ROBAN SYSTEMS
AN HELPING HAND FOR DISABLED PEOPLE

A PROJECT REPORT

submitted by

Ajay Das K, Fathima Irfana T P, Muhammed Fais M T, Rabeeh C

Register Number: CCV19CS002, CCV19CS007, CCV19CS015, CCV19CS019

TO

The APJ Abdul Kalam Technological University

in partial fulfilment of the requirements for the award of the Degree of

Bachelor Of Technology

In

Computer Science and Engineering

Department Of Computer Science and Engineering

MGM COLLEGE OF ENGINEERING AND PHARAMACEUTICAL

SCIENCES

VALANCHERY

JUNE 2023

DECLARATION

I undersigned hereby declare that the main project report " **Roban systems - An helping hand for disabled people**", submitted for partial fulfilment of the requirements for the award of the degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a Bonafede work done by me under supervision of **MS. ANJANA**, Asst. Professor, Department of Computer Science and Engineering. This submission represents my ideas in my own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the sources. We also declare that we have adhered to academic honesty and integrity ethics and have not misrepresented or fabricated any data, idea, fact, or source in our submission. We understand that any violation of the above will causes disciplinary action by the institute and/or the University and can also evoke penal action from the sources that have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed as the basis for the award of any degree, diploma or similar title of any other University.

Place: Valanchery

Signature

Date: 20 June 2023

Ajay Das K (CCV19CS002)

Fathima Irfana T P(CCV19CS007)

Muhammed Fais M T (CCV19CS015)

Rabeeh C (CCV19CS019)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING MGM COLLEGE OF ENGINEERING AND PHARMACEUTICAL SCIENCES, VALANCHERY

CERTIFICATE

This is to certify that the project report entitled " **Roban Systems - An helping Hand for disabled people**" submitted by Ajay Das K, Fathima Irfana T P, Muhammed Fais M T, Rabeeh C(CCV19CS002, CCV19CS007, CCV19CS015, CCV19CS019)to the APJ Abdul Kalam Technological University in partial fulfilment of the requirements for the award of the Degree of Master of Technology in Computer Science and Engineering is a bonafide record of the project work carried out by her under my guidance and supervision during the year 2022- 2023. This report in any form has not been submitted to any other University or Institute for any purpose

Internal Supervisor External Supervisor Head of the Department

ACKNOWLEDGEMENT

Alone we can do so little; together we can do so much. Likewise, the present project work has been undertaken and completed with many people's direct and indirect help, and I would like to acknowledge the same.

First and foremost, I take immense pleasure in thanking the **Management** and respected principal **DR. PROF BABU JOHN,** for providing me with the wider facilities.

We sincerely thank **MS. SHABNA M**, Head of the Department of Computer Science and Engineering for allowing me to present this project and for timely suggestions.

We wish to express my deep sense of gratitude to the project coordinator **MS. SHABNA M**, Head of the Department of Computer Science and Engineering, who coordinated on the right path. Words are inadequate in offering my thanks to **MS. ANJANA T**, Asst Professor Department of Computer Science and Engineering, for her encouragement and guidance in carrying out the project.

Needless to mention that the teaching and the non-teaching faculty members had been the source of inspiration and timely support in the conduct of our project. We would like to express our heartfelt thanks to our beloved parents for their blessings, and our classmates for their help and wishes for the successful completion of this project.

Above all, we would like to thank the Almighty God for the blessings that helped us complete the venture smoothly. Without their support, it would be impossible for us to finish our work. That is why we wish to dedicate this section to recognize their support.

AJAY DAS K

FATHIMA IRFANA TP

MUHAMMED FAIS M T

RABEEH C

ABSTRACT

Smart home applications are pervasive and have gained popularity due to the overwhelming use of the Internet of Things (IoT). The revolution in IoT technologies made homes more convenient, efficient and perhaps more secure. The need to advance smart home technology is necessary at this stage as IoT is abundantly used in the automation industry. However, most of the proposed solutions are lacking in certain key areas of the system i.e., high interoperability, data independence, privacy, and optimization in general. The use of machine learning algorithms requires high-end hardware and is usually deployed on servers, where computation is convenient, but at the cost of bandwidth. However, more recently edge AI-enabled systems are being proposed to shift the computation burden from the server side to the client side enabling smart devices. In this paper, we take advantage of edge AI-enabled technology to propose a fully featured cohesive system for smart homes based on IoT and edge computing. The proposed system makes use of industry standards adopted for fog computing as well as providing robust responses from connected IoT sensors in a typical smart home. The proposed system employs edge devices as a computational platform in terms of reducing energy costs and providing security, while remotely controlling all appliances behind a secure gateway. A case study of human fall detection is evaluated by a custom lightweight deep neural network architecture implemented over the edge device of the proposed framework. The case study was validated using the Le2i dataset. During the training, the early stopping threshold was achieved with 98% accuracy for the training set and 94% for the validation set. The model size of the network was 6.4 MB which is significantly lower than other networks with similar performance.

Keywords:- Artificial intelligence, Edge intelligence, IoT, Smart home, Deep learning

TABLE OF CONTENTS

Contents **Page No**

ACKNOWLEDGEMENT	I
ABSTRACT	II
TABLE OF CONTENTS	III
ABBREVIATION	IX
LIST OF FIGURES	XI
Chapter 1. INTRODUCTION	1
1.1. GENERAL BACKGROUND	2
1.2. MOTIVATION	3
1.3. OBJECTIVE	4
Chapter 2. LITERATURE SURVEY	6
2.1 ENABLING AUTOMATION AND-EDGE INTELLIGENCE OVER RESOURCE CONST	6
2.2 INTERACTING WITH A DIGITAL TWIN USING AMAZON ALEXA	6
2.3 AMAZON ALEXA TRAFFIC TRACES	7
2.4 IOT-BASED ELECTRO-SYNTHESIS ECOSYSTEM	8
2.5 ACCESS CONTROL AND SURVEILLANCE IN A SMART HOME	8
2.6 SMART IOT SURVEILLANCE MULTI-CAMERA MONITORING SYSTEM.	8

2.7 TRUSTBUILDER: A NON-REPUDIATION SCHEME FOR IOT CLOUD APPLICATIONS. 9

2.8 ACCURACY DETERMINATION USING DEEP LEARNING TECHNIQUE IN CLOUD-BASED IOT SENSOR ENVIRONMENT 9

2.9 SELF-SECURED MODEL FOR CLOUD-BASED IOT SYSTEMS 10

2.10 TAKING MQTT AND NODEMCU TO IOT: COMMUNICATION IN THE INTERNET OF THINGS 10

Chapter 3. SYSTEM OVERVIEW 11

3.1 PROBLEM STATEMENT 11

3.2 SYSTEM OUTLINE 12

3.3 PROPOSED METHODOLOGY 13

3.4 HARDWARE IMPLEMENTATION 16

3.4.1 IOT – SENSORS 16

3.4.2 STANDARD DEVICES 17

3.5 INTERNET OF THINGS – SOFTWARE 17

3.5.1 DATA COLLECTION 17

3.5.2 DEVICE INTEGRATION 17

3.5.3 REAL-TIME ANALYTICS 18

3.5.4 APPLICATION AND PROCESS EXTENSION 18

3.6 INTERNET OF THINGS - TECHNOLOGY AND PROTOCOLS 18

3.6.1 NFC AND RFID 18

3.6.2 LOW-ENERGY BLUETOOTH 18

3.6.3 LOW-ENERGY WIRELESS 18

3.6.4 RADIO PROTOCOLS 19

3.6.5 LTE-A 19

3.6.6 WIFI-DIRECT 19

3.7 FEATURE EXTRACTION 19

3.7.1 VOICE COMMAND 19

3.7.2 MODULE WISE ARCHETECTURE 20

3.7.3 ECONOMICAL 20

Chapter 4. REQUIREMENT ANALYSIS 21

4.1 FUNCTIONALITIES OF THE PROPOSED SYSTEM 21

4.2 MODULES 21

4.2.1 BED AND VENTILATION MODULE 22

4.2.2 BASIC NEEDS MODULE 22

4.2.3 SECURITY AND ENTERTAINMENT MODULE 23

4.2.4 PATIENT MONITOR MODULE 23

4.3 REQUIREMENTS 24

4.3.1 SYSTEM CONFIGURATION 24

4.3.2 ARDUINO IDE 24

4.3.3 NODEMCU 8266 25

4.3.4 RELAY 26

4.3.5 PCB 26

4.3.6 PUSH BUTTON 26

4.3.7 SERVO MOTOR 27

4.3.8 SMPS 28

4.3.9 IR BLASTER 28

4.3.10 AMAZON FIRE TV STICK 28

4.3.11 DAHUA IMOU RANGER 2 29

4.3.12 HDMI MONITOR 29

4.3.13 ALEXA ECHO 30

4.3.14 MAX3010 30

4.3.15 OLED DISPLAY MODULE 31

4.3.16 BLYNK IOT 31

4.3.17 ROUTER 32

4.3.18 AMAZON ALEXA APP 32

4.3.19 PUMP/VALVE 32

Chapter 5. SYSTEM DESIGN 33

5.1 DESIGN METHODOLOGY 33

5.2 SYSTEM ARCHITECTURE 33

5.2.1 SENSING LAYER 34

5.2.2 NETWORK LAYER 34

5.2.3 DATA PROCESSING LAYER 34

5.2.4 APPLICATION LAYER 35

5.3 PROPOSED FRAMEWORK 35

Chapter 6. PROJECT PANNING 38

6.1 PROTEUS DESIGN SUITE 38

6.2 AURDINO COMPILER 38

Chapter 7. IMPLEMENTATION AND TESTING 39

7.1. SETTING UP ARDUINO IDE 39

7.2 SETTING UP PCB 39

7.3 PROGRAMMING THE MICROCONTROLLER 41

7.4 SETTING UP AMAZON ALEXA AND FIRE STICK 41

7.5 SETTING UP THE SMART DEVICES 41

7.6 SETTING UP ALEXA ROUTINES. 42

7.7 SETTING UP BLYNK IOT 42

7.8 PROTEUS VSM 43

7.9 REAL-WORLD TESTING 44

7.9.1 CONNECTING DEVICES BY COMMANDS 44

7.9.2 MANUALLY CONNECT THE DEVICE 46

7.9.3 BLYNK APP TEST 47

Chapter 8. CONCLUSION AND FUTURE SCOPE 49

8.1 CONCLUSION 49

8.2 FUTURE SCOPE 50

REFERENCE 52

APPENDIX 54

A.1 Codes Flashed To Node Mcu 8266 54

A.2 Code for Basic needs module. 54

A.3 Code for Bed And Ventilation Module 58

A.4 Code patient monitor Module 60

ABBREVIATION

IoT	Internet of Things
Soc	System On Chip
NFC	Near Field Communication
RFID	Radio-Frequency Identification
NLP	Natural Language Processing
PCB	Printed Circuit Board
NodeMCU	Node Microcontroller Unit
P2P	peer-to-peer
Wi-Fi	Wireless Fidelity
SMPSA	Switched-Mode Power Supply
IR	Infrared
HDMI	High-Definition Multimedia Interface
LCD	Liquid Crystal Display
LED	Light-Emitting Diode
DAS	Data Acquisition System
MQTT	MQ Telemetry Transport
GCC	GNU Compiler Collection

URL	Uniform Resource Locator
IC	Integrated Circuit
USB	Universal Serial Bus
SpO2	Percentage Of Oxygen In The Blood
BPM	Beat Per Minute
ESP	Event Stream Processing
VR	virtual reality
AR	Augmented Reality

LIST OF FIGUERES

NO	Title	Page.No
Fig 3.1	Disability by Population Chart in India	11
Fig 3.2	Taxonamy of IoT used by disabled people	13
Fig 3.3	Proposed Methodology	14
Fig 3.4.1	List of some of the measurement devices used in IoT	16
Fig 3.7.1	Alexa command interface	20
Fig 4.2	Modules of Roban System	21
Fig 4.2.1	Bed And Ventilation Module	22
Fig 4.2.2	Basic Needs Module	22
Fig 4.2.3	Security And Entertainment Module	23
Fig 4.2.4	Patient Monitor Module	23
Fig 4.3.2	Arduino Ide	24
Fig 4.3.3	Nodemcu 8266	25
Fig 4.3.4	Relay	26
Fig 4.3.6	Push Button	27
Fig 4.3.7	DSC 3019 Servo Motor	27
Fig 4.3.9	IR Blaster	28
Fig 4.3.10	Amazon Fire Tv Stick	29

Fig 4.3.11	IP camera	29
Fig 4.3.12	HDMI Monitor	30
Fig 4.3.13	Alexa Echo	30
Fig 4.3.14	MAX3010	30
Fig 4.3.15	OLED Display Module	32
Fig 4.3.19	Different Pumps And Solenoid Valve	33
Fig 5.1	Design Methodology	
Fig 5.2	System Architecture	
Fig 5.3	Proposed Framework	
Fig 7.2	Schematic of basic need module and patient monitor module	
Fig 7.8	Proteus VSM	
Fig 7.9	Output Implementation in real world	
Fig 7.9.1.0	Alexa app interface	
Fig 7.9.1.1	Firestick interface	
Fig 7.9.1.2	Security interface	
Fig 7.9.3.0	Blynk app interface	
Fig 7.9.3.1	BLYNK monitoring using the app	
Fig 7.9.3.2	BLYNK web interface	

CHAPTER 1

INTRODUCTION

Over the last few decades, many researchers have focused on connecting everyday objects and weaving a web of interconnected devices. These can be physical devices, automobiles, embedded systems or everyday home appliances. The connectivity of these devices is enabled by the underlying technology known as the Internet of Things (IoT) [1]. These devices can communicate with one another and seamlessly collect and transfer data in between to adapt and reciprocate to dynamic situations. The IoT is gaining tremendous traction from the IT industry as well as individual enthusiasts alike. This is because microcontroller technologies have evolved that enable these complex infrastructures and are more accessible to average developers. The use of single board computers also refers to as System on Chip (SoC) in IoT-based systems, enables rapid interconnection of environmental sensors and decent computational capabilities along with vision sensors that make it easy to develop fairly in large applications. Advancements in machine learning algorithms for mobile and low-power devices also pave the way for intelligent devices in IoT systems. A typical home automation system connects heterogeneous devices, collects data, and decides based on the observed data from these sensors. The means of communication between these devices is typically Wi-Fi or Bluetooth and in some cases a cellular network (3G/4G/5G) [2]. We were focusing on how IoT can drastically improve the lives of disabled people by addressing limited access in many ways. We also spoke about how smart homes and home automation systems are rising in popularity, allowing people to control various aspects of their homes from a single place without getting up. However, for disabled people, this technology holds much more significance. It is not adding additional value but solving some of their biggest hurdles like being unable to access specific items or devices. Smart homes single-handedly allow disabled people to live more independently with an easy-access lifestyle. With the sophistication of IoT technology, it has become a possibility for disabled people to rely lesser on their caregivers

1.1 GENERAL BACKGROUND

The Internet of Things (IoT) refers to the network of physical objects or "things" embedded with sensors, software, and connectivity that allows them to collect and exchange data over the Internet. These objects can be anything from everyday devices like smartphones and smartwatches to complex systems such as industrial machinery and smart city infrastructure.

Key elements of IOT include:

1. Things /devices: these are the physical objects devices equipped with sensors, actuators, and connectivity that enable them to collect and transmit data. Examples include wearable devices, smart home appliances, industrial sensors, and vehicles.

2. Connectivity: IOT devices use various communication technologies to connect and transmit data. This can include Wi-Fi, cellular networks, Bluetooth, Zigbee, NFC (Near Field Communication), and more.

3. Data Collection: IoT devices gather data through built-in sensors, which can include temperature, humidity, pressure, motion, light, and many other types of sensors depending on the device's purpose.

4. Data Processing: Once collected, IoT devices can process the data locally or transmit it to the cloud for further analysis and storage. This allows for real-time monitoring, analytics, and decision-making based on the data received.

5. Cloud Computing: The cloud infrastructure plays a vital role in IoT by providing storage, processing power, and services for managing and analyzing vast amounts of data generated by IoT devices.

6. Data Analysis and Applications: The data collected from IoT devices can be analyzed using advanced analytics techniques, including machine learning and artificial intelligence, to extract meaningful insights and enable various applications. These applications can range from improving operational efficiency in industries, enabling smart homes and cities, enhancing healthcare services, optimizing energy consumption, and enhancing the overall quality of life.

7. Security and Privacy: IoT devices face security and privacy challenges due to the massive amounts of data they generate and transmit. Protecting data, ensuring device authentication,

and implementing secure communication protocols are crucial to maintaining the integrity and privacy of IoT systems.

IoT has a wide range of applications across industries, including healthcare, transportation, agriculture, manufacturing, energy management, retail, and more. It is expected to revolutionize various aspects of our lives by enabling smarter, more connected, and more efficient systems.

1.2 MOTIVATION

There are tons of pros in IOT based. The motivation behind the IOT stems from a variety of factors and potential benefits. Some key motivations driving IOT initiatives:

1. Efficiency and Automation: IoT aims to enhance efficiency and automate processes by connecting and integrating devices, systems, and data. By enabling real-time monitoring, remote control, and automation, IoT can streamline operations, reduce manual intervention, and optimize resource utilization.

2. Data-driven Insights: IoT generates vast amounts of data from connected devices. This data can be leveraged to gain valuable insights, enabling businesses and organizations to make informed decisions, optimize processes, and identify patterns and trends. Data analytics in IoT can lead to improved efficiency, predictive maintenance, and better resource allocation.

3. Improved Quality of Life: IoT has the potential to enhance the quality of life for individuals. Smart homes, for example, enable remote monitoring and control of appliances, lighting, security systems, and more. IoT applications in healthcare can enable remote patient monitoring, personalized treatments, and faster emergency responses. Additionally, IoT can contribute to creating smart cities with improved transportation systems, energy management, and environmental sustainability.

4. Cost Savings: IoT can help organizations reduce costs by optimizing resource usage, minimizing downtime through predictive maintenance, and improving energy efficiency. For example, smart grid systems can optimize energy distribution and reduce wastage, resulting in cost savings. Similarly, in industries, IoT-enabled asset tracking and inventory management can lead to cost reductions and streamlined operations.

5. Enhanced Safety and Security: IoT can improve safety and security across various domains. For instance, smart surveillance systems can detect and respond to security threats more effectively. In industries, IoT can monitor worker safety, ensure compliance with regulations, and detect hazardous conditions. Additionally, IoT can contribute to improving road safety through connected vehicle technologies.

6. Innovation and New Business Models: IoT opens up opportunities for innovation and the creation of new business models. It enables the development of new products, services, and experiences. For example, connected wearable devices have revolutionized the fitness and healthcare industries, while smart home devices have created new markets for home automation.

7. Environmental Impact: IoT can contribute to environmental sustainability by optimizing resource usage, reducing energy consumption, and improving waste management. Smart grids, smart buildings, precision agriculture, and environmental monitoring systems are examples of IoT applications that can help address environmental challenges.

It's important to note that while there are numerous motivations for IoT projects, each specific initiative may have its own unique goals and objectives based on the industry, context, and desired outcomes.

1.3 OBJECTIVE

The main objective of this project is to connect everybody’s objects to the internet and enable them to communicate with each other, gather data and perform tasks autonomously. The ultimate objective is to enhance, convenience, and productivity in various aspects of our lives.

Consider an example, it's a hot day and you are coming home from your office just to find your room has already been set at appropriate temperature, appropriate lights have been turned on and coffee is already brewed for you.

So how did this happen? When you're commuting home, your phone sends your approximate arrival time to your home network, combines it with today's weather gathered via sensors or Internet service and the time required by AC to cool your room to a particular temperature, the AC is turned on at a particular time and when you reach home, the room is already cool. All this was possible because all devices were communicating with each other and taking intelligent decisions. You could even manage the device state via your mobile app so as not to go near the physical switch or use an IR blaster remote.

Another example might be, your Fitbit sending your heart rate, sleeping data and other things to the cloud in real-time and some algorithm being able to predict your health condition and inform you to see a doctor etc before anything serious happens.

Thus IoT is changing how we look at everyday convenience and interact with things or rather how they interact with each other to make your life simpler.

CHAPTER 2

LITERATURE SURVEY

2.1 ENABLING AUTOMATION AND-EDGE INTELLIGENCE OVER RESOURCE CONST

Smart home applications are pervasive and have gained popularity due to the overwhelming use of the Internet of Things (IoT). The revolution in IoT technologies made homes more convenient, efficient and perhaps more secure. The need to advance smart home technology is necessary at this stage as IoT is abundantly used in the automation industry. However, most of the proposed solutions are lacking in certain key areas of the system i.e., high interoperability, data independence, privacy, and optimization in general. The use of machine learning algorithms requires high-end hardware and is usually deployed on servers, where computation is convenient, but at the cost of bandwidth. However, more recently edge AI-enabled systems are being proposed to shift the computation burden from the server side to the client side enabling smart devices. In this paper, we take advantage of edge AI-enabled technology to propose a fully featured cohesive system for smart homes based on IoT and edge computing. The proposed system makes use of industry standards adopted for fog computing as well as providing robust responses from connected IoT sensors in a typical smart home. The proposed system employs edge devices as a computational platform in terms of reducing energy costs and providing security, while remotely controlling all appliances behind a secure gateway. A case study of human fall detection is evaluated by a custom lightweight deep neural network architecture implemented over the edge device of the proposed framework. The case study was validated using the Le2i dataset. During the training, the early stopping threshold was achieved with 98% accuracy for the training set and 94% for the validation set. The model size of the network was 6.4 MB which is significantly lower than other networks with similar performance.

2.2 INTERACTING WITH A DIGITAL TWIN USING AMAZON ALEXA

The Digital Twin is an evolving concept with many facets and applications, for instance, engineering simulation, system control, and product-centric information management. This article focuses on the latter where literature uses the Product Avatar concept to refer to a product's digital counterpart. Such an avatar used to have one or more graphical interfaces to support user interactions with information about a product item. Over the last few years, voice user interfaces became more mature, and companies, such as Amazon and Google, used them to create digital assistants that support their users during tasks or by taking them over directly. This paper focuses on the hypothesis that a company could use a voice-enabled digital assistant to interact with item-level information. Our study used product tracking and tracing, and quality control in the production as a realistic application case. The design of the assistant is based on the information needs outlined in the Electronic Product Code Information Services (EPCIS) standard. We implemented this design in a small-scale demonstrator on an Echo Show 5 smart speaker with an integrated touch display and an embedded Amazon Alexa assistant. This paper concludes that significant technological barriers, such as low transcription accuracy for object identifier information and the handling of factory noise, remain. A significant non-technological barrier is the mistrust regarding the closed voice assistant technologies from companies, such as Amazon and Google. An approach to address the latter barrier is to use open technologies, such as the privacy-focused assistant Mycroft or Mozilla's transcription solution DeepSpeech.

2.3 AMAZON ALEXA TRAFFIC TRACES

The number of devices that make up the Internet of Things (IoT) has been increasing every year, including smart speakers such as Amazon Echo devices. These devices have become very popular around the world where users with a smart speaker are estimated to be about 83 million in 2020. However, there has also been great concern about how they can affect the privacy and security of their users [1]. Responding to voice commands requires devices to continuously listen for the corresponding wake word, with the privacy implications that this entails. Additionally, the interactions that users may have with the virtual assistant can reveal private information about the user. In this document, we publicly share two datasets that can help conduct privacy and security studies on the Amazon Echo Dot smart speaker. The included data contains 300.000 raw PCAP traces containing all the communications between the device and Amazon servers from 100 different voice commands in two different languages. The data can be used to train machine learning algorithms to find patterns that can characterize both, the voice commands and people using the device as well as Alexa as the device generating the traffic.

2.4 IOT-BASED ELECTRO-SYNTHESIS ECOSYSTEM

In this paper, we presented the electro-synthesis ecosystem using esp8266 with OTA compilation integrated, the MQTT Broker for the control system that would be installed on a Raspberry Pi and MQTT protocol for the wireless data transfer layer which provides the electrosynthesis machine to work 24/7 and with the AI that can be implemented, the whole ecosystem would be smart and the data-driven can be used in QC verification. This system is designed and simulated and the source codes are available via GitHub (https://github.com/iraniothome/ChemIoT).

2.5 ACCESS CONTROL AND SURVEILLANCE IN A SMART HOME

In the past years, smart home solutions have become more and more popular with the introduction of a high number of both Internet of Things (IoT) applications and smart devices. The home automation and security systems market is in continuous growth, traditional security systems are evolving fast, and more and more people choose Smart home solutions. In this paper, we propose two IoT-based systems in the context of Smart homes: qToggle for multiple home automation, and MotionEyeOS, a video surveillance OS for single-board computers. Most goggle devices are based on ESP8266/ESP8285 chips or Raspberry Pi boards and smart sensors, while MotionEye uses Raspberry Pi boards.

2.6 SMART IOT SURVEILLANCE MULTI-CAMERA MONITORING SYSTEM

The surveillance digital monitoring process market has changed aggressively over the last 15 years. Where the process was needed for people to secure their businesses by a simple surveillance camera system to give them that much-needed peace of mind. However, the existing monitoring systems are very expensive due to internal specific SDK configuration which was embedded in the camera itself. Some of them provide a solution such as a high-priced customized command centre that has several screens view which communicate with several cameras to monitor the cameras with a specific detection analysis module. In this paper, a cheaper solution to the monitoring system for existing surveillance cameras is introduced to overcome the solution. This system only implements open-source image processing methods to produce a monitoring system with customizable modules for video analytics (video content analysis) with the input of live video sources. This is to allow a wider range of camera models that can be used for this system. A real-time analysis module will assist in the use of the system to ease the user. The combined video stream and the module will help the user immensely for surveillance propose. Based on the result, the proposed system achieved a higher affordability level of up to 95% with 90% usability compared to existing products.

2.7 TRUSTBUILDER: A NON-REPUDIATION SCHEME FOR IOT CLOUD APPLICATIONS

The IoT cloud computing paradigm is emerging for IoT applications. In such a paradigm, how to guarantee non-repudiation service provisioning has attracted research efforts recently. While existing solutions could work for distributed IoT cloud applications, storage, computation, and economic cost are still a practical challenging concern. In this paper, we propose a new, cost-lower non-repudiation scheme for IoT cloud applications. The proposed scheme guarantees that neither the IoT client nor the cloud could re-repudiate a service enjoyment and provisioning. Specifically, the proposed scheme employs a blockchain to achieve non-repudiation. First, when the cloud provides a service, it encrypts the service, stores a cryptographic hash digest of the encrypted service on the blockchain, and then sends the encrypted service to the IoT client. Second, the IoT client needs to acknowledge the hash digest on the blockchain to ob- in the service. Third, the cloud sends the decryption key to the blockchain under the IoT client's public key to finish service provisioning. We show that the proposed scheme achieves non-repudiation fairly. We prototyped, evaluated, and open-sourced the proposed scheme. Experimental results confirmed the efficiency of the proposed scheme and the speedup compared with the state-of-the-art solution.

2.8 ACCURACY DETERMINATION USING DEEP LEARNING TECHNIQUE IN CLOUD-BASED IOT SENSOR ENVIRONMENT

devices. However, when connected to wireless connections, unlimited access to IoT gadgets poses potential risks. As it eases cost constraints on sensor nodes, the cloud service with IoT networks has received greater attention. In addition, the high complexity of the distribution and networking of IoT makes them vulnerable to attacks. Intrusion detection systems (IDSs) are selected to ensure the security of reliable information and operations. IDS successfully detects anomalies in complex network situations and guarantees network security. Deep Convolution Network (DCN) IDS have a slow learning curve and poor categorization precision. Deep Learning (DL) methods are often used in a wide range of safety data processing, imaging, and signal processing like Poor transfer learning ability, reusability of modules, and integration. To overcome the constraints of Machine Learning (ML) IDS is intended to provide a comprehensive mechanism to learn the detection mechanism for multi-cloud IoT environments. The proposed IDS approach increases training efficiencies while increasing detection accuracy. Experimental investigations of the proposed system using the considered database confirm that the performance of the proposed system is capable and in the range of acceptance relative to existing methods. Further, achieving detection capability, reliability, and accuracy of 97.51, 96.28, and 94.41% respectively are achieved.

2.9 SELF-SECURED MODEL FOR CLOUD-BASED IOT SYSTEMS

A difficult problem to solve concerns the secure installation and startup of devices connected to the Internet of Things (IoT) via the Internet. To provide additional value-added services, this article deals with the verified configuration of IoT devices in a secure manner using the Internet. Following a review of the safe self-configuration limitations imposed on IoT and Cloud technologies; offer a Cloud-based architecture that enables the communication between IoT devices and several federated Cloud services. Specifically discuss two situations, one cloud environment and federated cloud infrastructure interact with IoT devices, and handle unique issues. In addition, it provides many operational design features that take into account the truly open hardware and software products already on the market.

2.10 TAKING MQTT AND NODEMCU TO IOT: COMMUNICATION IN THE INTERNET OF THINGS

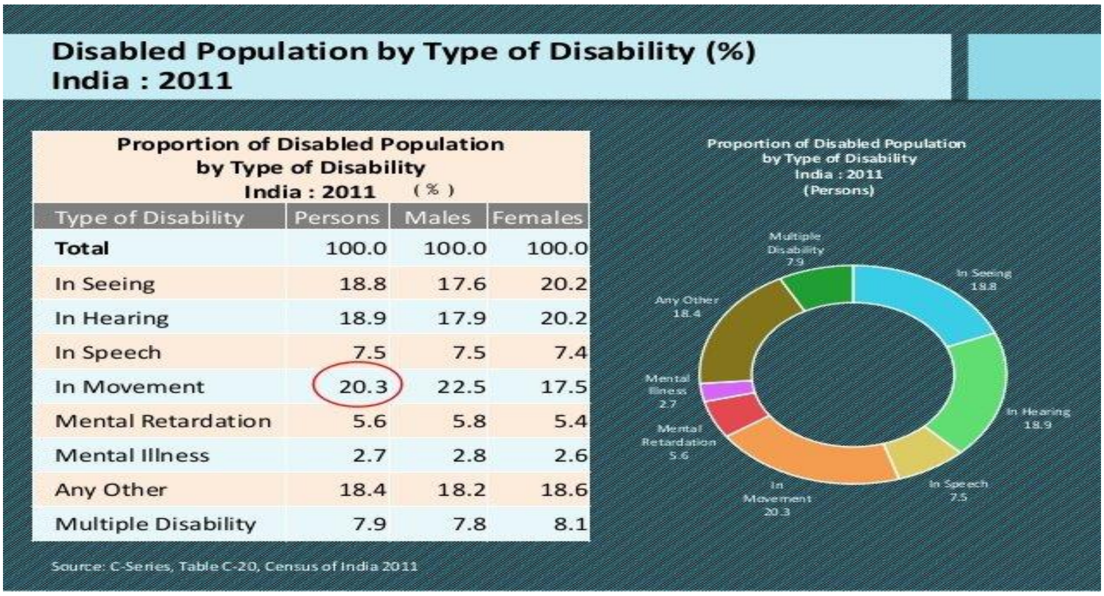
The Internet of Things (IoT) allows connection among devices using the Internet with the ability to gather and exchange data. These devices are usually attached with micro-controllers like Arduino, sensors, actuators and internet connectivity. In this context, Message Queuing Telemetry Transport protocol (MQTT) plays an important role to exchange the data or information between the devices in IoT without knowing the identities of each other. This paper presents different service models for communication in the Internet of Things(IoT). Model A presents the use of serial USB as a transmission medium while Model B uses the Message Queuing Telemetry Transport protocol (MQTT) which deploys a Wi-Fi module (ESP8266-12) to connect the system to the internet. For communication, the concept of publisher and subscriber is used. Messages are published or subscribed with the help of a broker or server. This agent is in charge of dispersing messages to intent clients depending on the choice of the topic of a message. The broker in MQTT is also called the server. Some brokers used in MQTT are: -Mosquitto, Adafruit, hiveMQ

CHAPTER 3

SYSTEM OVERVIEW

3.1 PROBLEM STATEMENT

Disabilities is an umbrella term, covering impairments, activity limitations, and participation restrictions. An impairment is a problem in body function or structure, An activity limitation is a difficulty encountered by an individual in executing a task or action, and A participation restriction is a problem experienced by an individual in involvement in life situations. As per Census 2011, in India, out of the total population of 121 crores, about 2.68 Cr persons are 'Disabled' (2.21%of the total population) Out of 2.68 crores, 1.5 crores are males and 1.18 crore are females Majority (69%) of the disabled population resided in rural areas.



Proportion of Disabled Population by Type of Disability India : 2011 (%)

Type of Disability	Persons	Males	Females
Total	100.0	100.0	100.0
In Seeing	18.8	17.6	20.2
In Hearing	18.9	17.9	20.2
In Speech	7.5	7.5	7.4
In Movement	20.3	22.5	17.5
Mental Retardation	5.6	5.8	5.4
Mental Illness	2.7	2.8	2.6
Any Other	18.4	18.2	18.6
Multiple Disability	7.9	7.8	8.1

Fig 3.1 Disability by Population Chart in India

People with disabilities often face various challenges in their day-to-day lives, hindering their ability to perform routine tasks independently and limiting their overall quality of life. It is crucial to explore innovative solutions that leverage the power of the Internet of Things (IoT) to address these challenges and empower individuals with disabilities. The goal of this project is to develop an IoT-based system that can provide assistance, accessibility, and convenience for people with disabilities, enabling them to navigate their environment, communicate effectively, and perform daily activities more efficiently. People with disabilities often face various challenges in their day-to-day lives, hindering their ability to perform routine tasks independently and limiting their overall quality of life. It is crucial to explore innovative solutions that leverage the power of the Internet of Things (IoT) to address these challenges and empower individuals with disabilities. The goal of this project is to develop an IoT-based system that can provide assistance, accessibility, and convenience for people with disabilities, enabling them to navigate their environment, communicate effectively, and perform daily activities more efficiently. People with disabilities often face various challenges in their day-to-day lives, hindering their ability to perform routine tasks independently and limiting their overall quality of life. It is crucial to explore innovative solutions that leverage the power of the Internet of Things (IoT) to address these challenges and empower individuals with disabilities. The goal of this project is to develop an IoT-based system that can provide assistance, accessibility, and convenience for people with disabilities, enabling them to navigate their environment, communicate effectively, and perform daily activities more efficiently.

3.2 SYSTEM OUTLINE

The Internet of Things (IoT) has the potential to transform the lives of persons with disabilities, our societies, and industries. Today's IoT devices and services are increasingly accessible to persons with disabilities; some IoT technologies are specifically designed for persons with disabilities, and others are repurposed by persons with disabilities (see figure below). The IoT and its associated data collection are producing accessibility-related advances, ranging from smart home devices to self-driving cars. IoT-based services are also empowering persons with disabilities to participate more fully and autonomously in everyday life by reducing some needs for human intermediaries or accommodations. Data derived from persons with disabilities use of IoT devices and services can provide insights into the challenges or opportunities experienced by persons with disabilities while using IoT devices. These insights can be used to enhance existing IoT products or develop new ones.

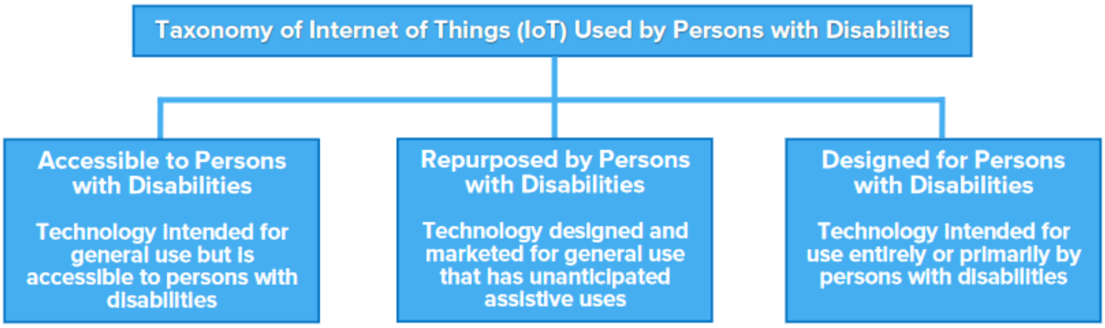


Fig 3.2Taxonomy of IoT used by disabled people

Despite the potential benefits of IoT devices and services for persons with disabilities, unique privacy risks and challenges can be raised by the collection, use, and sharing of data about users. Depending on the circumstances, privacy can be enhanced or diminished by technologies, creating tension between privacy gains and losses.

How people balance those tensions depends upon context—including how the service or device is used, who is using it, and individuals' preferences and values. Some members of the disability community may consider benefits and privacy risks differently than other communities. This consideration—evaluating the ways IoT devices and services allow persons with disabilities to enhance their privacy vs. creating privacy risks via data collection—deserves more nuanced consideration and engagement

3.3 PROPOSED METHODOLOGY

Amazon Echo and Google Home have been one of the most publicized uses of IoT in people's homes. To use utilities and basic services within a house, people have to no longer depend on switches and conventional physical methods of interaction. For example, IoT devices allow a visually impaired user to change the heat settings without needing to program a controller. These devices work by using a voice recognition system, without any kind of special setup. For people suffering from paralysis or those who are completely bedridden, such technologies are no less than a boon as they perform functions like unlocking a door without In most cases, people with disabilities require constant monitoring which can often be challenging and demanding for them. Michael J Fox Foundation uses IoT devices to monitor hundreds of people with disabilities and long-term illnesses. They gather millions of diverse data points that hold a clue to cure a disease. The data collected is so enormous that it can be used not only for those who are ill and challenged but also to evolve newer models for developing preventive measures for any new kind of disease as well. IoT will break the accessibility barriers. IoT-enabled smart environments will create an enabling system that embodies inclusiveness, a smart ecosystem that helps challenged people to live their lives freely.

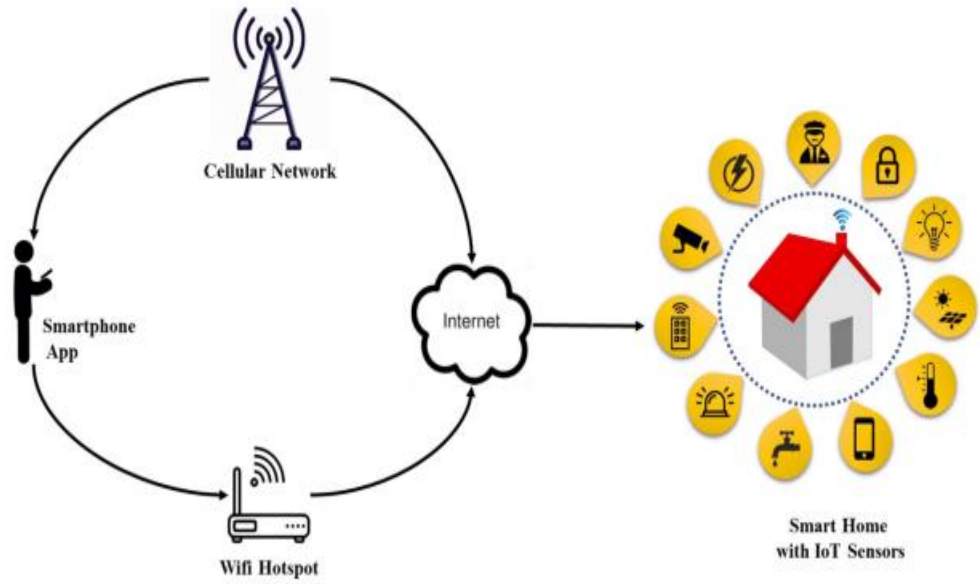


Fig 3.3 Proposed Methodology

In this project we use the NodeMCU, it is a development board that is a powerful solution to program microcontrollers and be part of the Internet of Things (IoT). The NodeMCU development board, based on ESP8266EX, is a cute module with a microcontroller, integrated Wi-Fi receiver, and transmitter. NodeMCU supports several programming languages; hence, it is very easy to upload programs from any computer over a micro-USB port. I have been playing with the NodeMCU for quite a while now and, I have to say, it is a lot more fun than the other available IoT modules. When it comes to prototyping — just another perfect, relatively cheap, easy-to-learn, and user-friendly minuscule magic module.

Amazon Alexa Automation, in general, is a big victory for the regular consumer in terms of convenience. However, it brings a much bigger convenience and independence factor to people with disabilities, especially anyone who is blind, in a wheelchair, paraplegic, bedridden because of a spinal cord injury, or doesn’t have good motor skills. It saves them a lot of time and energy by not making them interact with other devices that they may not have skills for or are unable to use because of various disabilities. The only device they interact with is Echo, through voice, and it provides them with the results and information they are looking for instantly, and thus, saves them a lot of trouble. A person in a wheelchair doesn’t have to try to reach a light switch that’s in an awkward corner of a room, a person with not good motor skills doesn’t have to flip through pages or operate an e-reader to read their books, and a blind person doesn’t need to navigate a website on an electronic device to order a pizza anymore. Automation through the Internet of Things doesn’t only have to be at home. A device like Alexa can be installed by an employer at work as well so that employees with disabilities can be more comfortable in their work environments. A device like Echo is not expensive, and it just makes the ability to provide accommodations an inherent part of the system, and not an afterthought.

This is just the beginning though. The kind of features Amazon keeps adding to Echo is mind-boggling, and very exciting to say the least.

We connect security needs with Alexa’s help of Internet of Things security solutions, so building managers and operators can monitor and secure their facilities more effectively. Security surveillance systems and intelligent CCTV systems have improved visibility for organisations and businesses to safeguard their assets, buildings and workers. A smart camera solution can now detect and interpret incidents autonomously. Robustly has the perfect solution to get your CCTV devices connected. Our 3G/4G routers provide secure and remote access to your CCTV network and are customisable to suit your exact needs. It’s common to find a connected IoT camera in nearly every smart application, from doorbells and elevators to building management systems. Operating within a network of motion sensors, smoke detectors, and alarms, smart CCTV cameras can alert an operator to an incident and even act autonomously to request help from emergency responders. Smart security systems are in high demand across a range of spaces, including commercial buildings, factories, warehouses and public transport, and are being leveraged in various ways

3.4 HARDWARE IMPLEMENTATION

The hardware utilized in IoT systems includes devices for a remote dashboard, devices for control, servers, a routing or bridge device, and sensors. These devices manage key tasks and functions such as system activation, action specifications, security, communication, and detection to support-specific goals and actions.

3.4.1 IOT − SENSORS

the most important hardware in IoT might be its sensors. These devices consist of energy modules, power management modules, RF modules, and sensing modules. RF modules manage communications through their signal processing, WiFi, ZigBee, Bluetooth, radio transceiver, duplexer, and BAW The sensing module manages to sense through assorted active and passive measurement devices.

S.No	Devices	
1.	Accelerometers	temperature sensors
2.	Magnetometers	proximity sensors
3.	Gyroscopes	image sensors
4.	acoustic sensors	light sensors
5.	pressure sensors	gas RFID sensors
6.	humidity sensors	micro flow sensors

Fig 3.4.1List of some of the measurement devices used in IoT

3.4.2 STANDARD DEVICES

The desktop, tablet, and cell phone remain integral parts of IoT as the command centre and remotes. The desktop provides the user with the highest level of control over the system and its settings. The tablet provides access to the key features of the system in a way resembling the desktop and also acts as a remote. The cell phone allows some essential settings modification and also provides remote functionality. Other key connected devices include standard network devices like routers and switches.

3.5 INTERNET OF THINGS - SOFTWARE

IoT software addresses its key areas of networking and action through platforms, embedded systems, partner systems, and middleware. These individual and master applications are responsible for data collection, device integration, real-time analytics, and application and process extension within the IoT network. They exploit integration with critical business systems (e.g., ordering systems, robotics, scheduling, and more) in the execution of related tasks.

3.5.1 DATA COLLECTION

This software manages sensing, measurements, light data filtering, light data security, and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine-to-machine networks. Then it collects data from multiple devices and distributes it by settings. It also works in reverse by distributing data over devices. The system eventually transmits all collected data to a central server.

3.5.2 DEVICE INTEGRATION

Software-supporting integration binds (dependent relationships) all system devices to create the body of the IoT system. It ensures the necessary cooperation and stable networking between devices. These applications are the defining software technology of the IoT network because, without them, it is not an IoT system. They manage the various applications, protocols, and limitations of each device to allow communication.

3.5.3 REAL-TIME ANALYTICS

These applications take data or input from various devices and convert it into viable actions or clear patterns for human analysis. They analyze information based on various settings and designs to perform automation-related tasks or provide the data required by the industry.

3.5.4 APPLICATION AND PROCESS EXTENSION

These applications extend the reach of existing systems and software to allow a wider, more effective system. They integrate predefined devices for specific purposes such as allowing certain mobile devices or engineering instruments access. It supports improved productivity and more accurate data collection.

3.6 INTERNET OF THINGS - TECHNOLOGY AND PROTOCOLS

IoT primarily exploits standard protocols and networking technologies. However, the major enabling technologies and protocols of IoT are RFID, NFC, low-energy Bluetooth, low-energy wireless, low-energy radio protocols, LTE-A, and WiFi-Direct. These technologies support the specific networking functionality needed in an IoT system in contrast to a standard uniform network of common systems.

3.6.1 NFC AND RFID

RFID (radio-frequency identification) and NFC (near-field communication) provide simple, low-energy, and versatile options for identity and access tokens, connection bootstrapping, and payments.RFID technology employs 2-way radio transmitter-receivers to identify and track tags associated with objects. NFC consists of communication protocols for electronic devices, typically mobile devices and standard devices.

3.6.2 LOW-ENERGY BLUETOOTH

This technology supports the low-power, long-use need of IoT function while exploiting a standard technology with native support across systems.

3.6.3 LOW-ENERGY WIRELESS

This technology replaces the most power-hungry aspect of an IoT system. Though sensors and other elements can power down over long periods, communication links (i.e., wireless) must remain in listening mode. Low-energy wireless not only reduces consumption but also extends the life of the device through less use.

3.6.4 RADIO PROTOCOLS

ZigBee, Z-Wave, and Thread are radio protocols for creating low-rate private area networks. These technologies are low-power but offer high throughput, unlike many similar options. This increases the power of small local device networks without the typical costs.

3.6.5 LTE-A

LTE-A, or LTE Advanced, delivers an important upgrade to LTE technology by increasing not only its coverage but also reducing its latency and raising its throughput. It gives IoT tremendous power by expanding its range, with its most significant applications being vehicles, UAVs, and similar communication.

3.6.6 WIFI-DIRECT

Wi-Fi-Direct eliminates the need for an access point. It allows P2P (peer-to-peer) connections with the speed of Wi-Fi, but with lower latency. Wi-Fi-Direct eliminates an element of a network that often bogs it down, and it does not compromise on speed or throughput.

3.7 FEATURE EXTRACTION

This project contains many features for helping disabled people. the basic features are listed below.

3.7.1 VOICE COMMAND

Alexa is built based on natural language processing (NLP), a procedure of converting speech into words, sounds, and ideas. Amazon records your words. Indeed, interpreting sounds takes up a lot of computational power, the recording of your speech is sent to Amazon’s servers to be analysed more efficiently. Computational power: refers to the speed at that instructions are carried out and is normally expressed in terms of kilo flops, megaflops, etc. Amazon breaks down your “orders” into individual sounds. It then consults a database containing various words’ pronunciations to find which words most closely correspond to the combination of individual sounds. It then identifies important words to make sense of the tasks and carry out corresponding functions. For instance, if Alexa notices words like “sport” or “basketball”, it would open the sports app. Amazon's servers send the information back to your device and Alexa may speak. If Alexa needs to say anything back, it would go through the same process described above, but in reverse order (source)

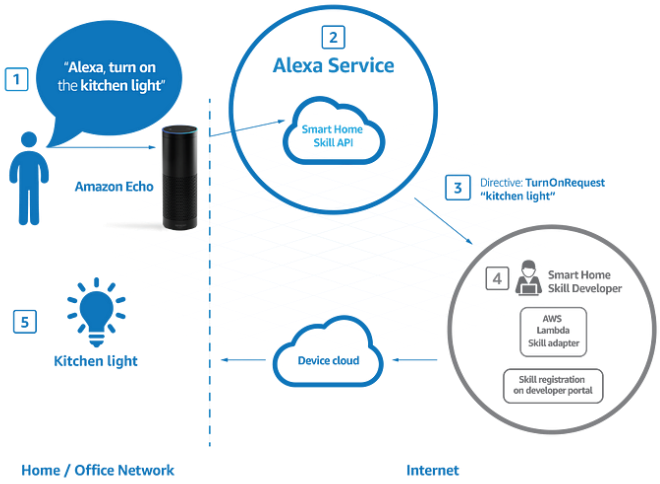


Fig 3.7.1 Alexa command interface

3.7.2 MODULE WISE ARCHETECTURE

The IoT module introduces the Internet of Things, which extends Internet connectivity from computers and related devices to other physical devices or common objects and leverages technologies such as embedded systems, wireless sensors, and automation.

3.7.3 ECONOMICAL

decrease the cost of excessive travel. to and fro, for either patients or doctors could be greatly reduced, resulting in cost savings as IoT devices can help in real-time patient monitoring. and better than human labour.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONALITIES OF THE PROPOSED SYSTEM

Fulfilling all needs of the patient including physical and mental needs. Medical needs include water and oxygen supply and basic needs include ventilation control, and security monitoring .and an external person (Doctor or relative) can monitor the patient's health status and prescribe the treatment etc. And Roban system full fill all entertainment needs including all video streaming platforms and audio platforms.

4.2 MODULES

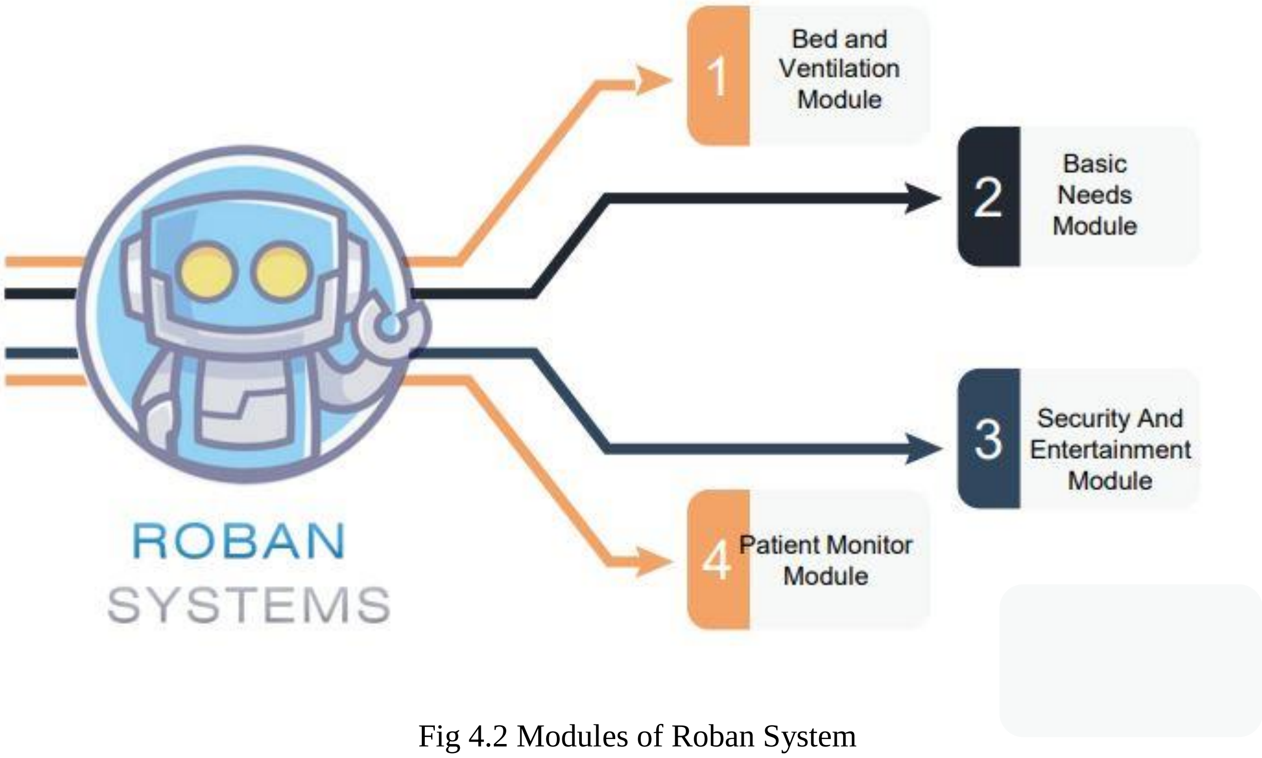


Fig 4.2 Modules of Roban System

Roban system consists of 4 modules .it consist Bed and ventilation Basic needs module Security and entertainment module patient monitor module

4.2.1 BED AND VENTILATION MODULE

The bed and ventilation module is used to adjust the bed position and adjust the ventilation by sliding the curtains. Disabled people can adjust the bed positions by themselves through voice commands, so they can acquire their comfortable positions. By controlling the ventilation system by voice command they can adjust the natural lighting and ventilation.

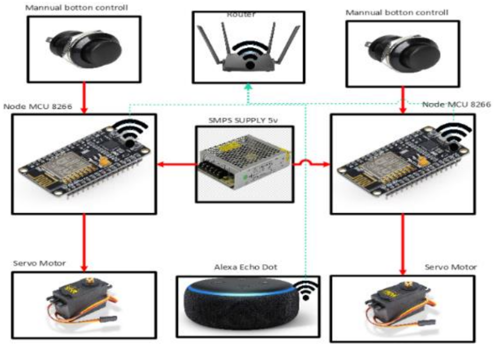


Fig 4.2.1 Bed And Ventilation Module

4.2.2 BASIC NEEDS MODULE

The basic need module consists of an Artificial ventilation system like fans, or exhaust fans, a lighting system like AC and DC lights, warm lights and a pest control system. Disabled people can adjust the bed positions by themselves through voice commands so they can full fill their needs using the IoT

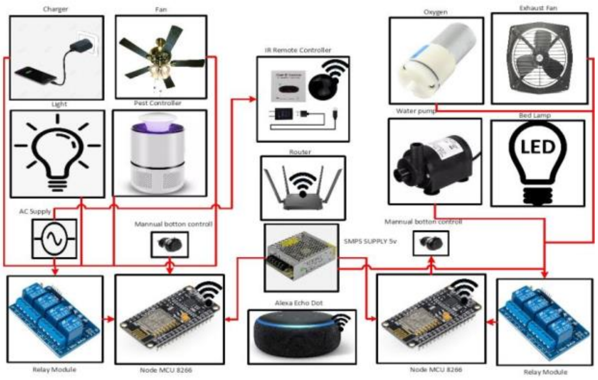


Fig 4.2.2 Basic Needs Module

4.2.3 SECURITY AND ENTERTAINMENT MODULE

Disabled people are seen in a closed environment so there is a chance of mental depression. So we connect entertainment like TV, social media, and streaming platforms in this module Disabled people can use them through voice commands. And security is also essential for them so we connected a security camera we can use this camera to monitor the patient's external and internal environment.

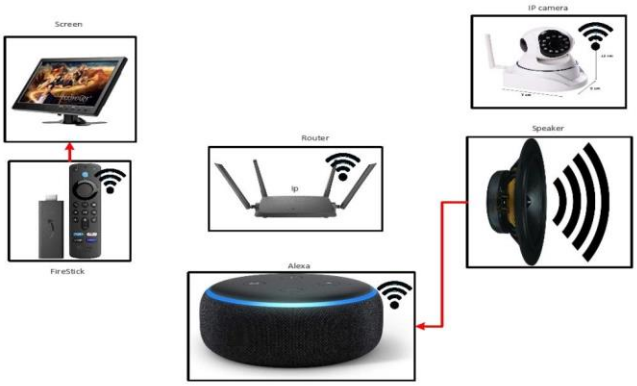


Fig 4.2.3 Security And Entertainment Module

4.2.4 PATIENT MONITOR MODULE

This module consists of several sensors that are connected over IoT, and by that sensor, we can monitor the patient's health and make health plans and it can also alert the authorities when there are any emergencies happen. We can implement patient monitoring like temp, oxygen, pulse, etc. and monitor the environment of the patient.

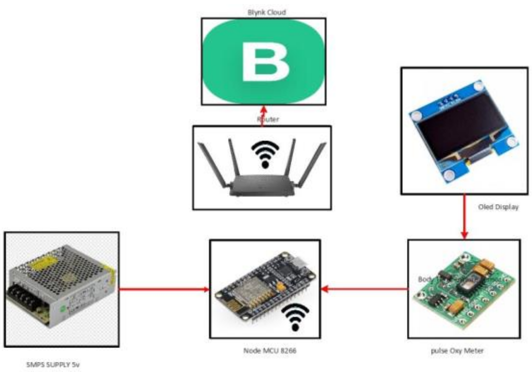


Fig 4.2.4 Patient Monitor Module

4.3 REQUIREMENTS

The requirements to fulfil the project include hardware and software requirements. The main requirements are listed below.

4.3.1 SYSTEM CONFIGURATION

The system must be full fill the configuration of 4GB RAM and a minimum operating system of Windows 7 or more.

4.3.2 ARDUINO IDE

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux**.** It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment. The program or code written in the Arduino IDE is often called sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.'The Arduino IDE will appear as:

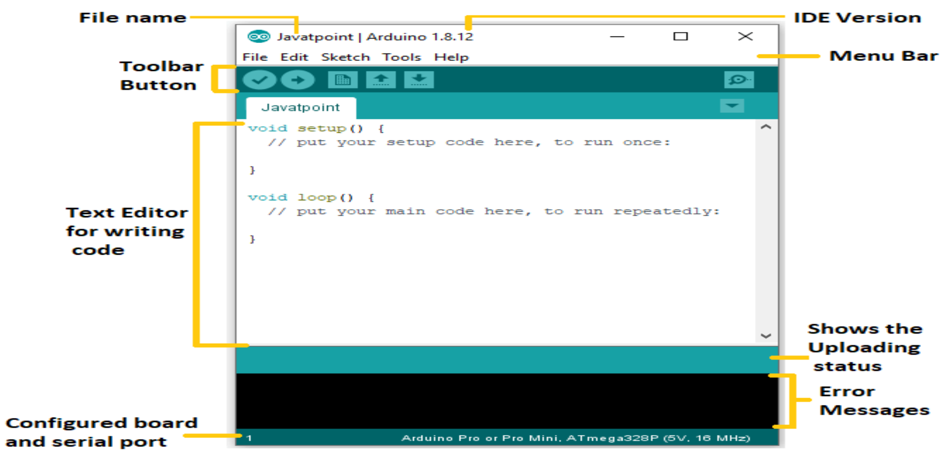


Fig 4.3.2 Arduino Ide

4.3.3 NODEMCU 8266

The NodeMCU development board is a powerful solution to program microcontrollers and be part of the Internet of Things (IoT). The NodeMCU development board, based on ESP8266EX, is a cute module with a microcontroller, integrated Wi-Fi receiver, and transmitter. NodeMCU supports several programming languages; hence, it is very easy to upload programs from any computer over a micro-USB port. I have been playing with the NodeMCU for quite a while now and, I have to say, it is a lot more fun than the other available IoT modules. When it comes to prototyping — just another perfect, relatively cheap, easy-to-learn, and user-friendly minuscule magic module NodeMCU V1.0 The first generation of the NodeMCU Development Kit had the version number V0.9. The second generation had version number V1.0, and this newer version used ESP-12E (not ESP-12), which comes with 4 MB of flash memory. The new version comes with the CP2102 serial chip (not CH340) and it works well and without a hassle. However, I noticed that, when it comes to many Chinese online vendors, there's a blurring mix-up of generation and version names. The common name used by them for version 1.0 is V2, and most of those V2 boards are fabricated by or at least marked with "Amica" Technically, it's "NodeMCU V1.0 with ESP-12E module

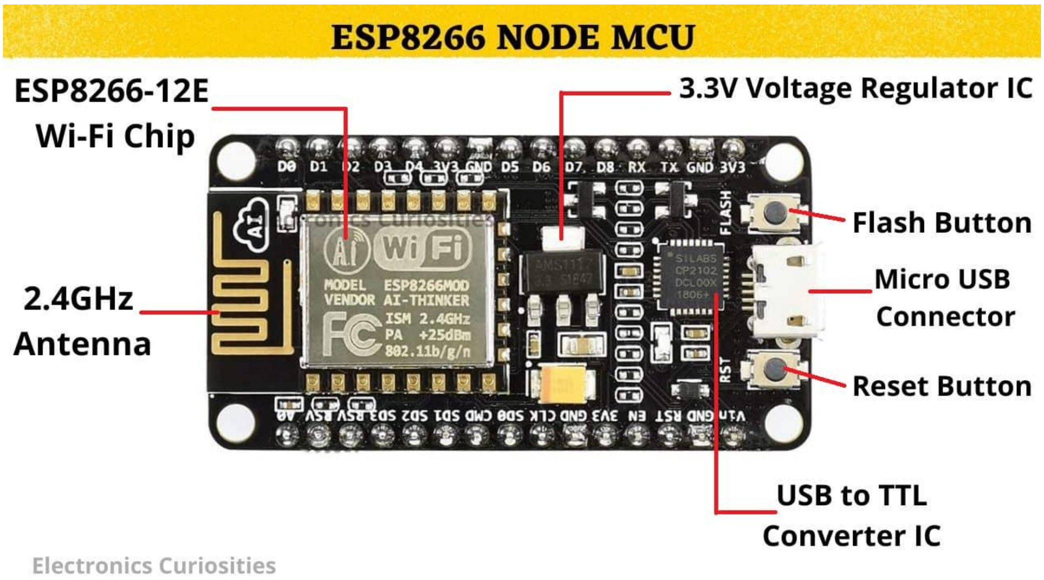


Fig 4.3.3 Nodemcu 8266

4.3.4 RELAY

A relay module is a relay that's been mounted on a board with other components to provide isolation and protection. This makes them easier to use in a variety of applications. The use of relay module devices offers a simple and convenient way to control electrical equipment systems remotely.

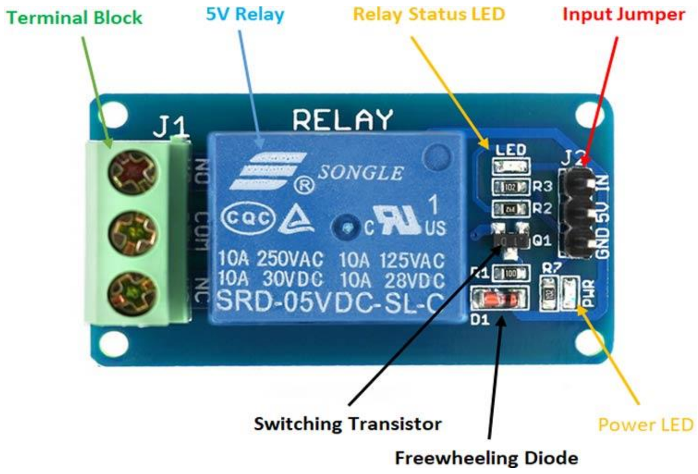


Fig 4.3.4 Relay

4.3.5 PCB

PCB is an acronym for *printed circuit board*. It is a board that has lines and pads that connect various points. In the picture above, some traces electrically connect the various connectors and components. A PCB allows signals and power to be routed between physical devices. SoldSoldierthe metal that makes the electrical connections between the surface of the PCB and the electronic components. Being metal, solder also serves as a strong mechanical adhesive.

4.3.6 PUSH BUTTON

A push-button (also spelt pushbutton) or simply button is a simple switch mechanism to control some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal The surface is usually flat or shaped to accommodate the human finger or hand, to be easily depressed or pushed. Buttons are most often biased switches, although many un-biased buttons (due to their physical nature) still require a spring to return to their un-pushed

Fig 4.3.6 Push Button

4.3.7 SERVO MOTOR

A servo motor is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a servo mechanism. If the motor is powered by a DC power supply then it is called a DC servo motor, and if it is an AC-powered motor then it is called an AC servo motor. For this tutorial, we will be discussing only the DC servo motor working. Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics. A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy cars, RC helicopters and planes, Robotics, etc.

Fig 4.3.7DSC 3019 Servo Motor

4.3.8 SMPS

A switched-mode power supply (switching-mode power supply, switch-mode power supply, switched power supply, SMPS, or switcher) is an electronic power supply that incorporates a switching regulator to convert electrical power efficiently.

4.3.9 IR BLASTER

That device is a universal remote control. It’s a smart IR blaster (infrared) remote which works with Alexa as well as Google Assistant devices. In layman’s terms, this means you can control the TV or any system wirelessly using light signals.

Fig 4.3.9 IR Blaster

4.3.10 AMAZON FIRE TV STICK

Amazon Fire Tv stick is a media streaming device which lets you stream video, install apps, play music etc. on your tv. is not built on the Android platform and it converts your normal tv to a smart tv. You can install Android apps on the device and play games and enjoy music too. It's an HDMI stick and all you need is an HDMI port on your television and you're good to go. Fire OS is the operating system that runs Amazon's Fire TV and tablets. Fire OS is a fork of Android, so if your app runs on Android, it will most likely run on Amazon's Fire devices too.

Fig 4.3.10 Amazon Fire Tv Stick

4.3.11 DAHUA IMOU RANGER 2

1080P H.265 Wi-Fi Pan & Tilt Camera With 1080P Full HD live monitoring and 0~355° pan & -5~80°tilt features, Ranger 2 ensures every corner of your home is completely covered. Human Detection quickly finds human targets in images and immediately sends a notification to you. Privacy Mode helps protect your privacy when you are home, and the Smart Tracking function is also inbuilt,

Fig 4.3.11 IP camera

4.3.12 HDMI MONITOR

An HDMI monitor is a monitor that is connected through a high-definition multimedia interface (HDMI) port. The connection is generally the first obstacle you encounter when buying a monitor. Most LCD monitors use VGA and HDMI ports, with the latter being somewhat more important. The camera follows moving objects when it detects motion.

Fig 4.3.12 HDMI Monitor

4.3.13 ALEXA ECHO

Alexa is Amazon's voice AI. Alexa lives in the cloud and is happy to help anywhere there's internet access and a device that can connect to Alexa. Making Alexa part of your day is as simple as asking a question. Alexa can play your favourite song, read the latest headlines, dim the lights in your living room, and more.

Fig 4.3.13 Alexa Echo

4.3.14 MAX3010

MAX30100 is an integrated pulse oximeter and heart-rate monitor sensor solution. It’s an optical sensor that derives its readings from emitting two wavelengths of light from two LEDs – a red and an infrared one – then measuring the absorbance of pulsing blood through a photodetector. This particular LED colour combination is optimized for reading the data through the tip of one’s finger. It is fully configurable through software registers and the digital output data is stored in a 16-deep FIFO within the device. It has an I2C digital interface to communicate with a host microcontroller.

Fig 4.3.14 MAX3010

4.3.15 OLED DISPLAY MODULE

2.44 cm (0.96 Inch) I2C/IIC 128x64 OLED Display Module 4 Pin - White Color is a precise small, White OLED module which can be interfaced with any microcontroller using I2C/IIC protocol. It is having a resolution of 128x64.OLED (Organic Light-Emitting Diode) is a self-light-emitting technology composed of a thin, multi-layered organic film placed between an anode and cathode. In contrast to LCD technology, OLED does not require a backlight. OLED possesses high application potential for virtually all types of displays and is regarded as the ultimate technology for the next generation of flat-panel displays.OLEDs' basic structure consists of organic materials positioned between the cathode and the anode, which is composed of electric conductive transparent Indium Tin Oxide (ITO). The organic materials compose a multi-layered thin film, which includes the Hole Transporting Layer (HTL), Emission Layer (EML) and Electron Transporting Layer (ETL). By applying the appropriate electric voltage, holes and electrons are injected into the EML from the anode and the cathode, respectively. The holes and electrons combine inside the EML to form exactions, after which electro-luminescence occurs. The transfer material, emission layer material and choice of electrode are the key factors that determine the quality of OLED components.

Fig 4.3.15 OLED Display Module

4.3.16 BLYNK IOT

Blynk is an IoT platform for iOS or Android smartphones that are used to control Arduino, Raspberry Pi and NodeMCU via the Internet. This application is used to create a graphical interface or human-machine interface (HMI) by compiling and providing the appropriate address on the available widgets.

4.3.17 ROUTER

A router is a device that provides Wi-Fi and is typically connected to a modem. It sends information from the internet to personal devices like computers, phones, and tablets. These internet-connected devices in your home make up your Local Area Network (LAN).

4.3.18 AMAZON ALEXA APP

also known simply as Alexa, is a virtual assistant technology largely based on a Polish speech synthesizer named Ivona, bought by Amazon in 2013. It was first used in the Amazon Echo smart speaker and the Echo Dot, Echo Studio and Amazon Tap speakers developed by Amazon Lab126.

4.3.19 PUMP/VALVE

Reciprocating pumps are a type of positive displacement pump, consisting of a: Fluid-End (consisting of two chambers, suction and discharge, separated by spring-loaded valves) and a Suction Inlet (where fluid flows from piping through a valve into the first chamber, aka suction chamber). Valves regulate and control flow and pressure in pumping systems. They also play an important role in site safety. Understanding the types of valves and how they work can help end users select the right valves for their application’s use.

Fig 4.3.19 Different Pumps And Solenoid Valve

CHAPTER 5

SYSTEM DESIGN

5.1 DESIGN METHODOLOGY

The Internet of Things design is a special approach to building every part of the IoT architecture. Usual methods to manage, store, and transfer data are not enough to handle the data within the IoT ecosystem (read about that in the article IoT and Big Data). The development approach must be adopted to the peculiarities of the Internet of Things. A single network can count thousands or more connected gadgets, user interfaces in IoT require new design practices, the experience of IoT users differs from the experience of web users, and interactions and contact points between users and IoT systems are unique.

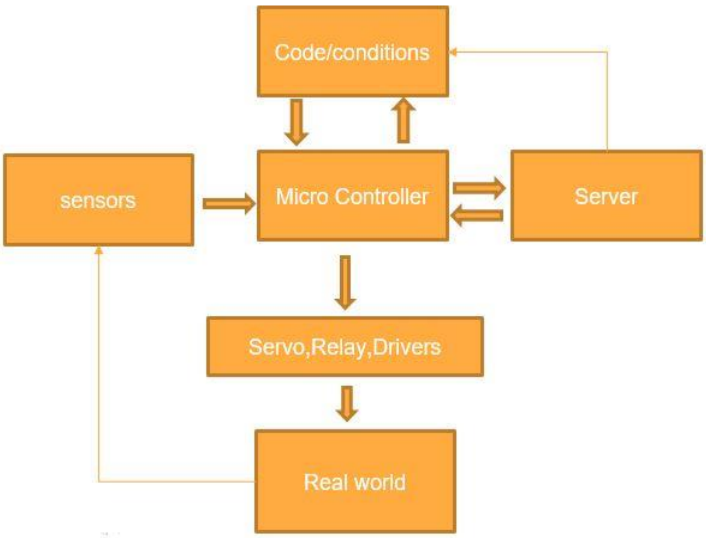


Fig 5.1 Design Methodology

5.2 SYSTEM ARCHITECTURE

Internet of Things (IoT) technology has a wide variety of applications and the use of the Internet of Things is growing so faster. Depending upon different application areas of the Internet of Things, it works accordingly as per it has been designed/developed. But it has not a standard defined architecture of working which is strictly followed universally. The architecture of IoT depends upon its functionality and implementation in different sectors. Still, there is a basic process flow based on which IoT is built.

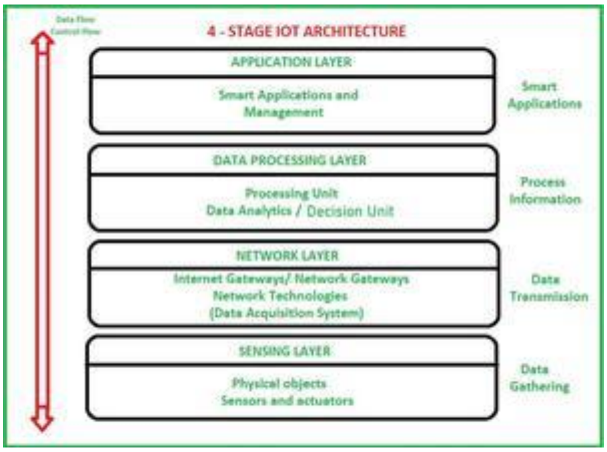


Fig 5.2 System Architecture

5.2.1 SENSING LAYER

Sensors, actuators, and devices are present in this Sensing layer. These Sensors or Actuators accept data(physical/environmental parameters), process data, and emit data over the network.

5.2.2 NETWORK LAYER

Internet/Network gateways and Data Acquisition Systems (DAS) are present in this layer. DAS performs data aggregation and conversion functions Collecting data and aggregating data then converting analogue data of sensors to digital data etc.). Advanced gateways which mainly open up a connection between Sensor networks and the Internet also perform many basic gateway functionalities like malware protection and filtering sometimes decision-making based on inputted data and data management services, etc.

5.2.3 DATA PROCESSING LAYER

This is the processing unit of the IoT ecosystem. Here data is analyzed and pre-processed before sending it to the data centre from where data is accessed by software applications often termed business applications where data is monitored and managed and further actions are also prepared. So here Edge IT or edge analytics comes into the picture.

5.2.4 APPLICATION LAYER

This is the last layer of the 4 stages of IoT architecture. Data centres or cloud is the management stage of data where data is managed and is used by end-user applications like agriculture, health care, aerospace, farming, defence, etc.

5.3 PROPOSED FRAMEWORK

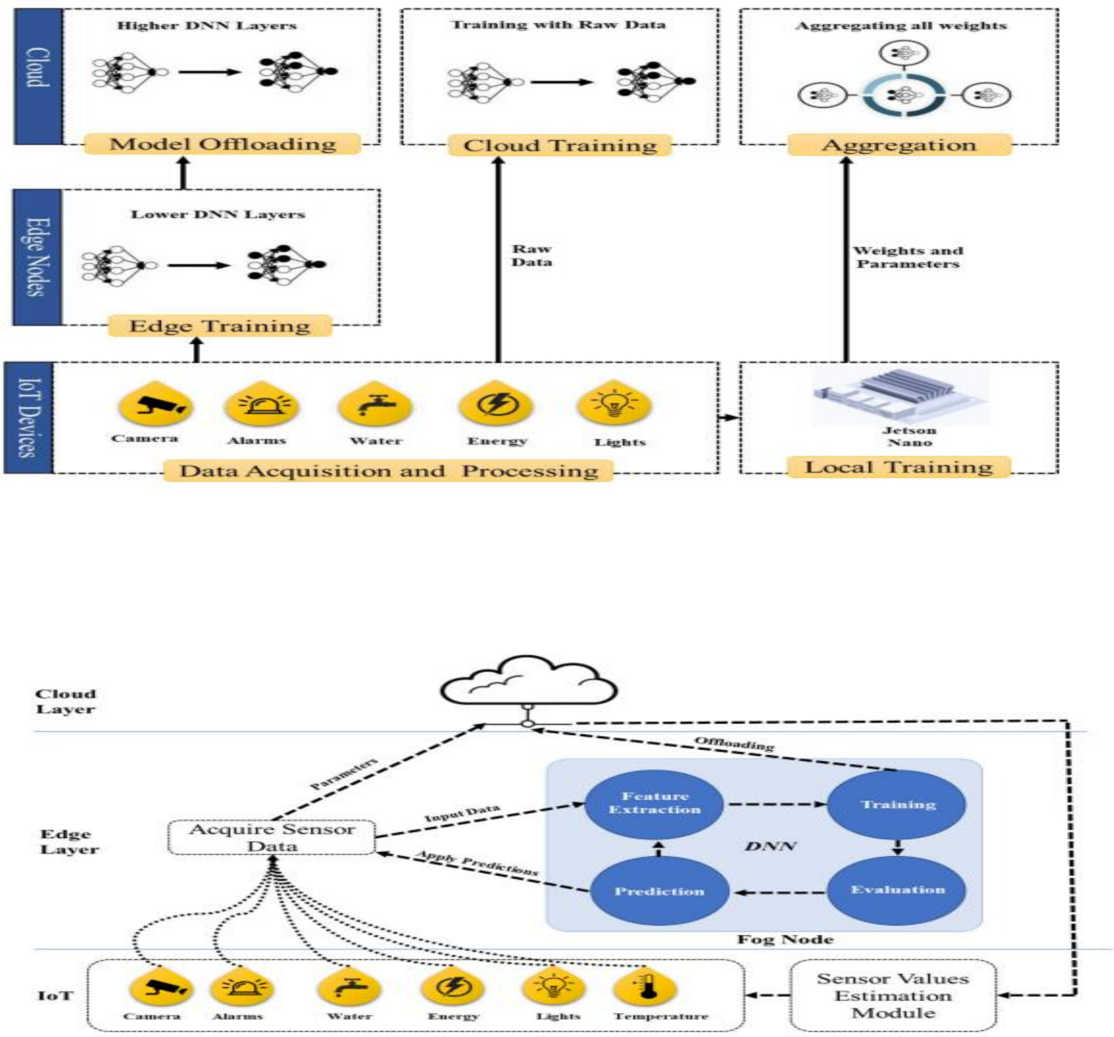


Fig 5.3 Proposed Framework

Home automation, security, safety, privacy, and energy consumption is an emerging field of technology and can be implemented with different microcontrollers and processors i.e., Desktop PC, laptops, Raspberry Pi, Arduino, etc., All these microcontroller and processors have their pros and cons, but Raspberry Pi is more versatile and efficient option compared to other units. The smart home concepts can have many components including 1) Sensors, 2) Gateways, 3) Protocols, 4) Firmware and 5) Cloud-based databases. The main focus of bringing AI to the smart home is to automate certain processes that need more than thresholds and pre-defined configurations. For example, one such task is fall detection. The process of integrating AI in the smart home is more natural and requires a significant amount of computational overhead. Today, most of the data generated by IoT devices resides outside the scope of the cloud so it is more natural to bring the machine learning capabilities closer to the source of the data and reduce the amount of bandwidth it consumes by transferring the entire dataset for each training session. Therefore, we leverage an approach called mobile edge compcomputinere the collaborative nature of the architecture makes it easy to train data locally on the device, and then send data to the higher-level layers of the DNN to train it on the edge server. The remaining data is relayed to the cloud where the resource-intensive tasks are offloaded and the remaining mode is trained on the cloud d.the architecture derived from federated learning proposed by Konecˇny´ et al. The authors proposed a method called on deon-demand learning co-inference with edge synergy. The authors argue that using the right sizing approach for DNN layers, the early exit point of the layer must be maintained to offload computationally intense tasks to the cloud from the compacomparativeurce constraint devices. We make use of the same technique discussed here and build upon our framework to work with SqueezeNet for fall detection using Jetson Nano as an edgaedgerocessing node. In this architecture, the data acquisition and processing layer acquires data and is processed locally on the Raspberry Pi as it is the main processing device for IoT devices. The Raspberry Pi itself is not capable of training the model as it lacks GPUthe capabilities that Jetson Nano possesses. The Jetson Nano receives the data and trains a preliminary model for text and multimedia data acquired from the surveillance camera. The multimedia data from the surveillance cameras are much larger in size compared to other devices, therefore the data is trained on the edge node, where the deep learning model is trained using Jetson Nano capable of performing certain levels of computation. However, we assume that the number of multimedia devices may grow in the future and the edge device mentioned here may not respond to all data for training. For this purpose, we split the model and lower-level layers are placed on the server, while the rest of the higher-level layers of the DNN are placed on the cloud with presumably an infinite amount of resources. The cloud model can receive weights from the edge server, it can also update parameters sent from the Jetson Nano or it can also be trained by the data directly sent to the cloud by the IoT devices themselves however, the data sent by the devices directly can compromise the acy and may require additional security. The aggregation module of the cloud layer provides means of aggregation for all parameters sent to the cloud and it helps update all the weights of a single model that can send predictions and estimations along with trends to the IoT devices. To deploy AI services through the cloud, we make use of cloud services provided by Amazon, called Amazon Web Services. It provides off-the-shelf technologies to integrate and deploy AI in IoT applications without too many complications. Moreover, it has a built-in dashboard to support Jetson Nano and is compatible with most IoT applications in general. Details of the working of the edge device and device-based learning for the IoT devices are presented in Fig. 3. The edge layer is equipped with a slightly more powerful device Jetson Nano that acts as a Fog Node and performs machine learning tasks for text and multimedia data. The device receives data and sends it to a feature extraction model. Then the features are sent to train the model itself, where they can be assessed for offloading the cloud or evaluated for training accuracy. The final predictions are sent back to the device and thereafter to the cloud for storage and updating of the cloud-based learning model. The cloud sends updated parameters to the Raspberry Pi which controls and overlooks the connected IoT devices. In this case, the Raspberry Pi changes the values of certain IoT devices like temperature sensors, and surveillance cameras to change parameters. For instance, the temperature sensor can raise the threshold based on the values sent by the cloud model for the next day. In the case of water management, the estimation module can predict the water requirements for the day based on values estimated by the model on previous days of the week. The estimation module uses a REST API and MQTT protocol to connect to the devices and updates the relevant sensor with appropriate results received from the model.

CHAPTER 6

PROJECT PANNING

6.1 PROTEUS DESIGN SUITE

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards. We used this suit to make a schematic and test our schematic virtually.

6.2 AURDINO COMPILER

The Arduino IDE brings its own GCC compiler. It is a version, that can compile for the AVR platform (don't know, if the standard version is capable of that). When we activate verbose output for compilation in the preferences of your Arduino IDE, we can see, what calls to GCC the IDE is doing. The Arduino development software performs some minor pre-processing to turn your sketch into a C++ program. Next, the dependencies of the sketch are located. It then gets passed to a compiler (avr-gcc), which turns the human-readable code into machine-readable instructions (or object files) Arduino uses a variant of the C++ programming language. The code is written in C++ with an addition of special methods and functions. However, it's possible to use Arduino with Python or another high-level programming language. Platforms like Arduino work well with Python, especially for applications that require integration with sensors and other physical devices.

CHAPTER 7

IMPLEMENTATION AND TESTING

7.1. SETTING UP ARDUINO IDE

Install the ESP8266 board in Arduino IDE, following these next instructions: In your Arduino IDE, go to File> PreferencesEnter http://arduino.esp8266.com/stable/package_esp8266com_index.json into the "Additional Boards Manager URLs" field as shown in the figure below. Then, click the "OK" button: Note: if already have the ESP32 boards URL, can separate the URLs with a comma Open the Boards Manager. Go to Tools > Board > Boards Manager…Search for ESP8266 and press the install button for the "ESP8266 by ESP8266 Community": That's it. It should be installed after a few seconds.

7.2 SETTING UP PCB

1. Turn the soldering iron on and let it warm up for a few minutes. Wet the sponge with a few ounces of clean water. When the soldering iron is hot, clean the tip on the sponge, then tin the tip by melting a small amount of fresh solder on it.
2. Look at the schematic and the circuit board's component side. Identify how to orient all the parts. Transistors, ICs and other components need to be placed correctly. Begin by inserting sockets for any ICs, cables, and other parts that need them. Put the sockets on the component side. To keep them from slipping out, you can bend a couple of pins over slightly on the trace side of the board.
3. Touch the soldering iron tip to the leads protruding through the pad holes for a few seconds. This is on the board's trace side. Let enough solder melt on the joint to cover it and form a smooth, shiny joint. Try not to dwell on the joint with the iron for more than about ten seconds. Solder each lead to its corresponding pad, then move on to the next socket. Let the solder cool for a minute. If any leads protrude from the solder pad longer than a couple of millimetres, clip them off with the diagonal cutters.
4. Insert a few resistors and capacitors into the board. Check the orientation of any polarized capacitors. Resistor leads are usually bent at a 90-degree angle with the part,

so the leads can slip through the board. Capacitor leads may be bent or go through straight, depending on the part. Solder each lead as in Step 3. Clip off excess leads after soldering. Continue to insert, orient, solder, and clip parts until all of them are done.

5. Prepare stranded wire, if any is required, by twisting the bare copper strands tightly. Heat the wire for a few seconds and let a thin coat of solder melt in. Insert and solder wire into the board as required. As above, clip off the excess. The wire may be for power, signals, or testing. For short jumpers, you can use clipped excess leads from Step 4.
6. Insert ICs and other parts into their sockets, checking their orientation. This completes the board's assembly.

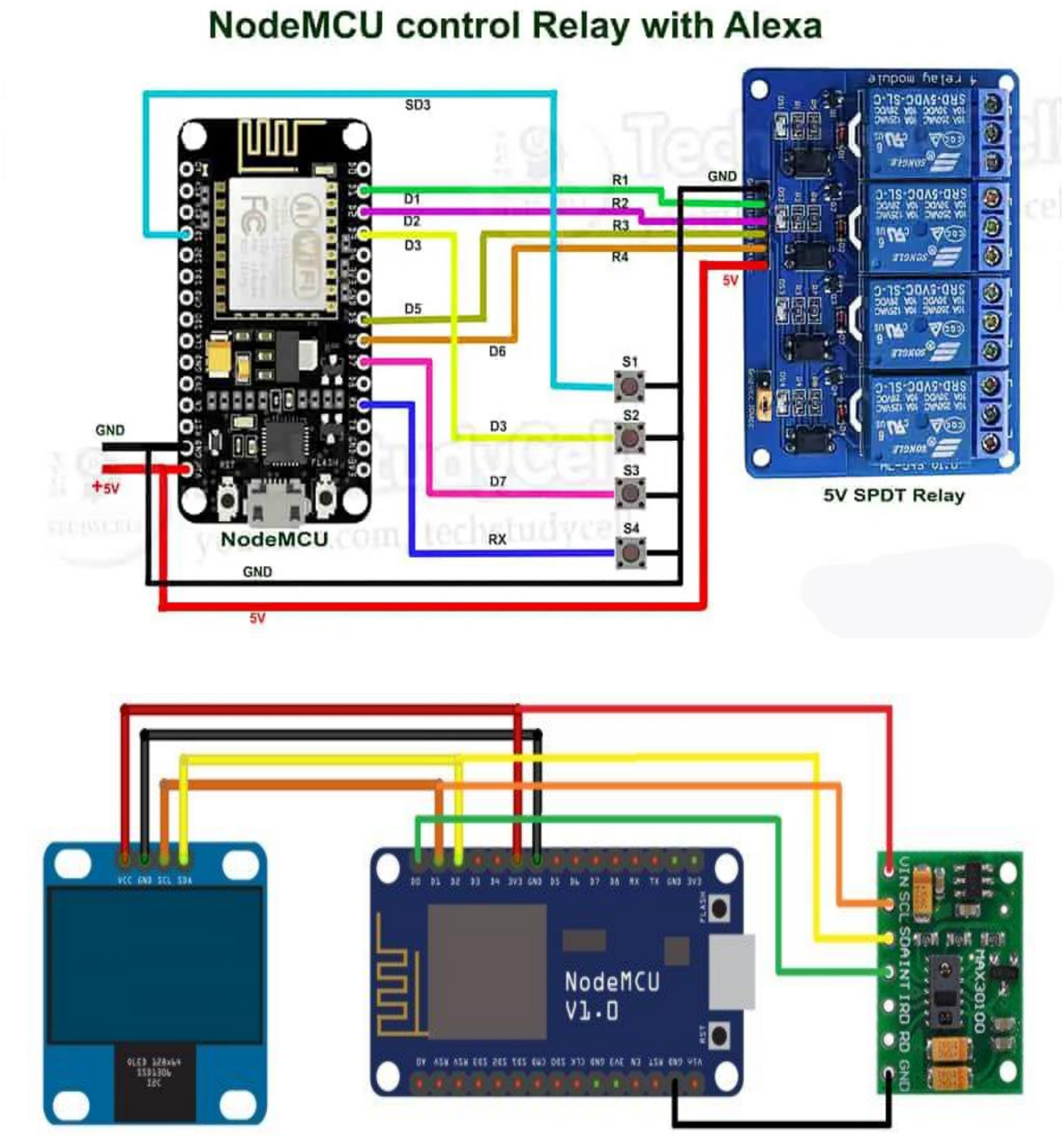


Fig 7.2 Schematic of basic need module and patient monitor module

7.3 PROGRAMMING THE MICROCONTROLLER

- Wire up the ESP module using a USB cable, and Setup up the Arduino IDE.
- Type the program and compile.
- Go to Tools > Board > Search for ESP 8266, and select the desired board.
- Flash your code! Now you're ready to use your ESP8266 as a stand-alone module.

7.4 SETTING UP AMAZON ALEXA AND FIRE STICK.

1. Download Alexa on a Smartphone device and make a user account.
2. Plug in your device.
3. Open the Alexa app.
4. Open more and select Add Device.
5. Select Amazon Echo.
6. Plug your Fire TV into your TV's HDMI port and then plug the power adapter into an outlet.
7. Use the USB cable to connect the power adapter and your Fire TV.
8. Put batteries in your Fire TV remote.
9. Turn on your TV and switch to the correct HDMI input.
10. If your remote doesn't pair automatically, press the Home button until "Press to start" appears on-screen (about 10 seconds).
11. Follow the on-screen instructions to connect to Wi-Fi and download the latest software update.
12. Sign in with your Amazon account.
13. Set up the fire tv app on your smartphone and connect the fire tv to the Echo device.

7.5 SETTING UP THE SMART DEVICES

Simply plug in and power on your new smart home device, and say, “Alexa, discover devices.” we can also add smart home devices through the Amazon Alexa app. Just open the app and select Devices, choose the **+** icon, and select Add Device. Then choose the device type and brand, and follow the on-screen prompts.

7.6 SETTING UP ALEXA ROUTINES.

If you have the relevant smart devices, you'll be amazed at how convenient it is to be able to turn off your Amazon Fire TV, adjust your lights, or lock your door with nothing more than your voice. Alexa Routines take that convenience to the next level. Alexa Routines are essentially shortcuts for Alexa—they make several things happen with one request. And you can customize them. For example, you can set up an Alexa Routine so that when you say, "Alexa, it's bedtime," Alexa will turn on the nightlight and start playing sleep sounds. The easiest way to get started is with Featured Routines, which are like templates and are available to use right away. To enable a Featured Routine, open the Alexa app and select more from the bottom navigation. Then, from the menu, select Routines and tap Featured. Select a Featured Routine, tap Enable, and you're all set. Learn more about Alexa Routines, and all the creative ways you can automate your smart home Another way Alexa can help make your home smarter is with Alexa Hunches. Hunches give Alexa the power to provide helpful suggestions about your connected smart devices. For example, if you have smart locks, Alexa can let you know if you forgot to lock the door when you leave home. Or if one of your smart lights is on when you say "Goodnight," Alexa alerts you and offers to turn it off.

7.7 SETTING UP BLYNK IOT

Registering an account on Blynk is the easiest way to build your own IoT product with a mobile app that works with the hardware of your choice. Once you log in have a brief Blynk Tour to have an idea of how the platform works. Install Blynk Library Blynk Library is an extension that runs on your hardware. It handles connectivity, device authentication in the cloud, and command processing between the Blynk app, Cloud, and hardware. It's highly flexible whether you are starting from scratch, or integrating Blynk into an existing project. Download Blynk App and open it Quick Start automatically creates a basic interface in your Blynk app, so you can monitor and control your device instantly. Now add your logic with Blynk's drag-and-drop app constructor, packed with IoT features: Visualize and plot data from any sensor, Control relays, motors, and any electrical appliances Get push notifications, send emails, add multiple devices etc.

7.8 PROTEUS VSM

Proteus VSM was the first product to bridge the gap between schematic and PCB for embedded design, offering a system-level simulation of microcontroller-based designs inside the schematic package itself. Over fifteen years later, Proteus VSM is still leading the field with more microcontroller variants and peripherals than any competing product, better measurement and debug tools and a consistent focus on innovation. Proteus VSM provides a unique development platform for the embedded engineer. It allows you to specify a program (HEX file, COF File, ELF/DWARF2 File, UBROF File etc) as a property of the microcontroller part on the schematic and during simulation will show the effects of the program on the schematic we have created. we can change our 'hardware' by rewiring the schematic, changing component values for resistors, capacitors etc. and deleting or adding new components to the design. You can change your firmware in the IDE of your choice and, once compiled, test the new code on the new system at the press of a button. This gives us total freedom to experiment with different ideas and find the optimal design solution for your project. The schematic serves as a 'virtual prototype' for the firmware and it's quick and easy to make changes to either.

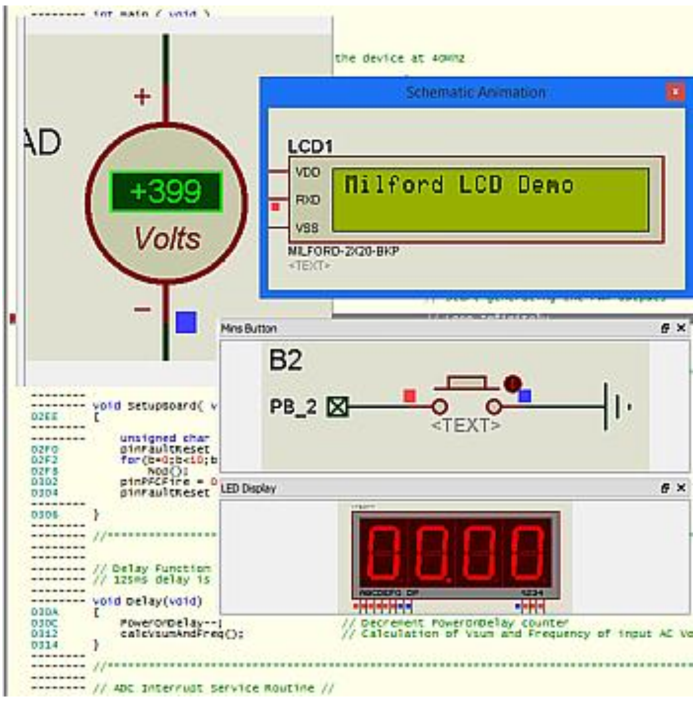


Fig 7.8 Proteus VSM

7.9 REAL-WORLD TESTING

We tested our Roban system with real word uses and we got 100% output as we expected. The system works perfectly without any errors. And fulfilling all the needs.

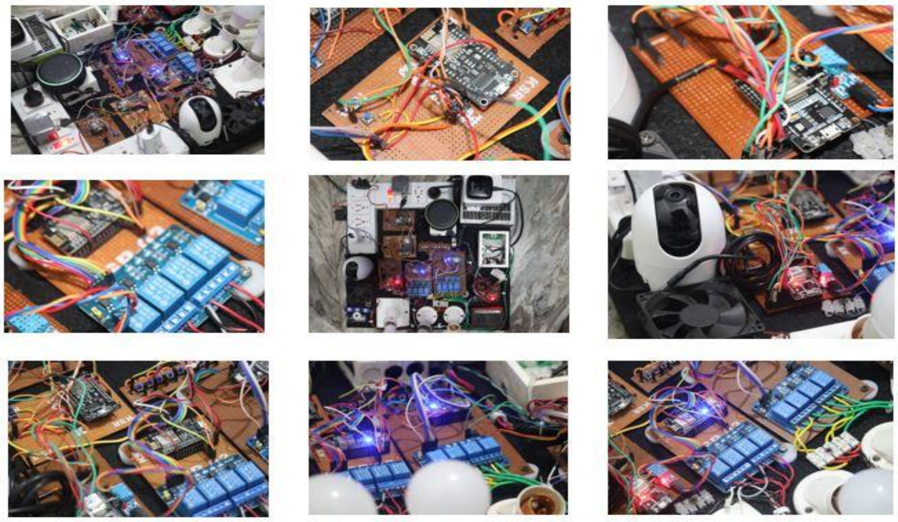


Fig 7.9 Output Implementation in real world

7.9.1 CONNECTING DEVICES BY COMMANDS

Connect our device to a power source and make sure it’s in pairing mode. Wait a minute or two for Alexa to pick up the device on our network. If no notification appears, say “Alexa, discover devices” to see if the device is detected. we should then receive a notification on any mobile device with the Alexa app that a new device was found. Turn the device on and off make sure Alexa is connected to the device by saying “Alexa, turn on [name of device].” Look for any lights to turn on and listen for any sounds to indicate the device has been activated. When you say “Alexa, turn off [name of device],” the device should then turn off. Change device settings. we can view and modify certain settings for the device. Open the Alexa app and tap the Devices icon on the bottom toolbar, then tap the Plugs or All Devices icon on the Devices screen. Tap the name of the newly discovered device and then tap the Settings gear in the upper-right corner of the screen. At the Settings screen, tap the Edit Name link to change the name of the device. Here, we can also disable the device if we’re no longer using it. we can then re-enable it if we want to use it again. Tap the trash can icon to delete the device if we. don’t plan to use it ever again

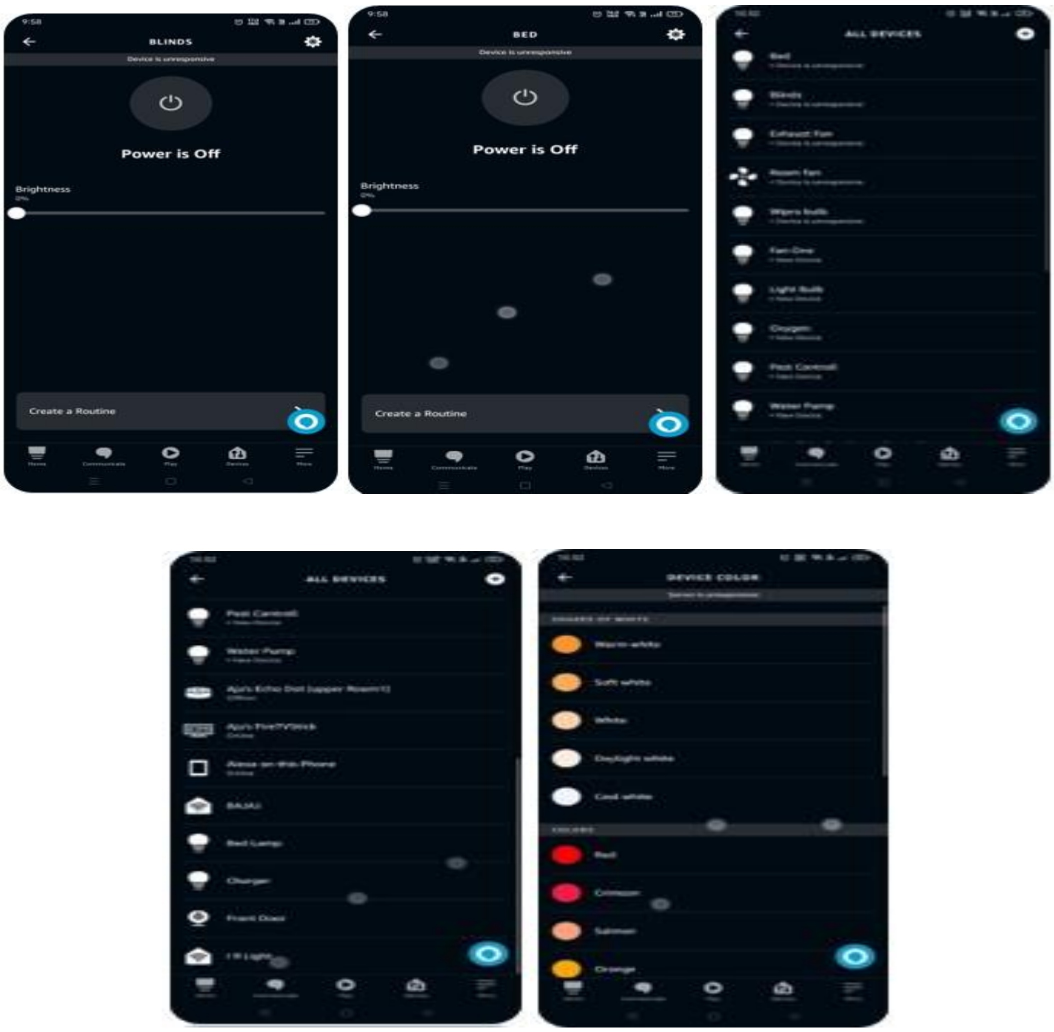


Fig 7.9.1.0 Alexa app interface

Fig 7.9.1.1 Firestick interface

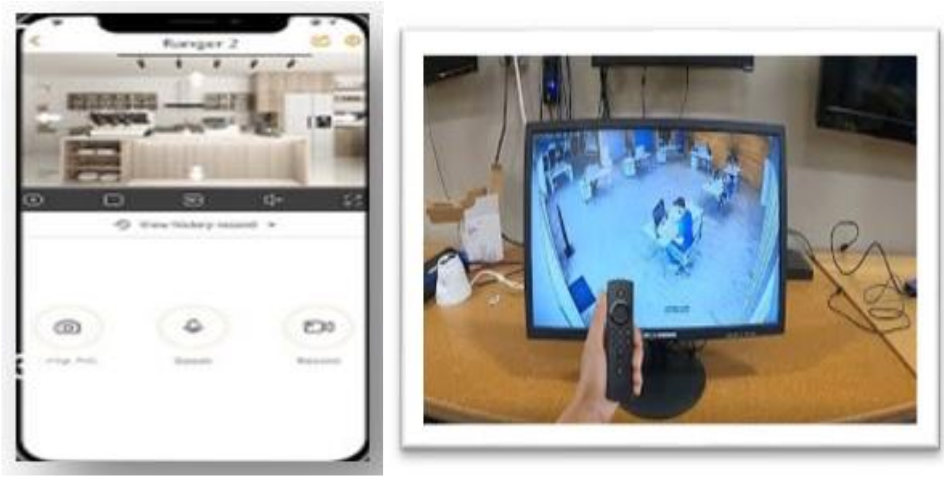


Fig 7.9.1.2 Security interface

7.9.2 MANUALLY CONNECT THE DEVICE

Add device If Alexa fails to detect your new smart device, or you deleted your saved Wi-Fi passwords for security reasons, we can still set up your smart device manually through the Alexa app. Plug in your device and make sure Bluetooth is enabled on your phone or tablet. In the app, tap Devices, tap the + button in the upper-right corner, and then select Add Device. Chose the type of device Chose the type of device you want to set up among the different categories, such as Amazon Echo, Light, Plug, Switch, Camera, Lock, Thermostat, or Speaker. Choose the brand of the device, then make sure your device is plugged in and powered on. Tap Next to continue. In some cases, you will next have to download the app for the device and use that to manually set it up. In other cases, we can simply scan the barcode from the device, its product box, or its instruction booklet. Follow the instructions for your specific device until it's connected to Wi-Fi and detected by the Alexa app Set up the device If prompted, tap Set Up Device to continue setting it up. we can then tap Choose Group to add the device to a specific group of items inside the Alexa app. We can now turn your device on and off by speaking to Alexa.

7.9.3 BLYNK APP TEST

we will try to make a Smart Health Monitoring Device that can measure SpO2 (percentage of oxygen in the blood) and heart rate in BPM (Beat Per Minute). This wearable device can be used by athletes to monitor their heart rate and blood oxygen levels during a workout. The best part of this project is that you can connect this device to an Android app Blynk that will record and regularly update the data for both SPO2 & BPM on the internet. Even anyone can monitor the data from any part of the world as data are uploaded on the server. As there is an availability of online data, this project can be used to monitor the health of a patient online. O, download and install the Blynk Application from the Google Play store. IOS users can download it from the App Store. Once the installation is completed, open the app & sign-up using your Email id and Password.

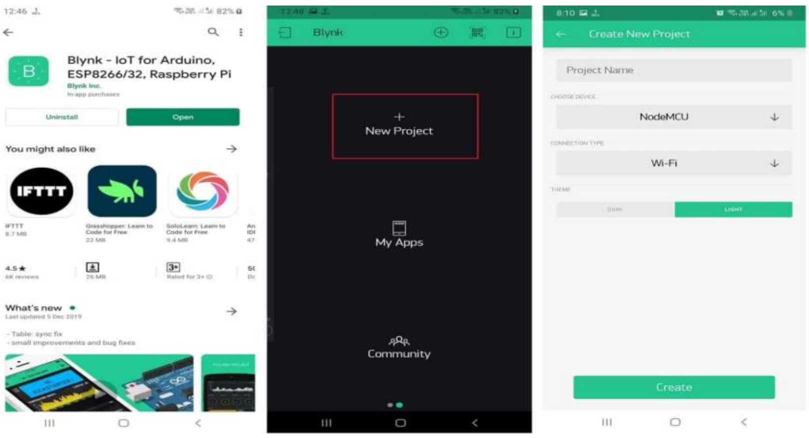


Fig 7.9.3.0 Blynk app interface

Now click on "New Project". In the pop up set the parameters like Project name, Board and connection type as shown in the photo above. For this MAX30100 ESP8266 project select the device as NodeMCU and the connection type as Wi-Fi. Then click on Create.

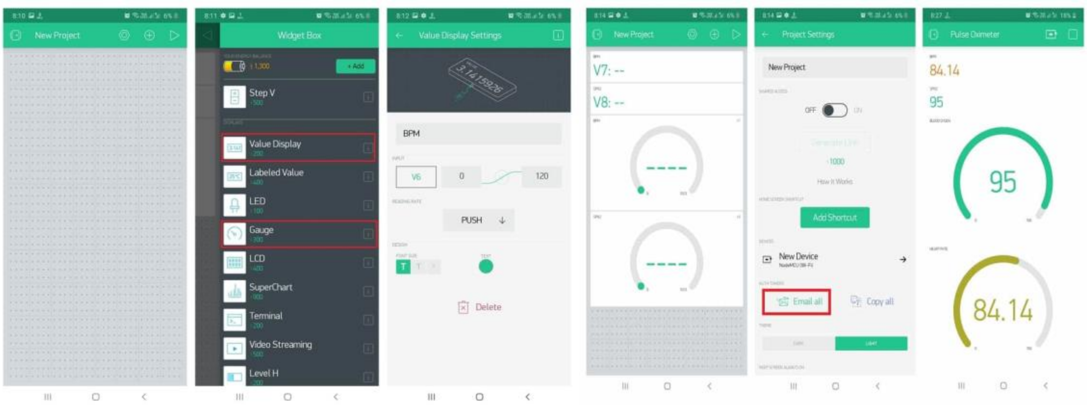


Fig 7.9.3.1 BLYNK monitoring using the app

Now click on the “+” sign to add the widgets. We need to read the value of BPM & SpO2. So select a pair of widgets named Value Display **&** Gauge**.** After dragging the widgets, set their parameters as shown in the image above. Click on **Value Display** and set the pin to “**V7**” & “**V8**“. Similarly, in gauge settings, set the output pin to “V7” & “V8”.After the successful creation of the Project, go back to the setting and click on Send Email. You will get an Authenticate ID on registered mail. Save the Authenticate ID for future reference.

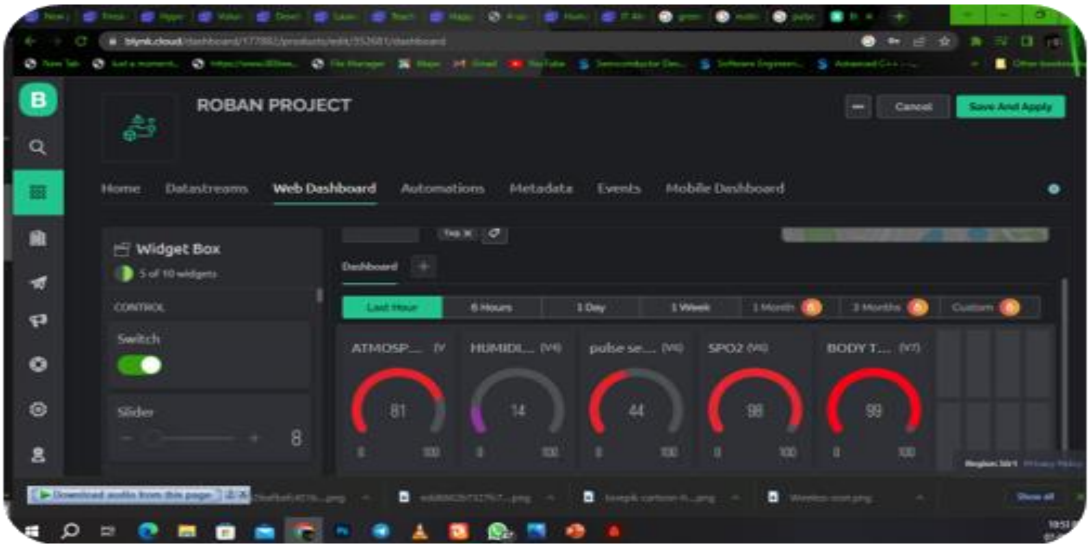


Fig 7.9.3.2 BLYNK web interface

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

8.1 CONCLUSION

In conclusion, the utilization of Internet of Things (IoT) technology for disabled individuals has the potential to revolutionize their lives and empower them to live more independently and inclusively. By seamlessly connecting various devices and systems, IoT enables smart and accessible solutions that cater to the specific needs of disabled people across different domains.

One of the key benefits of IoT for disabled individuals is the enhanced control and automation it provides. Through a network of interconnected devices, individuals can manage their environments, appliances, and assistive technologies with greater ease. This level of control not only promotes independence but also reduces the reliance on caregivers or support systems, allowing disabled individuals to lead more self-determined lives.

IoT technology also enables real-time monitoring and remote assistance, offering a higher level of safety and security for disabled individuals. Wearable devices, sensors, and smart home systems can collect and transmit data, providing valuable insights into health conditions, movement patterns, and potential risks. This data can be utilized by healthcare professionals, family members, or emergency services to respond promptly and appropriately, minimizing the chances of accidents or medical emergencies.

Furthermore, IoT solutions can bridge the accessibility gap by transforming physical spaces and public infrastructure into inclusive environments. By incorporating sensors, voice recognition, and adaptive interfaces, public facilities, transportation systems, and workplaces can become more user-friendly for individuals with disabilities. This integration of IoT into urban planning and infrastructure promotes inclusivity and ensures equal access and opportunities for all.

Despite the immense potential, it is important to address the challenges and concerns associated with IoT for disabled individuals. Privacy and security must be prioritized to safeguard personal data and protect against potential vulnerabilities or unauthorized access. Additionally, the affordability and compatibility of IoT devices and systems should be considered to ensure accessibility for individuals from diverse economic backgrounds.

In conclusion, IoT technology holds great promise for disabled people, offering them increased independence, safety, and inclusivity. By harnessing the power of connectivity, IoT can transform the lives of individuals with disabilities, fostering a more inclusive society that values and supports their rights and aspirations. With ongoing research, innovation, and collaboration, the full potential of IoT for disabled individuals can be realized, empowering them to thrive and participate fully in all aspects of life.

8.2 FUTURE SCOPE

The future scope of using the Internet of Things (IoT) for disabled people is vast and promising, with ongoing advancements in technology and an increasing focus on inclusivity. Here are some key areas where IoT can continue to make a significant impact:

1. Assistive Technologies: IoT can play a crucial role in developing advanced assistive technologies for disabled individuals. From smart prosthetics and exoskeletons to brain-computer interfaces, IoT integration can enhance the functionality and control of these devices. Real-time data collection and analysis can lead to more personalized and adaptive assistive technologies that cater to the specific needs of each individual.
2. Healthcare Monitoring and Management: IoT-enabled healthcare solutions can revolutionize the way disabled individuals manage their health conditions. Smart wearable devices, embedded with sensors, can continuously monitor vital signs, track medication adherence, and provide early warning signs of potential health issues. This real-time data can be shared with healthcare providers, enabling proactive interventions and remote healthcare management.
3. Accessibility in Smart Cities: As cities become increasingly connected, IoT can contribute to building more accessible environments for disabled individuals. Smart transportation systems can offer personalized navigation, ensuring accessible routes and real-time information on public transport options. Smart buildings and public spaces can incorporate IoT technologies to automate accessibility features, such as adjustable ramps, voice-activated controls, and adaptive lighting.
4. Social Inclusion and Communication: IoT can facilitate improved social inclusion and communication for disabled individuals. Connected devices and smart home systems can enable seamless interaction with friends, family, and caregivers. Voice-activated assistants and smart communication devices can enhance the communication abilities of individuals with speech impairments. Moreover, virtual reality (VR) and augmented reality (AR) technologies, coupled with IoT, can create immersive and inclusive social experiences.
5. Independent Living and Safety: IoT can further empower disabled individuals to live independently while ensuring their safety. Smart home automation systems can monitor and control various aspects of the living environment, including temperature, lighting, and security, through voice commands or wearable devices. Fall detection sensors,

emergency call systems, and predictive analytics can mitigate risks and provide prompt assistance in case of accidents or emergencies.

6. Adaptive Learning and Employment: IoT can revolutionize adaptive learning and employment opportunities for disabled individuals. Connected learning platforms can offer personalized educational content and adaptive interfaces to accommodate different learning styles and abilities. IoT-enabled workplace environments can provide assistive technologies, ergonomic adjustments, and real-time feedback systems to support disabled individuals in their professional endeavours.

As IoT technologies continue to evolve, it is essential to address challenges such as data privacy, security, and affordability. Collaboration between technology developers, disability advocates, and policymakers will be crucial to ensure that IoT solutions are accessible, inclusive, and ethically implemented.

In conclusion, the future of IoT for disabled people holds immense potential. By leveraging the power of connectivity, IoT can enhance accessibility, independence, and overall quality of life for individuals with disabilities. Continued research, innovation, and collaboration will pave the way for transformative IoT solutions that empower disabled individuals, promote inclusivity, and create a more equitable society.

REFERENCE

1. "Internet of Things for Disabled People: Opportunities and Challenges" by Abir Awad et al. - This research article provides an in-depth analysis of the opportunities and challenges associated with using IoT for disabled people. It discusses various applications and technologies that can enhance the lives of individuals with disabilities. [Link:

https://www.researchgate.net/publication/325593589_Internet_of_Things_for_Disabled_People_Opportunities_and_Challenges]

2. "Accessible Internet of Things for People with Disabilities" - This book by Simon Judge explores the concept of accessible IoT for people with disabilities. It covers topics such as assistive technologies, smart homes, healthcare applications, and accessibility considerations. [Link: https://www.springer.com/gp/book/9783030034691]
3. "The Internet of Things for People with Disabilities" by Samuel Greengard - This article published on the TechTarget website provides an overview of IoT applications and benefits for people with disabilities. It discusses how IoT can improve accessibility, healthcare, and independent living. [Link: https://internetofthingsagenda.techtarget.com/feature/The-Internet-of-Things-for-people-with-disabilities]
4. "Microsoft's AI and IoT Solutions for People with Disabilities" - This resource from Microsoft showcases their AI and IoT solutions designed to empower people with disabilities. It includes case studies, videos, and documentation on their accessibility initiatives and technologies. [Link: https://www.microsoft.com/en-us/ai/empower-people-with-disabilities]
5. "Smart Cities for All: A Vision for an Inclusive, Accessible Urban Future" - This initiative by G3ict and World Enabled focuses on promoting smart cities that are inclusive and accessible for all, including people with disabilities. It provides resources, reports, and case studies related to IoT and accessibility in smart cities. [Link: https://smartcities4all.org/]
6. Online Courses and Tutorials:
    - IoT for Beginners: Getting Started with the Internet of Things (Udemy) - This beginner-level course provides a comprehensive introduction to IoT concepts, applications, and technologies. It covers the basics of IoT architecture, sensors, connectivity, and IoT-enabled devices. [Link: https://www.udemy.com/course/iot-internet-of-things]
    - Internet of Things (IoT) and Artificial Intelligence (AI) at Scale (Coursera) - Offered by IBM, this course explores the integration of IoT and AI technologies. It covers IoT platforms, data analytics, and AI algorithms for developing smart applications. [Link: https://www.coursera.org/learn/iot-ai-at-scale]

APPENDIX

A.1 Codes Flashed To Node Mcu 8266

To enable ESP8266 firmware flashing GPIO0 pin must be pulled low before the device is reset. Conversely, for a normal boot, GPIO0 must be pulled high or floating.If we have a NodeMCU dev kit(ESP 12) then you don't need to do anything, as the USB connection can pull GPIO0 low automatically while flashing Go to Operation tab of Flasher software and click on Flash(F) button. By doing this ESP8266 module will enter into flash mode, we may check this by clicking on Log tab of Flasher Software. It is showing Acknowledge successful. After successful acknowledgment it show the status of set base address.

A.2 Code for Basic needs module.

```
#ifdef ARDUINO_ARCH_ESP32
#include <WiFi.h>
#else
#include <ESP8266WiFi.h>
#endif
#include <Espalexa.h>
#define RelayPin1 5  //D1
#define RelayPin2 4  //D2
#define RelayPin3 14 //D5
#define RelayPin4 12 //D6
#define SwitchPin1 10  //SD3
#define SwitchPin2 0   //D3
#define SwitchPin3 13  //D7
#define SwitchPin4 3   //RX
#define wifiLed   16   //D0
int toggleState_1 = 0; //Define integer to remember the toggle state for relay 1
int toggleState_2 = 0; //Define integer to remember the toggle state for relay 2
int toggleState_3 = 0; //Define integer to remember the toggle state for relay 3
int toggleState_4 = 0; //Define integer to remember the toggle state for relay 4boolean
connectWifi();
//callback functions
void firstLightChanged(uint8_t brightness);
void secondLightChanged(uint8_t brightness);
void thirdLightChanged(uint8_t brightness);
void fourthLightChanged(uint8_t brightness);
// WiFi Credentials
const char* ssid = "Project_Roban";
const char* password = "1234roban_aj";
// device names
String Device_1_Name = "Light Bulb";
String Device_2_Name = "Fan One";
String Device_3_Name = "Pest Controll";
String Device_4_Name = "Charger";
boolean wifiConnected = false;
Espalexa espalexa;
void firstLightChanged(uint8_t brightness){
  if (brightness == 255)    {
      digitalWrite(RelayPin1, LOW);
      Serial.println("Device1 ON");
```

```
      toggleState_1 = 1;  }
  else {  digitalWrite(RelayPin1, HIGH);
    Serial.println("Device1 OFF");
    toggleState_1 = 0;  }}
void secondLightChanged(uint8_t brightness){
  if (brightness == 255)   {
      digitalWrite(RelayPin2, LOW);
 Serial.println("Device2 ON");
      toggleState_2 = 1;   }
  else  {    digitalWrite(RelayPin2, HIGH);
    Serial.println("Device2 OFF");
    toggleState_2 = 0;  }}
void thirdLightChanged(uint8_t brightness){
    if (brightness == 255)    {
      digitalWrite(RelayPin3, LOW);
      Serial.println("Device3 ON");
      toggleState_3 = 1;   }
  else  {
    digitalWrite(RelayPin3, HIGH);
    Serial.println("Device3 OFF");
    toggleState_3 = 0;  }}
void fourthLightChanged(uint8_t brightness){
  if (brightness == 255)    {
      digitalWrite(RelayPin4, LOW);
      Serial.println("Device4 ON");
      toggleState_4 = 1;    }
  else  {
    digitalWrite(RelayPin4, HIGH);
    Serial.println("Device4 OFF");
    toggleState_4 = 0;  }}
void relayOnOff(int relay){
    switch(relay){
      case 1:
             if(toggleState_1 == 0){
              digitalWrite(RelayPin1, LOW); // turn on relay 1
              toggleState_1 = 1;
              Serial.println("Device1 ON");     }
             else{  digitalWrite(RelayPin1, HIGH); // turn off relay 1
              toggleState_1 = 0;
              Serial.println("Device1 OFF");      }
             delay(100);
      break;
      case 2:
             if(toggleState_2 == 0){
              digitalWrite(RelayPin2, LOW); // turn on relay 2
              toggleState_2 = 1;
              Serial.println("Device2 ON"       }
             else{
              digitalWrite(RelayPin2, HIGH); // turn off relay 2
              toggleState_2 = 0;
```

```
            Serial.println("Device2 OFF");  }
           delay(100);
    break;
    case 3:
           if(toggleState_3 == 0){
            digitalWrite(RelayPin3, LOW); // turn on relay 3
            toggleState_3 = 1;
            Serial.println("Device3 ON");
            }else{
            digitalWrite(RelayPin3, HIGH); // turn off relay 3
            toggleState_3 = 0;
            Serial.println("Device3 OFF"); }
           delay(100);
    break;
    case 4:
           if(toggleState_4 == 0){
            digitalWrite(RelayPin4, LOW); // turn on relay 4
            toggleState_4 = 1;
            Serial.println("Device4 ON");     }
           else{
            digitalWrite(RelayPin4, HIGH); // turn off relay 4
            toggleState_4 = 0;
            Serial.println("Device4 OFF");    }
           delay(100);
    break;
    default : break;
    }}
boolean connectWifi(){
  boolean state = true;
  int i = 0;
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  Serial.println("");
  Serial.println("Connecting to WiFi");
  Serial.print("Connecting...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    if (i > 20) {
      state = false; break;  }
    i++; }
  Serial.println("");
  if (state) {
    Serial.print("Connected to ");
    Serial.println(ssid);
    Serial.print("IP address: ");
    Serial.println(WiFi.localIP());  }
  else {
    Serial.println("Connection failed.");  }
  return state;}
```

```
void addDevices(){
  espalexa.addDevice(Device_1_Name, firstLightChanged); //simplest definition, default state
off
  espalexa.addDevice(Device_2_Name, secondLightChanged);
  espalexa.addDevice(Device_3_Name, thirdLightChanged);
  espalexa.addDevice(Device_4_Name, fourthLightChanged);
  espalexa.begin();}
void setup(){
  Serial.begin(115200);
  pinMode(RelayPin1, OUTPUT);
  pinMode(RelayPin2, OUTPUT);
  pinMode(RelayPin3, OUTPUT);
  pinMode(RelayPin4, OUTPUT);
  pinMode(wifiLed, OUTPUT);
  pinMode(SwitchPin1, INPUT_PULLUP);
  pinMode(SwitchPin2, INPUT_PULLUP);
  pinMode(SwitchPin3, INPUT_PULLUP);
  pinMode(SwitchPin4, INPUT_PULLUP);
  digitalWrite(RelayPin1, HIGH);
  digitalWrite(RelayPin2, HIGH);
  digitalWrite(RelayPin3, HIGH);
  digitalWrite(RelayPin4, HIGH);
  wifiConnected = connectWifi();
  if (wifiConnected)  {
    addDevices();  }
  else  {
    Serial.println("Cannot connect to WiFi. So in Manual Mode");
    delay(1000);  }}
void loop(){
   if (WiFi.status() != WL_CONNECTED)  {
    digitalWrite(wifiLed, HIGH); //Turn off WiFi LED
    if (digitalRead(SwitchPin1) == LOW){
      delay(200);
      relayOnOff(1);         }
    else if (digitalRead(SwitchPin2) == LOW){
      delay(200);
      relayOnOff(2);    }
    else if (digitalRead(SwitchPin3) == LOW){
      delay(200);
      relayOnOff(3);  }
    else if (digitalRead(SwitchPin4) == LOW){
      delay(200);
      relayOnOff(4);    }  }
  else  {  digitalWrite(wifiLed, LOW);
    if (digitalRead(SwitchPin1) == LOW){
      delay(200);
      relayOnOff(1);        }
    else if (digitalRead(SwitchPin2) == LOW){
      delay(200);
      relayOnOff(2);    }
```

```
  else if (digitalRead(SwitchPin3) == LOW){
    delay(200);
    relayOnOff(3);}
  else if (digitalRead(SwitchPin4) == LOW){
    delay(200);
    relayOnOff(4);   }
      if (wifiConnected){
    espalexa.loop();
    delay(1);   }
  else {  wifiConnected = connectWifi(); // Initialise wifi connection
    if(wifiConnected){
    addDevices()   }  } }}
```

A.3 Code for Bed And Ventilation Module

```
#include <ESP8266WiFi.h>
#include <Espalexa.h>
#include <Servo.h>
Servo blindServo;
static const int servoPin = 15; // The Pin which the Servo is connected to
const int button1Pin = 5; // The Pin which Button 1 is connected to
const int button2Pin = 4; // The Pin which Button 2 is connected to
int servoPosition = 0; // This sets the initial postiion of the servo
boolean connectWifi();
void servoPositionChanged(uint8_t servoPosition);
const char* ssid = "Project_Roban";
const char* password = "1234roban_aj";
boolean wifiConnected = false;
Espalexa espalexa;
void setup(){
  Serial.begin(115200);
  blindServo.attach(servoPin);
  pinMode(button1Pin, INPUT_PULLUP);
  pinMode(button2Pin, INPUT_PULLUP);
  wifiConnected = connectWifi();
  if (wifiConnected) {
    espalexa.addDevice("Bed", servoPositionChanged, 20);
  } else {
    while (1) {
      Serial.println("Cannot connect to WiFi. Please check data and reset the ESP.");
      delay(2500); }  }}
void loop(){
  while (digitalRead(button1Pin) == LOW && servoPosition < 179) {
    blindServo.attach(servoPin); // This activates the servo (Attaches the servo)
    servoPosition++;
    blindServo.write(servoPosition);
    Serial.println(servoPosition);
    delay(15);
    blindServo.detach(); }
```

```
  while (digitalRead(button2Pin) == LOW  && servoPosition > 0) {
    blindServo.attach(servoPin);
    servoPosition--;
    blindServo.write(servoPosition);
    Serial.println(servoPosition);
    delay(15);
    blindServo.detach();  }
  espalexa.loop();
  delay(1);}
void servoPositionChanged(uint8_t servoPosition) {
  Serial.print("Servo Position changed to ");
  if (servoPosition < 180) {
    blindServo.attach(servoPin);
    Serial.print("Servo Position ");
    Serial.println(servoPosition);
    blindServo.write(servoPosition);
    servoPosition = servoPosition;
    delay(1000);
    blindServo.detach(); }
  else  { Serial.print("Servo Out of Range ");
    Serial.println(servoPosition);  }}
boolean connectWifi() {
  boolean state = true;
  int i = 0;
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  Serial.println("");
  Serial.println("Connecting to WiFi");
  Serial.print("Connecting...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    if (i > 20) {
      state = false; break;   }
   i++; }
  Serial.println("");
  if (state) {
    Serial.print("Connected to ");
    Serial.println(ssid);
    Serial.print("IP address: ");
    Serial.println(WiFi.localIP());  }
  else {   Serial.println("Connection failed.");  }
  return state;}
```

A.4 Code patient monitor Module

```
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"
#define BLYNK_PRINT Serial
```

```
#include <Blynk.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "Wire.h"
#include "Adafruit_GFX.h"
#include "OakOLED.h"
#define REPORTING_PERIOD_MS 1000
OakOLED oled;
char auth[] = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"; // You should get Auth Token in the
Blynk App.
char ssid[] = "xxxxxxxxxx";  // Your WiFi credentials.
char pass[] = "xxxxxxxxxx";
PulseOximeter pox;
float BPM, SpO2;
uint32_t tsLastReport = 0;
const unsigned char bitmap [] PROGMEM={
0x00, 0x00, 0x00, 0x00, 0x01, 0x80, 0x18, 0x00, 0x0f, 0xe0, 0x7f, 0x00, 0x3f, 0xf9, 0xff,
0xc0,
0x7f, 0xf9, 0xff, 0xc0, 0x7f, 0xff, 0xff, 0xe0, 0x7f, 0xff, 0xff, 0xe0, 0xff, 0xff, 0xff, 0xf0,
0xff, 0xf7, 0xff, 0xf0, 0xff, 0xe7, 0xff, 0xf0, 0xff, 0xe7, 0xff, 0xf0, 0x7f, 0xdb, 0xff, 0xe0,
0x7f, 0x9b, 0xff, 0xe0, 0x00, 0x3b, 0xc0, 0x00, 0x3f, 0xf9, 0x9f, 0xc0, 0x3f, 0xfd, 0xbf, 0xc0,
0x1f, 0xfd, 0xbf, 0x80, 0x0f, 0xfd, 0x7f, 0x00, 0x07, 0xfe, 0x7e, 0x00, 0x03, 0xfe, 0xfc, 0x00,
0x01, 0xff, 0xf8, 0x00, 0x00, 0xff, 0xf0, 0x00, 0x00, 0x7f, 0xe0, 0x00, 0x00, 0x3f, 0xc0, 0x00,
0x00, 0x0f, 0x00, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00};
void onBeatDetected(){
  Serial.println("Beat Detected!");
  oled.drawBitmap( 60, 20, bitmap, 28, 28, 1);
  oled.display();}
void setup(){
  Serial.begin(115200);
  oled.begin();
  oled.clearDisplay();
  oled.setTextSize(1);
  oled.setTextColor(1);
  oled.setCursor(0, 0);
  oled.println("Initializing pulse oximeter..");
  oled.display();
  pinMode(16, OUTPUT);
  Blynk.begin(auth, ssid, pass);
  Serial.print("Initializing Pulse Oximeter..");
  if (!pox.begin())   {
     Serial.println("FAILED");
     oled.clearDisplay();
     oled.setTextSize(1);
     oled.setTextColor(1);
     oled.setCursor(0, 0);
     oled.println("FAILED");
     oled.display();
     for(;;); }
```

```
  else {
       oled.clearDisplay();
       oled.setTextSize(1);
       oled.setTextColor(1);
       oled.setCursor(0, 0);
       oled.println("SUCCESS");
       oled.display();
       Serial.println("SUCCESS");
       pox.setOnBeatDetectedCallback(onBeatDetected);  }}
void loop(){
  pox.update();
  Blynk.run();
  BPM = pox.getHeartRate();
  SpO2 = pox.getSpO2();
   if (millis() - tsLastReport > REPORTING_PERIOD_MS){
      Serial.print("Heart rate:");
      Serial.print(BPM);
      Serial.print(" bpm / SpO2:");
      Serial.print(SpO2);
      Serial.println(" %");
      Blynk.virtualWrite(V7, BPM);
      Blynk.virtualWrite(V8, SpO2);
      oled.clearDisplay();
      oled.setTextSize(1);
      oled.setTextColor(1);
      oled.setCursor(0,16);
      oled.println(pox.getHeartRate());
      oled.setTextSize(1);
      oled.setTextColor(1);
      oled.setCursor(0, 0);
      oled.println("Heart BPM");
      oled.setTextSize(1);
      oled.setTextColor(1);
      oled.setCursor(0, 30);
      oled.println("Spo2");
      oled.setTextSize(1);
      oled.setTextColor(1);
      oled.setCursor(0,45);
      oled.println(pox.getSpO2());
      oled.display();
      tsLastReport = millis();    }}
```

****