Health Monitoring System Using IoT with Machine Learning

A Project Report

submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of the requirements for the award of degree of

Bachelor of Technology

in

Computer Science and Engineering

by

MUHAMMAD RASI K P(STM19CS030)

NUHA SALIH(STM19CS040)

NANDITHA P(STM19CS038)

VAISHNAVI RAJEEVAN(STM19CS057)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ST. THOMAS COLLEGE OF ENGINEERING AND TECHNOLOGY

MATTANNUR

May 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ST. THOMAS COLLEGE OF ENGINEERING AND TECHNOLOGY

MATTANNUR

2022 - 2023

CERTIFICATE

This is to certify that the report entitled **Health Monitoring System Using IoT with Machine Learning** submitted by **MUHAMMAD RASI K P** (STM19CS030) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the project work carried out by them under my guidance and supervision. This report in any form has not been submitted to any other university or institute for any purpose.

Internal Supervisor	**External Supervisor**
Project coordinator	**Head of Department**

DECLARATION

I hereby declare that the project report **Health Monitoring System Using IoT with Machine Learning**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of **Mrs. ANU C**

This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources.

I also declare that, I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the university and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other university.

MUHAMMAD RASI K P

Mattannur

15-05-2023

ACKNOWLEDGEMENT

I take this opportunity to express my deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. I are extremely thankful to my Principal **Dr. SHINU MATHEW JOHN** for giving us his consent for this project.

I express my sincere thanks to **Mrs. AMITHA I C**, Head of the Department, Department of Computer Science and Engineering for providing us with all the necessary facilities and support.

I would like to express my sincere gratitude to the Project coordinator, **Mr. JITHIN S**, Asst. Professor, Department of Computer Science and Engineering for the support and co-operation.

I would like to place on record my sincere gratitude to our project guide **Mrs. ANU C**, Asst. Professor, Department of Computer Science and Engineering for the guidance and mentorship throughout this work.

Finally I thank my family, and friends who contributed to the succesful fulfillment of this Project work.

MUHAMMAD RASI K P

Abstract

The Internet of Things (IoT) is a young technology that is drastically improving, with many new improvements in the medical and health sectors. Health care is an important part of human life. It helps prevent diseases and improve the quality of life. Today, people don't have the time to visit the hospital for routine checkups and maintain a medical record of their health status. To overcome this, we introduced a system called the Health Monitoring System. It is an IoT framework for healthcare monitoring systems integrated with ML approaches so that patients' health status can be monitored periodically and regularly. The patient's data are collected using an IR and blood pressure sensor, compared using the machine learning SVM algorithm, and then it is determined if the patient has diabetes or not. Using this data, we also try to predict the percentage chance of kidney failure in a person due to uncontrolled diabetes.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Human lifestyle is based on many factors such as irregular food habits, no nutrition diet, environmental pollutions, lack of proper exercise, indefinite work, restlessness, and increases stress levels which leads to greater disaster in human health. The modern visionary of healthcare industry is to provide better healthcare to patient anytime in thais world in a more economic and patient friendly manner. Therefore for increasing patient care efficacy, there exist a need to improve patient monitoring devices. Internet of Things (IoT) is an emerging technology that is drastically improving with many new enhancements in medical and health domains. Human lifestyle is based on many factors such as irregular food habits, no nutrition diet, environmental pollutions, lack of proper exercise, indefinite work, restlessness, and increases stress levels which leads to greater disaster in human health. The modern visionary of healthcare industry is to provide better healthcare to patient anytime in thais world in a more economic and patient friendly manner. Therefore for increasing patient care efficiancy, there exist a need to improve patient monitoring devices. The Internet of Things (IoT) describes the network of physical objects—“things”—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. Internet of Things (IoT) is an emerging technology that is drastically improving with many new enhancements in medical and health domains. The IoT health-monitoring platform has provided us with a significant benefit in the advancement of contemporary medicine. IoT devices are widely used in the medical sector. IoT Health wearable devices are taking new challenges by upgrading with innovative technology and resources. Using health wearable devices, in/out patient’s health status can be monitored periodically and regularly. This project introduces an IoT application framework E-Healthcare Monitoring System (EHMS) integrated with Machine learning (ML) techniques to design an advanced automation system. Using this system it will connect, monitor and decisions making for proper diagnosis. At present authors suggested a framework E-Health care monitoring system. It is basically an application model design with various machine learning models. Here IoT wearable sensor collects data from the patient. The collected data from numerous health wearable devices are processed to E-Health care monitoring system. Later EHMS applies machine learning approaches on raw data to analyze the health condition of the patient for better decision making and diagnosis. Disorder analysed report, doctor appointment will acknowledge to the patient, doctor, and caretakers. IoT Health wearable devices are taking new challenges by upgrading with innovative technology and resources. Using health wearable devices, in/out patient's health status can be monitored periodically and regularly. This project introduces an IoT application framework E-Healthcare Monitoring System integrated with Machine learning (ML) techniques to design an advanced automation system.

1.1 Background

Health monitoring is the major problem in today's world. Due to lack of proper health monitoring, patient suffer from serious health issues. Also most humans live a busy life in which going to a doctor for weekly or even monthly checkup is an impossible task. Without monitoring your health it is not possible to whether you are a healthy or sick person. This problem leads to the design of a product which monitors your health every day without going to a doctor.There are lots of IoT devices now days to monitor the health of patient over internet. Health experts are also taking advantage of these smart devices to keep an eye on their patients. With tons of new healthcare technology start-ups, IoT is rapidly revolutionizing the healthcare industry.

1.2 Existing System

Using modern technological tools such as Internet of Things (IoT), machine learning and Artificial Intelligence along with Big data makes the job of physicians much easier in digging out the root cause of disease and predicting its seriousness using modern algorithms. In this research work, the machine learning algorithms are used to monitor the health conditions of the humans. Initial training and validation of machine learning algorithms are performed using the UCI dataset. Testing phase is carried out by collecting heart rate, blood pressure and temperature of the person using IoT setup. Testing phase estimates the prediction of any abnormalities in the health condition from the sensor data collected through the IoT framework. Statistical analysis is performed from data accumulated into the cloud from IoT device to estimate the accuracy in prediction percentage. Also, from the results obtained from the K-Nearest Neighbor outperforms other conventional classifiers.

1.3 Problem Definition

Nowadays the cause of unhealthy human lifestyle is based on many factors such as irregular food habits, no nutrition diet, environmental pollutions, lack of proper exercise, indefinite work and Since it is impossible for us to go to the doctor everyday for a daily checkup as well as maintain a proper health record or keep a track of our health. We introduced this Health Monitoring system in order to keep a track of our health wherever we want using an iot wearable device. Also using this system reduces the stress in people to make time to go to doctors everyday to keep a track of their health and find time to maintain a healthy lifestyle.

Sometimes people cannot go to the hospital to do a checkup in their busy schedule if they have early signs of diabetes melitus as well as they will not know the symptoms of diabetes melitus and this can kill the person silently causing any kidney disease due to uncontrolled diabetes because of the unawareness. So in the proposed system we are doing daibetic prediction to check if the person is diabetic or not. This will help the person to take immediate measures if they have diabetes as well as help the person to take prior precautions before it leads to any harmful kidney disease due to uncontrolled diabetes.

1.4 Objectives

It will be a very handy tool as it shows all the data collection and information by using just only the internet. So, it reduces the workloads and stress of the relatives of the patient who work outsides. For being connected to the health care system through IoT, doctors can improve the diagnosisaccuracy as they are getting all the necessary patient data at hand. In a word, we can say that it allowsmonitoring patient continuously and remotely. By using this system, we can get approximate result based on patient health. Moreover, it will beless error, collect data in less time and more accuracy than any human performances. When a patient gets health service at home on a real time basis, there is no need for unnecessarydoctor or nursing visit. In particular, this project helps to cut down cost for hospital stays and readmissions. Through IoT, doctors and relatives can do their individual job without any hesitation as they canmonitor the patient's health condition from anywhere. Moreover, it will send alerts whenever a particular health parameter goes beyond the ideal limit.

Furthermore, by receiving SMS alert doctorsand relatives can take necessary action. Lastly, we can say that it saves lives in case of emergency.

1.5 Scope

The project focuses on the development of an IoT based health monitoring system that involves both hardware and software. The prototype system consists of sensors and a data processing broker. The prototype allows authorized users to monitor the health data and the location of the health system user through the internet. Besides, the authorized user can easily trace back to the previous health data by accessing to the database that stores all the user's health data.

Since the focus of this project is on the implementation and involvement of IoT in the health monitoring system and due to budget limitation, the accuracy of the sensors used in this system will not be taken into consideration as the sensors used in this system are not medically verified and are not suitable to be used for any serious medical analysis by any means.

Chapter 2

Literature Review

Before beginning a new study, a literature review establishes familiarity with and understanding of current research in a certain topic. It includes information on current projects, such as their characteristics, problems, and solutions. We picked four articles for survey: IoT based Health Monitoring & Automated Predictive System to Confront COVID-19, Smart-monitor: patient monitoring system for IoT-based healthcare system using deep learning, An Efficient Health Monitoring System with Temprature and Heart rate Sensors using IOT and E-Healthcare Monitoring System using IoT with Machine Learning Approaches. IoT based Health Monitoring & Automated Predictive System to Confront COVID-19 [1]. Smart-monitor: patient monitoring system for IoT-based healthcare system using deep learning [2]. An Efficient Health Monitoring System with Temprature and Heart rate Sensors using IOT [3]. E-Healthcare Monitoring System using IoT with Machine Learning Approaches [4].

2.1 IoT based Health Monitoring & Automated Predictive System to Confront COVID-19

The COVID-19 was first detected in the Wuhan of China's Hubei province in December 2019. In March 2020, the WHO (World Health Organization) announced the virus's massive outbreak as a pandemic. Since the outbreak of COVID-19 at an enormous scale, most countries worldwide had been declared lockdowns by their decision-makers. Such measures seemed necessary from the given perspective, yet at the same time, it has been causing a massive loss in the world economy. Most importantly, the outbreak couldn't be retained since its first detection in China. The continuous spread of the virus resulted in it becoming a pandemic. As of the second week of September 2020, the virus has affected 213 countries with an infected count of more than 29,095,917 peoplenand a death toll of approximately 926,824 human lives. Such huge numbers could have been avoided if specific measures were taken and if proper health monitoring systems were used to treat these people. On the contrary, most of these cases received no treatment at all. A theory has become quite established and proven that people with other severe illnesses are most likely to lose lives due to the COVID-19 infection. Therefore, there have been quite a few cases of great confusion about whether a patient is displaying symptoms because of the virus or symptoms of other diseases. This project represents a brief idea to confront the spread of such an illness using the concept of IoT. The Internet of Medical Things (IoMT) is a corresponding field of IoT working in the sector of Medicine & Biology. Such a research field can be helpful in the fight against COVID-19 and can avail healthcare service to the patients remotely where possible. After the virus outbreak, some airports have used AI to detect suspected COVID-19 cases on humans through image processing, thermal scanning, computer vision, and Big Data. For humans to confront a pandemic like COVID-19 requires to not be in physical contact with the infected persons. This is the scenario where the proposed system can be implemented to reduce the risk of the healthcare service providers getting infected as well as being able to respond effectively in time. The goals of the developed system are set out to implement an effective health monitoring system for COVID-19 patients that can provide real-time biological information and also can establish a communication channel between the patients and healthcare professionals while maintaining a physical distance. Also, the system has been developed with the target to calculate the possible rate of coronavirus present in a person's body. The overall contribution of the paper can be summarized as follows,

- The paper represents a system that provides a brief idea of how an IoT based health monitoring system can be established to confront diseases like the COVID-19.
- The paper combines an ML strategy to predict the severeness of the patients. The data is aggregated using IoT devices, and the system can maintain realtime communication with the patients.
- The paper further validates the method using real-time data executed in the cloud platform.

Various compartmental models have been introduced to prevent pandemics like COVID-19. Compartmental models generally construct a simple mathematical foundation that projects the spread of infectious disease and detects the severity of the disease in the patient using external symptoms. Ordinary Differential Equations (ODEs) are the foundation of these compartmental models. ODE and other mathematical methods are sufficient in modeling an infectious disease, but they misjudge the randomness of being infected, cured, and dead. Therefore, we implemented an IoT based environment that solves these issues by uploading real-time biological data of the patient to the program developed in the cloud network. The sensors and processing units upload the data with a specific IP address for the particular cloud storage. These data are then taken into the compartmental model, which are then processed and analyzed.

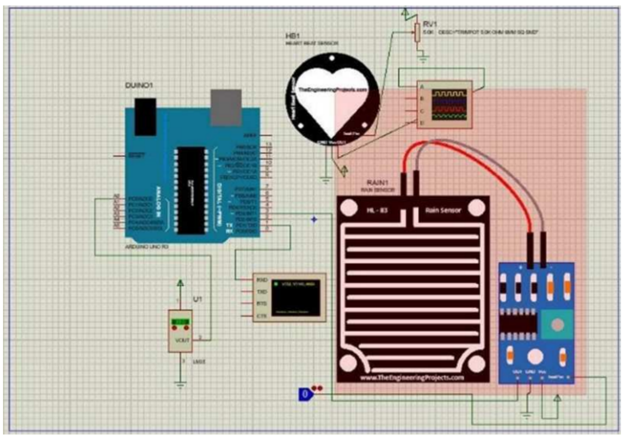


Figure 2.1: Embedded Multiple Sensors

1. **Data Pre-processing:-**

   In this step, we preprocess the dataset to feed the data to the Machine Learning algorithms. Firstly, we convert the whole dataset to a Pandas data frame structure. Then we drop less important feature values from the data frame. After that, both the feature variables and the target variables are converted into arrays of floating-point numbers so that the algorithms can understand them. Besides, we also split the arrays into training and validation sets to properly evaluate the Machine Learning pipeline's performance.

2. **Baseline Architecture:-**

   A supervised learning strategy is implemented to train ML algorithms. As the dataset contains binary values, regression models perform better in this case. Regression models are designed based on algorithms that analyze continuous data. The algorithms used in this step are state-of-the-art classifiers implemented using Scikit-Learn.

3. **Evaluation Matrix:-**

   The evaluation of the proposed model has been done by the accuracy evaluation metric that is based on the confusion matrix. A confusion matrix abridges the outputs received from the prediction measuring the performance of the machine learning classification algorithms that are composed of four measures: true positive (TP), true negative (TN), false positive (FP), and false-negative (FN). The Accuracy shows how often the classifier is correct. Accuracy is calculated by the following relation,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \ldots (1)$$

4. **Experimental Setup:-**

   In our cloud network model, Python is used for managing data, preprocessing, experimenting, and evaluating the model. For further analysis and assessment, we have tested some neural network architectures implemented in Keras. But they did not perform up-to-the-mark compared to the regression-based classifiers. Therefore, we have chosen to depend on the best performing algorithms. NumPy, a Python library, has been used to perform basic mathematical operations. Tensor-

Flow has been used to generate and control the whole pipeline's cloud infrastructure processing resources.

2.2 Smart-monitor: patient monitoring system for IoT-based healthcare system using deep learning

Reliable physiological signal monitoring is a conventional problem in healthcare system. Body Sensor Network (BSN)-based monitoring system is designed to have a secured healthcare system with timely assist. It is also vital to monitor and predict the signal deviation. By this IoT employed physiological signal monitoring system user can enable and predict individual condition like heart attack, chronic fever, elder care and also support, preventive measure and wellness. The quality of care can be improved reducing the patient travel problem, providing reliable assist when needed. The proposed IoT-based healthcare system employed deep learning algorithm with an intelligent sensor network to acquire a vital human physiological signal. The collected signal has shared a wireless medium to the central cloud server for analysis and visualization. IoT is an assistive technology for transport, infrastructure development, traffic monitoring, etc. From this advantage, IoT makes connection with the signal in modern healthcare system. In terms of healthcare system, it provides first connection and improved quality of life. Conceptual view of the proposed IoT based healthcare system for accurate signal monitoring. The primary focus of this work to implement deep learning for prediction. IoT is used for connecting sensor for acquiring continuous time series data and do data analysis in cloud-based.
The main contribution of the research paper is

- To go for an IoT-based signal acquiring system.
- To employ a deep learning model for prediction of physiological signal for data visualization and reliable assist from service provider.

The system includes an intelligent sensor for automatic continuous acquiring of physiological signal, wireless data transmission to the cloud server by the coordination of National Instruments myRIO processor with Wi-Fi module. NI-myRIO is capable of providing high-speed connection to all the things (intelligent sensor) connected to the signal monitoring system. The proposed learning architecture consists of the following four major modules.

1. **Monitoring Unit :** An intelligent sensor array with myRIO processor.
2. **Processing Unit :** myRIO wireless transmission using Wi-Fi module that was enabled.
3. **Visualization and Storage Unit :** IoT gateway for data visualization for process.
4. **Learning Unit :** Signal feature prediction and notification module.

The real-time implementation of the proposed SmartMonitor IoT framework using EEG, ECG, pulse rate, temperature, and pressure sensors is shown in Figure 2.2.

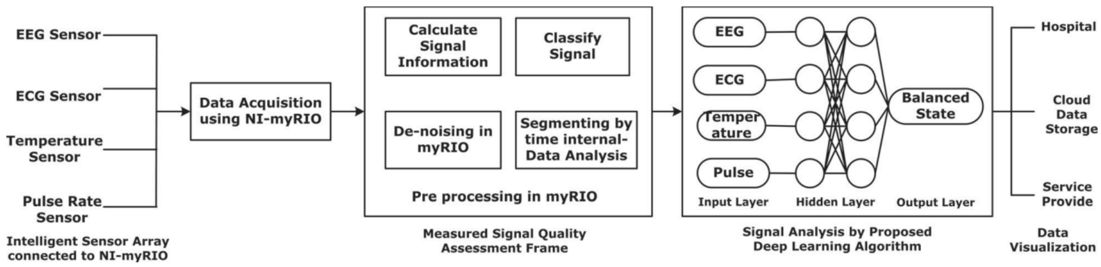


Figure 2.2: Signal Log system interconnection for classification

Proposed deep Learning Model for signal classification for learning the acquired patient physiological signal and to classify in cloud-based IoT system, we have proposed a Deep Neural Network (DNN) to extract features of the acquired signal in the sensor array. Based on training with standard signal a DNN structure is used by the scoresearch approach. In this approach, we compute a scoring function to search for constraint-based signal. Consider the probability distribution of a two-signal counterpart. The signal measured by BSN and standard data value are given by Bayesian theorem equation 1;

$$P(M \mid N) = \frac{P(M \cap N)}{P(N)} \text{ for } P(N) \neq 0 \ldots\ldots\ldots\ldots (1)$$

This probability value is considered as weight update. The factor connects these two factors in a likelihood ratio. Hence, we have considered a three-layer back propagation model from Equation (1). The weight connection of input–output layer for internal activity is obtained by Equation (2).

$$Y_i = \sum_{i=1}^{n} W_{ij} X_j \dots\dots\dots\dots (2)$$

where Wij is the weight associated to the connection node, Xi represents weight, and Yj is the total weight for final output. The activation function for total processed output is given by $Y_{\text{net}} = f(Y_i)$. The maxpooling in the proposed structure is given by Equation (3). This gives the new weight value and actual output.

$$Y(n+1) = Y(n) + \eta[t - y(n)] * X_i(n) \dots\dots\dots\dots (3)$$

To train this DNN for determining abnormality for the acquired signal, a stochastic descent method was used. The algorithm for learning the signal acquired is given in Algorithm 1. In this method a standard value is searched by gradient at each point. This method has the advantage such as to predict a new acquired signal. Figure 2.3 shows the implementation of the proposed algorithm using a Deep learning model. In Figure 2.3, we have shown the number of input layer sensors with the hidden layer function. After acquiring a different measured signal, categories have been constructed by a three-

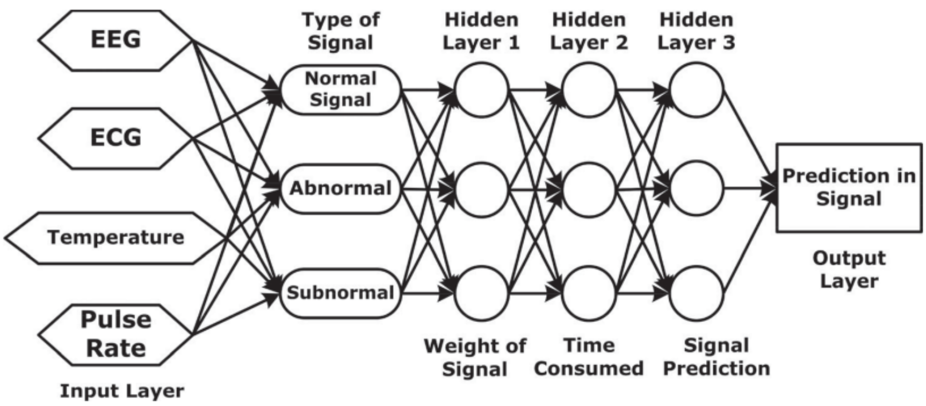


Figure 2.3: DCNN learning model for Signal Log

layer back propagation learning rule. We have adopted criteria as the first layer is linear separable, the second layer is fully connected, and the third layer that is presented by linear activation function was considered for building the deep learning model. Features like time of acquiring, magnitude of signal were measured. This makes the proposed learning model to develop an IoT-based signal monitoring system.

2.3 An Efficient Health Monitoring System with Temprature and Heart rate Sensors using IOT

The authorities of healthcare are planning to provide advanced treatment in clinical organizations. Similarly, the features of Electronic Health applications and E -Health are used by many people to improve their healthcare. The SMS is submitted to the person. A health monitoring system is to be designed which monitors the symptoms like heart rate, oxygen saturation percentage and temperature of the body in IoT network. Internet of things is a technology which enables things around the world to connect with each other using internet. In this technology there will not be human intervention in connecting the things together which is a unique feature of it. The framework of IoT consists of three layers. Control layer, Device layer and Transport layer. Control layer is a method of security that can be used to control and manage the resources in a system. Device layer consists of various devices, types of sensors and many controllers. Transport layer will focus on communication at end-to-end providing features like high reliability, avoiding congestion and assuring that packets will be reached in the order they were sent.

1. **Internet of Things (IoT) :**

   IoT in today's world has become the most buzzed word and is being taught everywhere to everyone as a part of their research or academics. That is the importance that it has gained in almost every field. IOT is indulged in most of the fields like engineering, health sector, manufacturing sector etc. IOT is a network connecting devices together so that they can speak with each other and can decide on their own without any human intervention. For example consider an area that need to be monitored with the help of surveillance cameras. There we deploy the cameras and imagine if they are connected through IOT there need not be much man power required as the cameras will talk to each other and take the decisions like changing their mode i.e. day or night, alerting the person if there is anyone entering into that area, fetching the latest upgrades if there are any without manual operation etc. This not only reduces the man power but also makes the system intelligent.

2. **Health Monitoring :**

   Health monitoring is important to administer any changes in an individual health.

It can also detect any early symptoms of a disease. To keep track of the health we have many wearable devices that can be carried by a person like watch, ring, chain etc. The main parameters that need to be administered are pulse, oxygen levels, temperature, blood pressure etc. With the advent of IOT even remote health monitoring is also made possible where a patient can be administered by a doctor at his home so that the patient is out from the clinical settings. With IOT the parameters that need to be observed every minute are stored in cloud where it will be useful for the doctor to predict health condition of the patient in coming future.

3. **Node MCU :**

   Node MCU is basically a firmware developed for the chip ESP8266 Wi-Fi SOC and hardware is the ESP-12 module. Node MCU is commonly available as development kits where it can be used to build prototypes for IOT applications. Its pin out consists of 3.3V voltage pins, Ground pins to provide power supply and ground to the board. The board also consists of GPIO pins where they can be used to provide input or output to or from the board. Along with these pins RST pin is reserved for reset where it resets the whole program whenever required. Some other pins are Master Out Slave-In(MOSI), Master In Slave-Out(MISO), Serial Clock(SCK) and Chip Select(CS) which indicates that it follows an SPI protocol for establishing a serial communication with the peripheral devices. The development board also contains a 3.3V voltage regulator to take care of the excess voltage that could pass through any module on the board, CP2102 which is a USB to TTL converter; 2.4Ghz antenna for a wide coverage range etc.It has an inbuilt Wi-Fi/ Bluetooth. The pin out of Node MCU can be referred from figure 1. Coding language that is used for Node MCU is Lua which demands the user to again learn the commands which could be a tedious task. Alternatively Node MCU board can also be coded using Arduino IDE which is flexible and easy.

4. **Temperature Sensor :**

   LM 35 series temperature sensors are popularly used for sensing the temperature of the near surroundings. This sensor outputs an analog voltage proportional to the temperature in Celsius. It consists of 3 pins as shown in figure2 namely Vcc, Gnd and Vout where Vcc and Gnd are used to provide the necessary power supply to the device and as the name itself suggest Vout is the pin from where we get the output voltage that is to be provided to any development board like Arduino, Raspberry Pi, Node MCU in this case etc. As this sensor offers advantages like low sensitivity, précised output, elimination of external circuitry etc. it is mostly being used in all the projects where temperature measurement is a concern.

5. **Heart rate Sensor :**
   As heart rate is one of the important features for estimating one's health condition there is a need to measure the same. Here in this research we present a pulse sensor which measures the pulse of a person. In today's wearable devices heart rate sensors are inbuilt so that they measure the heart rate. This sensor could be also used to trace out if any person feels anxious, it tracks the sleeping pattern of an individual which are used to detect the ailment of a person in some situations. Which consists of 3 pins namely Vcc, Gnd to provide power supply to the device and a signal pin which gives out the heart rate of a person that is connected to any one analog pin of arduino A0-A5. The heart rate can be displayed in any display device like LCD or it can be directly stored in the cloud platform like thing speak.

2.4 E-Healthcare Monitoring System using IoT with Machine Learning Approaches

Human lifestyle is based on many factors such as irregular food habits, no nutrition diet, environmental pollutions, lack of proper exercise, indefinite work, restlessness, and increases stress levels which leads to greater disaster in human health. The modern visionary of healthcare industry is to provide better healthcare to patient anytime in thais world in a more economic and patient friendly manner. Therefore for increasing patient care efficacy,there exist a need to improve patient monitoring devices. Internet of Things (IoT) is an emerging technology that is drastically improving with many new enhancements in medical and health domains. Human lifestyle is based on many factors such as irregular food habits, no nutrition diet, environmental pollutions, lack of proper exercise, indefinite work, restlessness, and increases stress levels which leads to greater disaster in human health.The modern visionary of healthcare industry is to provide better healthcare to patient anytime in thais world in a more economic and patient friendly manner. Therefore for increasing patient care efficiancy,there exist a need to improve patient monitoring devices. The Internet of Things (IoT) describes the network of physical objects “things” that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. Internet of Things (IoT) is an emerging technology that is drastically improving with many new enhancements in medical and health domains. The IoT health-monitoring platform has provided us with a significant benefit in the advancement of contemporary medicine. IoT devices are widely used in the medical sector. IoT Health wearable devices are taking new challenges by upgrading with innovative technology and resources. Using health wearable devices, in/out patient’s health status can be monitored periodically and regularly. This project introduces an IoT application framework E-Healthcare Monitoring System (EHMS) integrated with Machine learning (ML) techniques to design an advanced automation system. Using this system it will connect, monitor and decisions making for proper diagnosis. At present authors suggested a framework E-Health care monitoring system (EHMS). It is basically an application model design with various machine learning models. Here IoT wearable sensor collects data from the patient. The collected data from numerous health wearable devices are processed to E-Health care monitoring system. Later EHMS applies machine learning approaches on raw data to analyze the health condition of the patient for better decision making and diagnosis. Disorder analysed report, doctor appointment will acknowledge to the patient, doctor, and caretakers. IoT Health wearable devices are taking new challenges by upgrading with innovative technology and resources. Using health wearable devices, in/out patient’s health status can be monitored periodically and regularly. This project introduces an IoT application framework E-Healthcare Monitoring System (EHMS) integrated with Machine learning (ML) techniques to design an advanced automation system. Using this system it will connect, monitor and decisions making for proper diagnosis.

EHMS is an IoT application which can regulate online patient health monitoring and periodical check-up for various chronic diseases. Essentially patient’s criticality and abnormality condition can be easily identified in the EHMS application. IoT health wearable devices are used for collecting data from a patient in various living areas such as hospitals, homes, and work- stations, etc. Data collected are instantly stored in the local EHMS server use machine learning classification algorithms to build the training models. Collected patients information is used for prediction, review analysis, decision making, and data visualization. It can be shared by both doctor and Inpatient and Outpatient along with a patient caretaker. EHMS can provide qualitative and security services like regular monitoring, valuable collection of data, proper diagnosis analysis, and in-time patient services. In this scenario, the E-Health care monitoring system uses a few IoT health wearable devices carry real-time information from the human body. Here some of the wearable devices such as Temperature sensor, Heartbeat sensor, Blood pressure sensors and Eye lance based Diabetes sensors are acceptable. This system is aimed to design a new application model which gives a better solution and significant improvement for many isolated health services using the Internet of Things (IoT) architecture and Machine Learning techniques. EHMS compatibility with IoT wearable device: Generally, IoT technology functions with physical sensors, wireless networks, routing gateways, and cloud-based security applications. Sensors are connected to the EHMS application for data collection, patient tracking. The proposed system architecture is shown in figure 1.

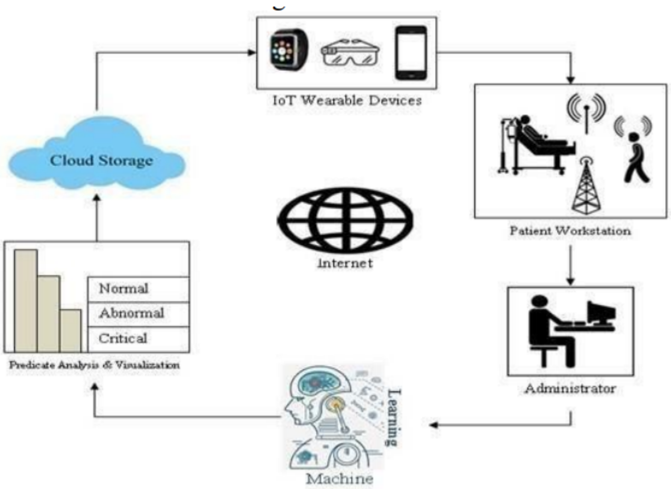


Figure 2.4: Proposed EHMS IoT model architecture

Every individual patient is required to register with a unique wearable device connected to Global Positioning System (GPS) or communication channels. Wearable devices are convenient and it can be easily carried by the patient at all times. Using these devices the patient's health condition can be tracked and along with the geographical location can also be identified. The main feature of EHMS is collecting patient real-time data through an easily wearable sensor device connected to a remote and local administrative server. These wearable sensors devices are connected through wireless sensor networks, routing antennas, and Base stations. Collected information will be directed into the admin server at numerous intervals. Primarily these constrained devices and IoT application are handling additional challenges.

1. **EHMS compatibility with Machine learning and Cloud:**
   After receiving the data from the wearable device the admin server implements a machine learning approach. ML can be applied to existing real-time datasets for prediction of patient condition. EHMS application uses new machine learning algorithms to extract abnormality occurred in the data and data visualization gives a clear understanding to classify the patient's condition. These analysis reports are stored in cloud secured services [8]. Data is distributed through EHMS automated application to all the stakeholders who are involved in the patient real-time monitoring system.

2. **IoT Analytics:**
   IoT analytics can play a role by means of applying machine learning techniques on the datasets collected from IoT devices. Every smart automation system needs analysis, prediction and security to specify its goals in real-time. IoT analytics is required for pragmatics automation systems such as smart home automation, smart agriculture, and smart health [9-10]. Machine learning mechanism provides proper learning and training to EHMS automated machine through which it can perform data classification, data interpretation, data analysis, and data visualization.

3. **Data Modeling and Automation Analysis:**
   IoT analytics data modeling and high-level automation analysis is provided through ML tools. Collected data is processed and stored through ML techniques in the EHMS server. Data is further stored in cloud storage for high availability to all the stakeholders. Finally, EHMS application provides training and prediction analysis based on supervised data for better decision making using some efficient machine learning approaches.

A summarised format of all the four papers are shown below. The table consists of name of the papers being selected, name of the authors, technology used, features and disadvantage of the system.

PAPER	AUTHOR	TECHNOLOGY	FEATURES	DISADVANTAGE
IoT based Health Monitoring & Automated Predictive System to Confront COVID-19	Md Mashrur Sakib ,Maksudur Rahman,Md. Mohsin Kabir and M. F. Mridha	1. Machine Learning SVM algorithm	1. Improved to use as a wearable device. 2. It won't cause any harm to the health.	Less Accurate biological sensors.
Smart-monitor: patient monitoring system for IoT-based healthcare system using deep learning	Pandia Rajan Jeyaraj & Edward Rajan Samuel Nadar	1. IoT-based healthcare system using deep learning	1. Feature extraction and signal abnormalities by analysis using WEKA.	Average Accuracy of 97.5% with a computational time of 65 s was obtained.
An Efficient Health Monitoring System with Temprature and Heart rate Sensors using IOT	K.V.Sowmya, V.Teju	1. Machine Learning SVM algorithm	1. The designed prototype model 2. Low cost 3. High accurate system	Average Accuracy of 97.5% with a computational time of 65 s was obtained
E-Healthcare Monitoring System using IoT with Machine Learning Approaches	Brahmaji Godi, Sangeeta Viswanadham, Om Prakash Samantray	1. EHMS System using Machine learning	1. Store check-up reports 2. Receive an SMS services.	Apply some more machine learning algorithms to improve the efficiency.

Table 2.1: Literature Survey table

Chapter 3

Proposed System

System analysis is the investigation of a system or its components to determine its goals. It is a problem-solving method that improves the system and ensures that all of the system's components work together to reach the goal. This chapter includes the proposed system and various hardware parts, which include an IR sensor, an Arduino Uno, a MAX30100 sensor, a WiFi module, an LCD display, and details of the SVM algorithm and software used in the project.

3.1 Proposed System

Health care monitoring System is an IoT application which can regulate online patient health monitoring and periodical check-up for various chronic diseases. Essentially patient's criticality and abnormality condition can be easily identified in the EHMS application. IoT sensors are used for collecting data from a patient in various living areas such as hospitals, homes, and work- stations, etc. Data collected are instantly stored in the server and then using machine learning classification algorithms to build the training models. Collected patients information is used for prediction, review analysis etc. In this scenario, the E-Health care monitoring system uses a few IoT health sensors that carry real-time information from the human body. Here some of the devices such as Blood pressure sensors and Glucose sensor is used. It compatibility with IoT devices. Generally, IoT technology functions with physical sensors, wireless networks, routing gateways, and cloud-based security applications Sensors are connected to the website for data collection etc.

Using these devices the patient's health condition can be tracked and diabetes can be identified .These devices are connected through wireless sensor networks, routing antennas etc. That Collected information will be directed into the admin server at numerous intervals. Primarily these constrained devices and IoT application are handling additional challenges. After receiving the data from the device the admin server implements a machine learning approach. ML can be applied to existing real-time datasets for prediction of patient condition. It new machine learning algorithms to extract abnormality occurred in the data and data visualization gives a clear understanding to classify the patient's condition. These analysis reports are stored in cloud secured services IoT analytics can play a role by means of applying machine learning techniques on the datasets collected from IoT devices. Every smart automation system needs analysis, prediction and security to specify its goals in real-time. Collected patients information is used for diabetes prediction due to uncontrolled diabetes. This system is very useful for people to take a prior precaution before coming diabetics.

3.2 Hardware

Mostly, we are using four hardware components. IR receiver and transmitter for detecting glucose, a Max30100 sensor will be used for blood pressure, an ESP8266 is a wifi module to send data to the cloud and LCD displays will be used for output or printing results on the display.

3.2.1 Circuit Diagram

First To all the components, connect VCC and GND. An LED that indicates sensing is connected together with the IR emitter and receiver to the A0 pin. SDL and SDD pins of the Arduino are connected by the MAX30100 Sensor Part. ESP8266 To send data, connect the 9 and 10 pins. 2 to 6 digital pins are often connected for communication in LCD screens. In order to link the enabling pin and register selection, use pins 11 and 12.

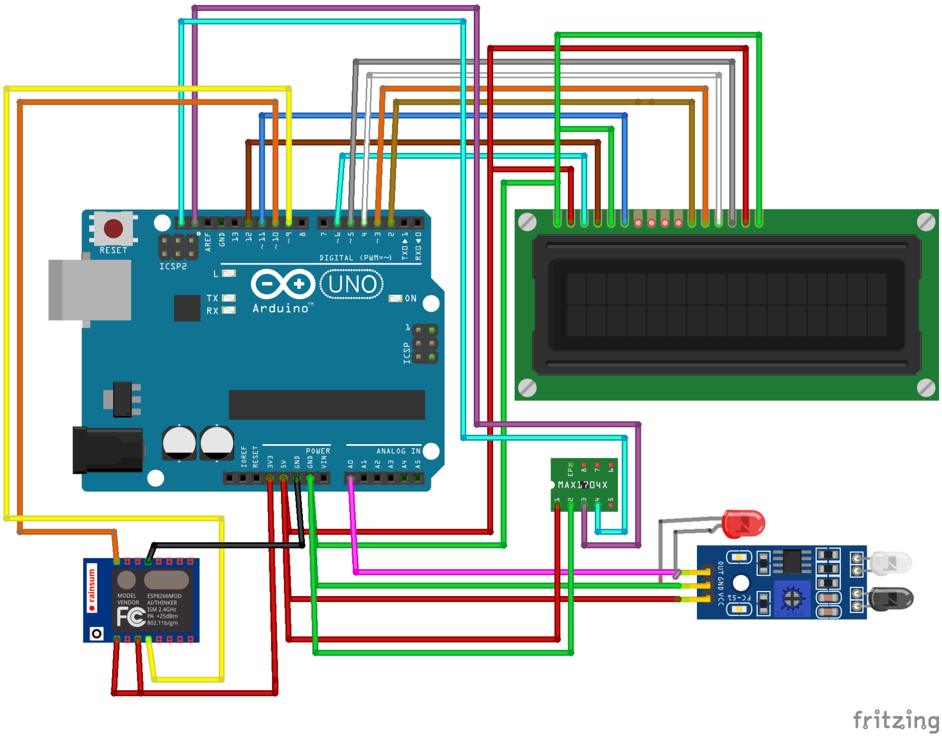


Figure 3.1: Circuit Diagram

3.2.2 IR Sensor

Monitoring of glucose level of blood is important to avoid complications of diabetic and damage to organs. Since invasive method of glucose level measurement is painful and causes damage to nerves, non-invasive method is used as an alternative. The main aim of this work is to develop a simple, reliable, painless, cost effective and portable device for glucose measurement.

When a light ray passes through biological tissues, it is both absorbed and scattered by the tissues. Light scattering occurs in biological tissues due to the mismatch between the refraction index of extracellular fluid and the membranes of the cells. Variation in glucose level in blood affects the intensity of light scattered from the tissue. Beer-Lambert Law plays a major role in absorbance measurement which states that absorbance of light through any solution is in proportion with the concentration of the solution and the length path traveled by the light ray. Light transport theory describes light attenuation as

$$I = I_0 e_{eff}^{-\mu} L \quad ....(1)$$

where, I is the reflected light intensity, Io is the incident light intensity and L is the optical path length inside the tissue. Attenuation of light inside the tissue depends on the coefficient known as effective attenuation coefficient (μ_{eff}), which is given by

$$\mu_{eff} = [3\mu_s\,(\mu_s + \mu_s)]^{1/2} \quad ....(2)$$

The absorption coefficient (μ_a) is defined as the probability of absorption of photons inside the tissue per unit path length, which is given by

$$\mu_a = 2.303 \in C \quad ....(3)$$

$\in$ is the molar extinction coefficient, C is the tissue chromophore concentration and the reduced scattering coefficient (μ_s) is given by equation 4.

$$\mu_s = \mu_s(1 - g) \quad ....(4)$$

where g is anisotropy and (μ_s) is scattering coefficient. Hence from the equations (1) to (4) it can be concluded that (μ_a) depends on the glucose concentration in blood. Thus with the increase in blood glucose concentration, the scattering property of blood decreases.

3.2.3 Block Diagram of Glucose measuring

The proposed work is based on NIR optical technique. NIR light source of 940 nm wavelength is chosen because it is suitable for measuring blood glucose concentration. The sensing unit consists of NIR emitter and NIR receiver (photodetector) positioned on either side of the measurement site (fingertip) as shown in figure 3.1.When the NIR light is propagated through the fingertip in which it interacts with the glucose molecule, a part of NIR light gets absorbed depending on the glucose concentration of blood and remaining part is passed through the finger tip. The amount of NIR light passing through the fingertip depends on the amount of blood glucose concentration.

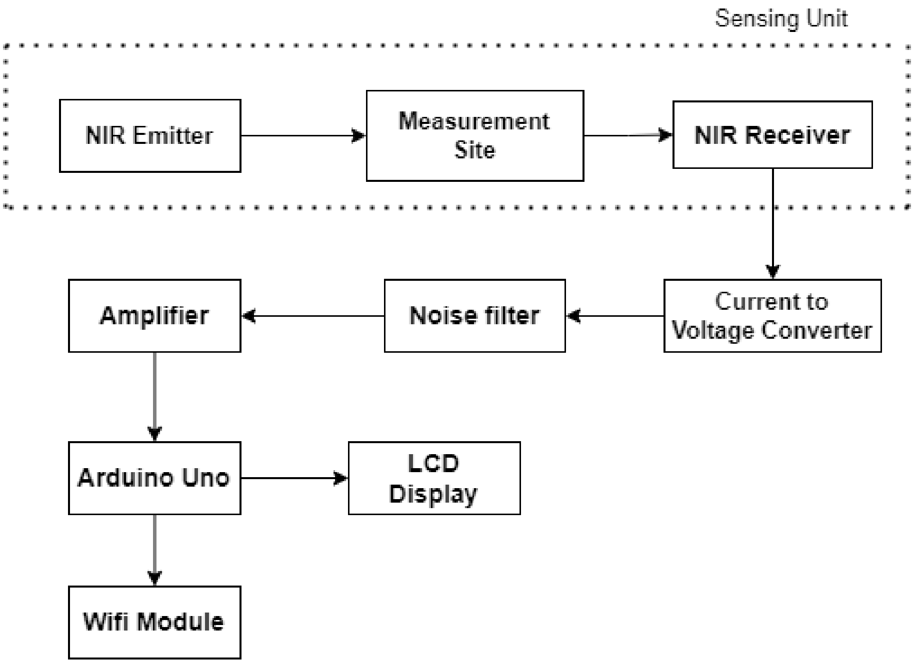


Figure 3.2: Block Diagram of Glucose Measuring

The transmitted signal is detected by the photodetector. The output current of the photo detector is converted into voltage signal and then it is filtered and amplified. The inbuilt ADC block is used for converting the received analog signal to digital form. This digital signal is processed by using second order regression analysis to predict the blood glucose value and the blood glucose value is displayed on the LCD display and also that data will passes through WiFi module for further diabetic prediction.

S.No	Analog Voltage(mv)	Glucose Level(mg/dl)
1	499	142
2	509	146
3	519	156
4	519	157
5	548	177
6	524	159
7	543	209
8	568	133
9	573	179
10	586	224
11	592	175
12	597	187
13	607	196
14	627	191
15	695	167

Table 3.1: Analoge Voltage and Glucose Level

The analog voltage measured at analog pin (Ao) of Arduino uno microcontroller and the corresponding glucose concentration measured. The polynomial equation relating the analog voltage and the glucose level is computed by using regression tool.

$$y = (8 * 10)^{-5}x^2 + 0.1873x + 46.131$$

where x and y are analog voltage (mV) and glucose level (mg/dl) respectively.
Arduino program is written to determine the glucose level for the given analog voltage. The continuous analog voltage values received from the photo detector while placing the finger in-between NIR emitter and photo detector are stored in an array and they are averaged. The microcontroller calculates the glucose value corresponding to this average analog voltage and displays it on both LCD display and send cloud through WiFi module.

3.2.4 Arduino Uno

The Arduino UNO is the best board to get started with electronics and coding. If this is your first experience tinkering with the platform, the UNO is the most robust board you can start playing with. The UNO is the most used and documented board of the whole Arduino family.

Arduino UNO is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

An Arduino UNO is the use of the microcontroller. The UN is an ATMEGA 328P-based microcontroller module.32 kB of flash memory are available in ATMEGA 328P to store code. The board consists of 14 optical pins for input and outputs, 6 analogue inputs, 16 MHz quartz crystal, USB and ICSP. Use the Arduino app to program the UNO Figure 3.2.

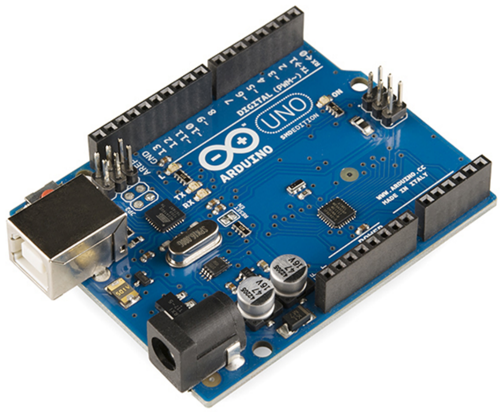


Figure 3.3: Arduino Uno

3.2.5 MAX30100 Sensor

The module features the MAX30100 – a modern, integrated pulse oximeter and heart rate sensor IC, from Analog Devices. It combines two LEDs, a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry (SpO2) and heart rate (HR) signals. On the right, the MAX30100 has two LEDs – a RED and an IR LED. And on the left is a very sensitive photodetector. The idea is that you shine a single LED at a time, detecting the amount of light shining back at the detector, and, based on the signature, you can measure blood oxygen level and heart rate. The MAX30100 chip requires two different supply voltages: 1.8V for the IC and 3.3V for the RED and IR LEDs. So the module comes with 3.3V and 1.8V regulators. This allows you to connect the module to any microcontroller with 5V, 3.3V, even 1.8V level I/O.

Figure 3.4: MAX30100 Sensor

3.2.6 Wifi Module

A low cost Wi-Fi chip with complete TCP/IP functionality, the arduino- compatible ESP8266 is a stunningly built-in MCU (Micro Controller Unit) that allows you to manipulate I/O digital pins via simple almost pseudo-code-like programming language. This computer is manufactured by Espressif Systems, a China based in Shanghai Figure 3. The Wi-Fi library for ESP8266 has been developed based on ESP8266 SDK, using the naming conventions and overall functionality philosophy of the Arduino WiFi library. Over time, the wealth of Wi-Fi features ported from ESP8266 SDK to esp8266 / Arduino outgrew Arduino WiFi library and it became apparent that we would need to provide separate documentation on what is new and extra.

This documentation will walk you through several classes, methods and properties of the ESP8266WiFi library. If you are new to C++ and Arduino, don't worry. We will start from general concepts and then move to detailed description of members of each particular class including usage examples.

Figure 3.5: Wifi Module

3.2.7 LCD Display

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.

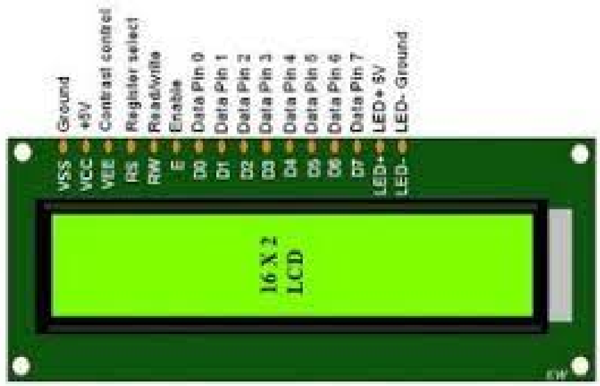


Figure 3.6: LCD Display

3.3 SVM Algorithm

Machine learning Prediction algorithms for Health Monitoring System. Machine learning (ML) is a mathematical and statistical approach which is inter related through data mining techniques with supervised and unsupervised learning methods. The main classification algorithms of ML such as Support vector machine (SVM), Decision tree, KNearest Neighbour (K-NN) and Regression algorithms. Support Vector Machine (SVM) identifies

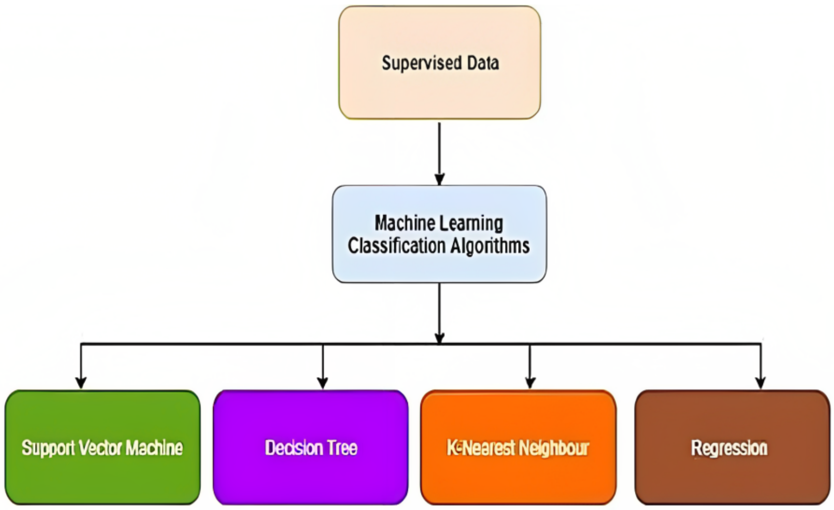


Figure 3.7: Machine learning classification algorithms

better prediction. It is a predictive method and a supervised machine learning model used for a large scale of supervising data. It can perform the best optimization by using both classification and regression techniques. Detailed classification analysis of SVM in data visualization:

1. Best discrimination analysis between the two classes in N-Dimensional space with intersecting planes at each other.

2. Maximize the distance between the data items in all classes. Its main constraint is to identify the margin distance and variation in the data item on the hyperplane.

3. Using the margin hyperplane can identify nearby data items and missing values accurately.

4. It can remove some classification errors and improvise class exactly.

5. It is unable to segregate the two classes using the hyper lines if in the case and data items lie in another class, it can be named as outlier sets. vi. It cannot supervise the outlier sets.

The succeeding features have been provided to help us to forecast whether a person is diabetic or not: Pregnancies: No of times pregnant, Glucose level: Plasma glucose concentration over 2 hrs in an oral glucose tolerance test, Blood Pressure level: Diastolic blood pressure (mm Hg), Skin Thickness: Triceps skinfold thickness (mm), Insulin level: 2Hour serum insulin (mu U/ml), BMI: Body mass index (weight in kg/(height in m)2), Diabetes Pedigree Function: Diabetes pedigree function (a function which scores the likelihood of diabetes based on family history), Age: Age (years). Description of Diabetes Dataset with its ranges are shown in table 3.2.

item Name	**Normal**	**Low**	**Abnormal**	**Critical**
Pregnancies	3<=5	1<=3	6<=9	=>10 above
Glucose	70mg/dL	110mg/dL<130	130mg/dL<=160	=>160 above
Blood Pressure	<80mm Hg	80mm Hg<=100	90mm Hg<=100	=>100mm Hg
BMI	18.50<=25	25<= 30	30<=35	=>35
Age	21-28	28-34	35-45	46-50

Table 3.2: Description of Diabetics dataset

3.3.1 Softwares Used

1. NumPy:-

   It is a library for programming languages that provides support for large, multi-dimensional arrays and matrices, as well as a substantial amount of high level mathematical operations that may be carried out on these arrays. It offers a sizable library of sophisticated mathematical functions that can be used to these arrays and matrices as well as robust data structures that guarantee effective use of arrays and matrices in computations.

2. Pandas:-

   It is a Python-based software library with specialised data structures and operations for dealing with time series and mathematical tables that is intended for data manipulation and analysis. Merging is one of the data manipulation methods supported by Pandas.

3. Scikit-learn:-

   For the Python programming language, it is a free machine learning library that is used to create statistical and machine learning models. Scikit-learn gives us the ability to create various machine learning models for clustering, classification, and regression as well as statistical tools for evaluating these models.

4. Flask:- A popular mini web platform for Python API development is Flask. It is a straightforward yet effective web framework with the capacity to scale up to complex projects, making it ideal for getting started quickly. Python is used to create web apps with Flask. A quick debugger is offered, and there is an integrated development server.

5. Pickle:- Pickle allows for flexibility when deserializing objects. You can easily save different variables into a Pickle file and load them back in a different Python session, recovering your data exactly the way it was without having to edit your code.

6. MySQL:- An interface for connecting from Python to a MySQL database server is called MySQL Python/Connector. It is constructed on top of MySQL and implements the Python Database API. The MySQL server will offer all service needed to manage your database. Once the server is operational, you may use MySQL Connector/Python to link your Python programme to it.

Chapter 4

System Analysis

System analysis is the process of examining and evaluating a system or a specific aspect of a system, such as its structure, components, functions, operations, and interactions. The goal of system analysis is to identify the strengths and weaknesses of a system, and to develop recommendations for improvements or enhancements. This can involve gathering and analyzing data, defining requirements, identifying constraints and opportunities, and developing strategies to optimize system performance.

4.1 Feasibility Study

To define the practicality of the proposed system, a detailed study is conducted evaluating the project's feasibility from all the angles such as its technical, operational. Feasibility study determine the risk and return of pursuing a plan of action, several steps and best practices should be considered before moving forward. The study is also designed to identify potential issues and problems that could arise from pursuing the project. There are several benefits to feasibility studies, including helping project managers discern the pros and cons of undertaking a project before investing a significant amount of time and capital into it.

4.1.1 Technical Feasibility

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development.The web part of the project can be easily implemented using VGG19 Architecture in a user friendly Integrated Development Environment (IDE) that is pycharm which is a dedicated Python IDE providing a wide range of essential tools for Python developers, tightly integrated to create a convenient environment for productive Python, web, and data science development. Flask is a micro web framework written in Python and does not require particular tools or libraries. The development of the project does not meet any type of the technical difficulties

4.1.2 Operational Feasibility

In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment. The proposed system is operationally feasible because it is simple to be used by the user. Processed according to the programmer's implementation. The changes that had been undertaken to reach the current state includes, building of about us page, adding extra features. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

4.1.3 Economic Feasibility

It is economically feasible to implement the proposed project once the system is put into use. This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

4.2 Requirement Analysis

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications. Requirements analysis is an important aspect of project management.

Requirements analysis involves frequent communication with system users to determine specific feature expectations, resolution of conflict or ambiguity in requirements as demanded by the various users or groups of users, avoidance of feature creep and documentation of all aspects of the project development process from start to finish.

4.2.1 Hardware Requirements:

The E-Health care monitoring system uses a few IoT health wearable devices carry real-time information from the human body. Here some of the wearable devices such as Glucose and Blood pressure sensors are used to predict the output.

1. Arduino Uno
2. Power Adapter
3. IR Sensor
4. Blood Pressure Sensor
5. Wifi Module (ESP8266)
6. LCD Display
7. Wires and connectors

4.2.2 Software Requirements:

System requirements are the required specifications a device must have in order to use certain hardware or software. All computer software needs certain hardware components or other software resources to be present on a computer.

1. Windows 10 build version 19044.1865
2. Python version 3.9.13
3. Chrome version 108.0.5359.124
4. Visiual Studio 2022 version 17.4. 3
5. Arduino IDE version 1.8.13
6. Arduino cloud storage

4.2.3 Life Cycles Used

A software life cycle model (also termed process model) is a pictorial and diagrammatic representation of the software life cycle. A life cycle model represents all the methods required to make a software product transit through its life cycle stages. It also captures the structure in which these methods are to be undertaken.

In this project, we are using the waterfall model and the iterative model. The waterfall model is the simplest model of the software development paradigm. It says all the phases of SDLC will function one after another in a linear manner. That is, when the first phase is finished, only the second phase will start, and so on. The iterative model leads the software development process in iterations. It projects the process of development in a cyclic manner, repeating every step after every cycle of the SDLC process, and it is also helpful in hardware part coding.

Chapter 5

System Design and Schedule

The proposed system is depicted in a variety of diagrams to provide a clear view of the entire system. Architecture, Usecase diagram, Data flow diagram, Er diagram and Gantt chart is illustrated and explained in detail below.

5.1 Architecture Diagram

The physical implementation of a software system's components is mapped out visually in an architectural diagram. It consists of a group of ideas that make up an architecture, including all of its guiding principles, basic building blocks, and constituent parts.

The architecture in our system consists of 9 components that is Glucose sensor,Blood pressur sensor,Power supply,Arduino Uno,Wifi Module,SVM Algorithm,Age,BMI,Dataset and Diabetic or Not, with 2 main blocks: Hardware and Software.

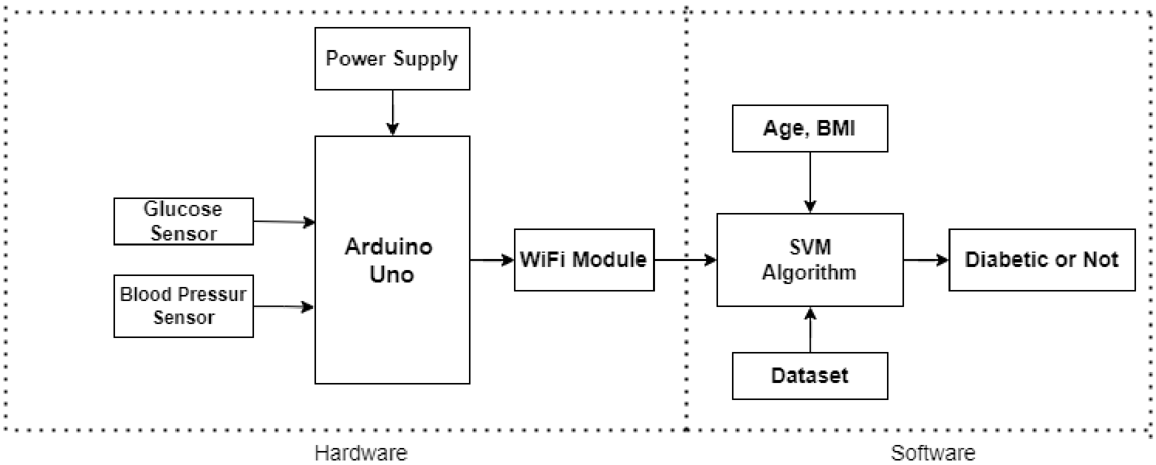


Figure 5.1: Architecture

5.2 Use Case Diagram

A use case diagram is used to graphically describe the user's probable interactions with the system in the system design. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures. It is made up of actors who represent interactions with the system through services or use cases. There are three players in the use case diagram: the patient,sytem and the system. The patient has collect data through sensors ,that data will processed and sending to cloud storage. The system processes the login, receiced send data, extract data, comparing data set,predict data set, and finally Display prediction or result.

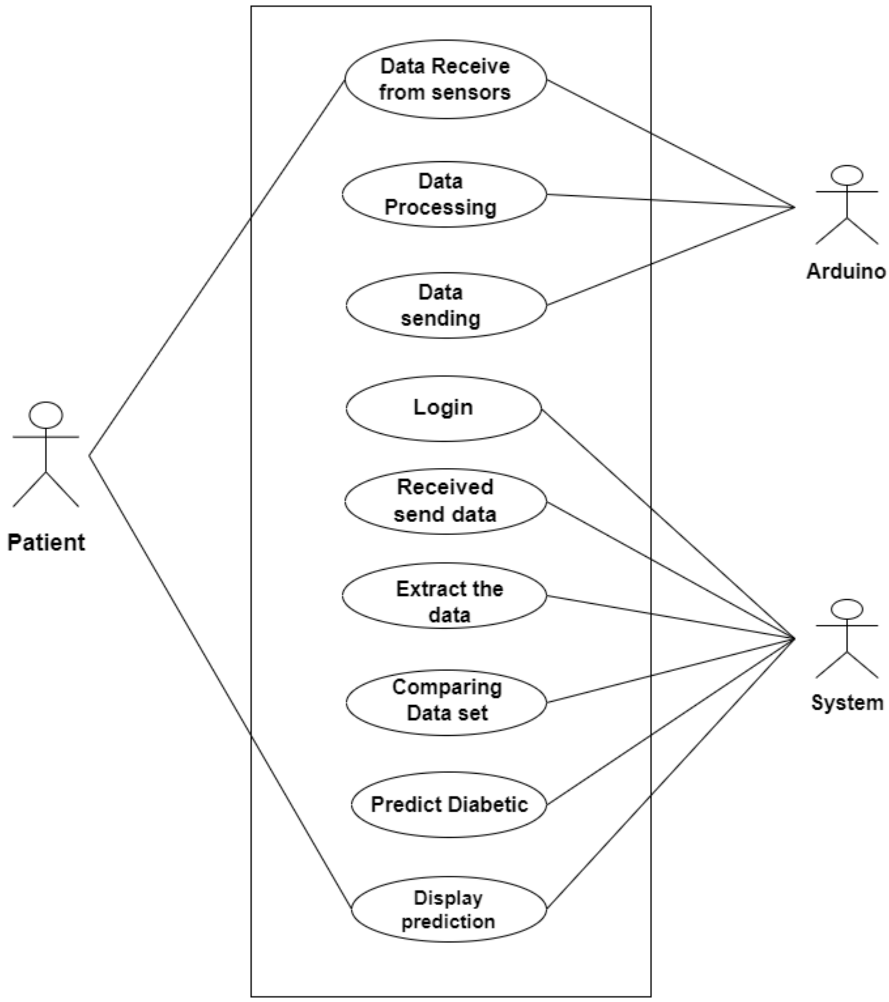


Figure 5.2: Use case Diagram

5.3 Data Flow Diagram

A data-flow diagram is a visual representation of how data moves through a process or system. It offers details on the inputs and outputs of each entity as well as the process itself. There are four common methods of notation used in DFDs. They provide greater detail than a context diagram as they display each process involved within the information system as an individual circle, meaning the end result will contain multiple circles. A DFD also has a shape for data stored to represent where data is sent and retrieve from,such as specific database. Data stores are represented as a three sided rectangle shape. The level in Data flow diagram are used to represent progressive degrees of detail about the system or process. Mainly there are 3 levels in data flow diagram, which are 0-level DFD,1-level DFD, 2-level DFD.

5.3.1 Level 0

It is also known as context diagram. It represents the entire system as a single bubble with input and output data indicated by incoming outgoing arrows. The zeroth level describes an abstract picture of the system as a single process with external entities. There are two external entities :-

- Patient
- Diabetic or Not

The functioning of Diabetic prediction system is represented in a circle. The patient is linked to a Diabetic prediction system in order to predict diabetic and obtain diabetic prediction information from the system.

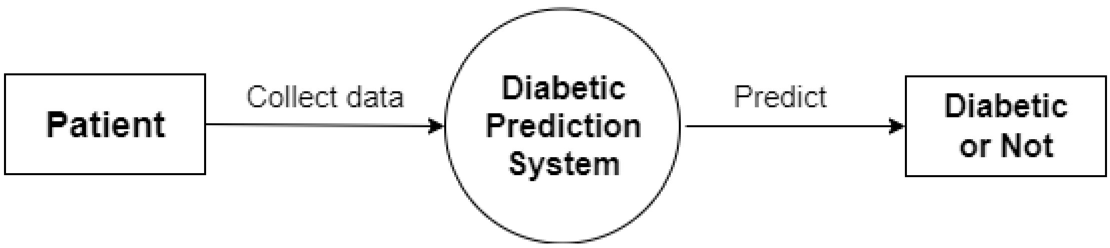


Figure 5.3: Data Flow:Level 0

5.3.2 Level 1

In level 1 DFD, the context diagram is decomposed into multiple processes. The first level focuses on several processes associated with each entity and emphasises the primary functions. Manly diabetic prediction system are divided into 4 sub process login,wifi module,prediction and report. collect patient data using sensors and input manually the patient will go login using id admin will provide a password login is successful manually collected 3 data BMI,no.of pregnancies, and age it passes through the prediction system sensor collected data already stored database with the help of wifi modules, finally create report patient is diabetic or not

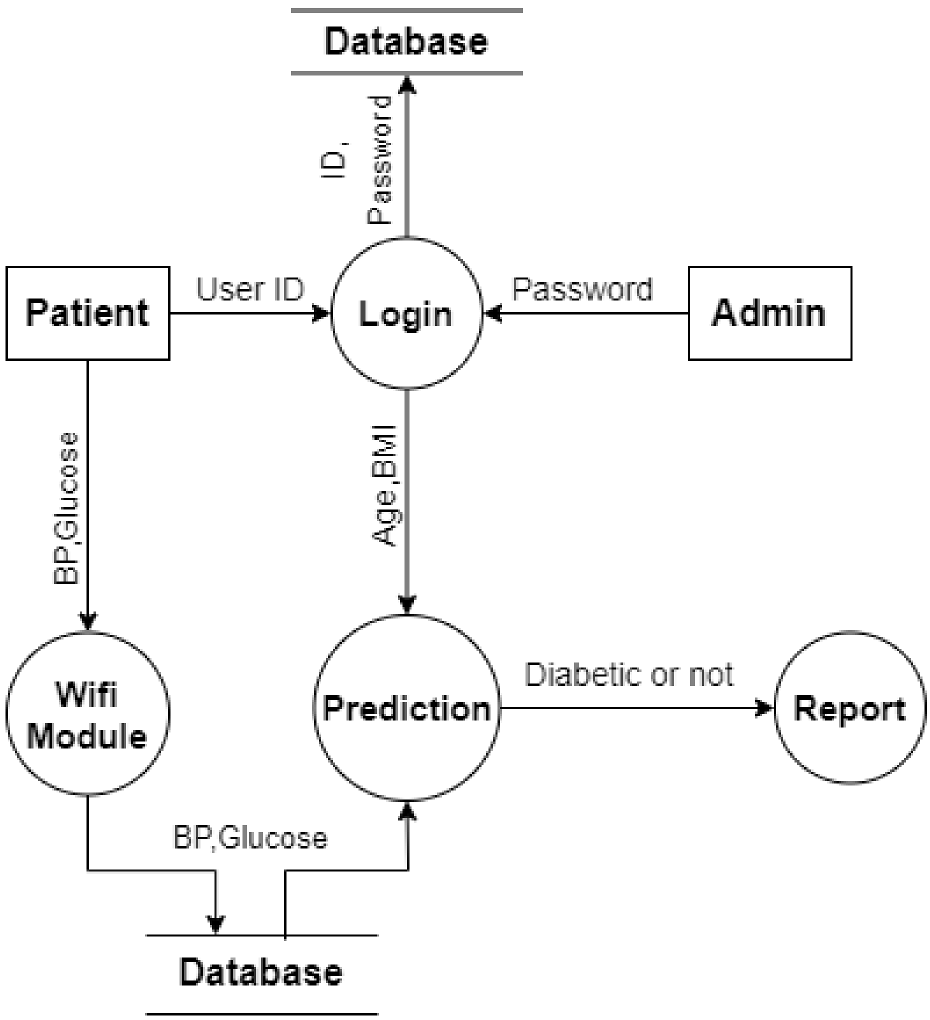


Figure 5.4: Data Flow:Level 1

5.3.3 Level 2

It is used to plan or record the specific detail about the system's functioning. Level 2 is a more in-depth version of our system. The work is similar to that outlined before in level 1 for the entities. Another type of entity is assign IP address,SVM algorithm,comparison and matching. The data received is validated from the database, and the data as input by the patient. The final result will also show patient.

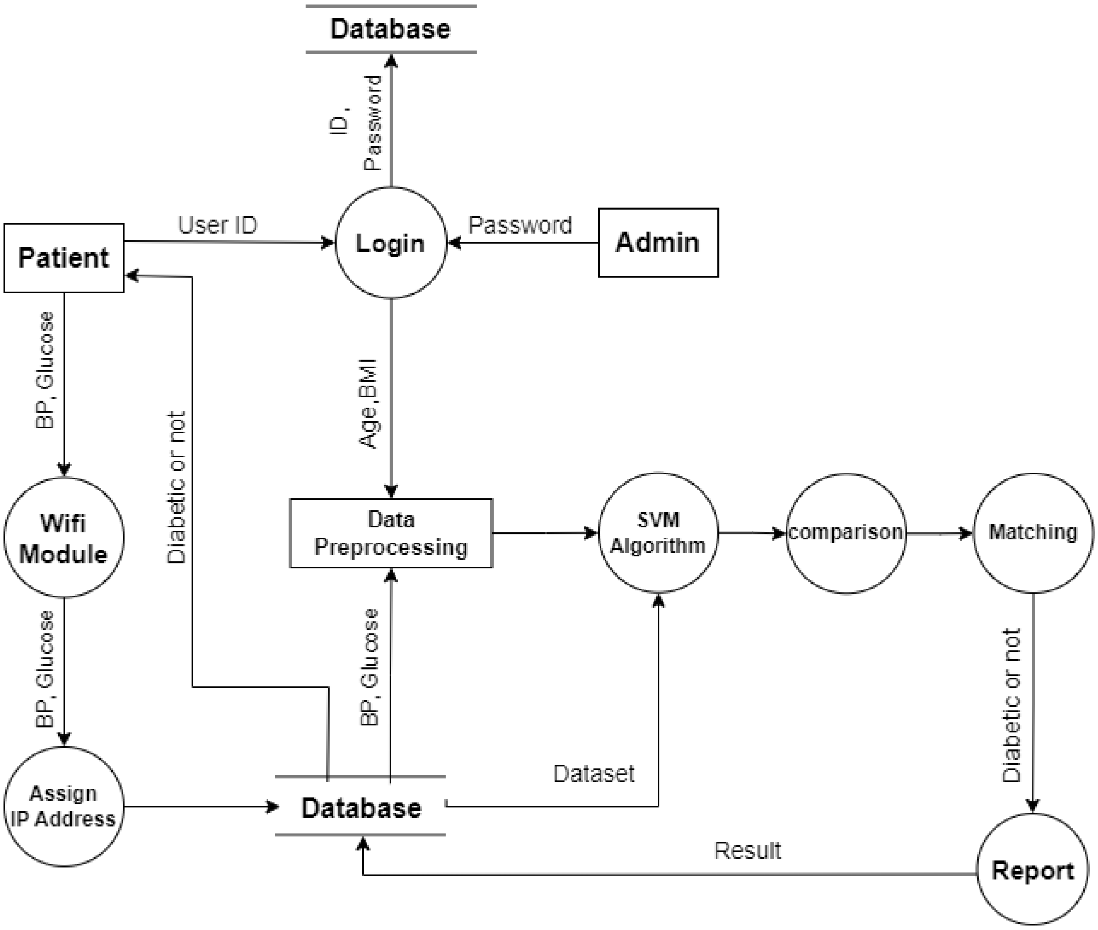


Figure 5.5: Data Flow:Level 2

5.4 ER diagram

An entity-relationship diagram, or ER diagram, is essential for modeling the data stored in a database. It is the basic design upon which a database is built. ER diagrams specify what data we will store: the entities and their attributes. They also show how entities relate to other entities. ER diagrams or ERD's are composed of three main elements: entities, attributes, and relationships. Entities - typically displayed in a rectangle, entities can be represented by objects, persons, concepts, or events that contain data.

The Er diagram contains Eignt entity: patient, admin, logi, Wifi module, input data, cloud report and dataset. BP and Glucose are two attributes of the patient. Input data attributes are no.of pregnanies, BMI ,Age.

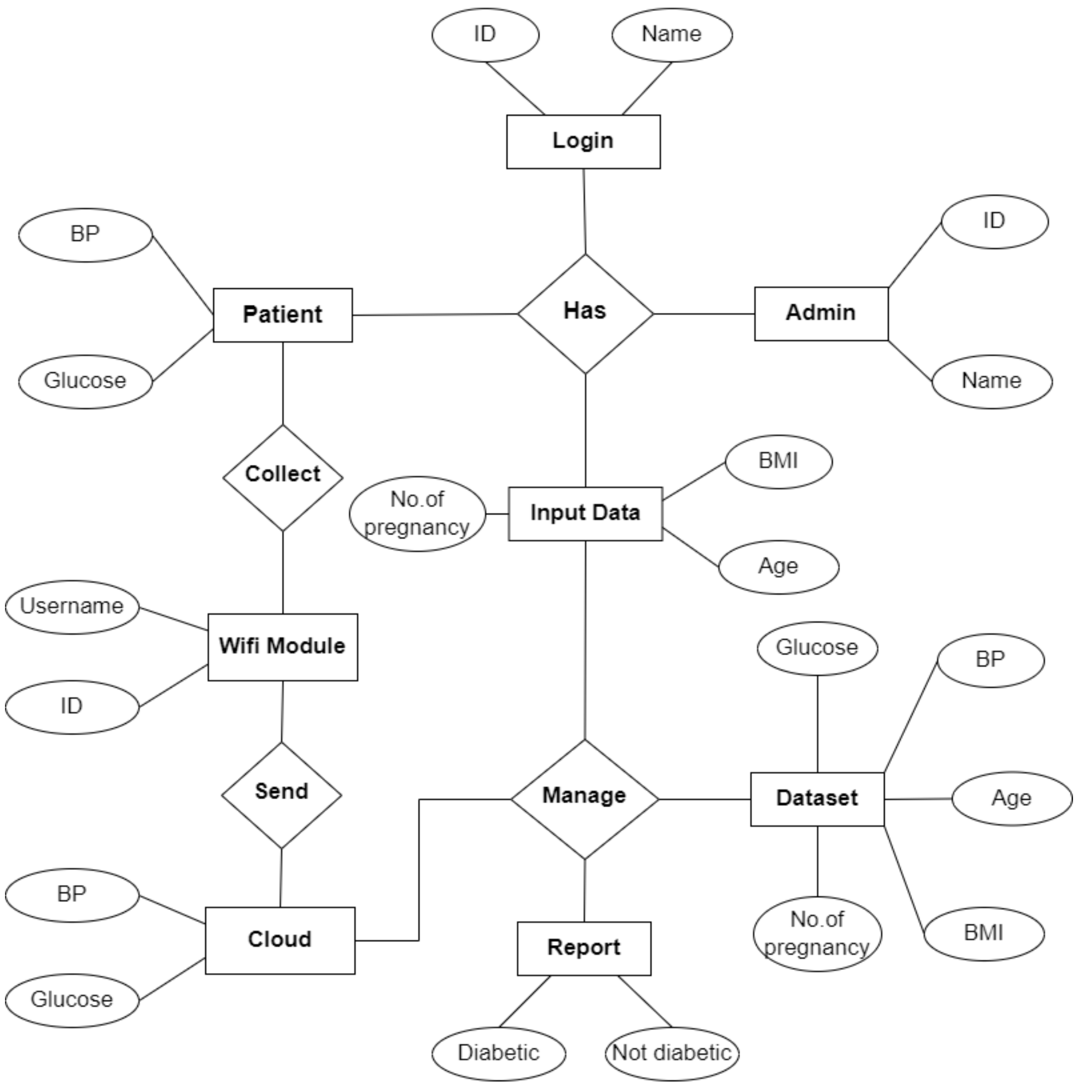


Figure 5.6: Er Diagram

5.5 Gantt Chart

A Gantt chart is a visual depiction of a project's schedule plotted against time, which includes the time necessary to accomplish different activities such as tasks or events. Topic selection, Literature survey, proposed system, Architecture design, and module division. We had completed topic selection in seven days. The literature survey of this module will take seven days. Proposed system will completed in 14 days ,architecture design will completed in 12 days and module division will completely last 10 days .

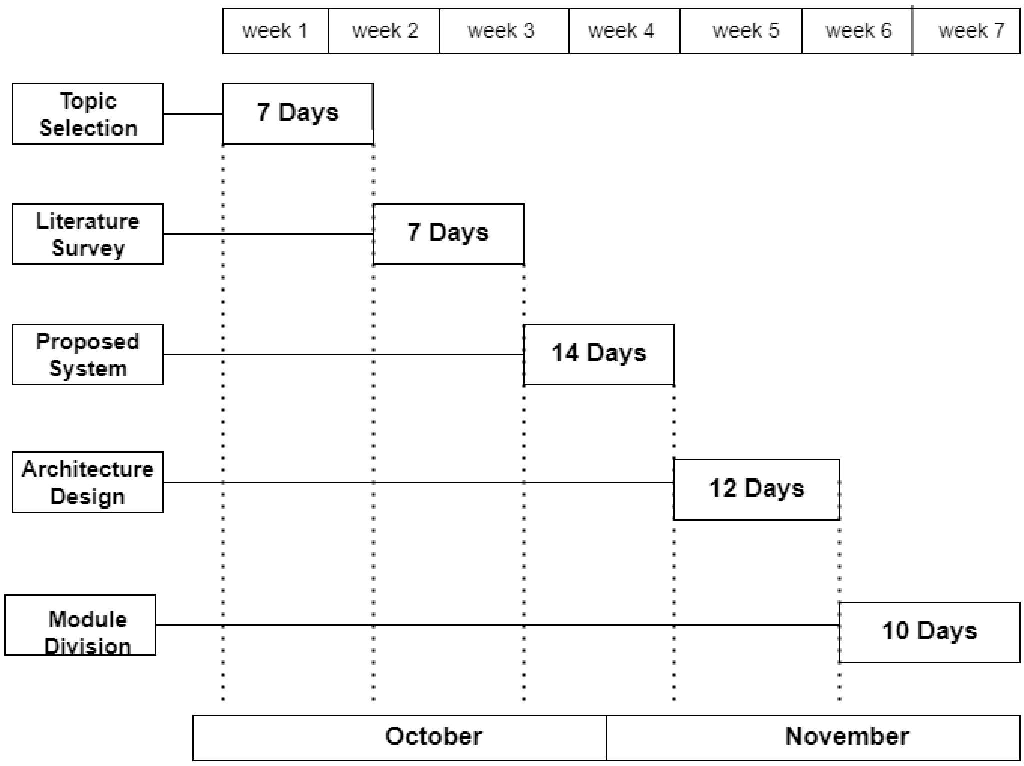


Figure 5.7: Gantt Chart

Chapter 6

System Implementation

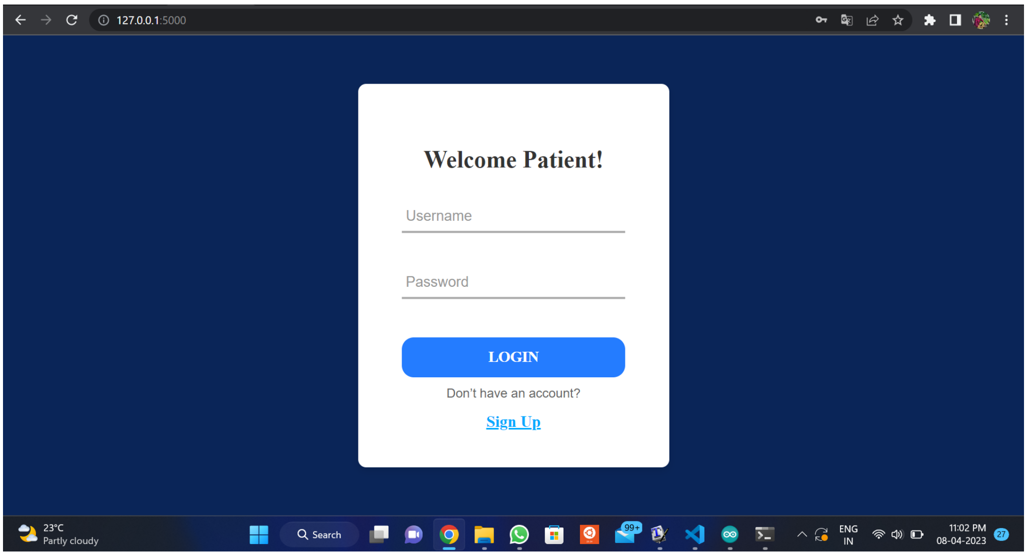


Figure 6.1: Login Page

A login page with a welcome patient notice for website or application where patients need to access their personal health information. The login page displays a welcome message and prompts the user to enter their login credentials consist of a username and password.

If a user does not have an account, they can sign up for one by clicking on the ”sign up” button. This typically takes the user to a registration page where they are asked to provide personal information, such as their full name, desired username and password. Once the registration process is complete, the user can then use their new login credentials to access their personal health information.

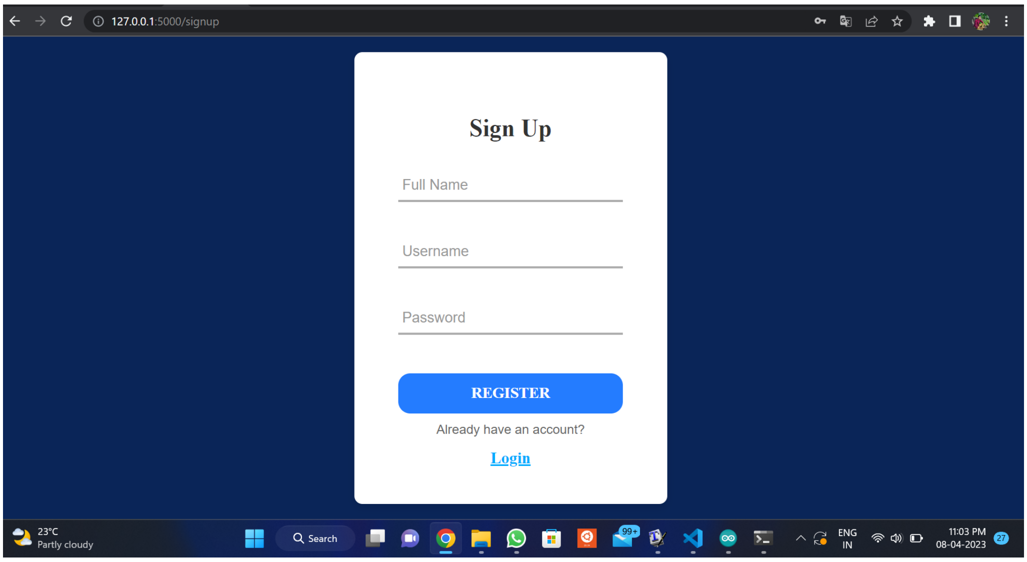


Figure 6.2: SignUp Page

A sign-up page is typically used on a website or application to allow users to create an account and gain access to specific features or content. The sign-up page typically includes fields for the user's full name, username, and password, as well as a "Register" button. The full name field typically asks for the user's first and last name.The username field usually requires the user to choose a unique username that will be used to identify them on the platform. The password field typically requires the user to create a strong, unique password to protect their account. Best practices for password security include requiring a minimum length, requiring the use of both letters and numbers, and prohibiting the use of common words or phrases. If the user already has an account, they can typically click on a "login" button to access their account. This takes them to a separate login page where they can enter their username and password to access their account.

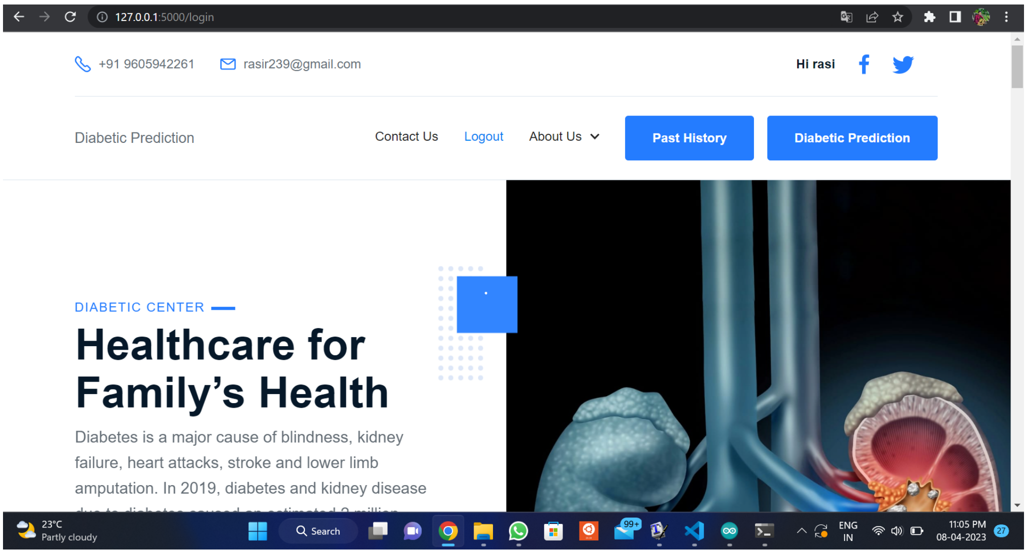


Figure 6.3: Home Page

A home page for diabetes prediction Website includes various webpages for easy user access. The home page mainly contains all the details and information about diabetes making the it important for the user to have basic knowledge about diabetes before the prediction. It also provides the user with information like which specialist to consider in case of diabetes. It contains a past history page that provides the user with a record of the user's previous inputs and predictions from the diabetic predictor page. This includes data such as the date and time of each prediction, as well as the user's input values for glucose level, blood pressure, height, weight, age, and number of pregnancies.

A diabetes prediction Page that allows the user to predict if they have diabetes or not with suitable input. This page includes a "contact us"Button that allows the user to contact us in case of any error or problem in the functioning of the website. This page includes an "about us" Button that directly takes the user to a page where the user is provided with all the details of the website founder and website implementer. A "logout" button is also included in the home page to provide the user to exit from the website after their use .

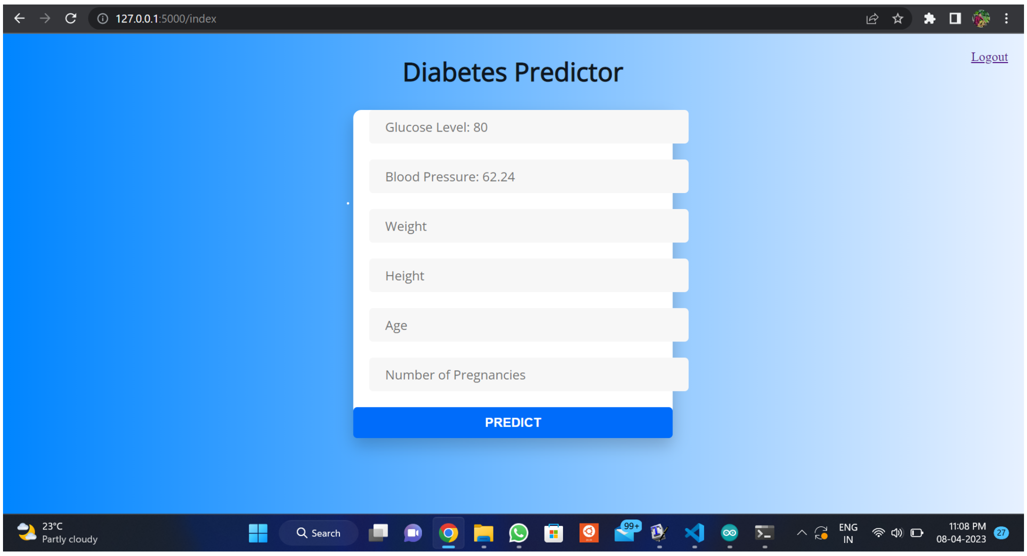
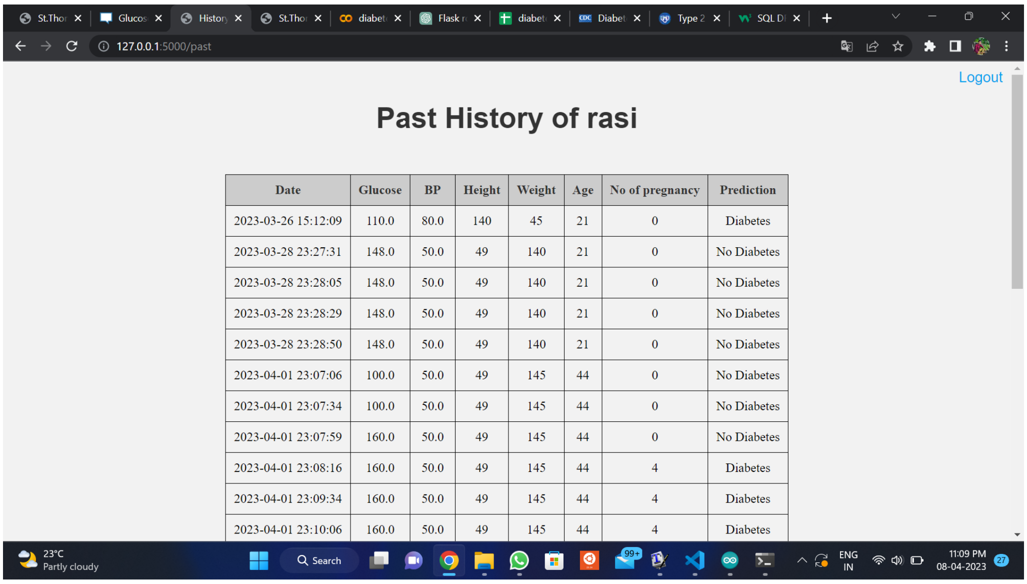


Figure 6.4: prediction Page

A prediction page for a diabetic predictor typically allows users to input various personal and health data, such as glucose level, blood pressure, weight, height, age, and number of pregnancies, to predict the likelihood of developing diabetes. Users may be prompted to input their data into specific fields, and may be required to complete all fields before the prediction can be made. Once all data has been inputted, the predictor algorithm will run and display the predicted likelihood of developing diabetes based on the user's input. This information may be displayed as a percentage, with a higher percentage indicating a higher risk of developing diabetes. The page may also include a "logout" button to allow the user to securely log out of their account once they are finished using the predictor. This can help ensure the privacy and security of the user's personal and health information.



Date	Glucose	BP	Height	Weight	Age	No of pregnancy	Prediction
2023-03-26 15:12:09	110.0	80.0	140	45	21	0	Diabetes
2023-03-28 23:27:31	148.0	50.0	49	140	21	0	No Diabetes
2023-03-28 23:28:05	148.0	50.0	49	140	21	0	No Diabetes
2023-03-28 23:28:29	148.0	50.0	49	140	21	0	No Diabetes
2023-03-28 23:28:50	148.0	50.0	49	140	21	0	No Diabetes
2023-04-01 23:07:06	100.0	50.0	49	145	44	0	No Diabetes
2023-04-01 23:07:34	100.0	50.0	49	145	44	0	No Diabetes
2023-04-01 23:07:59	160.0	50.0	49	145	44	0	No Diabetes
2023-04-01 23:08:16	160.0	50.0	49	145	44	4	Diabetes
2023-04-01 23:09:34	160.0	50.0	49	145	44	4	Diabetes
2023-04-01 23:10:06	160.0	50.0	49	145	44	4	Diabetes

Figure 6.5: Past History

A past history page displays a record of the user's previous inputs and predictions from the diabetic predictor page. This can include data such as the date and time of each prediction, as well as the user's input values for glucose level, blood pressure, height, weight, age, and number of pregnancies. The page may display this information in a table with each row representing a separate prediction and input data. The table include columns for each input value, as well as a column for the predicted likelihood of developing diabetes based on the input data. This page is useful for tracking changes in the user's health and predicting future risk of developing diabetes based on past trends. It can also be used to monitor the effectiveness of lifestyle changes or medical treatments in reducing the risk of diabetes.

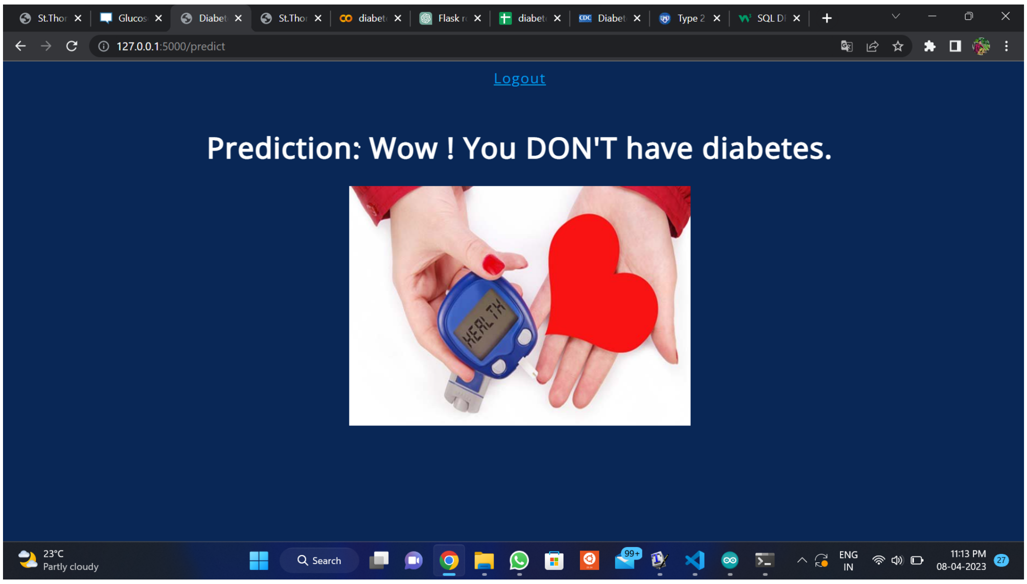


Figure 6.6: Result Page :No Diabetic

A result page for a diabetic predictor that shows no diabetic prediction displays a message such as "Wow! You don't have diabetes" along with an image or icon to congratulate the user on their healthy status.

Additionally, the page include a "logout" button to allow the user to securely log out of their account once they are finished using the predictor. This can help ensure the privacy and security of the user's personal and health information.

The page should also be designed to provide a positive and supportive message to the user, encouraging them to maintain healthy habits and monitor their health in the future.

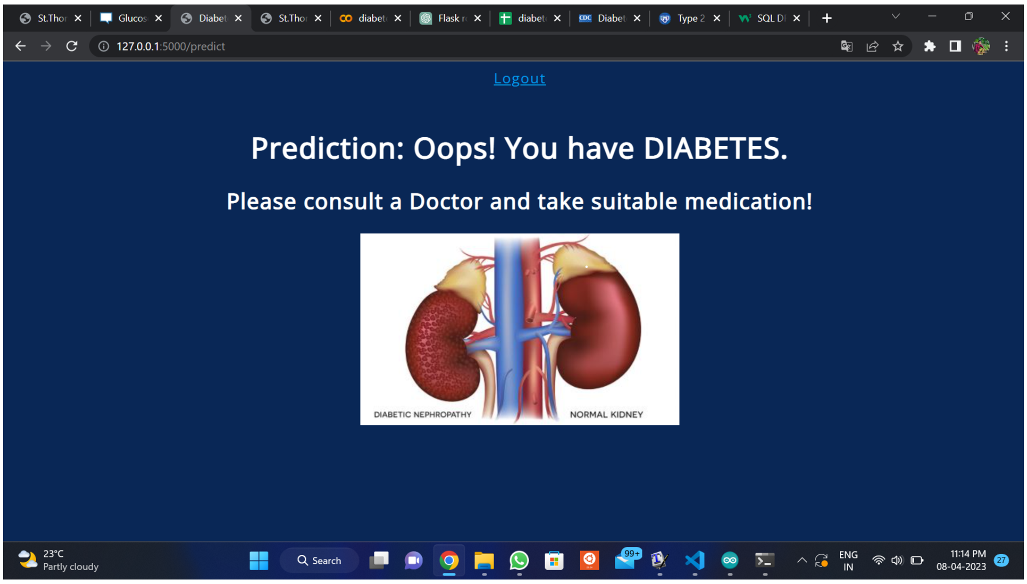


Figure 6.7: Result Page :Diabetic

A result page for a diabetic predictor that shows a prediction of diabetes typically displays a message such as ”Oops! You have diabetes” along with an image or icon to encourage the user to seek medical attention and take suitable medication. The page may also provide information on what steps the user should take next, such as consult a doctor and take suitable medications. The page also include a ”logout” button to allow the user to securely log out of their account once they are finished using the predictor. This can help ensure the privacy and security of the user’s personal and health information. The page should also be designed to provide a supportive and encouraging message to the user, emphasizing the importance of seeking medical attention and taking steps to manage their condition.

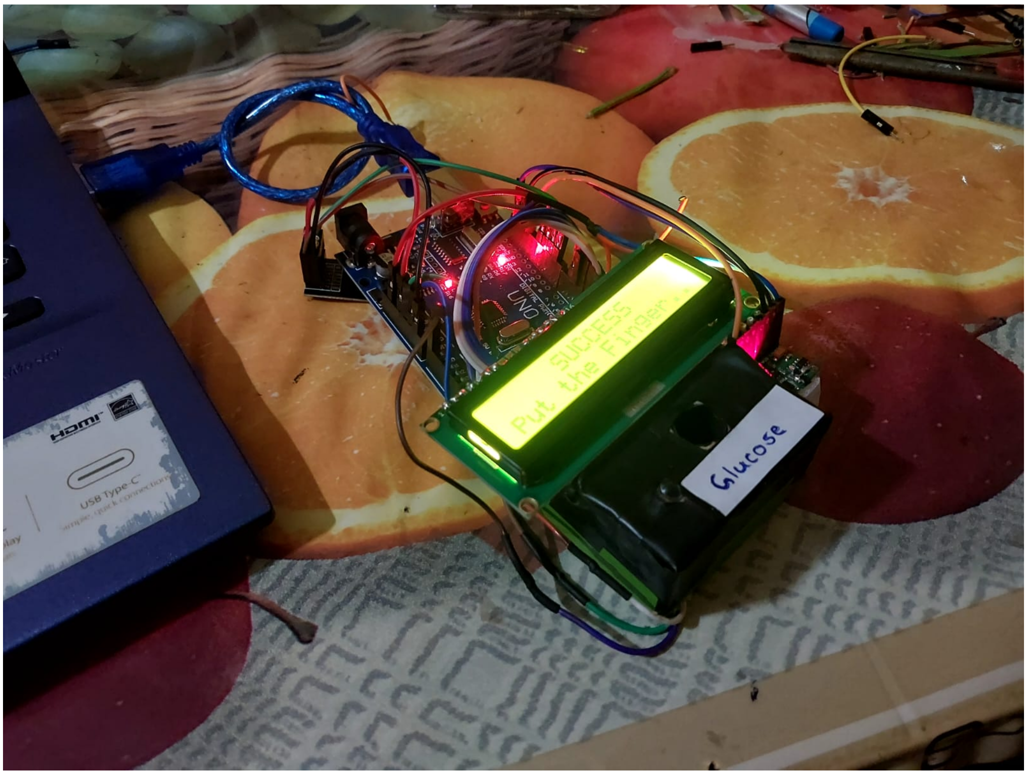


Figure 6.8: Hardware

In the hardware part we have a glucose sensor,blood pressure sensor ,arduino UNO, Wifi module. To measure the glucose level we have 2 inbuilt IR sensors and to measure blood pressure we have max30100 sensor. Using an Arduino UNO we process the data. Wifi module send the data that we obtain from the sensors to the website for further prediction, Also to display the result in LCD display.

Chapter 7

Testing

Software Testing is a process of executing the application with an intent to find any software bugs. It is used to check whether the application met its expectations and all the functionalities of the application is working. The final goal of testing is to check whether the application is behaving in the way it is supposed to under specified conditions. All aspects of the code are examined to check the quality of application. The primary purpose of testing is to detect software failures so that defects may be uncovered and corrected. The test cases are designed in such way that scope of finding the bugs is maximum.

7.1 Testing Levels

There are various testing levels based on the specificity of test.

- Unit testing: Unit testing refers to tests conducted on a section of code in order to verify the functionality of that piece of code. This is done at the function level.
- Integration Testing: Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Its primary purpose is to expose the defects associated with the interfacing of modules.
- System Testing: System testing tests a completely integrated system to verify that the system meets its requirements.
- Acceptance testing: Acceptance testing tests the readiness of application, satisfying all requirements.

- Performance testing: Performance testing is the process of determining the speed or effectiveness of a computer, network, software program or device such as response time or millions of instructions per second etc.

Performance of the system can be determined based on the system/application responsiveness under all kinds of load. Performance testing in IoT framework is little different than traditional performance testing. IoT devices generates a lot of data which is saved in server and analyzed for immediate decisions. Hence IoT system must be built for high performance and scalability. And to measure these two key attributes, it is important to understand the business value for which it is build i.e. in our case patient health data. Hence it is necessary to simulate real world models, network conditions etc.

7.2 Performance Testing Challenges in IoT

1. IoT does not have a standard protocol set to establish a connection between IoT application and devices. IoT protocols used range from HTTP, MQTT, AMQP and more. These protocols are still in early phases of development and different IoT vendors come up specific protocol standards. Since these are new protocols, current performance testing tools may or may not support them.

2. IoT devices or sensors spread across different places and use different network to connect to servers to send and receive data. As a part of PT, we can simulate devices from different locations using different networks such as 2G, 3G, 4G, Bluetooth, WIFI etc.

3. Sometimes IoT implementations require the data from device that needs to be processed at runtime and based on data received, corresponding decision to be made. These decisions are generally notifications or alerts. As a part of PT, these notifications are to be monitored i.e. time taken to generate the notification.

The proposed method is validated by measuring the glucose reading of 20 individuals using both invasive and proposed noninvasive method and the readings are tabulated in table 3. The accuracy of the proposed glucose measurement device is measured using Clarke Error Grid Analysis and Surveillance Error grid analysis.

S.No	Glucose Value Obtained by Invasive method	Glucose Value Obtained by Non Invasive method	Difference
1	117	118	+1
2	143	143	0
3	112	115	+3
4	106	103	-3
5	166	169	+3
6	193	192	-1
7	88	88	0
8	108	110	+2
9	110	117	-7
10	134	151	-17
11	245	213	32
12	299	252	47
13	145	139	6
14	211	186	25
15	152	111	41
16	205	219	-14
17	120	129	-9
18	157	142	15
19	117	122	-5
20	164	170	-6

Table 7.1: Comparison of invasive glucose and proposed glucose level

Chapter 8

Result Analysis

This health monitoring system model consists of a dataset that is trained and tested to give the best results. There are over 700 data points in the dataset. The system used various machine learning algorithms for training and classification, but it gave the best results using the SVM algorithm. This model has an accuracy of around 78%. The trained model is later used to predict if the user has diabetes or not. The predictions on the test dataset are computed by calculating the precision and recall in Table II.

Using basic inputs like age, BMI, and the input that is obtained through the health monitoring device, we can predict the diabetes of a patient. We can say that the persona accuracy measure for a given data set when a person is diabetic is when precision is 0.78 and recall is 0.68. When the person is not diabetic, the precision is 0.84 and the recall is 0.92. This model can make more accurate predictions with visual analysis.

	Precision	**Recall**	**Support**	**F1 Score**
Diabetic	0.78	0.68	56	0.72
No Diabetic	0.84	0.92	100	0.88

Table 8.1: Accuracy measure of SVM Algorithm for a given dataset

Chapter 9

Conclusion

In this hectic society, most of the people are affected by numerous chronical and non-chronical diseases. People like academic students, research scholars, Doctors, Pharm chemists, and scientists also found a better solution to medical problems for many decades. By using the wearable device the patient information can be easily tracked and verified by the concerned doctor and ground the result analysis for Treatment services. If in case the patient abnormal condition is found in EHMS analysis, the doctor has the possibility to take appropriate decisions based on the abnormal results. Treatment can be started immediately and relevant medication setups can be sent in an ambulance within less time to the patient. Health Management system challenges with various health-related services such as daily health alerts, doctor appointment, suggests food diet, E-Health check-up reports and many more through SMS services. Consequently, hospitals can customize and provide the best suggestable E Health services. Health Management applications will also support future enhancements. For an proposed method, we have collected a diabetic data set from online sources. The dataset collected is assumed to be same as the dataset collected through IoT-based wearable devices. The dataset was passed through a machine learning algorithm called SVM which has achieved an accurate report.

References

[1] K.V.Sowmya,V.Teju,*“An Efficient Health Monitoring System with Temprature and Heart rate Sensors using IOT”*Publisher: European Journal of Molecular & Clinical Medicine, ISSN 2515-8260 Volume 08, Issue 2 , 2021

[2] Md Mashrur Sakib Choyon,Maksudur Rahman,Md. Mohsin Kabir and M. F. Mridha *“IoT based Health Monitoring & Automated Predictive System to Confront COVID-19”,* 2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)

[3] Brahmaji Godi, Sangeeta Viswanadham, Om Prakash Samantray,*”E-Healthcare Monitoring System using IoT with Machine Learning Approaches”*,University College London from IEEE Xplore (2020)

[4] P.Daarani & A.Kavithamani (2019) *“Blood Glucose level Monitoring by noninvasive method using nera Infra Red Sensor”* International Journal of Latest Trends in Engineering and Technology pp. 141-147

[5] Jha, S. K. (2019). *”Collaborative handshaking approaches between internet of computing and internet of things towards a smart world.”* Telecommunication Systems, 70(4), 617-634.

[6] Khari, M. G. (2019). *”Securing data in Internet of Things (IoT) using cryptography and steganography techniques”*. IEEE Transactions on Systems, 50(1), 73-80.

[7] hatterjee, J. M. (2018). *”Internet of Things based system for Smart Kitchen”*. International Journal of Engineering and Manufacturing

[8] Wan, J. A.-a. (2018:298). *”Wearable IoT enabled real-time health monitoring system”*. J Wireless Com Network, (pp. 1-11).

[9] Deepti Sisodia, D. S. (2018). *"Prediction of Diabetes using Classification Algorithms"*. Procedia Computer Science, 132, pp. 1578-1585.

[10] H. N. Saha, D. P. (2017). *"Internet of Thing based healthcare monitoring system."* IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), (pp. 531-535). Vancouver, BC.

[11] Gaurav Raj, N. R. (2017, July). *"IoT Based EMG Monitoring System."* International Research Journal of Engineering and Technology (IRJET), 4(7).

[12] Khari, M. K. (2016). *"Internet of Things: Proposed security aspects for digitizing the world"*. 3rd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 2165-2170). IEEE.

[13] B.K, B. D. (2015, July). *"Secured Smart Health Care Monitoring System Based on IoT"*. International Journal on Recent and Innovation Trends (IJRIT), 3(7), 4958-4961.

[14] Viraj Shinde, Tushar Bacchav, Jitendra Pawar and Mangesh Sanap *"Smart-monitor: patient monitoring system for IoT-based healthcare system using deep learning"*,International Journal of Engineering Research and Technology (IJERT), Vol. 3, Issue 1, January – 2014.

[15] D. K. Rathore, A. U. (2013). *"Wireless patient health monitoring system"*. International Conference on Signal Processing and Communication (ICSC), (pp. 415-418). Noida.

[16] Claudia C. Gutiérrez, R. M. (May 17, 2011). *"e-Health monitoring applications: What about Data Quality"*. HamIS 2011 Health Ambient Information Systems Workshop 2011. Gramado, Brazil.

Appendix A

Sample Code

```
import numpy as np
import pandas as pd
from flask import Flask, request, jsonify, render_template,redirect,
    url_for, session
import pickle

import requests
import time
import json

from flask_mysqldb import MySQL
import MySQLdb.cursors
import re

app = Flask(__name__)

#databse connection datas
app.secret_key = "rafasafa"

app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = 'rafasafa'
app.config['MYSQL_DB'] = 'patient'

print("Sucessfully connected")
```

```
mysql = MySQL(app)
'''
url="https://thingspeak.com/channels/2091780/feeds/last.json?api_key=
    F2BEH7B2BK6ST3W9"
response=requests.get(url)
data_disc=json.loads(response.text)
fetch_glucose=data_disc['field1']
print("Glucose",fetch_glucose)
print(":",data_disc['field2'])
'''

model = pickle.load(open('model.pkl', 'rb'))

dataset = pd.read_csv('diabetes.csv')

dataset_X = dataset.iloc[:,[1, 2, 5, 7, 0]].values

from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range = (0,1))
dataset_scaled = sc.fit_transform(dataset_X)



@app.route('/')

def home1():
     return render_template('login.html')

@app.route('/login', methods =['GET', 'POST'])
def login():
        msg = ''
        if request.method == 'POST':
                username = request.form['username']
                password = request.form['password']
                session["user"]=request.form.get('username')
                print(session["user"])

                cursor = mysql.connection.cursor(MySQLdb.cursors.
                    DictCursor)
```

```
                cursor.execute('SELECT * FROM USER WHERE user = % s AND
                    pass = % s', (username, password, ))
                account = cursor.fetchone()

                if account:
                        session['loggedin'] = True
                        session['id'] = account['id']
                        session['user'] = account['user']
                        print("ok")
                        msg = username
                        return redirect(url_for('home',username = msg))
                        return render_template('home.html', username =
                           msg )
                elif not username or not password:
                        msg = 'Please fill out the form !'

                else:
                        msg = 'Incorrect username / password !'

        elif request.method == 'POST':
                msg = 'Please fill out the form !'
        return render_template('login.html', invalied = msg)

@app.route('/home')
def home():
        msg = session["user"]
        return render_template('home.html',username = msg)



@app.route('/feedback', methods =['GET', 'POST'])
def feedback():

        msg = ''
        if request.method == 'POST':

                name = request.form['name']
                email = request.form['email']
                no = request.form['no']
                message = request.form['Message']
```

```
            if not name or not email or not no:
                    msg = 'Please fill out the form !'

            elif not re.match(r'[A-Za-z]+', name):
                    msg = 'Name should only contain Letters!'

            elif not re.match(r'\b[A-Za-z0-9._%+-]+@[A-Za-z0
               -9.-]+\.[A-Z|a-z]{2,}\b', email):
                    msg = 'Invalid email address!'
            elif not re.match(r'[0-9]+', no):
                    msg = 'Number Should only Digits!'
            elif not re.match(r'^\d{10}$', no):
                    msg = 'Phone number should be exactly 10 digits
                       !'
            else:
                    cursor = mysql.connection.cursor(MySQLdb.
                       cursors.DictCursor)
                    cursor.execute('INSERT INTO feedback VALUES (%
                       s, % s, % s, % s)', (name,email,no,message)
                       )
                    mysql.connection.commit()
                    msg = 'Thank you! Your submission has been
                       received!'

        elif request.method == 'POST':
            msg = 'Please fill out the form !'
        return render_template('contact.html',answer = msg)

@app.route('/about')
def about():
        return render_template('about.html')


@app.route('/predict1')
def predict1():
        return render_template('predict1.html')
```

```
@app.route('/index')
def index():

                url="https://thingspeak.com/channels/2091780/feeds/last
                    .json?api_key=F2BEH7B2BK6ST3W9"
                response=requests.get(url)
                data_disc=json.loads(response.text)
                fetch_glucose=data_disc['field1']
                fetch_bp=data_disc['field2']
                print("Glucose",fetch_glucose)
                print("BP:",data_disc['field2'])
                return render_template('index.html',glucose=
                    fetch_glucose , bp=fetch_bp)



@app.route('/past')
def past():
    user = session["user"]
    username = user
    cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
    cursor.execute('SELECT * FROM data WHERE username = %s', (username
        ,))
    data = cursor.fetchall()
    if not data:
        msg = 'No data found for user'
        return render_template('past1.html', account=msg)
    else:
            return render_template('past1.html', account=data)

@app.route('/signup', methods =['GET', 'POST'])
def signup():
        msg = ''
        if request.method == 'POST':

                name = request.form['name']
                username = request.form['username']
                password = request.form['password']
                cursor = mysql.connection.cursor(MySQLdb.cursors.
                    DictCursor)
```

```
            cursor.execute('SELECT * FROM user WHERE user = % s', (
               username, ))
            account = cursor.fetchone()
            if account:
                    msg = 'Account already exists !'

            elif not username or not password or not name:
                    msg = 'Please fill out the form !'

            elif not re.match(r'[A-Za-z]+', name):
                    msg = 'name must contain only characters!'

            elif not re.match(r'[A-Za-z0-9]+', username):
                    msg = 'Username must contain only characters
                       and numbers !'

            elif (len(password) < 6):
                    msg= 'password should be at least 6'



            else:
                    cursor.execute('INSERT INTO user VALUES (NULL,
                       % s, % s, % s)', (name,username,password))
                    mysql.connection.commit()
                    msg = 'You have successfully registered !'
                    #cursor.execute('CREATE TABLE %s (id int
                       AUTO_INCREMENT PRIMARY KEY,username varchar
                       (20) NOT NULL,glucose int,bp int,height int
                       ,weight int,age int,pregnancy int,logindate
                        datetime default now())',(username))
                    #mysql.connection.commit()
        elif request.method == 'POST':
                msg = 'Please fill out the form !'
        return render_template('signup.html',invalied = msg)




@app.route('/predict',methods=['POST'])
```

```
def predict():
        if request.method == 'POST':

                glucose = request.form.get('glucose')
                if not glucose:
                        url="https://thingspeak.com/channels/2091780/
                            feeds/last.json?api_key=F2BEH7B2BK6ST3W9"
                        response=requests.get(url)
                        data_disc=json.loads(response.text)
                        fetch_glucose=data_disc['field1']
                        glucose=fetch_glucose
                else:
                        glucose = float(request.form['glucose'])

                bp = request.form.get('bp')
                if not bp:
                        url="https://thingspeak.com/channels/2091780/
                            feeds/last.json?api_key=F2BEH7B2BK6ST3W9"
                        response=requests.get(url)
                        data_disc=json.loads(response.text)
                        fetch_bp=data_disc['field2']
                        bp=fetch_bp
                else:
                        bp = float(request.form['bp'])

                weight = float(request.form['weight'])
                height = float(request.form['height'])
                meter = height / 100
                square  = meter * meter
                bmi = weight / square
                age = int(request.form['age'])
                preg = int(request.form['pregnancy'])
                print("BMI = ",round(bmi,2))

                data = np.array([[ glucose, bp, bmi, age, preg]])

                prediction = model.predict( sc.transform(data) )

                if prediction == 1:
```

```
                        pred = "Diabetes"
                elif prediction == 0:
                        pred = "No Diabetes"
                output = pred
                user=session["user"]

                username=user
                cursor = mysql.connection.cursor(MySQLdb.cursors.
                    DictCursor)
                cursor.execute('INSERT INTO data VALUES (now(),0,%s,%s
                    ,%s,%s,%s,%s,%s,%s)', (username,glucose,bp,weight,
                    height,age,preg,pred))
                mysql.connection.commit()



@app.route('/logout')
def logout():
    session.clear()  # Clear the session data
    session['log']=""
    return '''<script>alert("You Are Loged Out ");window.location="/"</
        script>'''
    return redirect(url_for('login'))  # Redirect to the login page

if __name__ == "__main__":
    app.run(debug=True)
```

A.1 Training

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import csv

import warnings
warnings.filterwarnings('ignore')
```

```
import pickle
dataset = pd.read_csv('diabetes.csv')

 # Step 3: Data Preprocessing

dataset_X = dataset.iloc[:,[1, 2, 5, 7, 0]].values
dataset_Y = dataset.iloc[:,8].values
dataset_X

from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range = (0,1))
dataset_scaled = sc.fit_transform(dataset_X)

dataset_scaled = pd.DataFrame(dataset_scaled)

X = dataset_scaled
Y = dataset_Y



from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size =
    40, random_state = 42, stratify = dataset['Outcome'] )

# Step 4: Data Modelling

from sklearn.svm import SVC
svc = SVC(kernel = 'linear', random_state = 42)
svc.fit(X_train, Y_train)

# Logistic Regression Algorithm
from sklearn.linear_model import LogisticRegression
logreg = LogisticRegression(random_state = 42)
logreg.fit(X_train, Y_train)

# K nearest neighbors Algorithm
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 24, metric = 'minkowski', p =
    2)
```

```
knn.fit(X_train, Y_train)

# Naive Bayes Algorithm
from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()
nb.fit(X_train, Y_train)

# Decision tree Algorithm
from sklearn.tree import DecisionTreeClassifier
dectree = DecisionTreeClassifier(criterion = 'entropy', random_state =
    42)
dectree.fit(X_train, Y_train)

# Random forest Algorithm
from sklearn.ensemble import RandomForestClassifier
ranfor = RandomForestClassifier(n_estimators = 11, criterion = 'entropy
    ', random_state = 42)
ranfor.fit(X_train, Y_train)

svc.score(X_test, Y_test)
logreg.score(X_test, Y_test)

Y_pred_logreg = logreg.predict(X_test)
Y_pred_knn = knn.predict(X_test)
Y_pred_svc = svc.predict(X_test)
Y_pred_nb = nb.predict(X_test)
Y_pred_dectree = dectree.predict(X_test)
Y_pred_ranfor = ranfor.predict(X_test)

from sklearn.metrics import accuracy_score
accuracy_logreg = accuracy_score(Y_test, Y_pred_logreg)
accuracy_knn = accuracy_score(Y_test, Y_pred_knn)
accuracy_svc = accuracy_score(Y_test, Y_pred_svc)
accuracy_nb = accuracy_score(Y_test, Y_pred_nb)
accuracy_dectree = accuracy_score(Y_test, Y_pred_dectree)
accuracy_ranfor = accuracy_score(Y_test, Y_pred_ranfor)

# Accuracy on test set
print("Logistic Regression: " + str(accuracy_logreg * 100))
```

```
print("K Nearest neighbors: " + str(accuracy_knn * 100))
print("Support Vector Classifier: " + str(accuracy_svc * 100))
print("Naive Bayes: " + str(accuracy_nb * 100))
print("Decision tree: " + str(accuracy_dectree * 100))
print("Random Forest: " + str(accuracy_ranfor * 100))

Y_pred = svc.predict(X_test)
pickle.dump(svc, open('model.pkl','wb'))
model = pickle.load(open('model.pkl','rb'))
print(model.predict(sc.transform(np.array([[86, 66, 26.6, 31]]))))
```

Appendix B

Bill of Purchasing Materials

No. of Products	Model	HSN Code	Quantity	Price	Tax	Total
Male To Female Jumper Wires (10cm) - 40 Pieces pack	EC-8979		1	Rs.54.00	IGST (18%)	Rs.54.00
MAX30100 Pulse Oximeter Heart Rate Sensor Module	EC-3212		1	Rs.119.00	IGST (18%)	Rs.119.00
UNO R3 SMD Atmega328P Board - Clone Compatible Model	EC-2111		1	Rs.385.00	IGST (18%)	Rs.385.00
DS18B20 Water Proof Temperature Sensor Probe	EC-1265		1	Rs.63.00	IGST (18%)	Rs.63.00
ESP-01 ESP8266 Serial WIFI Transceiver Module	EC-0804		1	Rs.108.00	IGST (18%)	Rs.108.00
400 Points Half Size Solderless Breadboard	EC-0719		1	Rs.42.00	IGST (18%)	Rs.42.00
USB A To B Cable - Cable for Arduino - Blue Color	EC-0657		1	Rs.40.00	IGST (18%)	Rs.40.00
12V 1A DC Power Supply Adapter	EC-0116		1	Rs.99.00	IGST (18%)	Rs.99.00
					Sub-Total	Rs.910.00
					Free Shipping	Rs.0.00
					IGST (18%)	Rs.163.80
					Total	Rs.1,073.80

Figure B.1: bill of Purchasing Material

COMPONENT	QUANTITY	COST
Arduino Uno	1	400
MAX30100	1	120
ESP266 (Wifi Module)	1	110
IR Sensor	2	20
Cable	1	50
Jumper Wire(40 piece)	1	60
BreadBoard	1	60
Resistor	5	20
LED	2	10
LCD Display	1	300
Total		1150 Rs

Table B.1: Bill of Material