Paramov: Smart IOT based Paralyzed Patient Monitoring System

A PROJECT REPORT

Submitted by

Vanshaj Barnwal [RA2011032010035]
Vinay Poddar [RA2011032010061]

Under the Guidance of

Ms. Sai Santhiya D

Assistant Professor, Department of Networking and Communications

in partial fulfillment of the requirementsfor the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING
with specialization in Internet of Things

DEPARTMENT OF NETWORKING AND COMMUNICATIONS
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE ANDTECHNOLOGY
KATTANKULATHUR- 603 203

MAY 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that 18CSP109L/18CSP111L project report [titled "**Paramov: Smart IOT Based Paralyzed Patient Monitoring System** " is the bonafide work of "**Vanshaj Barnwal [Register Number: RA2011032010035], Vinay Poddar [RA2011032010061]"** who carried out the project work under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Mrs. Saisanthiya D
SUPERVISOR
ASSISTANT PROFESSOR
DEPARTMENT OF
NETWORKING AND
COMMUNICATIONS

Dr. Annapurani K
HEAD OF THE DEPARTMENT
DEPARTMENT OF
NETWORKING AND
COMMUNICATIONS

Internal Examiner

External Examiner

Department of Networking and Communications
SRM Institute of Science & Technology
Own Work Declaration Form

Degree/ Course	: B. Tech in Computer Science and Engineering with Specialization in the Internet of Things
Student Names	: Vanshaj Barnwal, Vinay Poddar
Registration Number	: RA2011032010035, RA2011032010061
Title of Work	: Paramov: Smart IOT Based Paralyzed Patient Monitoring System

I / We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is my / our own except where indicated, and that I / We have met the following conditions:

- Clearly referenced / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:	
I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above.	
RA2011032010035	RA2011032010061

ACKNOWLEDGEMENT

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, Dr. T.V.Gopal, for his invaluable support.

We wish to thank Dr. Revathi Venkataraman, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, Dr. Annapurani Panaiyappan K, Professor, Department of Networking and Communication, SRM Institute of Science and Technology, for her suggestions and encouragement at all stages of the project work.

We want to convey our thanks to our Project Panel Members, Mrs. Saisanthiya D, Assistant Professor, Dr. Mary Subaja Christo, Associate Professor and Panel Head, and
Dr. T. Preethiya, Assistant Professor, and Dr. Gouthaman P, Assistant Professor, SRM Institute of Science and Technology, for their input during the project reviews and support. We register our immeasurable thanks to our Faculty Advisor, Dr. Mary Subaja Christo, Associate Professor, and Mrs. Saisanthiya D, Assistant Professor Department of Networking and Communications, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, Ms. Sai Santhiya D, Assistant Professor, Department of Networking and Communications, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship. She provided us with the freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank the Networking and Communications Department staff and students, of SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank parents, family members, and friends for their unconditional love, constant support, and encouragement.

Vanshaj Barnwal [RA2011032010035]

Vinay Poddar [RA2011032010061]

TABLE OF CONTENTS

ABSTRACT

Paramov is a cutting-edge technical Smart IoT-Based Paralyzed Patient Monitoring System (SIPMS) which offers a novel solution to the problems associated with monitoring paralyzed patients. Whether from neurological illnesses, spinal cord injuries, or other medical issues, paralysis frequently necessitates ongoing monitoring to identify patient requirements and well-being and give essential help. Continuous data streams track vital signs, activity, sleep patterns, and interactions with the environment. This allows for the early identification of infections, falls, and complications and initiates preventive measures. System sees a distinct future. Real-time data wirelessly travels to a central hub, acting as a silent monitor over their well-being, eliminating the need for intrusive checks. The smart gloves must provide real-time feedback on finger movements in order to assist caretakers in making informed judgments. This smart paralysis system has a number of sensors, including as flex, temperature, and tilt sensors. The data is subsequently sent via Wi-Fi to a specialized smartphone app, allowing doctors, nurses, or other caregivers to swiftly access and comprehend the user's present state of health. This device could be worn on the finger or designed tofit within a person's clothes. The application of Paramov has the capability to completely transform the care of patients who are paralyzed by providing a preventive, intelligent monitoring system that boosts overall quality of life, lowers the risk of challenges, and improves patient outcomes. In addition, Paramov features intuitive user interfaces, including as web dashboards and mobile applications, which enable medical professionals and caregivers to remotely check on patients'health condition, view past data, and customize alarm settings. Because of its accessibility, patients who are paralyzed can receive better treatment and their carers can rest easy knowingthat their patients are being monitored. Furthermore, it may be implemented in a variety of healthcare settings, such as hospitals, rehabilitation facilities, and home care settings, because to its scalability and versatility.

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

IOT – Internet of Things

R&D – Research and Development

SMTP – Simple Mail Transfer Protocol

GSM – Global System for Mobile Communication

LoRaWAN – Long Range Wide Area Network

Wi-Fi – Wireless Fidelity

MQTT – Message Queuing Telemetry Transport

IPv6 - Internet Protocol version 6

LoWPAN - Low-Power Wireless Personal Area Network

DTLS – Datagram Transport Layer Security

NFC – Near-Field Communication

BLE – Bluetooth Low Energy

RFID - Radio-Frequency Identification

IIOT – Industrial Internet of Things

IP – Internet Protocol

AMQP - Advanced Message Queuing Protocol

XMPP - Extensible Messaging and Presence Protocol

CHAPTER 1

INTRODUCTION ON PARALYSIS MONITORING SYSTEM

The Internet of Things (IoT) has become a game-changing idea in technology that is redefining how we conduct business, interact with our physical surroundings, and go about our daily lives. The internet of things (IoT) is a network of physically connected "things" that are equipped with sensors, software, and communication technologies to gather, share, and use data. These items can be anything from commonplace gadgets like smart home appliances and cellphones to specialized machinery used in industrial, medical, and agricultural settings, among other fields.

1.1. General

The "things" themselves, which are outfitted with a variety of sensors and actuators, form the basis of the Internet of Things ecosystem. Thanks to these sensors, the devices are able to gather a wide range of environmental data, such as temperature, humidity, light, motion, location, andmuch more. The devices actuators enable them to take actions in response to the data they gather. For instance, an Internet of Things-enabled smart thermostat can gather temperature data and modify heating or cooling systems appropriately.

The network of mechanical, digital, and computer devices that can communicate with one another without the need for human-to-human or human-to-computer interaction. These devices are linked together by unique identifiers. These objects include cars, appliances, and other household appliances, as well as other items embedded with electronics, software, sensors, actuators, and connectivity that allows them to communicate and share data. This opens up possibilities for more direct communication between computer systems and the real world, for a deeper comprehension of the world, and for increased effectiveness, productivity, and decision-making.

IoT features such as sensor networks are crucial because of their inherent intelligence, which is further divided into two categories: object intelligence and network intelligence. The following features define a network's intelligence:

The communication standards of the layers connecting nodes to the internet should be open tostandardization for layers interacting with the physical world, such as tags and sensors.

Both object addressability—which gives an IP address—and multifunctionality—which allows a network designed for one application to be used by other applications—are importantfeatures.

Using the internet and cutting-edge technologies like real-time localization, embedded sensors,and near field communications, everyday objects can become intelligent and respond to their immediate surroundings. This is known as object smartness.

An actuator is a tool that changes the environment around it, much like the temperature controller in an air conditioner. At the network's edge or on a server, data can be remotely processed and stored. If data preprocessing is feasible, it is typically done at the sensor or another adjacent device. After processing, the data is typically sent to a distant server. An Internet of Things object's processing and storage capacity is also constrained by its available resources. Because of their limitations in terms of size, energy, power, and computational capacity, these resources are frequently severely limited. Apart from the challenges associated with collecting and organizing data, there are also communication problems.

Because Internet of Things devices are typically installed in geographically dispersed locations, their primary means of communication is wireless. The wireless channels are frequently unstable and exhibit high distortion rates. In this case, transmitting data consistently and with a minimal number of retransmissions is a significant issue, so communication technologies areessential to the research by IoT gadgets. Actuators allow us to directly alter the physical world, or we can something that is virtual. For instance, we can communicate with other intelligent items.

The physical world's capacity to accept a change often depends on its current state at that specific time. This is known as context awareness. Every decision is made with the circumstances in mind, as different applications may behave differently in different environments. For example, a person may prefer not to have his office interrupt him while he is on vacation. Sensors, actuators, compute servers, and the communication network are the essential parts of an Internet of Things architecture. However, there are numerous software related considerations. We first require a middleware in order to connect and manage all of these heterogeneous components. To link a vast array of devices, numerous standards are needed. Numerous fields, such as energy conservation, home automation, fitness, education, entertainment, social life, and health care, have applications for the Internet of Things. Here is a small and prescribed detail of the project:

- Problem: Due to their restricted movement and ability to communicate, individuals who are paralyzed have difficulty receiving continuous monitoring and must rely on time-consuming, sporadic evaluations. In order to provide proactive monitoring, current systems need real-time insights and IoT integration. A solution that provides early identification of patient demands and requirements, prompt action, and remote access to information about patients is required by caregivers. To bridge these gaps, Paramov is desperately needed. Paramov uses wearable sensors and Internet of Things (IoT) technology to facilitate remote monitoring, data processing, and alert generating in order to give real-time monitoring. Paramov aims to enhance patient outcomes and lessen caregiver stress by fusing contemporary technology with patient care.
- Solution: The solution is Paramov, a Smart Internet of Things-based Paralysis Patient Monitoring System (SIPMS) that mixes wearable sensors, wireless communication, and data analytics. Paralyzed patients can give SIPMS real-time physiological data, including their movement patterns and health indicators. Through IoT connectivity, this data is transmitted to a monitoring unit for analysis. The system instantly tells caregivers or medical specialists when it detects irregularities that indicate demands or requirements. Remote monitoring and access to historical data are made possible via user-friendly interfaces, like those found in smartphone apps. SIPMS offers proactive, continuous monitoring that raises the level of living and improves patient care while reducing the likelihood of issues.
- Benefits: Numerous advantages exist for raising the standard of care and boosting patients' general wellbeing when using an intelligent Internet of things-based paralyzed patient monitoring system. First of all, it makes it possible to continuously monitor vital signs and

gives healthcare professionals access to real-time data for prompt response in the event of any irregularities. By ensuring early identification of health conditions, this continuous monitoring lowers the likelihood of complications and readmissions to the hospital. Second, IoT-enabled system sensors can monitor patient motions and activity levels, enabling individualized rehabilitation plans catered to specific requirements. This encourages movement and active engagement in therapy sessions, both of which are critical for avoiding muscle atrophy and preserving joint flexibility. Furthermore, caregivers may keep an eye on their patients' health from a distance thanks to remote monitoring capabilities, which give patients and their families peace of mind and let medical staff use their resources more effectively. All things considered, an intelligent Internet of Things (IoT) paralysed patient monitoring system boosts patient safety, enhances treatment results, and encourages liberty and a good quality of life.

- Challenges: There were numerous challenges which has been faced during the project. It included managing power consumption so that the sensors and other devices consume less power to extend the battery life, there were certain challenges while integrating the hardware components to the cloud using the code as we needed to ensure that the real-time data reaches the cloud. While creating the application we have kept in mind that user experience has to be hassle-free so that they can easily trace their consumption level.

The coming era of technology will witness a growth beyond mobile internet to massive IoT within the next three to four years. The next generation will primarily focus on improving the functionality of critical communication use cases and speeding up data transmission. IoT will allow for quicker, more responsive, and real-time applications thanks to 5G connectivity. Edge computing will speed up the development of IoT and lower latency. Smarter, more independent gadgets and data-driven insights will result from the combination of AI and machine learning. Blockchain will increase trust and security in IoT. Our lives and the world will be better off as IoT is expanded into healthcare, smart cities, agriculture, and environmental monitoring. Privacy and security will be at the forefront of IoT development as interoperability and standardization become increasingly important. In the end, IoT promises to create a more efficient, sustainable, and connected world.

1.2. Purpose

An intelligent Internet of Things-based paralysed patient monitoring system is essential for improving the care and quality of life for paralyzed people. Such a system provides continuous tracking and handling of vital health data, such as temperature of the body, and movement patterns, by seamlessly incorporating the internet of things technology. With the help of this ongoing observation, medical professionals may quickly identify any departures from the usual, allowing for faster action and a lower chance of consequences.

In addition, the system may be set up to automatically notify caretakers or medical professionals in the event of an emergency, guaranteeing prompt aid when required. This intelligent monitoring system enhances patient outcomes and gives patients as well as their families peace of mind by offering proactive and comprehensive treatment that promotes independence and self-assurance in controlling the patient's health.

Moreover, the technology surmounts geographic obstacles by enabling remote monitoring, guaranteeing patients residing in remote or underprivileged locations access to specialised care and assistance. In the end, a sophisticated Internet of Things (IoT) monitoring system transforms the way paralyzed people receive healthcare, improving the quality of life, encouraging independence, and promoting proactive, individualized treatment.

This cutting-edge gadget combines technology, connectivity, and practical features to meet multiple important goals:

- Remote Monitoring: The technology allows for constant remote monitoring of body temperature and vital signs using Internet of Things (IoT) sensors and devices. Healthcare practitioners can take immediate action in the event of any anomalies thanks to real-time data transfer.

- Fall Detection: To improve safety and reduce the likelihood of accidents, fall detection is an essential component of a modern IoT-based monitoring system. With the use of cutting-edge sensors, the system watches the patient's every move and looks for patterns that could point to a fall. Upon detection of a fall caretakers or emergency services receive instant warnings, facilitating timely assistance. Furthermore, preventive steps based on forecasting can be taken to stop falls before they happen. Providing mobility assistance, modifying the surroundings, or alerting the individual to take safety measures are a few examples. The system enhances the overall health and quality of life of patients who are paralyzed and gives caregivers peace of mind by reducing the risk of fall-related injuries through the combination of real-time monitoring and predictive capabilities.

- Immediate Feedback: In a clever Internet of Things-based paralyzed patient monitoring system, instantaneous feedback guarantees prompt action and assistance. Real-time warnings and notifications are generated by critical movements or emergency scenarios, such as falls, with this function. For example, caretakers or healthcare providers are promptly notified by a smartphone application if the system notices abnormalities in their vital signs or patterns of motion. This feature allows for immediate feedback, which improves patient safety, permits quick emergency reaction, and makes proactive paralysis patient care easier.

- Enhanced Communication: In order to facilitate smooth contact among patients, family members, and healthcare practitioners, improved interaction is an essential component of an intelligent connected to the internet of paralyzed patient monitoring system. Patients who are paralyzed can effortlessly convey their requirements, worries, and preferences to healthcare providers and carers by using integrated communication services. Moreover, regardless of physical distance, caregivers and medical professionals can send real-time instructions, support, and status updates to patients via remote monitoring. Through enhanced communication, patients as well as their support systems can feel more connected and reassured, and early response in the event of an emergency is made possible. In addition to increasing the effectiveness of patient treatment, this system also improves the whole experience as well as the standard of life for people who are paralyzed by utilizing Internet of Things technology to foster better communication.

1.3. Scope

Offering complete assistance and care for people with paralysis, a contemporary Internet of Things-based paralysed patient monitoring system represents a groundbreaking development in healthcare. The standard of life and the safety of patients who are paralyzed are improved by this system, which makes use of a network of linked sensors and equipment to monitor several vital parameters, movements, and ambient elements in real-time. Smart IoT-based paralysed patient monitoring systems have wider applications than only treating individual patients; they can be used for population health monitoring and other more expansive healthcare projects. Patterns and trends can be found at the population level through the collection and analysis of data from several patients. This allows for better resource allocation, preventative measures, and healthcare monitoring. The scope of smart water bottles includes:

- Consumer Market: There is a lot of potential in the market for consumers for intelligent IoT-based paralyzed patient monitoring solutions. These technologies provide continuous tracking, analysis of information, and remote accessibility, giving caregivers peace of mind and improving patient quality of life as specialized health care and assisted living options gain more attention. The needs of patients who are paralyzed and their carers are specifically satisfied by functions like constant health data tracking, fall detection, prescription reminders, and emergency alarms. The potential of the market is further fueled by the expanding use of IoT devices and advances in sensor technology. Furthermore, the need for creative healthcare solutions specifically designed to address mobility concerns is expected to grow as the world's population ages. However, for broad acceptance, it will be essential to guarantee price, usability, and data security. In this emerging industry, partnerships between insurers, tech companies, and healthcare providers can guarantee sustainable growth and accelerate market penetration.

- Patients with Spinal Cord Injuries: People who have spinal cord injuries frequently become paralyzed and need to have their vital movements, skin health, and general well-being constantly monitored. Caregivers can receive continuous information on their condition from an intelligent IoT-based monitoring system, which can swiftly notify them of any irregularities or crises.

- Elderly People: Paralysis from stroke or neurological illnesses is more likely to occur as people age. An Internet of Things (IoT) monitoring system that not only gives their caregivers peace of mind but also guarantees the safety and independence of elderly folks can be very beneficial.

- Home Care Settings: A lot of people who are paralyzed would rather have treatment in the convenience of their own residences. IoT-based monitoring solutions allow medical practitioners to keep an eye on patients from a distance, preserving their independence while making sure they get help quickly when they need it.

An IoT-based paralysed patient monitoring system can be used by patients,caretakers, Through catering to the unique requirementsof these populations, this technology holds the potential to transform the way paralytic patient istreated and enhance the lives of those who are affected.

1.4. Internet of Things

We are aware that everything in the modern world is online, which makes life easier. The Internet of Things, or IOT, is one of the rapidly expanding technologies. It enables us to create web and mobile connected application, such as those for smart homes, healthcare, transportation, and other areas.

By utilizing cutting-edge technology to improve patient care, increase efficiency, and guarantee prompt interventions, an intelligent Internet of things-based paralysed patient monitoring system meets important needs in the healthcare industry. For patients as well as caregivers, paralysis—whether from strokes, spinal cord injuries, or other illnesses—presents many difficulties. Real-time data insights are frequently lacking from traditional monitoring systems, which causes response delays and less-than-ideal care. However, a paradigm shift in healthcareis taking place thanks to the introduction of the IoT (Internet of Thing) technology, which offersrevolutionary options for patient monitoring in paralysis.

A smart IoT-based system's capacity to offer remote, ongoing patient monitoring is one of its main advantages. IoT devices may collect data continuously, in contrast to traditional monitoring techniques that depend on sporadic examinations or stationary equipment. This allows healthcare providers to obtain real-time information independent of a patient's location. For those who are paralyzed, early identification of fluctuations in temperature of the body and movement is ensured by this constant monitoring. Healthcare practitioners can prevent future deterioration and improve patient outcomes by quickly identifying these problems and launching appropriate therapies.

Furthermore, an integrated perspective of the condition of the patient is provided by IoT-enabled sensors integrated into a range of medical devices and wearables. These sensors are capable of tracking environmental variables including humidity and temperature, as well as vital signs, movement patterns, and medication compliance. Through the process of gathering and evaluating this information, the system is able to identify minute variations in the patient's state, offering important insights into their general health. For example, changes in vital signs or departures from usual activity patterns may be signals of possible health hazards and may encourage medical professionals to take preventative action.

Moreover, intelligent Internet of Things technologies enable smooth communication and cooperation between healthcare stakeholders, such as patients, family members, and medical practitioners. A integrated methodology for patient care can be fostered by the safe sharing and real-time access to pertinent data made possible by cloud-based platforms and connected devices. Automatic alerts and notifications on the patient's condition can be sent to caregivers, guaranteeing prompt reactions to crises or pressing circumstances. Similarly, using real-time data analysis, healthcare providers can modify treatment regimens, offer individualized care suggestions, and remotely monitor patients' progress.

Wi-Fi-enabled monitoring systems are not only more cost-effective and operationally efficient, but they also provide better patient care. Healthcare institutions can improve resource efficiencyand productivity by automating data collecting and processing procedures. In order to improveresource allocation and lower avoidable hospital readmissions, predictive analytics algorithms, for instance, can foresee patient care needs. Remote monitoring capabilities also reduce the demand for personal consultations as well as hospital visits, which saves money for healthcareorganizations and patients alike.

Smart IoT technology also creates new opportunities for proactive and individualized healthcareinterventions. Utilizing machine learning algorithms as well as predictive analysis, the system is capable of analyzing substantial patient data sets to pinpoint patterns, patterns, and risk factors linked to issues related to paralysis. By customizing remedies and preventive measures to each patient's unique needs, healthcare providers can maximize therapeutic outcomes and minimizeunfavourable occurrences through the use of data-driven approaches. Furthermore, early intervention techniques, like telemedicine care or remotely rehabilitation sessions, can be swiftly adopted thanks to constant monitoring, which lessens the strain on medical facilities andraises patient satisfaction.

In conclusion, the deployment of an intelligent Internet of Things (IoT)-based paralysed patient monitoring system is a noteworthy development in the provision of healthcare, providing all- encompassing, individualized, and real-time patient care solutions. Through the utilization of data analytics, remote communication, and linked device capabilities, these systems enable medical professionals keep track of patients more effectively, work together more effectively, and take pre-emptive measures. Additionally, they facilitate the delivery of individualized care,increase operational effectiveness, and improve patient engagement, all of which eventually improve the quality of life and results for those who are paralyzed and their carriers. Smart Internet of Things (IoT) solutions will become more and more important in determining how patient tracking and management are shaped in the future as the healthcare industry continues to change.

1.4.1. Components of IOT:

- Devices and Sensors: The essential elements that connect the digital and physical realms are sensors and devices. Wearable technology and smart thermostats are two examples of devices that have sensors built in to record data from the real world, such as location, motion, and temperature. As the "eyes and ears" of the Internet of Things, sensors gather data from its surroundings. Following transmission and processing, this data is used for a number of purposes, including automation, data-driven decision-making, and remote monitoring. Devices and sensors are essential in turning commonplace items into intelligent, networked entities that improve our lives by offering convenience and new perspectives.

- Connectivity: The foundation of the Internet of Things is connectivity, which makes it easier for systems and devices to communicate with one another. Internet of Things (IoT) devices use various communication protocols and technologies, such as Bluetooth, Wi-Fi, cellular networks, and LoRaWAN, for data transmission. This makes it possible for smooth, real-time data exchange, which is necessary for data-driven decision-making, control, and monitoring. Users can access and interact with their IoT devices remotely thanks to connectivity, which guarantees that the devices can transmit information to the cloud or other data processing layers. It's an essential part that makes the Internet of Things function and opens up a world of possibilities for smart homes, healthcare, transportation, and other industries.

- Data Processing: In the Internet of Things, data processing is a crucial stage that converts unprocessed data from sensors and devices into insightful knowledge. Some techniques for analyzing, filtering, and organizing data—often in the cloud or at the edge—include data analytics, machine learning, and artificial intelligence. Real-time monitoring, pattern recognition, and decision-making are made possible by data processing, which also makes it possible for IoT applications to automate procedures, deliver useful information, and improve user experiences. It is crucial for deciphering the enormous amounts of data generated by Internet of Things (IoT) devices, encouraging innovation, productivity, and thoughtful decision-making across a variety of industries and uses.

- User Interface: User interface acts as a conduit between users and the massive network of data and connected devices. It consists of voice assistants, web dashboards, and mobile apps that let users communicate with and manage Internet of Things devices. This interface makes it easy for users to monitor and manage their Internet of Things ecosystem by giving them real-time access to device status, data, and settings. IoT technology is made more approachable and useful by user-friendly interfaces, which enable people and organizations to take advantage of connected devices' capabilities, automate processes, and make well informed decisions—all of which contribute to increased convenience and productivity in a variety of spheres of life and business.

- Security: Access control, encryption, secure firmware updates, and authentication are examples of IoT security techniques. Cyberattacks and privacy violations may result from vulnerabilities in IoT systems. Data confidentiality and integrity need to be upheld in order to safeguard people and organizations. In order to protect IoT devices and networks from potential threats and maintain their resilience and trustworthiness, strong security protocols are necessary. Security measures are essential to reduce risks and create a safe foundation for a networked, data-driven world as the IoT ecosystem grows.

1.4.2. Layers of IOT:

- Perception Layer: Real-world data is captured by the IoT's foundational component, the Perception Layer. Actuators, devices, and sensors are all part of it. Sensors gather information from the physical world, including flex, motion and temperature sensors. Actuators use this data to control their movements, such as locking doors or modifying thermostat settings. By processing data close to the source, edge devices can lower latency. This layer acts as a link between the digital and physical realms, gathering vital data that powers the Internet of Things ecosystem and allowing for data-driven insights, automation, and real-time monitoring.

- Network Layer: In the Internet of Things, the Network Layer enables data transmission to higher layers and device-to-device communication. It uses Bluetooth, cellular networks, Wi-Fi, and other communication technologies to manage connectivity between IoT devices, gateways, and central systems. Acting as go-betweens, gateways gather information from various sources and send it to data processing layers or the cloud. At this layer, edge computing can also take place to optimize data transfer and lower latency. By establishing connections between IoT devices and enabling real-time monitoring, control, and analysis, the Network Layer guarantees effective and secure data transmission.

- Middleware Layer: The middleware layer acts as a crucial bridge connecting the upper and lower layers. It prepares data gathered by IoT devices for analysis and action by processing, organizing, and managing it. This layer provides security features like encryption and authentication, filters and aggregates data, and converts protocols to guarantee compatibility. By combining data from different sources, middleware offers a thorough understanding of the IoT ecosystem. In the end, it enables real-time monitoring, automation, and data-driven decision-making within the IoT architecture by converting raw data into insightful knowledge and making it available for applications and services.

- Application Layer: Data is used practically at the Application Layer. It consists of a number of software programs and applications that use the information gathered from lower layers to provide automation, user interaction, and insights. This layer can include sophisticated software systems, web-based dashboards, and mobile apps. It permits real-time IoT device monitoring, analysis, and control, empowering users to act and make decisions based on data. The Application Layer enhances convenience and efficiency across many domains by enabling a broad range of Internet of Things applications, from industrial automation to smart home control.

1.4.3. IOT Network Technologies:

LPWAN networks are starting to adopt NB-IoT as the standard. These technologies facilitate intelligent decision-making in a variety of industries by enabling real-time monitoring, control, and automation. IoT networks improve the delivery of healthcare by enabling smart medical devices and facilitating remote patient monitoring. Field sensors collect data for precision farming in agriculture, which maximizes crop yields and efficient use of resources. By linking systems and machinery, industrial IoT (IIoT) networks transform manufacturing and enable streamlined operations and predictive maintenance. IoT is used by smart cities to effectively manage traffic, dispose of waste, and save energy. IoT network security and scalability are critical for maintaining data integrity and the ecosystem's continued expansion. IoT network technologies, taken as a whole, redefine connectivity and provide a basis for increased productivity, creativity, and overall quality of life in a variety of fields.

- LPWAN: For Internet of Things (IoT) and machine-to-machine (M2M) applications, Low Power Wide Area Network (LPWAN) technology is a wireless communication protocol that enables long-range communication with low power consumption. LPWANs are especially well suited for tying together devices that must send modest amounts of data over long distances while using a lot of battery power. Large-scale geographic coverage is one of LPWANs' primary advantages, which makes them perfect for uses in smart city, smart agriculture, and industrial IoT applications. Lower frequencies—typically in the subGHz range—are used by LPWANs to accomplish this, enabling signals to traverse larger spaces and more successfully pass through obstructions like vegetation and buildings. Many LPWAN technologies exist; two of the most well-known ones are Sigfox and LoRaWAN technology . For example, LoRaWAN uses unlicensed frequency bands to provide flexibility and cost-effectiveness while enabling long-range communication through the use of spread spectrum modulation. Opting in the unlicensed Industrial, Scientific, and Medical (ISM) bands. Because of its low consumption of energy, LPWAN technology is unique in that it enables connected devices to run for years on tiny, cheap batteries without needing to be replaced. This feature is essential for Internet of Things applications where devices might be placed in difficult-to-reach or remote areas.

- Cellular: Through a network of connected base stations, cellular network technology provides wireless communication over a large geographic area for both voice and data transmission. The 4G, 5G, and 2G cellular networks are currently the most widely used. In addition to supporting, these networks offer wireless communication services to billions of users worldwide. These applications include web surfing, video streaming, and various Internet of Things devices. A base station or antenna that serves each of the cells that make up a cellular network's coverage area. Mobile devices smoothly transition between these cells as they move, keeping their connection open at all times. The radio frequency band can be used more effectively thanks to the cellular architecture, which permits interference free reuse of the same frequencies across various cells. The newest cellular technology, 5G, promises even faster data speeds, reduced latency, and more network capacity. To achieve these improvements, it introduces new technologies such as millimeter-wave frequencies. The Internet of Thing (IoT), which is expected to have more and more connected devices, will depend heavily on 5G to support and enable features like augmented reality and driverless cars. Reliable and secure transmission is ensured by cellular networks operating on licensed bands of frequency assigned by regulatory authorities. Cellular technology is a mainstay of contemporary telecommunications due to its broad adoption, which has revolutionized communication and information access. Cellular networks will be essential in determining the direction of connectivity in the future and in facilitating the development of novel applications and services as technology advances.

- Bluetooth Low Energy: The wireless communication technology known as Bluetooth Low Energy (BLE), or Bluetooth Smart, is intended for short-range and power-efficient data transfer. It is a feature of Bluetooth 4.0 and later versions, and its low power consumption makes it perfect for a wide range of applications, especially those involving wearable technology and the Internet of Thing (IoT). In order to minimize interference from other electronic devices operating in the same frequency range, BLE uses a frequency hopping spread spectrum. BLE functions in the 2.4 gigahertz band. BLE's capacity to communicate over short distances—typically up to 100 meters—while using a lot less power than conventional Bluetooth is one of its primary characteristics. Devices that run on batteries, like monitors for fitness, smartwatches, medical equipment, and different types of Internet of Things sensors, are a good fit for BLE. It allows these gadgets to communicate with mobile phones, tablets, and other devices that are compatible without rapidly depleting their batteries. Fewer packets of data, lower transmitting power, and the option to switch to low-power sleep patterns when not actively broadcasting data are some of the power-saving techniques used to accomplish this. BLE's simplicity as well as ease of integration are also important features. The central, which looks for and communicates to peripherals, and the peripheral devices, which broadcasts data, are its two main functions that it supports. Every new version of BLE brings improvements in terms of features, range, and data transfer rates. BLE has been evolving continuously. The technology's adoption has spread across numerous industries and is now considered acceptable for short-range wireless communication. Bluetooth low-energy technology is a short-range and low power consumption wireless communication standard. Its effectiveness, adaptability, and interoperability have rendered it a popular option for many applications, facilitating seamless communication in the quickly expanding wearable and Internet of Things ecosystem.

- Zigbee: Wireless communication technology known as Zigbee was created with short-term, low-power systems, and low-data-rate applications in mind. Operating on the IEEE 802.15.4 protocol, it is specifically designed for devices that have low power requirements, which makes it ideal for use in industrial automation, automation of homes, healthcare, and other Internet of Things (or IoT) applications. The ability of Zigbee to establish a mesh network that enables communication between devices even when they are not in close proximity to one another is one of its primary features. Every device, or node, in a Zigbee meshes network has the ability to function as a router by sending signals to other nodes, increasing dependability and extending the network's overall range. Zigbee networks are more resilient and adaptable thanks to this mesh topology, which lowers the possibility of communication failure by allowing devices to identify multiple communication paths. Zigbee uses the unlicensed, worldwide 2.4 GHz frequency for its operations. This band is divided into 16 channels, and in order to prevent interference from different wireless technologies using the same frequency range, Zigbee devices randomly select channels. Zigbee devices are able to run on small batteries for a longer period of time because of the use of low-power devices, short-range transmissions. While devices in a tree topology may interact with one another and transmit information through intermediary nodes, devices in a star topology interact via a central coordinator. As was already mentioned, the mesh topology is especially useful in scenarios where sensors may be dispersed over a large area. The application profiles for Zigbee specify how particular types of devices are supposed to interact with one another on the network. This guarantees compatibility between gadgets made by various manufacturers. For instance, there are profiles for automation of homes and lighting control called Zigbee automation and Zigbee Smart Link, respectively. One important feature of Zigbee networks is security. To secure data during transmission, the technology uses strong encryption techniques like Advanced Encryption Standard (AES). Zigbee further enhances a secure transmission environment by supporting features like network key management and device authentication. Zigbee is a low-power, short-range wireless communication technology designed for Internet of Things applications. Its energy-efficient design, mesh network formation capabilities, support for multiple network topologies, and application profile definitions make it a popular option for IoT applications such as healthcare, home and manufacturing automation, and others where dependable and energy-efficient transmission is crucial.

- NFC: A short-range communication method known as near field communication, or NFC, allows devices to communicate with one another when they are brought close together—typically within just a few centi-meters. NFC is suitable for a range of applications, such as cashless transactions, transmission of data, and identification. It works on the concepts of electromagnetic induction & is designed to be simple and convenient. NFC is based on an inductive connection of two coils, one for each communication device, and operates at a frequency of 13.56 M-Hz. Active and passive are the two main ways that NFC communication occurs. Active Mode: This mode enables bidirectional data exchange between the two devices by enabling them to produce unique radio frequency fields. Peer to peer communication via this mode is frequently used to share files between cellphones or establish connections with other NFC-enabled devices. Passive Mode: In this mode, the target reacts while the initiator, or one device, creates the radio frequency field. This mode is frequently utilized in situations where data collected by NFC tags is accessed or contactless payments are made. The simplicity and usability of NFC is one of its main benefits. All it takes to connect two devices is to bring them close to one another; complicated setup or pairing procedures are not needed. Because of this, NFC can be used in a range of scenarios where seamless and rapid interactions are crucial. Contactless payment systems, which allow customers to make purchases by tapping their smartphones or NFC-enabled cards on point-of-sale terminals, make extensive use of NFC technology. NFC is used by mobile payment apps like Samsung Pay, Apple Pay, and Google Pay to facilitate safe and easy transactions. Transferring data among devices also uses NFC. When two NFC-enabled devices are brought close to one another, for example, users may exchange contact details, images, or other files. This function is frequently used to make Bluetooth pairing and Wi-Fi setup easier. NFC is also used in access control and identification applications. You can use NFC cards or tags to gain secure access to events, public transportation, and buildings. These tags have the capacity to hold data that NFC capable devices can read to authorize access or deliver pertinent data. One important feature of NFC technology is security. NFC data transmissions may be encoded to guard against unwanted access. NFC communication's short range also provides an extra degree of physical security because successful data exchange requires close proximity between devices.

- RFID: Radio waves are used in radio-frequency identification, or RFID, technology to transfer data between an RFID tag and a reader. It is widely used in many different applications to track and identify people, animals, and objects. The three primary parts of an RFID system are the radio frequency identification (RFID) tag, the reader for RFID, and the database or backend system. RFID tags are tiny electronic devices made up of an antenna and a microchip. Information is stored in the microchip, and radio frequency signals are used by the antenna to communicate with RFID readers. There are two varieties of RFID tags: passive and active. Although tags that are active have their own electrical power and can transmit signals over greater distances, passive tags depend on the power that is emitted by the reader that reads RFID tags for sending data. RFID devices are gadgets that communicate with tags with RFID tags by emitting radio frequency signals. The information on the tag's microprocessor is read by the reader, which then has the ability to send it to a desktop or backend device for further processing. Depending on the needs of the application, sensors may be fixed or portable and come in a variety of frequencies. The data gathered by RFID readers is processed and managed by the backend system. The system in question could be an inventory control system, a system that controls access for security reasons, or a database that holds data regarding the tagged items. Numerous sectors and uses of RFID technology exist, including: This technology is widely used in logistics and supply chain operations to track and manage inventory. It lowers errors and boosts overall efficiency by providing immediate insight into the transportation of goods. RFID is used in retail to improve the overall shopping experience, prevent theft, and manage inventory. RFID tags are used by retailers to monitor the flow of merchandise from the storage facility onto the store shelves. RFID tags or cards are frequently used for parking lots, events, and building access control. They offer an easy and safe method of controlling entry and departure. In healthcare settings, RFID is used for tracking assets, medication management, and patient tracking. It enhances effectiveness and lowers mistakes in patient care.In agricultural and management of wildlife, RFID chips are employed for identifying animals and tracking. They let farmers and researchers keep an eye on the whereabouts and well-being of their animals.

- Wi-Fi: Wi-Fi, an acronym for broadband connectivity, is a popular wireless technology that eliminates the requirement for physical wires when connecting devices to the World Wide Web and neighborhood networks. It functions using the IEEE 802.11 series of standards and is now a crucial component of contemporary networking, offering wireless access in a variety of settings, including homes, workplaces, and public areas. Among WiFi technology's salient features are: Typically, routers or wireless access points are used to set up Wi-Fi networks. By using wireless technology, these devices send and receive data, establishing a LAN, or local area network, that enables connections and communication between numerous devices. The 2.4 GHz & 5 GHz bands of frequencies are used by WiFi. Although the 2.4 GHz band has a wider frequency range, it is more vulnerable to interference from devices that are using the same band. Applications requiring quicker and more dependable connections can benefit from the 5 GHz band's higher rate of data transfer and lack of congestion. Data transfer rates are specified by Wi-Fi standards, and performance and speed are improved with every new generation of Wi-Fi standards. the 802.11b 802.11g, 802.11n, 802.11ac, & the most recent norm, 802.11ax (Wi-Fi 6) are examples of common Wi-Fi standards. Specifically, Wi-Fi 6 brings new features like faster data transfer speeds, more effectiveness in congested areas, and enhanced multi-device support. Security protocols are used by Wi-Fi networks to safeguard data while it is being transmitted. In order to safeguard wireless communication and prevent unauthorized access to the network, encryption standards such as WEP encryption have been developed. MIMO technology improves Wi-Fi performance by using multiple antennas for simultaneous data transmission and reception. As a result, network efficiency rises and data transfer rates are enhanced. The IEEE establishes Wi-Fi standards, which guarantee compatibility between gadgets made by various manufacturers. Users can now easily connect a variety of Wi-Ficapable devices thanks to this. With the ability to utilize the internet as well as network resources via a wide range of gadgets, including tablets, smartphones, laptops, smart TVs, and Internet of Things devices, Wi-Fi connectivity has become widely used. Wi-Fi has been integrated into a number of industries, include healthcare, education, hospitality, and manufacturing, due to its widespread adoption and ease of use. Even though Wi-Fi has many benefits in terms of portability and user-friendliness, issues like congestion in the networks and security worries are still being addressed by constant improvements in Wi-Fi protocols and technology. The direction of wireless connectivity and communication in the future will probably be greatly influenced by the development of Wi-Fi.

- Ethernet: A popular and widely accepted method of wired local area networks (LAN) is Ethernet. It offers a dependable and effective way for devices connected to a local network to talk to one another and access the wider internet. The International IEEE 802.3 norm, which outlines both the physical as well as data connection-layer requirements for wired LANs, governs Ethernet technology. Among Ethernet technology's salient features are: Multiple physical media, such as coaxial, fiber optic, and twisted-pair copper cables, can be used to operate Ethernet. Twisted-pair wires with RJ45 connectors are the foundation of the most widely used form of Ethernet in use today. Applications requiring high bandwidth and longer distances are served by Ethernet instead of fiber optics. Ethernet provides a frame-based protocol for communication and functions at the OSI model's data link layer. The data payload, type or length subject matter, destination and source MAC addresses of the data, and a cyclical redundancy check (CRC) enabling error detection are all included in an Ethernet frame. When Ethernet first came into use, several devices shared a single communication path via a shared medium called a bus topology. Data corruption could result from collisions between two devices transmitting data at the same time. Collision domains were divided with the advent of Ethernet switches, enabling full-duplex in interaction among devices and doing away with collisions. With the initial ten megabits per second (Ethernet), 100 megabits per second (Fast Ethernet), one gigabit per second (Gigabit Ethernet), ten gigabits per second (Gbps) (10 Gigabyte Ethernet), 75 Gbps, 40 Gbps, and 100 Gbps, to the latest technologies enabling speeds of 400 Gbps, Ethernet supports a variety of data transfer speeds. The specific needs of the network determine which speed to choose. Power lines can be used to extend Ethernet in addition to conventional wired connections. The transfer of data over existing power lines is made possible by this technology, which is called Ethernet over Powers Lines (EoPL). This offers an alternative for connecting to networks in places where running fresh wires may be difficult. Ethernet cables can now transmit data and electricity thanks to a feature called PoE. With this, there is no longer a need for separate power cables for gadgets like IP cameras, VoIP cellphones, and wireless access points. The majority of wired LANs are built on Ethernet, which provides a dependable and affordable way to connect devices inside a limited geographic area. It is frequently utilized in many industries, homes, and data centers. Ethernet keeps developing to accommodate faster data transfers, more efficiency, and enhanced features to satisfy the ever-increasing needs of contemporary networking.

1.4.4. INTERNET LAYER IOT NETWORK TECHNOLOGIES:

For efficient communication and data exchange among a broad spectrum of networked devices, IOT technology require an Internet layer. As the foundational layer of the Internet Protocol (IP) suite, it guarantees that data packets can pass through a variety of networks, allowing devices to connect with each other anywhere on the planet.

The large number of devices requiring unique IP addresses in the IoT context makes IPv6, a crucial part of the Internet layer, even more important. Every IoT entity has a unique identifier thanks to its extended address space, which supports the exponential growth of connected devices.

Additionally, regardless of the underlying network technologies, heterogeneous IoT devices and systems can communicate thanks to the Internet layer's promotion of interoperability. The Internet layer provides a uniform and global language for data transmission regardless of the device's connection method—Wi-Fi, cellular networks, or other protocols.

To safeguard confidentiality and data integrity in communications over the Internet of Things,protocols like Datagram Transport Layer Security (DTLS) are used at this layer, where securityconsiderations are crucial. To put it simply, the Internet layer is the central component of IoT network technologies, allowing for a unified and globally interconnected ecosystem that supports the Internet of Things' scalability and functionality. This layer is connected to commonly used IoT technologies.

- IPv6: IP addresses are used at the web layer to identify devices. IoT applications usually use IPv6 instead of legacy IPv4 addressing. IPv6 uses 128 bits and can provide 2 128 addresses (about 3.4 × 10 38 or 340 billion billion billion billion) addresses, whereas IPv4 can only provide 32-bit addresses, which can only provide about 4.3 billion addresses overall—less than the current number of connected IoT devices. Not every IoT device actually requires public addresses. Tens of billions of devices are anticipated to connect to the Internet of Things over the course of the next several years, many of which will be installed in private networks with private address ranges that only use gateways to communicate with other devices or services on external networks.

- 6LoWPAN: All thanks towards the IPv6 Low Power Cellular Private Area Network (6LoWPAN) standard, IPv6 may be employed over 802.15.4 wireless connections. Wireless sensor networks frequently use 6LoWPAN, and home automation devices use the Thread protocol, which is also run over 6LoWPAN.

- RPL: The Internet Layer contains routing. The Routing Protocol for IPv6 in Low-Power and Lossy en Networks (RPL) is designed to route traffic using IPv6 over low-power devices networks, like those constructed with 6LoWPAN. RPL, which is pronounced "ripple," is intended to route packets in networks with constraints, like wireless sensor networks, where packet loss is high or unpredictable and not every device is always reachable. Using dynamic metrics and constraints such as minimizing latency or energy consumption, RPL can construct a graph of the network's nodes and then calculate the optimal path.

1.4.5. APPLICATION-LAYER IOT NETWORK TECHNOLOGY:

IOT apps make extensive use of HTTP and HTTPS interfaces. HTTP and HTTPS are widely used in internet-based applications as well. Constrained Application Protocol, or CoAP, is frequently used in conjunction with 6LoWPAN over UDP. It functions similarly to a lightweight version of HTTP. In IoT applications, messaging protocols:

- MQTT: Through the use of the publish/subscribe paradigm, MQTT enables decoupled device communication. According to this model, devices can function as subscribers who receive messages or as publishers who send messages. By controlling message distribution, the broker—the central entity—ensures effective communication amongst devices. MQTT has several advantages, one of which is its low bandwidth and resource requirements. Due to its header-only design, the protocol minimizes data exchange, which makes it perfect in scenarios where network bandwidth is expensive or scarce. Because IoT devices frequently operate in environments with limited power and processing capabilities, this efficiency is especially important. Moreover, MQTT offers Quality of Service (QoS) tiers to ensure message delivery. The two QoS levels offer different levels of assurance. Level 0 denotes a maximum of one delivery, Level 1 guarantees a minimum of one delivery, and Level 2 offers a precise delivery, thereby enhancing communication process reliability. Furthermore, MQTT can adjust to different network architectures due to its intrinsic flexibility. With its support for both secure and non-secure connections, it can be used in a variety of contexts. Mechanisms that ensure the confidentiality and integrity of the data being transferred, such as Secure Sockets Layer (SSL) and Transport Layer Security (TLS), are frequently used to implement security. To summarize, MQTT is a key component of Internet of Things communication due to its efficiency, lightweight design, and support for a publish/subscribe model. Its broad industry adoption attests to its dependability in coordinating the smooth transfer of data in the networked Internet of Things world.

- AMQP: A robust, open-standard messaging protocol called AMQP was developed to enable compatible, dependable system-to-system communication. In contrast to certain other messaging protocols, AMQP is not restricted to a particular application domain and can be utilized in a variety of contexts, such as financial services, enterprise messaging, and the Internet of Things (IoT). AMQP functions primarily through a client-server architecture. The producer, who is in charge of sending messages, and the consumer, who is in charge of receiving them, are the two main parts. An organization that serves as a middleman and facilitates communication between producers and consumers is called a message broker. This broker, which is essential to the AMQP ecosystem, makes sure that messages are exchanged in a dependable and organized manner. AMQP is renowned for its ability to accommodate a wide range of messaging patterns. It supports both the publish/subscribe model and point-to-point communication, but it mostly adheres to the message queue paradigm, in which messages are held in queues until they are read by a recipient. Because of its adaptability, AMQP can meet a variety of communication requirements. Message integrity and dependability are highly valued in AMQP. It defines various modes of delivery, enabling messages to be designated as either transient or persistent based on whether or not they should withstand system failures. This guarantees the persistence of important information even in the event of unforeseen disruptions. Additionally, AMQP ensures that systems can communicate synchronously and in both directions by supporting a variety of communication patterns, such as request/reply. This is especially helpful in situations where prompt replies and acknowledgements are essential. An additional fundamental component of AMQP is security. In order to safeguard data integrity and confidentiality while it is being transmitted, the protocol can be implemented over secure channels like Transport Layer Security (TLS). It also includes mechanisms for authorization and authentication. The strengths of AMQP are found in its adaptability, consistency, and focus on facilitating safe and systematic communication among various systems. Its importance in contemporary messaging architectures is highlighted by its broad adoption in enterprise environments and its industry-wide applicability.

- XMPP: An open-standard communication protocol for real-time messaging, presence data, and teamwork is called XMPP. Jabber was the original name of this flexible protocol, but XMPP is now widely used for voice and video calls, instant messaging, and other collaborative applications because of its extensibility and decentralized design. Fundamentally, XMPP functions through a client-server architecture, in which clients are applications used by end users and servers act as messengers. The unique feature of XMPP is its federated architecture, which facilitates smooth communication between users on various servers. XMPP serves as the basis for a global, networked messaging system because of its decentralized model, which encourages interoperability. The fact that XMPP supports presence information is one of its main advantages. Users have the ability to express their real-time communication status, availability, and willingness. This feature, which gives users insight into their contacts' online presence, is essential to many instant messaging and collaboration apps. Because of XMPP's high degree of extensibility, new features can be integrated using extensions or XEPs (XMPP Extension Protocols). The key to XMPP's flexibility is its extensibility, which allows the protocol to change and grow to meet new demands for communication. XMPP is versatile due in part to its extensions, which enable multi-user chats and file transfers. The top priority in XMPP is security. The data integrity and confidentiality can be guaranteed when using the protocol over encrypted connections. Moreover, end-to-end encryption, user authentication, and authorization are supported by XMPP, offering a secure communication platform. Because XMPP is opensource, a thriving ecosystem of both proprietary and open-source clients and servers is supported. This variety adds to the protocol's adaptability and broad use in a variety of industries, such as social networking, healthcare, and the Internet of Things, where realtime communication is crucial. With its emphasis on security, extensibility via XEPs, decentralized architecture, and support for presence information, XMPP is a robust and adaptable protocol that serves as the foundation for real-time communication and collaboration in a wide range of applications.

CHAPTER 2

LITERATURE REVIEW

We were able to get a basic understanding of the technologies that were in use earlier thanks to a thorough analysis of the articles that were previously published about health monitoring system for paralytic person. We were given a brief overview of the improvements that need tobe made in the next iteration of the health monitoring system for paralytic person innovation by the architecture diagram, sensors, devices, and modules. A few of the problems we ran intowhen developing the Smart IOT based Paralysis Patient Monitoring System were with its design and system architecture, components, technology used, functioning, and impact on patient needs and habits. The publications that we consulted when creating the Smart IOT based Paralysis Patient Monitoring System are really helpful to us. Each article plays a crucial function because it has greatly aided us in keeping track the health of patients, reminding us about the technology used for alerting the caregivers, etc.

People who are severely paralyzed can express their needs and thoughts thanks to the suggested system. Additionally, it supports the patient in realizing their intellectual potential, which occasionally resolves the mental illness that the physician has identified. Any patient, regardlessof age, can effortlessly operate the novel and distinctive user interface offered by the suggested patient. The suggested approach[1] creates a novel method for detecting eye movements and blinks by combining several current techniques. To identify motion to the left, right, or not at all, utilize the Eye Motion Algorithm. In the suggested system, the Eye Blink Detection Algorithm is employed to identify whether eye blinks are voluntary and involuntary. The patientcan communicate and navigate the suggested system more effectively with the aid of these twoalgorithms. Utilizing a consumer-grade PC/laptop and a $23.53 Logitech webcam, the suggested system lowers costs while increasing usage in a variety of settings, including homes, government and private hospitals, and personal care homes. Because there is no need for specialized labor for system setup or maintenance, the system will cost less. The primary disadvantage is that the system's accuracy decreases in low light, and the patient's eye must always be in line of sight with the webcam.

Internet of Things enabled Being in a distant location allows paralysis patient health care [2] tobetter understand the patients' demands and state of health. As a help for those afflicted with paralysis, this system model plays a crucial part. The messages are stored in the IoT cloud, which also provides a summary of the patient's health over time. Additional advancements couldbe achieved by incorporating gyro and flex sensors, which would expand the systems' potential uses. A flex sensor can be used to identify the motion generated by individual fingers, and a gyro sensor can measure the hand's rotational movement. Additionally, the GSM module can be utilized to send SMS messages to the registered individual, conveying the messages from paralyzed patients.

This work proposes a comprehensive diagnosis system based on S-band wireless status information for non-contact identification of aberrant human gaits. The suggested system can aid in the early identification [3] of anomalies in gait, which is crucial for clinical diagnosis. Four human gait patterns—normal, festination, small stride, and turn—were the principal subjects of our attention. Wavelet transform is used to filter the noise in wireless data, and hamper filters are used to identify outliers. Once the data image is obtained, it is linearized, resized, and chopped to the appropriate size. Because of SVM's excellent precision and resilience across a wide range of conventional machine learning situations, it was chosen. It was utilized for both multi-class and binary classification. Analyses comparing the use of the image processing module with its non-use show that it can increase system performance by at least 5%. Our method's multi-class accuracy rate of 96.7% shows that it is an effective, affordable, and dependable way to automatically detect and diagnose paralysis agitans based on their aberrant gaits. It is necessary to continue working toward improving detection speed in the future.

When a patient reaches the paralysis stage, it is very difficult to predict when they might become conscious again. The paralysis could last for several days, weeks, months, or even years at a time. In these situations [4], it becomes challenging for hospital staff to continuously watch and monitor the patient, and minor body movements, reasonable indicators, and abnormal activitiesmay go unnoticed.

People's concerns about their health have increased significantly in recent years due to the risein ailments that occur daily. Therefore, it is crucial to keep an eye on the health. The patient willhave continuous physical monitoring, and the doctor, seated in front of a computer screen, willbe able to access information regarding the patient's status. The doctor will receive an alert through mail if the patient's condition deviates, allowing him to diagnose the issue and maybe save the patient's life. This project's primary goal[5] is to periodically update the doctor on the patient's health status so that, in the event of an anomaly, the doctor can act appropriately right away.

With an aging population, the proposed system exposes the flaws in our current healthcare system and offers a fast, secure, portable, and inexpensive solution by utilizing technically sound devices (sensors, Arduino, Raspberry Pi, and IoT) and systems (IoT, network topologies). Even though this system is a useful tool for remote patient care, it can yet be improved in the future to increase its effectiveness. The system [6] can be made more user-friendly for patients by including a video chat between the patient and the doctor. Message alerts can be generated by a GSM/GPRS module anytime patient sensor data approaches abnormalities. One's life will become more convenient and independent with the addition of additional health related sensors.

Since health care administrations constitute a vital component of modern society [7], computerizing these services eases the burden on individuals and streamlines the process of measurement. Furthermore, patients can depend on this system because of its ease of use. The purpose of developing such a system is to save healthcare expenses by reducing hospital stays, doctor visits, and diagnostic testing. The suggested system can be further enhanced and made more adaptable by adding additional propelled sensors, for example. With the aid of several sensors, the system is anticipated to track and perceive continuing (real-time) information and contribute to improving the quality of healthcare.

This essay addresses the value of a communication system for individuals who are paralyzed [8] and emphasizes its advantages, including the ability to express ideas and enhanced quality of life. Future prospects for an Internet of Things-based paralysis healthcare system are also mentioned. It is recommended that machine learning algorithms be integrated in order to learn from user data, improve proactive treatment, and create health forecasts. It is suggested that the system be integrated with other medical devices, like wheelchairs and smart beds, to offer complete health monitoring and support. It is said that more sophisticated sensors, including as pressure and EEG sensors, are being developed to give paralyzed patients more precise and in-depth health data.

Users can identify their health parameters using an Internet of Things-based solution, which may assist them manage their health over time. Patients may eventually seek medical attentionif necessary. They could quickly and conveniently share with the doctor their health parameterdata through a single application. As far as we are aware, one of the most sought-after options for health monitoring is the Internet of Things. The created [9] Arduino-based Internet of Things health monitoring system is the subject of this study. The device will use Bluetooth to transfer the patient's blood SpO2 levels, pulse rate, and body temperature readings to an app. An IOT- based health monitoring system built on an Arduino platform has been developed. This means that the proposed system has three axes of measurement capability: blood oxygen levels, heartrate, body temperature, and hand muscle activity. Every sensor's measured value is sent to the controller, to which the node mcu is wirelessly attached. The data will be gathered and sent to the bio app, which displays the specifics of the patient's health. There will be a more sophisticated approach in the future that allows one to view the body's entire functioning via a smartphone.

All patients' lives would be made easier by an Internet of Things-based real-time health monitoring system, since it would allow them to measure various health markers independently. Without going to the hospital, patients can transmit the results they received from the device to their physician. This gadget is intended to save costs and time while also lessening the possibility that a patient's condition would worsen as a result of travel. In order to conduct medical examinations digitally and provide the results on an OLED screen and on their mobile app, the device requires 5 inputs from the patient. Because this technology [10] yields results instantly, patients' wait times will be shortened. Everyone can afford the equipment because it is economical and produces results quickly.

The project's outcomes are displayed in a chart format; while the values may not exactly matchthe original values, they nonetheless adhere to the requirements of their respective areas. Healthconditions have been successfully monitored by the Remote Health Monitoring project module [11]. Everyone benefits from it in their everyday practice of health consciousness. It facilitates the assessment of health at any age, from toddlers to senior citizens. This module offers a pleasant experience with a variety of sensors while cutting down on patient time and hospital visits. Present-day futuristic scopes are typically a unified unit that takes up less room, is easier to operate in any kind of location, and doesn't compromise performance, particularly when usedfor outdoor monitoring.

There is now a real-time, portable gadget available. It is applicable to everyday medical uses. The suggested system makes use of a number of sensors to record the patient's vital signs and related biomarkers. It then makes this data available locally or remotely. The NodeMCU microcontroller is utilized for data processing and communication to the Firebase server, which facilitates the delivery of health records. The designed system allows for both local and remotemonitoring. For local monitoring [12], readings and plots are sent to an application for mobile devices within the local network's range or displayed on the built-in display. For remote monitoring, readings are sent to the Firebase cloud-based server. In the latter method, a reliable medical facility houses the remote monitoring dashboard. The suggested system's results were contrasted with those obtained from a commercial medical instrument in order to verify the measurement accuracy. It is shown that there is close relation between the two devices' measurements. The HR, blood oxygen level, and body temperature readings had the greatest recorded error rates—2.67%, 2.04%, and 1.58%, respectively. Furthermore, there was strong agreement between the linear regression test and the Bland-Altman plot in comparison to the reference data. This demonstrates the efficacy and suitability of the suggested approach for use with patients and senior citizens in their homes and medical facilities.

In this paper, we have proposed [13] and implemented a Smart Health Monitoring System. It is working successfully. By using biomedical sensors, we saved patient's data viz. temperature and heart beat rate in SD card. The data is further uploaded in the server. We also developed anandroid application named s-Health. In this app patient can see nearby hospitals, home remedies, use medicine reminder and doctors' can see their patients' health parameter in s- Health application to diagnose the results sitting far away from the patients. For future work, we can increase the functionality of system by adding more sensors and by making our app more dynamic in terms of nearby hospitals and home remedies.

We have developed and deployed a smart system for health monitoring in this study. It is operating effectively. We recorded the patient's temperature and heart rate on an SD card using biomedical sensors. The information is then added to the server. Additionally, we created the Android app s-Health. Patients can use this app to view hospitals in their area, try home treatments, and set up medication reminders. Doctors, on the other hand, can view their patients' health parameters in the Health application and diagnose patients from a distance. In the future [14], we can improve the system's usefulness by including additional sensors and enhancing the dynamic nature of our software with regard to neighbouring medical facilities and at-home treatments.

We attempt to provide a quick description of the issue facing paralysis patients and the many approaches that might be employed in the survey study. While using a wheelchair as a substitute is easy, maintaining its effectiveness is challenging and costly. According to the survey [15], using an Android application can be done in place of using a wheelchair. The primary benefit of utilizing an Android application is its low cost and its ability to be utilized anywhere withoutcausing any harm. It takes less time to utilize the Android application.

This article provides a thorough analysis with the Internet of Things, or IoT, in healthcare, including its applications, advantages, and drawbacks [19]. The results demonstrate how the Internet of Things (IoT) is revolutionizing the healthcare industry by providing tailored therapies and early diagnosis, increased operational efficiency, and easier access to evidence-based decision-making. Notwithstanding the apparent advantages, problems with privacy, security, and interoperability still exist and need for more study and development of technology. The possibility of IoT to transform healthcare delivery and make it more patient-centered, efficient, and accessible is highlighted in this paper.

CHAPTER 3

PROPOSED METHODOLOGY ON PARALYSIS MONITORING SYSTEM

An Internet of Things-based paralysis monitoring system uses sensors to monitor a paralyzed person's mobility, muscle activity, and vital signs. These sensors gather data in real time and send it wirelessly to a cloud platform or central hub. The system can identify abnormalities, notify caregivers of possible health problems or emergencies, and offer information about the patient's state for individualized therapy through data analysis. IoT makes it possible to monitor patients remotely, improving their autonomy and enabling quick action when needed. For those who are paralyzed, this technology promotes ongoing, individualized healthcare management, enhancing their standards of life and giving caretakers peace of mind.

3.1. Hardware and Methods

The suggested methodology's innovation consists of hardware elements such as Wi-Fi modules and sensors. Together, these parts track the temperature of the body and its movementsand deliver required information to the caretaker. The methods included in the solution are:

- Planning the project and gathering requirements: Determine the objectives, target market, and project scope(as seen in Figure 3.1). Specify the exact features and functionalities of the Smart IOT Based Paralysed Patient Monitoring System, define exact performance and usability requirements, Establish a budget and schedule for the undertaking.

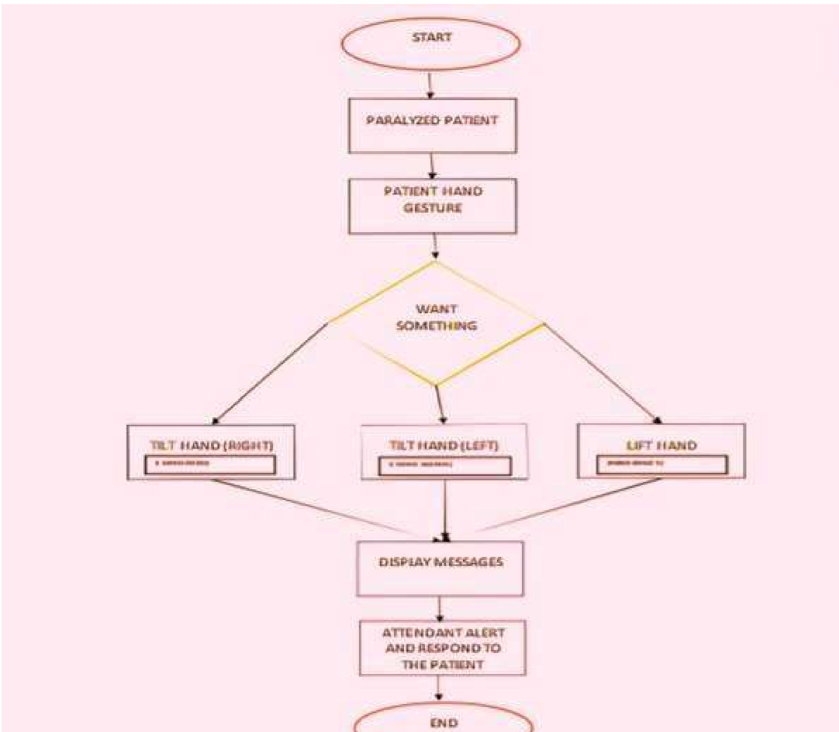

Figure 3.1: Block Diagram of Paralysis Monitoring System

- Selection and Integration of Sensors: Select the appropriate sensors to monitor the water intake and surrounding conditions. These might include a hand gloves, temperature sensor, flex sensor, wifi module. Make sure the chosen sensors are accurate, dependable, and energy-efficient to increase the device's battery life.

- Flex Sensor: Bending or flexing can be detected by a flex sensor(as seen in Figure 3.2). It is composed of a flexible base that is normally covered in a thin layer of plastic and has conductor material bonded to it. According to the angle of curving of the sensor, the amount of resistance of the substance varies. An output signal that can be read is then produced by measuring this change in resistance. Robotics, virtual reality platforms, medical equipment, and consumer electronics are just a few of the industries that use flex sensors. They are employed in robotics to pinpoint control by detecting the motion of robotic arms or joints. To improve how users communicate with virtual environments, flex sensors can be integrated into gloves for virtual reality systems to capture hand movements. Flex sensors are used by medical equipment to track joint or muscle movement for therapeutic or diagnostic purposes. Flex sensors can also help with gesture detection and ergonomic design in consumer gadgets like gaming controllers and wearables. Flex sensors, taken as a whole, provide a flexible and dependable way to identify twisting or flexibility in a variety of applications, enhancing efficiency and control in a range of technical systems.

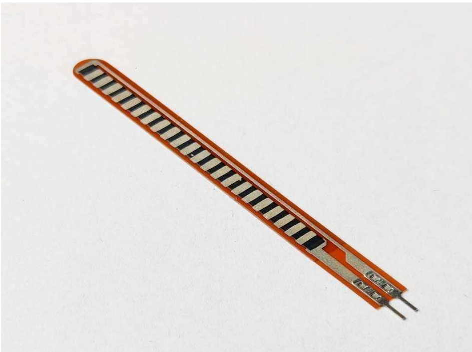

Figure 3.2: Flex Sensor

- Temperature Sensor: A popular and adaptable digital temperature sensor, the DS18B20 is utilized for temperature measurement in a variety of applications. It is perfect for both professional and hobbyist projects because of its primary advantages, which include wide temperature range, high precision, and simple interface. Its functionality is founded on the one-wire protocol developed by Dallas Semiconductor and the fundamentals of digital communication. It is possible to connect many DS18B20 sensors(as seen in Figure 3.3) to a particular microcontroller pin since each sensor has a distinguishing 64-bit serial code that is recorded in its ROM. Temperature measurements with a resolution of as little as 0.0625°C are obtained from the sensor by converting temperature to a 12-bit signal that is digital. The DS18B20 is used by sending a command to initiate temperature conversion from a microcontroller or other suitable device, and then reading the digital temperature information from the sensor via the one-wire communications protocol. In order to communicate, the master device—typically a microcontroller—sends precise command to a sensor and receives data in return. All things considered, the DS18B20's precision, ease of use, and digital output make it a simple yet reliable temperature sensing option for a variety of applications, from industrial automation to environmental monitoring.

Figure 3.3: Temperature Sensor

- ESP8266: The ESP8266 is a well-liked and extensively utilized microcontroller and Wi-Fi module in the embedded systems and IoT fields. The ESP8266 a small and affordable device for connecting gadgets to local networks or the internet because it combines a microcontroller with integrated Wi-Fi capabilities. The low cost of ESP8266 modules has led to their extensive use in Internet of Things applications and projects. Because the ESP8266 platform is open-source, developers can access and alter its firmware and software, giving it a great deal of customization capability. It has numerous GPIO ports that can be used for both digital and analog input and output in a variety of applications. Since it can communicate via Wi-Fi, the ESP8266 is perfect for Internet of Things applications where devices must exchange data via a wireless network. Power-saving features are provided by the ESP8266, which can be crucial for Internet of Things devices that run on batteries. Many Internet of Things applications, including smart devices, home automation, sensor networks, and remote monitoring, frequently use the ESP8266(as seen in Figure 3.4). By offering a reasonably priced and easily accessible platform for connecting devices to the internet, it has significantly contributed to the growth of the Internet of Things. Because of their capabilities, affordability, and versatility, ESP8266 modules are essential in the Internet of Things world. These modules from Espressif Systems offer a simple and easy way to upgrade a variety of electronic devices with Wi-Fi connectivity. The ESP8266 is a great option for Internet of Things applications because of its small form factor, low power consumption, and compatibility with many different programming languages. With the help of these modules, developers can link devices to the internet and facilitate data sharing and communication. Their extensive documentation, strong community support, and ecosystem of development tools that simplify the IoT development process are additional factors contributing to their popularity. All things considered, ESP8266 modules are a fundamental component that facilitates IoT development and aids in the general uptake of connected technologies.

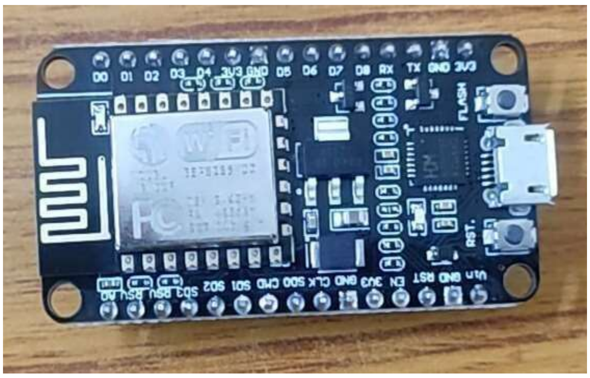

Figure 3.4: ESP8266 Sensor

- Connection Methods: Select the connection type that will be utilized to transfer information to a mobile application. Wi-Fi and BLE technology are common choices. For connectivity, the module for Wi-Fi has been used in this project. In this project, the Wi-Fi module for the ESP8266 was used. Construct the hardware components needed to establish a dependable connection with the program.

- Designing and prototyping hardware: Construct the Smart IOT Based Paralysed Patient Monitoring System external form while considering components like hardware material, form, and integration. Create a functioning prototype to confirm the integration of sensors and connectivity components.

- Create an application: The smartphone app should be user-friendly and compatible with Android. Integrate the application(as seen in Figure 3.5) with the temperature and flex sensors and networking features to receive and interpret real-time data. Provide tools for tracking past data, creating customized recommendations. Reminders that adapt should be put into place based on patient vital sign. While safely storing user data, provide privacy controls and options for data sharing.

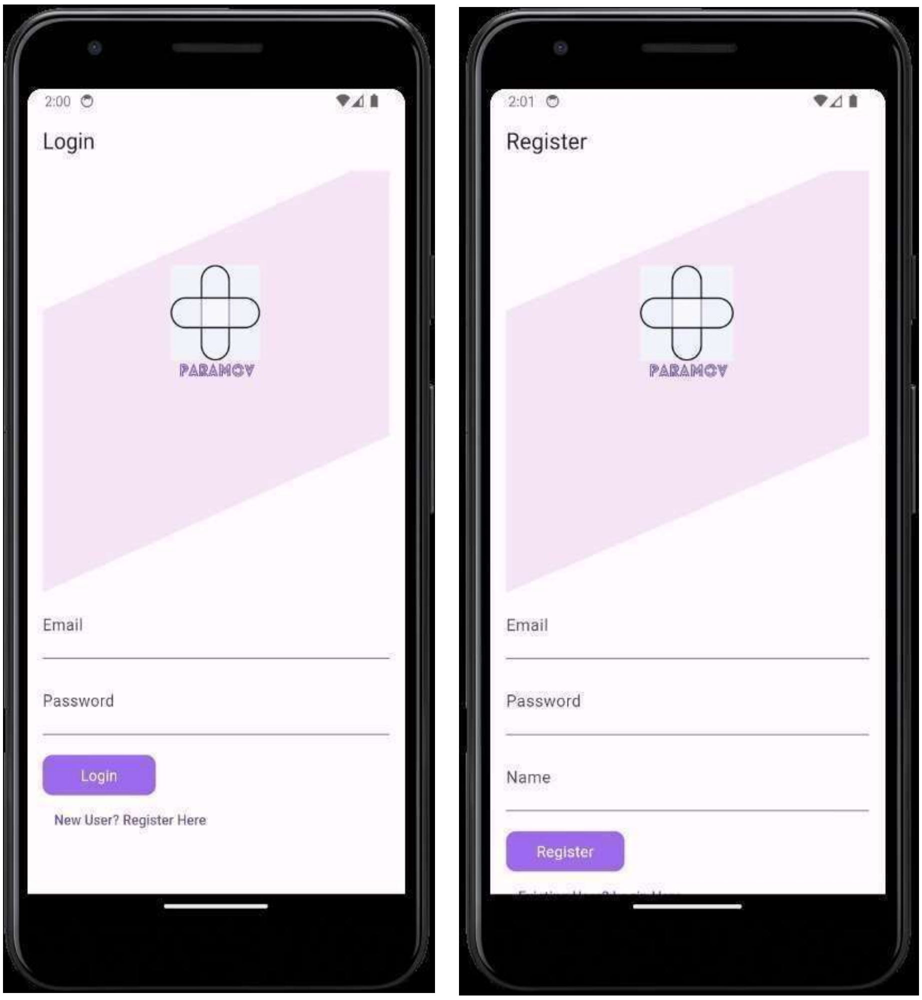

Figure 3.5: Mobile Application of Paralysis Monitoring System

3.2. Modules

Multiple modules make up an Internet of Things (IoT)-based paralysed patient monitoring system that is intended to provide thorough and instantaneous monitoring of the patient's condition. The sensing module, which has a number of sensors like movement and temperature sensors, is one important module. These sensors gather information about the patient's activity and vital signs continuously. The communication module is an additional critical module that enables the transfer of sensor data to a cloud-based platform or a central processing unit. For smooth data transfer, this module makes use of wireless communication protocols like Wi-Fi. The essential component of the monitoring system is the cloud-based platform or central processing unit. In real time, it takes information collected by sensors, processes it, and examines the patient's health status. In order to immediately notify caretakers or medical experts of emergencies or anomalous patterns, algorithms for machine learning can be used. In addition, the user interface module gives caregivers and medical experts access to a graphical interface via which they may set up alarm levels, check real-time monitoring modifications, and access patient data. A feedback module might also be included of the system, which would enable caregivers to interact with the patient via the internet and issue instructions. To sum up, the Internet of Things (IoT)-based paralysed patient monitoring system(as seen in figure 3.6) combines sensors, interactions, analyzing, interface for users, and feedback modules to guarantee ongoing and effective patient health status monitoring, allowing for prompt intervention and better care.

Figure 3.6: Architecture Diagram of Paralysis Monitoring System

The innovation proposed in this article deals with three stages of development:

- Input Stage: The input stage involves the connection of temperature sensor(DS18B20), flex sensor and WiFi modules. This innovation uses the ESP8266 module because of its low cost and low power consumption, which is crucial if the device runs on batteries. A single microcontroller pin can have numerous sensors linked to it thanks to the 1-Wire protocol for communication used by the digital temperature sensor DS18B20. It has a distinct 64-bitserial codes that makes each sensor distinct. The sensor has a large temperature range of operation and provides excellent accuracy (±0.5°C) throughout. It is well-liked for many temperature sensing applications because to its tiny size, inexpensive cost, and straightforward interface. The DS18B20 temperature sensor is appropriate for usage in difficult situations because it is physically contained in a tiny, waterproof container. Three pins make up this connector: DQ (data), GND (ground), and VDD (power supply). The sensor and DQ pin are connected via the DQ pin. With a timing-based protocol, the sensor and microcontroller exchange information about temperature. The microcontroller sends a command to start the conversation, and the sensor respond with temperature information. In order to initiate temperature conversion, the microcontroller instructs the sensor during operation. After that, the DS18B20 takes a temperature reading and records the data in the internal registers. The temperature data is then sent serially across the 1-Wire bus to the microcontroller, which can request it from the sensor. The microcontroller can precisely ascertain the temperature detected by the DS18B20 sensor by interpreting this data. All things considered, the DS18B20 sensor provides a dependable and user-friendly option for temperature detection in a range of applications.

- Processing Stage: Following data collection, the data must be transmitted to the cloud (using the WiFi-Module ESP8266) via the MQTT protocol utilizing publish and subscribe messaging transport. The Internet of Things (IoT) is best suited for MQTT, a simple, efficient messaging protocol designed for networks with erratic performance, high latency, or low bandwidth. Through a central broker, clients or devices communicate with one another via the publish-subscribe architecture of MQTT. Publishers target particular topics with messages, and subscribers receive messages from subjects that pique their interest. For Internet of Things (IoT) devices, MQTT guarantees effective data exchange, low overhead, and low power consumption. Because it provides Quality of Service levels for message delivery, it can be used for both asynchronous and synchronous communication, which makes it a popular option for Internet of Things and remote monitoring applications. Whenever there is a discernible change in the bottle's water level, the cloud receives the real-time data that was obtained from the sensor. The Firebase is used as it will also help in the authentication process when it needs to save the credentials of the individual in the cloud.

- Output Stage: An application on a smartphone connects the user and the smart IoT-based paralyzed patient monitoring system. In order for user information collection to be collected in the firebase, they must first register and log in. Following information collection, the program will display specifics. Firebase is used by an application for firebase authentication, and it stores all of the user data. By obtaining real-time data, the app tracks their health monitoring.

Here is a small and prescribed detail of the project:

Problem Statement**:**

Due to their restricted movement and ability to communicate, individuals who are paralyzed have difficulty receiving continuous monitoring and must rely on time-consuming, sporadic evaluations. In order to provide proactive monitoring, current systems need real-time insights and IoT integration. A solution that provides early identification of patient demands and requirements, prompt action, and remote access to information about patients is required by caregivers. To bridge these gaps, Paramov is desperately needed. Paramov uses wearable sensors and Internet of Things (IoT) technology to facilitate remote monitoring, data processing, and alert generating in order to give real-time monitoring. Paramov aims to enhance patient outcomes and lessen caregiver stress by fusing contemporary technology with patient care.

Solution:

The solution is Paramov, a Smart Internet of Things-based Paralysis Patient Monitoring System (SIPMS) that mixes wearable sensors, wireless communication, and data analytics. Paralyzed patients can give SIPMS real-time physiological data, including their movement patterns and health indicators. Through IoT connectivity, this data is transmitted to a monitoring unit for analysis. The system instantly tells caregivers or medical specialists when it detects irregularities that indicate demands or requirements. Remote monitoring and access to historical data are made possible via user-friendly interfaces, like those found in smartphone apps. SIPMS offers proactive, continuous monitoring that raises the level of living and improves patient care while reducing the likelihood of issues.

Benefits:

Numerous advantages exist for raising the standard of care and boosting patients' general wellbeing when using an intelligent Internet of things-based paralyzed patient monitoring system. First of all, it makes it possible to continuously monitor vital signs and gives healthcare professionals access to real-time data for prompt response in the event of any irregularities. By ensuring early identification of health conditions, this continuous monitoring lowers the likelihood of complications and readmissions to the hospital. Second, IoT-enabled system sensors can monitor patient motions and activity levels, enabling individualized rehabilitation plans catered to specific requirements. This encourages movement and active engagement in therapy sessions, both of which are critical for avoiding muscle atrophy and preserving joint flexibility. Furthermore, caregivers may keep an eye on their patients' health from a distance thanks to remote monitoring capabilities, which give patients and their families peace of mind and let medical staff use their resources more effectively. All things considered, an intelligent Internet of Things (IoT) paralysed patient monitoring system boosts patient safety, enhances treatment results, and encourages liberty and a good quality of life.

Challenges:

There were numerous challenges which has been faced during the project. It included managing power consumption so that the sensors and other devices consume less power to extend the battery life, there were certain challenges while integrating the hardware components to the cloud using the code as we needed to ensure that the real-time data reaches the cloud. While creating the application we have kept in mind that user experience has to be hassle-free so that they can easily trace their consumption level. The challenges are:

- Range: The typical data transmission distances for IoT devices connected to a network can be used to characterize the network: PAN is short-range, meaning that distances are measured in meters. An example of this would be a wearable fitness tracker that uses Bluetooth Low Energy (BLE) to connect with a mobile app. LAN stands for local area network, and it can cover short- to medium-range networks over hundreds of meters. Examples of these networks include home automation systems and sensors installed inside manufacturing lines that connect wirelessly to a gateway device located inside the same structure. MAN stands for long-range (city-wide), where distances up to a few kilometers are measured. An example of this would be a mesh network topology connected smart parking sensors placed throughout the city. Long-range wireless networks (WAN) are those where the distances are measured in kilometers. An example of this would be the agricultural sensors placed throughout a sizable farm or ranch to keep an eye on the environmental conditions and microclimates throughout the property. The data from the IoT devices should be retrieved by the network and sent to the correct location. It is necessary to choose a network protocol that fits the range. For WAN applications that need to function over a few kilometers' distance, for instance, BLE should not be used. If it's difficult to send data over the necessary range, think about edge computing. Edge computing analyses data directly from the devices, as opposed to using information coming from a data center that is miles away or another location.

- Bandwidth: The capability or speed of data transfer of a networked communication channel is referred to as bandwidth, which is a basic concept in network technology. Calculated over bit per sec, it indicates the maximum amount of data that can be transferred over a network in a specific amount of time. One of the most important metrics for assessing the effectiveness and speed of data transfer within a network is bandwidth. The following are important concepts regarding bandwidth in the field of network technology to grasp: Bandwidth and speed of data transmission are frequently interchangeable terms. More data transmission capacity is indicated by a higher bandwidth, which enables quicker communication between connected devices. The speed at which data can be transmitted or downloaded depends on the quality of the network connection. Bits per second is the common unit of measurement for bandwidth. Kbps is examples of common units. One network connection with 100 Mbps of bandwidth, for instance, can send 100 million bits of information in a second. Bandwidth is frequently divided into both upstream and downstream components, or components. The rate of data transfer from the device used by the user to the network is referred to as downstream bandwidth, and the velocity of data transfer from the networks to the user's device is referred to as upstream bandwidth. The majority of residential internet services have asymmetrical network connections, where the downstream bandwidth is typically greater than the upstream bandwidth. The efficiency of networked services and applications is directly impacted by the available bandwidth. Faster downloads, more fluid streaming of videos, and quicker online activity response times are all made possible by increased bandwidth. On the other hand, a restricted bandwidth can cause slower data transfer rates, which can cause delays and worse performance. A number of factors can influence how much actual bandwidth users experience. The effective bandwidth can be affected by a number of factors, including network congestion, the quantity of individuals using the same network, the caliber of the infrastructure, and the effectiveness of networking hardware. Furthermore, the physical medium—copper cables, fiber optics, wireless channels, etc.—that is utilized for network connections affects the amount of bandwidth that is available. Network latency is the term used to describe the lag or delay in data transmission between two points, whereas capacity measures the capacity of information transfer. For real-time applications wherein timely data delivery is critical, like online gaming or video calling, low latency is essential. Network latency is influenced by a number of other factors, including processing delays and routing efficiency, so high bandwidth by itself does not ensure low latency. The amount of bandwidth needed can change depending on the kinds of applications and users utilizing a network. The ability of a network to support more users or more data demands is referred to as scalability. In order to satisfy the bandwidth demands of growing user bases or changing technological requirements, networks must be built to scale efficiently. The downstream as well as upstream speeds are equal in symmetrical bandwidth. This is typical of business-focused services where customers require steady download and upload speeds. On the other hand, a lot of home internet providers provide asymmetrical bandwidth, meaning that the upload speed is slower than the download speed.

- Power Supply: A power supply is an essential part of electronic devices and systems because it provides the electricity required for them to function. It is in charge of transforming electrical current from a source—like an outlet or battery—into a format that can be used to power electronic components and circuits. Power supplies are essential to the dependable and effective operation of electronic systems. They are available in a variety of forms, each intended for a particular use. In the world of electronics, power supplies areessential because they provide the electrical energy required for a variety of devices to function. Selecting the appropriate solution for a given application and guaranteeing dependable and effective performance in a variety of electronic systems requires an understanding of the various types, features, and factors related to power supplies.

- Connectivity: IoT devices don't always have constant connectivity. Certain devices are made to connect on a regular basis. On the other hand, devices may occasionally stop working because of connectivity problems on an unstable network. There are situations when quality of service problems, like handling interference or channel contention on a shared spectrum wireless network, arise. In the event that uninterrupted service is deemed essential for IoT landscape design, designs ought to account for sporadic connectivity and explore all feasible avenues to guarantee it.

- Compatibility: It is possible for devices to communicate with one another, with systems, with equipment, and with technology. Interoperability can present difficulties because there are so many distinct devices connected to the Internet of Things. Traditionally, maintaining interoperability on the Internet has involved adopting standard protocols. Industry members agree on standards, which steer clear of numerous alternative designs and directions. Issues with incompatibility, and consequently interoperability, can be avoided with appropriate standards and participants who accept them. Standardization procedures for the Internet of Things occasionally find it difficult to keep up with advancements and modifications. Written and published in accordance with future iterations of standards that are still pending.

- Security: A crucial component of information technology, network security aims to protect data's availability, integrity, and confidentiality while it travels across computer networks. Network security is becoming more and more crucial as technology develops and more areas of our lives are linked by networks. The following are important concepts in network security to grasp: Verifying a user's, device's, or system's identity when they try to connect to a network is called authentication. Robust authentication techniques, like multi-factor authentication, biometrics, and username/password combinations, aid in preventing unwanted access. Authorization establishes a user's or device's level of network access after they have been authenticated. In order to guarantee that users have access to only the resources and information required for their tasks, permissions are granted based on roles. The process of transforming information into an encrypted form that is only readable by those with permission is known as encryption. This guarantees that even in the unlikely event that unauthorized parties obtain the data, they will be unable to decipher or utilize it without the necessary decryption key. A trusted network within a company and an untrusted external network, like the internet, are separated by firewalls. They guard against cyber threats and stop unauthorized access by keeping an eye on and managing network traffic that comes and goes in accordance with pre-established security rules. IDPSs keep an eye on system and/or network activity to spot malicious activity or policy infractions. They offer an extra line of protection against online attacks since they can quickly identify and address security issues. Users can access a secure network remotely by using virtual private networks (VPNs) to create safe, secure connections over the internet. This is essential for protecting the privacy of information sent over open networks, particularly when workers must connect from outside of the company to corporate networks. To keep a network environment safe, complete security measures and processes must be established. These documents provide policies and procedures for safeguarding information, incident response, access controls, user behaviour, and other security-related topics. Frequent monitoring and security audits assist in locating weak points, odd behaviour, and possible security breaches. Organizations can quickly identify and address security incidents by keeping a close eye on network activity and logs. Keeping network devices, operating systems, and software updated with the newest security patches is essential for fixing vulnerabilities that are known to exist. Maintaining patch levels regularly reduces the possibility of malicious actors exploiting vulnerabilities. Securing individual devices, or "endpoints," that are connected to a network is known as endpoint security. To make sure that endpoints follow security policies, this includes device management, antivirus software, and EDR (endpoint detection and response) solutions. An essential part of network security is educating users about dangers and best practices. Security incidents are frequently caused by human error, and a knowledgeable user base greatly enhances the overall resilience of the network. Even with precautions taken, security incidents can still happen. Organizations can minimize the impact and speed up recovery from security breaches through the implementation of an established incident response plan. Organizations must adhere to specific to an industry regulations and standards in order to guarantee that the security requirements are met. Guidelines for protecting sensitive data are provided by regulatory structures, such as PCI DSS for debit and credit card data and HIPAA for healthcare information. The safety of networks is a constant challenge in today's connected world that calls for a multifaceted strategy. To build a strong defence against the constantly changing landscape of cyber threats, it is imperative to integrate innovations in technology, policies, and user education. It is crucial to maintain a proactive and adaptable network security strategy because cyberattacks are becoming more and more sophisticated.

3.3. Advantages over traditional methods

Delivery of healthcare and patient care can be revolutionized by using a smart IoT-based paralysed patient monitoring system, which offers various advantages over existing techniques. IoT device and technology integration improves patient outcomes by streamlining data gathering and analysis in real time and enhancing the monitoring process. Here are a few main benefits:

- Real-time Monitoring: IoT-based solutions make it possible to continuously monitor paralyzed patients in real time, giving medical professionals immediate access to patient data and vital signs. Conventional techniques sometimes depend on sporadic manual inspections, which might cause delays in identifying important alterations in patient circumstances. Healthcare providers may remotely monitor patients with portable sensors as well smart medical equipment, and they can get instant warnings in the event of an emergency or other irregularity.

- Remote Accessibility: Healthcare providers can use linked devices like PCs, tablets, and smartphones to retrieve patient data at any time and from any location thanks to Internet of Things technologies. This remote accessibility improves patient monitoring effectiveness by allowing medical staff to supervise several patients at once and act quickly when needed. Traditional approaches, on the other hand, could necessitate the physical presence of medical personnel at the site of the patient during assessing, which can be problematic, particularly in big healthcare facilities as well as in emergencies.

- Continuous Data Collection: Continuous data gathering is made possible by Internet of Things (IoT)-based paralyzed patient monitoring devices, which record a wide range of patient metrics, including body temperature, heart rate, blood pressure, and oxygen saturation. When contrasted with irregular manual measures made using conventional techniques, this continuous data gathering offers a more comprehensive and precise assessment of the patient's medical condition. Healthcare professionals can also find patterns, anticipate prospective problems, and create individualized treatment regimens for each patient by analyzing trends in longitudinal data.

- Data Analytics and Insights: Health care practitioners may instantly evaluate vast amounts of patient data by using IoT platforms with sophisticated analytics features, which let them derive useful insights and useful information. In order to help healthcare practitioners make educated judgments and take preventive measures to avert unfavourable outcomes, machine learning algorithms are able to recognize anomalies, patterns, and patterns of patient data. With the help of these analytics-driven insights, medical professionals may provide paralyzed patients with more individualized and efficient care, improving patient satisfaction and treatment results.

- Patient Empowerment and Engagement: Patients can actively engage in their own care management as well as rehabilitation process with the help of Internet of Things-based paralyzed patient monitoring devices. With the convenience of their homes, patients can track their progress, keep an eye on their vital signs, and follow their recommended treatment plans thanks to wearable sensors and smartphone health apps.

- Cost-efficiency and Scalability: IoT-based paralysed patient monitoring systems may be more expensive to establish initially than more conventional approaches, but may provide advantages in terms of scalability and long-term cost reductions. IoT technology enable healthcare businesses streamline operations, enhance efficiency, and lower total healthcare costs by lowering demand for frequently visits to the hospital, minimizing the risk of problems, and optimizing resource use. Furthermore, IoT solutions ensure adaptability and versatility in dynamic healthcare contexts by being easily scaled to meet changing healthcare needs and expanding patient populations.

To sum up, IoT-based paralysed patient monitoring systems, with their instantaneous tracking,remote portability, continuous data gathering, enhanced information analysis, patient empowerment, and cost-efficiency, offer a number of advantages over conventional technologies. Healthcare practitioners may improve clinical results, give better care, and change how paralyzed patients receive treatment by utilizing the potential of IoT technologies.

3.4. Applications

A ground-breaking development in healthcare technology is the use of intelligent Internet of Things-based paralyzed patient monitoring systems, which serve a particular target market with particular requirements and difficulties. With the help of this cutting-edge technology, paralyzed patients' medical problems can be monitored and managed in real-time by combining Internet of Things, or IoT, technology, sensors, and data analytics. A comprehensive understanding of the health landscape, the incidence of paralysis, the difficulties faced by caregivers, and the potential advantages of IoT solutions are necessary to comprehend the intended audience for such applications. Paramov have a range of applications in various sectors:

- Hospitals and Rehabilitation Centres: Smart IoT-based paralysed patient monitoring systems must be implemented in hospitals and rehabilitation centres. These clinics assist people recovering from a range of illnesses that cause paralysis, including as spinal cord injuries, strokes, and other maladies. By utilizing these devices, medical practitioners can keep an eye on their patients' vitals, body movements, and activities from a distance. Real-time data collecting makes it possible to respond quickly to emergencies, guaranteeing that patients receive medical attention quickly and improving patient outcomes in the process. These monitoring devices track important health data continually, providing a complete approach to patient care. Through the analysis of this data, medical professionals are able to make well-informed decisions about medical treatment and rehabilitation techniques by gaining insightful knowledge about patients' progress over time. Monitoring systems, for example, can identify variations in a patient's vital signs as well as movement patterns, which may point to problems or improvements in their health. Furthermore, because these systems are remote, healthcare personnel are not burdened as much because they can monitor patients around-the-clock without having to physically be present. This improves productivity and facilitates more effective use of available resources, guaranteeing that patients receive care and attention when they need it most. Moreover, these monitoring systems offer insights that go beyond providing immediate patient care. They support investigations meant to advance knowledge about paralysis and its treatment.

- Patients Demographics: People who are paralyzed due to injuries to the spinal cord, neurological conditions or severe illnesses are the primary target population for intelligent Internet of Things-based paralysed patient monitoring systems. Whatever the cause, paralysis affects a wide range of people in all age groups, with each person coping with a different level of disability and limitations on their ability to function. Continuous monitoring becomes imperative for these patients in order to prevent complications that are intrinsic to their disease from occurring. Pressure ulcers, UTIs, respiratory issues, and muscle atrophy are just a few of the numerous dangers that call for prompt monitoring and action. Patients who are immobilized frequently worry about pressure ulcers, which are caused by prolonged pressure on particular body parts and, if neglected, can result in serious infections. Similar to this, bladder failure makes urinary tract infections a serious concern that requires close observation and prompt treatment to avoid systemic consequences. Chronic respiratory problems, which are frequently made worse by limited movement, require ongoing attention to prevent pneumonia or respiratory distress. Furthermore, the effects of extended immobility, such as muscular atrophy, highlight the necessity of customized exercise programs and physical treatment in order to maintain muscle mass and avert functional loss. Essentially, the wide range of issues linked to paralysis highlights how vital it is to have ongoing monitoring enabled by intelligent IoT-based solutions in order to protect the health and well-being of those who are affected.
- Caregivers: When IoT-based monitoring solutions for patients with paralysis are implemented, caregivers feel a great sense of relaxation. These technologies, which provide real-time notifications on the patient's state, lessen the constant need for constant care. Caregivers can minimize stress and anxiety related to the unpredictable nature of taking care of a paralyzed person by swiftly addressing any emergent problems thanks to this instant notification feature. In addition to offering comfort, the system's sophisticated data analytics features provide a more profound comprehension of the patient's healthcare patterns. Caregivers can make well-informed decisions and create individualized care plans by examining large data sets to obtain insightful knowledge. By adjusting their approach to the distinctive needs and tastes of the patient, caregivers can maximize the quality of care they provide. Caregivers may proactively address possible health issues, avoid complications, and guarantee the patient's long-term well-being when they have instant access to relevant information. To put it simply, IoT-based monitoring solutions relieve the strain of constant observation while also equipping caregivers with the skills and resources they need to provide patients who are paralyzed with the best care possible.
- Healthcare Professionals: Internet of Things-based monitoring devices designed for patients with paralysis mark a major breakthrough in patient care for medical professionals such as doctors, nurses, and therapists. Healthcare professionals may oversee patients' conditions and modify treatment programs as needed thanks to these devices, which allow for distant monitoring of patients' progress. Healthcare providers can better manage patient care and resource allocation by leveraging remote access to real-time data. This guarantees that patients receive individualized treatment and prompt interventions. Distinguishing hurdles to access and providing patients with uninterrupted support, the system's smooth communication functionalities enable online consultations and remote therapeutic sessions. Additionally, these devices improve the precision of medical assessments by providing constant observation of indicators of health and rehabilitative activities.

- Insurance Providers: The deployment of Internet of Things-based patient monitoring systems for individuals with paralysis has the potential to yield substantial benefits for insurance companies. Insurance firms can use the abundance of data generated by these systems to more precisely evaluate risk profiles. Insurance companies can better customize insurance plans to match the needs of specific individuals by gaining insights into policyholders' health and activities through the analysis of real-time health data. Additionally, insurers can promote preventive management of healthcare among policyholders by offering incentives for adoption of these technologies. Continuous monitoring can lessen the chance of expensive hospital stays and medical procedures for paraplegic persons by identifying and preventing possible health issues. Insurance companies can reduce the risks associated with chronic illnesses like pressure ulcers and bacterial infections of the urinary tract, which are prevalent in people who are paralyzed, by encouraging preventative care and early intervention. By preventing the need for costly treatments or emergency interventions, this preventive strategy not only improves policyholder health outcomes but also lowers healthcare costs for insurance companies. Furthermore, insurers can improve the accessibility of care for patients who are paralyzed, especially those who live in remote or underserved locations, by making investments in technologies that provide monitoring remotely and telehealth services.
- Medical Device Manufacturers: The field of paralysed patient monitoring is being advanced by medical equipment manufacturers, especially those who are concentrating on Internet of Things technology. These firms collaborate with healthcare stakeholders to develop novel solutions that address the specific issues faced by patients who are paralyzed. They achieve this by integrating innovative technology for sensors, data analytics, and networking. Manufacturers can gather information regarding indicators of health, patterns of movement, and environmental variables in real-time by utilizing developments in technology for sensors, including wearable technology and smart sensors. With their non-intrusive and comfortable design, these sensors allow for continuous monitoring without interfering with the patient's everyday activities. Furthermore, manufacturers can process enormous volumes of data and derive valuable insights into patient behaviors and health status through the use of advanced data analytics algorithms. Moreover, connectivity features make it possible for healthcare practitioners and monitoring devices to communicate seamlessly, enabling remote monitoring and prompt intervention when needed. Medical device makers improve usability as well as integration across different healthcare settings by establishing interconnecting platforms and standards, which guarantee interoperability with current healthcare systems. Additionally, in order to develop intuitive as well as user-friendly interfaces for patients as well as healthcare professionals, manufacturers give priority to user-centered design principles. Manufacturers make sure their goods satisfy the unique requirements and preferences of patients and caregivers who are paralyzed by asking for input from end users at every stage of the development process. In the end, medical device makers enable healthcare professionals to provide proactive and individualized treatment while empowering patients who are paralyzed to enjoy more independent lives through their creative solutions. These businesses contribute significantly to improving the standard of life and health result for this susceptible group by attending to the changing demands of paralyzed people.

CHAPTER 4

RESULTS

The deployment of an intelligent Internet of Things (IoT) paralysed patient monitoring equipment has demonstrated noteworthy progress in the healthcare sector by providing improved assistance and care for paralyzed patients. This cutting-edge technology combines a variety of sensors, actuators, and Internet of Things (IoT) gadgets to continually monitor paralyzed patients' vital signs, gait patterns, and surroundings to provide prompt intervention and individualized treatment. Healthcare practitioners can remotely track patients' health state in real-time, enabling preventative intervention and lowering the risk of consequences, through seamless integration of data acquired from wearable technology, the surrounding sensors, and medical equipment.

Enhancing the safety of patients and their quality of life is one of this intelligent IoT-based monitoring system's most noteworthy results. Healthcare professionals can identify any variations from standard values and take early action to prevent adverse occurrences by continuously tracking vital signs, such as blood pressure, heart rate, and oxygen saturation levels. Caregivers may also monitor patient movement patterns thanks to the system's connection with motion sensors, which helps avoid issues brought on by immobility. Additionally, the technology can further improve patients' general well-being by creating a safe and comfortable atmosphere for them by evaluating environmental data, such as humidity levels and room temperature.

Moreover, better resource management and healthcare delivery have resulted from the deployment of this monitoring system. Healthcare professionals can more efficiently manage their workflows and resource allocation by remotely tracking patients' health state and only acting when necessary. In addition to increasing healthcare delivery efficiency, this lessens the strain on medical institutions, enabling them to handle a higher patient volume and offer more effective treatment to those in need. Furthermore optimizing the care process and enhancing adherence of patients to treatment programs, the system can use IoT technology to automate mundane chores like reminders for medication and appointment scheduling.

Offering telemedicine and remote patient monitoring services is another noteworthy benefit of the intelligent Internet of Things-based paralyzed patient monitoring system. Patients may get prompt medical advice and action without having to visit the hospital frequently by sending accurate medical data to healthcare experts via secure internet platforms. Along with lowering healthcare costs and raising patient happiness, this also makes care more accessible to patients who reside in rural or underserved areas. Furthermore, with the use of remote diagnostics and virtual consultations, the system gives patients the ability to actively manage their care and take charge of their health.

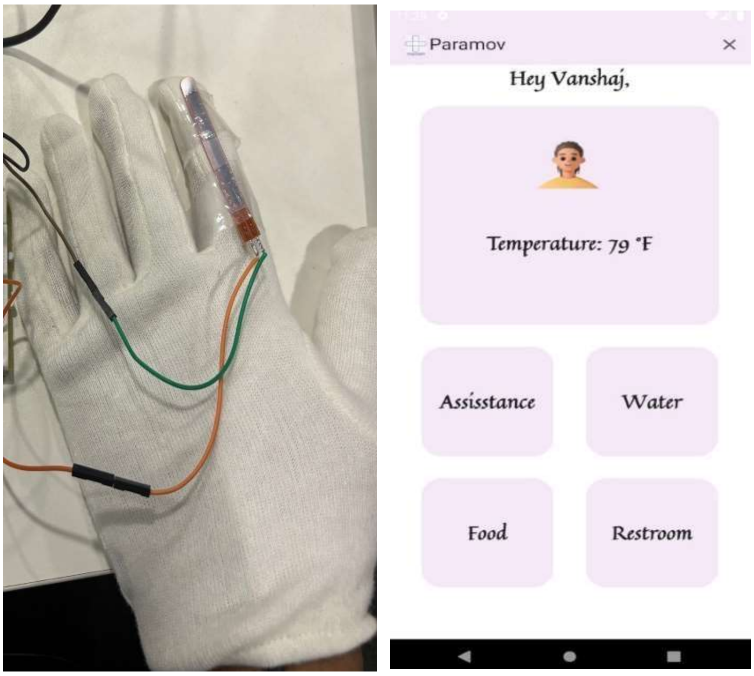

Figure 4.1: Application without Notification

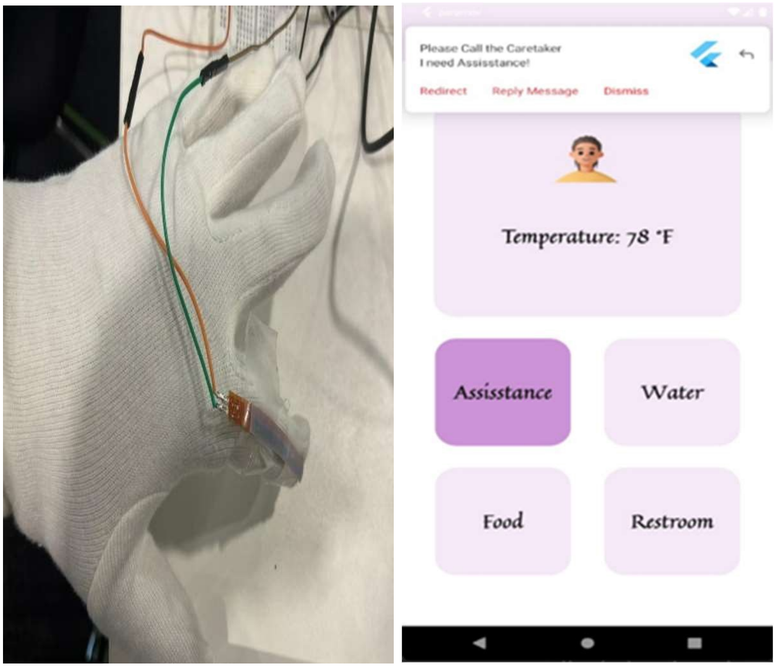

Figure 4.2: Application with Notification for Assistance

Two different instances of hand motion are shown in the above illustration. In the first motion, the alert is not displayed (as seen in Figure 4.1) when the hand is static, and in the second image, we can observe that the alert for assistance (as seen in Figure 4.2) appears when the paralyzed patient's fingers move. As of right now, we've maintained the alert that will appear if the hand's fingers move at an angle greater than 45 degrees, however it can be adjusted to suit your needs. A small amount of time is allowed to determine whether the patient truly needs help or not. As occasionally, even when a patient does not seem in require of assistance, there may be finger motions.

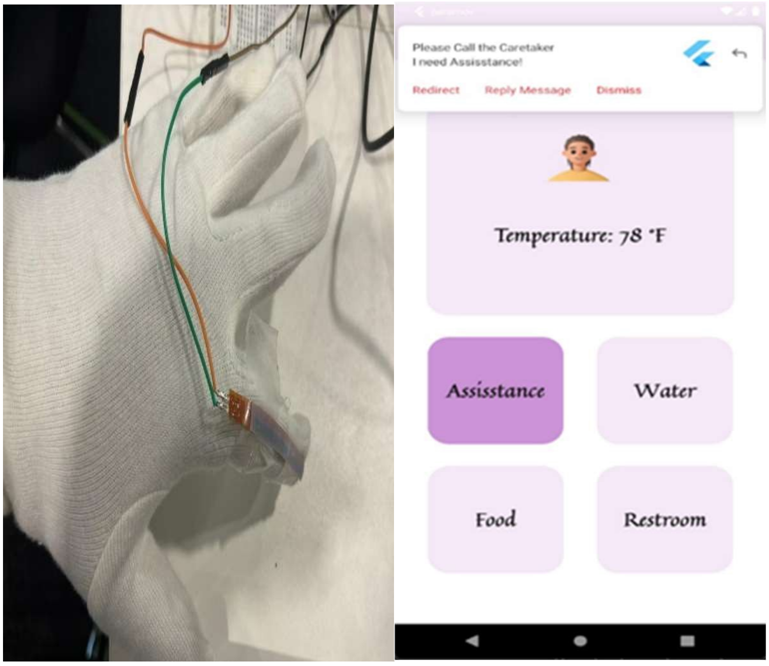

Figure 4.3: Application with Notification for Water

Two different instances of hand motion are shown in the above illustration. In the first motion, the alert is not displayed (as seen in Figure 4.1) when the hand is static, and in the second image, we can observe that the alert for water (as seen in Figure 4.3) appears when the paralyzed patient's fingers move. As of right now, we've maintained the alert that will appear if the hand's fingers move at an angle greater than 45 degrees, however it can be adjusted to suit your needs. A small amount of time is allowed to determine whether the patient truly needs help or not. As occasionally, even when a patient does not seem in require of assistance, there may be finger motions.

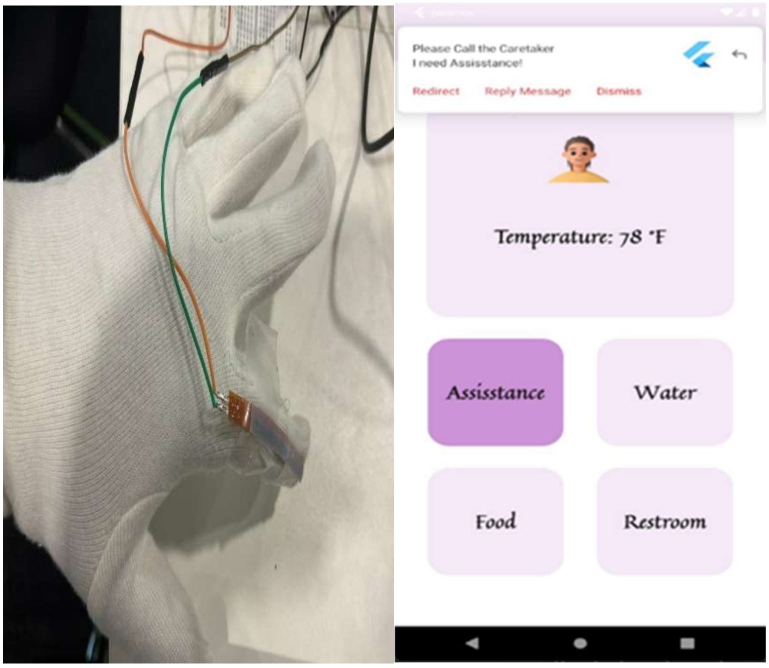

Figure 4.4: Application with Notification for Food

Two different instances of hand motion are shown in the above illustration. In the first motion, the alert is not displayed (as seen in Figure 4.1) when the hand is static, and in the second image, we can observe that the alert for food (as seen in Figure 4.4) appears when the paralyzed patient's fingers move. As of right now, we've maintained the alert that will appear if the hand's fingers move at an angle greater than 45 degrees, however it can be adjusted to suit your needs. A small amount of time is allowed to determine whether the patient truly needs help or not. As occasionally, even when a patient does not seem in require of assistance, there may be finger motions.

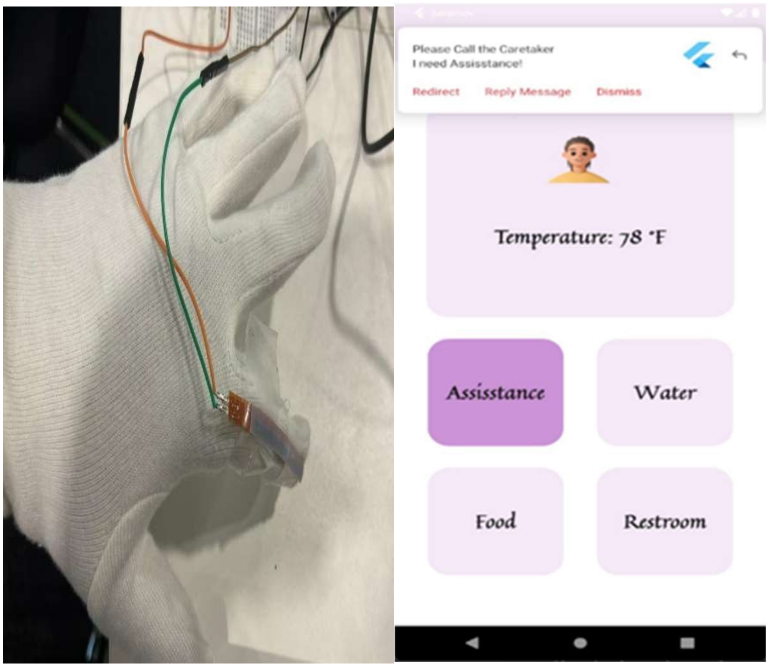

Figure 4.5: Application with Notification for Restroom

Two different instances of hand motion are shown in the above illustration. In the first motion, the alert is not displayed (as seen in Figure 4.1) when the hand is static, and in the second image, we can observe that the alert for restroom (as seen in Figure 4.5) appears when the paralyzed patient's fingers move. As of right now, we've maintained the alert that will appear if the hand's fingers move at an angle greater than 45 degrees, however it can be adjusted to suit your needs. A small amount of time is allowed to determine whether the patient truly needs help or not. As occasionally, even when a patient does not seem in require of assistance, there may be finger motions.

CHAPTER 5

CONCLUSION

The tremendous ramifications for healthcare are highlighted by the conclusion of the study on a smart IoT-based paralyzed patient monitoring system. It demonstrates how well the system monitors important health indicators and delivers timely interventions for people with paralysis. The technology shows great promise for transforming patient care delivery, even in the face of obstacles like technical problems and worries about data security. Further study should look on features like forecasting analytics and incorporating user feedback to improve the usability of the system. In summary, the research highlights the significance of sustained innovation in IoT- driven healthcare solutions to cater to the changing requirements of individuals with paralysis,enhance therapeutic results, and minimize healthcare expenditures.

The main conclusions and findings from the research project are included in the summary of findings at the end of the research or study on an IoT-based paralysis patient monitoring system. The effectiveness of the Internet of Things-based monitoring system in thoroughly tracking vital health metrics of individuals with paralysis is a major finding of these studies. Vital signs including temperature of the body and mobility are frequently included in these measures; these are important for determining the patient's general state of health and identifying any anomalies from the norm. It is likely that the study explores the particular processes that IoT devices use to reliably and continually monitor these characteristics. For example, it might investigate how body temperature fluctuations are monitored in real-time and any health problems can be detected early thanks to wearable sensors or gadgets that are incorporated into the patient's surroundings.

Likewise, motion sensors or intelligent wearables that record activity levels and movement patterns may be used by the system to track mobility parameters. The results might also emphasize how the system helps to enable proactive interventions and individualized healthcare strategies. Healthcare providers can quickly detect any alarming patterns or deviations by using IoT technology to remote and in real time monitor critical health parameters. This enables immediate interventions and modifications to a patient's treatment plan. This proactive strategy may help paralyzed patients experience better overall outcomes, lower their risk of problems, and receive higher-quality care overall.

Incorporating an intelligent Internet of Things (IoT)-based paralyzed patient monitoring systemhas numerous advantages and transformative potential for patients as well as healthcare professionals alike. One major benefit is that the system may be continuously monitored by IoT devices, which allows for early health issue diagnosis and timely response. Healthcare workers can quickly spot potential difficulties or changes from baseline medical condition by using the system's real-time surveillance of important health metrics, such as temperature of the body and mobility. By reducing the likelihood of unfavourable occurrences or consequences, this proactive strategy not only improves the safety of patients but also the overall results of healthcare.

Furthermore, the Internet of Things-based system's remote monitoring features provide patients and caregivers with unmatched ease and peace of mind. Because of their limited mobility or need for specialized care, paralyzed individuals frequently encounter obstacles when trying to obtain healthcare services. Patients no longer need to stay in the hospital for extended periods of time when they may receive continuous treatment and surveillance from the ease of their homes thanks to remote monitoring. In addition to improving patient satisfaction, this also lessens the strain on medical facilities and staff, resulting in more effective and economical care delivery. The deployment of an Internet of Things-based monitoring system also affects other aspects of the healthcare system, such as population health management and resource allocation.Healthcare practitioners can obtain significant insight into patient health patterns and trends byusing technology to expedite data collection and analysis. This enables more focused interventions and preventive actions. In the end, the use of these systems is a big step toward enhancing patient-centred care, changing the way healthcare is delivered, and raising the standard of living for those who are paralyzed.

The deployment of an intelligent IoT-based paralyzed patient monitoring system has many advantages, but it also has drawbacks. The technical difficulties associated with the devices of the Internet of Things themselves are one major problem. These devices might have problems with dependability, connectivity, or compatibility with the current infrastructure of hardware and software. Careful preparation and coordination are necessary to overcome a key obstacle that stands in the way of providing optimal connectivity and interoperability.

The requirement to address issues with data security and privacy is another crucial factor. Sensitive health data collection and transmission via IoT devices carries a risk of misuse of personal data, illegal access, and data breaches. Healthcare providers or patients themselves may be resistant to or skeptical of the implementation of a connected to the internet of monitoring system. Relying on technology to monitor patients may cause anxiety in certain healthcare providers who worry that it may erode the traditional patient-provider connection or result in information overload. Patients may also voice worries about the perceived intrusiveness of ongoing monitoring or the accuracy with which IoT devices record their health state. Furthermore, there are issues with the monitoring system's scalability and durability, especially in larger health care facilities or areas with limited resources.

Healthcare businesses must continue to invest in and be committed to providing the IoT infrastructure with necessary technical assistance, maintenance, and updates. Furthermore, addressing patient gaps in digital literacy and access to technology may provide challenges to the equitable adoption and widespread use of IoT-based monitoring technologies. It is critical to recognize and deal with the different obstacles and constraints that come with putting it into practice. Healthcare stakeholders should optimize the potential advantages of IoT technology while minimizing potential hazards and guaranteeing satisfaction and security for patients by proactively resolving technical, secrecy, and adoption-related concerns.

Overall, the summary of results emphasizes how much the Internet of Things-based monitoring system has advanced patient care for paralyzed patients. These technologies have the power to substantially enhance patient outcomes and quality of life by efficiently monitoring important health markers and facilitating prompt treatments.

CHAPTER 6

FUTURE SCOPE

The potential for intelligent IoT-based paralysis patient monitoring systems is enormous, especially when combined with hospital and medical facility data. With improved monitoring, management, and individualized treatment, these systems have the potential to completely transform the way paraplegic patients receive healthcare as technology develops. Numerous advantages result from integrating with hospital as well as medical facility data. First of all, it makes it possible for healthcare professionals to quickly and accurately make judgments by giving them instantaneously information regarding a patient's vital signs, treatment plans, and medical history. By guaranteeing prompt actions and lowering the risk of problems, this connection makes it easier for healthcare providers to communicate with the monitoring system. Furthermore, Internet of Things (IoT)-based monitoring systems have the ability to offer complete and ongoing treatment outside the hospital walls by utilizing hospital data. With remote patient monitoring, patients can continue to get excellent medical care while staying in their homes. By lowering hospital admissions and readmissions, this relieves the strain on healthcare facilities while simultaneously improving patient comfort and independence.

Forecasting and personalized medicine are further made possible by connection with hospital data, which enables the collection and analysis of enormous volumes of patient data. By analyzing data trends, predictive machine learning models can spot possible problems before they become more serious or predict health decline, allowing for proactive actions that enhancepatient outcomes. Optimizing the distribution of resources and provision of healthcare is a majorbenefit of combining hospital data with IoT-based monitoring systems. Hospitals can more effectively distribute employees, supplies, and facilities to suit patient needs by monitoring thehealth of patients and resource consumption in real-time. Data analysis findings can also guide the development of focused therapies to meet the needs of particular patient populations, healthcare policies, and resource planning. However, there are additional difficulties and factorsto take into account when combining IoT-based systems for monitoring with hospital data.

Data security, privacy, and regulatory compliance are at the top of the list. To protect patient information and guarantee compliance with healthcare legislation, strong data encrypting it controls on access, and compliance procedures must be put in place. Standard formats for data and communication protocols are also necessary since interoperability problems might occur when combining systems from various suppliers or platforms. To address these issues and create interoperable frameworks that enable smooth data integration and sharing, cooperation among medical professionals, software developers, and regulatory agencies is crucial.

To sum up, the potential for smart IoT-based paralysis patient monitoring systems that are coupled with hospital as well as medical facility data to improve patient care, enhance healthcare delivery, and spur invention in the healthcare sector is enormous. These integrated systems provide the potential to transform healthcare delivery.

REFERENCES

[1] Pandey, M., Chaudhari, K., Kumar, R., Shinde, A., Totla, D., & Mali, N. D. (2018, August). Assistance for paralyzed patient using eye motion detection. In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) (pp. 1-5). IEEE.

[2] Kumar, M. V., Pandurangan, K., & Vinu, R. (2021, November). Automated paralysis patient monitoring system. In 2021 IEEE National Biomedical Engineering Conference (NBEC) (pp. 71-76). IEEE.

[3] Fan, Xiao, Wanrong Sun, Aifeng Ren, Dou Fan, Nan Zhao, Daniyal Haider, Xiaodong Yang, and Qammer H. Abbasi. "Detection and diagnosis of paralysis agitans." IEEE Access 6 (2018): 73023- 73029.

[4] Varghese, N., Hepsiba, D., & Anand, L. V. (2022, April). Health Monitoring and Observatory System for Paralysed Patients using Blynk Application. In 2022 6th International Conference on Devices, Circuits and Systems (ICDCS) (pp. 322-326). IEEE.

[5] Priya, A. D., & Sundar, S. (2019, March). Health monitoring system using IoT. In 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN) (pp. 1-3). IEEE.

[6] Zeb, K., Mahmood, A., Uddin, W., Khan, M. A., & Kim, H. J. (2019, January). Healthcare monitoring system and transforming monitored data into real time clinical feedback based on IoT using Raspberry Pi. In 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-6). IEEE.

[7] Shaikh, S., & Chitre, V. (2017, May). Healthcare monitoring system using IoT. In 2017 International Conference on Trends in Electronics and Informatics (ICEI) (pp. 374-377). IEEE.

[8] Annepu, Visalakshi, Adnan Naji Jameel ALTamimi, H. S. S. Aljibori, Naga Raju Challa, Kalapraveen Bagadi, M. N. Mohammed, Oday I. Abdullah, M. Alfiras, and Rabab Alayham Abbas Helmi. "Implementation of an Efficient IoT Enabled Automated Paralysis Healthcare System." In 2023 IEEE 8th International Conference on Engineering Technologies and Applied Sciences (ICETAS), pp. 1-6. IEEE, 2023.

[9] Usha, S., Jeevitha, G., Logesh, M., Karthik, M., Kaviyaa, M., & Prasanth, S. S. (2023, September). IoT based Integrated Health Care Monitoring System. In 2023 4th International Conference on Smart Electronics and Communication (ICOSEC) (pp. 353-358). IEEE.

[10] Akash, M. R. R., & Shikder, K. (2020, February). IoT based real time health monitoring system. In 2020 Research, Innovation, Knowledge Management and Technology Application for Business Sustainability (INBUSH) (pp. 167-171). IEEE.

[11] Akram, P. Saleem, M. Ramesha, Sai Aamani Sindhu Valiveti, Shaik Sohail, and K. Teja Samba Siva Rao. "IoT based remote patient health monitoring system." In 2021 7th

International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1, pp. 1519-1524. IEEE, 2021.

[12] Siam, Ali I., Mohammed A. El-Affendi, Atef Abou Elazm, Ghada M. El-Banby, Nirmeen A. El-Bahnasawy, Fathi E. Abd El-Samie, and Ahmed A. Abd El-Latif. "Portable and real-time IoT-based healthcare monitoring system for daily medical applications." IEEE Transactions on Computational Social Systems (2022).

[13] Khan, T., & Chattopadhyay, M. K. (2017, August). Smart health monitoring system. In 2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC) (pp. 1-6). IEEE.

[14] Kishore, A. S., Chinni, G. R., JayaLakshmi, G., & Reddy, K. S. K. (2023, February). Smart Healthcare Monitoring System Using IoT Technology. In 2023 11th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON) (pp. 1-5). IEEE.

[15] Pirani, Z., Momin, A., Kadri, A., & Shaikh, A. (2018, June). Survey of Numerous Accessible Applications for Paralysis Patients. In 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS) (pp. 1965-1969). IEEE.

[16] Starner, T., Weaver, J., & Pentland, A. (1997, October). A wearable computer based american sign language recognizer. In Digest of Papers. First International Symposium on Wearable Computers (pp. 130-137). IEEE.

[17] Deepasri, T., Gokulpriya, M., Arunkumar, G., Mohanraj, P., & Shenbagapriya, M. (2017). Automated paralysis patient health care monitoring system. South Asian Journal of Engineering and Technology, 3(2), 85-92.

[18] Abin Kumar, A., Raj, A., Johnson, J., & Krishnan, N. R. (2023, November). Project on health care system for paralysis patient using IOT. In AIP Conference Proceedings (Vol. 2546, No. 1). AIP Publishing.

[19] Atadoga, Akoh, Toritsemogba Tosanbami Omaghomi, Oluwafunmi Adijat Elufioye, and Ifeoma Pamela. "Internet of Things (IoT) in healthcare: A systematic review of use cases and benefits." (2024).

[20] Beenish, H., Nasar, F., Sheikh, E., & Fahad, M. (2021). Design and implementation of monitoring system for paralysis patient using IoT. *KIET Journal Of Computing And Information Sciences*, *4*(2), 10-10.

[21] Haneesh, L., Ajay, S., & Varuna, W. R. (2023). Monitoring Paralysis Patients Using IoT. *Journal of Data Acquisition and Processing*, *38*(2), 2708.

[22] Kumar, S. K., Dhineshkumar, M., Rajeswari, A., Manimegalai, R., Kalaranjini, R. M., & Manikandan, G. (2021, December). Development of an IoT Based Domestic Monitoring and Alerting System for Paralysed Patients. In *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)* (pp. 586-591). IEEE.

APPENDIX – 1

- Actuators: Actuators are parts or devices that react to signals from controllers or Internet of Things systems by moving physically. This could be any kind of mechanical response, including movement and adjustment.
- IOT: Embedded with sensors and software, the Internet of Things (IoT) is a network of interconnected devices that can exchange and gather data.
- Sensor: A sensor is a device that measures and detects physical characteristics and then transmits that information to other IoT devices as signals or data.
- Gateways: IoT device communication and data exchange are facilitated by gateways, which are devices that link various networks.
- Cloud Computing: Utilizing a network of distant servers hosted on the internet for data processing, storing, and management—a technique commonly employed in Internet of Things applications—is known as cloud computing.
- Edge Computing: Improving latency in Internet of Things systems through edge computing, which processes data close to the point of generation instead of depending on a centralized cloud.
- Middleware: Applications and devices connected to the Internet of Things that serve as a bridge to transmit and receive data between one another.
- Protocols: Protocols are essential for Internet of Things interoperability because they specify how data is transferred between devices in a network.
- Security: The procedures and guidelines put in place to guard against cyberattacks, illegal access, and data breaches on IoT networks and devices.
- Interoperability: The capacity of various IoT systems and devices to interact with one another and function as a cohesive unit.
- Big Data: Complicated, sizable datasets produced by Internet of Things (IoT) devices that call for sophisticated analytics to extract value.
- Machine Learning: Algorithms used in machine learning, a branch of artificial intelligence, allow systems to learn from and improve upon data without the need for explicit programming.
- Application Programming Interface (API): A collection of guidelines that permit interactivity and communication between various software programs.
- Firmware: IoT devices' firmware is embedded software that controls the hardware at a low level.
- IPv6: A protocol designed to handle the increasing number of Internet of Things devices, offering a significantly higher number of distinct IP addresses.
- Smart Grid: An intelligent electrical distribution system known as a "smart grid" makes use of the Internet of Things to maximize energy production, distribution, and consumption.

APPENDIX – 2

Hardware Code:

```
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS D2
int flexpin1 = A0;
int flexpin2 = D1;
const int numReadings = 10; // Number of readings to average for stability
int sensorValues1[numReadings];  // Array to store sensor readings
int sensorValues2[numReadings]; // Array to store sensor readings
int readIndex = 0;

OneWire  oneWire(ONE_WIRE_BUS);

DallasTemperature sensors(&oneWire);

float Celsius = 0;
float Fahrenheit = 0;


#define FIREBASE_HOST "paramov-b4f20-default-rtdb.firebaseio.com" //Your Firebase
Project URL goes here without "http:" , "" and "/"
#define FIREBASE_AUTH "VdGxUcAqgZ30qPer5O3WkUsr0wfKIEwWEv7H2ww7"
//Your Firebase Database Secret goes here

#define WIFI_SSID "Iotvb" //WiFi SSID to which you want NodeMCU to connect
#define WIFI_PASSWORD "vanshaj123" //Password of your wifi network

// Declare the Firebase Data object in the global scope
FirebaseData firebaseData;

/* 4, Define the FirebaseAuth data for authentication data */
FirebaseAuth auth;

/* Define the FirebaseConfig data for config data */
FirebaseConfig config;


void setup() {
  sensors.begin();
  Serial.begin(9600);
  // pinMode(flexpin1, INPUT_PULLUP);
  // pinMode(flexpin2, INPUT_PULLUP);
  Serial.println("Serial communication started\n\n");
```

```
WiFi.begin(WIFI_SSID, WIFI_PASSWORD); //try to connect with wifi
  Serial.print("Connecting to ");
  Serial.print(WIFI_SSID);

  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }

Serial.println();
  Serial.print("Connected to ");
  Serial.println(WIFI_SSID);
  Serial.print("IP Address is : ");
  Serial.println(WiFi.localIP()); //print local IP address
// Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH); // connect to firebase

/* Assign the database URL and database secret(required) */
  config.database_url = FIREBASE_HOST;
  config.signer.tokens.legacy_token = FIREBASE_AUTH;
  Firebase.begin(&config, &auth);

  Firebase.reconnectWiFi(true);
  delay(1000);
}


void loop() {
  sensors.requestTemperatures();

  Celsius = sensors.getTempCByIndex(0);
  Fahrenheit = sensors.toFahrenheit(Celsius);

  Serial.print(Celsius);
  Serial.print(" C ");
  Serial.print(Fahrenheit);
  Serial.println(" F");
  int flexVal1;
  int flexVal2;
  flexVal1 = analogRead(flexpin1);
  // Serial.print("Test: ");
  // Serial.print(flexVal1);
  // Store reading in array for averaging
  sensorValues1[readIndex] = flexVal1;
  flexVal2 = analogRead(flexpin2);
  sensorValues2[readIndex] = flexVal2;
  // Increment read index and handle overflow
  readIndex = (readIndex + 1) % numReadings;
```

```
// Calculate average sensor reading
  float avgSensorValue1 = calculateAverage(sensorValues1);
  float avgSensorValue2 = calculateAverage(sensorValues2);
  // Convert sensor value to degrees (assuming linear relationship)
 float angle1 = map(avgSensorValue1, 0.0, 1023.0, 0.0, 180.0); // Adjust range based on
sensor behavior
  float angle2 = map(avgSensorValue2, 0.0, 1023.0, 0.0, 180.0); // Adjust range based on
sensor behavior
  Serial.print("Sensor1: ");
  Serial.print(angle1);
  // Serial.print(" Sensor2: ");
  // Serial.println(angle2);
  delay(1000);
// Firebase Error Handling And Writing Data At Specifed Path******************

if (Firebase.setInt(firebaseData, "/need1", int(angle1))) { // On successful Write operation,
function returns 1
Serial.println("Value Uploaded Successfully");
        delay(1000);

 }
// if (Firebase.setInt(firebaseData, "/need2", int(angle2))) { // On successful Write operation,
function returns 1
// Serial.println("Value Uploaded Successfully");
//          delay(1000);

// }
if (Firebase.setInt(firebaseData, "/temperature", int(Fahrenheit))) { // On successful Write
operation, function returns 1
Serial.println("Value Uploaded Successfully");
        delay(1000);

 }
else {
Serial.println(firebaseData.errorReason());
}
}

int calculateAverage(int values[]) {
  int sum = 0;
  for (int i = 0; i < numReadings; i++) {
    sum += values[i];
  }
  return sum / numReadings;
}

float map(float value, float low1, float high1, float low2, float high2) {
  return (value - low1) * (high2 - low2) / (high1 - low1) + low2;
}
```

Application Code:

```
import 'package:flutter/foundation.dart';
import 'package:paramov/components/notification_service.dart';
import 'package:paramov/screens/login_screen.dart';
import 'package:paramov/screens/registration_screen.dart';
import 'package:flutter/material.dart';
import 'package:animated_text_kit/animated_text_kit.dart';
import  'package:paramov/screens/vital_screen.dart';
import 'package:paramov/widget/purple_overlay.dart';
import 'package:paramov/widget/add_button.dart';
import 'package:paramov/components/text_design.dart';

class WelcomeScreen extends StatefulWidget {
  static const String id = 'welcome_screen';

  @override
  State<WelcomeScreen> createState() => _WelcomeScreenState();
}

class _WelcomeScreenState extends State<WelcomeScreen> {
  @override
  void initState() {
    // TODO: implement initState
    super.initState();
    if (!kIsWeb) {
      NotificationController.initializeContext(context);
      NotificationController.startListeningNotificationEvents();
      NotificationController.requestFirebaseToken();
    }
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Text(
              'Get Instant Monitoring for Patients',
              style: TextStyle(
                color: Colors.black,
                fontSize: 26.0,
                fontWeight: FontWeight.w700,
              ),
              textAlign: TextAlign.center,
            ),
            SizedBox(height: 20.0),
```

```
          Text(
            'Paramov streams real-time patient health insights for abnormality detection and
interventions.',
            style: TextStyle(
              color: Colors.black,
              fontSize: 15.0,
              fontWeight: FontWeight.w400,
            ),
            textAlign: TextAlign.center,
          ),

          SizedBox(height: 25.0),
          Stack(
            children: [
              // Background image
              // Tilted purple overlay
              Purple_overlay(),
              Center(
                child: Image.asset(
                  'images/doc1.png', // Replace with your image path
                  width: MediaQuery.of(context).size.width *
                      0.8, // 80% of screen width
                  height: MediaQuery.of(context).size.height * 0.5,
                ),
              ),
            ],
          ),
          SizedBox(height: 5.0), // Add spacing between image and button
          AddButton(
            text: 'Get Started',
            width: 0.8,
            height: 0.15,
            onTap: () async {
              await NotificationController.createNewNotification(
                  "Application Started", "Testing Notification");
              Navigator.pushNamedAndRemoveUntil(context,
                  RegistrationScreen.id, (Route<dynamic> route) => false);
            },
          ),
        ],
      ),
    ),
  );
 }
}

class _VitalScreenState extends State<VitalScreen> {
  final _auth = FirebaseAuth.instance;
  String? need1;
```

```
@override
void initState() {
  super.initState();
  if (!kIsWeb) {
    NotificationController.initializeContext(context);
    NotificationController.startListeningNotificationEvents();
    NotificationController.requestFirebaseToken();
  }
  getCurrentUser();
}

void getCurrentUser() async {
  try {
    final user = await _auth.currentUser;

   if (user != null) {
      logedinuser = user;
      print(logedinuser);
      MyName = user.displayName!;
      print(MyName);
    }
  } catch (e) {
    print(e);
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.white,
    appBar: AppBar(
      leading: null,
      actions: <Widget>[
      IconButton(
            icon: Icon(Icons.close),
            onPressed: () {
              //Implement logout functionality

              // msgStream();
              _auth.signOut();
              Navigator.pushNamedAndRemoveUntil(
                  context, WelcomeScreen.id, (Route<dynamic> route) => false);
            }),
      ],
      title: Row(
        children: [
        Image.asset(
            'images/logo.png',
            width: 30.0,
            height: 30.0,
          ),
```

```
          Text('Paramov'),
        ],
      ),
      backgroundColor: Colors.purple.withOpacity(0.1),
    ),
    body: SafeArea(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        crossAxisAlignment: CrossAxisAlignment.stretch,
        children: <Widget>[
          MessageStream(),
        ],
      ),
    ),
  );
 }
}

class MessageStream extends StatelessWidget {
 DatabaseReference ref = FirebaseDatabase.instance.ref();
 @override
 Widget build(BuildContext context) {
  return //stream helps display snapshot of messages in column
      StreamBuilder(

//declaring StreamBuilder as QuerySnapshot
          //alaways declare any function beforehand
          stream: ref.onValue,
          builder: (context, snapshot) {
            if (!snapshot.hasData) {
            return Center(
                child: CircularProgressIndicator(
                    backgroundColor: Colors.lightBlueAccent),
              );
            } else {
              Map<dynamic, dynamic> messages = snapshot.data!.snapshot.value
                  as dynamic; //to access the data in async
              //async from steambuilder contains QuerySnapshot, QuerySnapshot contains
document list

              var need1;
              var need2;
              var need3;
              var need4;
              var temperatureText;
              //typecasting object to String
              need1 = messages['need1'];
              need2 = messages['need2'];
              need3 = messages['need3'];
              need4 = messages['need4'];
```

```
temperatureText = messages['temperature'];
nmethod(need1, need2, need3, need4);
// int level1 = need1;
// int level = 1500;
// int consumption = level - level1;
return Center(
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Text(
        'Hey Vanshaj,', //${MyName}
        style: GoogleFonts.quintessential(
        textStyle: TextStyle(
            fontSize: 25.0, fontWeight: FontWeight.w600),
        ),
        textAlign: TextAlign.start,
      ),
      SizedBox(height: 20.0),
      // Large container with rounded corners
      Container(
        height: 300.0,
        width: 340.0, // Occupy full width
        decoration: BoxDecoration(
          borderRadius: BorderRadius.circular(25.0),
          color: Colors.purple.withOpacity(0.1),
        ),
        child: Column(
          children: [
          Image.asset(
              'images/image1.png', // Replace with your image path
              width: 150.0, // 80% of screen width
              height: 150.0,
            ),
            SizedBox(height: 20.0),
            Text(
            'Temperature: ${temperatureText} °F',
              style: GoogleFonts.quintessential(
              textStyle: TextStyle(
                   fontSize: 25.0,
                   fontWeight: FontWeight.w600),
              ),
            ),
          ],
        ),
      ),
      const SizedBox(height: 30.0), // Add spacing

      // Row with two containers
      Row(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        children: [
```

```
              Container(
                height: 150.0,
                width: 150.0,
                decoration: BoxDecoration(
                  borderRadius: BorderRadius.circular(25.0),
                  color: need1 < 11
                      ? Colors.purple.withOpacity(0.1)
                      : Colors.purple.withOpacity(0.5),
                ),
                child:  Center(
                  child:   Text(
                  'Assisstance',
                  style: GoogleFonts.quintessential(
                    textStyle: TextStyle(
                        fontSize: 25.0,
                        fontWeight: FontWeight.w600),
                  ),
                )),
              ),
              Container(
                height: 150.0,
                width: 150.0,
                decoration: BoxDecoration(
                  borderRadius: BorderRadius.circular(25.0),
                  color: need2 < 12
                      ? Colors.purple.withOpacity(0.1)
                      : Colors.purple.withOpacity(0.5),
                ),
                 child: Center(
                  child:   Text(
                  'Water',
                  style: GoogleFonts.quintessential(
                    textStyle: TextStyle(
                        fontSize: 25.0,
                        fontWeight: FontWeight.w600),
                  ),
                )),),
            ],
          ),
          const SizedBox(height: 30.0), // Add spacing
          // Row with two remaining containers
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
            children: [
Container(
                height: 150.0,
                width: 150.0,
                decoration: BoxDecoration(
                  borderRadius: BorderRadius.circular(25.0),
```

```
color: need3 < 23
    ? Colors.purple.withOpacity(0.1)
    : Colors.purple.withOpacity(0.5),
),
child: Center(
  child:  Text(
  'Food',
  style: GoogleFonts.quintessential(
    textStyle: TextStyle(
        fontSize: 25.0,
        fontWeight: FontWeight.w600),
  ),
 )
 ),
),
Container(
  height: 150.0,
  width: 150.0,
  decoration: BoxDecoration(
    borderRadius: BorderRadius.circular(25.0),
    color: need4 < 23
        ? Colors.purple.withOpacity(0.1)
        : Colors.purple.withOpacity(0.5),
  ),
  child: Center(
    child:  Text(
    'Restroom',
    style: GoogleFonts.quintessential(
      textStyle: TextStyle(
          fontSize: 25.0,
          fontWeight: FontWeight.w600
      ),
    ),
  )
  ),
),
],
),
],
),
);
}
```

Login and Registration Screen Code

```
import 'package:flutter/material.dart';
import 'package:paramov/components/rounded_button.dart';
import 'package:paramov/constants.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:paramov/screens/vital_screen.dart';
import 'package:modal_progress_hud_nsn/modal_progress_hud_nsn.dart';
import 'package:paramov/widget/add_button.dart';
import 'package:paramov/widget/purple_overlay.dart';
import 'package:cloud_firestore/cloud_firestore.dart';

enum LoginState { Login, Register }

class RegistrationScreen extends StatefulWidget {
  static const String id = 'Registration_screen';
  @override
  _RegistrationScreenState createState() => _RegistrationScreenState();
}

class _RegistrationScreenState extends State<RegistrationScreen> {
  final _auth = FirebaseAuth.instance;
  bool showspinner = false;
  late String name;
  late String email;
  late String password;
  final _formKey = GlobalKey<FormState>(); // Key for form validation
  LoginState _currentState = LoginState.Login;
  void toggleState() {
    setState(() {
      _currentState = _currentState == LoginState.Login
          ? LoginState.Register
          : LoginState.Login;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(_currentState == LoginState.Login ? 'Login' : 'Register'),
      ),
      body: ModalProgressHUD(
        inAsyncCall: showspinner,
        child: SingleChildScrollView(
        padding: EdgeInsets.all(16.0),
        child: Form(
            key: _formKey,
            child: Column(
              crossAxisAlignment: CrossAxisAlignment.start,
```

```
children: [
  Stack(
  children: [
    // Background image
    // Tilted purple overlay
    Purple_overlay(),
    Center(
      child: Padding(
        padding: const EdgeInsets.only(top: 10.0),
        child: Image.asset(
          'images/logo.png', // Replace with your image path
          width: MediaQuery.of(context).size.width *
              0.3, // 80% of screen width
          height: MediaQuery.of(context).size.height * 0.3,
        ),
      ),
    ),
  ],
  ),
  // Email field
  TextField(
    keyboardType: TextInputType.emailAddress,
    textAlign: TextAlign.center,
    onChanged: (value) {
      //Do something with the user input.
      email = value;
    },
    decoration: kInput.copyWith(hintText: 'Enter your email'),
  ),
  SizedBox(height: 10.0),
  // Password field
  TextField(
    obscureText: true, //this change txt into dot or star form
    textAlign: TextAlign.center,
    onChanged: (value) {
      //Do something with the user input.
      password = value;
    },
    decoration: kInput.copyWith(hintText: 'Enter your password.'),
  ),
  // Additional fields for registration (conditionally displayed)
  if (_currentState == LoginState.Register)
    SizedBox(height: 10.0),
  if (_currentState == LoginState.Register)
    TextField(
      textAlign: TextAlign.center,
      onChanged: (value) {
        //Do something with the user input.
        name = value;
      },
```

```
    decoration: kInput.copyWith(hintText: 'Enter your Name.'),
  ),
SizedBox(height: 20.0),
// Login/Register button based on current state
AddButton(
  text:
      _currentState != LoginState.Login ? 'Register' : 'Login',
  width: 0.3,
  height: 0.1,
  onTap: () async {
    if (_currentState != LoginState.Login) {
      // print(email);
      // print(password);
      setState(() {
        //to show loading animation
        showspinner = true;
      });
      try {
        final newuser =
            await _auth.createUserWithEmailAndPassword(
                email: email, password: password);
        if (newuser.user != null) {
          await FirebaseFirestore.instance
              .collection('Users')
              .doc(newuser.user!.displayName)
              .set({'email': email, 'name': name});
          Navigator.pushNamedAndRemoveUntil(context,
              VitalScreen.id, (Route<dynamic> route) => false);
        }
        setState(() {
          //to show loading animation
          showspinner = false;
        });
      } catch (e) {
        print(e);
      }
    }
    if (_currentState == LoginState.Login) {
      setState(() {
        //to show loading animation
        showspinner = true;
      });
      try {
        final user = await _auth.signInWithEmailAndPassword(
            email: email, password: password);
        if (user != null) {
          Navigator.pushAndRemoveUntil(
              context,
```

```
                        MaterialPageRoute(
                          builder: (context) => VitalScreen(
                              user: email,
                            )),
                        (Route<dynamic> route) => false);
                  }
                  setState(() {
                    //to show loading animation
                    showspinner = false;
                  });
                } catch (e) {
                  print(e);
                }
              }
            },
          ),
          // Toggle button to switch between Login and Register
          TextButton(
            onPressed: toggleState,
            child: Text(_currentState != LoginState.Login
                ? 'Existing User? Login Here'
                : 'New User? Register Here'),
          ),
        ],
      ),
    ),
  ),
),
);
}
}
```

VitalScreen Code:

```
import 'dart:convert';
import 'package:flutter/foundation.dart';
import 'package:paramov/components/notification_service.dart';
import 'package:paramov/screens/welcome_screen.dart';
import 'package:flutter/material.dart';
import 'package:paramov/components/Nmethod.dart';
import 'package:paramov/constants.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_database/firebase_database.dart';
import 'package:paramov/widget/purple_overlay.dart';
import 'package:google_fonts/google_fonts.dart';

final _firestore = FirebaseFirestore.instance;
User? logedinuser;
var MyName;
```

```
class VitalScreen extends StatefulWidget {
  final String user;
  VitalScreen({required this.user});

  static const String id = 'Vital_screen';

  @override
  _VitalScreenState createState() => _VitalScreenState();
}

class _VitalScreenState extends State<VitalScreen> {
  final _auth = FirebaseAuth.instance;
  String? need1;
  @override
  void initState() {
    super.initState();
    if (!kIsWeb) {
      NotificationController.initializeContext(context);
      NotificationController.startListeningNotificationEvents();
      NotificationController.requestFirebaseToken();
    }
    getCurrentUser();
  }

  void getCurrentUser() async {
    try {
      final user = await _auth.currentUser;

      if (user != null) {
        logedinuser = user;
        print(logedinuser);
        MyName = user.displayName!;
        print(MyName);
      }
    } catch (e) {
      print(e);
    }
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      appBar: AppBar(
        leading: null,
        actions: <Widget>[
        IconButton(
              icon: Icon(Icons.close),
              onPressed: () {
```

```
              //Implemented logout functionality

              // msgStream();
              _auth.signOut();
              Navigator.pushNamedAndRemoveUntil(
                  context, WelcomeScreen.id, (Route<dynamic> route) => false);
            }),
        ],
        title: Row(
          children: [
          Image.asset(
              'images/logo.png',
              width: 30.0,
              height: 30.0,
            ),
            Text('Paramov'),
          ],
        ),
        backgroundColor: Colors.purple.withOpacity(0.1),
      ),
      body: SafeArea(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: <Widget>[
            MessageStream(),
          ],
        ),
      ),
    );
  }
}

class MessageStream extends StatelessWidget {
  DatabaseReference ref = FirebaseDatabase.instance.ref();
  @override
  Widget build(BuildContext context) {
    return //stream helps display snapshot of messages in column
        StreamBuilder(
            //declaring StreamBuilder as QuerySnapshot
            //alaways declare any function beforehand
            stream: ref.onValue,
            builder: (context, snapshot) {
              if (!snapshot.hasData) {
              return Center(
                  child: CircularProgressIndicator(
                      backgroundColor: Colors.lightBlueAccent),
                );
```

```
} else {
  Map<dynamic, dynamic> messages = snapshot.data!.snapshot.value
      as dynamic; //to access the data in async

  var need1;
  var need2;
  var need3;
  var need4;
  var temperatureText;
  //typecasting object to String
  need1 = messages['need1'];
  need2 = messages['need2'];
  need3 = messages['need3'];
  need4 = messages['need4'];
  temperatureText = messages['temperature'];
  nmethod(need1, need2, need3, need4);
  // int level1 = need1;
  // int level = 1500;
  // int consumption = level - level1;
  return Center(
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Text(
          'Hey Vanshaj,', //${MyName}
          style: GoogleFonts.quintessential(
          textStyle: TextStyle(
              fontSize: 25.0, fontWeight: FontWeight.w600),
          ),
          textAlign: TextAlign.start,
        ),
        SizedBox(height: 20.0),
        // Large container with rounded corners
        Container(
          height: 300.0,
          width: 340.0, // Occupy full width
          decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(25.0),
            color: Colors.purple.withOpacity(0.1),
          ),
          child: Column(
            children: [
            Image.asset(
                'images/image1.png', // Replace with your image path
                width: 150.0, // 80% of screen width
                height: 150.0,
              ),
              SizedBox(height: 20.0),
              Text(
```

```
        'Temperature: ${temperatureText} °F',
        style: GoogleFonts.quintessential(
        textStyle: TextStyle(
            fontSize: 25.0,
            fontWeight: FontWeight.w600),
        ),
      ),
    ],
  ),
),
const SizedBox(height: 30.0), // Add spacing

// Row with two containers
Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    Container(
      height: 150.0,
      width: 150.0,
      decoration: BoxDecoration(
        borderRadius: BorderRadius.circular(25.0),
        color: need1 < 11
            ? Colors.purple.withOpacity(0.1)
            : Colors.purple.withOpacity(0.5),
      ),
      child:  Center(
        child:   Text(
        'Assisstance',
        style: GoogleFonts.quintessential(
          textStyle: TextStyle(
            fontSize: 25.0,
            fontWeight: FontWeight.w600),
        ),
      )),
    ),
    Container(
      height: 150.0,
      width: 150.0,
      decoration: BoxDecoration(
        borderRadius: BorderRadius.circular(25.0),
        color: need2 < 12
            ? Colors.purple.withOpacity(0.1)
            : Colors.purple.withOpacity(0.5),
      ),
      child:  Center(
        child:   Text(
        'Water',
```

```
          style: GoogleFonts.quintessential(
            textStyle: TextStyle(
                fontSize: 25.0,
                fontWeight: FontWeight.w600),
          ),
        )),
      ),
    ],
  ),
  const SizedBox(height: 30.0), // Add spacing

  // Row with two remaining containers
  Row(
    mainAxisAlignment: MainAxisAlignment.spaceEvenly,
    children: [
      Container(
        height: 150.0,
        width: 150.0,
        decoration: BoxDecoration(
          borderRadius: BorderRadius.circular(25.0),
          color: need3 < 23
              ? Colors.purple.withOpacity(0.1)
              : Colors.purple.withOpacity(0.5),
        ),
        child:  Center(
          child:   Text(
          'Food',
          style: GoogleFonts.quintessential(
            textStyle: TextStyle(
                fontSize: 25.0,
                fontWeight: FontWeight.w600),
          ),
        )),
      ),
      Container(
        height: 150.0,
        width: 150.0,
        decoration: BoxDecoration(
          borderRadius: BorderRadius.circular(25.0),
          color: need4 < 23
              ? Colors.purple.withOpacity(0.1)
              : Colors.purple.withOpacity(0.5),
        ),
        child:  Center(
          child:   Text(
          'Restroom',
          style: GoogleFonts.quintessential(
            textStyle: TextStyle(
                fontSize: 25.0,
                fontWeight: FontWeight.w600),
```

```
                    ),
                  )),
                ),
              ],
            ),
          ],
        ),
      );
    }
  });
 }
}
```

Notification Functionality Code:

```
import 'package:awesome_notifications/awesome_notifications.dart';
import 'package:awesome_notifications_fcm/awesome_notifications_fcm.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:http/http.dart' as http;

import '../main.dart';

String? bd;
String? ttl;

class NotificationController extends ChangeNotifier {
  static final NotificationController _instance =
      NotificationController._internal();

  factory NotificationController() {
    return _instance;
  }

  String _firebaseToken = '';
  String get firebaseToken => _firebaseToken;

  String _nativeToken = '';
  String get nativeToken => _nativeToken;

  static ReceivedAction? initialAction;

  NotificationController._internal();
  static BuildContext? context;

  static initializeContext(BuildContext context1) {
    context = context1;
  }
```

```
static Future<void> initializeLocalNotifications() async {
  await AwesomeNotifications().initialize(
      null, //'resource://drawable/res_app_icon',//
      [
        NotificationChannel(
            channelKey: 'alerts',
            channelName: 'Alerts',
            channelDescription: 'Notification tests as alerts',
            playSound: true,
            onlyAlertOnce: true,
            groupAlertBehavior: GroupAlertBehavior.Children,
            importance: NotificationImportance.High,
            defaultPrivacy: NotificationPrivacy.Private,
            defaultColor: Colors.deepPurple,
            ledColor: Colors.deepPurple)
      ],
      debug: true);

  // Get initial notification action is optional
  initialAction = await AwesomeNotifications()
      .getInitialNotificationAction(removeFromActionEvents: false);
}

static Future<void> initializeRemoteNotifications(
    {required bool debug}) async {
  await Firebase.initializeApp();
  await AwesomeNotificationsFcm().initialize(
      onFcmSilentDataHandle: NotificationController.mySilentDataHandle,
      onFcmTokenHandle: NotificationController.myFcmTokenHandle,
      onNativeTokenHandle: NotificationController.myNativeTokenHandle,
      licenseKeys:
          // On this example app, the app ID / Bundle Id are different
          // for each platform, so i used the main Bundle ID + 1 variation
          [
        // me.carda.awesomeNotificationsFcmExample
        'B3J3yxQbzzyz0KmkQR6rDlWB5N68sTWTEMV7k9HcPBroUh4RZ/Og2Fv6Wc/lE'
            '2YaKuVY4FUERlDaSN4WJ0lMiiVoYIRtrwJBX6/fpPCbGNkSGuhrx0Rekk'
            '+yUTQU3C3WCVf2D534rNF3OnYKUjshNgQN8do0KAihTK7n83eUD60=',

        // me.carda.awesome_notifications_fcm_example
        'UzRlt+SJ7XyVgmD1WV+7dDMaRitmKCKOivKaVsNkfAQfQfechRveuKblFnCp4'
            'zifTPgRUGdFmJDiw1R/rfEtTIlZCBgK3Wa8MzUV4dypZZc5wQIIVsiqi0Zhaq'
            'YtTevjLl3/wKvK8fWaEmUxdOJfFihY8FnlrSA48FW94XWIcFY=',
      ],
      debug: debug);
}

/// *****
///     NOTIFICATION EVENTS LISTENER
/// *****
/// Notifications events are only delivered after call this method
```

```
static Future<void> startListeningNotificationEvents() async {
  AwesomeNotifications()
      .setListeners(onActionReceivedMethod: onActionReceivedMethod);
}

static Future<void> getInitialNotificationAction() async {
  ReceivedAction? receivedAction = await AwesomeNotifications()
      .getInitialNotificationAction(removeFromActionEvents: true);
  if (receivedAction == null) return;

  // Fluttertoast.showToast(
  //     msg: 'Notification action launched app: $receivedAction',
  //  backgroundColor: Colors.deepPurple
  // );
  print('Notification action launched app: $receivedAction');
}

/// *****
///     NOTIFICATION EVENTS
/// *****
///
@pragma('vm:entry-point')
static Future<void> onActionReceivedMethod(
  ReceivedAction receivedAction) async {
  print(receivedAction);
  if (receivedAction.actionType == ActionType.SilentAction ||
      receivedAction.actionType == ActionType.SilentBackgroundAction) {
    // For background actions, you must hold the execution until the end
    print(
        'Message sent via notification input: "${receivedAction.buttonKeyInput}"');
    // await executeLongTaskInBackground();
  } else {
    try {
      await Navigator.pushNamed(context!, 'plan');
      print("Navigation successful");
    } catch (e, stackTrace) {
      print("Navigation error: $e");
    }
    print("done");
  }
}

/// *****
///     REMOTE NOTIFICATION EVENTS
/// *****

/// Use this method to execute on background when a silent data arrives
/// (even while terminated)
@pragma("vm:entry-point")
```

```
static Future<void> mySilentDataHandle(FcmSilentData silentData) async {
  Fluttertoast.showToast(
      msg: 'Silent data received',
      backgroundColor: Colors.blueAccent,
      textColor: Colors.white,
      fontSize: 16);

  print('"SilentData": ${silentData.toString()}');

  if (silentData.createdLifeCycle != NotificationLifeCycle.Foreground) {
    print("bg");
  } else {
    print("FOREGROUND");
  }

  print('mySilentDataHandle received a FcmSilentData execution');
  await executeLongTaskInBackground();
}

/// Use this method to detect when a new fcm token is received
@pragma("vm:entry-point")
static Future<void> myFcmTokenHandle(String token) async {
  Fluttertoast.showToast(
      msg: 'Fcm token received',
      backgroundColor: Colors.blueAccent,
      textColor: Colors.white,
      fontSize: 16);
  debugPrint('Firebase Token:"$token"');

  _instance._firebaseToken = token;
  _instance.notifyListeners();
}

/// Use this method to detect when a new native token is received
@pragma("vm:entry-point")
static Future<void> myNativeTokenHandle(String token) async {
  Fluttertoast.showToast(
      msg: 'Native token received',
      backgroundColor: Colors.blueAccent,
      textColor: Colors.white,
      fontSize: 16);
  debugPrint('Native Token:"$token"');

  _instance._nativeToken = token;
  _instance.notifyListeners();
}

/// *****
///     REQUESTING NOTIFICATION PERMISSIONS
/// *****
///
```

```
static Future<bool> displayNotificationRationale() async {
  bool userAuthorized = false;
  BuildContext context =
      Paramov.navigatorKey.currentContext!; //static using navigator
  await showDialog(
      context: context,
      builder: (BuildContext ctx) {
        return AlertDialog(
          title: Text('Get Notified!',
              style: Theme.of(context).textTheme.titleLarge),
          content: Column(
            mainAxisSize: MainAxisSize.min,
            children: [
              Row(
                children: [
                Expanded(
                    child: Image.asset(
                      'images/logo.png',
                      height: MediaQuery.of(context).size.height * 0.3,
                      fit: BoxFit.fitWidth,
                    ),
                  ),
                ],
              ),
              const SizedBox(height: 20),
              const Text(
                  'Allow Awesome Notifications to send you beautiful notifications!'),
            ],
          ),
          actions: [
            TextButton(
                onPressed: () {
                  Navigator.of(ctx).pop();
                },
                child: Text(
                  'Deny',
                  style: Theme.of(context)
                      .textTheme
                      .titleLarge
                      ?.copyWith(color: Colors.red),
                )),
            TextButton(
                onPressed: () async {
                  userAuthorized = true;
                  Navigator.of(ctx).pop();
                },
                child: Text(
                  'Allow',
```

```
              style: Theme.of(context)
                  .textTheme
                  .titleLarge
                  ?.copyWith(color: Colors.deepPurple),
            )),
        ],
      );
    });
  return userAuthorized &&
      await AwesomeNotifications().requestPermissionToSendNotifications();
}

static Future<void> executeLongTaskInBackground() async {
  print("starting long task");
  await Future.delayed(const Duration(seconds: 4));
  final url = Uri.parse("http://google.com");
  final re = await http.get(url);
  print(re.body);
  print("long task done");
}

static Future<void> resetBadge() async {
  await AwesomeNotifications().resetGlobalBadge();
}

/// *****
///    REMOTE TOKEN REQUESTS
/// *****

static Future<String> requestFirebaseToken() async {
  if (await AwesomeNotificationsFcm().isFirebaseAvailable) {
    try {
      return await AwesomeNotificationsFcm().requestFirebaseAppToken();
    } catch (exception) {
      debugPrint('$exception');
    }
  } else {
    debugPrint('Firebase is not available on this project');
  }
  return '';
}

///  *****
///    NOTIFICATION CREATION METHODS
/// *****
///
static Future<void> createNewNotification(bd, ttl) async {
  bool isAllowed = await AwesomeNotifications().isNotificationAllowed();
  if (!isAllowed) isAllowed = await displayNotificationRationale();
  if (!isAllowed) return;
```

```
await AwesomeNotifications().createNotification(
    // schedule: NotificationAndroidCrontab.minutely(
    //  allowWhileIdle: true,
    //  referenceDateTime: DateTime(
    //      DateTime.now().year,
    //      DateTime.now().month,
    //      DateTime.now().day,
    //      DateTime.now().hour,
    //      DateTime.now().minute,
    //      5), // 30th second
    // ),
    content: NotificationContent(
        id: -1, // -1 is replaced by a random number
        channelKey: 'alerts',
        autoDismissible: false,
        title: ttl,
        // 'Huston! The eagle has landed!',
        //locked: true,
        //  criticalAlert: true,
        color: Colors.redAccent,
        body: bd,
        // "A small step for a man, but a giant leap to Flutter's community!",
        bigPicture: 'https://storage.googleapis.com/cms-storage-
bucket/d406c736e7c4c57f5f61.png',
        largeIcon: 'https://storage.googleapis.com/cms-storage-
bucket/0dbfcc7a59cd1cf16282.png',
        //'asset://assets/images/balloons-in-sky.jpg',
        notificationLayout: NotificationLayout.BigPicture,
        payload: {'notificationId': '1234567890'}),
    actionButtons: [
      NotificationActionButton(
          showInCompactView: true,
          color: Colors.white,
          key: 'REDIRECT',
          label: 'Redirect',
          autoDismissible: false),
      NotificationActionButton(
          color: Colors.white,
          showInCompactView: true,
          key: 'REPLY',
          label: 'Reply Message',
          requireInputText: true,
          actionType: ActionType.SilentAction),
      NotificationActionButton(
          color: Colors.white,
          showInCompactView: true,
          key: 'DISMISS',
          label: 'Dismiss',
          actionType: ActionType.DismissAction,
          isDangerousOption: true)
```

```
    ]);
  }
}
```

Firebase Code:

```
// File generated by FlutterFire CLI.
// ignore_for_file: lines_longer_than_80_chars, avoid_classes_with_only_static_members

import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;
import 'package:flutter/foundation.dart'
  show defaultTargetPlatform, kIsWeb, TargetPlatform;

/// Default [FirebaseOptions] for use with your Firebase apps.
///
/// Example:
/// dart
/// import 'firebase_options.dart';
/// // ...
/// await Firebase.initializeApp(
///  options: DefaultFirebaseOptions.currentPlatform,
/// );
///
class DefaultFirebaseOptions {
 static FirebaseOptions get currentPlatform {
  if (kIsWeb) {
   return web;
  }
  switch (defaultTargetPlatform) {

   case TargetPlatform.android:
    return android;

   case TargetPlatform.iOS:
    return ios;

   case TargetPlatform.macOS:
    return macos;

   case TargetPlatform.windows:
    throw UnsupportedError(
     'DefaultFirebaseOptions have not been configured for windows - '
     'you can reconfigure this by running the FlutterFire CLI again.',
    );

   case TargetPlatform.linux:
    throw UnsupportedError(
     'DefaultFirebaseOptions have not been configured for linux - '
     'you can reconfigure this by running the FlutterFire CLI again.',
    );
```

```
      default:
        throw UnsupportedError(
          'DefaultFirebaseOptions are not supported for this platform.',
        );
    }
  }

  static const FirebaseOptions web = FirebaseOptions(
    apiKey: 'AIzaSyDNfLEkXbazD08UOQLhL6s2XBO2Qr5HwVo',
    appId: '1:610685915691:web:d93ec0bdc069a0834dd06e',
    messagingSenderId: '610685915691',
    projectId: 'paramov-b4f20',
    authDomain: 'paramov-b4f20.firebaseapp.com',
    storageBucket: 'paramov-b4f20.appspot.com',
  );

  static const FirebaseOptions android = FirebaseOptions(
    apiKey: 'AIzaSyBMkBO4AaA_rrtyX4YKoF8zoLhV0iCc_LU',
    appId: '1:610685915691:android:93f1a1ebac2a45d54dd06e',
    messagingSenderId: '610685915691',
    projectId: 'paramov-b4f20',
    storageBucket: 'paramov-b4f20.appspot.com',
  );

  static const FirebaseOptions ios = FirebaseOptions(
    apiKey: 'AIzaSyDimdMczdSX48DecZeXzp_2gfXLZKs0jmE',
    appId: '1:610685915691:ios:35d9926a4c0d59c94dd06e',
    messagingSenderId: '610685915691',
    projectId: 'paramov-b4f20',
    storageBucket: 'paramov-b4f20.appspot.com',
    iosBundleId: 'com.example.paramov',
  );

  static const FirebaseOptions macos = FirebaseOptions(
    apiKey: 'AIzaSyDimdMczdSX48DecZeXzp_2gfXLZKs0jmE',
    appId: '1:610685915691:ios:abdfc896f566c74a4dd06e',
    messagingSenderId: '610685915691',
    projectId: 'paramov-b4f20',
    storageBucket: 'paramov-b4f20.appspot.com',
    iosBundleId: 'com.example.paramov.RunnerTests',
  );
}
```

PAPER PUBLICATION STATUS

acceptance mail

79 of many

Acceptance Notification (Paper ID 753) - IEEE 2nd TQCEBT 2024

External Inbox x

Microsoft CMT <email@msr-cmt.org> Wed, Feb 28, 10:41AM

to me, lawanyaraj

Dear Sir/Madam

We are pleased to inform you that your paper has been accepted for the Oral Presentation as a full paper for the- "IEEE Sponsored Trends in Quantum Computing & Emerging Business Technologies-TQCEBT 24". Pune, Maharashtra, India with following reviewers' comment.

All accepted and presented papers will be submitted to IEEE Xplore for the further publication and will be indexed by Scopus.

We are looking forward to seeing you in Pune, Maharashtra, India!

You should finish the three steps below before the deadline or you will be deemed to withdraw your paper:

Step 1: Complete the Registration Process (The last date of payment is 31 January 2024)

FOR INDIAN/ NON-INDIAN AUTHOR

CHRIST(DEEMED TO BE UNIVERSITY)-LAVASA CAMPUS

CHRIST
(DEEMED TO BE UNIVERSITY)
LAVASA, PUNE, INDIA

TQCEBT 24-Regular Indian Receipt

Name	D SAISANTHIYA
Address	SRMIST
Pincode	603203
Email	saisantd@srmist.edu.in
Category	Student
Amount	5000
Total Amount	5000.0
Transaction Id	cabd721daa79a592b176
Transaction Date	05-03-2024

Click here to Print

© 2016 www.southindianbank.com. All Rights Reserved.

PLAGIARISM REPORT

vinay_major_project_report_7percent.pdf

ORIGINALITY REPORT

7%	4%	3%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

6 www.coursehero.com
Internet Source

2 mrcet.com
Internet Source

3 Tarannum Khan, Manju K. Chattopadhyay. "Smart health monitoring system", 2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC), 2017
Publication

4 Ali I. Siam, Mohammed A. El-Affendi, Atef Abou Elazm, Ghada M. El-Banby et al. "Portable and Real-Time IoT-Based Healthcare Monitoring System for Daily Medical Applications", IEEE Transactions on Computational Social Systems, 2022
Publication

5 Submitted to Intercollege
Student Paper

dokumen.pub
Internet Source

7	www.hindawi.com Internet Source	<1%
8	www.journal.esrgroups.org Internet Source	<1%
9	Visalakshi Annepu, Adnan Naji Jameel AL-Tamimi, H.S. S. Aljibori, Naga Raju Challa et al. "Implementation of an Efficient IoT Enabled Automated Paralysis Healthcare System", 2023 IEEE 8th International Conference on Engineering Technologies and Applied Sciences (ICETAS), 2023 Publication	<1%
10	Submitted to Webster University Student Paper	<1%
11	eprints.utm.my Internet Source	<1%
12	link.springer.com Internet Source	<1%
13	Submitted to Nottingham Trent University Student Paper	<1%
14	Submitted to Softwarica College Of IT & E-Commerce Student Paper	<1%
15	Submitted to Queen Mary and Westfield College Student Paper	<1%

16	healthmanagement.org Internet Source	<1%
17	Submitted to Liberty University Student Paper	<1%
18	www.ijraset.com Internet Source	<1%
19	developer.ibm.com Internet Source	<1%
20	ijsra.net Internet Source	<1%
21	Xiao Fan, Wanrong Sun, Aifeng Ren, Dou Fan, Nan Zhao, Daniyal Haider, Xiaodong Yang, Qammer H. Abbasi. "Detection and Diagnosis of Paralysis Agitans", IEEE Access, 2018 Publication	<1%
22	Submitted to Ravensbourne Student Paper	<1%
23	S Akshay, S Dhanush, Sujayeendra Rao. "iAlert: An Alert System based on Eye Gaze for Human Assistance", 2023 IEEE 3rd International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET), 2023 Publication	<1%

24	"Sustainable Security Practices Using Blockchain, Quantum and Post-Quantum Technologies for Real Time Applications", Springer Science and Business Media LLC, 2024 Publication	<1%
25	Ninet Varghese, D Hepsiba, L.D. Vijay Anand. "Health Monitoring and Observatory System for Paralysed Patients using Blynk Application", 2022 6th International Conference on Devices, Circuits and Systems (ICDCS), 2022 Publication	<1%
26	Submitted to The University of the West of Scotland Student Paper	<1%
27	sameermitter.yolasite.com Internet Source	<1%
28	Submitted to University of Hertfordshire Student Paper	<1%
29	Submitted to University of Wales, Lampeter Student Paper	<1%
30	www.ncbi.nlm.nih.gov Internet Source	<1%
31	www.eurchembull.com Internet Source	<1%

32	www.geeksforgeeks.org Internet Source	<1 %
33	Babeș-Bolyai University Publication	<1 %
34	Submitted to Baker College Student Paper	<1 %
35	Submitted to Montana Tech of the University of Montana Student Paper	<1 %
36	Submitted to University of Greenwich Student Paper	<1 %
37	vdoc.pub Internet Source	<1 %
38	www.cse.tkk.fi Internet Source	<1 %
39	M Vijay Kumar, Kaythry Pandurangan, R Vinu. "Automated Paralysis Patient Monitoring System", 2021 IEEE National Biomedical Engineering Conference (NBEC), 2021 Publication	<1 %
40	www.igi-global.com Internet Source	<1 %
41	Submitted to Higher Education Commission Pakistan Student Paper	<1 %

42	Preeti, Chhavi Rana. "IoT and Cloud Based health monitoring system Using Machine learning", 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), 2022 Publication	<1%
43	green.org Internet Source	<1%
44	www.aqara.com Internet Source	<1%
45	www.reddit.com Internet Source	<1%
46	J Bethanney Janney, U Aakesh, Yaswant Rajasekaran, Renny Valentina, S. Krishnakumar, T. Sudhakar. "Design of an automated healthcare system using hand gestures for paraplegic people", 2023 International Conference on Bio Signals, Images, and Instrumentation (ICBSII), 2023 Publication	<1%
47	S Sivanandam, A Abdul Ajees, B Kamesh, W Kevin Samraj. "Supportive System for Hemiplegic Patients with Health Monitoring Capabilities", 2023 9th International Conference on Smart Structures and Systems (ICSSS), 2023 Publication	<1%

48	S. Usha, G. Jeevitha, M. Logesh, M. Karthik, M. Kaviyaa, S. Sri Prasanth. "IoT based Integrated Health Care Monitoring System", 2023 4th International Conference on Smart Electronics and Communication (ICOSEC), 2023 Publication	<1%
49	ebin.pub Internet Source	<1%
50	"Human-Centric Smart Computing", Springer Science and Business Media LLC, 2024 Publication	<1%
51	"International Conference on Innovative Computing and Communications", Springer Science and Business Media LLC, 2023 Publication	<1%
52	Imran khan, Kamran Zeb, Asif Mahmood, Waqar Uddin, Muhammad Adil Khan, Saif-ul-Islam, Hee Je Kim. "Healthcare Monitoring System and transforming Monitored data into Real time Clinical Feedback based on IoT using Raspberry Pi", 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 2019 Publication	<1%
53	Kamalendu Pal. "chapter 6 Semantic Interoperability in Internet of Things", IGI	<1%

Global, 2022
Publication

54	Narasimhaiah Narahari, Rachapudi Praveen Sam. "Canny aspiration paraphernalia framework based healthcare monitoring system and secure medical interoperability", Concurrency and Computation: Practice and Experience, 2023 Publication	<1%
55	en.wikipedia.org Internet Source	<1%
56	eresearch.stikom-bali.ac.id Internet Source	<1%
57	hk.aconf.org Internet Source	<1%
58	Kamalendu Pal. "chapter 2 Internet of Things Applications Architecture for Industrial Interoperability Business Service", IGI Global, 2023 Publication	<1%

Exclude quotes On
Exclude bibliography On
Exclude matches Off

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Deemed to be University u/s 3 of UGC Act, 1956)

Office of Controller of Examinations

REPORT FOR PLAGIARISM CHECK ON THE DISSERTATION/PROJECT REPORTS FOR UG/PG PROGRAMMES

(To be attached in the dissertation/ project report)

1	Name of the Candidate **(IN BLOCK LETTERS)**	Vanshaj Barnwal Vinay Poddar
2	Address of the Candidate	Department of Networking and Communications, School of Computing, SRM Institute of Science & Technology, Kattankulathur. Mobile Number: 8969734227 Department of Networking and Communications, School of Computing, SRM Institute of Science & Technology, Kattankulathur. Mobile Number: 9123675855
3	Registration Number	RA2011032010035 RA2011032010061
4	Date of Birth	21/08/2000 31/08/2000
5	Department	Networking and Communications
6	Faculty	College of Engineering and Technology
7	Title of the Dissertation/Project	Paramov: Smart IOT Based Paralytic Patient Monitoring System
8	Whether the above project /dissertation is done by	Group Project Vanshaj Barnwal, RA2011032010035 Vinay Poddar, RA2011032010061 :
9	Name and address of the Supervisor / Guide	Ms. Saisanthiya D Mail ID: saisantd@srmist.edu.in Mobile Number: 9786454422 Department of Networking and Communications, School of Computing, SRM Institute of Science & Technology, Kattankulathur.
10	Name and address of Co-Supervisor / Co- Guide (if any)	**N/A**

11	Software Used	VS Code Arduino IDE		
12	Date of Verification	23/03/2023		
13	**Plagiarism Details: (to attach the final report from the software)**			
Chapter	**Title of the Chapter**	**Percentage of similarity index (including self citation)**	**Percentage of similarity index (Excluding self-citation)**	**% of plagiarism after excluding Quotes, Bibliography, etc.,**
1	Introduction	-	1%	-
2	Literature Review	-	1%	-
3	Proposed Methodology	2%	1%	-
4	Result	1%	1%	-
5	Conclusion	-	1%	-
6	Future Scope	-	-	-
Appendices			5%	

I / We declare that the above information have been verified and found true to the best of my / our knowledge.

Signature of the Candidate	**Name & Signature of the Staff (Who uses the plagiarism check software)**
Name & Signature of the Supervisor/ Guide	**Name & Signature of the Co-Supervisor/Co-Guide**

Name & Signature of the HOD