**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

**PNR No. : Roll No.:**

1. Name of the Students :

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. Title of the project :

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Name of the Guide :

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Teaching Experience of the Guide \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Is this your first submission? Yes No

Signature of the student Signature of the Guide

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of the Coordinator

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Smart Health Monitoring System**

**A Project Report**

Submitted in partial fulfillment of the

Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)**

**By**

Member 1 Seat Number :

And

Member 2 Seat Number :

**Under the esteemed guidance of**

**Mrs. Teacher**

****

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**SIES COLLEGE OF COMMERCE AND ECONOMICS**

***(Affiliated to University of Mumbai)***

**MUMBAI-400022**

**MAHARASHTRA**

**2018-2019**

**SIES COLLEGE OF COMMERCE AND ECONOMICS**

***(Affiliated to University of Mumbai)***

**MUMBAI-MAHARASHTRA-400022**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

****

**CERTIFICATE**

This is to certify that the project entitled, **“SMART HEALTH MONITORING SYSTEM”,** is bonafied work of **MEMBER 1** bearing Seat No.: ( ) and **MEMBER 2** bearing seat No. :( ) Submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

**Internal Guide Coordinator**

**External Examiner**

**Date: College Seal**

**ABSTRACT**

Present day's IoT brings the gadgets together and assumes a fundamental part in different methodologies like smart home mechanization, savvy urban areas, vehicle parking, traffic control, brilliant industries, smart environment, agribusiness fields and patient health monitoring system and so on. One of the approaches is to monitor the health state of the patient and screen it to doctors or paramedical staff through the IoT, as it is hard to screen the patient for 24 hours. So here the patient health condition or status i.e. Pulse rate, Body Temperature, panic button and so on can be measured by utilizing the Non-invasive sensors. These sensors are associated with the Arduino Uno board, it gathers the information i.e. biomedical data from the sensors and the detected biomedical information can be transmitted to the server. The “Thingspeak” named new cloud is utilized here to place the detected information into the server. From this server the information can be envisioned to the specialists and other paramedical staff by Thingspeak android app. In this way by utilizing this Smart health monitoring system diminishes the exertion of specialists and paramedical staffs to screen the patient for 24 hours and furthermore lessens the time and cost of support.

**ACKNOWLEDGEMENT**

**DECLARATION**

I hereby declare that the project entitled, **“Smart Health Monitoring System”** has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

**Name and Signature of the Student**

**Table of Contents**

Chapter 1: Introduction 8

1.1 Background 8

1.2 Objectives 9

1.3 Purpose and Scope 10

1.3.1 Purpose 10

1.3.2 Scope 10

1.3.3 Limitation of the study 11

1.4 Applicability 11

Chapter 2: SURVEY OF TECHNOLOGIES 12

Chapter 3: SYSTEM ANALYSIS 14

3.1 Problem Definition 14

3.2 Proposed System 14

3.3 Requirement Analysis 15

3.4 Requirement Specification 16

3.4.1 Hardware Requirements 16

3.4.2 Software Requirements 16

3.5 Conceptual Design 18

3.6 Justification of Selection of Technology 19

Chapter 4: SYSTEM DESIGN 21

4.1 Module design 21

4.2 Procedural Design 21

4.2.1 Circuit Diagram 21

4.2.2 Algorithms 22

4.2.3 Logic Diagrams 22

4.3 UML Diagrams 25

4.3.1 Use Case Diagrams 25

4.3.2 State Transition Diagram 25

4.4 User Interface 26

# **Chapter 1: Introduction**

## **1.1** **Background**

A Remote health monitoring system is an extension of a hospital medical system where a patient’s vital body state can be monitored remotely. Traditionally the detection systems were only found in hospitals and were characterized by huge and complex circuitry which required high power consumption. Continuous advances in the semiconductor technology industry have led to sensors and microcontrollers that are smaller in size, faster in operation, low in power consumption and affordable in cost. This has further seen development in the remote monitoring of vital life signs of patients especially the elderly. The remote health monitoring system can be applied in the following scenarios:

1. A patient is known to have a medical condition with unstable regulatory body system. This is in cases where a new drug is being introduced to a patient.

2. A patient is prone to heart attacks or may have suffered one before. The vitals may be monitored to predict and alert in advance any indication of the body status.

3. Critical body organ situation

4. The situation leading to the development of a risky life-threatening condition. This is for people at an advanced age and maybe having failing health conditions.

5. Athletes during training. To know which training regimes will produce better results. In recent times, several systems have come up to address the issue of remote health monitoring.

The systems have a wireless detection system that sends the sensor information wirelessly to a remote server.

## **1.2** **Objectives**

Here the main objective is to design a Remote Patient Health Monitoring System to diagnose the health condition of the patients. Giving care and health assistance to the bedridden patients at critical stages with advanced medical facilities have become one of the major problems in the modern hectic world. In hospitals where many patients whose physical conditions must be monitored frequently as a part of a diagnostic procedure, the need for a cost-effective and fast responding alert mechanism is inevitable. Proper implementation of such systems can provide timely warnings to the medical staffs and doctors and their service can be activated in case of medical emergencies. For this, here we use the idea of network technology with wireless applicability, providing each patient a unique ID by which the doctor can easily identify the patient and his/her status of health parameters. Using the proposed system, data can be sent wirelessly to the Patient Monitoring System, allowing continuous monitoring of the patient. Contributing accuracy in measurements and providing security in proper alert mechanism give this system a higher level of customer satisfaction and low-cost implementation in hospitals. Thus, the patient can engage in his daily activities in a comfortable atmosphere where distractions of hardwired sensors are not present. The system we proposed here is efficient in monitoring the different physical parameters of many number bedridden patients and then in alerting the concerned medical authorities if these parameters bounce above its predefined critical values. Thus, remote monitoring and control refer to a field of industrial automation that is entering a new era with the development of wireless sensing devices.

Such a connectivity with external devices and services will allow for taking preventive measure (e.g., upon foreseeing an upcoming heart stroke) or provide immediate care (e.g., when a user falls and needs help)

## **1.3 Purpose and Scope**

### **1.3.1 Purpose**

Design a Remote Patient Health Monitoring System (RPHMS) which has heartbeat detection system, temperature detection system and immediate response panic detection system. A doctor or health specialist can use the system to monitor remotely of all vital health parameters of the patient or person of interest. An attempt at designing a remote healthcare system made with locally available components.

i) The temperature, pulse rate and emergency panic response data is collected. The data collected was transmitted wirelessly to a receiver module.

ii)The detected analog signal was then digitized to give a digital value that was read on the LCD.

iii) A simple cloud server where hosted with a database for all the vital data to be accessed remotely whenever required.

### **1.3.2 Scope**

Solving the healthcare problem with engineering approach by developing a remote health care system. In so doing bridging of the gap between the doctor and patient with modern current available components sensors.

Another significance is giving back to society. This is to help the senior citizens who most of the time are alone and people are known to have heart problems. A heartbeat detector is also necessary to monitor their heart conditions.

It is also proof of classroom concepts in real life scenarios. The analog and digital electronics learnt is put to practice and tested.

**1.3.3 Limitation of the study and delimitations of the study– scope**

The scope of the project was limited to cardiac signal, body temperature and remote viewing of the collected data.

The second Physiological category comprising of EEG monitor and Ultrasound detector were to be designed with more strict specifications. The most important specification considered was that they should be safe to use and accurate. This is because the physiological information being detected determines the severity of a critical life-threatening situation.

EEG signal and ultrasound were left out due to financial constraints on the type of probes and ICs needed in detection and time allocated for the project. Sourcing and fabrication of the circuit necessary for the ultrasound and EEG detectors.

**1.3.4 Applicability**

1) IOT Monitoring proves really helpful when we need to monitor & record and keep track of changes in the health parameters of the patient over the period of time. So with the IOT health monitoring, we can have the database of these changes in the health parameters. Doctors can take the reference of these changes or the history of the patient while suggesting the treatment or the medicines to the patient.

2) Hospital stays are minimized due to Remote Patient Monitoring.

3) Hospital visits for normal routine checkups are Minimized.

4) Patient health parameter data is stored over the cloud. So it is more beneficial than maintaining the records on printed papers kept in the files. Or even the digital records which are kept in a particular computer or laptop or memory device like pen- drive. Because there are chances that these devices can get corrupt and data might be lost. Whereas, in case of IOT, the cloud storage is more reliable and does have minimal chances of data loss.

# **Chapter 2: SURVEY OF TECHNOLOGIES**

Before going to the implementation of different technologies let us see the types of Technologies Available for the project:

**Arduino UNO**

An Arduino is a microcontroller motherboard. A microcontroller is a simple computer that can run one program at a time, over and over again. It is very easy to use.

An Arduino board is best used for simple repetitive tasks: opening and closing a garage door, reading the outside temperature and reporting it to Twitter, driving a simple robot.

Arduinos contain a number of different parts and interfaces together on a single circuit board. The design has changed through the years, and some variations include other parts as well. But on a basic board, you’re likely to find the following pieces:

* A number of pins, which are used to connect with various components you might want to use with the Arduino. These pins come in two varieties:
  + Digital pins, which can read and write a single state, on or off. Most Arduinos have 14 digital I/O pins.
  + Analog pins, which can read a range of values, and are useful for more fine-grained control. Most Arduinos have six of these analog pins.
* These pins are arranged in a specific pattern, so that if you buy an add-on board designed to fit into them, typically called a “shield,” it should fit into most Arduino-compatible devices easily.
* A power connector, which provides power to both the device itself, and provides a low voltage which can power connected components like LEDs and various sensors, provided their power needs are reasonably low. The power connector can connect to either an AC adapter or a small battery.

**Programming in Arduino**

Arduino programs are written in the Arduino Integrated Development Environment (IDE). Arduino IDE is a special software running on your system that allows you to write sketches (synonym for program in Arduino language) for different Arduino boards. The Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language. After the sketch is written in the Arduino IDE, it should be uploaded on the Arduino board for execution.

# **Chapter 3: SYSTEM ANALYSIS**

## **3.1** **Problem Definition**

In the healthcare domain of IoT, patients will not have to make as many trips to the doctor anymore, since they can upload the collected data from the sensors to the cloud from the comfort of their home for a doctor or trained specialist to review. This can be achieved for an health monitoring application on the mobile device, which will collect the bio-signal data using a micro-controller and then upload to the cloud for keeping a record of the unstructured data. This will reduce the waiting time for the triage at the hospitals and minimize visits, but more importantly reducing the cost of personnel and administrative operations. This convenience increases the quality of life for the patients as they can enjoy other activities instead of spending time commuting to the hospital/clinic and waiting in long triage queues. Pattern recognition and analysis can also be applied in real time across a large set of data to support things like predicting heart strokes for cardiovascular patients.

The huge volume of data produced from the sensors is in an unstructured format, which is very complex to understand and requires different data storage mechanisms than the typical database management system (DBMS). In summary, distributed computing, cloud computing and faster processors allow the analysis of this data explosion manageable in order to make improvements in human life, environment interaction as well as social connection.

## **3.2** **Proposed System**

The idea of building an integrated IoT and cloud based solution for healthcare applications has been around for sometime. Using the idea of IoT and cloud techniques, this project presents a solution to use an IO microcontroller board, which obtains the bio-signal data from a person using pulse monitor and other sensors and sends it to the cloud (Thingspeak) and mobile can access these data from the cloud . When monitoring the the patient, the monitored data associated with these sensors being displayed on the mobile app is uploaded to Thingspeak’s private database. The Filestream and Filetable technologies present in Thingspeak’s Server allow the storage of unstructured data. With the proper hardware components like the IO microcontroller and the sensors, the solution can monitor the health of a person in any environment at low costs, without having to purchase any costly monitoring devices.

**3.3 Requirement Analysis**

**Arduino Uno:**

Since it’s the latest and greatest version and most capable of handling smart health monitoring needs.

**Esp8266 (Wifi Sensor):**

This will allow you to communicate with and controller and the Thingspeak server.

**Pulse rate sensor:**

This will detect the pulse rate of the person being monitored

**LM35 (Temperature Sensor):**

This will detect the body temperature of the person being monitored

**Button (Panic Button):**

It is used in the emergency situation when the person being monitored is not feeling any good by pressing this button an emergency sos is being transmitted to everyone who are registered as emergency contact in the database.

## **3.4 Requirement Specification**

### **3.4.1 Hardware Requirements**

|  |
| --- |
| Arduino Processor ATMEGA 328 |
| Memory 2K SRAM, 32K Flash, 1kB EEPROM |
| Network ESP8266 wifi interfacing |
| Components LM35 temperature sensor,  Pulse Rate sensor, Push button,  10K Resistor,  220 or 100 ohms Resistors  Connection Blocks  Breadboard |
| Wires Male - Female Jumper wires |

### 

### **3.4.2 Software Requirements**

Arduino IDE:

The **Arduino integrated development environment (IDE)** is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.

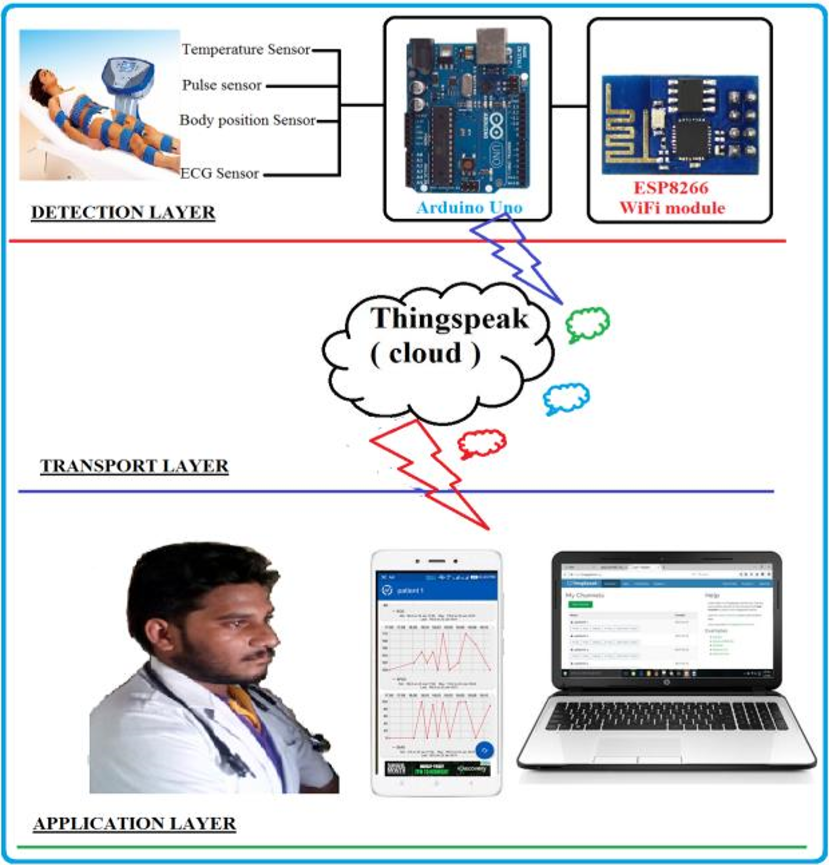
The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring.[[4]](https://en.wikipedia.org/wiki/Arduino_IDE#cite_note-4) The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution.[[5]](https://en.wikipedia.org/wiki/Arduino_IDE#cite_note-5) The Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

Thingspeak:

"**ThingSpeak** is an [open source](https://en.wikipedia.org/wiki/Open_source) [Internet of Things](https://en.wikipedia.org/wiki/Internet_of_Things) (IoT) application and [API](https://en.wikipedia.org/wiki/API) to store and retrieve data from things using the [HTTP](https://en.wikipedia.org/wiki/HTTP) protocol over the Internet or via a Local Area Network. ThingSpeak enables the creation of sensor logging applications, location tracking applications, and a social network of things with status updates"

ThingSpeak is an IoT platform, that allows you to connect and save sensor data in the cloud and develop IoT applications. Also, the platform provides apps that let you analyze and visualize data. MATLAB support helps you act on data. Sensor data can be easily integrated and sent from Arduino or Raspberry Pi or any other IoT gateway.

## **3.5 Conceptual Design**



## **3.6 Justification of Selection of Technology**

## **Reason #1: It is portable**

Heart rates vary by age with young people usually have a higher heart rate than people over 60 years old. One common issue for every age group is that they are not very good at dealing with medical equipment. Also, larger, stationary monitoring equipment can only be used in hospitals while a patient is in bed. The need for a device, which allows patients to track their own health conditions is clear.

Qualities of a smart health monitoring systems:

* small size
* easy-to-use
* light weight
* portable

## **Reason #2: It can monitor health conditions all the time**

People use HMSs in hospitals and clinics, for home care and to track the vitals of athletes including their heart rate, blood pressure, and body temperature. All this data can be tracked by various sensors integrated into the HMS.

Health issues are occurring more frequently than 30 years ago because of industrialization, climate change and technical progress that has led to decreased physical activity. Saving lives requires monitoring the health conditions of people who have chronic diseases or heart-related problems daily. Detecting problems early can extend lives and reduce morbidity from disease.

## **Reason #3: It can be cheaper than existing solutions**

The cost-effectiveness of the solution is one of the main factors when deciding to develop an HMS. Either way, the number of lives that could be saved depends on whether they have the necessary tools. This is why development a system that only has the necessary functions will help reduce expenses on the HMS design.

Another aspect that can reduce costs is using easily accessible, widely used and fully configurable components to develop the HMS. Using programmable components removes the likelihood that you will choose an inconvenient, non-cost-effective device for the system. The best solution is to build a healthcare monitoring system based on FPGA.

## **Reason #4: It simplifies health monitoring for medical staff**

A simple system is most useful when it comes to monitoring health conditions of a particular patient. This patient can stay at home and people close-by or doctors on duty will always be updated to respond to emergencies.

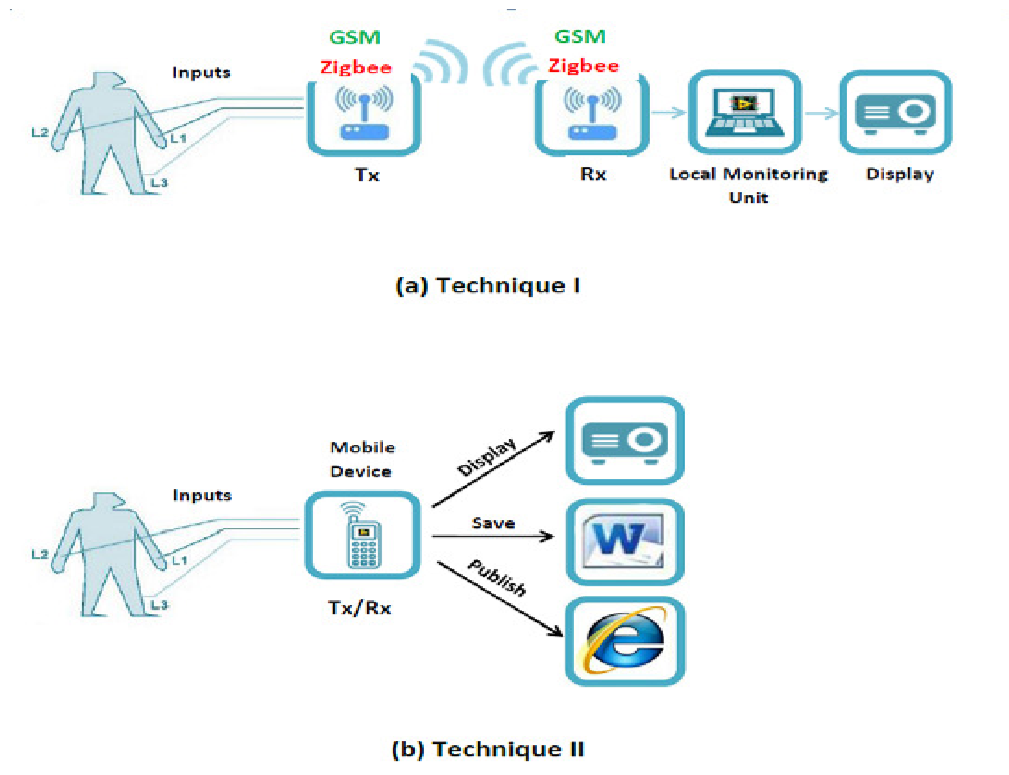
However, medical organizations have many patients and using a system like this for each patient is not cost-effective or efficient. They need one advanced system to monitor every patient in a facility. This is why a health monitoring system for hospitals is needed, so that the HMS would be able to store and recognize data for each patient.

## **Reason #5: It is multifunctional**

The multi-functionality of the HMS provides medical staff with timely information and suggestions to stabilize a patient

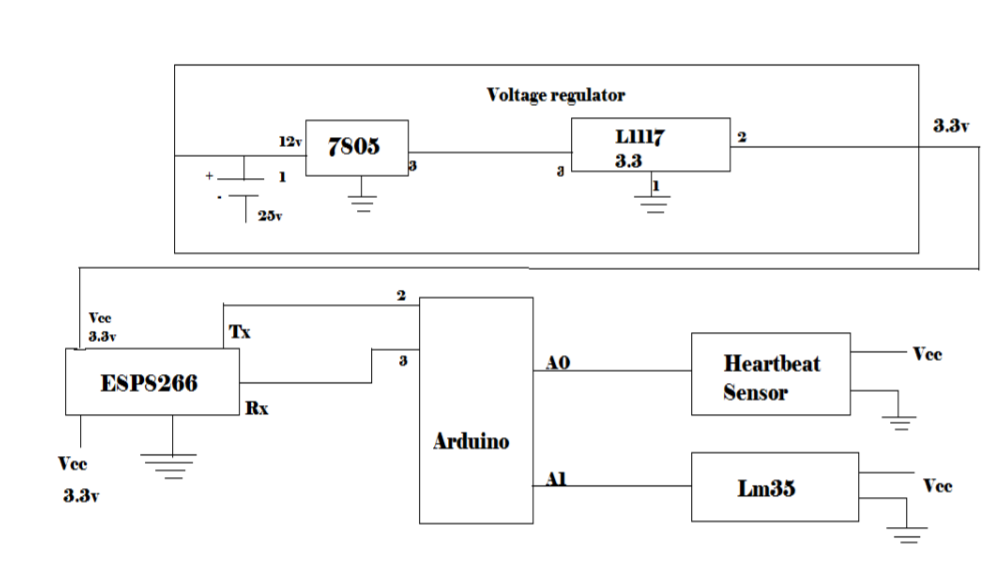
# **Chapter 4: SYSTEM DESIGN**

## **4.1 Module design**



## **4.2 Procedural Design**

### **4.2.1 Circuit Diagram**



### **4.2.2 Algorithms**

1. Start

2. Click On the App/ visit the Website

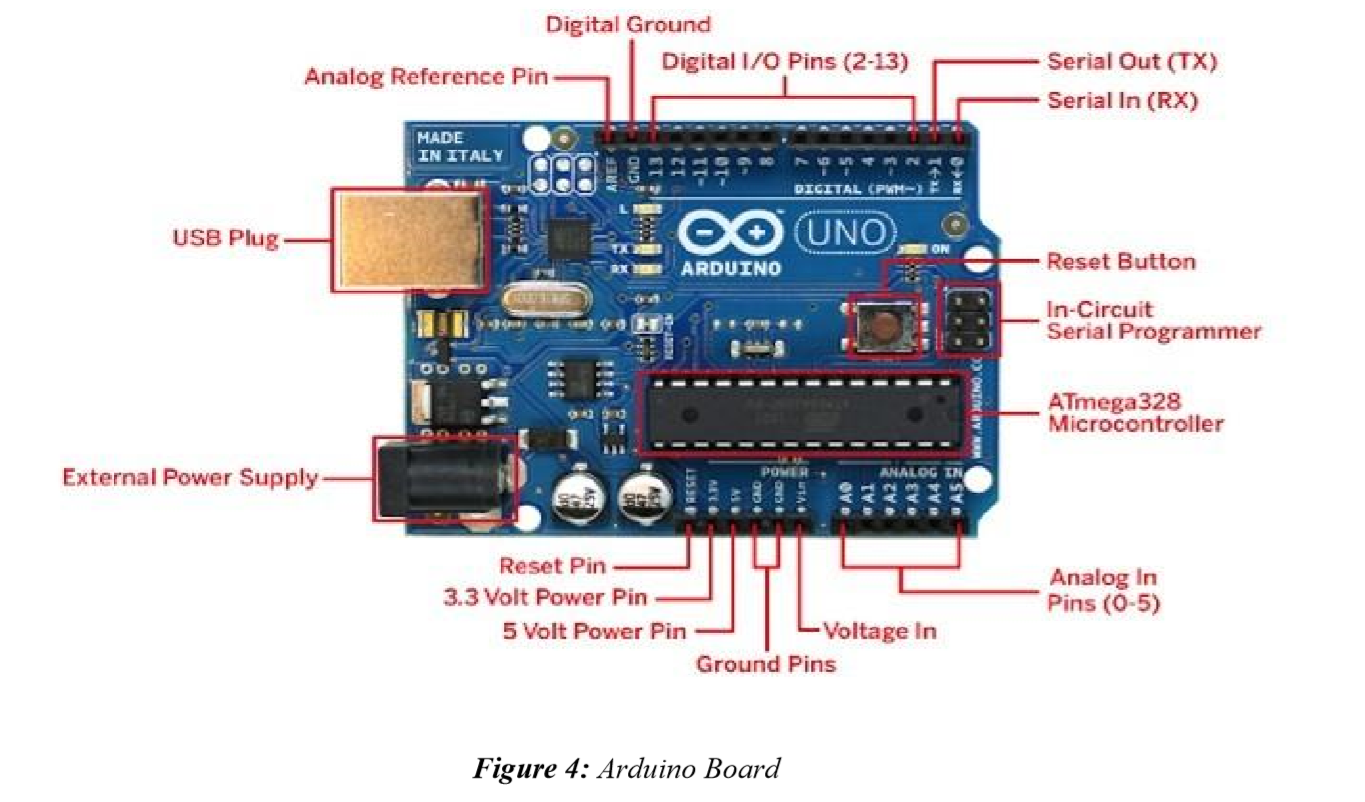
3. Select Home Appliances

4. Control Home Appliances

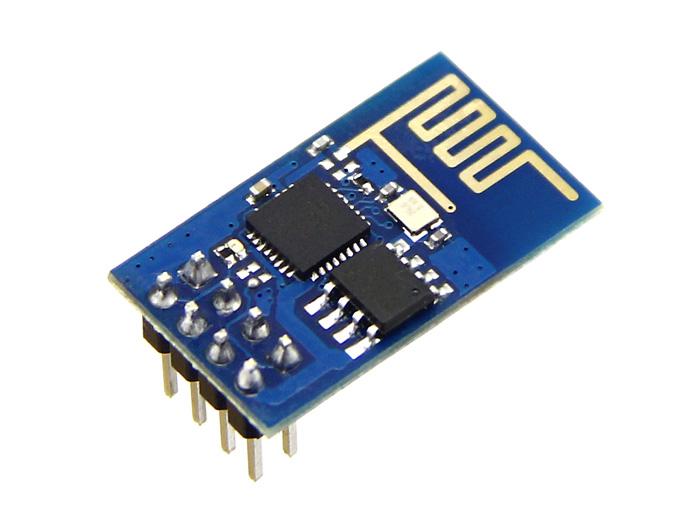
5. Stop

### **4.2.3 Logic Diagrams**

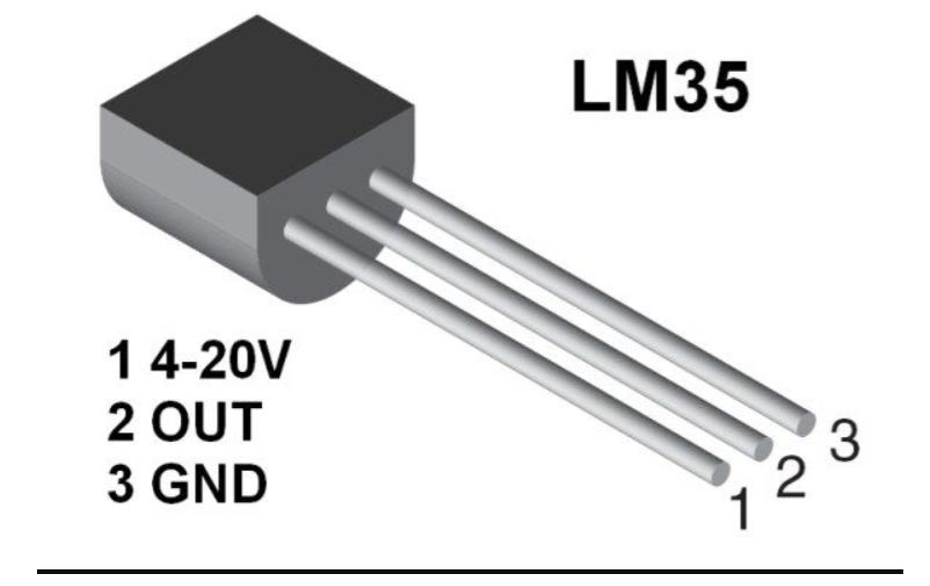
**Arduino**

****

**ESP8266 Wi-Fi module**

****

**LM35 Temperature Sensor**

****

**Pulse Rate Sensor**

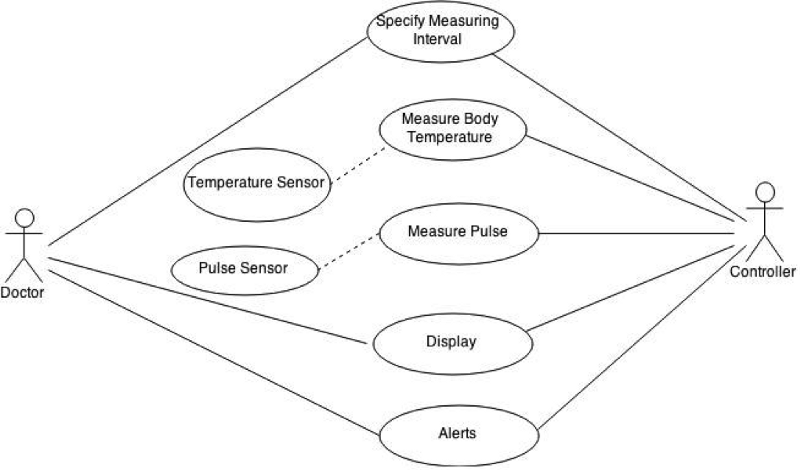
****

**Push Button**

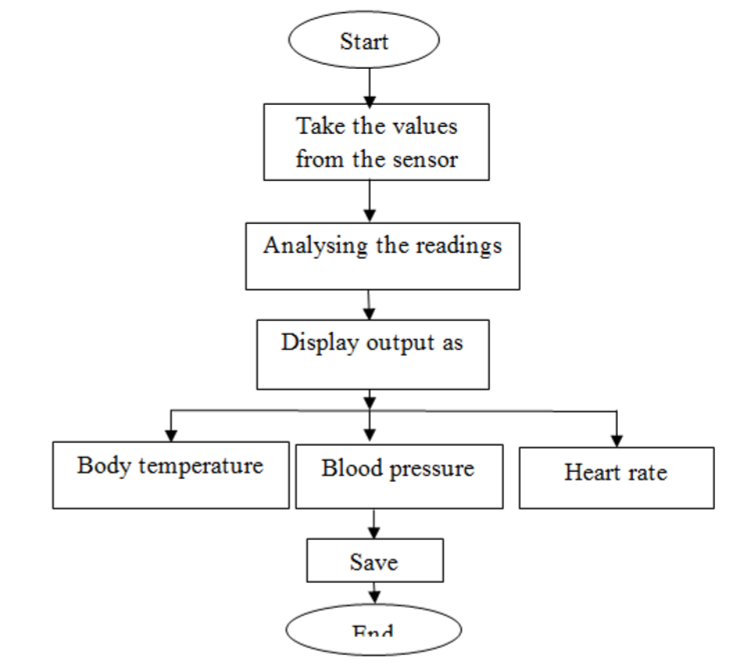
****

## **4.3 UML Diagrams**

### **4.3.1 Use Case Diagrams**



### **4.3.2 State Transition Diagram**



## **4.4 User Interface**

## 

### 