

Towards an Internet-of-Things and Machine Learning-based Health Monitoring and Management System.

Project report submitted in partial fulfilment of the requirement for
the degree of Bachelor of Technology

in

Computer Science and Engineering/Information Technology

By

SATVIK GARG (181303)
PRADYUMN PUNDIR (181329)

UNDER THE SUPERVISION OF

Dr. [Pradeep Kumar Gupta](#)



Department of Computer Science & Engineering and
Information Technology

**Jaypee University of Information Technology,
Waknaghat, 173234, Himachal Pradesh, INDIA**

I

Candidate's Declaration

I hereby declare that the work presented in this report entitled “**Towards an Internet-of-Things and Machine Learning-based Health Monitoring and Management System**” in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from January 2022 to May 2022 under the supervision of (**Dr. Pradeep Kumar Gupta**) (Associate Professor).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Satvik Garg
(181303)

Pradyumn Pundir
(181329)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. [Pradeep Kumar Gupta](#)
Associate Professor
Computer Science & Engineering and Information Technology
Jaypee University of Information Technology, Waknaghat
8/3/2022

II

Acknowledgement

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully.

I am really grateful and wish my profound indebtedness to the Supervisor **Dr. Pradeep Kumar Gupta, Associate Professor**, Department of CSE Jaypee University of Information Technology, Wazirpur, Delhi. Deep Knowledge & keen interest of my supervisor in the field of **“Machine Learning” and “Internet of Things”** to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Pradeep Kumar Gupta**, Department of CSE, for his kind help to finish my project.

I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

Satvik Garg (181303)

Pradyumn Pundir (181329)

III

Table of Contents

Content	Page No.
Candidate's Declaration	I
Acknowledgement	II
List of Abbreviations	IV
List of Figures	V
List of Tables	VI
Abstract	VII
1. Introduction	1-8
1.1 Introduction	1
1.2 Problem Statement	4
1.3 Objective	4
1.4 Methodology	5
1.5 Technical Requirements	7
2. Literature Survey	9-13
3. System Development	14-38

3.1 Dataset Description	14
3.2 Algorithms for the Proposed Model.....	15
3.2.1 Decision Tree.....	16
3.2.2 Random Forest.....	16
3.2.3 KNN.....	16
3.2.4 XGBoost.....	16
3.2.5 Extra Trees.....	17
3.2.6 Hyper Optimization.....	17
3.3 Proposed Framework.....	18
3.3.1 Firebase.....	19
3.3.2 Html.....	20
3.3.3 CSS.....	20
3.3.4 Bootstrap.....	21
3.3.5 Jinja.....	22
3.3.6 Flask.....	22
3.3.7 JavaScript.....	22
3.3.8 Working.....	23
3.4 Website Hosting.....	29
3.4.1 DNS (Domain Name System).....	30
3.4.2 Linux.....	31
3.4.3 Apache.....	31
3.4.4 MYSql.....	32
3.4.5 PhP.....	32
3.5 IoT-based Weighing Scale.....	33
3.5.1 Hx711 Load cell Amplifier.....	37
3.5.2 Node MCU.....	37
3.5.3 LCD Display.....	38

4. Performance Analysis	39-43
5. Conclusions	44
6. References	45-50
7. Appendices.....	51

IV

List of Abbreviations

1. KNN - K Nearest Neighbour
2. GA - Genetic Algorithm
3. RS - Random Search
4. GS - Grid Search
5. IoT - Internet of Things
6. GBD - Global Burden of Disease
7. BMI - Body Mass Index
8. TPOT - Tree-based Pipeline Optimization Tool
9. GPU - Graphics Processing Unit
10. CNN - Convolutional Neural Networks
11. SVM - Support Vector Machine
12. RNN - Recurrent Neural Network
13. LSTM - Long short-term memory
14. OPT - Optuna Framework
15. RMSE - Root Mean Squared Error
16. HPO - Hyperparameter Optimization

V

List of Figures

Figure 1: Diabetes prevalence in the United States has increased by millions of individuals year after year.

Figure 2: Comparison of countries in terms of percentage of total population with BMI greater than 30

Figure 3 : Components of the Internet of Things.

Figure 4: Obesity Level Dataset Description

Figure 5: Body Fat Percentage Dataset Description

Figure 6: Workflow of the Proposed Model

Figure 7 : User Interface

Figure 8 : Sign In Portal

Figure 9 : User Email Verification Portal

Figure 10 : Screenshot of the authentication code generated and sent to the user email address

Figure 11: Portal to enter the details to create a user

Figure 12: Portal for BMI and Obesity Predictor

Figure 13 : Progress Tracker

Figure 14 : Web Portal for Diet Plan

Figure 15 : Calories Calculator Portal

Figure 16 : Web Portal for Weighing Scale

Figure 17 : Flowchart of IoT Model

Figure 18 : Connection between load cell and hx711

Figure 19 : Proposed Flow of Events

Figure 20 : Circuit Diagram

Figure 21: Hx711 Load Cell Amplifier

Figure 22: Node MCU

Figure 23 : LCD Display

Figure 24 : Plot of Optimization History

Figure 25: Test samples for testing the machine learning algorithms

VI

List of Tables

Table 1: Predicted Obesity Level Accuracy

Table 2: Predicted Body Weight Metrics

Table 3: Predicted Body Fat Percentage Metrics

Table 4: Differences in body fat percentages between real values and machine learning techniques.

VII

Abstract

In urban regions, sedentary life has reached an all-time high due to technological advancements in the last several years. As a consequence, young people are becoming fat and obese. Among the many negative effects of being overweight or obese include diabetes, heart disease, high blood pressure, and a host of other conditions. Recent years have seen a proliferation of machine learning applications in a range of sectors such as financing, banking, social good including forecasting and healthcare. In this work, we specifically touched upon the healthcare domain to develop a system that leverages machine learning and components of internet of things for measuring health levels and thus recommend a proper plan for future health goals of users. A dashboard for tracking progress and customising diet and exercise routines are also part of the system design. Using a range of daily living factors of users, we also include a framework that employs machine learning techniques such as Random Forest, Decision Tree, XGBoost, Extra Trees, and KNN to train models that can predict obesity levels, body weight, and fat percentage levels. In addition, as a part of machine learning subdomain, we analyzed variety of methods to improve the model's accuracy using hyperoptimization algorithms, including gridsearchCV, randomsearchCV, optuna framework and genetic algorithm. The framework is built using Python Flask. Using Internet of Things (IoT)-enabled weighing scales, the framework can also keeps track of the calories and macronutrients that are consumed by the users.

Chapter 1: Introduction

1.1 Introduction

Each year, around 4 million individuals die as a result of obesity, according to a report published in GBD 2017 Obesity Collaborators [1]. Weight gain in children and adults has increased from 4 percent in 1975 to 18 percent since then. Obesity and overeating are not just a concern in developed nations; they are also a worldwide one. There are 30 percent more instances in emerging and low-income nations than in industrialised countries [2], according to the World Health Organisation. A few of the drawbacks of being overweight include mental health difficulties and minor physical diseases, just to name a few. On the other hand, there does not seem to be a direct link between emotional well-being and obesity. In contrast, segregation increases the chance of poor self-esteem, depression, and motivation difficulties, as well as eating disorders, and communication problems in relationships, all of which impair persons ability to be happy [3]. Type 2 diabetes, high blood pressure, heart and renal disease, asthma, back pain, and other physical medical disorders are examples of physical medical conditions. As a result of this, about 3,000,000 people are murdered or seriously affected every year [4].

In today's world, technological innovation has enabled the development of sensor devices, as well as their refinement, which are useful in the monitoring and control of one's health. Applications Programs for cellphones and laptops may be quite useful in the maintenance of healthy lifestyles nowadays. Keeping tabs on obese patients is also a cinch for doctors, and they may counsel patients on how to help themselves stay healthy by increasing their physical activity levels. As a result, many people who are struggling with obesity end up trying a range of treatments and medications that are both ineffective and expensive [2]. Figure 1 depicts the rising prevalence of diabetes in the US population.

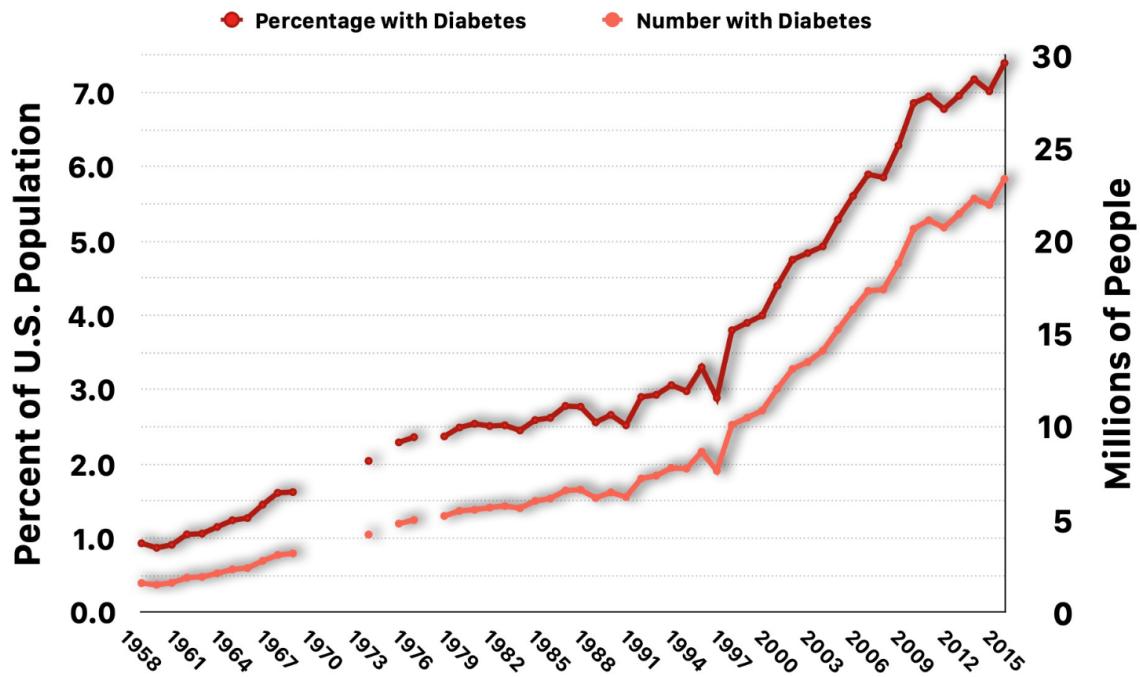


Figure 1: Diabetes prevalence in the United States has increased by millions of individuals year after year [65].

It is beneficial for individuals to utilise body mass index (BMI) to anticipate their own risk of becoming fat in order to encourage themselves to think about the problem of obesity from a more comprehensive viewpoint. The body mass index (BMI) of a person is calculated by dividing their weight in kilograms by their height in metres, which is represented as a square. A body mass index (BMI) more than 25 may be an indication of overweight or obesity, depending on the individual. An individual's BMI is influenced by a number of variables, including their eating habits, physical activity levels, water intake, and the quantity of unhealthy beverages and foods they consume [5]. However, body mass index (BMI) is not a reliable indicator of a person's total body fatness. When it comes to determining overall fitness, the BMI is out of date since it does not account for lean body mass [6]. Because of their lean body mass, which takes up less space, two persons of the same weight and height may seem to be very different. An individual's metabolism will be faster and less fat cells will be stored if the user has higher muscle mass. It is possible to determine a person's body fat percentage based on factors such as gender, arm thickness, and waist size [13].

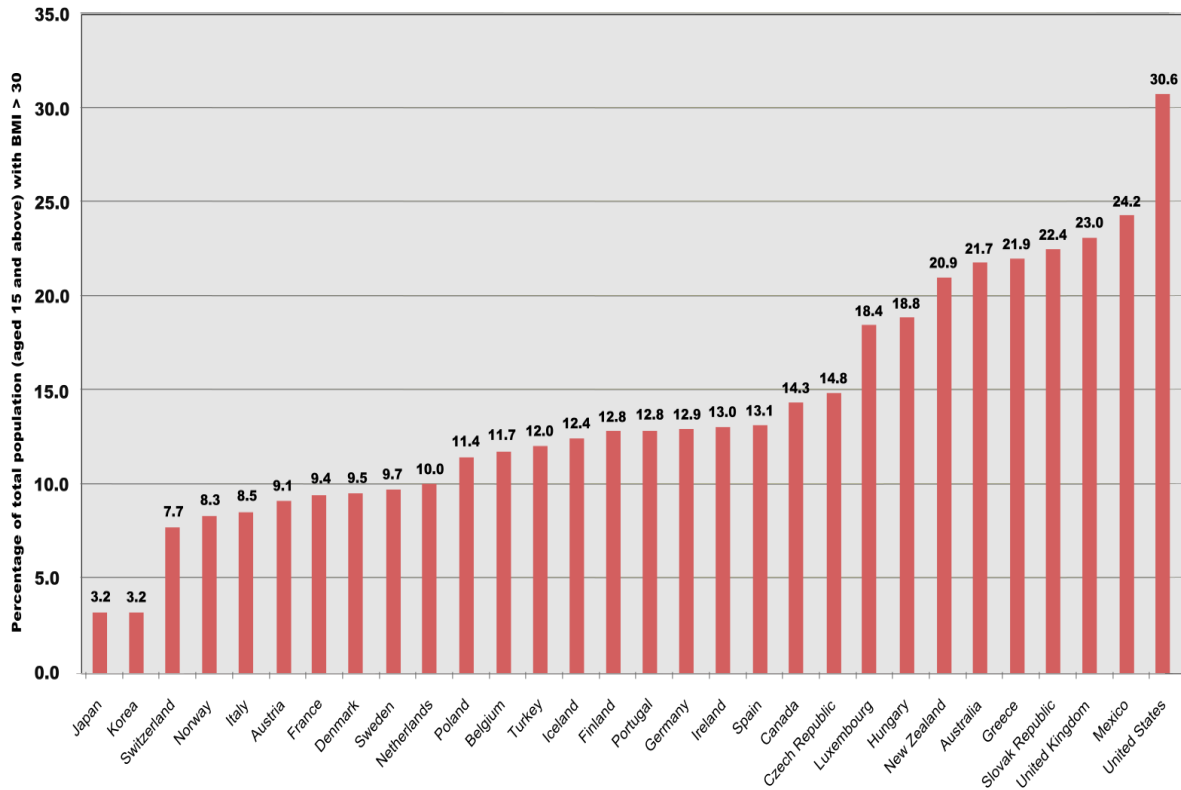


Figure 2: Comparison of countries in terms of percentage of total population with BMI greater than 30.

In order to maintain excellent health, it is necessary to consume foods that are high in nutritional value. Even though there is a plethora of disinformation present online about eating regimens that are not long-term sustainable in the fitness and health sector [7], there is some truth to it. People lose motivation when they do not perceive any improvement in their weight reduction objectives as a result of a lack of understanding about the caloric content of meals, such as the percentage of calories acquired from macro- and micronutrients. It is thus necessary to give a comprehensive online platform that forecasts BMI or obesity levels, as well as body fat percentage, and that also suggests dietary needs in accordance with the users' desired outcomes. One component of the architecture is a smart Internet of Things weighing scale that can tell you how many calories are in the food a user consumes. Line plots are used specifically on a dashboard to monitor a person's progress of fitness in a certain direction.

Fat percentages, bodyweight and obesity levels, are predicted using machine learning algorithms such as Random Forest, KNN, XGBoost, Extra Trees, and Decision Tree, amongst other techniques. Following that, the algorithms are optimised using the Genetic algorithm,

Random Search, Grid Search, and Optuna, among other methods of optimization. For the weighing scale, we used the Python Flask framework and components such as the HX711 load cell and the NodeMCU, among other things. The work completed for an exam will be referred to as: Section II is dedicated to works that have a relationship to a health-related issue in one way or another. The approaches that were utilised to develop the models and systems are shown in Segment III. In Section IV, we will learn how to explore and assess forecasts by using a range of different indicators. The basis of the website is discussed in detail in Segment V. Section VI provided an illustration of Internet of Things weighing scales. Portion VII of the document is the last section of the paper.

1.2 Problem Statement

Obesity is a big problem in today's society. Some of the disadvantages of being overweight include mental health issues and minor physical disorders, to mention a few. A substantial proportion of people assume that being overweight is similar to having cancer. It is also a major risk factor for diabetes, cardiovascular disease, and a variety of other conditions. As a consequence, it is critical to provide individuals with a comprehensive online platform that forecasts BMI or obesity levels, as well as body fat percentage, and also suggests what to eat depending on the users' goals. We used an Internet of Things scale to calculate the number of calories in a meal for tracking calories. On a dashboard, line graphs are used to demonstrate how far a person has progressed in a certain direction. Its diverse uses may be seen in the everyday lives of humans.

1.3 Objectives

A substantial amount of study has been conducted on the data collected by sensors and trackers for human activity detection. This information may be utilized for a multitude of purposes, including tracking all body movements, calculating how many calories one burned while exercising, measuring your pulse, and more. Wearable devices, on the other hand, are not affected by occlusion or view and can only gather low-dimensional sensor data from accelerometers, gyroscopes, and other sensors. Even while the bulk of these strategies work, they are not without problems. In terms of overall well-being and health, development is gradual [8]. It was also the focus of a few studies and research initiatives that investigated

general health concerns such as calorie estimations, IoT-based smart health systems, and BMI calculations. Moreover, merging all of these applications for a smart health system has not been done and is limited. As a consequence, we provide methodologies that use machine learning and IoT to build a multi-stream framework that assists in the identification of obesity in real time with high accuracy.

This project demonstrates how to build a web application that can be integrated with the Internet of Things' specified architecture. Data from a IoT device is retrieved by the application, which then sends it to a "collector." Anything used to gather and analyze data, such as text messages or medical advice, is referred to as a collector. This data must be readily available through all conceivable connections. Developing a web application that works with components of machine learning models being deployed is difficult, thus the purpose of this project was to propose the simulation version of the IoT based smart weighing scale. For healthy lifestyles, the system was designed to use a web application to get data from a web server, analyze and process the data collected, then provide a user-friendly user interface.

1.4 Methodology

The Internet of Things (IoT), which has been under development, is playing an essential part in the development of technological advancement, network topologies, and communication channels. When we say "things," we mean any physical object, which encompasses a broad range of devices, cars, and infrastructural components [60]. These entities are distinguishable and unique from one another because they are connected to the network. According to [61], the Internet of Things (IoT) is supposed to be about connecting different pieces of technology and objects that people use on a regular basis to one another via the use of the internet so that they may carry out a variety of different functions. Such systems may communicate with each other and analyze information.

The Internet of Things enables the combination of a broad range of sensors and other types of end devices with products that are hooked up to the internet via both wire and wireless connections. Mobile devices currently employ more than 14 sensors to generate raw data

such as current position and motion. The ubiquitous use of mobile computers makes this feasible. The Internet of Things (IoT) may offer a powerful communication platform for all conventional Internet kinds, allowing internet-enabled and non-internet-enabled objects to detect, connect, and exchange data [62]. This platform may be made accessible by incorporating IoT into a variety of activities. On the other hand, the healthcare industry has recently attracted considerable attention as a significant application sector that is expanding into the Internet of Things domain. A major reason for this is the addition of prescriptions, consultations, continual monitoring devices and tracking of food intake. Patient physiological indicators such as heart rate, blood glucose level, and body temperature are now routinely tracked using an array of medical equipment and sensors [63]. Patients who live far from medical facilities might benefit greatly from the use of medical sensors, which can measure and monitor several human physiological characteristics. Real-time information on a patient's status allows for a wide range of options in terms of better understanding the user's condition and faster reactions to health issues [64]. There are several applications for the Internet of Things (IoT), and the number of these applications is continually expanding. People's lives are going to become better as a result of the Internet of Things since it will be able to gather and analyze data that may lead to helpful information and knowledge.

The Internet of Things, according to a recent research [64], comprises numerous ways of communication. These modalities include data communication between devices or machines (M2M), transmission of data between machines and person (M2P), and data exchange between people and people (P2P). The term "M2M communication" refers to a connection that may be established between machines and other electronic devices without the need for human participation. Figure 3 depicts the components of the Internet of Things.

In this work, three machine learning models were constructed to predict obesity levels, body weight, and body fat percentage, among other things. Before using the models, a variety of machine learning methodologies were applied to prepare them for comparison. Machine learning algorithms may have their hyperparameters adjusted in order to improve and raise the model's accuracy. It may be difficult to forecast how to create hyperparameters and communicating hyperparameter combinations that will have the most influence on the model as a whole for a certain dataset.

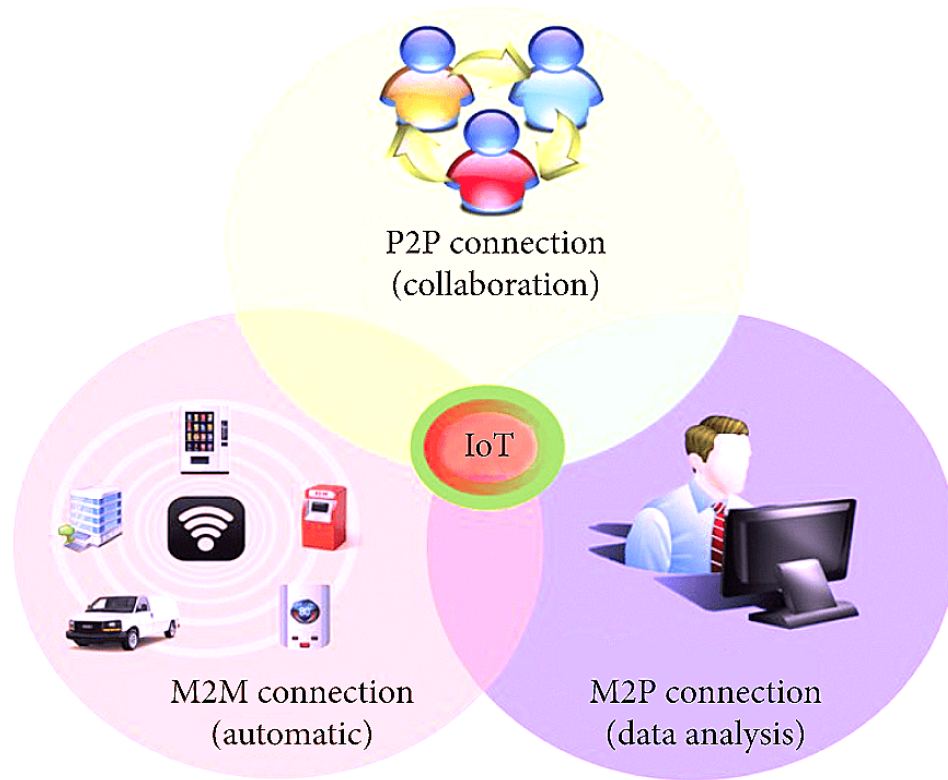


Figure 3 : Components of the Internet of Things.

Among the methods used for optimization are HPO models such as Random Search, Grid Search, Genetic Algorithm, and Optuna. To be more specific, we used the scikit package's RandomSearchCV and GridSearchCV methods to do random and grid searches, respectively. In addition, we implemented a tool called the Tree-based Pipeline Optimization Tool (TPOT) and Optuna to develop the evolutionary technique [29]. The TPOT, an automated machine learning tool, was created using a genetic search technique. To restore accuracy of Optuna, the trial object is responsible for supplying the hyper-parameters, while the research object carried out the final execution to train the algorithm.

1.5 Technical Requirements

This project necessitates high technical requirements. For data gathering and training, Python integrated libraries like as pandas and numpy were used. The scikit framework is used to call machine learning algorithms, which is constructed on a jupyter notebook and runs on a Google collaborative server with the GPU turned on. The Python programming language is

used for machine learning coding development. In terms of web application development, we used basic web technologies such as HTML and CSS for frontend development. The machine learning models were integrated using the flask library and successfully implemented on a local host using the pycharm IDE. The IoT based smart weighing scale system is developed using components namely, HX711, load cell, NodeMCU.

The following is how this examination work is organized: Section II comprises works that are linked to health frameworks. Section III describes the approach used to create the models and Internet - of - things system, as well as the website framework's functionality. Section IV introduces the analysis and evaluation of forecasts using several metrics. Section V brings the paper to a conclusion.

Chapter 2: Literature Survey

Much research was conducted on the obtained datasets collected by trackers and different sensors for detection of human activities. This information may be utilized for a number of purposes, including tracking all body movements, calculating how many calories one expended while exercising, measuring the pulse, and more [8]. In terms of overall well-being and health, development is gradual. It was also the focus of a few studies and research initiatives that investigated general health concerns like estimation of calories, Internet of Things based health systems, and body mass index calculations.

A mobile-based nutritional assessment software tool has been developed to enhance eating habits [9]. The authors examined real-time photographs of meals to determine how many nutrients they contained. To categorize photographs, the database of more than 100 classes and a mean of 1000 shots for every class were employed and deep learning techniques were utilized to accomplish this. There are now additional dishes from the subcontinent in the food-101 dataset. The findings reveal that the model's accuracy is 85% and that it is also quite effective on the crucial Food 101 dataset that was employed.

According to Enes Kocabey and colleagues [10], FACE2BMI may predict BMI levels by taking into account how a high BMI affects daily activities. The images were selected from the visual-BMI dataset and manually cut down to 4206 images. The two VGG based CNN models were used to search for characteristics in order to identify them. The BMI was then calculated using a support vector regression model. In terms of total accuracy, VGG-Face scored 65 percent, whereas VGG-Net scored 47 percent accuracy.

Sebastian Baumbach and his colleagues examined nine-games and health routines commonly found in gyms, such as abdominal workouts and movements like deadlifts and bench presses [11]. The author collected sensor data from 23 participants who wore smartwatches and utilized smartphones. The neural network was trained in this manner. Deep learning models have a 92 percent accuracy rate, which is higher than the 80 percent accuracy rate of SVM and Naive Bayes.

Mehak Gupta and her colleagues [12] also created a model that predicted a user's future weight levels using medical information from their childhood. This helps to figure out what to do before and how to do it in a more practical manner to prevent adult obesity by determining which children are at risk of being fat throughout childhood. LSTMs were applied to an EHR dataset that had not been supplemented. Set embeddings and attention layers were also used to aid in the comprehension of features and timestamps.

The growth of Internet of Things–based technologies is beginning to transform many people's health, particularly in the many care activities related with the healthcare system. In this area, IoT adoption may be used by providing vast opportunities for observation, research, treatment, and recommendations of illnesses such as being overweight or obese. Innovative applications, such as those for wearable devices and smartphones, have enabled Remote Patient Monitoring (RPM), which gives a better understanding of the patient's condition.

In the industry, the biggest chunk of mobile and wireless health technologies are referred to as "mHealth." There are a plethora of applications for smartphones accessible for download on the market. Furthermore, the advancement of IoMT has resulted in the development of monitoring applications, which has resulted in significant improvements in usability across a range of platforms [35, 36, 37]. According to a mobile health study, about 97,000 applications are registered at various marketplaces available for different operating systems were reported to be healthcare applications [37, 38]. The great majority of these apps are related to overall fitness and health or physical activity. For instance, a study published in [39] proposed a digital healthcare system whose aims were to promote awareness amongst adults and caregivers about the dangers of overweight and to aid them in keeping a healthy eating and dietary lifestyle. Furthermore, the authors of [40] proposed a mobile health architecture with the goal of preventing childhood obesity by encouraging healthy behaviors and prescribing automated methods through the use of a set of mHealth applications capable of providing alerts and texts to seniors in order to obtain the information. The goal of this mobile health architecture was to reduce childhood obesity by encouraging healthy habits and prescribing automated methods.

[41] investigated numerous software products for exercise programs that use employ a dedicated weighing scale to transmit information to the computer, and the method is known as machine-to-machine communication (M2M). The applications may provide suggestions

regarding the necessary therapies and facilities for treating the patient, as well as generate an analytical report for the physician to review.

Furthermore, the authors of [42] designed a research to test the impact of preventing obesity in adolescents utilizing a portable system builds on agent-based strategy that can collaborate with a smart watch to propose food and physical activity, as well as improve the healthy habits required to avoid juvenile obesity.

Furthermore, the study's authors [43] developed a device capable of storing the medical information of several patients at the same time and allowing real-time monitoring measures to be conducted. Using this Internet of Things device, obese patients and their doctors are able to identify any abnormalities in all of the tracked data.

Both studies in [44, 45] reported on a study of the efficacy of an experimental design based on nutrition suggestions sent by an application for weight reduction in overweight or obese individuals over the age of 18. The researchers concluded that weight reduction applications are beneficial for persons who are willing to restrict their calorie intake. On the other hand, installing an application on a smart phone is unlikely to result in a visible weight change for the average obese patient. Like the previous example, a study called [46] presented an architecture that might be used to monitor an ambulatory person's health condition through a wireless sensor network. The system is made up of a number of sensor nodes that, with the help of a core network and a private server, monitor the heart's and the rest of the body's activity. In addition to this, in order to network the sensors, they used a system design that is founded on a well-known accessible computer operating system. This allowed them to avoid the need to invest in expensive proprietary software.

Health monitoring systems that use the Internet of Things (IoT) were examined in both articles [47, 48]. Health monitoring systems, wireless communication, and security issues were all reviewed as well as the latest trends and developments in these systems' design, as were the drawbacks and benefits of each. The effects of the Internet of Things and mobile applications on human health are comparable to those described in [49, 50], which described a system capable of tracking the customer's weight reduction progress along the way and recommending meals with a nutritionist program to prevent the user from jeopardizing their

health. The effects of IoT and mobile apps on human health are comparable to those found in [49, 50].

IoT-learning has been defined and investigated by a group of experts in the field of IoT technology integration. The research explained in [51] suggested a customized training program for weight loss and ubiquitous learning that would make advantage of IoT-learning. This concept provides customers with not just tailored information, and also knowledge which corresponds to the specific information. IoT-learning provides feedback to users, who are often students, by collecting data from sensors in the surroundings. The major goal of this learning is to provide personalized training to consumers using information collected through IoT sensor technology. This article combines detection technologies in the internet of things (IoT) with learning systems.

There was a desire to investigate the impact of collecting health data through devices while also posing questions regarding when, why, how and with whom the data should be shared, what kinds of details to divulge, and why the data should be used [52, 53]. The protection of one's personal information is a moral quandary that causes great concern. Patients' privacy will be put at risk because of the enormous range of data that these devices may collect in the medical profession, many of which were not even considered possible only a few years ago [54]. There is an urgent need to identify and analyze hazards when it comes to data collecting in the clinical domain and public health, and therefore enhance decision-making when adopting and building a more trustworthy Internet of Medical Things [55, 56]. This is an urgent requirement because there is an urgent need to detect and analyze risks while collecting data for clinical and public health purposes.

Because statistical and numerical data can be acquired from many factors such as pressure, heart rate, and movement, the opportunity to profit from Artificial Intelligence (AI) via the Internet of Things (IoT) is now enormous. This is a fantastic window of opportunity. Also it includes strategies for automatic perceived notion, automated learning, comprehension, and rationale, which are all becoming more common [57]. The ultimate objective of artificial intelligence is to imitate human intellect; nevertheless, AI has several applications, the most significant of which is wellness. This branch of study owes a great deal of its present progress to discoveries made while attempting to diagnose and cure medical conditions [58, 59].

At this point in time, the computer and medical sciences were beginning to take an increased interest in this newly emerging field due to evidence of its usefulness and success. Along with significant achievements, artificial intelligence has a variety of advantages and disadvantages [58, 59]. The advantages include lower expenses, higher incomes, attainment of cutting-edge results, the possibility for progress, and the capacity to deal with complex challenges. However, since it is software, it needs regular updates and maintenance, both of which are time-consuming and must be performed with the aid of expert systems. It is a branch of artificial intelligence that allows computers to learn without being given explicit instructions. The word "intelligent" refers to machine learning's skills, which include the ability to solve problems, collect information from their surroundings, and adapt to their environment.

Chapter 3: System Development

This chapter is divided into various subparts that provides the overall workflow necessary for researchers to put together an investigation for future works.

3.1 Dataset Description

There are two datasets applied in this study: the “Estimation of obesity levels using eating habits and physical condition” dataset from the UCI ML repository (13) and a body fat percentage dataset from the National Institutes of Health (14). This dataset has 252 rows and 19 columns, while the obesity dataset contains 2111 rows and 17 columns. Figures 4 and 5 each include a description of the corresponding dataset, which may be seen by clicking on those figures. The daily living and lifestyle variables that are provided by obesity are the types of factors that may lead to the development of varying degrees of fitness and obesity. The Body Fat percentage dataset contains a variety of characteristics that range from different characteristics of body parts measurement, such as thighs, chests, and biceps, for the purpose of estimating the levels of body fat percentage in males specifically.

Attribute	Description	Attribute	Description
Gender	Gender	FAVC	Frequent consumption of high calorie food
Age	Age (years)	FCVC	Frequency of consumption of vegetables
Height	Height (In m)	NCP	Number of main meals
Weight	Weight (in kgs)	CAEC	Consumption of food between meals
SMOKE	Smoking habit	CH2O	Consumption of water daily
CALC	Consumption of alcohol	SCC	Calories consumption monitoring
MTRANS	Transportation used	FAF	Physical activity frequency
Nobeyesdad	Obesity levels	TUE	Time under technology devices
Family_history_with_overweight		Family member suffered from obesity	

Figure 4: Obesity Level Dataset Description

Attribute	Description	Attribute	Description
S. No	Case Number	B.F.1	Body fat % using Brozek's Equation
Age	Age (years)	B.F.2	Body fat % using Siri Equation
Height	Height (Inches)	Density	Density(gm/cm ³)
Weight	Weight (in lbs.)	AI	Adiposity Level
Abdomen	Abdomen circumference(cm)	Neck	Neck circumference(cm)
Hip	Hip circumference(cm)	Chest	Chest circumference(cm)
Thigh	Thigh circumference(cm)	Ankle	Ankle circumference(cm)
Knee	Knee circumference(cm)	Biceps	extended biceps circumference(cm)
Wrist	Wrist circumference(cm)	Forearm	Forearm circumference(cm)
Fat_free_weight	Fat free weight using Brozek's equation (lbs.)		

Figure 5: Body Fat Percentage Dataset Description

3.2 Algorithms for the Proposed Model

To construct classification and regression models, the KNN, Random Forest, Extra Tree, Decision Tree, and XGBoost algorithms are utilised. As far as general-purpose machine learning is concerned, deep learning techniques, such as neural networks, have consistently outperformed their competitors. Small to medium-sized datasets, on the other hand, are more suited to tree-based machine learning approaches. Beginning with data preparation and ending with model testing, the whole process that was used in this project to construct these algorithms is shown in Figure 6. In the following sections, we took a look at a variety of pipelines that are going to be used to supplement the overall framework.

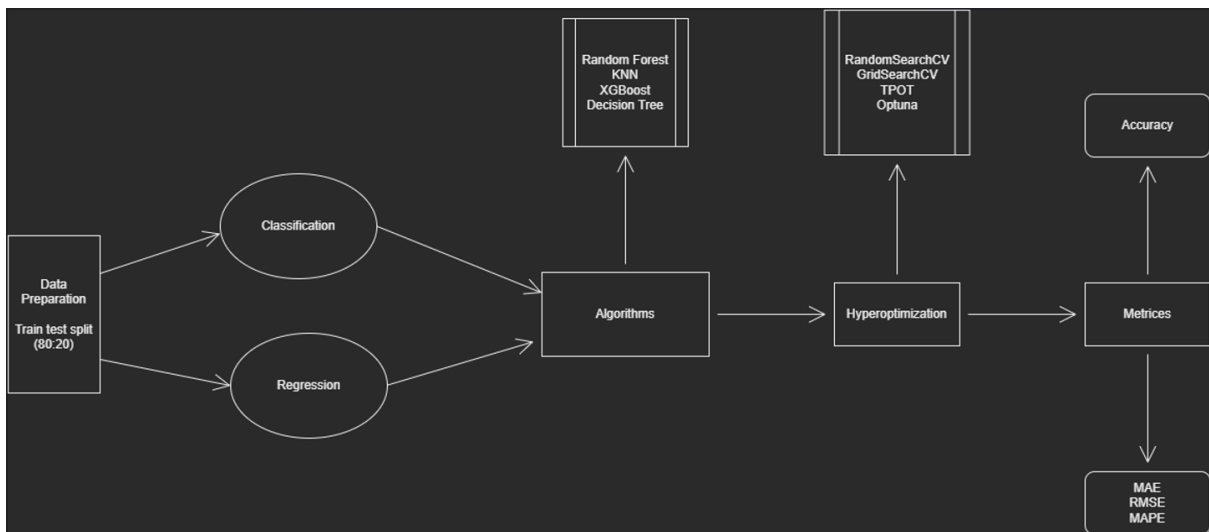


Figure 6: Workflow of the Proposed Model

3.2.1 Decision Tree:

An attribute, link, or branch in a decision tree represents a choice, and the leaf in the tree represents the outcome of the choice [17]. The value of a machine is defined by its categorization rather than by its actual worth. There is a level of difficulty involved with learning. Determiner trees are used to develop an explanatory model that may be utilised to determine the needed class or values of target variable decision-making parameters based on data that has already been acquired. Making forecasts using Decision Trees starts with the creation of a decision tree in the first place. It was essential to do a comparison between the estimated value of an attribute on a record and the actual value of the attribute. When the value of the node's value is compared to the value of the branch, the branch is followed and the next in the correlation is skipped in this instance.

3.2.2 Random Forest:

In order to improve the accuracy of the dataset's prediction, Random Forest [18] uses many decision trees on various subsets of the dataset to achieve this goal. Instead of employing a single decision tree to forecast the final outcome, this technique makes use of predictions from a large number of decision trees. It is based on the notion of ensemble learning, which is the integration of numerous classifiers in order to solve complicated problems and improve the performance of the model.

3.2.3 KNN:

The K-nearest neighbour approach [19] forecasts incoming data points based on feature similarity, and as a result, it is highly reliant on the coordination of the preparation set. In addition, it is referred to as "lazy learner computation" since it does not immediately benefit from the preparation set.

3.2.4 XGBoost:

Extraordinary gradient boosting, or XGBoost [20], is one of the most well-known ensemble-based gradient boosting approaches for tree-based machine learning algorithms, with improved execution and speed. Unlike other boosting algorithms, which try to enlarge the loss function, the goal of gradient boosting techniques is to maximise the loss function.

3.2.5 Extra Trees

Extra Trees, also known as extraordinarily randomised trees, are constructed using an ensemble machine learning approach, which is also known as supervised machine learning. Using the training data, this strategy generates a large number of decision trees that have not been trimmed. When doing regression-based tasks, the predictions of decision trees are averaged, and when performing classification tasks, majority voting is employed. Unlike the random forest, which produces each decision tree from a bootstrap sample of the training dataset, the extra tree fits each decision tree onto the whole training dataset, which is a significant advantage. With the additional tree technique, which selects a random split point and may sample features freely at each split point, the greedy approach is favoured since it allows for more flexibility.

3.2.6 Hyper Optimization

By altering the hyperparameters of a machine learning model, it is possible to increase the accuracy of the model. The use of hyperparameters has been demonstrated to have a considerable influence on the performance of a model; however, determining the optimal combination of parameter and communication parameters for a specific dataset is challenging. Finding model hyperparameters in a more objective manner may be accomplished by scanning a number of characteristics and picking the subset that gives the best model on a particular dataset [22]. The scikit-learn Python machine learning package [23] has a function called hyperparameter advancement, which is also known as hyperparameter tinkering. When using the hyper optimization system, a search space is assigned to you [24]. Despite the fact that each hyperparameter addresses a different optional dimension and the dimensions' range dictates the values that hyperparameters might take (such as categorical or real-valued), it is conceptually equal to a k-dimension space. This means so each contextual characteristics inside search space is shown by a part of a vector in the search space. Optimization is used in order to get the most amount of performance from a model while simultaneously adding the least amount of error possible.

3.3 Proposed Framework:

The website uses various Machine Learning models that help to predict the obesity levels, bodyweight and body fat percentage. The framework also provides its users to make customizable diet plans, workout plans, and a dashboard to track the progress. The frontend of the framework is designed using HTML, CSS, Bootstrap. The backend of the framework is built using JavaScript, Jinja2, Flask. The website has a token-based login and sign up authentication system. The authentication token is generated using the TimedJsonWebSignatureSerializer which is available in the 'itsdangerous' python library. The authentication code is then sent to the user's email address so that they can authenticate their credentials.

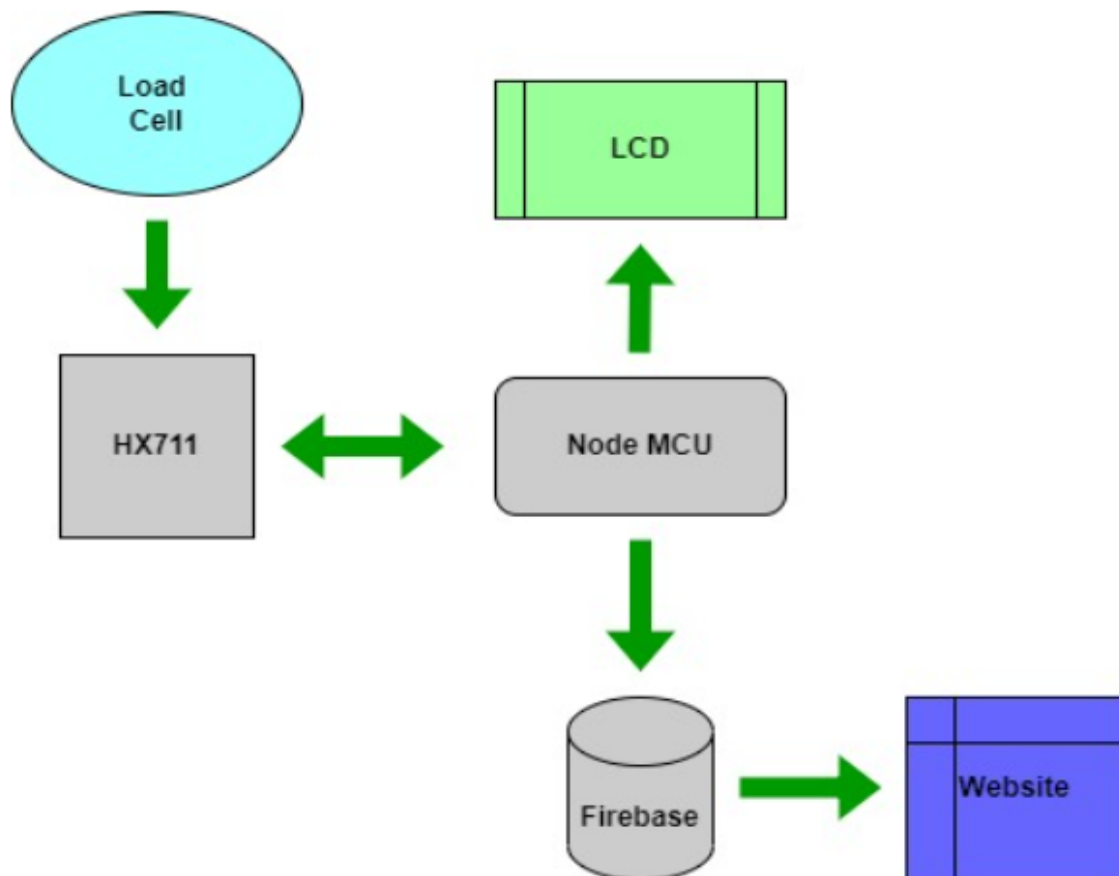


Figure 7 : Proposed Flow of Events of Framework

3.3.1 Firebase:

Firebase is a suite of tools for building apps and managing infrastructure on top of google cloud platform. It was founded by James Tamplin and Andrew Lee in 2011 after they discovered that developers were using their chat software to manage non-chat application data that inspired them to create a real-time database, a Json database that automatically stays in sync with your front-end application. It was later expanded to support user authentication and website hosting to act as a complete backend as a service. In 2014 it was acquired by google where it was further expanded with serverless computing via cloud functions push notifications, via cloud messaging and other google services.

In 2017 google acquired fabric from twitter and also launched a new document database called firestore to support a wider range of applications most importantly though firebase provides software development kits for virtually every platform to manage and scale this infrastructure with little to no back-end code to build a full stack application right now. First, create a free firebase project, then open up some code for your favourite front-end platform, install the firebase SDK then connect it to the cloud with your project credentials . Once initialised you can start interacting with your backend resources first we might want to know if the user is logged in or not we can listen to the current user with the on off state changed function the user will be null at first but we can log them in with their google account using the sign in with pop-up function now that we have a user logged in we might want to store something in a database. We can make a reference to a document in firestore then write JSON data to it while making a reference to the current user's user id but the amazing thing is that we can listen to updates to that data in real time by simply referencing the document using the snapshot function.

Anytime the data changes on the server it will be reflected in the UI with zero latency but that doesn't seem very secure. firestore security rules allow you to define access control logic in a simple readable format with common expression language, you can do almost everything from the front end but when you do need to run server-side code firebase cloud functions provides a tightly integrated serverless backend not only does it allow you to create http endpoints but functions can also be triggered in the background based on events that happen

in your project like rights to the database, user authentication, events file uploads and many others.

When developing locally, firebase has an emulator suite to run and test your code in a mock environment. When it is time to deploy, run the firebase deploy command to push your code to the cloud and allocate infrastructure to run it at any scale.

3.3.2 HTML:

Hypertext markup language or html is a language that defines the structure of content on a webpage development started in 1989 at cern by tim berners-lee who invented the world's first web browser world wide web. But he needed a publishing language to display content within it so he chose a syntax based on the standard generalised markup language which uses opening and closing tags to provide meaning to otherwise unorganised text like h for heading, p for paragraph, and most importantly a for anchor to link to a different web page.

The language underwent many changes during the initial rise of the internet and was standardised as html5 in 2008 the version we know and love today, modern html contains tags to handle a variety of different media types like video audio and canvas to draw custom graphics on the fly. The typical html element starts with an opening and closing tag, in the middle we have content which might be text or additional html elements called children.

The opening tag may also have one or more attributes which are key value pairs that can customise the appearance or behaviour of the element. To build your first web page create a file named index.html this file contains a hierarchy of nodes based on the document object model. A web page starts with an html tag and has exactly two children: a head and a body. The head contains things that are not directly shown in the webpage like a title or a meta tag to give the page a description but notice how the meta element has no closing tag closing tags are generally optional when an element is fully self-contained the body contains the actual ui for the end user.

3.3.3 CSS:

Cascading Style sheets or CSS is a stylesheet language famous for making websites look pretty. By default a browser will render a document as a boring black and white document,

but CSS allows you to describe the presentation of the markup by defining the various features such as width, size, colour, etc.

It was first released in 1996 and evolved into the CSS3 by 1999. The typical CSS will start with a selector that will match to an element of an HTML document, to create a declaration block braces are used that contain the properties that can change the appearance of that element. The CSS code must be written inside the Script tag in a HTML document or a separate CSS file can be created which must have the .css extension and that must be linked with the HTML file.

3.3.4 Bootstrap:

Bootstrap is an HTML, CSS, and JS package that focuses on making the process of creating informative web pages easier (as opposed to web apps). The main purpose to incorporate it in a web project is so that the project may take use of the color, size, font, and layout options given by Bootstrap. As a result, the most crucial factor to consider is whether or not the individuals in charge of producing the game find such options appealing. When Bootstrap is introduced to a project, it instantly starts supplying basic style definitions for all HTML components.

As a result, the appearance of text, tables, and form components is consistent across web browsers. Furthermore, developers may further alter the appearance of their contents by using the CSS classes supplied in Bootstrap. For example, Bootstrap supports tables in both light and dark colors, page headers, larger pull quotes, and highlighted text. Many JavaScript components are included in Bootstrap and are implemented as jQuery plugins. They include extra user interface components such as discussion bubbles, tooltips, and carousels. Each component of Bootstrap is made up of HTML markup, CSS declarations, and, in certain cases, JavaScript code. They also improve the functionality of existing interface components, such as introducing an auto-complete option for input fields.

The layout components of Bootstrap stand out the most since they have an impact on the whole web page. Because another element on the page is positioned within it, the basic component of the design is referred to as a "Container." The developer has the option of using a container with a fixed width or one with a fluid width. The second choice will always fill

the width of the web page, while the former will choose one of five established fixed widths that will vary based on the size of the screen used to display the page.

3.3.5 Jinja

Jinja is an open source web template engine developed in Python. Armin Ronacher created it, and the BSD Licence controls its distribution and usage. Jinja is a template engine similar to Django's, but it differs in that it supports Python-like expressions and ensures that templates are evaluated inside a sandbox. Because it is a text-based template language, this text-based template language may create both markup and source code.

Using the Jinja template engine, users may change not just tags and filters, but also tests and globals. Furthermore, unlike the Django template engine, the Jinja template engine allows the template designer to invoke functions on objects while passing in arguments. Jinja is the template engine that Flask uses by default. Jinja is used by Ansible, Trac, and Salt in addition to Flask.

3.3.6 Flask:

Python powers Flask, a framework for building micro websites. It's a microframework since it does not need the use of any particular tools or libraries to function. Because of this, other libraries can't use it to accomplish basic tasks like database abstraction and form validation. When it comes to Flask, on the other hand, new functionality may be introduced as if it were a native component of the framework. Form validation and upload processing, open authentication systems, and other framework-related utilities are accessible as extensions. The Flask framework is used by a variety of apps, including Pinterest and LinkedIn.

3.3.7 JavaScript:

Javascript is a single-threaded, garbage-collected, interpreted, or just in time compiled prototype-based dynamic language that is widely used in web development. Brendan Eich developed a rudimentary scripting language for the Netscape browser in about one week in

1995. It was originally named Mocha, but marketing gurus wanted it to sound like the new Java language, so it was changed to JavaScript. It is extensively used to build front-end online programs since it is the only non-web assembly language supported by browsers. JavaScript is a powerful and ever-evolving programming language. However, anything that can be written with JavaScript will be built with JavaScript, such as server-side applications using Node. For mobile applications, use React Native or Ionic, and for desktop apps, use Electron. Tools like the V8 engine and Chromium employ a just-in-time compiler to convert scripting language to machine code during runtime.

Despite being a single-threaded language, it excels at handling I/O-intensive activities thanks to a non-blocking event loop that enables background work to be queued without interfering with the main thread. We may log a value in the console using the built-in debugger to observe what's going on in the global context. To run this file in a browser or a back-end Node. The browser's JS server may be accessed using a script tag, and the browser will execute it when we view the HTML file containing the script.

On a website JavaScript is often used to grab an element from the document. Query selector will grab the first button then we can assign it to a variable with either var, let or const. var is for variables that can be reassigned while const is for variables that cannot be reassigned. To make the button interactive we can add an event listener to it by defining a function for the on click event we can use the function keyword here or the more concise arrow syntax, the event loop will execute this function whenever a new click occurs. Functions are first class objects to support functional programming patterns but JavaScript also supports classes and inheritance for object oriented patterns even though it is single threaded it can do work asynchronously. In this website we have used JavaScript to design the customizable diet plan functionality, with the help of which a user can easily plan their diet and customise it accordingly to their nutritional requirements.

3.3.8 Working:

The website employs Machine Learning models to provide predictions about the Body Weight, Body Fat Percentage, and Obesity Level Prediction, and it provides a Dashboard for the user to make it simple for them to keep track of their day-to-day progress. On the website, there is an option to plan a diet that may be customized to meet the needs of the person in

question about their nutritional intake. In addition, the website has a capability that enables users to weigh food items with the assistance of an Internet of Things-based weighing scale and see the nutritional values that are linked with the food item being weighed.

In order to get access to the framework, we are required to sign in to the website in the first place.

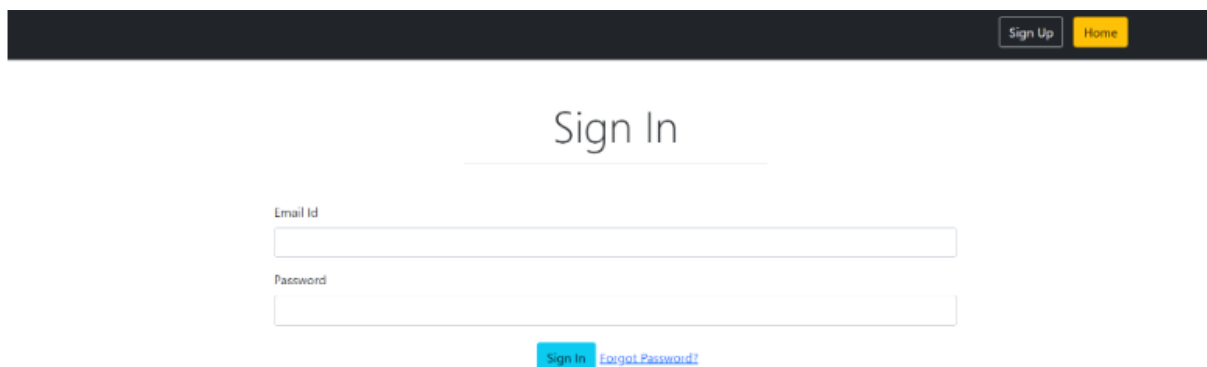
A screenshot of a web application's sign-in page. At the top, there is a dark horizontal navigation bar containing two buttons: 'Sign Up' and 'Home'. The 'Home' button is highlighted in yellow. Below the navigation bar, the text 'Sign In' is centered in a large, light gray font. Underneath this text, there are two input fields: the first is labeled 'Email Id' and the second is labeled 'Password'. Both fields are empty and have a light gray border. Below the password field, there is a blue 'Sign In' button and a blue text link that says 'Forgot Password?'. The entire sign-in form is centered on the page.

Figure 8 : Sign In Portal

If the user does not already have an account, then they will be required to create a new account by simply supplying the requested information. In order to do this, the user will first be required to submit their email address as shown in Figure 9.

An authentication code will be generated as soon as the user gives their email address. This will be accomplished with the assistance of the `TimedJsonBasedWebSerializer` and the `'itsdangerouslibrary'` written in Python. And will give you the produced code so that you can readily check it, as seen in Figure 10.

Enter Your Email Address

Email Id (Please Enter Same Email at the time of Filling signup Details)

Submit

Figure 9 : User Email Verification Portal

please copy this token ==> eyJhbGciOiJIUzUxMiIsImhhbmRhdC8MTY0Mzk3MzAyOSwiZXhwIjoxNjQzOTczMzI5fQ.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiIsImhhbmRhdC8MTY0Mzk3MzAyOSwiZXhwIjoxNjQzOTczMzI5fQ.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiIsImhhbmRhdC8MTY0Mzk3MzAyOSwiZXhwIjoxNjQzOTczMzI5fQ (valid for 5 minutes only)

Reply Forward

Figure 10 : Screenshot of the authentication code generated and sent to the user email address

In order to successfully establish a user account, the user must copy and paste the authentication code into the sign up box. Following the successful creation of the ID, the user is able to Sign In and have access to the features of the Framework. To begin, there is the Obesity Level Prediction as well as the BMI Prediction Model, both of which are shown in Figure 12.

Sign In

Home

Sign Up

First name

Last name

Email Id

Password

Gender

Choose...

Verification Link(Paste here)

☐ Agree to terms and conditions

Submit Details

Already have an account? Sign In

Figure 11 : Portal to enter the details to create a user

BMI and Obesity Predictor

Gender	Age	Weight (in Kg.)
Male	26	101
Family History Overweight	Height (in m.)	Frequent Consumption of High Calorie Food
Yes	1.85	Yes
Frequency of Consumption of Vegetables	Number of Main Meals	Consumption of Food Between Meals
0	2	Sometimes
Smoke	Consumption of Water Daily	Calories Consumption Monitoring
Yes	1	Yes
Physical Activity Frequency	Time Under Technological Devices	Consumption of Alcohol
0	2	No
Transportation Used		
Motorbike		

Calculate

Your BMI = 29.51

Your Obesity Status = Overweight_Level_II

Figure 12 : Portal for BMI and Obesity Predictor

In addition, the website has a dashboard, which allows users to quickly monitor and keep track of their day-to-day progress. For example, in Figure 13, there are two plots that demonstrate how the bodyweights changed on a weekly basis from 86 to 80 throughout the course of the 28-day period. It would be beneficial for them to do a weekly goals and performances analysis.

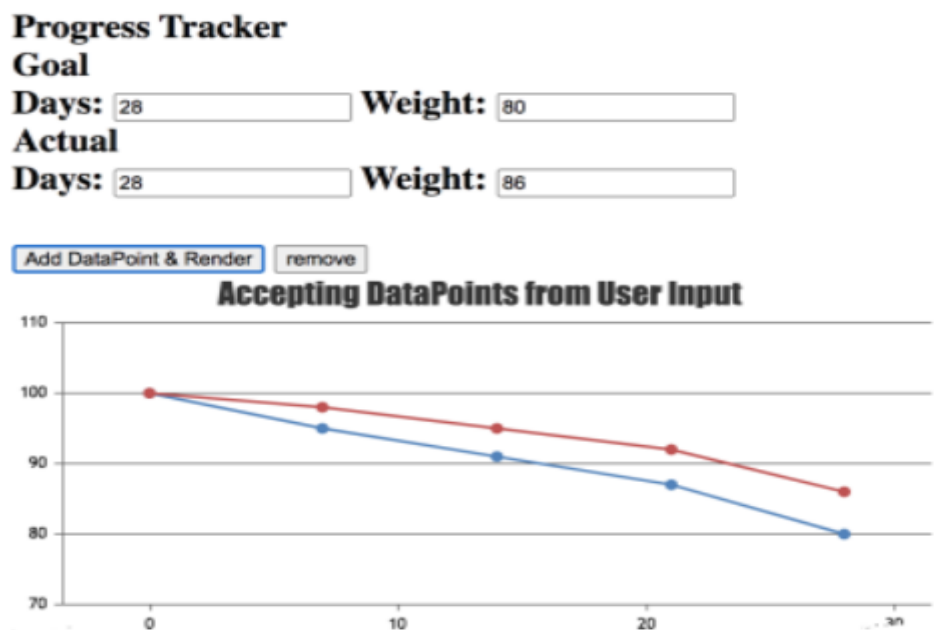


Figure 13 : Progress Tracker

Another important feature included in the Framework is a system that allows for personalised meal planning to be carried out. The user will have the capacity to adequately plan their diet as a result of this. The framework offers a mechanism that may be used to calculate the necessary number of calories for an individual based on their weight and height. It is useful to produce diet plans that are fully adjustable for the user, making it simpler for them to keep track of the calories they consume and their baseline metabolic rate.

Macros





ITEM	CALORIES	PROTEIN	CARB	FAT	QUANTITY
 Orange	51	0	13	0	<input type="text" value="2"/> <button>REMOVE</button>
 Banana	95	1	20	0	<input type="text" value="2"/> <button>REMOVE</button>
 Paneer	265	18	2	20	<input type="text" value="1"/> <button>REMOVE</button>
 NonFat Yoghurt	80	8	12	0	<input type="text" value="1"/> <button>REMOVE</button>
TOTAL	637	28gm	80gm	20gm	

Figure 14 : Web Portal for Diet Plan

Calories Calculator

Age

Height (in cm.)

Weight (in Kg.)

Gender

Male

Calculate

Basal Metabolic Rate (BMR) = 2005.0

Figure 15 : Calories Calculator Portal

Following the completion of the measurement, a user is then able to plan their meal with the assistance of an Internet of Things (IoT)-based smart weighing system, which enables us to conveniently monitor the nutritional statistics linked with the food item that is being measured. The first thing that is measured with the assistance of the weighing scale is the item's weight. The weight is then saved in the firebase after it is finished. After carrying out a number of computations, we are able to deduce the nutritional statistics pertaining to that

item based on the data that is saved in Firebase, which can be quickly accessible using a JavaScript application.

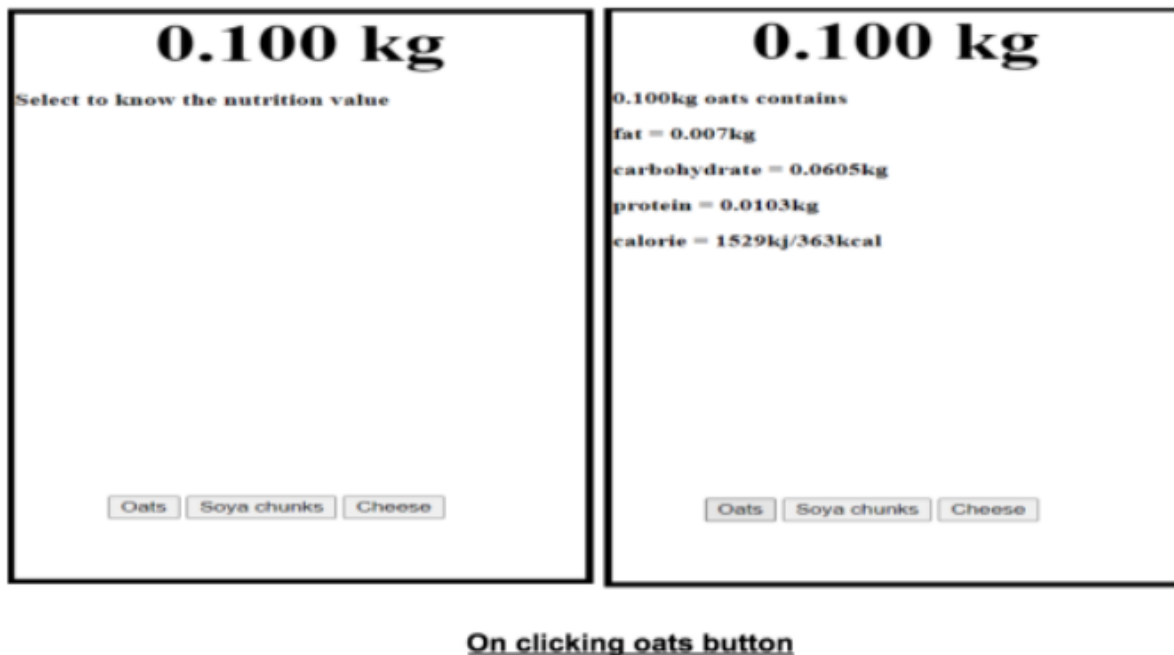


Figure 16 : Web Portal for Weighing Scale

3.4 Website Hosting:

A web hosting service is a kind of Internet hosting service that enables users to host their own websites on the servers of the firm. This sort of service provides consumers with the tools they need to construct and manage a website, as well as make the website available on the World Wide Web. Web hosts are companies that provide web hosting services to its clients. Web hosting may be performed by connecting our domain name with a flask web application. This web program is placed on a virtual private server, and the connection is made by setting Putty to operate with lamp installations. (Linux = Linux, Apache = Apache, MySQL = MySQL, and PHP = PHP).

There are Various Web Hosting Platforms where we can deploy our Framework like Firebase Hosting and Digital Ocean Platform. These platforms provide us with an I.P address that can be easily mapped with our domain name with the help of the Domain Name System (DNS). GoDaddy.com is the most famous web hosting company where we can get Domain Name and can even Host our Website.

3.4.1 Domain Name System:

The domain name system or DNS is like the phone book of the internet. It maps human readable urls or host names like `www.example.io` to the ip address of the server that hosts that site. When you type a url into the browser it makes a DNS query to figure out which unique ip address is associated with that hostname. First it will attempt to look in the local browser or operating system cache but if the cache is empty then we need to look up the ip address in the phone book which is the job of a server known as the DNS recursive resolver, it is recursive because it needs to make multiple requests to other servers starting with the root name server which itself will respond with the address of a top level domain DNS server which stores data about top level domains like `.com` or `.io`

The resolver makes a request there which will respond with the ip address of the authoritative name server that is the final source of truth that contains the requested website's ip address that gets sent back down to the client and is cached for future use that sounds complicated but it all happens within a fraction of a second you can think of it as a globally distributed database that handles more requests than any other database in the world.

When you register a domain name it is handled by a registrar accredited by the non-profit organisation ICANN, the record is then maintained by a registry operator who stores your DNS settings and propagates them to other DNS servers around the world as the registrant you will be assigned a zone file where you can configure the settings of your domain. Every zone has a startup authority record to tell other DNS servers who is in charge of that domain but most importantly we have a record it means address and maps a domain or subdomain to the ip address of its host. In other cases though we may want to forward a domain to another domain on the internet instead of an IP address in which case a canonical name or CName record would be used we have mx records for email and we can store arbitrary data with text records which is often used to communicate with third parties to verify ownership of the domain and finally we have the ns or name server record a url that the internet can ping to find our ip address and there's almost always two of them for better reliability.

3.4.2 Linux :

Linux is an open source operating system that makes it possible for humans to play with computers. It was created in 1991 by university of helsinki student Linus Torvalds, his goal was to create a free version of the MINIX operating system which itself was based on unix. Today it is used everywhere like the vast majority of web servers embedded apps like your smart tv, mobile devices, running android, and is by far the best choice for your personal computer. With many different distros to choose from like debian arch and fedora just to name a few just like windows or mac os linux is an operating system that manages a computer's memory and processes allowing hardware to communicate with software.

When you fire up a linux machine a bootloader will first put the system into memory at its core. We have the kernel which is like a seed from which the rest of the operating system grows. It is divided into multiple smaller subsystems like a process scheduler device drivers and a memory manager; they are exposed with the system call interface which itself is wrapped with the C standard library which provides an API that user applications can interact with. Now beyond the kernel we have applications that make the OS usable for humans which are provided primarily via the GNU project like a shell to interact with the kernel from the command line windowing systems on desktop developer utilities and countless other applications.

To get started install your favourite linux distro open the terminal and logged in as the root user on a specific linux machine, the home directory of the root user which is shortened to this tilde character show its full name with the print working directory command now use change directories with a double period to go to the root of the file system use the ls command to print all the subdirectories.

3.4.3 Apache:

The Apache Server is a cross-platform web server that is available as an free and open software. The Apache Software Foundation oversees the development and maintenance of Apache, which is done by an open community of developers.

Although the vast majority of Apache HTTP Server instances operate on Linux, recent versions also work on Microsoft Windows, OpenVMS, and a variety of Unix-like systems. Previous versions supported NetWare, OS/2, and other operating systems, as well as mainframe ports.

3.4.4 MYSql:

MySQL is a relational database management system (RDBMS) that anyone may use for free. Its name is a combination of "My," the name of one of its co-founders, Michael Widenius's daughter, and "SQL," an acronym for Structured Query Language. A relational database is used to arrange the data into one or more data tables, which enables the data to be related with one another and structured. SQL is a computer language that allows programmers to create, change, and retrieve data from relational databases while also controlling user access. SQL also allows programmers to restrict who has access to the databases. A relational database management system (RDBMS) like MySQL, in addition to relational databases and SQL, handles users, permits network access, and streamlines the process of evaluating database integrity and producing backups. It also works with an operating system to implement a relational database in a computer's storage system.

3.4.5 PhP:

PHP is a dynamic and interpreted programming language. It is used on the server to create interactive websites. It is still one of the most used languages for back-end web development. It powers content management systems like WordPress, top-tier websites like Wikipedia, and numerous more through frameworks like Symfony and Laravel. Even Facebook uses it, despite the fact that they designed its own custom compiler to convert it to machine code on their servers.

It was created in 1994 by Rasmus Lerdorf to manage his personal home page. Today we practise cognitive dissonance to tell ourselves PhP stands for hypertext preprocessor, it has a special place in history because it predates javascript and was one of the first languages to be embedded directly in HTML allowing websites to be built dynamically on a server. It is open source and was one of the technologies that revolutionised the web by making application development accessible to the average person. Its interpreter is implemented in C and has a

syntax inspired by Perl. To get started create a PHP file and add some basic html to it within this file you can go into php mode by opening a question mark php tag the code within these tags will be rendered on the server. Use echo to output a value and the location of the tag to declare a variable, give it a name that starts with a dollar sign followed by a value.

It is a weakly typed language so no type annotations are necessary, it also comes with a large number of predefined variables geared towards web development like `$_GET`, `$_POST` and `$_COOKIE` which contain information about the incoming http request now when this code is rendered on your server it will output an HTML string with the php tags being replaced by whatever you echoed. The language also supports multiple programming paradigms: you have first class functions which can be assigned to variables or used anonymously as arguments and it has tons and tons of built-in functions to do pretty much anything a web developer could ever imagine doing.

3.5 IoT-based Weighing Scale:

Internet of Things (IoT)-based technologies are starting to have a significant impact on the health of many individuals, especially in the many aspects of healthcare. Using the Internet of Things (IoT) in this field provides a wealth of potential for monitoring, study and treatment of disorders such as obesity. RPM (Remote Patient Monitoring) has been made possible by wearable gadgets and cellphones, which allow for a better awareness of a patient's state.

This is a very important piece of equipment that can determine the weight of a particular food item. When we set an item on a load cell, the load cell transmits the signal or weight to the node MCU. After that, the data is saved in Firebase, which is a Real Time Online Database. The data from Firebase can be simply recovered with the assistance of a Javascript program.

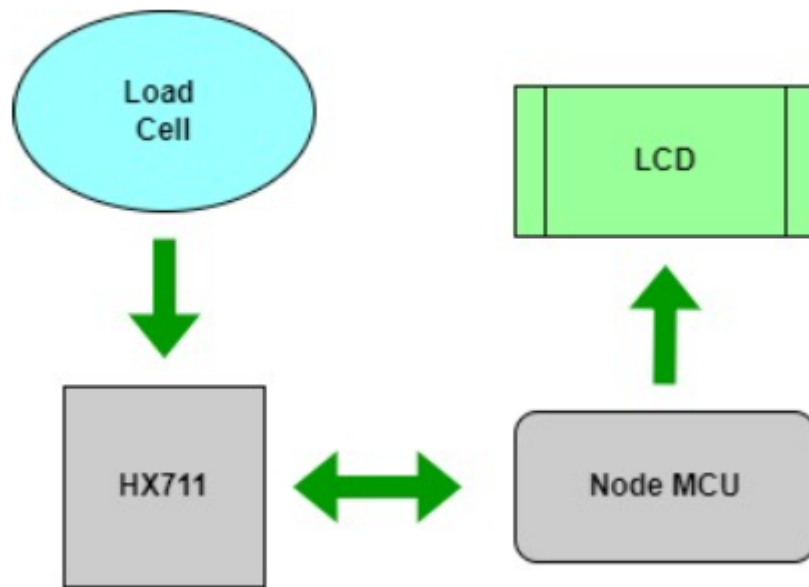


Figure 17 : Flowchart of IoT Model

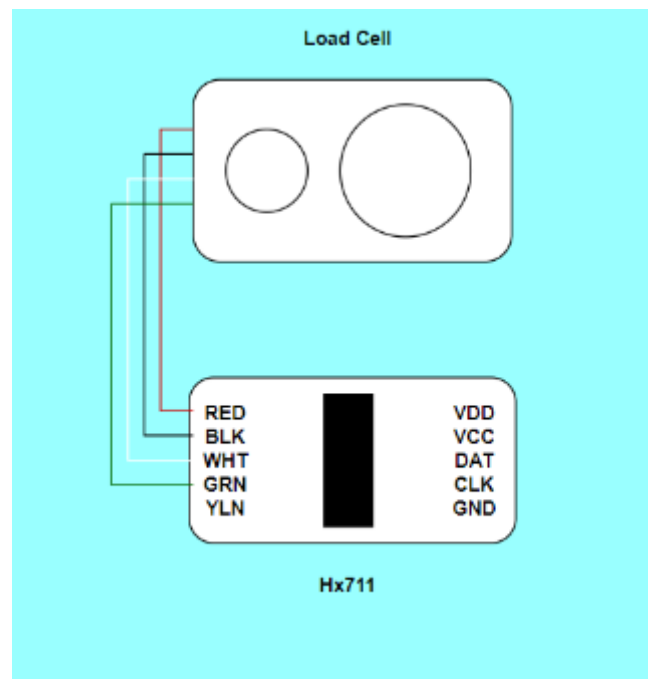


Figure 18 : Connection between load cell and hx711

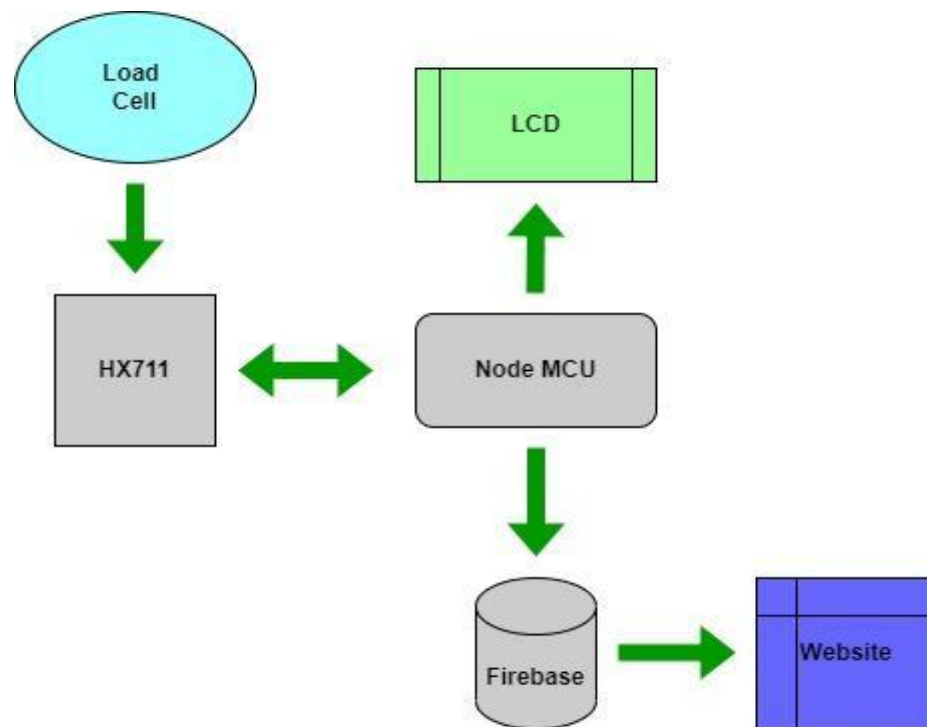


Figure 19 : Proposed Flow of Events

To get started, we have to put something on the load cell so that it can weigh what we have. The load cell sends quantifiable electric impulses to the Hx711 sensor, which in turn receives those signals. A measurement of the weight is taken, and the results are relayed to the Node MCU. After that, we have to find a way to store the data, and one option is to use Firebase, so Node MCU will transmit the data there. The further computations that need to be done may be done with the help of this information. Using a program that is based on JavaScript, it is simple to get access to the data that is saved in Firebase and put it to use in order to get the necessary results. One may quickly measure the weight of an item, look up the nutritional values of that item, and then arrange their diet in accordance with the information they have obtained.

The two pins, the DAT pin and the CLK pin are used for transferring the data and connected to D1 and D2 pins.

D1 and D2 are devoted to extracting values from the CLK and DAT pins, respectively. Gnd pins are used to close electrical circuits and maintain a common reference level throughout the circuit.

3.5.1 Hx711 Load cell Amplifier:

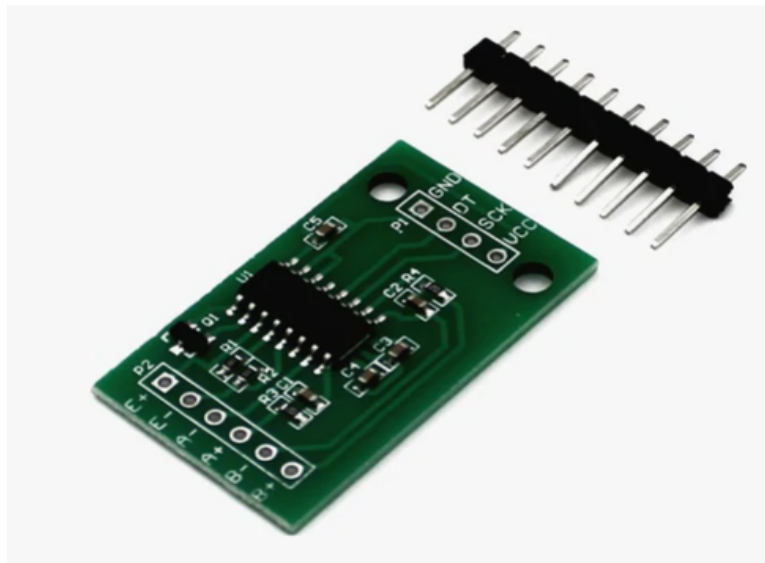


Figure 21: Hx711 Load Cell Amplifier

The Load Cell Amplifier is a small breakout board for the HX711 IC that enables you to easily read load cells for weight measurement.

3.5.2 Node MCU

NodeMCU (Node MicrocontrollerUnit) is an open-source development board used to prototype IoT projects. It has built-in WiFi connectivity support. It is also very cost-effective and uses less power. It enables us to use it as a microcontroller, allowing us to manipulate input and output.



Figure 22: Node MCU

3.5.3 LCD Display



Figure 23 : LCD Display

LCD (Liquid Crystal Display) displays are commonly used for multi-segment light-emitting diodes and seven segments.

Chapter 4: Performance Analysis

One of five algorithms used by us in this study to make three different kinds of predictions was Random Forest. Other algorithms such as K nearest Neighbour, Decision trees, XGBoost, and Extra Tree were some of the other algorithms. Random Search (RS), Grid Search (GS), Genetic Algorithm (GA), and Optuna framework (OPT) were used to make the optimization models even better. To do random searches, we used the RandomSearchCV and GridSearchCV methods from the scikit package, which are both in the scikit library. A computer programme called the Tree-based Pipeline Optimization Tool (TPOT) [29] was used to make the evolutionary method possible.

The TPOT, an automated machine learning tool, was made with the help of a genetic search method. It was the trial object that set the hyper-parameters, and the research object did the final execution as part of a project to make Optuna more accurate.

Table II shows how well the categorization of obesity levels was done with the test data. The random forest model with grid search and the Tpot classifier was the best at being accurate in the experiment. It scored 86 percent. If we look at all the models, decision trees, which have an accuracy of 73% to 75%, are the worst. The accuracy of the classifier did not get better when more trees were added to it; it stayed at 85% accuracy. If we want to know which of the many HPO models isn't in competition with each other, we can just look at random search.

Table 1: PREDICTED OBESITY LEVEL ACCURACY

Model	Random Search	Grid Search	Tpot	Optuna
Random Forest	0.84	0.86	0.86	0.85
Extra Trees	0.85	0.85	0.85	0.85
XGBoost	0.84	0.84	0.84	0.85

KNN	0.81	0.82	0.82	0.82
Decision Tree	0.73	0.76	0.74	0.73

The body weight forecasts in Table III are broken down into three metrics: the root mean square error (RMSE), the mean absolute error (MAE), and the mean absolute percent error (MAPE). Several studies have indicated that increasing or decreasing the RMSE has a negative impact on MAE and MAPE in certain circumstances. The root mean squared error (RMSE) and mean absolute error (MAE) values, for example, are 3.50 and 2.23 in the decision tree approach using Tpot. Instead, the RMSE increased to 3.88, while the MAE decreased to 2.13, indicating that the model during training concentrated on correcting smaller values rather than mending major defects, which were further penalised by the RMSE. Because of its low MAE and MAPE values, Optuna's Extra Tree Regressor augmented by Optuna was selected for use in the framework application's regression model.

Table IV demonstrates that the XGBoost method outperforms all other algorithms when it comes to forecasting body fat percentage in small datasets. Optuna outperforms the competition when it comes to the execution of the HPO model. Because it produces the fewest errors across all measures for our body fat % model, we chose XGBoost with Optuna as our preferred solution. Comparing KNN to other algorithms, it turned out to be less effective than predicted.

Table 2: PREDICTED BODY WEIGHT METRICS

Model	Metric	Random Search	Grid Search	Tpot	Optuna
Random Forest	MAPE	0.02	0.02	0.02	0.02
	MAE	1.65	1.64	1.70	1.67
	RMSE	2.81	2.80	2.86	2.85
Extra Trees	MAPE	0.02	0.02	0.02	0.02
	MAE	1.62	1.62	1.62	1.58
	RMSE	2.90	2.86	2.90	2.82
XGBoost	MAPE	0.026	0.025	0.025	0.023
	MAE	1.97	1.84	1.87	1.80

	RMSE	3.00	2.96	2.95	2.87
KNN	MAPE	0.044	0.041	0.042	0.043
	MAE	3.16	2.98	3.13	3.10
	RMSE	5.38	5.30	5.02	5.45
Decision Tree	MAPE	0.026	0.030	0.029	0.027
	MAE	2.00	2.30	2.20	2.10
	RMSE	3.80	3.70	3.50	3.90

Grid Search is a time-consuming and labour-intensive operation since it goes through every possible combination of the search space offered. When dealing with huge search spaces, it is not advisable to teach grid search. While random search is more efficient in terms of learning, it does not necessarily provide the best results. While Tpot is the most efficient option when searching across a big region, training it on a large number of generations and populations might take a long time. Because of the speed and quality of Optuna, it is now the most used HPO model in use. It is, on the other hand, difficult to find the optimal settings for the desired function. Optuna, on the other hand, outperforms the other HPO models in terms of execution as well as outcomes. As a consequence, even in small trials, The Optuna has the potential to provide excellent outcomes. In addition, as seen in Figure 24, the model contains numerous charts for examining hyperparameters, such as the optimization history plot, which may be found in the model's documentation.

Table 3: PREDICTED BODY FAT % METRICS

Models	Metrics	RandomSearchCV	Grid SearchCV	Tpot	Optuna
Extra Trees	MAPE	0.23	0.23	0.23	0.22
	MAE	3.11	3.11	3.12	3.10
	RMSE	3.75	3.78	3.78	3.74
XGBoost	MAPE	0.21	0.23	0.22	0.20
	MAE	2.90	3.10	3.00	2.80
	RMSE	3.60	3.74	3.70	3.50
KNN	MAPE	0.28	0.28	0.26	0.28
	MAE	3.70	3.70	3.50	3.70
	RMSE	4.50	4.49	4.40	4.50
Random Forest	MAPE	0.24	0.23	0.23	0.23
	MAE	3.15	3.16	3.17	3.10
	RMSE	3.80	3.81	3.85	3.80

Decision Tree	MAPE	0.26	0.24	0.26	0.23
	MAE	3.50	3.35	3.60	3.05
	RMSE	4.50	4.20	4.55	3.90

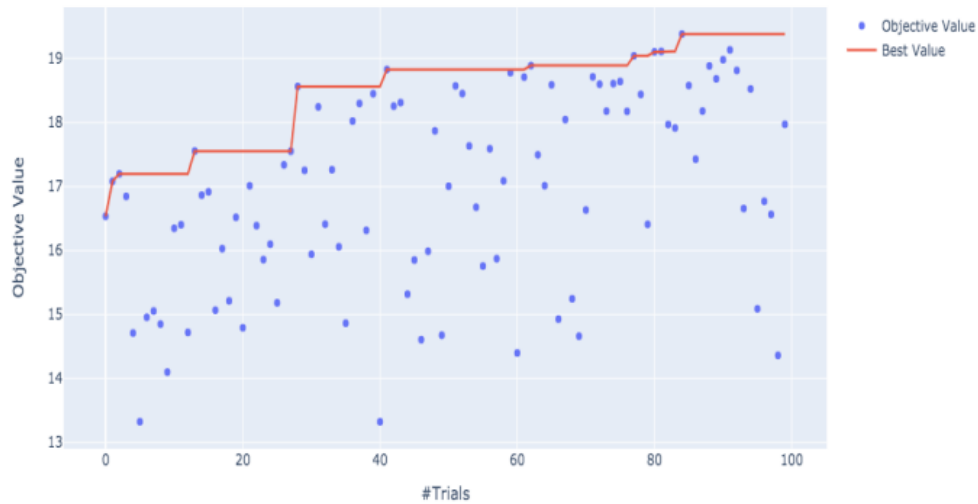


Figure 24 : Plot of Optimization History

Testing the models with data from the actual world is another crucial step in the process. This helps to validate the models and demonstrates how accurately they anticipate future events. We put the five machine learning algorithms through their paces by applying them to the first five rows of the test data of the body fat percentage dataset, which is provided in the sample input table in Figure 25. Table 4 displays the outcomes of these five different algorithms' applications.

	Age	Weight	Height	Neck	Chest	Abdomen	Hip	Thigh	Knee	Ankle	biceps	Forearm	Wrist
165	35	217.00	73.75	40.5	107.5	95.1	104.5	64.8	41.3	25.6	36.4	33.7	19.4
6	26	181.00	69.75	36.4	105.1	90.7	100.3	58.4	38.3	22.9	31.9	27.8	17.7
111	43	183.25	70.00	37.1	108.0	105.0	103.0	63.7	40.0	23.6	33.5	27.8	17.4
172	35	177.25	71.00	38.4	100.5	90.3	98.7	57.8	37.3	22.4	31.0	28.7	17.7
115	40	158.00	69.25	36.3	97.0	86.6	92.6	55.9	36.3	22.1	29.8	26.3	17.3

Figure 25: Test samples for testing the machine learning algorithms

Table 4: Differences in body fat percentages between real values and machine learning techniques.

(True Values)	Random Forest	Decision Tree	KNN	Extra Tree	XGBoost
19.10	20.15	21.10	20.52	19.17	16.16
19.20	17.44	9.30	14.64	16.59	14.20
27.55	29.15	27.55	25.19	29.35	30.15
20.35	18.36	18.0	19.66	17.95	17.75
16.70	17.72	22.45	13.67	17.10	17.73

Chapter 5: Conclusions

Overweight and obesity are both considered cancer by the general public, and they are both major risk factors for the development of diseases such as diabetes, circulatory stress, and heart disease. Three machine learning models were developed specifically for this inquiry. Calculate your obesity status, body weight, and body fat percentage using the chart below. Following that, the Python Flask framework was used to deploy the specified models. Another advantage is the ability to customise the website structure, as well as other elements like as a user foundation and diet regimes that can be adjusted. Additionally, an IoT-enabled weighing scale was used to define the meal, resulting in measurements of the calorie and macronutrient composition of the preprocessed food being taken. It is possible that in the future, technologies such as feature engineering will be employed to improve overall accuracy of models. The data for this study was gathered from the South American regions that were used for the research, which is a significant problem. In order to examine the inconsistencies, it is necessary to establish independent datasets from various geographies and demographic regions. Projects that make use of computer vision may benefit from applications that are developed specifically for that purpose.

REFERENCES

- [1] GBD 2017 Risk Factor Collaborators. (2018). Global, regional, and national comparative risk assessment of 84 behavioural, environmental and occupational, and metabolic risks or clusters of risks for 195 countries and territories, 1990–2017: a systematic analysis for the Global Burden of Disease Study 2017. *Lancet* (London, England), 392(10159), 1923.
- [2] Obesity, [https://www.who.int/health-topics/obesity#tab=tab 1](https://www.who.int/health-topics/obesity#tab=tab1)
- [3] Djalalinia, Shirin et al. "Health impacts of Obesity." *Pakistan journal of medical sciences* vol. 31,1 (2015): 239-42. doi:10.12669/pjms.311.7033
- [4] Ebbert, Jon O., Muhamad Y. Elrashidi, and Michael D. Jensen. "Managing overweight and obesity in adults to reduce cardiovascular disease risk." *Current atherosclerosis reports* 16, no. 10 (2014): 445.
- [5] Genena, D., and Amany A. Salama. "Obesity and eating habits among university students in Alexandria, Egypt: a cross sectional study." *World journal of nutrition and health* 5, no. 3 (2017): 62-68.
- [6] Grier, Tyson et al. "Does body mass index misclassify physically active young men." *Preventive medicine reports* vol. 2 483-7. 11 Jun. 2015, doi:10.1016/j.pmedr.2015.06.003
- [7] Shilpa, Joshi, and Viswanathan Mohan. "Ketogenic diets: Boon or bane?." *The Indian journal of medical research* 148.3 (2018): 251.
- [8] Qi, Jun, et al. "Examining sensor-based physical activity recognition and monitoring for healthcare using Internet of Things: A systematic review." *Journal of biomedical informatics* 87 (2018): 138-153.
- [9] Yunus, Raza, et al. "A framework to estimate the nutritional value of food in real time using deep learning techniques." *IEEE Access* 7 (2018): 2643-2652.

- [10] Kocabey, Enes, et al. "Face-to-BMI: Using computer vision to infer body mass index on social media." *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 11. No. 1. 2017.
- [11] Learning, Deep. "Towards a Digital Personal Trainer for Health Clubs."
- [12] Gupta, Mehak, et al. "Obesity Prediction with EHR Data: A deep learning approach with interpretable elements." *arXiv preprint arXiv:1912.02655* (2019).
- [13] Palechor, Fabio Mendoza, and Alexis de la Hoz Manotas. "Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from Colombia, Peru and Mexico." *Data in brief* 25 (2019): 104344.
- [14] Johnson, Roger W. "Fitting percentage of body fat to simple body measurements." *Journal of Statistics Education* 4.1 (1996).
- [15] Cat plot <https://seaborn.pydata.org/generated/seaborn.catplot.html>
- [16] Feng, Shuo, Huiyu Zhou, and Hongbiao Dong. "Using deep neural network with small dataset to predict material defects." *Materials & Design* 162 (2019): 300-310.
- [17] Song, Yan-Yan, and L. U. Ying. "Decision tree methods: applications for classification and prediction." *Shanghai archives of psychiatry* 27.2 (2015): 130.
- [18] Breiman, Leo. "Random forests." *Machine learning* 45.1 (2001): 5-32.
- [19] Wang, Lei, Latifur Khan, and Bhavani Thuraisingham. "An effective evidence theory based on k-nearest neighbour (knn) classification." *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*. Vol. 1. IEEE, 2008.
- [20] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016.
- [21] Geurts, Pierre, Damien Ernst, and Louis Wehenkel. "Extremely randomized trees." *Machine learning* 63.1 (2006): 3-42.
- [22] Feurer, Matthias, and Frank Hutter. "Hyperparameter optimization." *Automated Machine Learning*. Springer, Cham, 2019. 3-33.
- [23] Elsken, Thomas, Jan Hendrik Metzen, and Frank Hutter. "Neural architecture search: A survey." *J. Mach. Learn. Res.* 20.55 (2019): 1-21.
- [24] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *the Journal of machine Learning research* 12 (2011): 2825-2830.
- [25] Bergstra, James, and Yoshua Bengio. "Random search for hyperparameter optimization." *Journal of machine learning research* 13.2 (2012).

- [26] Shekar, B. H., and Guesh Dagneu. "Grid search-based hyperparameter tuning and classification of microarray cancer data." 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP). IEEE, 2019.
- [27] Goldberg, David E., and John Henry Holland. "Genetic algorithms and machine learning." (1988).
- [28] Akiba, Takuya, et al. "Optuna: A next-generation hyperparameter optimization framework." Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019.
- [29] Olson, Randal S., and Jason H. Moore. "TPOT: A tree-based pipeline optimization tool for automating machine learning." Workshop on automatic machine learning. PMLR, 2016.
- [30] Grinberg, Miguel. Flask web development: developing web applications with python. "O'Reilly Media, Inc.", 2018.
- [31] Canvas JS, <https://canvasjs.com/>
- [32] Al-Mutlaq, Sarah. "Getting started with load cells." Conteudo Online ' (2016).
- [33] Bento, Antonio Carlos. "IoT: NodeMCU 12e X Arduino Uno, Results of an experimental and comparative survey." International Journal 6.1 (2018).
- [34] Li, Wu-Jeng, et al. "JustIoT Internet of Things based on the Firebase real-time database." 2018 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE). IEEE, 2018.
- [35] J. N. S. Rub'i and P. R. D. L. Gondim, "Interoperable internet of medical things platform for e-health applications," International Journal of Distributed Sensor Networks, vol. 16, no. 1, 2020.
- [36] P. C. Santana-Mancilla, L. E. Anido-Rifon, J. Contreras- ' Castillo, and R. Buenrostro-Mariscal, "Heuristic evaluation of an IoMT system for remote health monitoring in senior care," International Journal of Environmental Research and Public Health, vol. 17, no. 5, Article ID 1586, 2020.
- [37] O. AlShorman, B. AlShorman, M. Al-khassaweneh, and F. Alkahtani, "A review of internet of medical things (IoMT) - based remote health monitoring through wearable sensors: a case study for diabetic patients," Indonesian Journal of Electrical Engineering and Computer Science, vol. 20, no. 1, pp. 414–422, 2020.
- [38] I. Machorro-Cano, U. Ramos-Deonati, G. Alor-Hernandez ' et al., "An IoT-based architecture to develop a healthcare smart platform," in Proceedings of the International Conference on Technologies and Innovation, pp. 133–145, Springer, Cham, Switzerland, October 2017

- [39] M. Alloghani, A. Hussain, D. Al-Jumeily, P. Fergus, O. Abuelma'Atti, and H. Hamden, "A mobile health monitoring application for obesity management and control using the internet-of-things," in *Proceedings of the 2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC)*, pp. 19–24, IEEE, Beirut, Lebanon, April 2016.
- [40] A. Alnuaimi, S. Rawaf, S. Hassounah, and M. Chehab, "Use of mobile applications in the management of overweight and obesity in primary and secondary care," *JRSM open*, vol. 10, no. 3, 2019.
- [41] S. Garg and P. Pundir, "MOFit: a framework to reduce obesity using machine learning and IoT," 2021, <https://arxiv.org/abs/2108.08868>.
- [42] J. T. de Souza, G. A. L. de Campos, C. Rocha et al., "An agent program in an IoT system to recommend activities to minimize childhood obesity problems," in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pp. 654–661, Brno, Czech Republic, March 2020.
- [43] D. Gupta, A. Parikh, and R. Swarnalatha, "Integrated healthcare monitoring device for obese adults using internet of things (IoT)," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 2, 2020.
- [44] R. Alturki and V. Gay, "The development of an Arabic weightloss app Akser Waznk: qualitative results," *JMIR formative research*, vol. 3, no. 1, Article ID e11785, 2019.
- [45] J. Mae, E. Oey, and F. S. Kristiady, "IoT based body weight tracking system for obese adults in Indonesia using realtime database," *Earth and Environmental Science*, vol. 426, no. No. 1, Article ID 012143, 2020.
- [46] U. B. Mahadevaswamy, "Wireless wearable smart healthcare monitoring using android," *International Journal of Computer Network and Information Security*, vol. 12, no. 2, p. 12, 2018.
- [47] J. B. Awotunde, R. G. Jimoh, S. O. Folorunso, E. A. Adeniyi, K. M. Abiodun, and O. O. Banjo, "Privacy and security concerns in IoT-based healthcare systems," in *The Fusion of Internet of Things, Artificial Intelligence, and Cloud Computing in Health Care*, pp. 105–134, Springer, Cham, Switzerland, 2021.
- [48] S. Nazir, Y. Ali, N. Ullah, and I. García-Magariño, "Internet of things for healthcare using effects of mobile computing: a systematic literature review," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 5931315, 20 pages, 2019.
- [49] S. M. Sefa-Yeboah, K. Osei Annor, V. J. Koomson, F. K. Saalia, M. SteinerAsiedu, and G. A. Mills, "Development of a mobile application platform for SelfManagement of obesity

using artificial intelligence techniques,” *International Journal of Telemedicine and Applications*, vol. 2021, Article ID 6624057, 16 pages, 2021.

[50] R. Alturki, M. J. AlGhamdi, V. Gay, N. Awan, M. Kundi, and M. Alshehri, “Analysis of an eHealth app: privacy, security and usability,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 4, pp. 209–214, 2020.

[51] I. Machorro-Cano, G. Alor-Hernández, M. A. ParedesValverde, U. Ramos-Deonati, J. L. Sanchez-Cervantes, and L. Rodríguez-Mazahua, “PISIoT: a machine learning and IoTbased smart health platform for overweight and obesity control,” *Applied Sciences*, vol. 9, no. 15, Article ID 3037, 2019

[52] A. I. Alharbi, V. Gay, M. J. AlGhamdi, R. Alturki, and H. J. Alyamani, “Towards an application helping to minimize medication error rate,” *Mobile Information Systems*, vol. 2021, Article ID 9221005, 7 pages, 2021.

[53] H. Ahmadi, G. Arji, L. Shahmoradi, R. Safdari, M. Nilashi, and M. Alizadeh, “%e application of internet of things in healthcare: a systematic literature review and classification,” *Universal Access in the Information Society*, vol. 18, no. 4, pp. 837–869, 2019.

[54] S. L. Holzmann and C. Holzapfel, “A scientific overview of smartphone applications and electronic devices for weight management in adults,” *Journal of Personalized Medicine*, vol. 9, no. 2, p. 31, 2019.

[55] F. H. Alqahtani, “%e application of the Internet of %ings in healthcare,” *Int. J. Comput. Appl*, vol. 180, no. 18, pp. 19–23, 2018.

[56] R. Alturki, “Research onion for smart IoT-enabled mobile applications,” *Scientific Programming*, vol. 2021, Article ID 4270998, 9 pages, 2021.

[57] A. A. Qaffas, R. Hoque, and N. Almazmomi, “%e internet of things and big data analytics for chronic disease monitoring in Saudi Arabia,” *Telemedicine and e-Health*, vol. 27, no. 1, pp. 74–81, 2021.

[58] A. Chattopadhyay, S. Mishra, and A. Gonzalez-Briones, “Integration of machine learning and IoT in healthcare domain,” in *Hybrid Artificial Intelligence and IoT in Healthcare*, pp. 223–244, Springer, Singapore, 2021.

[59] M. Ibrahim, M. Wedyan, R. Alturki, M. A. Khan, and A. AlJumaily, “Augmentation in healthcare: augmented biosignal using deep learning and tensor representation,” *Journal of Healthcare Engineering*, vol. 2021, Article ID 6624764, 9 pages, 2021.

[60] L. Jane, G. Atkinson, V. Jaime, S. Hamilton, G. Waller, and S. Harrison, “Intermittent fasting interventions for the treatment of overweight and obesity in adults aged 18 years and

over: a systematic review protocol,” JBI Database of Systematic Reviews and Implementation Reports, vol. 13, no. 10, pp. 60–68, 2015.

[61] R. Alturki, M. J. AlGhamdi, V. Gay, N. Awan, and M. Alshehri, “Privacy, security and usability for IoT-enabled weight loss apps,” *International Journal of Advanced Computer Science and Applications*, vol. 11, 2020.

[62] S. M. Riazul Islam, D. Daehan Kwak, M. Humaun Kabir, M. Hossain, and K. S. Kyung-Sup Kwak, “The internet of things for health care: a comprehensive survey,” *IEEE access*, vol. 3, pp. 678–708, 2015.

[63] L. Farhan, R. Kharel, O. Kaiwartya, M. Quiroz-Castellanos, A. Alissa, and M. Abdulsalam, “A concise review on Internet of Things (IoT)-problems, challenges and opportunities,” in *Proceedings of the 2018 11th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP)*, pp. 1–6, IEEE, Budapest, Hungary, July 2018.

[64] M. A. Khan, M. T. Quasim, F. Algarni, and A. Alharthi, “Internet of things: on the opportunities, applications and open challenges in Saudi arabia,” in *Proceedings of the 2019 International Conference on Advances in the Emerging Computing Technologies (AECT)*, pp. 1–5, IEEE, Munawwarah, Saudi Arabia, February 2020.

[65] “File:obesity country comparison - path.svg,” *Wikimedia Commons*. [Online]. Available: https://commons.wikimedia.org/wiki/File:Obesity_country_comparison_-_path.svg.

[Accessed: 11-May-2022]

APPENDICES

In this section, the implementation of this project is provided by explaining the code snippets attached. The github link of this project can be found at:
https://github.com/satvik-27199/Obesity_framework