www.ijarp.com ISSN 2456-9992 Page: 01-29

International Journal Advanced Research Publications

Review Article

Volume: 02
Issue: 06

# IOT ENABLED SMART BMI MONITORING SYSTEM

*Shreya Sunil Dongre

India.

Article Received: 24 April 2026, Article Revised: 14 May 2026, Published on: 04 June 2026

*Corresponding Author: Shreya Sunil Dongre

India.

DOI: https://doi-doi.org/101555/ijarp.6852

## INTRODUCTION

With the rapid increase in lifestyle-related health issues such as obesity and malnutrition, regular monitoring of Body Mass Index (BMI) has become essential for early health assessment. Conventional BMI measurement methods are mostly manual and lack features such as automated calculation, data storage, and remote monitoring. To overcome these limitations, the proposed project, "IoT-Enabled Smart BMI Monitoring System," using ESP32 aims to develop an intelligent and connected health monitoring solution. The system uses a load cell to measure weight and an ultrasonic sensor to measure height automatically. The ESP32 microcontroller processes these inputs, calculates BMI using the standard formula, and categorizes the user's health status. The measured data is

displayed locally and simultaneously uploaded to a cloud platform for remote monitoring and historical tracking. Additionally, a GSM module sends an automatic SMS notification to the registered mobile number, enhancing user awareness. By integrating embedded systems, sensor technology, and IoT communication, the proposed system provides a low-cost, portable, and smart healthcare solution suitable for hospitals, schools, fitness centres, and rural health camps.

It includes following features which make the proposed system reliable, intelligent, and suitable for real-world Healthcare applications:

### 1. Automatic Height Measurement

The system uses an ultrasonic sensor (HC-SR04) to measure the user's height automatically. The sensor calculates distance using the time-of-flight principle and provides accurate height measurement without manual intervention.

**2. Digital Weight Measurement**

A load cell with HX711 amplifier module is used to measure body weight digitally. The system converts mechanical force into electrical signals and processes them through ESP32 for precise weight calculation.

**3. Real-Time BMI Calculation**

The ESP32 microcontroller calculates BMI instantly using the standard formula:

$$BMI = \frac{Weight(kg)}{Height\ (m2)}$$

The system processes data in real time and ensures fast computation.

**4. Automatic BMI Classification based on predefined threshold values, the system categorizes users into:**

- Underweight
- Normal
- Overweight
- Obese

This helps in quick health assessment.

**LITERATURE SURVEY**

**Table No. 1 – Literature Review.**

| Sr. No. | Paper / Authors (Year) | Main focus | Tech / sensors used | Key contribution (short) |
|---|---|---|---|---|
| 1 | *IoT-Based Smart Health Monitoring System (IEEE, 2024)* | Real-time remote monitoring of health parameters. | ESP32/ESP8266, temperature sensor, pulse sensor, cloud platform. | Demonstrates IoT-based architecture for transmitting patient health data to cloud for remote supervision and alert generation. |
| 2 | *Design and Implementation of Digital BMI Measurement System (IJERT, 2023)* | Automated BMI calculation using digital sensors. | Load cell, ultrasonic sensor, microcontroller. | Proposes automatic height and weight measurement system for accurate BMI calculation, reducing manual errors. |
| 3 | *Cloud-Integrated Smart Healthcare Monitoring System (ScienceDirect, 2024)* | Integration of IoT devices with cloud database for healthcare tracking. | IoT sensors, Wi-Fi module, cloud dashboard, database server. | Presents real-time cloud data storage and visualization for continuous health monitoring and historical record analysis. |

| | | | | |
|---|---|---|---|---|
| 4 | *GSM-Based Health Alert System Using Embedded Systems (IJSRD, 2023)* | SMS alert mechanism for abnormal health conditions. | GSM module (SIM800/SIM900), microcontroller, LCD. | Shows implementation of automatic SMS notification system for immediate user awareness during abnormal readings. |
| 5 | *Smart IoT-Based Body Parameter Monitoring System (ResearchGate, 2024)* | Multi-parameter body monitoring with IoT connectivity. | Multi-parameter body monitoring with IoT connectivity. | Focuses on secure data storage, regulatory compliance, and centralized monitoring for pharmaceutical cold chain logistics. (SpringerLink) |

The literature review indicates that recent advancements in IoT and embedded systems have significantly improved healthcare monitoring technologies by enabling automation, real-time data processing, cloud connectivity, and remote patient supervision. Various research papers focused on developing smart healthcare systems using ESP32/ESP8266 microcontrollers, load cells, ultrasonic sensors, GSM modules, cloud platforms, and IoT communication technologies. Studies on digital BMI measurement systems demonstrated that automatic height and weight measurement using sensors can reduce manual calculation errors and improve measurement accuracy. Research on cloud-integrated healthcare systems highlighted the importance of real-time data storage, online monitoring, historical health record analysis, and remote accessibility through IoT dashboards. GSM-based healthcare systems emphasized automatic SMS alert generation for improving user awareness and emergency communication during abnormal health conditions. Other studies focused on multi-parameter body monitoring systems capable of integrating multiple sensors for centralized healthcare monitoring and secure data management. Overall, the literature review shows that IoT technology plays a vital role in modern healthcare by providing accurate, low-cost, automated, portable, and user-friendly monitoring systems. Based on these research works, the proposed IoT Enabled Smart BMI Monitoring System is designed to automatically measure height and weight, calculate BMI in real time, upload data to cloud platforms, send SMS alerts, and maintain digital health records for efficient smart healthcare monitoring.

## CHAPTER 3 SYSTEM DESIGN

### 3.1 System overview

The **IoT Enabled Smart BMI Monitoring System** is an intelligent healthcare monitoring device designed to automatically measure a person's **height**, **weight**, and **Body Mass Index (BMI)** using sensors, embedded systems, and IoT technology. The system minimizes manual calculations and provides fast, accurate, and real-time health assessment. The project mainly focuses on automation, digital healthcare monitoring, cloud-based data storage, and remote accessibility. The complete system is built using the **ESP32 microcontroller**, which acts as the central processing and communication unit.

The system measures:

- Body Weight using a Load Cell
- Height using an Ultrasonic Sensor
- BMI using mathematical processing inside ESP32

The BMI value is calculated automatically using the standard formula:

$$BMI = \frac{Weight\ (kg)}{Height^2\ (m^2)}$$

After calculation, the system classifies the user into different health categories such as:

- Underweight
- Normal
- Overweight
- Obese

The calculated results are displayed on an LCD/OLED display and simultaneously uploaded to a cloud platform through Wi-Fi connectivity for remote monitoring and historical record maintenance.

Additionally, the system includes a GSM module that sends automatic SMS alerts containing the BMI value and health category to the registered mobile number. This improves health awareness and enables preventive healthcare monitoring.

## WORKING OF THE SYSTEM

The complete working of the system takes place in sequential steps:

### Step 1: System Initialization

When power is supplied, the ESP32 initializes all connected hardware modules such as:

- Load Cell
- HX711 Amplifier
- Ultrasonic Sensor
- LCD Display
- GSM Module
- Wi-Fi Connection

The sensors are calibrated to ensure accurate readings before operation begins.

**Step 2: User Registration**

The user enters basic information such as:

- Name
- Age
- Gender
- Mobile Number

This registration enables personalized health monitoring and cloud-based data management.

**Step 3: Weight Measurement**

The user stands on the measuring platform.

The **Load Cell** senses the applied body weight and converts mechanical force into a small electrical signal. Since the signal generated by the load cell is very weak, the **HX711 amplifier module** amplifies and converts it into digital form before sending it to ESP32. The ESP32 processes the digital data and calculates the user's body weight accurately.

**Step 4: Height Measurement**

The **Ultrasonic Sensor (HC-SR04)** measures the user's height automatically.

The sensor transmits ultrasonic waves toward the user and receives the reflected echo signal. Based on the time taken for the echo to return, the distance is calculated using the speed of sound principle. The system subtracts this measured distance from the predefined mounting height to determine the actual height of the user.

This method provides:

- Non-contact measurement
- Better hygiene
- Accurate height detection

**Step 5: BMI Calculation**

After obtaining weight and height data, the ESP32 calculates BMI automatically using the embedded algorithm.

The calculation is processed in real time, ensuring fast and accurate results.

The BMI value is then compared with standard health ranges to determine the user's health condition.

**BMI Classification table**

**Table No. 2 – BMI Classification.**

| BMI Range | Health Category |
|---|---|
| Below 18.5 | Underweight |
| 18.5 – 24.9 | Normal |
| 25 – 29.9 | Overweight |
| Above 30 | Obese |

**Step 6: Display of Results**

The measured parameters are displayed on the LCD/OLED display, including:

- Weight
- Height
- BMI Value
- Health Category

The display provides immediate feedback to the user.

**Step 7: IoT Cloud Upload**

Using the built-in Wi-Fi capability of ESP32, the measured data is uploaded to an IoT cloud platform such as:

- ThingSpeak
- Firebase
- Blynk

The uploaded data can be monitored remotely through:

- Mobile Application
- Web Dashboard
- Cloud Database

This enables:

- Remote health monitoring
- Historical data analysis
- Multi-user monitoring
- Digital record management

**Step 8: Automatic SMS Alert**

The GSM module sends an SMS notification to the registered mobile number containing:

- User Name
- BMI Value

- Health Category

Example:

"Your BMI is 27.4 — Category: Overweight"

This feature increases health awareness and provides immediate health updates even without internet connectivity.

### 3.2 Design overview: (Block diagram)

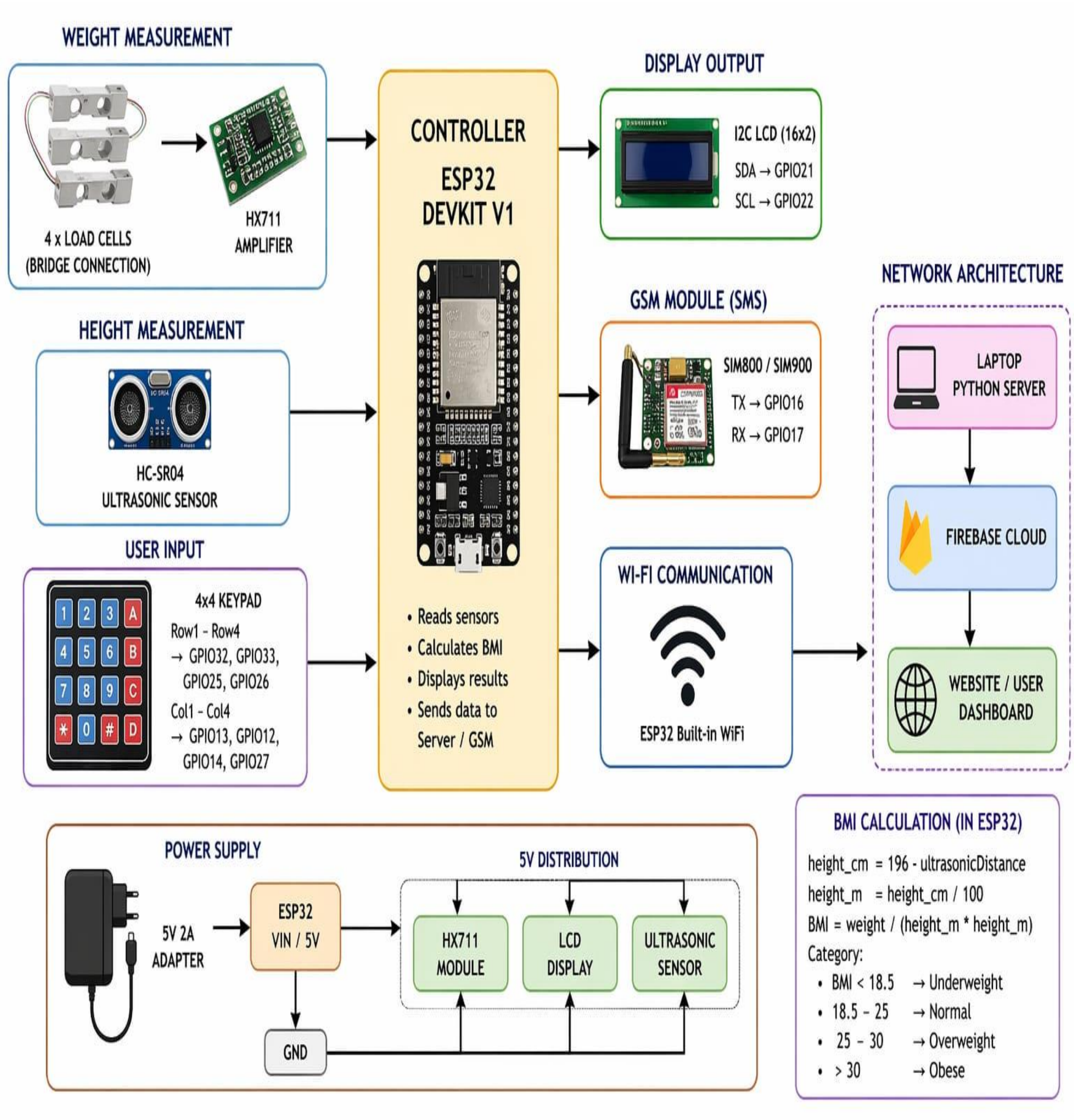



**Fig-3.1 Block Diagram of BMI System.**

**EXPLANATION:**

The above block diagram represents the complete architecture and working of the **IoT Enabled Smart BMI Monitoring System**. The ESP32 DevKit V1 acts as the main controller of the system and performs all processing, BMI calculation, display control, GSM communication, and cloud data transmission. For weight measurement, four load cells connected in bridge configuration are used along with the HX711 amplifier module. The load cells measure the user's body weight and send analog signals to the HX711 module, which amplifies and converts the signals into digital form before sending them to the ESP32. For height measurement, the HC-SR04 ultrasonic sensor is used to measure the distance between the sensor and the user's head, allowing the ESP32 to calculate the user's height automatically.

A 4x4 matrix keypad is connected to the ESP32 for user input such as entering details and controlling system operations. The calculated BMI value, along with height, weight, and health category, is displayed on the 16x2 I2C LCD display using SDA and SCL communication pins. The system also includes a GSM module (SIM800/SIM900) connected through TX and RX pins for sending SMS notifications containing BMI results and health status to the registered mobile number.

The ESP32 uses its built-in Wi-Fi capability for cloud communication and sends health data to the Firebase cloud server through a laptop-based Python server. The uploaded data can be accessed remotely through a website or user dashboard for continuous monitoring and digital healthcare record management. The entire system is powered using a 5V, 2A power supply adapter, which distributes regulated power to the ESP32, HX711 module, LCD display, and ultrasonic sensor. Inside the ESP32, BMI is calculated using the standard formula by converting measured height into meters and classifying the BMI value into categories such as underweight, normal, overweight, and obese.

### 3.3 Design approach

The design approach of the IoT Enabled Smart BMI Monitoring System is divided into major parts:

1. Software Design
2. Hardware Design

The system is designed using embedded systems, sensor technology, IoT communication, and automation techniques to achieve accurate and real-time BMI monitoring. The ESP32 microcontroller acts as the central controller of the system. It collects sensor data, processes the readings, calculates BMI, displays results, uploads data to cloud platforms, and sends SMS alerts to users.

The design approach focuses on:

- Accurate measurement
- Real-time processing
- IoT connectivity
- Automation
- User-friendly operation
- Low-cost implementation

**Software design**

The software section controls the complete functioning of the Smart BMI Monitoring System.

The software is developed using:

- Embedded C/C++
- Arduino IDE
- ESP32 Programming

The ESP32 receives input from:

- Load Cell
- HX711 Module
- Ultrasonic Sensor

The controller processes the data and performs BMI calculation automatically. The calculated BMI value is compared with predefined health ranges and categorized into:

- Underweight
- Normal
- Overweight
- Obese

The software also handles:

- LCD display operation
- IoT cloud communication
- GSM SMS alerts
- Data logging and storage

**Software Flowchart:**

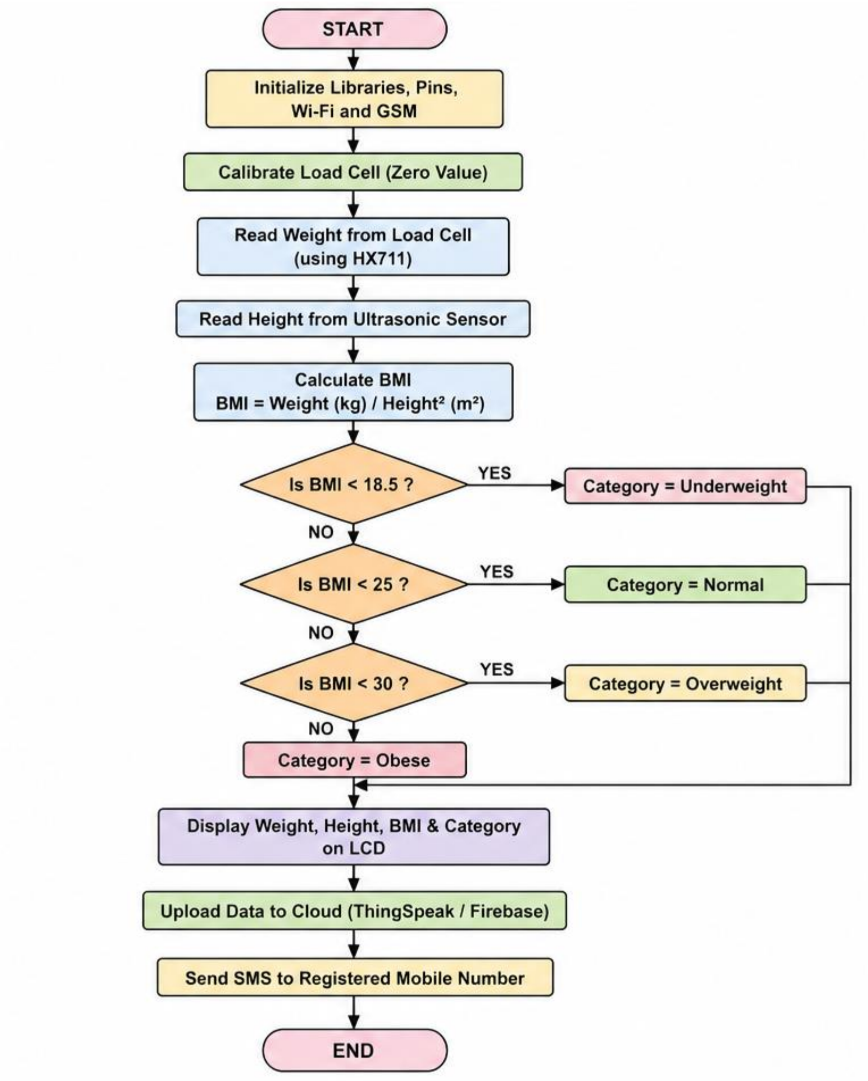



**Fig-3.2 Software Flowchart.**

## EXPLANATION

The sequential logic of the software is explained below:

### 1. Initialization

The system starts and initializes:

- ESP32 controller
- Load Cell
- HX711 module
- Ultrasonic sensor
- LCD display
- Wi-Fi connection
- GSM module

The software calibrates sensors to improve accuracy.

**2. User Registration**

The user enters details such as:

- Name
- Age
- Gender
- Mobile Number

The data is stored for personalized monitoring and cloud record management.

**3. Weight Measurement**

The load cell senses body weight and sends electrical signals to HX711.

The HX711 module:

- Amplifies signals
- Converts analog data to digital form
- Transfers data to ESP32

The software calculates the weight in kilograms.

**4. Height Measurement**

The ultrasonic sensor measures the distance between the sensor and the user’s head.

The software calculates height using distance measurement principles.

Distance formula:

$$Distance = \frac{Speed\ of\ Sound \times Time}{2}$$

**5. BMI Calculation**

The ESP32 calculates BMI automatically using:

$$BMI = \frac{Weight\ (kg)}{Height^2\ (m^2)}$$

The result is processed in real time.

### 6. BMI Classification

The software compares BMI with standard health ranges.

**Table No.3 – BMI Categories.**

| BMI Range | Category |
|---|---|
| Below 18.5 | Underweight |
| 18.5 – 24.9 | Normal |
| 25 – 29.9 | Overweight |
| Above 30 | Obese |

### 7. Display Operation

The LCD/OLED display shows:

- Weight
- Height
- BMI value
- Health category

### 8. Cloud Upload

The ESP32 uploads data to IoT cloud platforms using Wi-Fi connectivity.

Supported platforms:

- ThingSpeak
- Firebase
- Blynk

### 9. SMS Alert

The GSM module sends SMS notifications containing:

- BMI value
- Health category

This increases user awareness and enables preventive healthcare monitoring.

### Hardware design

The hardware design consists of sensing, processing, communication, and display modules integrated together to perform automatic BMI monitoring.

The hardware architecture is designed for:

- Accurate measurements
- Stable operation
- Low power consumption

- Real-time performance

**Hardware Flowchart:**

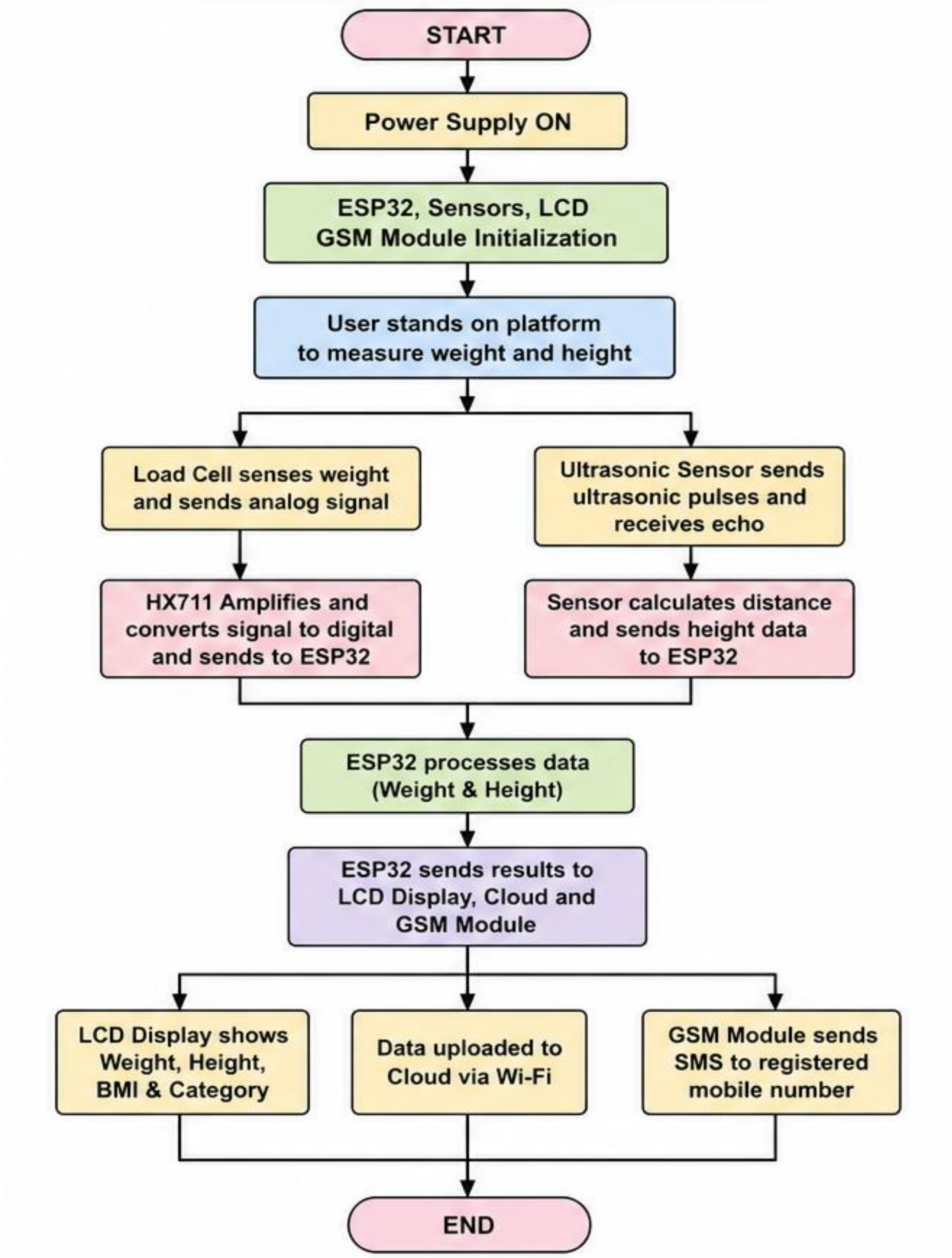



**Fig-3.3 Hardware Flowchart.**

**EXPLANATION:**

The above flowchart explains the complete working process of the **IoT Enabled Smart BMI Monitoring System**. Initially, the system starts when the power supply is switched ON. After power initialization, the ESP32 microcontroller initializes all connected components such as sensors, LCD display, and GSM module. Once initialization is completed, the user stands on the platform for height and weight measurement. The load cell senses the user's weight and generates an analog electrical signal, which is then amplified and converted into digital form by the HX711 amplifier module before being sent to the ESP32. Simultaneously, the ultrasonic sensor transmits ultrasonic waves and receives the reflected echo signal to calculate the user's height, which is also sent to the ESP32 controller.

After receiving both height and weight data, the ESP32 processes the information and calculates the Body Mass Index (BMI). The processed results are then transmitted to different output systems. The LCD display shows the user's weight, height, BMI value, and health category. At the same time, the measured data is uploaded to the cloud platform through Wi-Fi connectivity for remote monitoring and digital record maintenance. Additionally, the GSM module sends an SMS notification containing BMI details and health status to the registered mobile number. Finally, after completing all operations, the system reaches the END stage, completing one full BMI monitoring cycle.

**CHAPTER 4**

**HARDWARE AND SOFTWARE TOOLS**

**Hardware Tools:**

**4.1 HARDWARE TOOLS**

Hardware tools are the physical electronic components used to build and implement the Smart BMI Monitoring System.

**1. ESP32 MICROCONTROLLER**

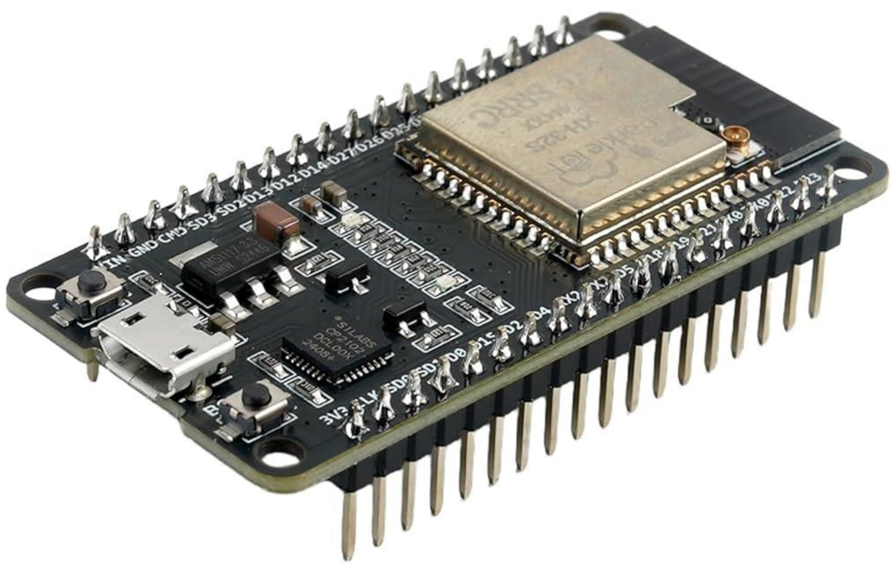


**Fig. 4.1 – ESP 32 Microcontroller.**

**Working**

The ESP32 acts as the central controller of the system. It receives sensor data, processes the readings, calculates BMI, controls display operations, uploads data to the cloud, and communicates with the GSM module.

The ESP32 contains:

- Dual-core processor
- Inbuilt Wi-Fi
- Bluetooth support
- Multiple GPIO pins

These features make it suitable for IoT healthcare applications.

**Uses**

- Main processing unit
- Sensor interfacing
- BMI calculation
- Wi-Fi communication
- Cloud data upload
- GSM communication

**2. LOAD CELL**

**Fig. 4.2 – 50 Kg Load Cell.**

**Working**

The load cell measures body weight using the strain gauge principle. When force is applied, the strain gauges deform slightly, causing a change in electrical resistance. This change produces a small electrical signal proportional to the applied weight.

**Uses**

- Measures user body weight
- Provides accurate weight readings
- Converts mechanical force into electrical signals

## 3. HX711 LOAD CELL AMPLIFIER MODULE

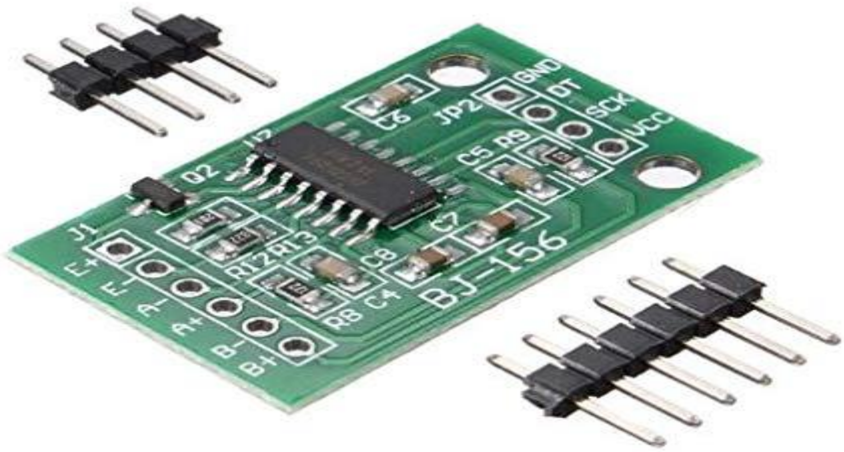



**Fig. 4.3 – HX711 Amplifier Module.**

**Working**

The HX711 module amplifies the weak analog signal generated by the load cell and converts it into digital form for ESP32 processing.

The module improves:

- Accuracy
- Signal stability
- Noise reduction

**Uses**

- Amplifies load cell signals
- Converts analog data to digital data
- Improves measurement precision

## 4. ULTRASONIC SENSOR (HC-SR04)


**Fig. 4.4 – Ultrasonic Sensor.**

### Working

The ultrasonic sensor measures height using ultrasonic wave reflection.

The sensor:

1. Sends ultrasonic pulses
2. Receives reflected echo signals
3. Measures echo return time
4. Calculates distance using speed of sound

Distance formula:

$$Distance = \frac{Speed\ of\ Sound \times Time}{2}$$

The system calculates the user's height from the measured distance.

### Uses

- Automatic height measurement
- Non-contact sensing
- Distance calculation

## 5. GSM MODULE (SIM800L / SIM900)

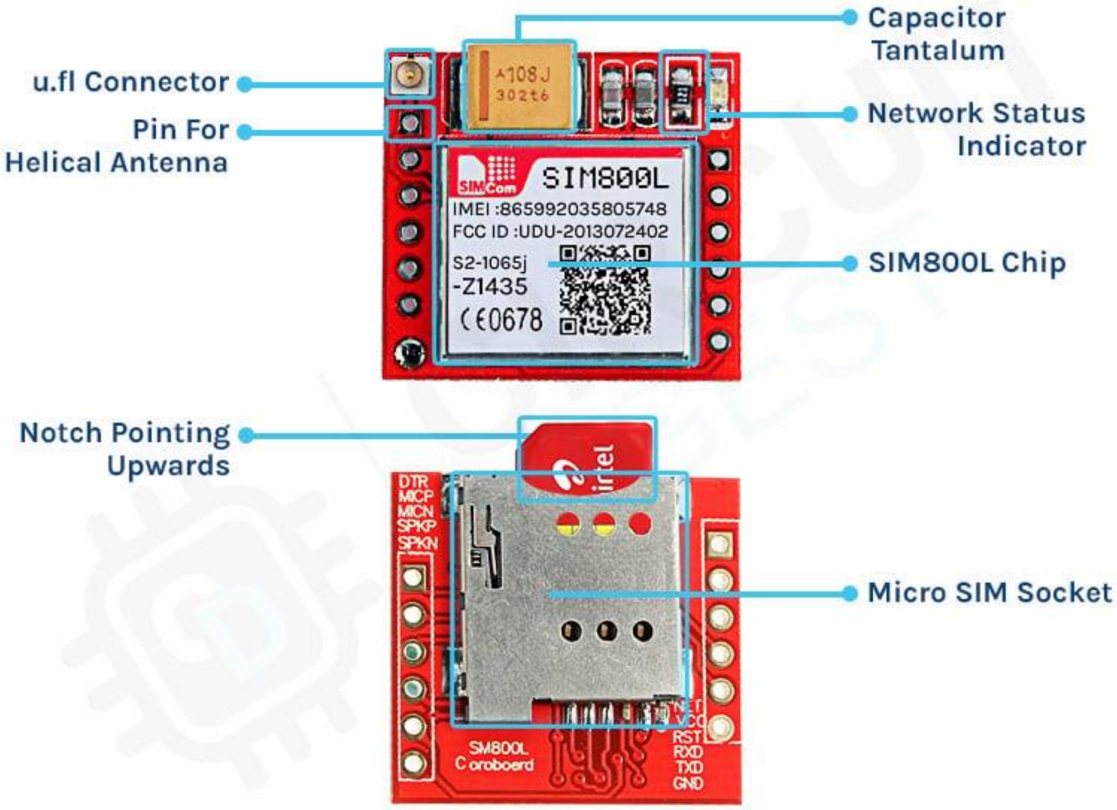



**Fig. 4.5 – SIM800L GSM Module.**

**Working**

The GSM module communicates with ESP32 through UART serial communication.

The ESP32 sends AT commands to the GSM module to transmit SMS notifications containing:

- BMI value
- Health category
- User information

**Uses**

- SMS alert generation
- User notifications
- Remote communication

## 6. LCD DISPLAY MODULE (16x2 WITH I2C)


**Fig. 4.6 – 16x2 LCD Display.**

**Working**

The LCD module displays real-time information received from ESP32 using I2C communication.

The display shows:

- Weight
- Height
- BMI value
- Health category

The I2C interface reduces wiring complexity by using only:

- SDA pin
- SCL pin

**Uses**

- Visual output display
- User interaction
- Real-time result display

## 7. 4X4 MATRIX KEY PAD

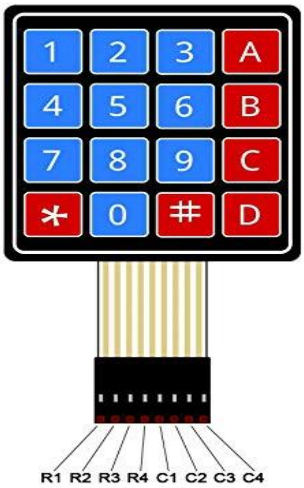



**Fig. 4.7 – 4x4 Matrix Keypad.**

**Working**

The 4x4 Matrix Keypad is used to provide user input to the ESP32 microcontroller for entering details such as:

- Name selection
- Age
- Mobile number
- Menu options
- User commands

The keypad consists of 16 keys arranged in a matrix format of:

- 4 Rows
- 4 Columns

## 8. POWER SUPPLY UNIT

**Fig. 4.8 – 12V 2A Adapter.**

**Working**

The power supply converts input voltage into regulated DC voltage required for electronic components. A regulated power supply ensures:

- Stable operation
- Proper voltage levels
- Component safety

**Uses**

- Powers ESP32
- Powers sensors and modules
- Maintains system stability

**9. PVC PIPE & BOARD PLATFORM STRUCTURE**

**Fig. 4.9 – PVC Pipes.**

**Working**

The mechanical platform supports the user during weight measurement.

The load cell is mounted beneath the platform for accurate force sensing, while the ultrasonic sensor is mounted vertically for height detection.

**Uses**

- Stable user standing platform
- Accurate weight measurement
- Sensor mounting structure
- Safe operation

## 10. CONNECTING WIRES AND PCB

**Working**

Connecting wires and PCB are used to establish electrical connections between all electronic components.

The PCB provides:

- Proper circuit layout
- Reduced wiring complexity
- Better durability

**Uses**

- Circuit integration
- Stable electrical connections
- Component mounting

**Software Tools:**

1. ARDUINO IDE

Arduino IDE Official Website

Arduino IDE is used to write, compile, debug, and upload Embedded C/C++ programs into ESP32.

The IDE provides:

- Code editor
- Serial monitor
- Library management
- Debugging tools

Uses

- ESP32 programming
- Sensor interfacing
- Testing and debugging
- Program uploading

2. EMBEDDED C/C++ PROGRAMMING

Embedded C/C++ is used to develop the software logic of the system.

The program handles:

- Sensor reading
- BMI calculation

- LCD display control
- GSM communication
- Cloud communication

Uses

- Automation logic
- Real-time processing
- Hardware control
- BMI algorithm implementation

3. IoT CLOUD PLATFORM

Examples:

- ThingSpeak
- Firebase
- Blynk

Working

The cloud platform stores and monitors health data uploaded by ESP32 through Wi-Fi communication.

The cloud dashboard provides:

- Real-time monitoring
- Historical data storage
- Graphical analysis
- Remote accessibility

Uses

- Cloud data storage
- IoT monitoring
- Health record management
- Remote access

4. SERIAL MONITOR

The Serial Monitor in Arduino IDE is used to observe real-time sensor readings and debug system performance.

Uses

- Debugging
- Sensor testing

- Error checking
- Data monitoring

5. DATABASE / CLOUD STORAGE

The database stores:

- User details
- Weight records
- Height records
- BMI history
- Date and time

This enables long-term healthcare monitoring and historical analysis.

Uses

- Data storage
- Historical tracking
- Multi-user record maintenance
- Digital healthcare management

## CHAPTER 5

## RESULTS

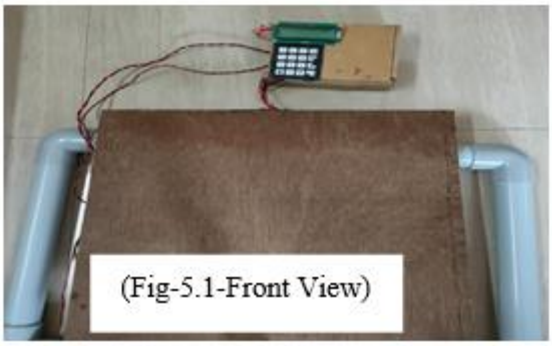
**Fig-5.2 Ultrasonic Sensor.**

This Figure shows the height measuring Ultrasonic Sensor of the model.

**Fig-5.3- The base for Weight Measurement.**

This Figure shows the base of the Model for weight measurement.

**F Fig-55.2**



(Fig-5.1-Front View)

This Figure shows the Front view of actual demonstrated model of the BMI Monitoring system.

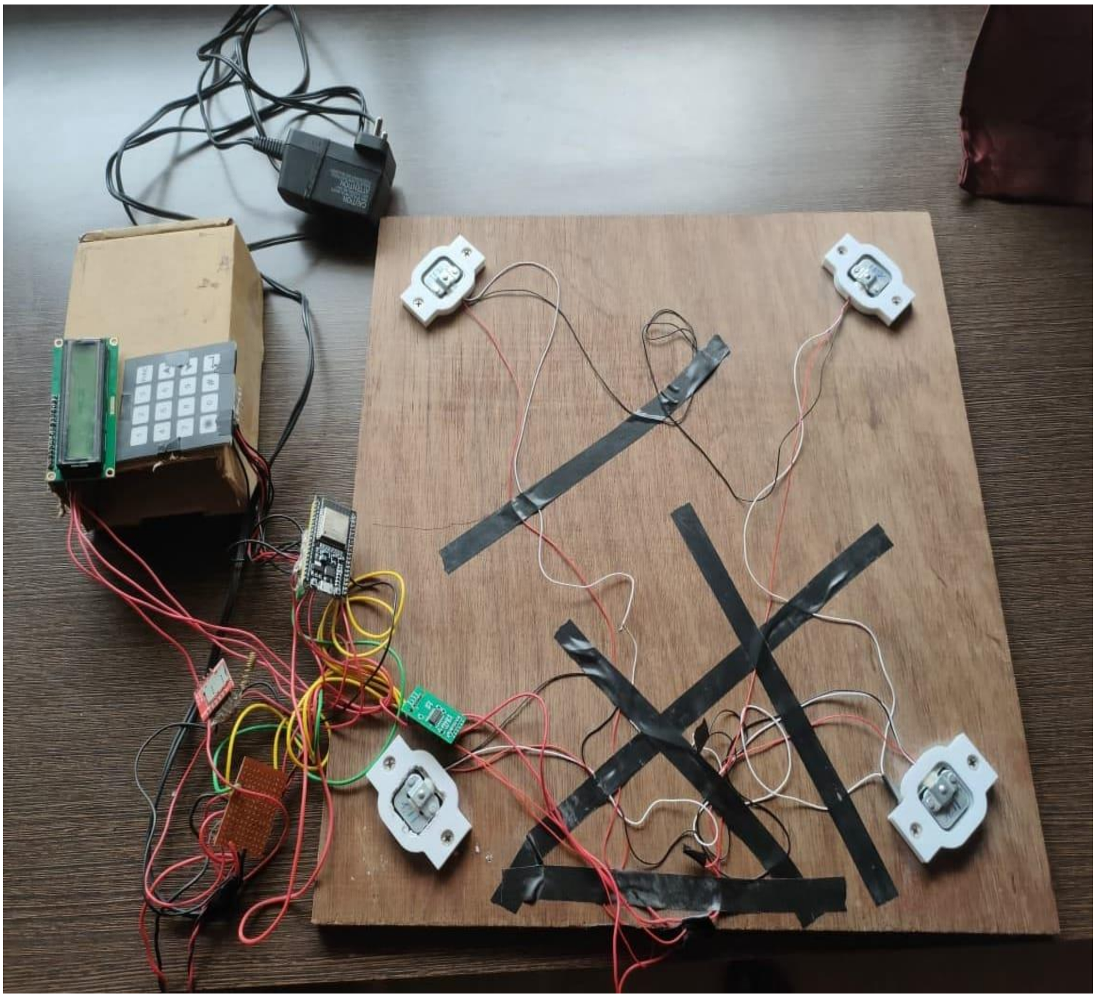


**Fig-5.4- Wiring Assembly.**

The Figure shows the wiring and hardware assembly of the IoT Enabled Smart BMI Monitoring System prototype. Four load cells are mounted at the corners of the wooden platform in bridge configuration for accurate weight measurement. The load cells are connected to the HX711 amplifier module, which sends processed weight data to the ESP32 microcontroller. The ESP32 is connected with a 16x2 LCD display and a 4x4 keypad for displaying BMI results and taking user input. All components are interconnected using jumper wires, and the wiring is secured using insulation tape for proper connection and stability. The setup represents the practical implementation of the smart BMI monitoring platform.

## CHAPTER 6

## ADVANTAGES AND LIMITATIONS

**Advantages:**

1. The system automatically measures height and weight and calculates BMI accurately

without manual calculation, reducing human errors and saving time.

2. The use of load cells and ultrasonic sensors provides fast and accurate measurement of body parameters in real time.
3. The ESP32 microcontroller processes data quickly, allowing the system to display BMI results instantly.
4. The IoT cloud connectivity enables real-time monitoring and storage of health records through mobile or web dashboards.
5. The GSM module sends automatic SMS alerts containing BMI value and health status to the registered mobile number, improving health awareness.
6. The system stores historical health records digitally, which helps in long-term health tracking and analysis.
7. The project is user-friendly and operates automatically with minimal human intervention.
8. The system is portable and compact, making it suitable for hospitals, schools, gyms, fitness centres, and rural health camps.
9. The project provides a low-cost healthcare solution using affordable embedded system components.
10. The system reduces manual workload for healthcare staff by automating measurement, calculation, and monitoring processes.

**Limitations**

1. The accuracy of the system depends on proper calibration of sensors, and incorrect calibration can lead to inaccurate BMI readings.
2. The ultrasonic sensor may be affected by environmental conditions such as noise, humidity, and improper echo reflection.
3. The IoT cloud monitoring feature requires stable internet connectivity, and remote monitoring may fail during network issues.
4. BMI alone cannot fully determine a person’s complete health condition because it does not measure body fat percentage or other body parameters.
5. The GSM alert system depends on mobile network availability, and SMS delivery may be delayed in weak signal areas.
6. The system requires periodic maintenance and recalibration of sensors to maintain accurate and stable performance.

## CONCLUSION

The **IoT Enabled Smart BMI Monitoring System** was successfully designed and implemented using embedded systems, sensor technology, GSM communication, and IoT cloud connectivity. The project provides an automated and efficient solution for measuring height, weight, and Body Mass Index (BMI) accurately in real time. The system uses a load cell for weight measurement and an ultrasonic sensor for height measurement. The ESP32 microcontroller processes the sensor data, calculates BMI automatically, classifies the health condition, and displays the results on the LCD screen. The measured data is also uploaded to the cloud platform for remote monitoring and historical record maintenance. Additionally, the GSM module sends SMS alerts containing BMI values and health status to the registered mobile number. The project reduces manual effort, improves accuracy, and supports digital healthcare monitoring. It is cost-effective, portable, user-friendly, and suitable for hospitals, schools, fitness centres, and rural healthcare camps.

Overall, the project demonstrates the effective integration of IoT technology and embedded systems in smart healthcare applications. The system can be further enhanced in the future by integrating additional health monitoring parameters such as heart rate, blood pressure, temperature, and SpO2 monitoring for complete healthcare analysis.

## REFERENCES

1. IEEE Xplore, *IoT-Based Smart Health Monitoring System*, IEEE Conference Publication,2024. (Describes real-time remote monitoring of health parameters using ESP32/ESP8266, sensors, and cloud platform for alert generation.)
2. International Journal of Engineering Research & Technology (IJERT), *Design and Implementation of Digital BMI Measurement System*, IJERT, 2023. (Presents automated BMI calculation using load cell, ultrasonic sensor, and microcontroller to reduce manual errors.)
3. ScienceDirect, *Cloud-Integrated Smart Healthcare Monitoring System*, Elsevier Publication,2024. (Explains IoT device integration with cloud database for real-time monitoring and historical health record analysis.)
4. International Journal for Scientific Research & Development (IJSRD), *GSM-Based Health Alert System Using Embedded Systems*, IJSRD, 2023. (Describes implementation of GSM module for automatic SMS alert mechanism in embedded healthcare systems.)

5. ResearchGate, *Smart IoT-Based Body Parameter Monitoring System*, Research Publication, 2024. (Focuses on multi-parameter body monitoring with IoT connectivity and centralized data storage.)
6. IEEE Xplore, IoT-Based Healthcare Monitoring and Alert System, IEEE Access, 2024. (Explains remote patient monitoring using IoT sensors, cloud storage, and GSM alert systems for healthcare applications.)
7. IEEE Xplore, ESP32-Based Smart Healthcare Monitoring System with Cloud Integration, IEEE Publication, 2024. (Describes real-time health data acquisition, cloud monitoring, and wireless communication using ESP32 technology.)
8. IEEE Xplore, Wireless Health Monitoring System Using GSM and IoT, IEEE International Journal, 2024. (Explains implementation of GSM communication and IoT cloud services for remote healthcare monitoring and alert generation.)
9. IEEE Xplore, Intelligent Body Parameter Monitoring System Using IoT, IEEE Conference Proceedings, 2023. (Focuses on monitoring body parameters such as weight and BMI using IoT-enabled embedded systems and cloud databases.)
10. IEEE Xplore, Wireless Health Monitoring System Using GSM and IoT, IEEE International Journal, 2024. (Explains implementation of GSM communication and IoT cloud services for remote healthcare monitoring and alert generation.)