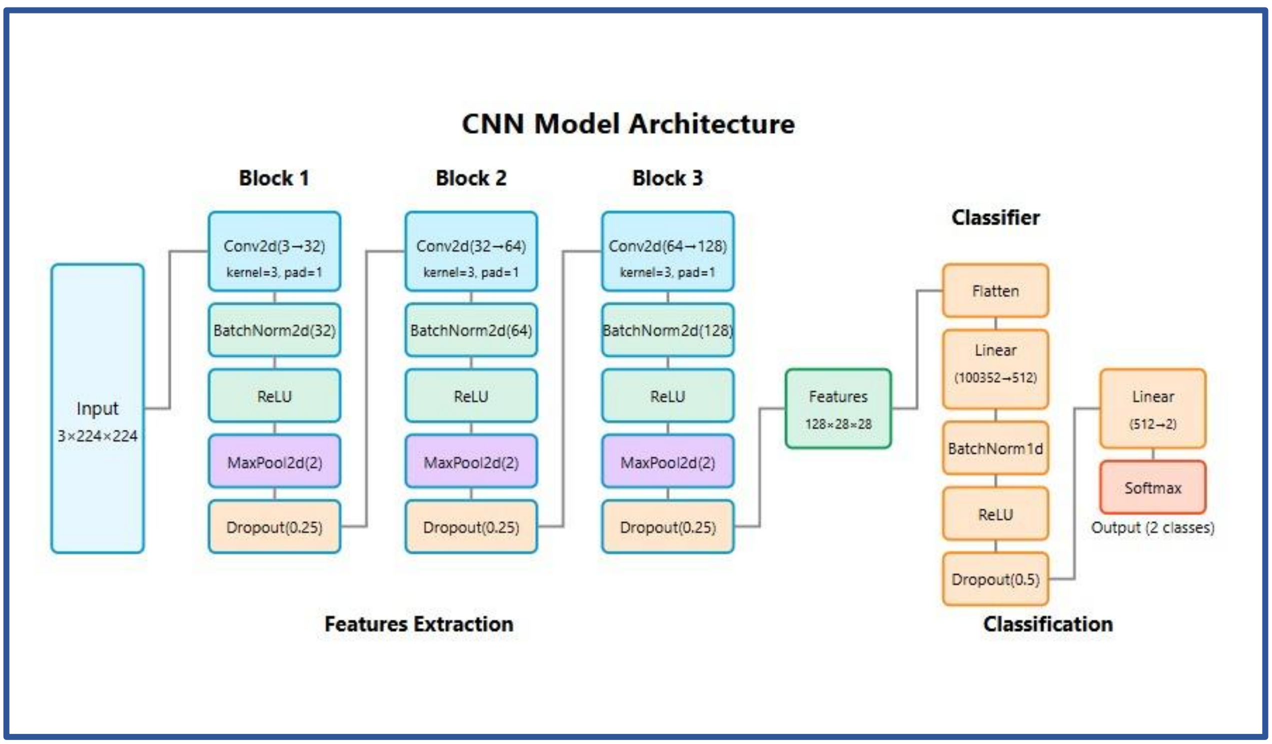


Figure 4.6 CNN Model Architecture

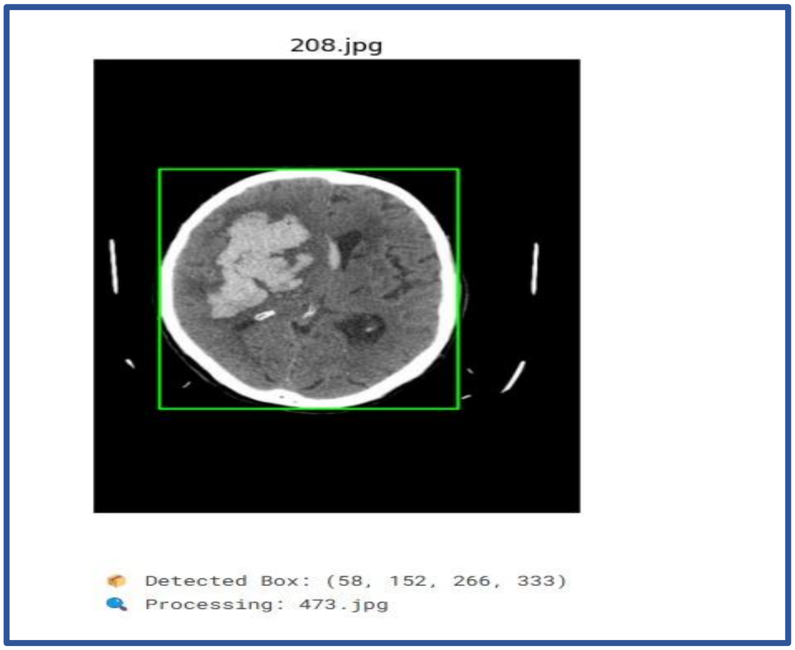


Figure 4.7 Detected abnormal region highlighted by a boundary box

This highlights the detected abnormal region in a brain CT scan using a green bounding box. The goal is to localize potential intracerebral hemorrhage areas for aiding clinical diagnosis and validating model-based stroke detection.

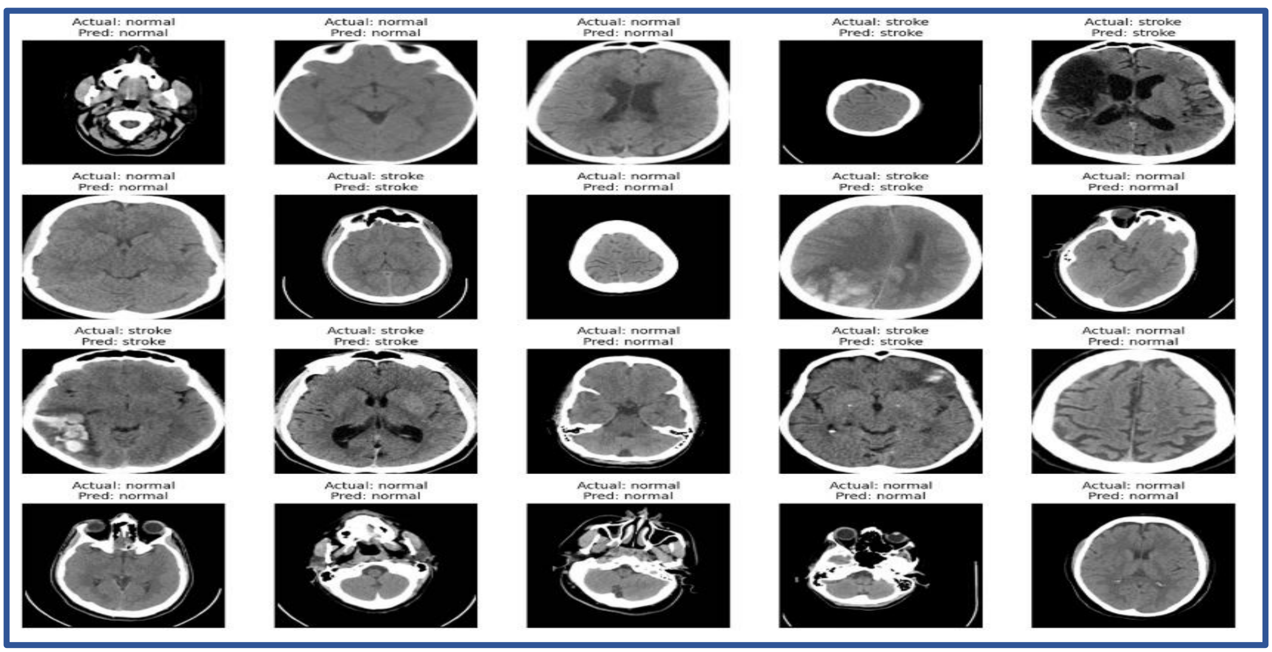


Figure 4.8 Stroke Classification

4.4 Application Interface

"Login Screen"

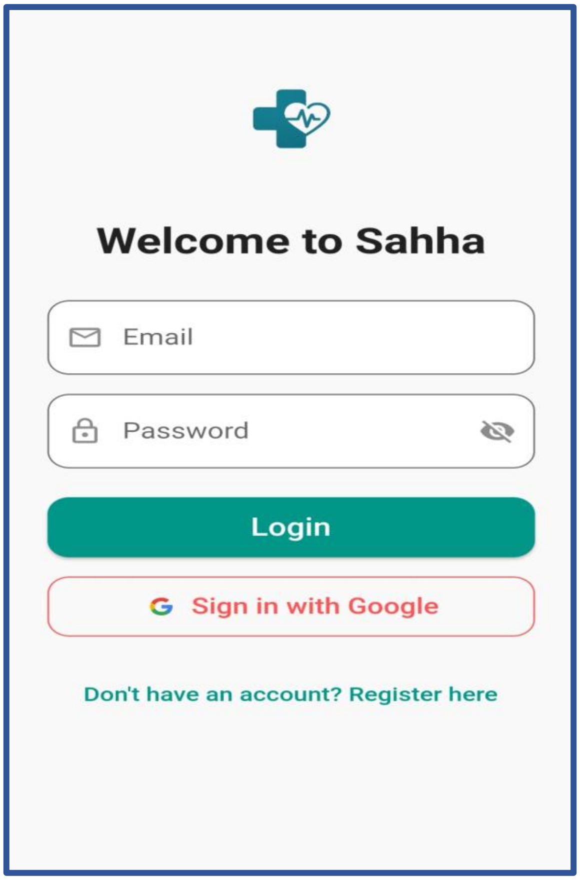


Figure 4.9 Login Screen

The login screen enables users to access the app via email/password or Google account. Firebase Authentication ensures user identity and secure session management.

“Dashboard Screen”

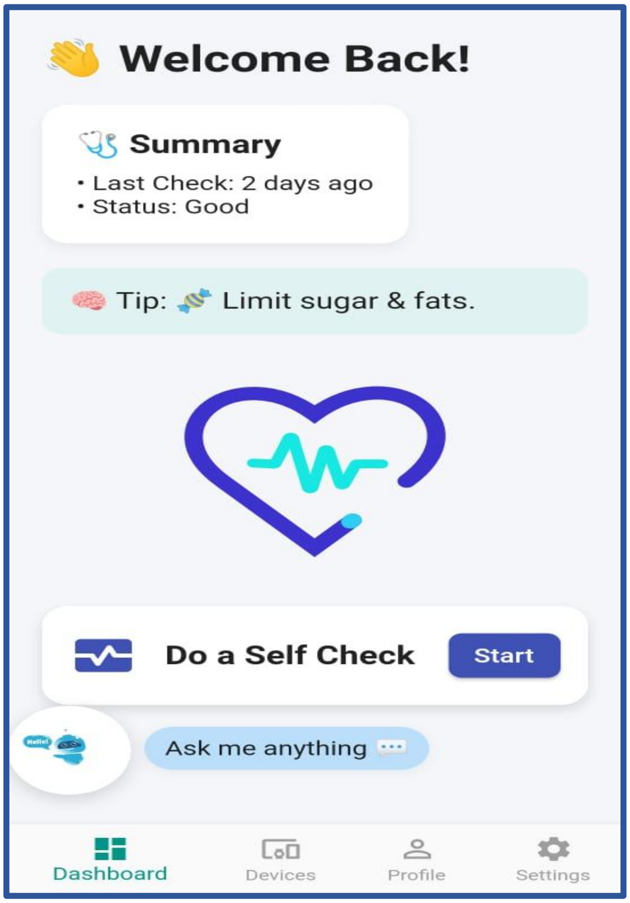


Figure 4.10 Dashboard Screen

The dashboard is the main landing screen after logging in. It displays the user's latest health status summary, including the result of the last self-check and its timing. The screen also provides rotating health tips and a direct button to perform a new self-check. At the bottom, users can interact with the chatbot for guidance or general questions.

“Profile Screen”

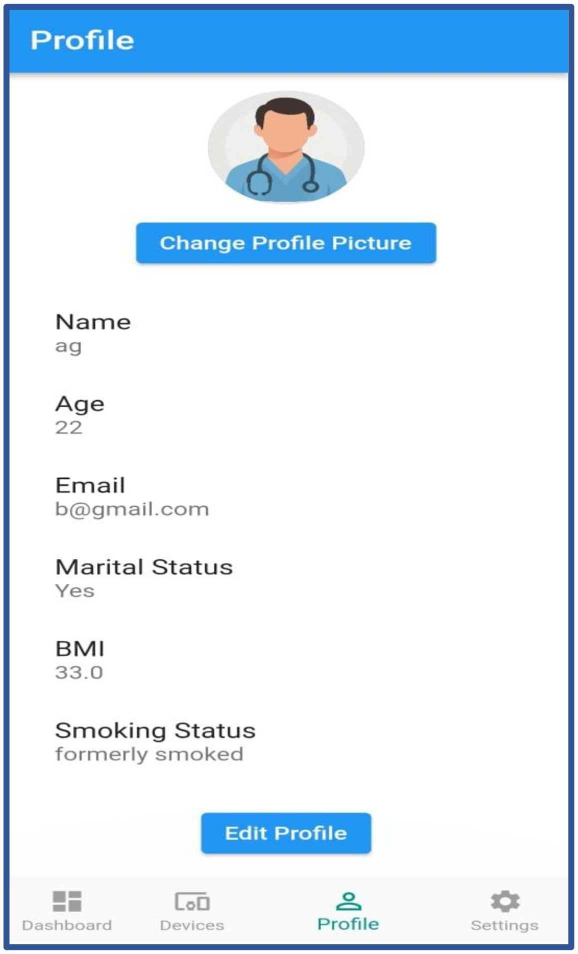


Figure 4.11 Profile Screen

The profile screen displays the user's personal and medical data including name, age, BMI, smoking status, and marital status. This information is used as input features for the AI prediction models. The screen also allows the user to change their profile picture and access the “Edit Profile” section for updates. All changes are synced directly with Firestore to ensure data consistency.

“Edit Profile Screen”

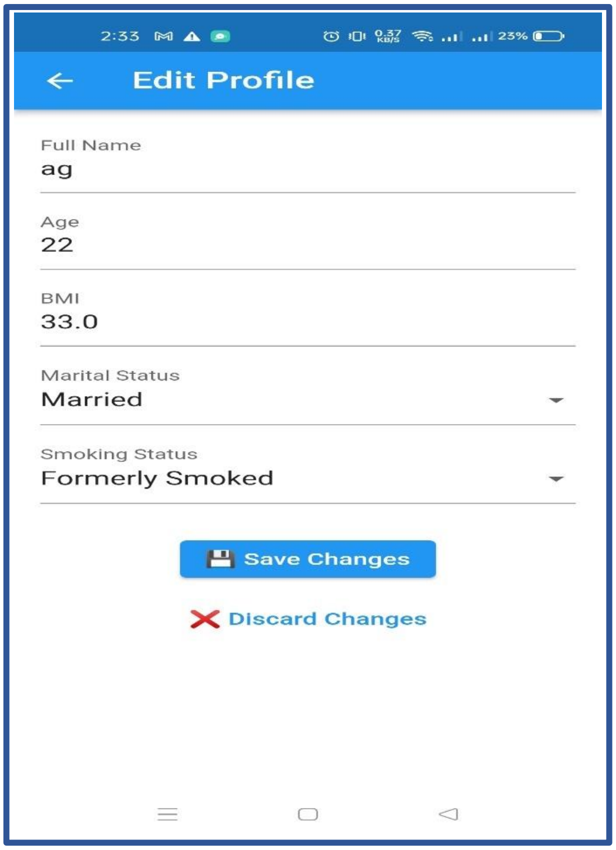


Figure 4.12 Edit Profile Screen

This screen allows users to update their personal and medical information, such as name, age, BMI, and smoking status. Changes are saved securely to Firestore, ensuring the AI model always uses the most recent data for predictions.

“Change Password Screen”

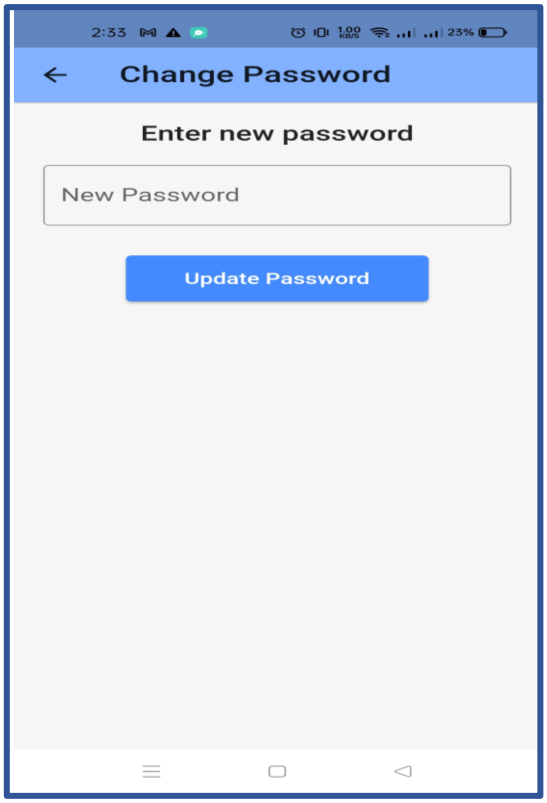


Figure 4.13 Change password screen

This screen allows users to securely update their password through Firebase Authentication, ensuring account protection and user privacy.

“Request Device Screen”

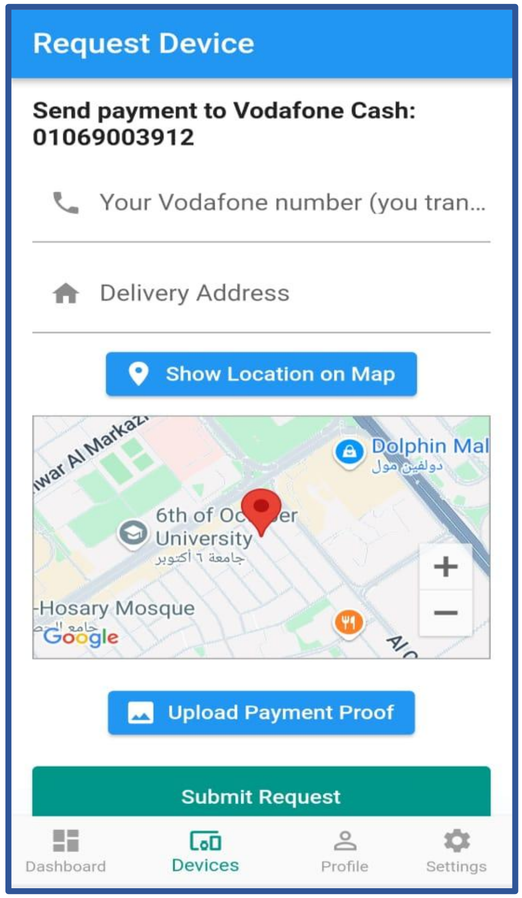


Figure 4.14 Request Device Screen

This screen allows the user to request the wearable hardware device used for performing self-checks. The patient provides their Vodafone Cash number, delivery address, and uploads proof of payment. The request is submitted along with their live location to facilitate accurate delivery.

“Self-Check Screen”

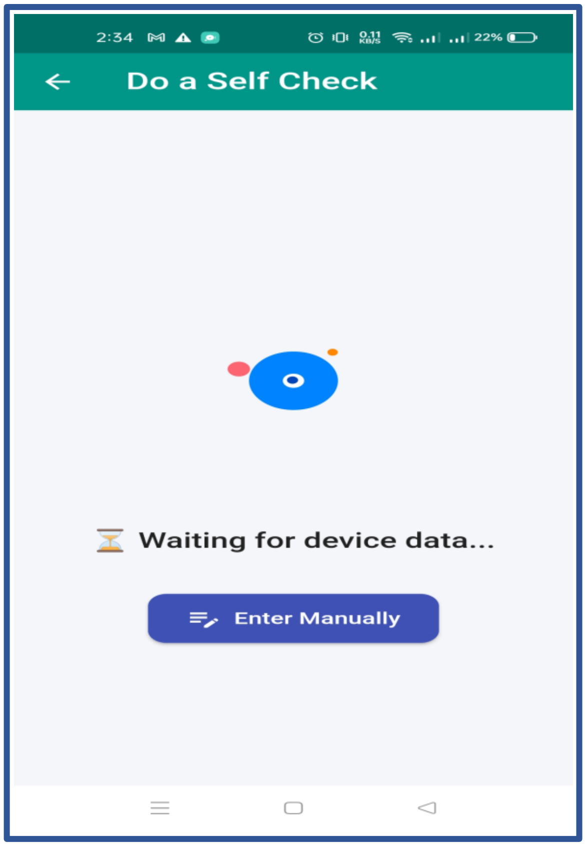


Figure 4.15 Self-Check Screen

In this screen, the application begins receiving real-time medical data via Bluetooth from the Arduino-based wearable device. Once the data is collected, it is sent to a Flask API hosted on Google Cloud, where two AI models process it to predict heart disease and stroke risk. The combined result is then returned and shown to the user.

“Chatbot Screen”

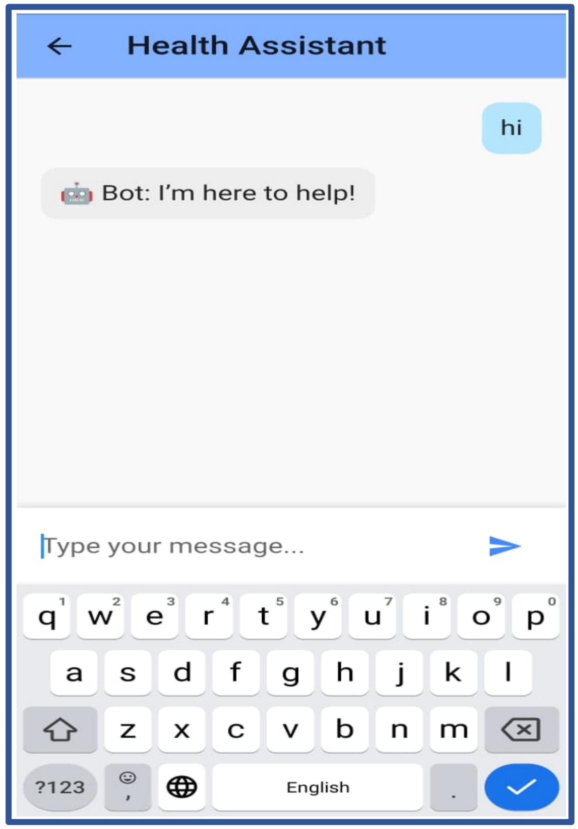


Figure 4.16 Chatbot Screen

This screen provides a chatbot interface where users can type messages and receive instant replies. The bot is designed to assist users with navigation, health tips, and general inquiries about the app’s features, enhancing overall accessibility and user support.

“Settings Screen”

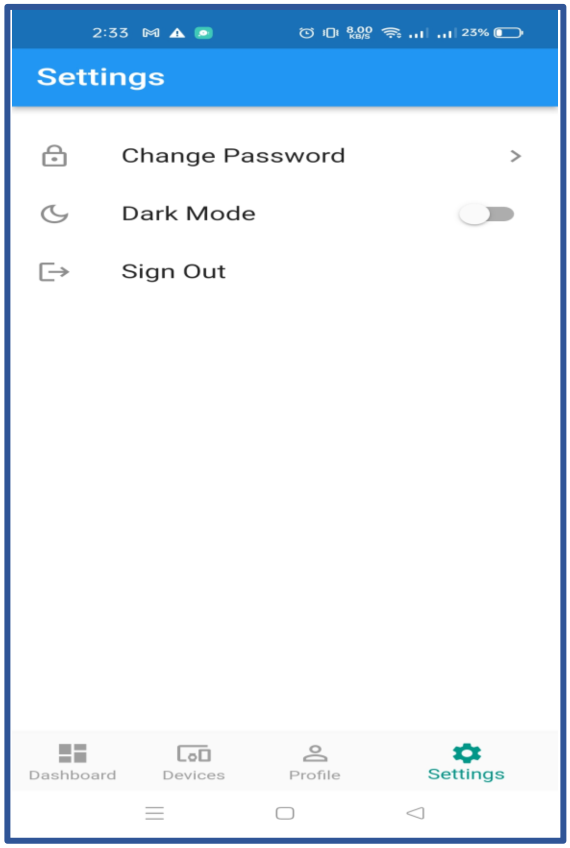


Figure 4.17 Settings Screen

This screen provides users with additional controls, including the ability to change their password, toggle dark mode for better accessibility, and securely sign out from the application. These settings enhance personalization and user control over app experience.

4.5 Embedded System Integration

The hardware module focuses on measuring key health parameters including heart rate, blood oxygen saturation (SpO2), and electrocardiogram (ECG) signals. These indicators are essential for assessing cardiovascular health and can be used to detect abnormalities that may precede a stroke event. A combination of sensors, display units, user controls, and wireless communication modules are integrated to provide a compact, real-time monitoring device.

4.5.1 System Components

The hardware design incorporates the following main components:

- **Microcontroller**: Arduino Nano 33 IoT
- **Sensors**:
  - MAX30102: for measuring heart rate and blood oxygen saturation
  - ECG sensor: for capturing the electrical activity of the heart
- **Display**: OLED screen for real-time data visualization
- **Feedback device**: Buzzer to provide audible heartbeat signals
- **Control interface**: Three push buttons to select and activate specific sensors

- **Wireless communication**: Bluetooth Low Energy (BLE) integrated on Arduino Nano 33 IoT for sending data to a mobile application

Each component is chosen for its compatibility, size, and low power consumption, making the system suitable for wearable or portable health monitoring.

4.5.1.1 Arduino Nano 33 IoT

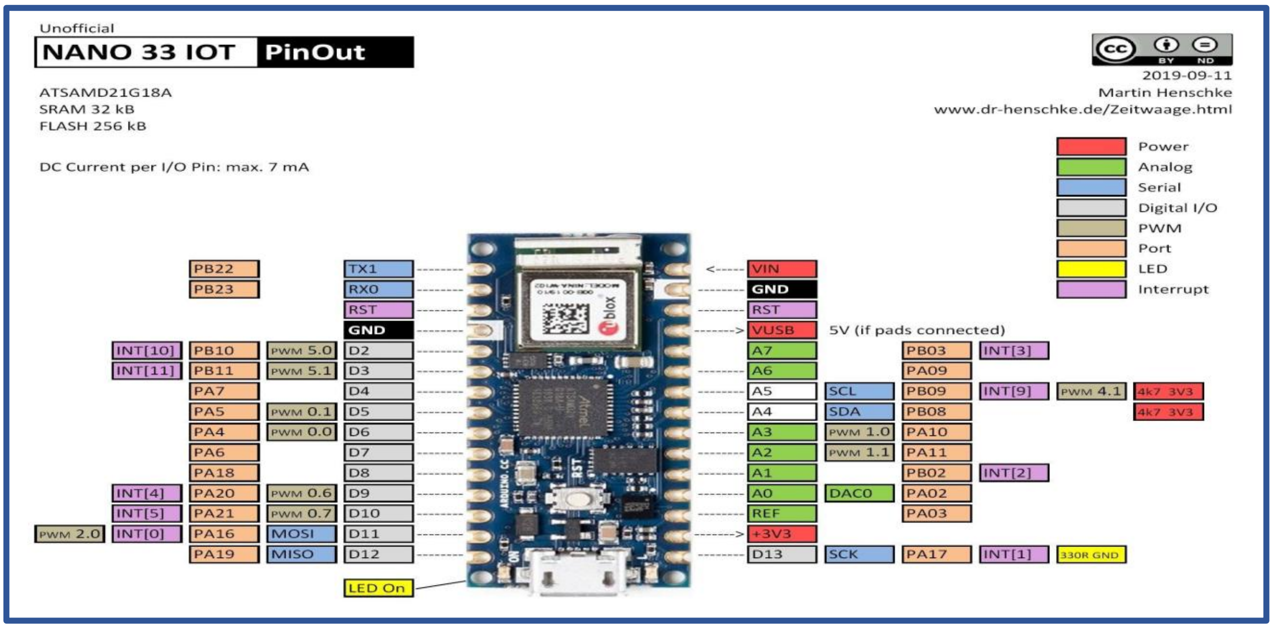


Figure 4.18 Arduino Nano 33

The Arduino Nano 33 IoT is a compact and powerful microcontroller board based on the SAMD21 Cortex-M0+ 32-bit low power ARM MCU. It offers multiple features that make it ideal for wearable and IoT health monitoring projects:

- **Processor**: 32-bit ARM Cortex-M0+ running at 48 MHz
- **Memory**: 256 KB Flash memory and 32 KB SRAM

- **Connectivity**: Integrated Wi-Fi and Bluetooth Low Energy (BLE) module based on the u-blox NINA-W102 module
- **Size**: Small form factor (45 x 18 mm), suitable for portable devices
- **Power**: Supports 3.3V logic and can be powered via USB or battery, enabling low power consumption modes
- **Interfaces**: Multiple digital and analog I/O pins, I2C, SPI, UART, and ADC channels for sensor integration

4.5.1.2 Sensor Descriptions and Display & Feedback Components

1) MAX30102 Sensor

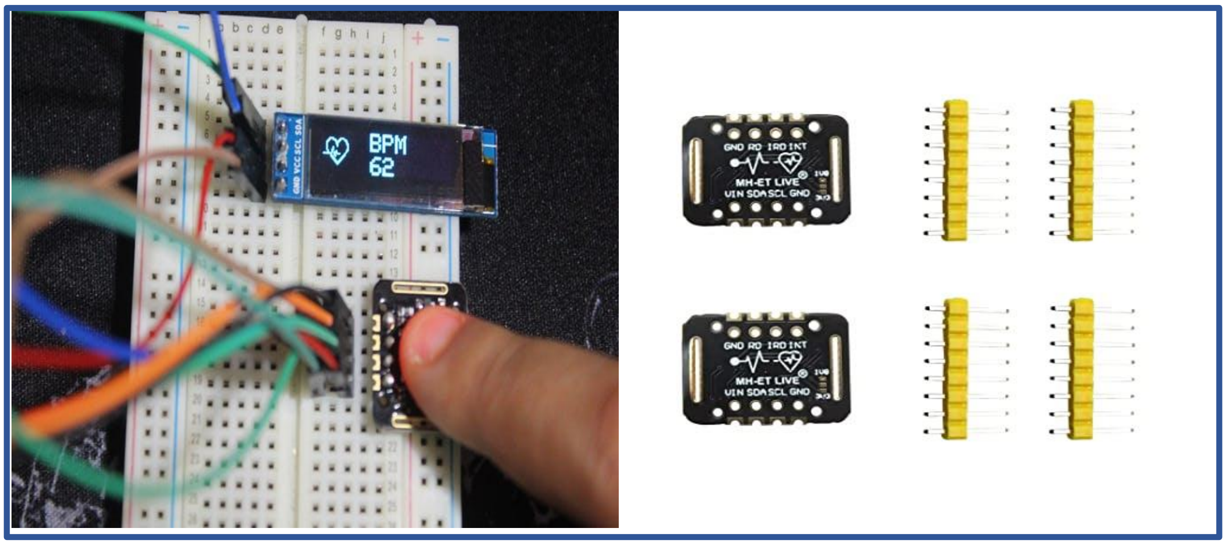


Figure 4.19 MAX30102 Sensor

The MAX30102 is an integrated pulse oximetry and heart rate monitor module. It uses photo detectors and LEDs to measure the absorption of red and infrared light through the skin, which varies with blood volume changes caused by heartbeats.

- **Measurement Parameters**:
    - **Heart Rate (BPM)**: Calculated by detecting the pulse waveform peaks
    - **Blood Oxygen Saturation (SpO2)**: Estimated based on the ratio of absorbed red and infrared light
- **Interface**: Communicates via I2C protocol with the microcontroller
- **Advantages**: Compact size, low power, and high sensitivity
- **Output**: Digital signals representing heart rate and SpO2 values

2) ECG Sensor

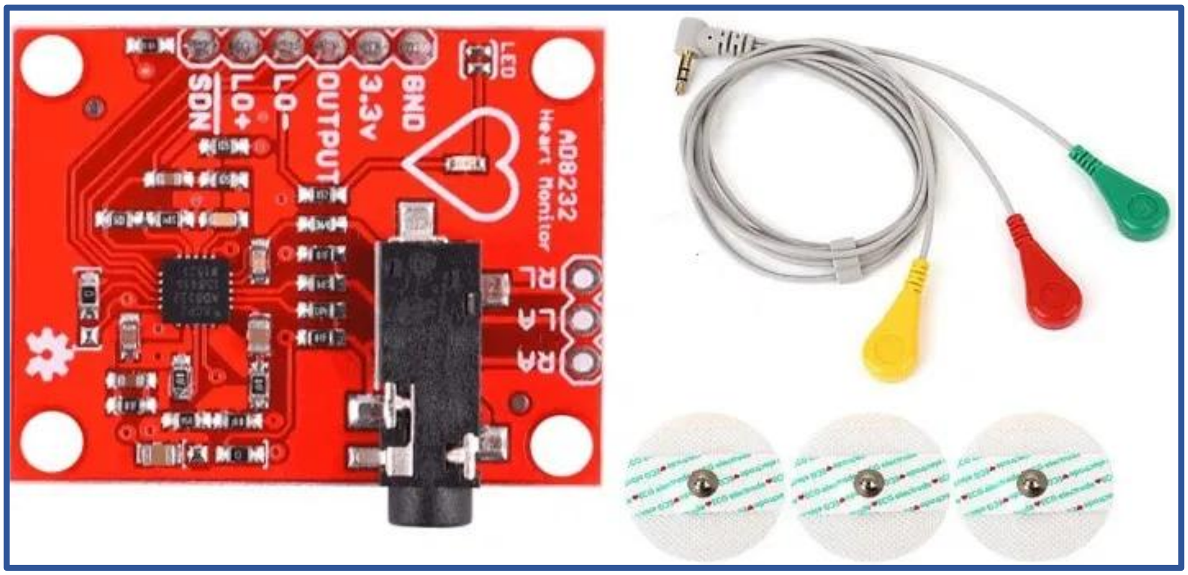


Figure 4.20 ECG Sensor

The ECG sensor captures the electrical activity generated by the heart's muscle contractions. Electrodes attached to the body detect voltage changes, which are then amplified and digitized.

- **Measurement**: Provides real-time electrocardiogram signals
- **Use**: Helps detect arrhythmias and other cardiac abnormalities that might indicate stroke risk
- **Interface**: Analog or digital output connected to Arduino analog input pins or via an ADC
- **Signal Characteristics**: Requires filtering and processing to remove noise and baseline drift

3) OLED Display

Figure 4.21 OLED Display

The OLED display used is typically a small (e.g., 128x64 pixels) monochrome screen that communicates over the I2C interface. It offers:

- **Advantages**:
    - High contrast and wide viewing angles
    - Low power consumption

- Compact size, ideal for wearable devices

- **Displayed Data**:
  - Current heart rate in beats per minute
  - Blood oxygen saturation percentage
  - ECG waveform visualization (real-time graph or simplified representation)

5) Buzzer

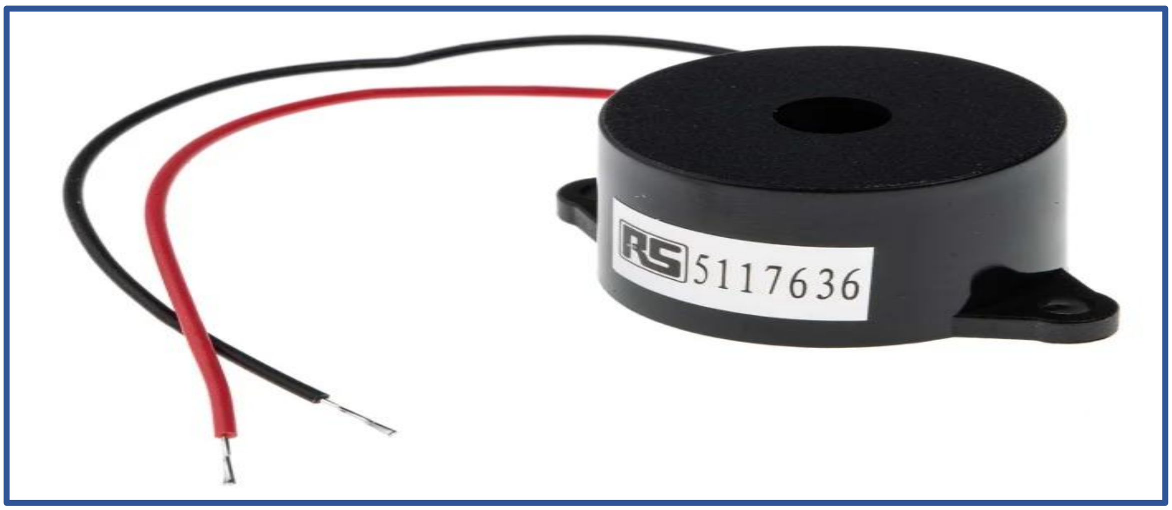


Figure 4.22 Buzzer

The buzzer generates an audible sound synchronized with each heartbeat detected by the MAX30102 sensor. This feedback helps the user perceive their pulse in real-time without looking at the display.

- **Type**: Piezo buzzer
- **Operation**: Triggered by the microcontroller each time a heartbeat is detected
- **Purpose**: Immediate physiological feedback and alerting mechanism

4.5.2 Control Interface

4.5.2.1 Push Buttons

The system includes **three push buttons**, each assigned to activate a specific sensor or function. This design allows the user to selectively monitor different physiological parameters based on their needs.

- **Button 1**: Activates the MAX30102 sensor module heart rate measurement
- **Button 2**: Activates the ECG sensor for capturing cardiac electrical signals
- **Button 3**: Activates the MAX30102 sensor module SpO2 measurement

4.5.2.2 Button Debouncing and Handling

To ensure reliable operation, software debouncing techniques are applied to prevent false triggers caused by mechanical bounce. The microcontroller continuously polls the button states and executes corresponding sensor activation routines based on user input.

4.5.3 System Integration

4.5.3.1 Data Acquisition Flow

Upon pressing a button, the Arduino Nano 33 IoT initializes communication with the corresponding sensor via I2C (for MAX30102) or analog input (for ECG). Sensor data is collected in real-time and processed using embedded algorithms.

4.5.3.2 Signal Processing Overview

Raw sensor signals undergo filtering to reduce noise:

- **MAX30102**: Digital filtering and peak detection algorithms compute BPM and SpO2 values
- **ECG**: Analog signals pass through hardware filters, then software filters (e.g., moving average) to extract meaningful waveform data

4.5.3.3 Display and Buzzer Synchronization

- The OLED display updates continuously with current measurements or waveforms.
- The buzzer produces a sound synchronized with each heartbeat detected by the MAX30102 sensor, providing auditory feedback.

4.5.3.4 Bluetooth Data Transmission

The Arduino's built-in BLE module streams the processed data wirelessly to a mobile application. This enables remote monitoring, data logging, and alerts based on threshold breaches.

4.5.4 Device Design and Structure

The embedded system is implemented inside a compact box designed for portability and real-time health monitoring.

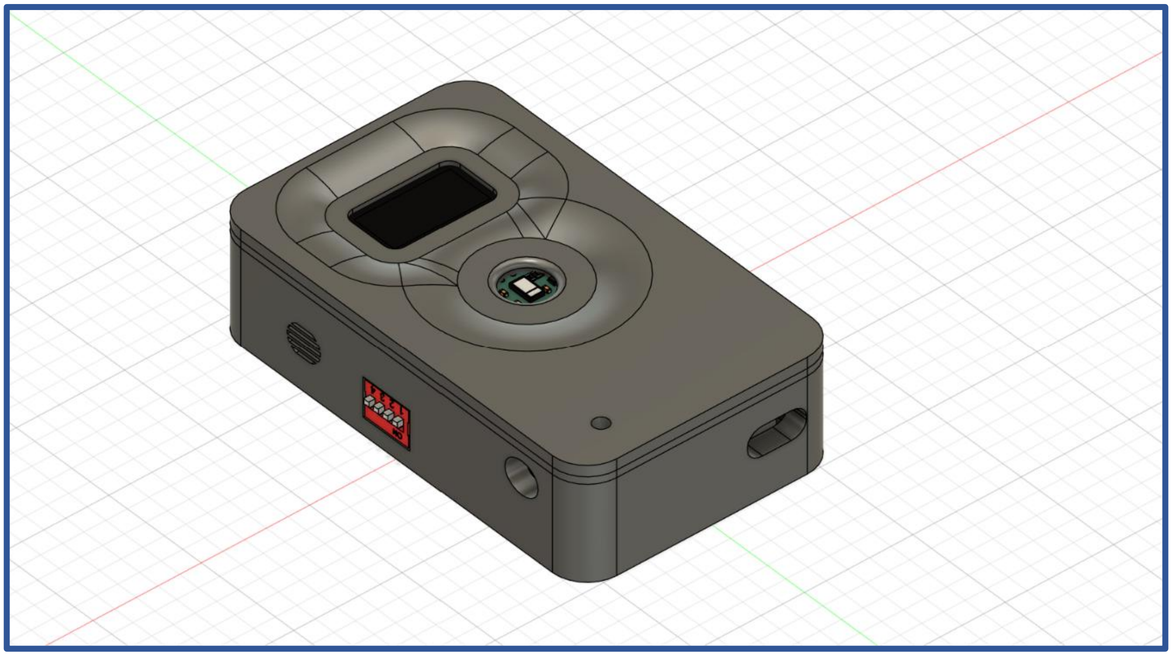


Figure 4.23 Device Box Design

All hardware components are neatly integrated to function seamlessly as a unified device.

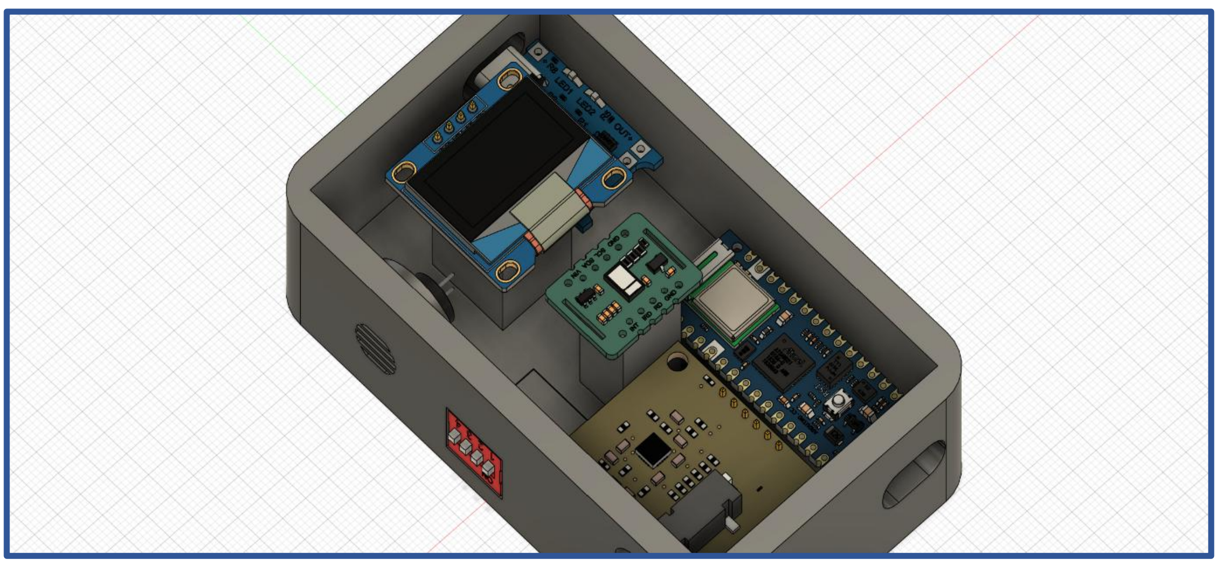


Figure 4.24 Integrated Hardware Components

Chapter Five:
Conclusion & Future Work

Chapter Five: Conclusion & Future Work

5.1 Conclusion

Our project highlights the development and significance of a mobile application designed to revolutionize healthcare management. This app offers an intuitive platform for users to monitor their health metrics and assess stroke risk effectively. By integrating functionalities such as user-friendly registration, seamless device connectivity, manual data input, AI-driven risk analysis, and a chatbot for medical support, the application provides a holistic healthcare solution.

The project underscores the importance of making healthcare accessible and interactive. Users can measure critical health parameters, such as ECG, SPO2, blood pressure, and heart rate, through connected devices or manual entry. These metrics are analyzed, and stroke risk is calculated in real time, ensuring users are informed and empowered to take proactive steps in their healthcare journey. The chatbot further enhances the experience by providing personalized guidance and support.

This mobile application represents a practical and impactful solution for health monitoring, combining ease of use with advanced technology. By streamlining health data collection, analysis, and feedback, it offers users a reliable and efficient tool for managing their well-being. The project serves as a testament to our commitment to creating innovative and user-focused solutions that address real-world healthcare needs.

5.2 Future Work

We're excited about the future expansion of our healthcare application, aiming to enhance health monitoring, accessibility, and real-time support. Our vision includes several impactful features:

1. **Integration of Emergency Response System:**
   We plan to add an emergency alert system that notifies caregivers or hospitals when stroke risk exceeds a critical threshold. This ensures fast medical response and improved patient safety.
2. **Advanced Wearable Compatibility:**
   Support for more wearable devices will be added to capture additional health metrics, including blood pressure and oxygen saturation in real time.
3. **Medical History and Doctor Portal:**
   A dedicated portal for doctors to review patients' medical history, monitor live data, and offer guidance remotely will be introduced to close the loop between patient and physician.

References

References

1. **Khosla, A., Cao, Y., Lin, C. C., Chiu, H. K., Hu, J., & Lee, H.** (2010). *An integrated machine learning approach to stroke prediction.* Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. https://dl.acm.org/doi/10.1145/1835804.1835830

2. **American Stroke Association (ASA)**. *Understanding Stroke: Symptoms, Risk Factors, and Prevention.* https://www.stroke.org/en/about-stroke

3. **Benjamin, E. J., Virani, S. S., Callaway, C. W., Chamberlain, A. M., Chang, A. R., Cheng, S., ... & Muntner, P.** (2018). Heart disease and stroke statistics—2018 update: a report from the American Heart Association. *Circulation*, *137*(12), e67-e492. https://doi.org/10.1161/CIR.0000000000000558

4. **Feigin, V. L., Norrving, B., & Mensah, G. A.** (2017). *Global Burden of Stroke.* Circulation Research, 120(3), 439–448. https://doi.org/10.1161/CIRCRESAHA.116.308413

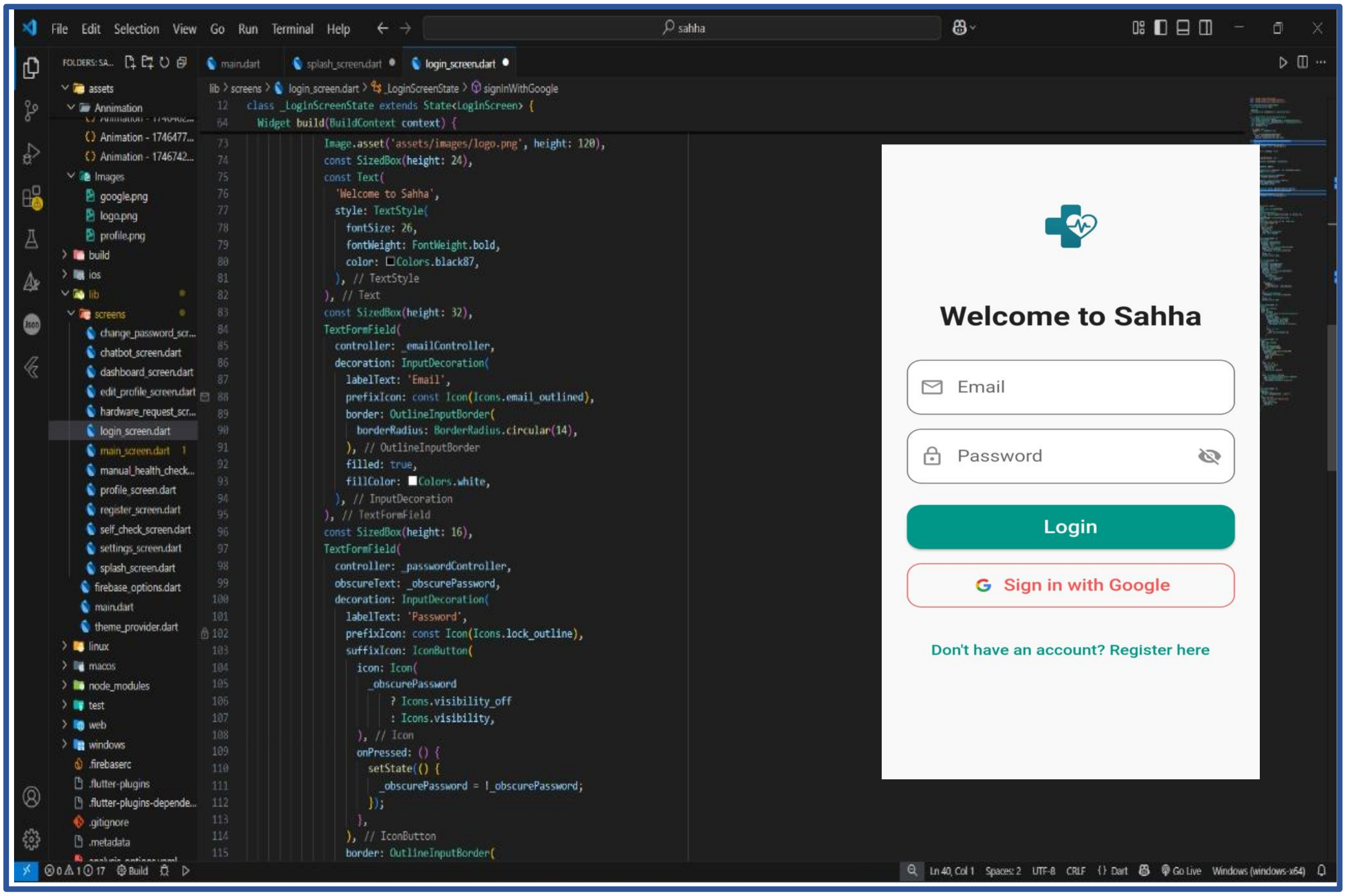
5. **Dev, S., Wang, H., Nwosu, C. S., Jain, N., Veeravalli, B., & John, D.** (2022). A predictive analytics approach for stroke prediction using machine learning and neural networks. *Healthcare Analytics*, *2*, 100032. https://doi.org/10.1016/j.health.2022.100032

6. **World Stroke Organization (WSO)**. *Stroke Prevention and Control Strategies.* https://www.world-stroke.org/world-stroke-day-campaign/prevent-stroke

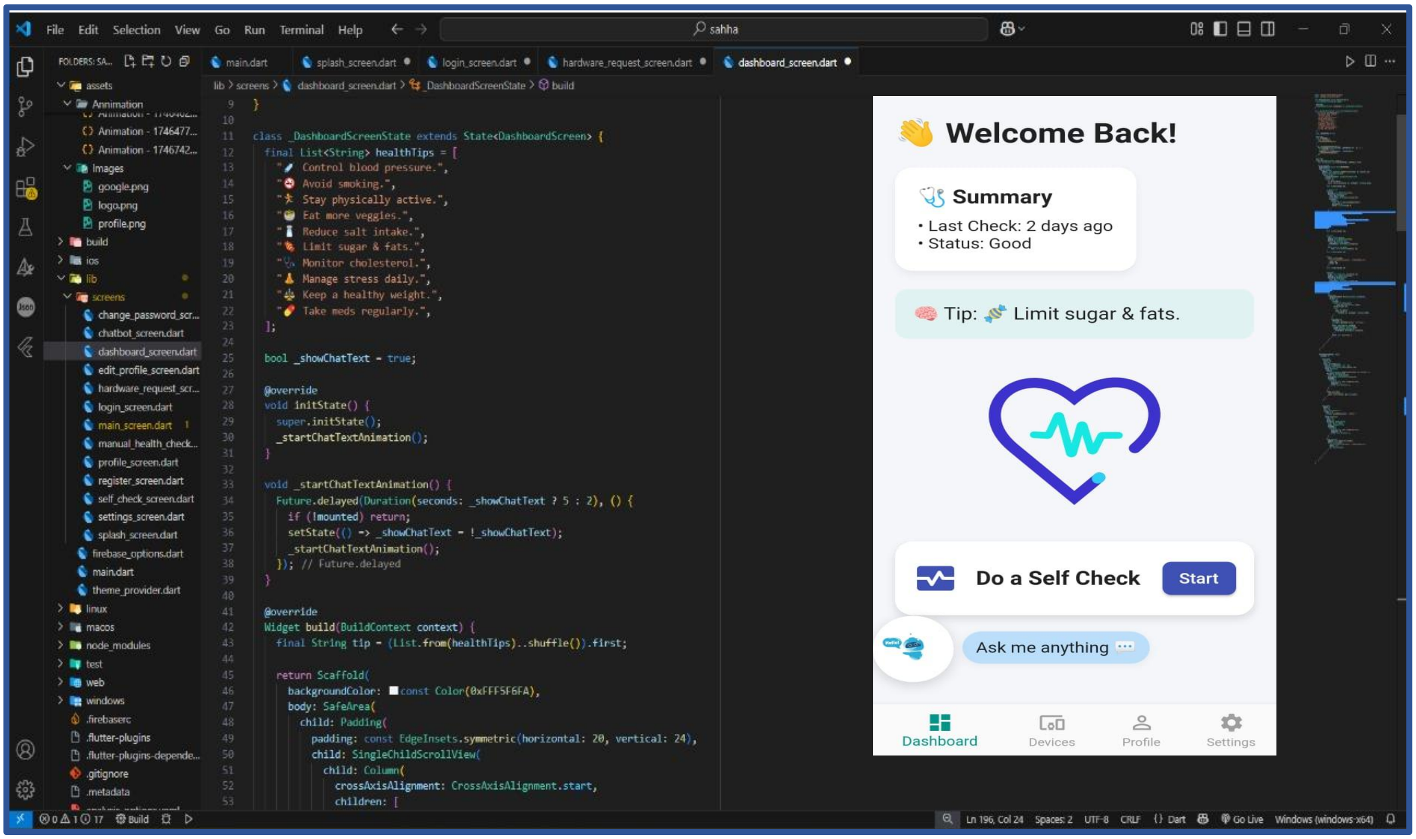
Appendices

Appendices

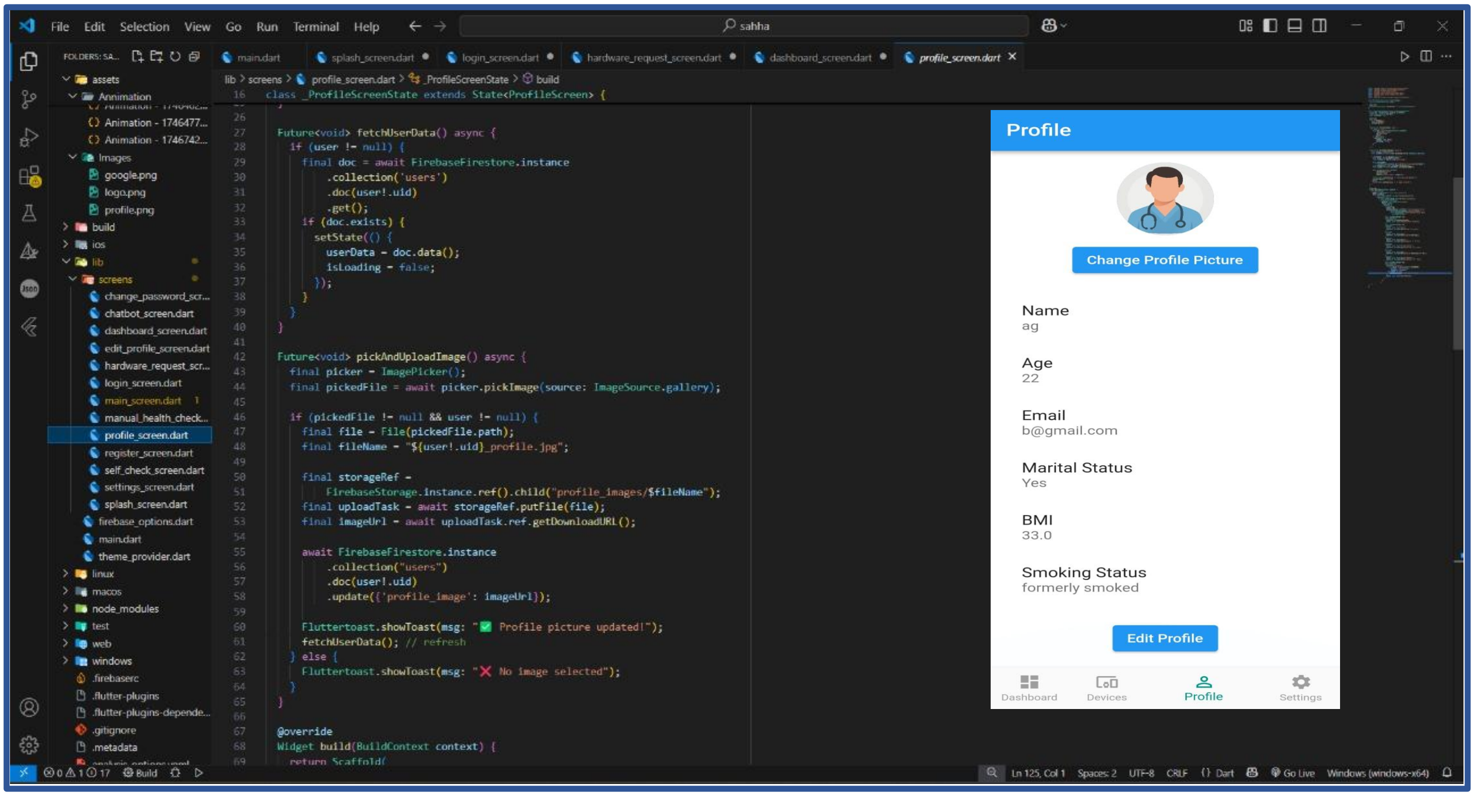
Appendix A ("Login Page")



Appendix B (“Dashboard Page”)



Appendix C ("Profile Page")



Appendix D (“Resampling data by SMOTE)

SMOTE

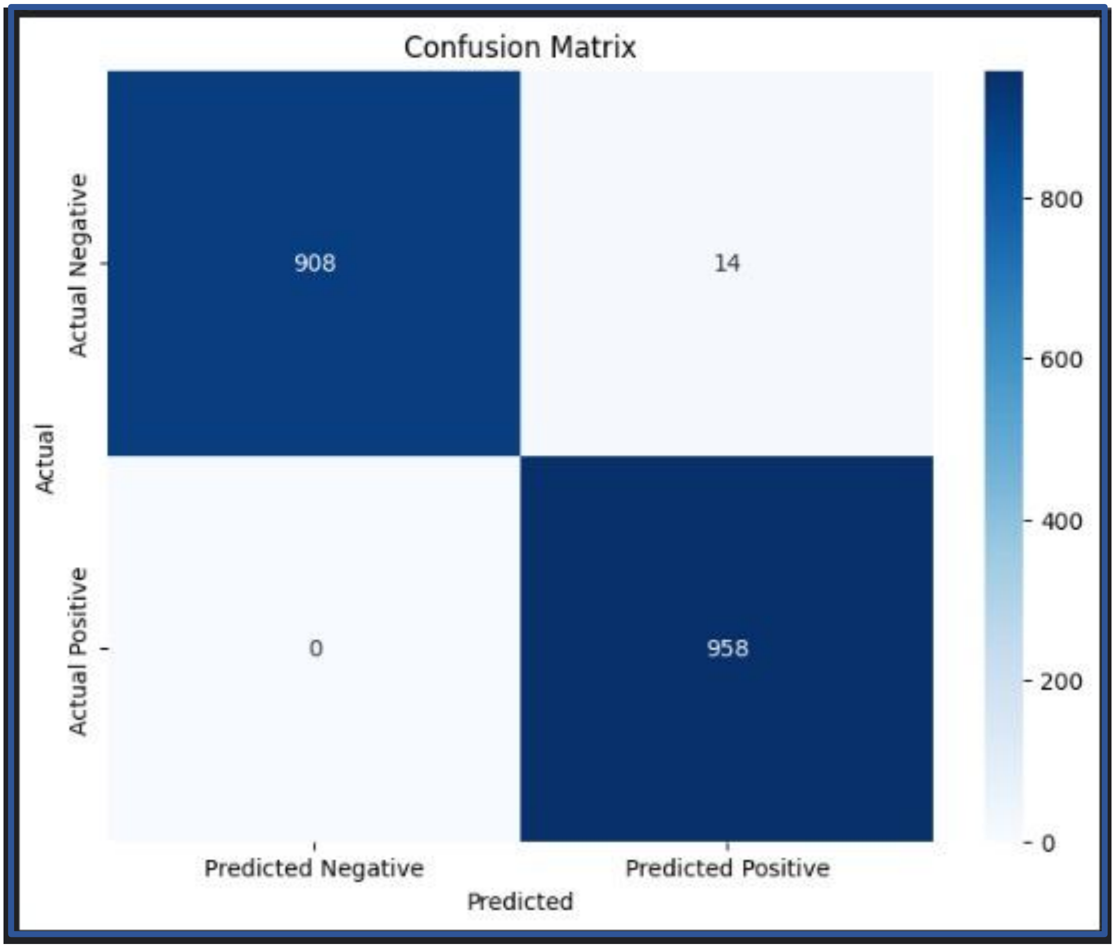
+ Code + Markdown

```
smote = SMOTE(random_state=42)
X_smsampled, y_smsampled = smote.fit_resample(X, y)#
```

```
evaluate_models(X_smsampled, y_smsampled)
```

	Model	Accuracy	Precision	Recall	F1
0	Logistic Regression	0.789362	0.789348	0.789362	0.789341
1	Decision Tree	0.912234	0.912233	0.912234	0.912233
2	Random Forest	0.950532	0.950662	0.950532	0.950519
3	Gradient Boosting	0.886702	0.886746	0.886702	0.886682
4	AdaBoost	0.853191	0.855065	0.853191	0.852867
5	Support Vector Machine	0.833511	0.835291	0.833511	0.833135
6	K-Nearest Neighbors	0.915957	0.924933	0.915957	0.915382
7	Naive Bayes	0.772872	0.772877	0.772872	0.772875
8	XGBoost	0.951596	0.951597	0.951596	0.951596

Appendix E (Confusion Matrix of the Model)



Appendix F (Chatbot Response)

```
[3] import gradio as gr
    import torch
    from unsloth import FastLanguageModel

    # --- Load Model ---
    max_seq_length = 4096
    dtype = torch.float16
    load_in_4bit = True

    # Load the model from the Hugging Face Hub
    model, tokenizer = FastLanguageModel.from_pretrained(
        model_name = "Ahmed-El-Sharkawy/Stroke-medical-model-finetuned",
        max_seq_length = max_seq_length,
        dtype = dtype,
        load_in_4bit = load_in_4bit,
    )
```

Medical LLaMA-3 Fine-tuned Chatbot

Input

I'm 70 years old; I have a headache. are these symptoms ?

Response

For older adults, a sudden severe headache, especially if accompanied by confusion, weakness, or altered consciousness, could be a sign of a hemorrhagic stroke and requires urgent medical attention.

Appendix G (BLE Initialization Code)

```
// BLE definitions
BLEService healthService("180D");
BLEFloatCharacteristic bpmCharacteristic("2A37", BLERead | BLENotify);
BLEFloatCharacteristic spo2Characteristic("2A5F", BLERead | BLENotify);
BLEIntCharacteristic ecgCharacteristic("2A58", BLERead | BLENotify);

void setup() {
  Serial.begin(115200);

  pinMode(BUTTON_PIN_1, INPUT_PULLUP);
  pinMode(BUTTON_PIN_2, INPUT_PULLUP);
  pinMode(BUTTON_PIN_3, INPUT_PULLUP);
  pinMode(BUZZER_PIN, OUTPUT);

  if (!BLE.begin()) {
    Serial.println("Starting BLE failed!");
    while (1);
  }
```

Appendix H (SpO_2 Value Calculation)

```
  if (irValue > 7000) {
    float ratio = (float)redValue / (float)irValue;
    spo2Value = 104 - (17 * ratio);

    spo2Characteristic.writeValue(spo2Value);  // Send to BLE

    Serial.print("SpO2: ");
    Serial.print(spo2Value);
    Serial.println(" %");

    display.setCursor(0, 0);
    display.println("SpO2 Measuring...");
    display.setTextSize(2);
    display.setCursor(10, 30);
    display.print(spo2Value);
    display.println("%");
  } else {
    display.setCursor(10, 20);
    display.println("Place Finger SpO2");
  }
  display.display();
  delay(1000);
}
```

Appendix I ("Embedded OLED Display")

```
BLE.setLocalName("3hmed_3li");
healthService.addCharacteristic(bpmCharacteristic);
healthService.addCharacteristic(spo2Characteristic);
healthService.addCharacteristic(ecgCharacteristic);
BLE.addService(healthService);
BLE.advertise();

if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
  Serial.println(F("OLED init failed"));
  while (true);
}

display.clearDisplay();
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(0, 0);
display.println("Initializing...");
display.display();
delay(2000);
```