SMART ATTENDENCE SYSTEM USING ESP 8266 WITH WEB PORTAL

An Internet of Things Project Report Submitted In partial fulfillment of the requirements for the award of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS & COMMUNICATION ENGINEERING

BY

VEERAMALLA JAMALAIAH
(22761A04J3)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
LAKIREDDY BALIREDDY COLLEGE OF ENGINEERING (AUTONOMOUS)
L.B. REDDY Nagar, Mylavaram-521230,
Approved by AICTE, New Delhi, Affiliated to JNTU, Kakinada
Accredited by NAAC & NBA
Certified by ISO21001:2018

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING (AUTONOMOUS)

L. B. Reddy Nagar, Mylavaram – 521 230.

Approved by AICTE, New Delhi &Affiliated to JNTUK, Kakinada

Accredited by NBA and NAAC

Certified by ISO 9001-2018

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that **INTERNET OF THINGS** report entitled **"SMART ATTENDENCE SYSTEM USING ESP 8266 WITH WEB PORTAL"** is duly presented and submitted by **VEERAMALLA JAMALAIAH (22761A04J3),** in partial fulfillment of requirement for the award of Bachelor of Technology in Electronics and Communication Engineering in Lakireddy Bali Reddy College of Engineering (A), Mylavaram, during the academic year 2024-2025.

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

The Satisfaction that accompanies that the successful completion of any task would be incomplete without the mention of people whose ceaseless co-operation made it possible, whose constant guidance and encouragement crown all efforts with success.

I express my deep sense of gratitude with reverence and humility to Government authorities for extending his support and for providing us with an environment to complete the project successfully.

I would like to express my sincerest gratitude to community people, for his support and for providing us with an environment to complete the project successfully.

I humbly express my thanks to our Principal **Dr**. **K. Appa Rao** for extending his support and for providing us with an environment to complete our project successfully.

I would also like to thank our Vice Principal, **Dr. B. Ramesh Reddy** sir for encouraging us which certainly helped to complete the project in time.

I would like to express our deep gratitude to Dean R & D, **Dr. S. Pichi Reddy** sir for his valuable suggestions.

I deeply indebted to our Head of Department **Dr. G. Srinivasulu,** who modeled us both technically and morally for achieving greater success in life.

I extremely thankful to my guides **Dr. P. Venkata Rao, Mrs. K. Balavani , Dr. Ravi Sekhara Reddy**, Department of Electronics and Communication Engineering, for his excellence guidance, timely and valuable suggestions and encouragement that enabled the success of the project.

I would like to thanks Internet of Things Teaching and non-teaching staff of Electronics and Communication, for his support throughout the period of our project work for making it a successful one.

I thank all my friends who helped me sharing knowledge and by providing material to complete the project in time.

VEERAMALLA JAMALAIAH
(22761A04J3)

Index

List of Figures

Abstract

This paper presents a smart attendance system based on the ESP8266 microcontroller and an integrated web portal that automates student attendance capture, storage, and reporting. The system uses low-cost ESP8266 boards to read unique identifiers (RFID/NFC tags or BLE beacons or simple ID input) at classroom entry points and sends attendance events over Wi-Fi to a centralized server. The web portal provides real-time visualization, secure user authentication, role-based access (administrators, teachers, students), historical attendance logs, automated reports (daily/monthly), and export options (CSV/PDF). Key contributions include a lightweight, reliable firmware for the ESP8266 that minimizes network overhead and power use, a RESTful API for secure, scalable communication between edge nodes and the server, and an intuitive web UI that simplifies monitoring and management. The system supports configurable class schedules, late/absence policies, and notification triggers (email/SMS/WhatsApp via API) for exceptions. Field tests demonstrate >99% event delivery reliability on typical campus Wi-Fi and reduced manual administrative effort. The design emphasizes affordability, ease of deployment, and privacy protection through encrypted communication and access controls. This solution is suitable for colleges, training centers, and corporate training environments seeking to replace manual attendance with an automated, web-enabled system that improves accuracy and administrative efficiency.

CHAPTER - I
INTRUDOCTION TO INTERNET OF THINGS

1.1 Predecessors of IoT

Before delving into the details of the Internet of Things (IoT), a discussion on the base technologies, which make up the crux of IoT, is required. A majority of these technologies, before the IoT era, were used separately for sensing, decision making, and automation tasks. The range of application domains of these technologies extended from regular domains like healthcare, agriculture, home monitoring, and others to specialized domains such as military and mining. Some of these precursor technologies still being used and often re-engineered for IoT are wireless sensor networks (WSN), machine-to-machine (M2M) communications, and cyber physical systems (CPS). All of these precursor paradigms have their distinct signatures and application scopes. A basic overview of these precursor technologies is covered in the subsequent sections in this chapter.

Wireless Sensor Networks

Wireless sensor networks (WSN), as the name suggests, is a networking paradigm that makes use of spatially distributed sensors for gathering information concerning the immediate environment of the sensors and collecting the information centrally. Here, the sensors are not standalone devices but a combination of sensors, processors, and radio units—referred to as sensor nodes—sensing the environment and communicating the sensed data wirelessly to a remote location, which may or may not be connected to a backbone network. Figure 3.1 shows the block diagram of the various standard components of a typical WSN node [4]. The exact specifications of each of these blocks vary depending on the implementation requirements and the network architect's choice.

Figure 3.2 shows a typical WSN implementation, where the master node aggregates data from multiple slave nodes, forwards it to a remote server utilizing access to the Internet through cellular connectivity. The stored data on the server can be visualized by a user or a subscriber to the system from anywhere in the world over the Internet. WSNs mainly follow a system of communication known as master–slave architecture. In a master–slave architecture, a single aggregator node, the master, is responsible for collecting data from various sensor nodes under its dominion or range of operations. The sensor nodes under the range of the master node are referred to as slave nodes. Multiple slave nodes communicate to the master node using lowpower short-range wireless radios such as Zigbee, Bluetooth, and WiFi for transferring

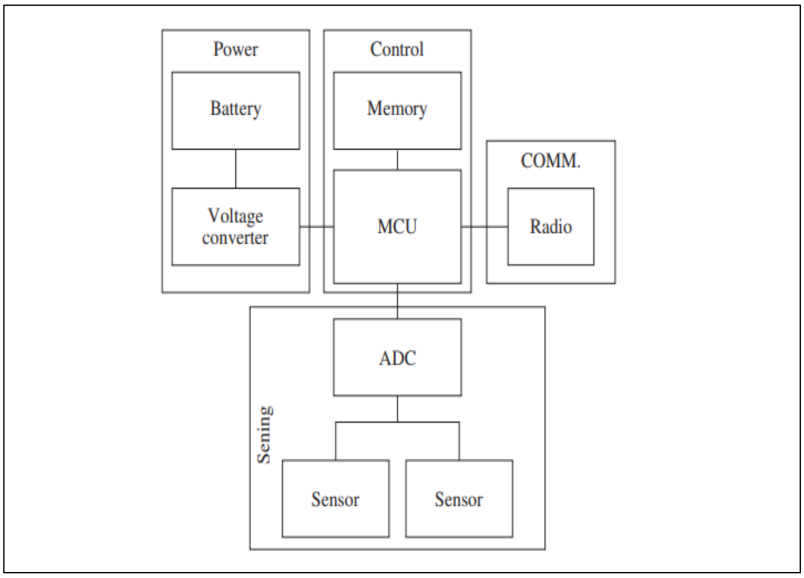

Figure 1.1 The typical constituents of a WSN node

Figure 3.2 shows a typical WSN implementation, where the master node aggregates data from multiple slave nodes, forwards it to a remote server utilizing access to the Internet through cellular connectivity. The stored data on the server can be visualized by a user or a subscriber to the system from anywhere in the world over the Internet. WSNs mainly follow a system of communication known as master–slave architecture. In a master–slave architecture, a single aggregator node, the master, is responsible for collecting data from various sensor nodes under its dominion or range of operations. The sensor nodes under the range of the master node are referred to as slave nodes. Multiple slave nodes communicate to the master node using lowpower short-range wireless radios such as Zigbee, Bluetooth, and WiFi for transferring their sensed data to a remote central server. Often, in popular WSN architectures, the master node connects the WSN to the Internet and acts as the gateway for the WSN. Upon collecting data from the slave nodes, the master node pushes the aggregated data to a remotely located central server using the Internet. The master node may be linked to the Internet through cellular connections, another gateway, or directly through a backbone infrastructure. WSNs must have the following distinguishing features:

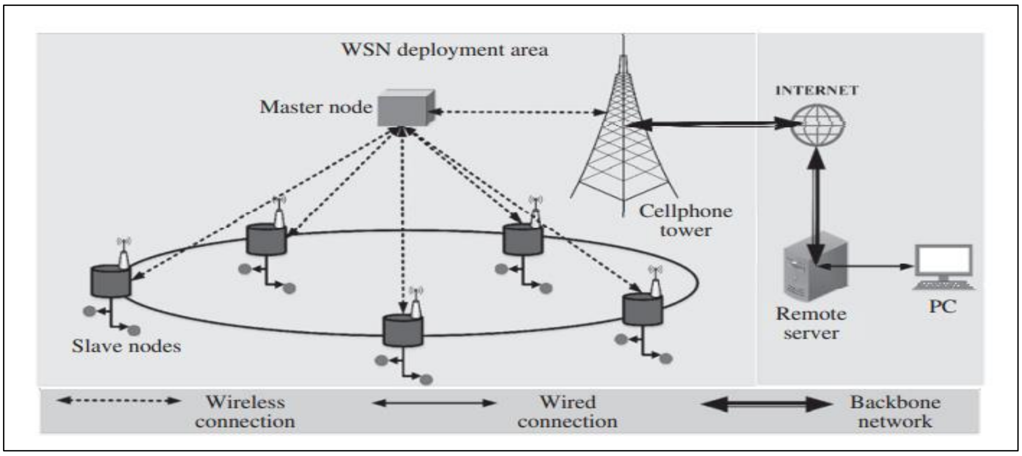

Figure 1.2 A typical WSN deployment

(i) Fault Tolerance:

The occurrence of faults in WSN nodes should not take down the whole WSN implementation, or hamper the transmission of data from nonfaulty nodes to the central location.

(ii) Scalability:

WSN implementations must have the feature of scalability associated with their architectures and deployments. In the event of a future increase or decrease of sensor node units, the WSN must support the scaling of the infrastructure without changing the whole implementation.

(iii) Long lifetime:

The lifetime or the energy replenishment cycle of WSNs must be long enough to make large-scale applications feasible. WSNs have been used for monitoring remote, harsh, and hard to access environments; in these environments, it is not feasible to regularly replenish the energy source of the WSN nodes, which necessitates the need for long node lifetimes.

(iv) Security:

The security of WSNs, if not considered, can easily compromise the security of the whole system, right back to the central server. As WSNs are used for a wide range of applications, some of which are crucial, security is one aspect

(v) Programmability:

The programmability of WSNs is important as it ensures the robustness of these systems. WSNs deployed in one application area can be reused for other applications just as easily with a change in sensors and the backend programs associated with it. Programmability also helps in providing a means of adjusting the parameters of the system in the event of a scale-up or scale-down operation.

(vi) Affordability:

As WSNs generally require multiple units, typically in the range of tens or hundreds of WSN nodes, the cost of the nodes and its affordability is vastly responsible for the acceptability of the system. Except for some specialized domains such as the military and the industry, where the sensing requirements are quite high and that too in harsh and challenging conditions, the majority of WSN applications are regular. To some extent, the cost of WSN deployments in these regular domains decide the acceptability of a WSN-based solution for that domain.

(vii) Heterogeneity:

The WSNs must support a wide number and various types of sensors and solutions, thus enabling heterogeneity. In the absence of heterogeneity, the WSN will tend to become very application-specific, which in turn would require major customizations even in the event of minor changes to the network or architecture.

(viii) Mobility:

WSNs must support the notion of mobility of nodes such that the nodes may be easily relocatable or mobile. Mobility would ensure the rapid deployability of WSN-based solutions in all environments types. WSNs have found numerous applications in domains such as agriculture, healthcare, military, industries, mining, and others. The main reason for the popularity of WSNs is attributed to the advantages they provide in the form of enhanced monitoring times, easy installation, and multiple implementations. Implementations on a large scale are possible due to high affordability, ease of replacement or upgradation, ease of modifying system parameters, ease of additional sensor integration with the sensor nodes, and other such factors.

1.1.1. Architectural components of WSN

WSNs that are similar to regular computer network paradigms may be explained in terms of a protocol stack, which is very similar to the ISO/OSI (Open Systems Interconnection developed by the International Standards Organization) stack. However, instead of seven layers similar to the OSI model, the WSN stack is made up of five layers (Figure 3.3):

1) Physical
2) data link
3) network
4) transport
5) application layer.

In addition to these five layers, the WSN stack further comprises five cross planes concerned with management tasks such as

1) power management plane,
2) mobility management plane
3) task management plane
4) QoS management plane
5) security management plane.

This section is divided into three parts:

1) Components of the WSN stack
2) cross-layer management planes,
3) WSN types.

We start with the physical layer, which is at the bottom of the stack and responsible for enabling transmission of signals over a physical medium between multiple WSN nodes/units. In regular computer networks, the medium can be both wired as well as wireless; however, in WSNs, the medium is strictly wireless. In WSNs, this layer is responsible for carrier frequency generation, carrier frequency selection, modulation/ demodulation, encryption/decryption, and signal detection. Typically, WSNs make use of the IEEE 802.15.4 standard for this layer because of its low cost, low energy budget, low data rate, and small form factor

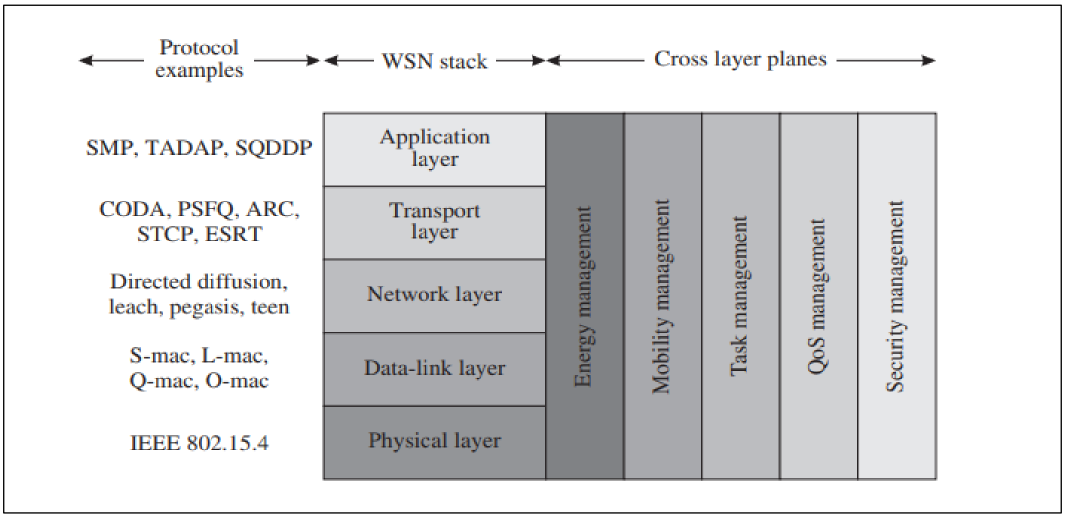

Figure 1.3 The various functional layers for a WSN communication and networking architecture

1.1.2. Cross-layer Management Planes

The use of OSI-like stacks for outlining the functionalities of WSNs faces limitations due to specialized operations of the stacks in areas requiring prolonged deployments with constrained energy and communication infrastructure, and mobility. Unlike regular computer networks, the OSI-like WSN stack does not fully describe the functionalities of WSN-based systems. This is because its specialized nature results in a strong correlation between the five WSN stack layers. Typically, solutions addressing WSN applications and functionalities make joint use of all the five layers. It is mainly because of this reason that the cross-layer management plane structure is more popularly accepted as a means of abstraction of WSN-based systems and solutions. Table 3.1 shows the relative positioning of the WSN crosslayer management planes, in terms of the functionalities they provide

1.1.3.Machine-to-Machine Communications

The machine-to-machine (M2M) paradigm, as the name suggests, implies a system of communication between two or more machines/devices without human intervention. Some basic examples of M2M communication in our daily lives include ATM machines signaling banks about the need to refill them with cash, power line monitoring systems in a house alerting a generator set of possible power failures and switching to generator-based supply, vending machines updating stock of items in their inventory and alerting a remote inventory of the need to refill certain depleting items, and others [5]. Figure 3.5 outlines the significant aspects of an M2M ecosystem. The task of any sensor network system (be it WSN, M2M, CPS, or IoT) is the sensing of a physical environment and converting the acquired data into a tangible output in the form of numbers using sensors. This sensing is followed by the transfer of sensed data to a remote device or location using a network, which may be wired or wireless. The data collected at the remote device from various sensors—homogeneous or heterogeneous—is converted to usable information, which can be utilized to define the course of actions for individual scenarios. This information is processed to decide upon the most valid and optimum course of action that must be undertaken to control the sensed environment desirably or as per requirements. Finally, actuators are put to work to modify or adjust the sensed environment. A centralized air conditioning system in a building is a good example of this paradigm. If the requirements of a particular room in that building dictate that the room must be kept at a constant temperature of 20◦ C, the sensors continuously sense the environment (temperature of the room) and direct the actuators controlling the cold air inlet to keep the room at the desired temperature. The massive and rapid developments in the field of wireless communication have significantly helped in the widespread deployment of M2M solutions world over. The 3rd Generation Partnership Project (3GPP) [6], which is responsible for unifying and benchmarking telecommunication standards across seven different

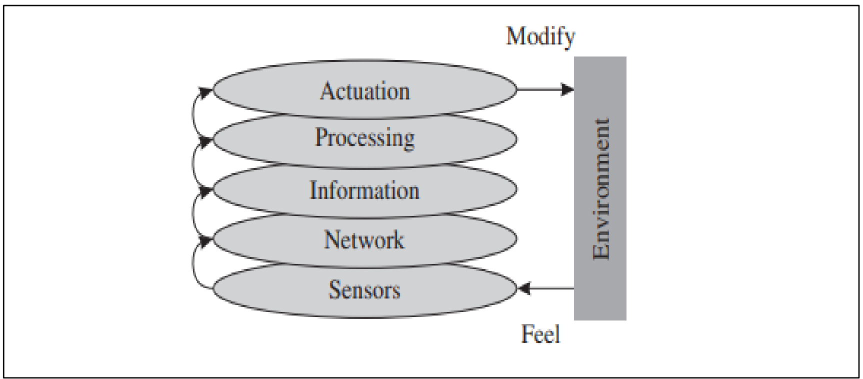

Figure 1.4 An overview of the M2M ecosystem

Introduction The modern-day advent of network-connected devices has given rise to the popular paradigm of the Internet of Things (IoT). Each second, the present-day Internet allows massively heterogeneous traffic through it. This network traffic consists of images, videos, music, speech, text, numbers, binary codes, machine status, banking messages, data from sensors and actuators, healthcare data, data from vehicles, home automation system status and control messages, military communications, and many more. This huge variety of data is generated from a massive number of connected devices, which may be directly connected to the Internet or connected through gateway devices. According to statistics from the Information Handling Services [7], the total number of connected devices globally is estimated to be around 25 billion. This figure is projected

One of the best examples of this explosion is the evolution of smartphones. In the late 1990's, cellular technology was still expensive and which could be afforded only by a select few. Moreover, these particular devices had only the basic features of voice calling, text messaging, and sharing of low-quality multimedia. Within the next 10 years, cellular technology had become common and easily affordable. With time, the features of these devices evolved, and the dependence of various applications and services on these gadgets on packet-based Internet accesses started rapidly increasing. The present-day mobile phones (commonly referred to as smartphones) are more or less Internet-based. The range of applications on these gadgets such as messaging, video calling, e-mails, games, music streaming, video streaming, and others are solely dependent on network provider allocated Internet access or WiFi. Most ofthe present-day consumers of smartphone technology tend to carry more than one of these units. In line with this trend, other connected devices have rapidly increased in numbers resulting in the number of devices exceeding the number of humans on Earth by multiple times. Now imagine that as all technologies and domains are moving toward smart management of systems, the number of sensor/actuator-based systems is rapidly increasing. With time, the need for location-independent access to monitored and controlled systems keep on rising. This rise in number leads to a further rise in the number of Internet-connected devices. The original Internet intended for sending simple messages is now connected with all sorts of "Things". These things can be legacy devices, modern-day computers, sensors, actuators, household appliances, toys, clothes, shoes, vehicles, cameras, and anything which may benefit a product by increasing its scientific value, accuracy, or even its cosmetic value IoT is an anytime, anywhere, and anything (as shown in Figure 4.2) network of Internet-connected physical devices or systems capable of sensing an environment and affecting the sensed environment intelligently. This is generally achieved using low-power and low-form-factor embedded processors on-board the "things" connected to the Internet. In other words, IoT may be considered to be made up of connecting devices, machines, and tools; these things are made up of sensors/actuators and processors, which connect to the Internet through wireless technologies. Another school of thought also considers wired Internet access to be inherent to the IoT paradigm.

For the sake of harmony, in this book, we will consider any technology enabling access to the Internet—be it wired or wireless—to be an IoT enabling technology. However, most of the focus on the discussion of various IoT enablers will be restricted to wireless IoT systems due to the much more severe operating constraints and challenges faced by wireless devices as compared to wired systems. Typically, IoT systems can be characterized by the following features [2]: • Associated architectures, which are also efficient and scalable.

• No ambiguity in naming and addressing.

• Massive number of constrained devices, sleeping nodes, mobile devices, and non-IP devices.
• Intermittent and often unstable connectivity

IoT is speculated to have achieved faster and higher technology acceptance as compared to electricity and telephony. These speculations are not ill placed as evident from the various statistics shown in Figures 4.3, 4.4, and 4.5.

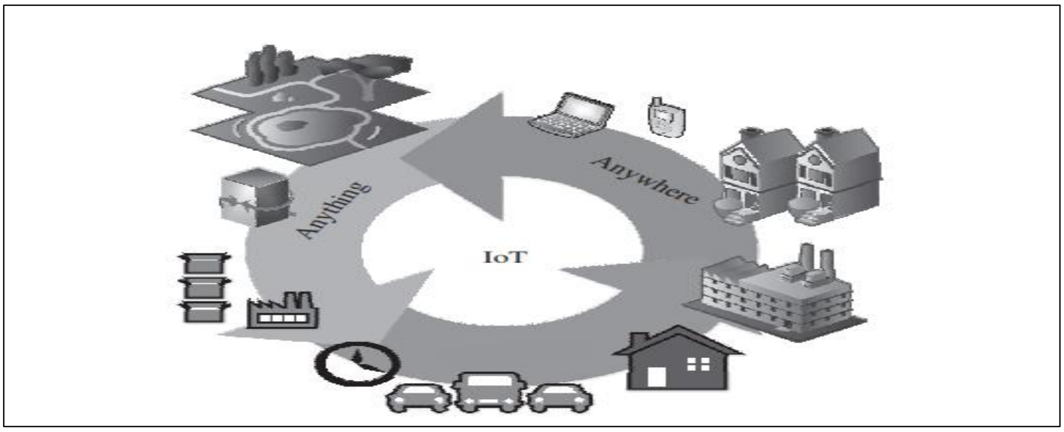

Figure 1.5

The three characteristic features—anytime, anywhere, and anything—highlight the robustness and dynamic nature of IoT

Evolution of IoT The IoT, as we see it today, is a result of a series of technological paradigm shifts over a few decades. The technologies that laid the foundation of connected systems by achieving easy integration to daily lives, popular public acceptance, and massive benefits by using connected solutions can be considered as the founding solutions for the development of IoT. Figure 4.6 shows the sequence of technological advancements for shaping the IoT as it is today. These sequence of technical developments toward the emergence of IoT are described in brief:

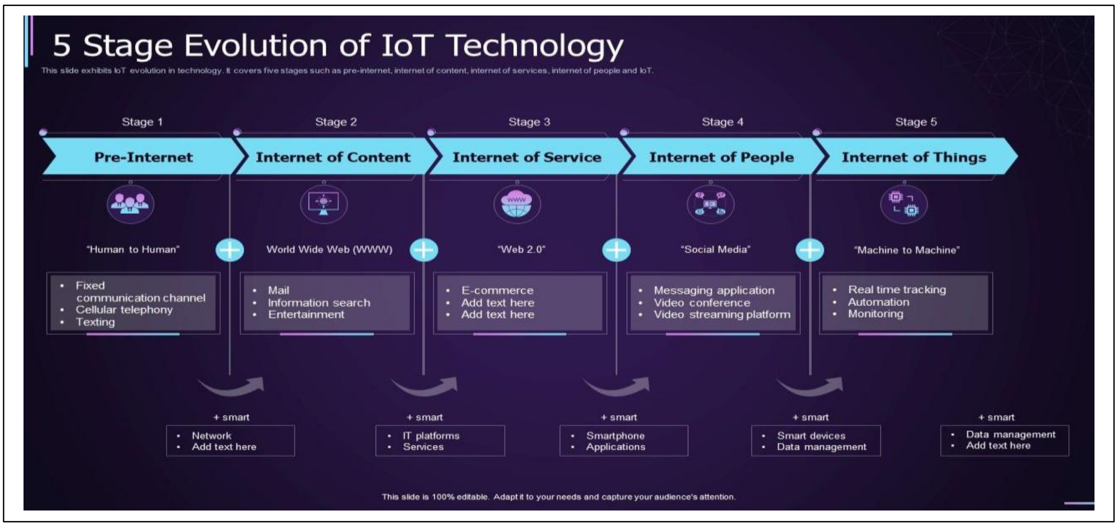

Figure 1.6

The sequence of technological developments leading to the shaping of the modernday IoT

The present-day IoT spans across various domains and applications. The major highlight of this paradigm is its ability to function as a cross-domain technology enabler. Multiple domains can be supported and operated upon simultaneously over IoT-based platforms. Support for legacy technologies and standalone paradigms, along with modern developments, makes IoT quite robust and economical for commercial, industrial, as well as consumer applications. IoT is being used in vivid and diverse areas such as smart parking, smartphone detection, traffic congestion, smart lighting, waste management, smart roads, structural health, urban noise maps, river floods, water flow, silos stock calculation, water leakages, radiation levels, explosive and hazardous gases, perimeter access control, snow level monitoring, liquid presence, forest fire detection, air pollution, smart grid, tank level, photovoltaic installations, NFC (near-field communications) payments, intelligent shopping applications, landslide and avalanche prevention, early detection of earthquakes, supply chain control, smart product management, and others. Figure 4.7 shows the various technological interdependencies of IoT with other domains and networking paradigms such as M2M, CPS, the Internet of environment (IoE), the Internet of people (IoP), and Industry 4.0. Each of these networking paradigms is a massive domain on its own, but the omnipresent nature of IoT implies that these domains act as subsets of IoT. The paradigms are briefly discussed here

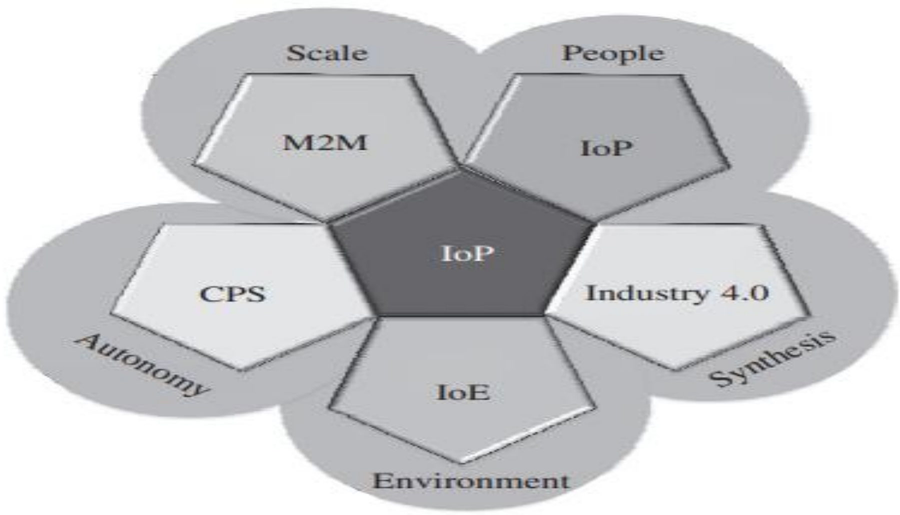

Figure 1.7

The interdependence and reach of IoT over various application domains and networking paradigms

1.1.3. IoT versus M2M

M2M or the machine-to-machine paradigm refers to communications and interactions between various machines and devices. These interactions can be enabled through a cloud computing infrastructure, a server, or simply a local network hub. M2M collects data from machinery and sensors, while also enabling device management and device interaction. Telecommunication services providers introduced the term M2M, and technically emphasized on machine interactions via one or more communication networks (e.g., 3G, 4G, 5G, satellite, public networks). M2M is part of the IoT and is considered as one of its sub-domains, as shown in Figure 4.7. M2M standards occupy a core place in the IoT landscape. However, in terms of operational and functional scope, IoT is vaster than M2M and comprises a broader range of interactions such as the interactions between devices/things, things, and people, things and applications, and people with applications; M2M enables the amalgamation of workflows comprising such interactions within IoT. Internet connectivity is central to the IoT theme but is not necessarily focused on the use of telecom networks.

1.1.4. IoT versus CPS

Cyber physical systems (CPS) encompasses sensing, control, actuation, and feedback as a complete package. In other words, a digital twin is attached to a CPS-based system. As mentioned earlier, a digital twin is a virtual system–model relation, in which the system signifies a physical system or equipment or a piece of machinery, while the model represents the mathematical model or representation of the physical system's behavior or operation. Many a time, a digital twin is used parallel to a physical system, especially in CPS as it allows for the comparison of the physical system's output, performance, and health. Based on feedback from the digital twin, a physical system can be easily given corrective directions/commands to obtain desirable outputs. In contrast, the IoT paradigm does not compulsorily need feedback or a digital twin system. IoT is more focused on networking than controls. Some of the constituent sub-systems in an IoT environment (such as those formed by CPS-based instruments and networks) may include feedback and controls too. In this light, CPS may be considered as one of the sub-domains of IoT, as shown in Figure 4.7.

1.1.5. IoT versus WoT

From a developer's perspective, the Web of Things (WoT) paradigm enables access and control over IoT resources and applications. These resources and applications are generally built using technologies such as HTML 5.0, JavaScript, Ajax, PHP, and others. REST (representational state transfer) is one of the key enablers of WoT. The use of RESTful principles and RESTful APIs (application program interface) enables both developers and deployers to benefit from the recognition, acceptance, and maturity of existing web technologies without having to redesign and redeploy solutions from scratch. Still, designing and building the WoT paradigm has various adaptability and security challenges, especially when trying to build a globally uniform WoT. As IoT is focused on creating networks comprising objects, things, people, systems, and applications, which often do not consider the unification aspect and the limitations of the Internet, the need for WoT, which aims to integrate the various focus areas of IoT into the existing Web is really invaluable. Technically, WoT can be thought of as an application layer-based hat added over the network layer. However, the scope of IoT applications is much broader; IoT also which includes non-IP-based systems that are not accessible through the web.

1.2. Enabling IoT and the Complex Interdependence of Technologies

IoT is a paradigm built upon complex interdependencies of technologies (both legacy and modern), which occur at various planes of this paradigm. Regarding Figure 4.8, we can divide the IoT paradigm into four planes: services, local connectivity, global connectivity, and processing. If we consider a bottom-up view, the services offered fall under the control and purview of service providers. The service plane is composed of two parts:

1) things or devices

2) low-power connectivity.

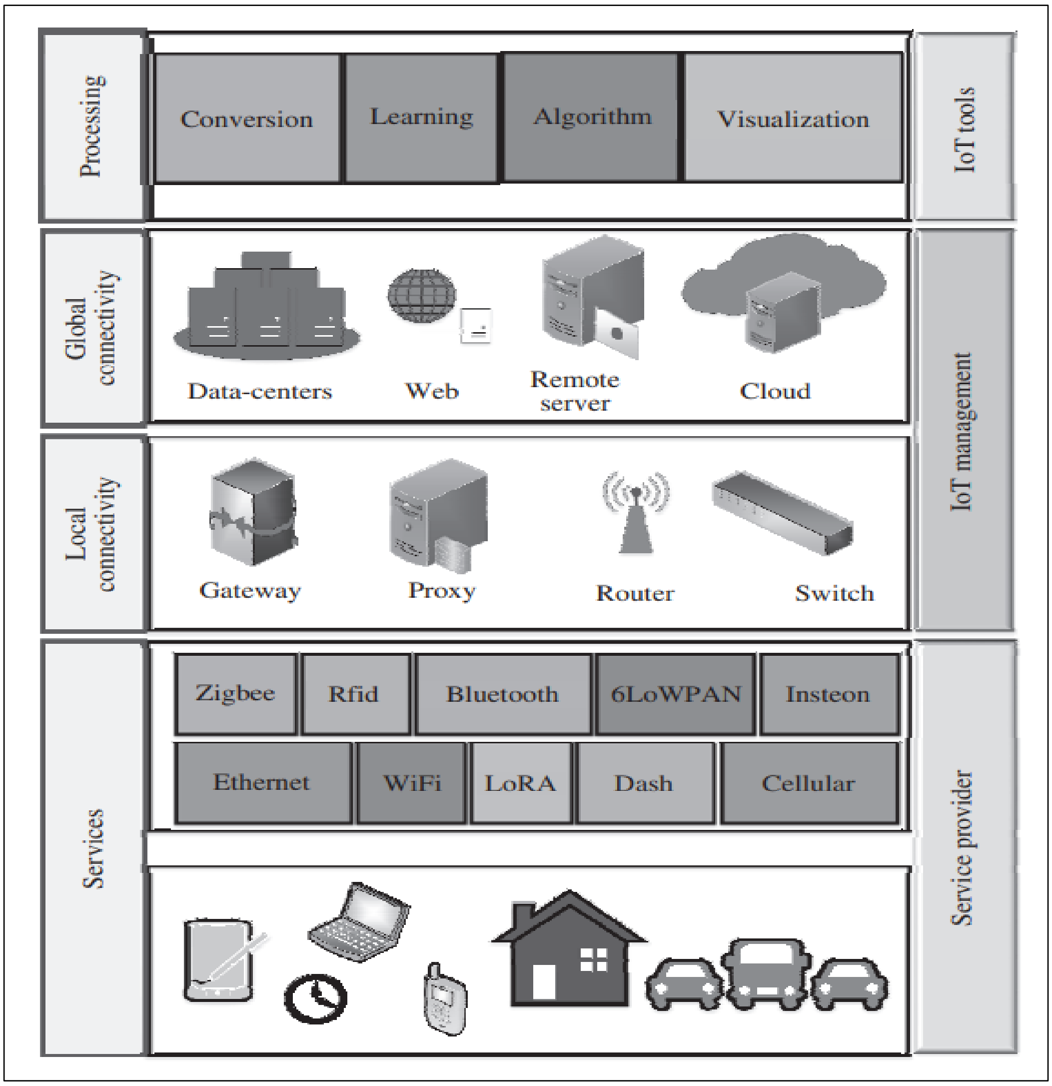

Figure 1.8 The IoT planes, various enablers of IoT, and the complex interdependencies among them

IoT Networking Components An IoT implementation is composed of several components, which may vary with their application domains. Various established works such as that by Savolainen et al. [2] generally outline five broad categories of IoT networking components. However, we outline the broad components that come into play during the establishment of any IoT network, into six types: 1) IoT node, 2) IoT router, 3) IoT LAN, 4) IoT WAN, 5) IoT gateway, and 6) IoT proxy. A typical IoT implementation from a networking perspective is shown in Figure 4.9. The individual components are briefly described here:

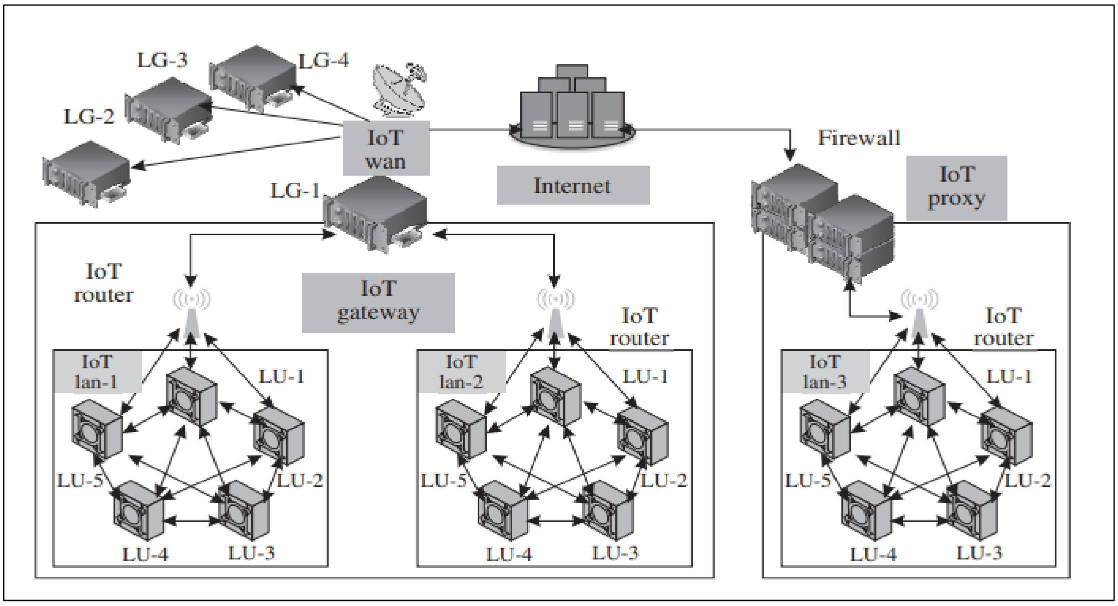

Figure 1.9

A typical IoT network ecosystem highlighting

the various networking components— from IoT nodes to the Internet

(i) IoT Node:

These are the networking devices within an IoT LAN. Each of these devices is typically made up of a sensor, a processor, and a radio, which communicates with the network infrastructure (either within the LAN or outside it). The nodes may be connected to other nodes inside a LAN directly or bymeans of a common gateway for that LAN. Connections outside the LAN are through gateways and proxies.

(ii) IoT Router:

An I oT router is a piece of networking equipment that is primarily tasked with the routing of packets between various entities in the IoT network; it keeps the traffic flowing correctly within the network. A router can be repurposed as a gateway by enhancing its functionalities.

(iii) IoT LAN:

The local area network (LAN) enables local connectivity within the purview of a single gateway. Typically, they consist of short-range connectivity technologies. IoT LANs may or may not be connected to the Internet. Generally, they are localized within a building or an organization.

(iv) IoT WAN:

The wide area network (WAN) connects various network segments such as LANs. They are typically organizationally and geographically wide, with their operational range lying between a few kilometers to hundreds of kilometers. IoT WANs connect to the Internet and enable Internet access to the segments they are connecting.

(v) IoT Gateway:

An IoT gateway is simply a router connecting the IoT LAN to a WAN or the Internet. Gateways can implement several LANs and WANs. Their primary task is to forward packets between LANs and WANs, and the IP layer using only layer 3. (vi) IoT Proxy: Proxies actively lie on the application layer and performs application layer functions between IoT nodes and other entities. Typically, application layer proxies are a means of providing security to the network entities under it ; it helps to extend the addressing range of its network.

1.2.1 IoT Sensing and Actuation

A major chunk of IoT applications involves sensing in one form or the other. Almost all the applications in IoT—be it a consumer IoT, an industrial IoT, or just plain hobby-based deployments of IoT solutions—sensing forms the first step. Incidentally, actuation forms the final step in the whole operation of IoT application deployment in a majority of scenarios. The basic science of sensing and actuation is based on the process of transduction. Transduction is the process of energy conversion from one form to another. A transducer is a physical means of enabling transduction. Transducers take energy in any form (for which it is designed)—electrical, mechanical, chemical, light, sound, and others—and convert it into another, which may be electrical, mechanical, chemical, light, sound, and others. Sensors and actuators are deemed as transducers. For example, in a public announcement (PA) system, a microphone (input device) converts sound waves into electrical signals, which is amplified by an amplifier system (a process). Finally, a loudspeaker (output device) outputs this into audible sounds by converting the amplified electrical signals.

1.2.2. Sensors

Sensors are devices that can measure, or quantify, or respond to the ambient changes in their environment or within the intended zone of their deployment. They generate responses to external stimuli or physical phenomenon through characterization of the input functions (which are these external stimuli) and their conversion into typically electrical signals. For example, heat is converted to electrical signals in a temperature sensor, or atmospheric pressure is converted to electrical signals in a barometer. A sensor is only sensitive to the measured property (e.g., a temperature sensor only senses the ambient temperature of a room). It is insensitive to any other property besides what it is designed to detect (e.g., a temperature sensor does not bother about light or pressure while sensing the temperature). Finally, a sensor does not influence the measured property (e.g., measuring the temperature does not reduce or increase the temperature). Figure 5.1 shows the simple outline of a sensing task. Here, a temperature sensor keeps on checking an environment for changes. In the event of a fire, the temperature of the environment goes up. The temperature sensor notices this change in the temperature of the room and promptly communicates this information to a remote monitor via the processor.

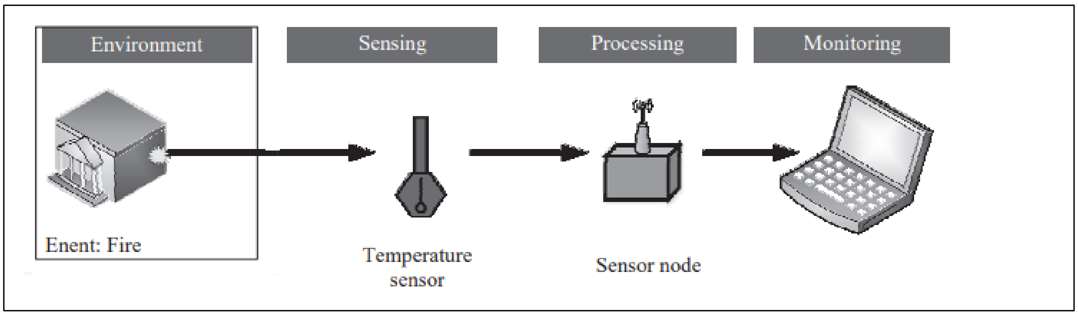

Figure 1.9 The outline of a simple sensing operation

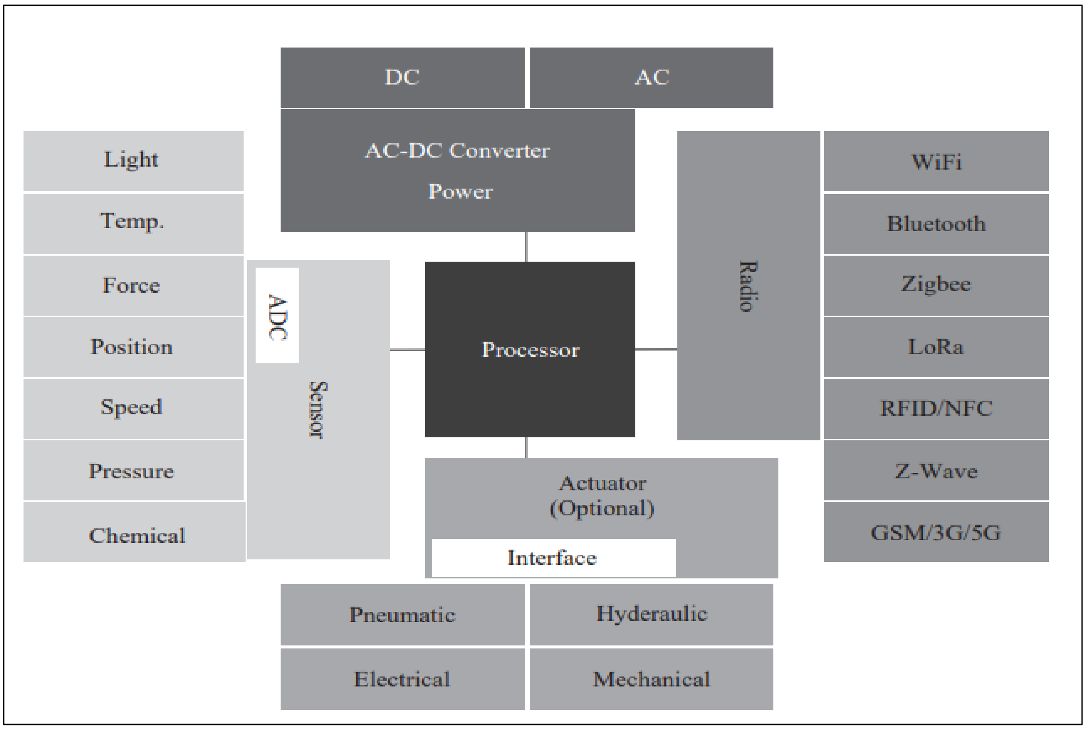

Figure 1.10

The functional blocks of a typical sensor node in IoT

1.3. Baseline Technologies of IoT

According to Jones, the top 10 emerging IoT technologies are:

1. IoT Security:

Security technologies will be required to protect IoT devices and platforms from both information attacks and physical tampering, to encrypt their communications, and to address new challenges such as impersonating "things" or denial-of-sleep attacks that drain batteries. IoT security will be complicated by the fact that many "things" use simple processors and operating systems that may not support sophisticated security approaches.

2. IoT Analytics:

IoT business models will exploit the information collected by "things" in many ways, which will demand new analytic tools and algorithms. As data volumes increase over the next five years, the needs of the IoT may diverge further from traditional analytics.

3. IoT Device (Thing) Management:

Long-lived nontrivial "things" will require management and monitoring, including device monitoring, firmware and software updates, diagnostics, crash analysis and reporting, physical management, and security management. Tools must be capable of managing and monitoring thousands and perhaps even millions of devices.

4. Low-Power, Short-Range IoT Networks.

Low-power, short-range networks will dominate wireless IoT connectivity through 2025, far outnumbering connections using widearea IoT networks. However, commercial and technical trade-offs mean that many solutions will coexist, with no single dominant winner.

5. Low-Power, Wide-Area Networks.

Traditional cellular networks don't deliver a good combination of technical features and operational cost for those IoT applications that need wide-area coverage combined with relatively low bandwidth, good battery life, low hardware and operating cost, and high connection density. Emerging standards such as narrowband IoT will likely dominate this space.

6. IoT Processors.

The processors and architectures used by IoT devices define many of their capabilities, such as whether they are capable of strong security and encryption, power consumption, whether they are sophisticated enough to support an operating system, updatable firmware, and embedded device management agents. Understanding the implications of processor choices will demand deep technical skills.

7. IoT Operating Systems.

Traditional operating systems such as Windows and iOS were not designed for IoT applications. They consume too much power, need fast processors, and in some cases, lack features such as guaranteed real-time response. They also have too large a memory footprint for small devices and may not support the chips that IoT developers use. Consequently, a wide range of IoT-specific operating systems has been developed to suit many different hardware footprints and feature needs.

8. Event Stream Processing:

Some IoT applications will generate extremely high data rates that must be analyzed in real time. Systems creating tens of thousands of events per second are common, and millions of events per second can occur in some situations. To address such requirements, distributed stream computing platforms have emerged that can process very high-rate data streams and perform tasks such as real-time analytics and pattern identification.

9. IoT Platforms.

IoT platforms bundle many of the infrastructure components of an IoT system into a single product. The services provided by such platforms fall into three main categories: Low-level device control and operations such as communications, device monitoring and management, security, and firmware updates; IoT data acquisition, transformation and management; IoT application development, including event-driven logic, application programming, visualization, analytics and adapters to connect to enterprise systems.

10.IoT Standards and Ecosystems.

Standards and their associated application programming interfaces (APIs) will be essential because IoT devices will need to interoperate and communicate, and many IoT business models will rely on sharing data between multiple devices and organizations. Many IoT ecosystems will emerge, and organizations creating products may have to develop variants to support multiple standards or ecosystems and be prepared to update products during their life span as the standards evolve and new standards and APIs emerge.

1.4. Classification of Sensors

The common IoT sensors include:

Temperature sensors, Pressure sensors, Motion sensors, Level sensors, Image sensors, Proximity sensors, Water quality sensors, Chemical sensors, Gas sensors, Smoke sensors, Infrared (IR) sensors, Humidity sensors, etc.

A description of each of these sensors is provided below.

Temperature sensors

Temperature sensors detect the temperature of the air or a physical object and concert that temperature level into an electrical signal that can be calibrated accurately reflect the measured temperature. These sensors could monitor the temperature of the soil to help with agricultural output or the temperature of a bearing operating in a critical piece of equipment to sense when it might be overheating or nearing the point of failure.

Pressure sensors

Pressure sensors measure the pressure or force per unit area applied to the sensor and can detect things such as atmospheric pressure, the pressure of a stored gas or liquid in a sealed system such as tank or pressure vessel, or the weight of an object.

Motion sensors

Motion sensors or detectors can sense the movement of a physical object by using any one of several technologies, including passive infrared (PIR), microwave detection, or ultrasonic, which uses sound to detect objects. These sensors can be used in security and intrusion detection systems, but can also be used to automate the control of doors, sinks, air conditioning and heating, or other systems.

Level sensors

Level sensors translate the level of a liquid relative to a benchmark normal value into a signal. Fuel gauges display the level of fuel in a vehicle's tank, as an example, which provides a continuous level reading. There are also point level sensors, which are a go-no/go or digital representation of the level of the liquid. Some automobiles have a light that illuminates when the fuel level tank is very close to empty, acting as an alarm that warns the driver that fuel is about to run out completely.

Image sensors

Image sensors function to capture images to be digitally stored for processing. License plate readers are an example, as well as facial recognition systems. Automated production lines can use image sensors to detect issues with quality such as how well a surface is painted after leaving the spray booth.

Proximity sensors

Proximity sensors can detect the presence or absence of objects that approach the sensor through a variety of different technology designs.

Water quality sensors

The importance of water to human beings on earth not only for drinking but as a key ingredient needed in many production processes dictates the need to be able to sense and measure parameters around water quality. Some examples of what is sensed and monitored include.

Chemical presence

(such as chlorine levels or fluoride levels),Oxygen levels (which may impact the growth of algae and bacteria),Electrical conductivity (which can indicate the level of ions present in water), pH level (a reflection of the relative acidity or alkalinity of the water),Turbidity levels (a measurement of the amount of suspended solids in water) Chemical sensors Chemical sensors are designed to detect the presence of specific chemical substances which may have inadvertently leaked from their containers into spaces that are occupied by personnel and are useful in controlling industrial process conditions.

Gas sensors

Related to chemical sensors, gas sensors are tuned to detect the presence of combustible, toxic, or flammable gas in the vicinity of the sensor. Examples of specific gases that can be detected include: Bromine (Br_2), Carbon Monoxide (CO), Chlorine (Cl_2), Chlorine Dioxide (ClO_2),Hydrogen Cyanide (HCN),Hydrogen Peroxide (H_2O_2), Hydrogen Sulfide (H_2S), Nitric Oxide (NO), Nitrogen Dioxide (NO_2), Ozone (O_3), etc.

Smoke sensors

Smoke sensors or detectors pick up the presence of smoke conditions which could be an indication of a fire typically using optical sensors (photoelectric detection) or ionization detection.

Infrared (IR) sensors

detect infrared radiation that is emitted by objects. Non-contact thermometers make use of these types of sensors as a way of measuring the temperature of an object without having to directly place a probe or sensor on that object. They find use in analyzing the heat signature of electronics and detecting blood flow or blood pressure in patients.

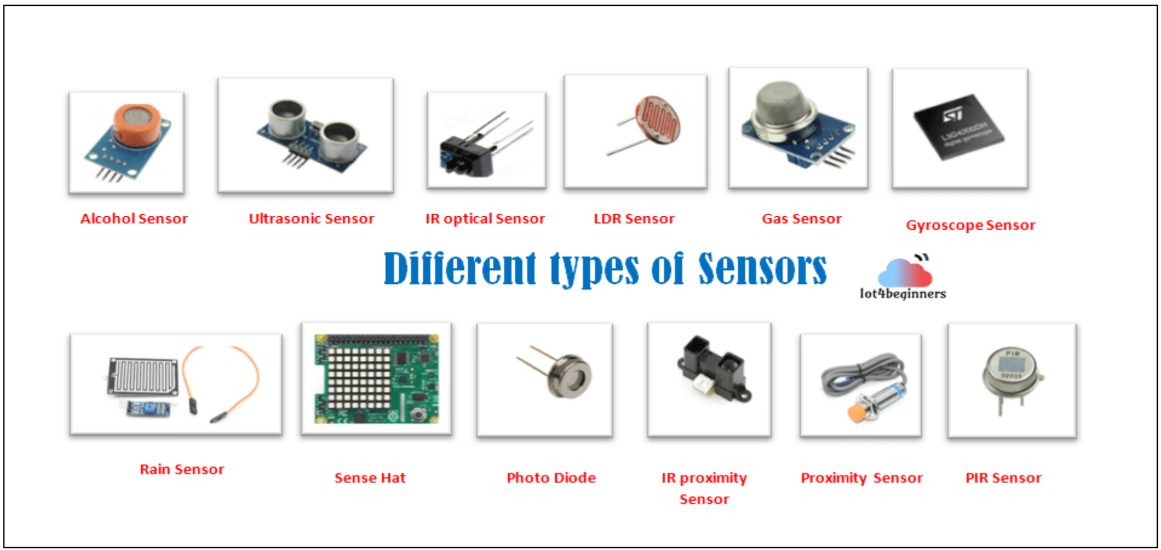

Fig.No 1.11

Sensors in the Internet of Things (IoT) Devices

CHAPTER – II

LITERATURE SURVEY

2.1.1 INTRODUCTION:

The Internet of Things (IoT) refers to the network of interconnected devices embedded with sensors, software, and other technologies that allow them to collect, exchange, and act on data over the Internet without human intervention. IoT enables smart decision-making and automation in daily life, industries, and cities.

IoT is a convergence of several technologies, including wireless communication, sensor networks, cloud computing, artificial intelligence, and big data analytics. It transforms traditional systems into intelligent ecosystems that optimize efficiency, safety, and productivity.

2.1.2. IoT-Based Healthcare-Monitoring System:

The term Internet of Things (IoT) was invented by Kevin Ashton in 1999 and refers to data on the Internet that are connected to evolving global service architecture [1,2]. IoT is the product of advanced research on information and communications technology. It can potentially enhance urban residents' quality of life. Since the global population is increasing at an astonishing rate, and the prevalence of chronic diseases is also on the rise, there is growing demand for designing cost-effective healthcare systems that can efficiently manage and provide a wide range of medical services while reducing overall expenses [3,4,5,6]. The IoT has become a key development area recently, enabling healthcare-monitoring system advancement. The IoT healthcare-monitoring system aims to accurately track people and connect various services and things in the world through the Internet to collect, share, monitor, store, and analyze the data generated by these things [7]. However, the IoT is a new paradigm where all connected physical objects in any intelligent application, such as smart city, smart home, and smart healthcare, are addressed and controlled remotely. Diagnosing disorders and monitoring patients is essential to providing medical care, and applying sensor networks to the human body will significantly assist in this endeavor. In addition, the information is readily accessible from any location in the world at any given time [8].

Patients with severe injuries or from certain areas may have difficulty reaching the hospital. Therefore, they can use video conferencing to communicate with their doctors to improve their health while saving money and time. Patients can use this technology to record their health conditions on their phones [9]. It is anticipated that the benefits of the IoT will be improved and result in individualized treatment, improving patient outcomes while saving healthcare management costs. IoT systems allow physicians to keep an eye on their patients remotely and schedule their appointments more efficiently. Patients also can improve their home healthcare to reduce their need for doctor visits and the likelihood of receiving unnecessary or inappropriate medical treatments in hospitals or clinics. For this reason, the quality of medical care and the overall safety of patients may improve, while the overall cost of care may decrease. The IoT holds significant potential in healthcare [7,10]. It will not be long before we have access to a health-monitoring system that can be used from the comfort of our homes and streamline hospital processes. IoT sensors should be densely deployed to monitor the body and environment continuously. This effort will enable the tracking of chronic-disease management and rehabilitation progress. In the future of virtual consultations for remote medical care, the IoT will be able to provide efficient data connections from multiple locations [11].

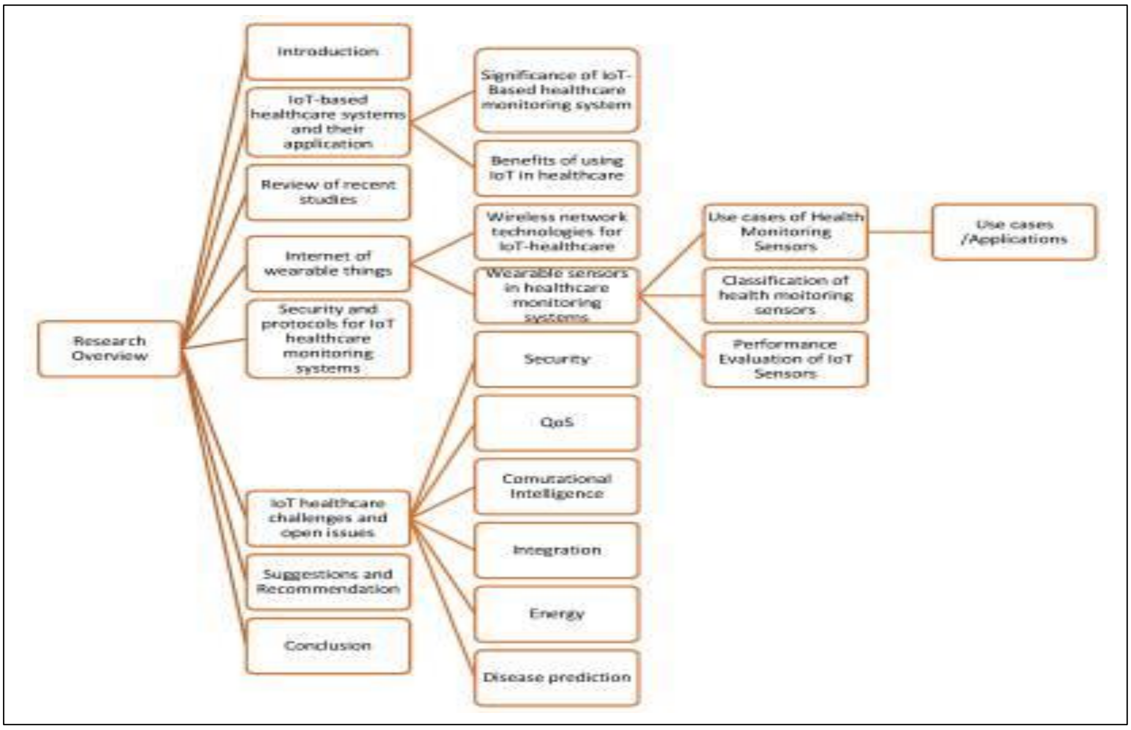

Figure 2.1

The IoT healthcare-monitoring

Most of the current implementations of the IoT and research on it are undeveloped and focus on deploying and configuring technology in various contexts and conditions. However, these practices are not widely used today. Therefore, this paper aims to evaluate related research on designing and implementing an IoT-based healthcare-monitoring system that improves quality of life. These systems rely heavily on IoT devices and sensors to connect patients with the healthcare providers best suited for their care.

2.1.3. Smart Irrigation Systems Using the Internet of Things:

Water management in agriculture has been transformed by smart irrigation systems that use the Internet of Things (IoT) to maximize resource use and boost crop yields. IoT-based irrigation, which reduces waste and improves sustainability in the face of growing water scarcity, combines sensors, actuators and real-time data analytics to deliver accurate water flow. The use of IoT in agriculture is reviewed in this paper, which also covers AI-driven decision-making algorithms and communication technologies like LoRa, Zigbee and Wi-Fi. In agriculture, IoT has been widely employed in agriculture to enable better farming and redefine conventional practices. Some of the necessary applications include monitoring crops, monitoring livestock, automation of the supply of water through real-time data, promotion of optimal water consumption and wastage minimization. Furthermore, increased agricultural yields, cost savings and water conservation are just a few benefits of using IoT in irrigation systems. Research also claimed that when compared to traditional methods, IoT-based irrigation systems can save up to 30% on water usage. IoT-based irrigation has drawbacks despite its benefits, including costly upfront costs, complicated technology and cyber security threats. The use of self-powered sensors, AI-powered predictive irrigation and blockchain-enhanced data security are some of the upcoming trends. Smart irrigation systems have the ability to transform precision agriculture and guarantee both environmental preservation and food security by removing implementation obstacles. Future research should focus on enhancing interoperability among IoT devices, improving decision-making algorithms and developing cost-effective solutions for small-scale farmers. With continuous innovation and widespread adoption, IoT-driven smart irrigation has the potential to address global food security concerns while preserving water resources, creating a more sustainable and technologically advanced agriculture industry.

2.1.4. Smart Traffic System uses IoT for intelligent traffic management:

1. Smart Infrastructure

Barcelona's commitment to smart infrastructure is evident in its extensive deployment of Internet of Things (IoT) devices and sensors throughout the city. These technologies are used to monitor and manage various urban systems, including:

- **Smart Lighting:** Barcelona installed 10,000 smart streetlights adjusting brightness based on traffic, reducing energy use by 30% and light pollution. Additionally, these streetlights are equipped with sensors to monitor air quality and noise levels, providing valuable data for urban planning.
- **Waste Management:** Barcelona's smart bins use sensors to alert when full, optimizing collection routes, leading to a 20% reduction in waste collection costs and significant cuts in carbon emissions from trucks.
- **Water Management:** Smart water meters and leak detection systems help conserve water and ensure efficient usage across the city. These technologies have enabled Barcelona to reduce water loss by 25%, ensuring a more sustainable water supply for its residents.

2. Sustainable Mobility

Barcelona has made significant strides in promoting sustainable mobility solutions to reduce traffic congestion and lower carbon emissions. Key initiatives include:

- **Public Transportation:** Barcelona integrates real-time data into its transit system for accurate bus and train schedules, promoting public transportation use. The introduction of smart bus stops with digital displays has improved user experience and increased public transport usage by 15%.

- **Bike-Sharing Programs:** Barcelona's bike-sharing program, Bicing, offers an eco-friendly alternative to traditional transportation methods, reducing the reliance on cars. Bicing offers 6,000 bikes across 420 stations, facilitating 14 million rides yearly, notably cutting congestion and pollution.

Figure 2.2

smart city with IOT

Electric Vehicles: Barcelona's extensive network of 500+ electric vehicle charging stations, coupled with incentives like free parking and access to low-emission zones, has significantly encouraged the shift towards cleaner transportation and reduced air pollution.

3. Citizen Engagement

A cornerstone of Barcelona's smart city strategy is its focus on citizen engagement and participation. The city has implemented several platforms and initiatives to involve residents in decision-making processes:

- **Decidim Barcelona:** An online platform where citizens can propose and vote on city projects and policies, fostering a sense of community and shared responsibility.
- **Citizen Sensors:** Residents can use mobile apps to report issues such as potholes or broken streetlights, thereby enabling the city to address these problems more efficiently.
- **Open Data Portal:** Barcelona provides access to a wealth of public data, allowing citizens and developers to create innovative solutions that address urban challenges.

4. Economic Development

Barcelona's smart city initiatives have also spurred economic development and innovation. The city has created a conducive environment for startups and tech companies, resulting in:

- **Innovation Districts:** Areas like the 22@ district serve as hubs for technology and innovation, attracting businesses and fostering collaboration. The district is home to over 1,500 companies, generating thousands of jobs and contributing significantly to the local economy.
- **Public-Private Partnerships:** The city partners with private companies to implement smart city solutions, leveraging expertise and resources from both sectors. These partnerships have resulted in innovative projects, such as the deployment of 5G technology and the creation of smart grids.
- **Job Creation:** The focus on technology and innovation has led to the creation of new job opportunities, contributing to the city's economic growth. The smart city initiatives have generated over 47,000 jobs, ranging from tech development to urban planning and sustainability roles.

2.2. RFID-based Attendance systems

1. Introduction

In today's fast-paced world, traditional methods of attendance tracking in various institutions have become time-consuming and inefficient. These methods lack real-time visibility, making it difficult to address issues like student absenteeism and proxy attendance. Manual attendance management has been a staple in the educational system, requiring significant administrative efforts, consuming valuable time, and prone to errors. To address these challenges, Radio Frequency Identification RFID technology has been introduced that offers a practical solution to streamline attendance tracking and managing processes. By using RFID-based Attendance systems, the accuracy and overall timeconsuming issues in managing student attendance can be addressed. Integrating RFID with IoT principles allows for interconnected devices and real-time data communication. IoT facilitates seamless data exchange between RFID readers, and databases enabling remote monitoring and instant access to attendance information. This paper proposes an innovative and cost-effective attendance monitoring solution that combines RFID technology, IoT principles, and the low-cost, Wi-Fienabled ESP8266 microcontroller. The proposed system can seamlessly integrate RFID readers with a centralized attendance database, enabling realtime data synchronization and remote monitoring capabilities. The RFID reader captures attendance data and wirelessly transmits it to a central database using the ESP8266's Wi-Fi capabilities. This wireless approach eliminates the need for expensive wired systems, reducing maintenance costs and enhancing the overall system design. With the integration of RFID technology, educational institutions can revolutionize their attendance monitoring processes, paving the way for enhanced efficiency and accuracy. By automating attendance

2.2.1 Literature Review:

Ms.G.T. Bharathy, Ms.S. Bhavanisankari, T. Tamilselvi [2021]:

The proposed system is an IoTbased smart attendance system using RFID. The existing manual attendance registration can be transformed into an efficient and error-free attendance management system. This system also consists of a GSM module system to send messages to parents or authorized persons and a GPS module system that gets the live location of students. The proposed system is easy to handle and very convenient for organizations.

Hariom Krishna and Narayan Vyas [2022]:

The proposed system focuses on better user interaction facilities for attendance monitoring and it is WI-FI enabled which makes it a portable device. Mobile applications are also developed that provide multiple device management facilities so that we can use multiple devices in a single user account. The application provides facilities to generate advanced sorting and filtering options. It also allows users to share reports directly from the application screen. For security purposes, google's Firebase management system is used for credential information and security of the user's data system.It can also provide a backup of the database which is provided as per the demand.

Prine Ana, Ukoette Jeremiah Ekah, Emmanuel Oyojta. [2022]:

The proposed system aims to provide a safe and secure environment on campus, a biometric security system is key. Biometric security systems are unique to each person, permanent, fast, accurate, not subjected to change, and impossible to forge. This system uses a fingerprint sensor with a microcontroller ESP32 and an OLED display for displaying the name and roll number of students. This system reduced the level of fraud and data manipulation in filling the attendance list during lectures and examinations.

Deki Juniasyah, Novi Herawadi, Danang Ade Muktiawan [2023]:

The proposed system aims to make it easier to manage attendance and calculate salaries, as well as ensure that people who must attend activities actually attended. Using RFID and ESP8266 microcontrollers, they developed an attendance system and successfully implemented it. The user's time stored in the database when making a presence corresponds to the real-time when making a presence.

Analysis / Summary

1. **Old Projects:**
    - Most RFID attendance systems were **offline** or relied on **SMS notifications**.
    - Hardware mostly included **Arduino with SD card** or **Arduino + GSM**, which increased cost and complexity.

- Limited or no **real-time web-based monitoring**.

2. **Our Project Advantages:**
   - Uses **ESP8266** → low-cost Wi-Fi enabled microcontroller.
   - **Web portal integration** allows admin and faculty to view attendance remotely in **real-time**.
   - Supports multiple sessions, automated record management, and analytics.
   - Scalable for multiple classes without additional GSM or cloud modules.
3. **Conclusion:**
   - Compared to previous RFID attendance systems, our project provides **real-time monitoring**, **low-cost deployment**, and **user-friendly web interface**, making it superior for modern classroom and institutional needs.

2.2.3 Literature Review on RFID Attendance Systems:

S.No	Author / Year	Title of Existing System	Hardware Used	Methodology / Technology	Features	Pros	Cons	Comparison with Our Project (ESP8266 Web Portal)
1	Priya et al., 2018	RFID Based Attendance Management System	Arduino UNO, RFID Reader (RC522), RFID Cards	Students swipe RFID card → Data stored on SD card / local PC	Simple RFID logging, offline storage	Easy to implement, low cost	No real-time updates, no web portal, manual record retrieval	Our system uses **ESP8266 Wi-Fi** → real-time updates to web portal; automated record storage; remote access
2	Kaur & Singh, 2019	Smart Attendance System using RFID and GSM	Arduino, RFID Reader, GSM Module	Sends SMS alerts to students/faculty on attendance	SMS notifications, basic tracking	Automated notification	Limited storage, no online dashboard	Our system uses **Web Portal** instead of SMS → centralized online records, easy analytics
3	Ahmed et al., 2020	IoT Based Student Attendance Monitoring	Raspberry Pi, RFID Reader, Cloud Server	RFID data uploaded to cloud via Raspberry Pi	Cloud storage, remote access	Centralized access, analytics possible	High hardware cost, complex setup	Our project is **ESP8266-based**, smaller, cheaper, and easy to deploy; still supports web portal analytics
4	Suresh & Reddy, 2021	RFID Attendance with Mobile App	Arduino Mega, RFID, Android App	Attendance recorded and sent to mobile app via Bluetooth	Mobile app notifications, offline mode	Mobile integration	Bluetooth range limited; no web dashboard	Our system uses **Wi-Fi module ESP8266** → works over internet, no Bluetooth limitation, web portal for admin
5	Our Project, 2025	Smart Attendance System with ESP8266 Using Web Portal	ESP8266, RFID Reader, Web Server	RFID scanned → ESP8266 sends data to web portal → Admin can view attendance online	Real-time web portal, multiple session tracking, automated records	Low-cost, real-time, web access, easy analytics	Requires Wi-Fi connection for portal	Improvement over all previous systems: **real-time cloud/web monitoring, session tracking, low-cost ESP8266 integration**

CHAPTER-III

SYSTEM DESIGN

3.1 INTRODUCTION:

Attendance management is a critical function in educational institutions, workplaces, and training centers. Traditional methods involve **manual entry of attendance**, such as calling names or signing sheets. These systems are prone to **errors, time delays, and manipulation**.

To address these challenges, **digital attendance systems** have emerged. Among them, **RFID-based systems** are widely used due to their **automation, reliability, and ease of use**. RFID (Radio-Frequency Identification) enables identification of individuals through **cards or tags**, eliminating manual errors and speeding up the attendance process.

However, many existing RFID systems still have limitations:

- Offline storage on microcontrollers or SD cards.
- Limited remote access or monitoring.
- Lack of real-time data visualization.

To overcome these limitations, our project integrates **RFID technology with ESP8266 Wi-Fi module** and a **web portal**, creating a **smart, automated, and real-time attendance system**.

3.1.2 ESP8266 Wi-Fi Module (NodeMCU/ESP-12)

The ESP8266 Wi-Fi Module (NodeMCU/ESP-12) is a low-cost, integrated system-on-a-chip (SoC) that provides Wi-Fi connectivity for embedded systems and is popular for IoT and rapid prototyping. The ESP-12 is the actual module chip, while NodeMCU is a development board that integrates the ESP8266 chip, includes a USB-to-serial converter, and provides easy access to General Purpose Input/Output (GPIO) pins for programming, often with a user-friendly firmware like Lua.

Key features

- Integrated MCU and Wi-Fi: The ESP8266 chip has a built-in processor and the ability to connect to a Wi-Fi network, allowing it to function independently or as a slave to another microcontroller.
- Networking: It supports the IEEE802.11 b/g/n wireless LAN standards and has a full TCP/IP stack, enabling it to act as a Wi-Fi client or host access points.

- GPIO access: The NodeMCU development board breaks out numerous GPIO pins, including PWM, IIC, and ADC, making it easy to connect sensors and other devices.
- Programming: The NodeMCU board is often pre-flashed with a user-friendly firmware like NodeMCU (Lua-based) or can be programmed using environments like the Arduino IDE by adding the necessary ESP8266 packages.

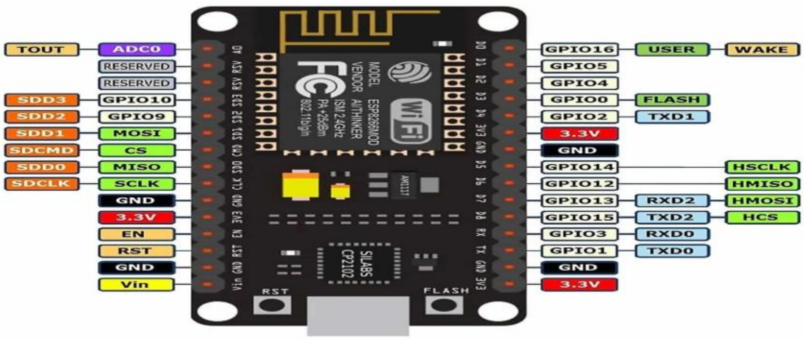

Figure 3.1

ESP 8266 PINS

- Development board components: The NodeMCU development kit includes a USB-to-serial chip (like CP2102 or CH340G) for easy code uploading, a voltage regulator, and other components for a complete, breadboard-friendly package.
- Low cost and community: The ESP8266 is a very inexpensive solution for adding Wi-Fi to a project and benefits from a large and active online community.

3.1.3. RFID reader module

An RFID reader module is a crucial component for communicating with RFID tags using radio waves. Its importance lies in enabling automatic data capture for tracking and inventory management in various applications like asset tracking, library systems, and contactless payments. A common and popular module for hobbyist projects is the RC522, which works by emitting an electromagnetic field to power and read passive tags at a distance of a few centimeters.

How RFID reader modules work

- **Transmission**: The module's antenna transmits a high-frequency radio wave, creating an electromagnetic field.

- **Tag activation**: When a compatible RFID tag enters this field, the tag's antenna harvests energy from the wave, which powers its microchip.
- **Data capture**: The tag then modulates the electromagnetic field to send its data back to the reader's antenna through a process called backscattering.
- **Processing**: The reader module's control unit receives and processes this data, which can then be sent to a microcontroller or a central database for analysis.

Figure 3.2 RFID modules

Importance and applications

- **Asset management**: RFID modules allow for automated tracking of assets like equipment, inventory in warehouses, and books in libraries, reducing manual labor and errors.
- **Access control**: They are used in office entry systems and for identifying people or objects.
- **Contactless payments**: Some modules are used in systems for contactless credit cards.
- **Automation**: They are a key part of automatic identification and data capture (AIDC) technologies.

3.1.4. RFID cards and tags:

RFID cards and tags are important because they allow for the wireless identification, tracking, and management of assets and people, which increases efficiency, security, and accuracy across many industries. Their purpose is to store and transmit data to a reader using radio waves, enabling contactless transactions, access control, and real-time inventory management without a direct line of sight.

Importance

- **Increased efficiency:** RFID systems can automate processes like inventory checks and access control, saving time and reducing labor costs.
- **Improved accuracy:** By eliminating manual data entry and direct line-of-sight scanning, RFID reduces human error in tracking items.
- **Enhanced security:** RFID provides contactless, secure access control for buildings and can be used for cashless payments and secure identification.
- **Real-time data:** RFID allows for real-time tracking of assets and inventory, which helps prevent stockouts, theft, and loss.

Figure 3.3 RFID cards

Purpose and uses

- **Asset and inventory management:** RFID tags are used to track products from manufacturing to the point of sale, providing real-time visibility into inventory levels and location.
- **Access control:** RFID cards are used for secure access to buildings, rooms, and events by verifying a person's identity.
- **Contactless payments:** The technology is used in public transport, retail, and payment systems, allowing for quick and easy cashless transactions.
- **Identification:** They serve as staff, student, and visitor ID badges, enabling identification and tracking within a facility.
- **Healthcare:** RFID is used to track medical equipment, manage medication inventory, and monitor the location of staff and patients.
- **Logistics and supply chain:** RFID tags track shipments and manage inventory throughout the supply chain, improving efficiency and accuracy

3.2. SOFT WARE COMPONETES :

3.2.1 The Arduino IDE:

The Arduino IDE is a cross-platform software for writing and uploading code (sketches) to Arduino boards, which is based on a simplified C++ language. Key features include a built-in text editor with syntax highlighting, a toolbar for common functions like verifying and uploading code, a compiler that translates the code into machine-readable instructions, and a serial monitor for communication with the board. The IDE also manages libraries and board definitions, and a new version adds features like auto-completion and debugging.

Core features

- **Cross-platform:** The software runs on Windows, macOS, and Linux.
- **Text editor:** An integrated editor for writing code with features like syntax highlighting, search/replace, and auto-formatting.
- **Toolbar:** Provides quick access to essential functions, including buttons for verifying, uploading, and managing sketches (new, open, save).
- **Compiler:** Converts your code into a format the Arduino board's microcontroller can understand.
- **Upload:** Allows you to upload the compiled code to the Arduino board via a USB connection.
- **Serial Monitor:** A tool to view data sent from the Arduino board, which is useful for debugging and monitoring.
- **Message and console areas:** Displays feedback during compiling and uploading, and provides error messages.
- **Board and port selection:** You must select the correct board and serial port from the Tools menu so the IDE can communicate with your hardware.

Figure 3.4.

Arduino IDE logo

Important functionalities

- **Sketchbook:** Stores your code files locally on your computer. It can also sync with the Arduino Cloud.
- **Library Manager:** Browse and install libraries that add functionality to your code, created by Arduino and the community.
- **Board and port management:** When you plug in your Arduino, you must select it and its corresponding serial port from the Tools menu to upload your sketch.
- **Examples:** The IDE includes a "Blink" example and other basic sketches to help you get started.
- **Customization:** You can change settings like the theme, font size, and save location through the Preferences menu.

3.2.2. XAMPP

XAMPP is a free, open-source, and cross-platform software package that creates a local web server environment for testing and development. It includes Apache, MariaDB (or MySQL), PHP, and Perl, making it easy for developers to run web applications on their own computer before deploying them to a live server.

Figure 3.5 XAmpp

- **Components:** XAMPP stands for **X** (cross-platform), **A**pache, **M**ariaDB, **P**HP, and **P**erl. It also includes other tools like phpMyAdmin for database administration.
- **Purpose:** It's a popular choice for creating a development environment to test and build web applications locally.
- **Ease of use:** The package is designed to be easy to install and use, making it a good option for beginners.

- **Functionality:** By running a local server, XAMPP allows developers to test their websites and applications without needing a live server, and it provides a seamless transition to a live environment.
- **Availability:** It is available for Windows, macOS, and Linux.

3.2.3. PhpMyAdmin:

phpMyAdmin is a free, open-source, web-based tool written in PHP, designed to handle the administration of MySQL and MariaDB database servers. Its importance stems from its ability to simplify and streamline various database management tasks through a user-friendly graphical interface, making database administration accessible to a wider range of users, including those without extensive command-line experience.

Key aspects highlighting its importance include:

- **User-Friendly Interface:**

It provides an intuitive web interface for managing databases, tables, columns, relations, indexes, users, and permissions, eliminating the need to write complex SQL queries for many common operations.

- **Database Management:**

Users can easily create, modify, and delete databases and tables, as well as add, edit, and delete data within tables.

- **SQL Execution:**

It allows users to execute SQL queries directly within the application, enabling advanced database operations, testing queries, and performing complex data manipulations.

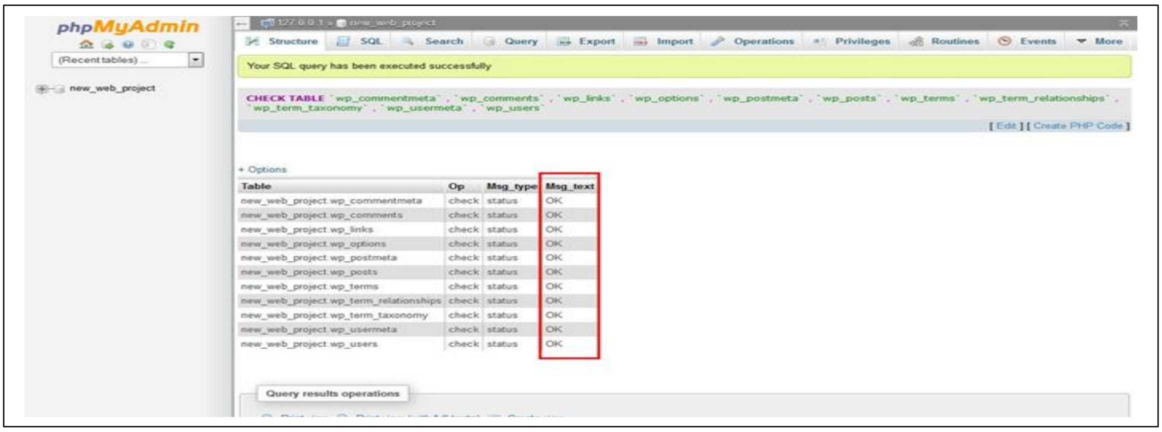

Figure 3.6

Phpmyadmin page

- **Data Import/Export:**

phpMyAdmin facilitates importing data from various file formats (e.g., CSV, SQL) into databases and exporting data in different formats (e.g., CSV, SQL, XML) for backups or analysis.

- **User and Permission Management:**

It offers tools to create and manage user accounts, assign specific privileges to users, and control access to databases and tables, ensuring proper security and access control.

- **Server Status Monitoring:**

It provides insights into the MySQL server's status, including traffic, connections, and failed attempts, which can be useful for performance monitoring and troubleshooting.

- **Accessibility:**

As a web-based tool, it can be accessed remotely through a web browser, making it convenient for managing databases from various locations.

CHAPTER – IV

SYSTEM CONSTRUCTION AND TESTING

4.1. INTRODUCTION OF PROJECT

The **Smart Attendance System with ESP8266 Using Web Portal** is an IoT-based solution designed to automate and simplify the process of recording student attendance. In this system, each student is provided with a unique **RFID card**. When a student scans their card on the **RFID reader**, the **ESP8266 microcontroller** reads the card's unique ID and immediately sends the attendance data over a **Wi-Fi connection** to a **web portal**. The web portal, connected to a **database**, stores and manages the attendance records in real-time. Faculty or administrators can log in to the portal to **view, update, and generate reports** for multiple sessions.

This system eliminates manual attendance entry, reduces errors, and allows for remote monitoring. By integrating **RFID technology, Wi-Fi communication, and a web-based interface**, the project ensures efficient, secure, and real-time attendance management suitable for modern educational institutions.

4.2. PROJECT IMPLIMENTATION:

4.2.1. CREATE A NEW DATA BASE:

Step 1: Install XAMPP

1. Download XAMPP from the official site: https://www.apachefriends.org
2. Install it on your computer (default location is fine).
3. After the XAMPP install to search in search bar as XAMPP CONTROL PANEL

   & to select the Run as administrator

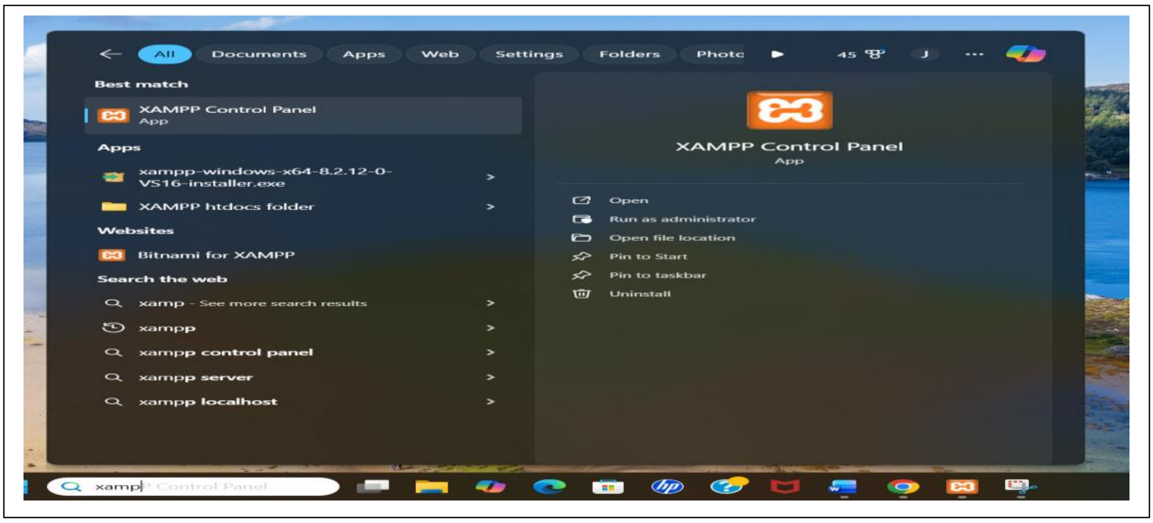

Figure 4.1

for searching of XAMPP CONTROL PANEL

4. Now you see the new window page like this

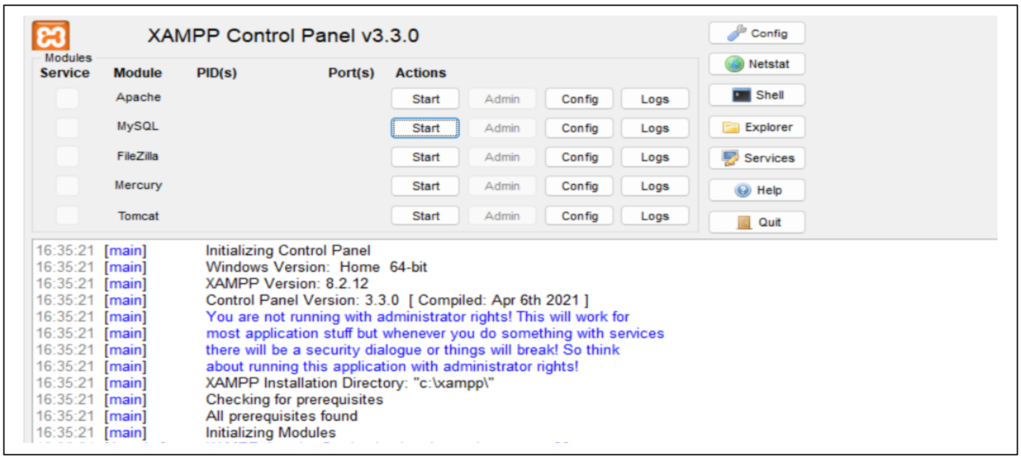

Figure 4.2

XAMPP CONTROL PANNEL V.3.30

Step 2: Start XAMPP Services

1. Start the following modules:

   Apache → for running PHP web pages

MySQL → for database server

You should see **green indicators** for both modules.

Figure 4.3

Start the server

Step 3: Open phpMyAdmin

1. In XAMPP Control Panel, click **Admin** next to MySQL.
2. This will open **phpMyAdmin** in your browser:
   - o URL: http://localhost/phpmyadmin
3. phpMyAdmin is the interface to create and manage databases.

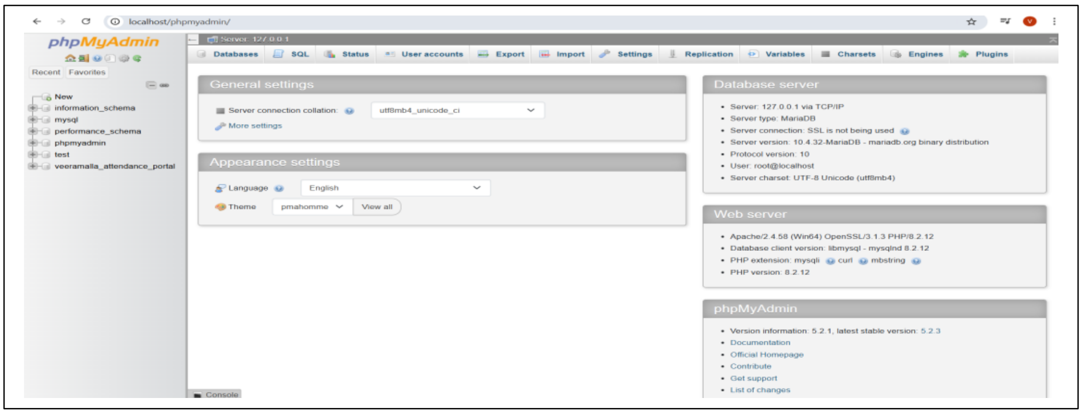

Figure 4.4.

http://localhost/phpmyadmin

Step 4: Create the Database

1. Click SQL tab in phpMyAdmin.
2. Paste/ write your SQL code

3. Click Go / Execute.
4. If successful, the database rfidattendance and all tables (faculty, course, student, device, attendance) will be created.

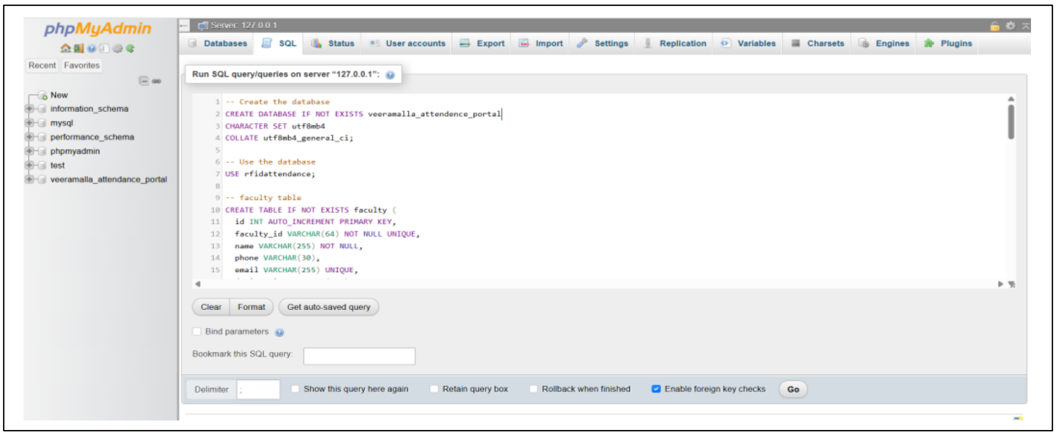

Figure 4.5

SQL code for create data base

Table	Action						Rows	Type	Collation	Size	Overhead
attendance	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
attendance_log	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
attendance_records	Browse	Structure	Search	Insert	Empty	Drop	9	InnoDB	utf8mb4_general_ci	16.0 KiB	-
attendance_sessions	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
biometric_registration	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
course_registration	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
esp_scans	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
faculty	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	utf8mb4_general_ci	48.0 KiB	-
studentregistration	Browse	Structure	Search	Insert	Empty	Drop	18	InnoDB	utf8mb4_general_ci	32.0 KiB	-
9 tables	**Sum**						**32**	**InnoDB**	**utf8mb4_general_ci**	**192.0 KiB**	**0 B**

Figure 4.6

Data base creates with Table

Step 5: Place Your Web Portal Files

1. Go to the **XAMPP installation folder** → htdocs
   - Example: C:\xampp\htdocs
2. Create a new folder for your project, e.g., rfid_portal.
3. Place your **PHP, HTML, CSS, JS files** in this folder.
4. Access your portal in browser.

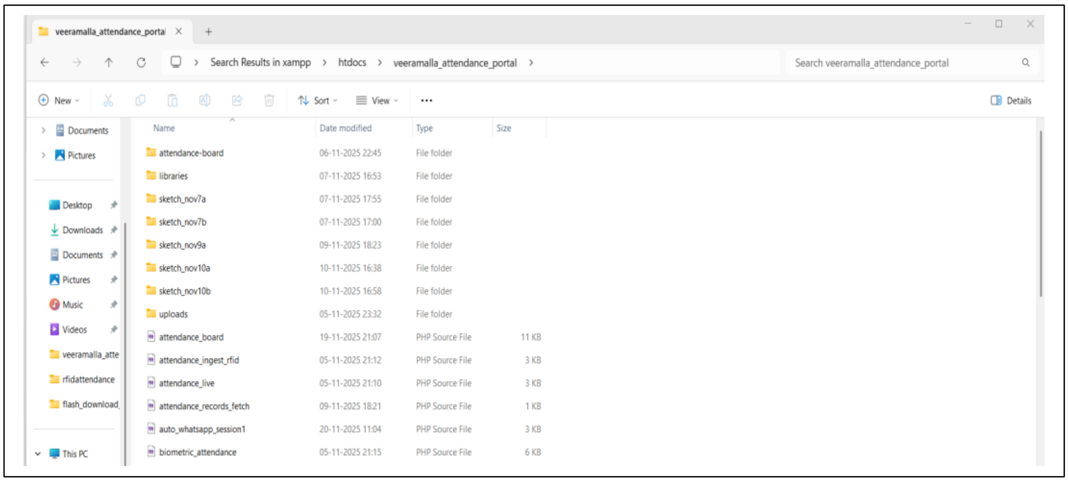

Figure 4.7

XAMPP installation folder → htdocs

4.3. WEB PORTAL DESIGN LAYOUT PROCESS

4.3.1. LOGIN PAGE DESIGN FLOW

Purpose

Entry point for faculty authentication.

Layout Structure

- Background:
  Smooth dark gradient (blue/teal tones).
- Foreground Card:
  Center-aligned login box with:
    - Title: *Faculty Login*
    - Input fields: Faculty ID, Password
    - Login button
    - Additional link: "Register New Faculty"

UI Principles Used

- Minimalistic form
- Center alignment for focus
- High contrast button

- Rounded corners for modern look
- Soft shadow to elevate card

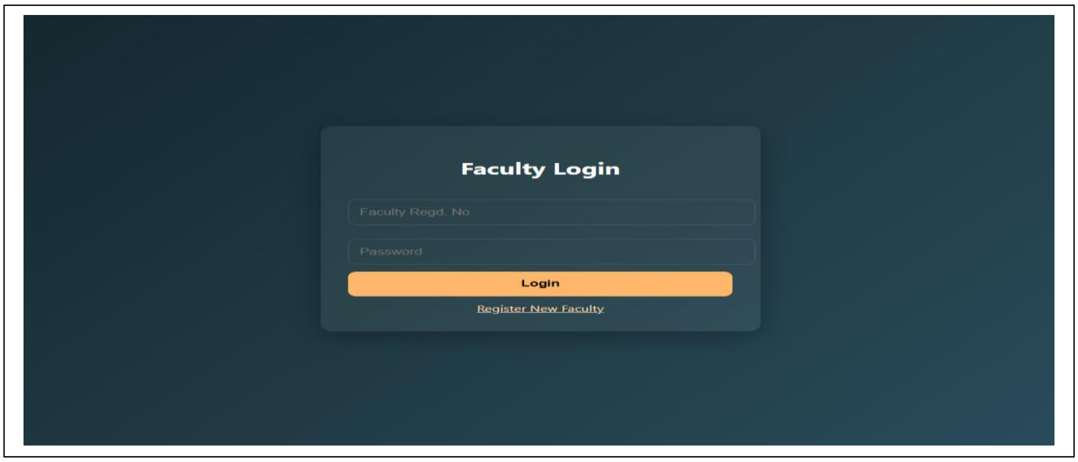

Figure 4.8 login page

4.3.2. MAIN DASHBOARD DESIGN FLOW

Purpose

Faculty home after login showing main available modules.

Header Layout

- **Left:**
  ▯ Portal title: *LBRCE Attendance Portal*
  ▯ Welcome message
- **Right:**
  ▯ Faculty name
  ▯ ID
  ▯ Profile picture
  ▯ Logout button

Body Layout

Displayed as **5 modules** using large tiles:

1. Faculty Registration
2. Student Registration
3. Course Registration
4. Attendance Board

5. Reports

UI Principles Used

- Card-based layout
- Soft spotlight glow for icons
- 2-line titles
- Uniform spacing and symmetry

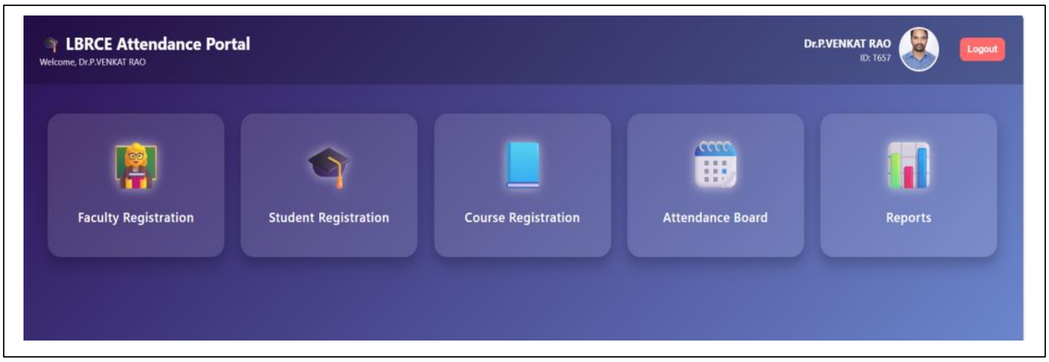

Figure 4.9

LBRCE Attendances portal Dash board

4.3.3. FACULTY REGISTRATION PAGE

Purpose

Register a new faculty account.

Layout Structure

- Top header consistent with dashboard.
- Left: Back button
- Center: Large form card containing:
    - Faculty Name, Regd. No
    - Department, Designation
    - Phone, Email
    - Username, Password
    - Photo upload

UI Design Patterns

- Two-column form layout

- Clear labels above each field
- Soft background card
- Highlighted "Register Faculty" button

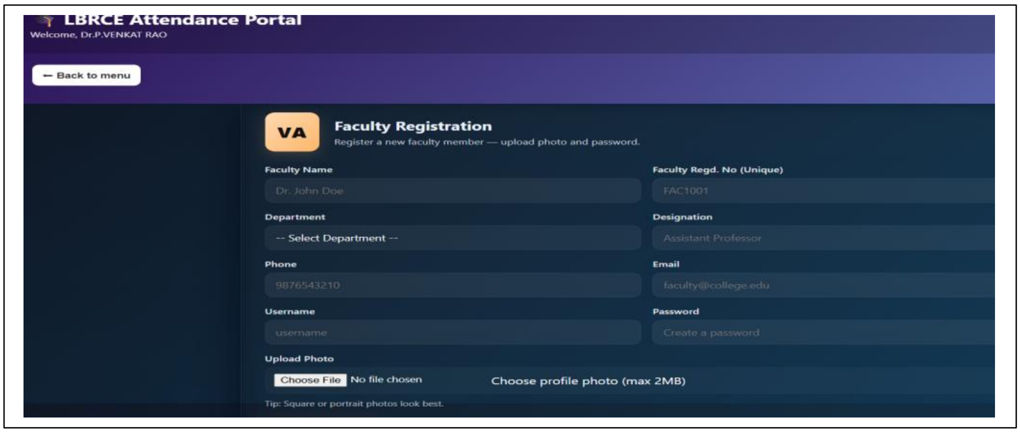

Figure 4.10

New Faculty Registrations

4.2.4. COURSE REGISTRATION PAGE

Purpose

Assign courses to faculty based on academic structure.

Layout Structure

- Header consistent across all pages.
- Form fields include:
  - Course Name
  - Course Code
  - Faculty Name & ID (auto-filled)
  - Program
  - Department
  - Year
  - Semester
  - Section
- Bottom: Wide "Register Course" button

Design Principles

- Clean white form area

- Center aligned container
- Proper spacing between rows
- Consistency across all registration forms

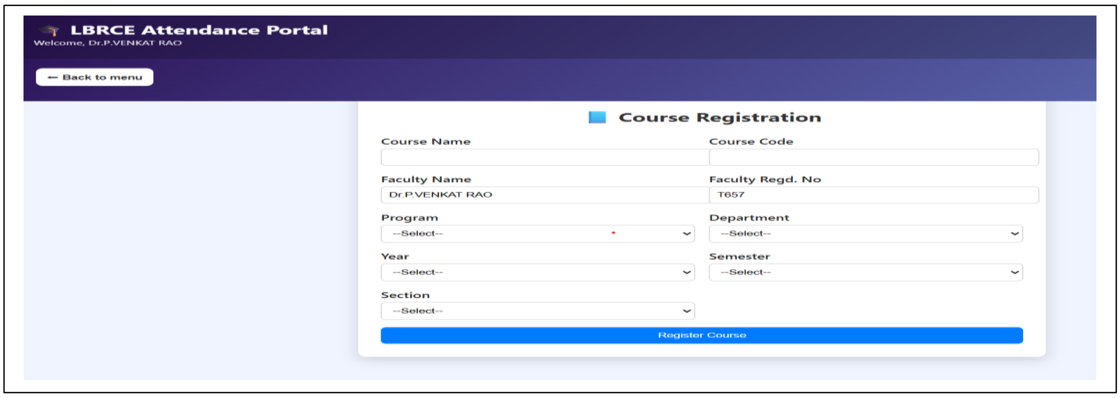

Figure 4.11.

course registrations

4.2.5 STUDENT REGISTRATION PAGE

Purpose

Enter student details.

Layout Structure

- Header identical to other pages.
- Left back button.
- Center form fields:
  - Name, Regd. No, Phone, Email
  - Program, Department, Year, Semester
  - Section
  - Photo upload
  - Submit button

Design Principles

- Dropdown menus for academic fields
- Section divided by categories (Personal / Academic / Upload)
- Bright green submit button for visibility

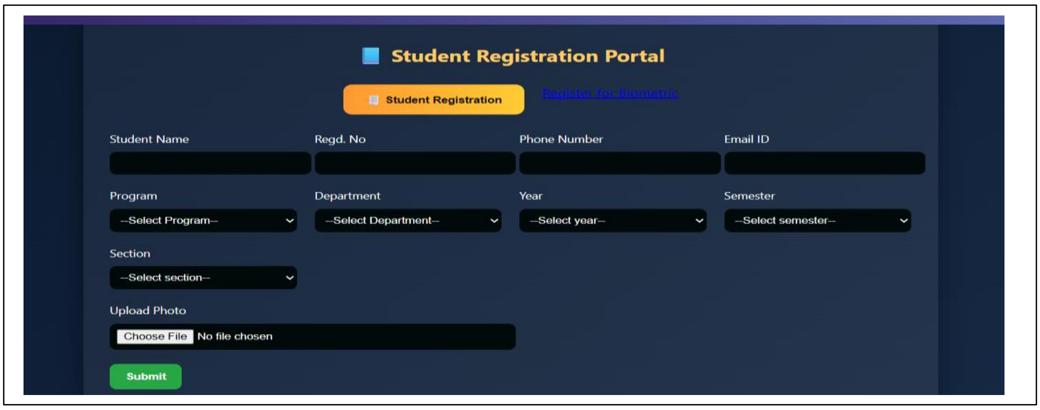

Figure 4.12

Student's registrations

4.2.6. Attendance board system Page

Purpose

Collect and enter complete student details for registration into the system.

Layout Structure

- **Header** consistent with all other pages for uniformity.
- **Left-side back button** for easy navigation.
- **Centered form area** containing all required input fields:
  - **Personal Details**: Name, Registration Number, Phone Number, Email
  - **Academic Details**: Program, Department, Year, Semester
  - **Section** selection
  - **Photo Upload** area
  - **Submit Button** positioned at the bottom

Design Principles

- Use **dropdown menus** for all academic-related fields to ensure correct and standardized selection.
- Divide form into clear **sections**:
  - Personal Information
  - Academic Information

- Photo Upload

- Highlight the **Submit** button in **bright green** to make it prominent and easily identifiable.

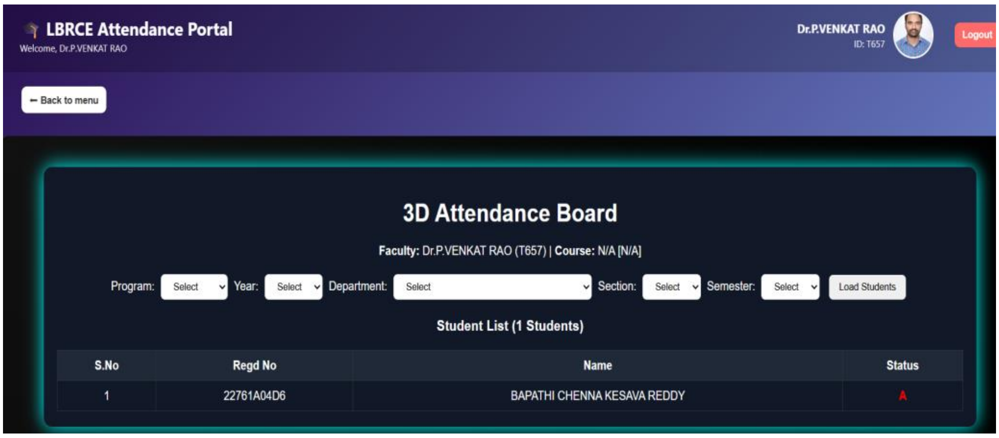

Figure 4.13

Smart RFID Attendance Board

4.3. Report page

4.3.1. Daily Attendance Report Page

Purpose

Allow staff to quickly generate attendance reports for a specific date using academic filters like Program, Year, Department, Section, and Semester.

Layout Structure

- Top navigation bar with tabs for different report types.
- Header titled *Daily Attendance Report* with a small icon.
- Horizontal filter row containing:
  - Program, Year, Department, Section, Semester
  - Date field
  - Load Report button at the end
- Wide spacing between fields to improve readability.

Design Principles

- Dropdowns used for all academic selections.
- Consistent labeling of each filter for clarity.
- Bright green Load Report button for high visibility.

- Clean horizontal alignment mimicking dashboard-style filter bars.

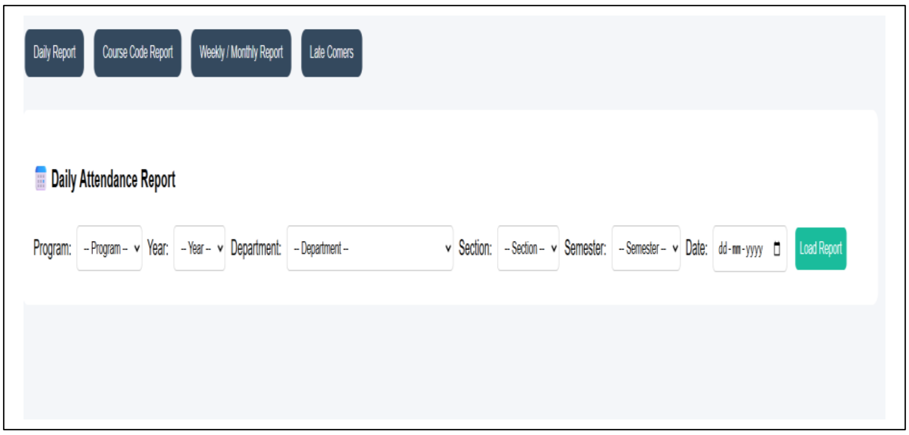

Figure 4.14 .

Daily attendance report

4.3.2. Course Code Report Page

Purpose

Provide faculty the ability to generate attendance reports filtered by course codes for specific classes or subjects.

Layout Structure

- Same header and top navigation style as other report pages.
- Central filter section containing:
  - Course Code dropdown
  - Department, Year, Semester dropdowns
  - Date or Date Range selection
  - Load Report button

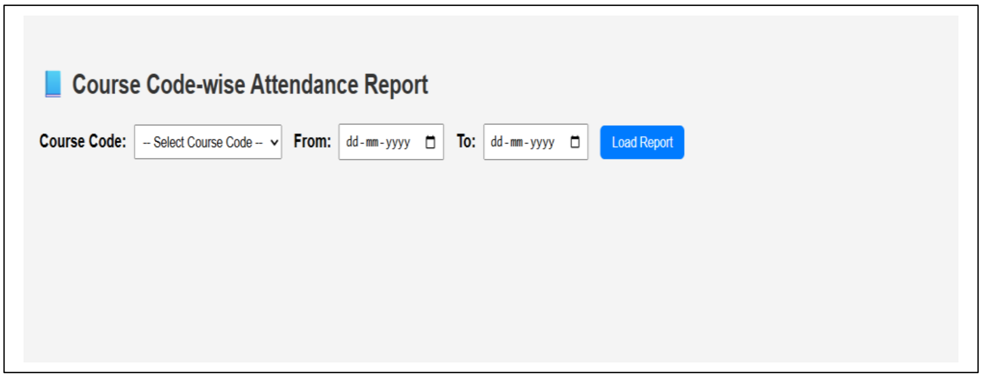

Figure 4.15 course code_wies Attendance Report

Design Principles

- Dropdown menus ensure accurate and standardized selection of course codes.
- Date range fields placed together for easy selection.
- Submit button placed at right side for consistent visual flow.
- Maintains same spacing and alignment as Daily Report Page.

4.3.3. Weekly / Monthly Attendance Report Page

Purpose

Generate attendance summaries for an entire week or month using academic filters and time range options.

Layout Structure

- Page header consistent with other report pages.
- Dropdowns for:
  - Program, Year, Department, Section
  - Week selector or Month selector
  - Semester dropdown
  - Load Report button

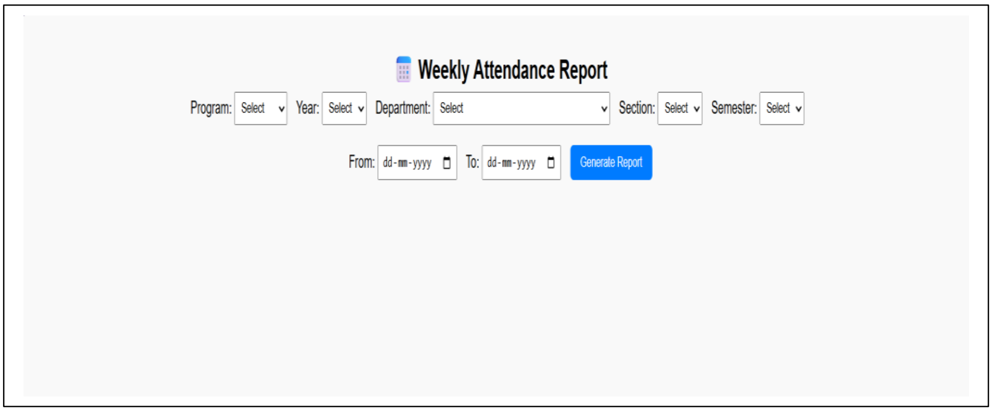

Figure 4.16 weekly Attendance Report

Design Principles

- Time-range selectors clearly separated (Weekly Block / Monthly Block).
- Dropdown grouping follows the same academic hierarchy.
- Report button highlighted in green for quick access.

- Layout spacing optimized for larger filter rows.

4.3.4. Late Comers Report Page

Purpose

Display and filter list of students who arrived late for a specific date or session.

Layout Structure

- Standard top navigation and header.
- Filter row containing:
  - Program, Department, Year, Section
  - Date field
  - Session dropdown (if needed)
  - Load Report button
- Results table displayed below filters.

Design Principles

- All academic filters arranged horizontally for quick scanning.
- Clear icon or color indicator to highlight "late" status.
- Date field placed near the end for natural left-to-right flow.
- Button remains green for consistent action branding.

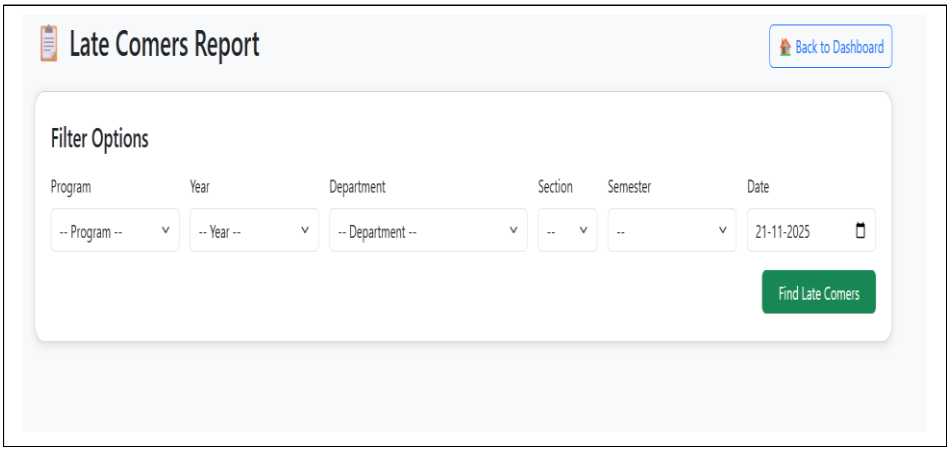

Figure 4.17 Late comers Report

4.4. Hardware Connection Process for RFID Attendance System Using Web Portal

1. Required Components

- RFID Reader (RC522 or EM-18)
- Microcontroller:
    - ESP8266/NodeMCU or ESP32 (recommended because it has Wi-Fi)
    - *OR* Arduino UNO + ESP8266 Wi-Fi module
- RFID Tags/Cards
- Jumper wires (Male–Female)
- Breadboard (optional)
- 5V USB power supply

4.4.1 RFID RC522 to ESP8266 (NodeMCU) Connection Process

The RC522 uses SPI communication, so we connect its SPI pins to the ESP8266 SPI pins.

RC522 Pin	ESP8266 (NodeMCU) Pin	Function
SDA	**D2 (GPIO4)**	SS (Chip Select)
SCK	**D5 (GPIO14)**	SPI Clock
MOSI	**D7 (GPIO13)**	SPI MOSI
MISO	**D6 (GPIO12)**	SPI MISO
IRQ	Not connected	(Optional, not needed)
GND	**GND**	Ground
RST	**D1 (GPIO5)**	Reset
3.3V	**3.3V**	Power

- **RC522 works only on 3.3V** — *never connect it to 5V*.
- Make sure all grounds (**GND-to-GND**) are connected.
- Use short wires to reduce interference.

4.4.2 I2C LCD → ESP8266 (NodeMCU) Connection

Your I2C LCD uses only **4 pins**: VCC, GND, SDA, SCL.

I2C LCD Pin	ESP8266 Pin	Function
VCC	**3.3V or VIN (5V)**	Power (most I2C LCDs need 5V)
GND	**GND**	Ground
SDA	**D2 (GPIO4)**	I2C Data
SCL	**D1 (GPIO5)**	I2C Clock

If your RC522 already uses D1 & D2 → shift the I2C pins to:

- **SDA → D3 (GPIO0)**
- **SCL → D4 (GPIO2)**
  The ESP8266 allows software I2C on any GPIO.

4.4.3. BUZZER → ESP8266 Connection

Use a 3.3V active buzzer (recommended).

For a passive buzzer, you must drive it using PWM.

Buzzer Pin	ESP8266 Pin	Function
+ (Positive)	**D8 (GPIO15)**	Control signal
– (Negative)	**GND**	Ground

Do NOT use 5V buzzers directly on GPIO pins.

SYMATIC CIRCUIT DIAGRAM

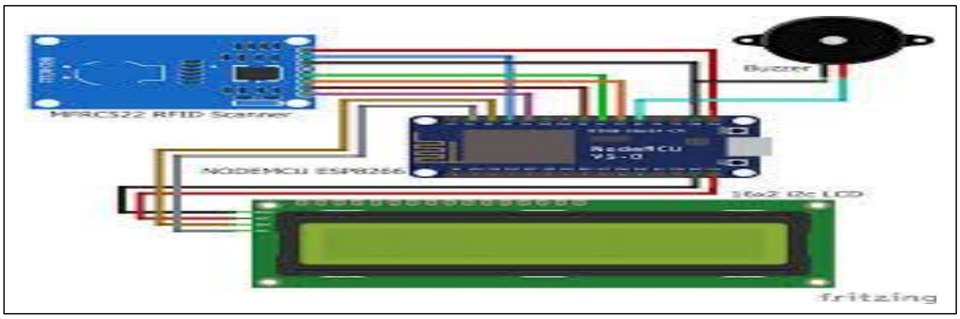

Figure 4.13 Sematic diagram

PROJECT CIRCUIT:

Figure 4.14. Implementation circuit

CHAPTER -V

RESULT AND DISCUSSION

5.1 TO FIX THE RFID CARD FOR STUDENTS:

Step 1: To Open the ARDUINO IDE Software and write the code for to fix/assign Each card each student

Step 2: Open the Arduino IDE soft ware and write the code and save in correct folder with name AS RFID_NAME_TAG

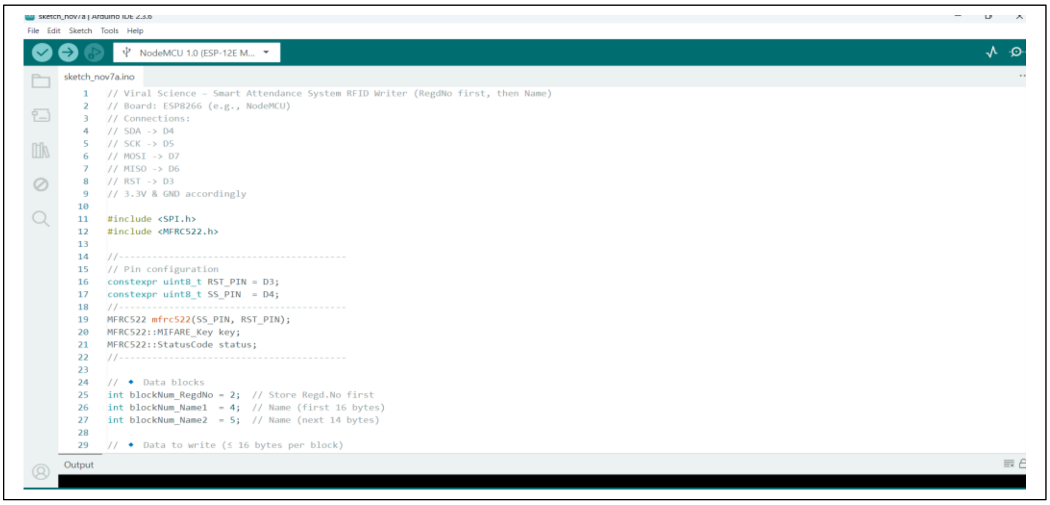

Figure 5.1 Arduino program page

Step 3: Select the Board and port for Board to select the name as Node MCU 1.0 (ESP 12E) & port is CMOS 3/4

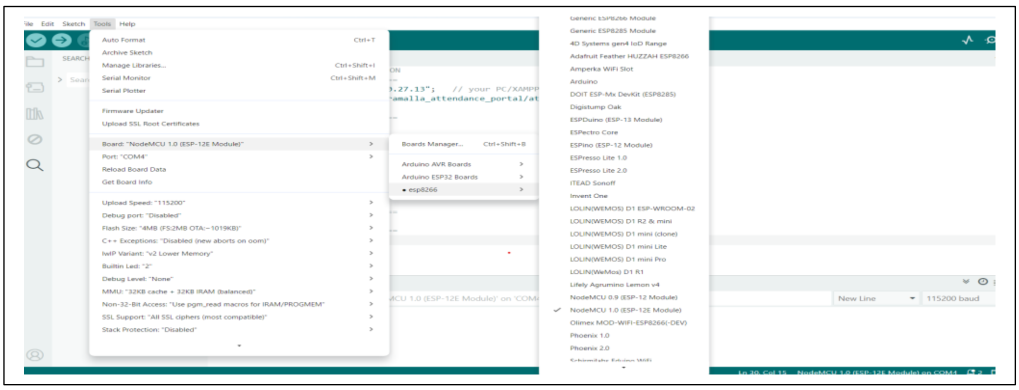

Figure 5.2 Selecting the boad

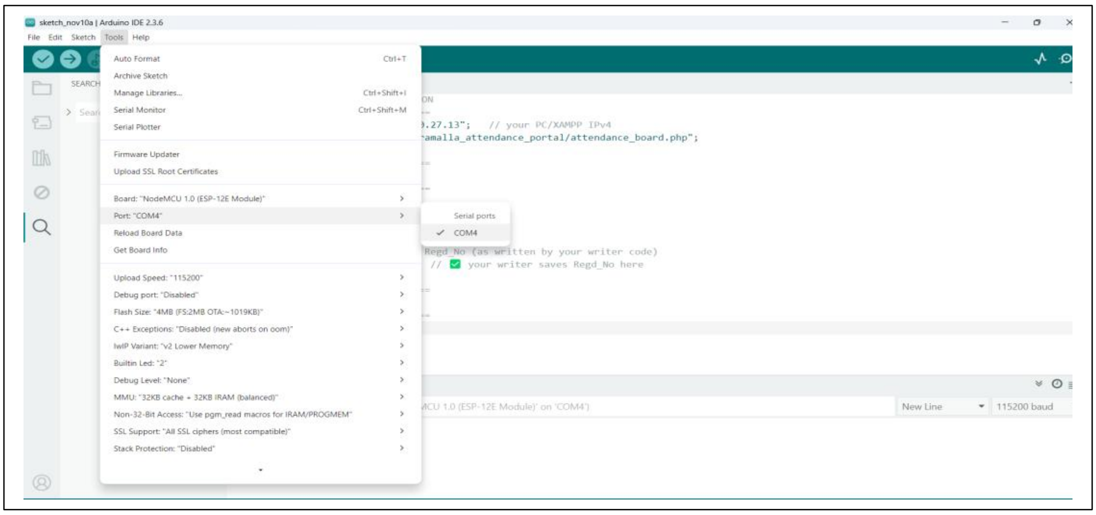

Figure 5.3 For selecting the port

Step 4: Click on the upload Batton and check if it is showing any error s and rectify the error

Step 5: After execution to go to tools and select the serial monitor (ctrl+shift+M) .

Step 6: To show the serial monitor screen in bellow of the window of program page .

Step 7: First to put the RFID card in front of the RFID module the reader to read CARD UID and store the student's name & regd_no in card memory.

```
Read success!
Reading Name Blocks...
Read success!
Read success!

--- Data Stored on RFID Tag ---
Regd.No : 22761A04D6□□□□□□
Name    : BAPATHI CHENNAKESAVA REDDY□□□□
-------------------------------

✅ Done. Remove card and scan next.
======================================
```

Figure 5.4 Data stored on RFID Tag

5.2 RFID ATTENDENCE SYSTEM:

Step 1: In Dash board to click on the **Attendance board** and select Required option for filtering the students.

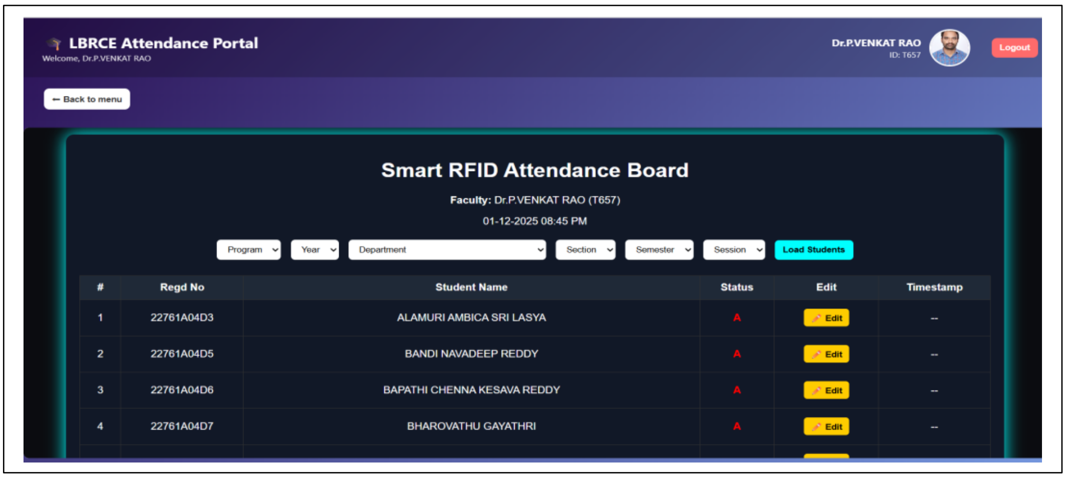

Figure 5.5.

Student detail as per dept, year, sem, section.

Step 2: Open the Arduino IDE and write the code for scanned the student Id card and to send the data to server the server is send the data to the Web page.

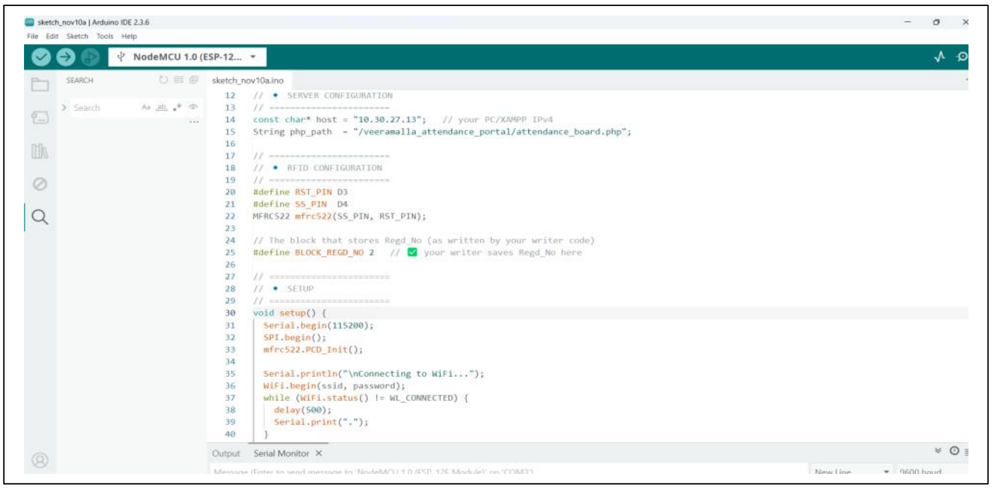

Figure 5.6 code for scanned data

Step 3: Then again to select the Board& port after this to select the upload Batton and execute the program as per the circuit.

Step 4: After the execution to select the serial monitor and shown the bellow the program

Figure 5.7. Serial monitor

Step 5: Then to go to attendance board page then scanned the RFID card. automatically attendance is posted the web page and high light that name & regd_no of the student.

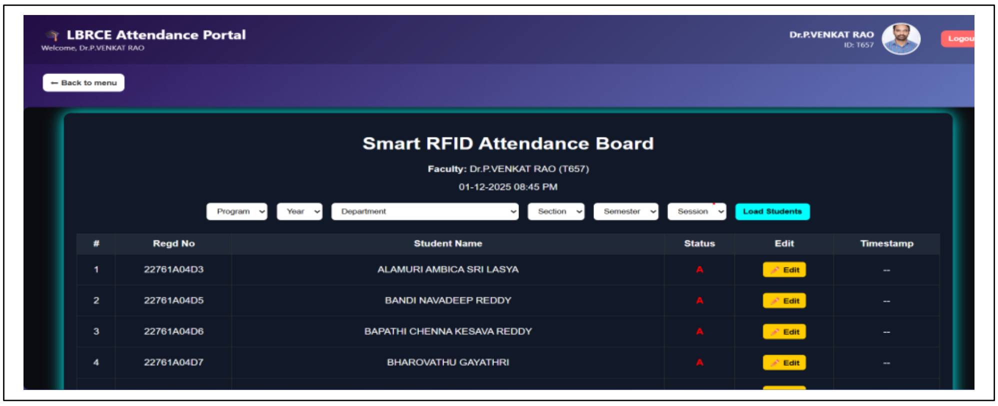

Figure 5.8. Before the scanning

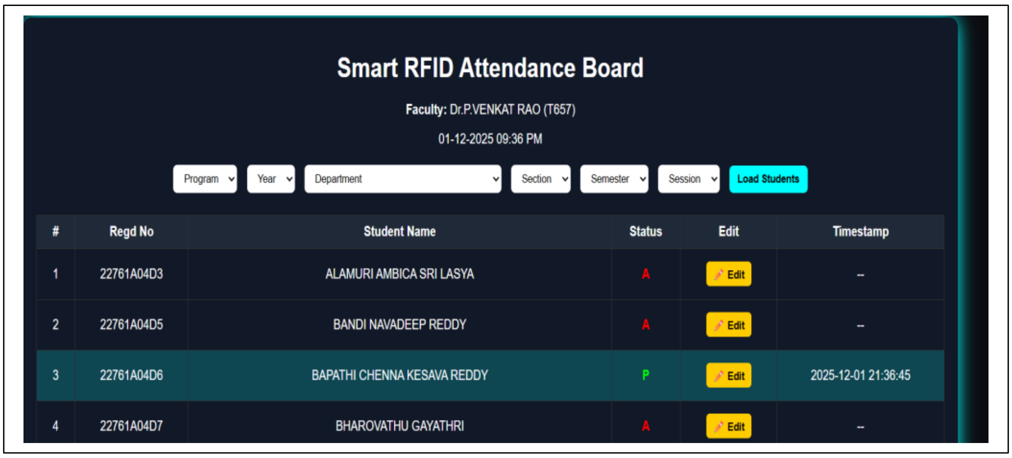

Figure 5.9. After scanning the RFID card

Step 6: After the 20 min the attendance is submit automatically in data base.

Edit	Copy	Delete	494	22761A04D3	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	495	22761A04D5	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	496	22761A04D6	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	P	1
Edit	Copy	Delete	497	22761A04D7	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	498	22761A04D8	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	499	22761A04D9	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	500	22761A04E0	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	501	22761A04E1	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	502	22761A04E2	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	503	22761A04E3	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	504	22761A04E4	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	505	22761A04E5	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	506	22761A04E6	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Edit	Copy	Delete	507	22761A04E7	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1
Console	Copy	Delete	508	22761A04E8	B.TECH	IV	Electronics & Communication Engineering	C	22ECE21	Dr.P.VENKAT RAO	2025-12-01 19:42:28	A	1

Figure 5.10. attendance data in Data base

Step7: To click on the back Batton and go to report then open new page.

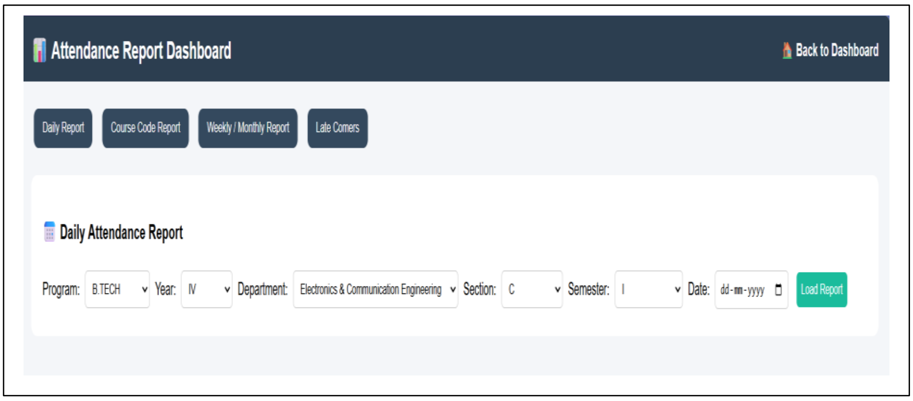

Figure 5.11. attendance report page

Step 8: For the Different operations to click required fields

1. **Daily Attendance Report**
2. Course Code-wise Attendance Report
3. Weekly Attendance Report
4. Late Comers Report

Step 9: For Different Reports to select the required Fields as per data. To get the that Data from data base.

Program: -- Program -- Year: -- Year -- Department: -- Department -- Section: -- Section -- Semester: -- Semester -- Date: dd-mm-yyyy Load Report

S.No	Regd No	Name	S1	S2	S3	S4	S5	S6	Total Present
1	22761A04D3	ALAMURI AMBICA SRI LASYA	A	2	A	A	A	A	1
2	22761A04D5	BANDI NAVADEEP REDDY	A	2	A	A	A	A	1
3	22761A04D6	BAPATHI CHENNA KESAVA REDDY	1	2	A	A	A	A	2
4	22761A04D7	BHAROVATHU GAYATHRI	A	A	A	A	A	A	0
5	22761A04D8	BHOGYAM LOKESWAR	A	2	A	A	A	A	1
6	22761A04D9	BHUKYA BALAJI	A	A	A	A	A	A	0
7	22761A04E0	BONIGE PRASANTHI	A	2	A	A	A	A	1
8	22761A04E1	CHEKURI HRUDAY CHOWDARY	A	A	A	A	A	A	0
9	22761A04E2	CHERUKULA BHUVANESWAR REDDY	A	2	A	A	A	A	1
10	22761A04E3	CHODAGAM KHYATHI SREE	A	2	A	A	A	A	1
11	22761A04E4	DARAPUREDDY KALYAN KUMAR	A	2	A	A	A	A	1
12	22761A04E5	DASARI ABHITEJ	A	A	A	A	A	A	0

Figure 5.12. Daily Attendance Report

Course Code-wise Attendance Report

Course Code: -- Select Course Code -- From: dd-mm-yyyy To: dd-mm-yyyy Load Report

S.No	Regd No	Name	19-Nov	20-Nov	01-Dec	Total Present
1	22761A04D3	ALAMURI AMBICA SRI LASYA	1	2	3	3
2	22761A04D5	BANDI NAVADEEP REDDY	A	A	1	1
3	22761A04D6	BAPATHI CHENNA KESAVA REDDY	1	A	2	2
4	22761A04D7	BHAROVATHU GAYATHRI	A	A	A	0
5	22761A04D8	BHOGYAM LOKESWAR	A	A	1	1
6	22761A04D9	BHUKYA BALAJI	A	A	A	0
7	22761A04E0	BONIGE PRASANTHI	A	A	1	1
8	22761A04E1	CHEKURI HRUDAY CHOWDARY	A	A	A	0
9	22761A04E2	CHERUKULA BHUVANESWAR REDDY	A	A	1	1
10	22761A04E3	CHODAGAM KHYATHI SREE	A	A	1	1
11	22761A04E4	DARAPUREDDY KALYAN KUMAR	A	A	1	1

Figure 5.13. Course Code-wise Attendance Report

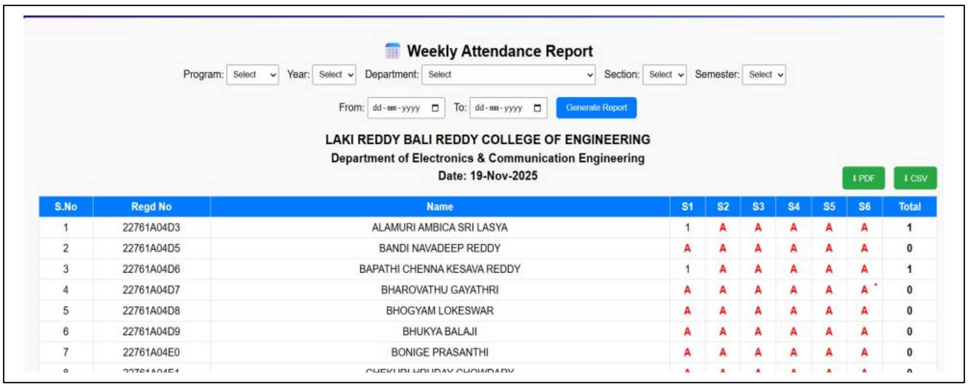

Figure 5.14. Weekly Attendance Report

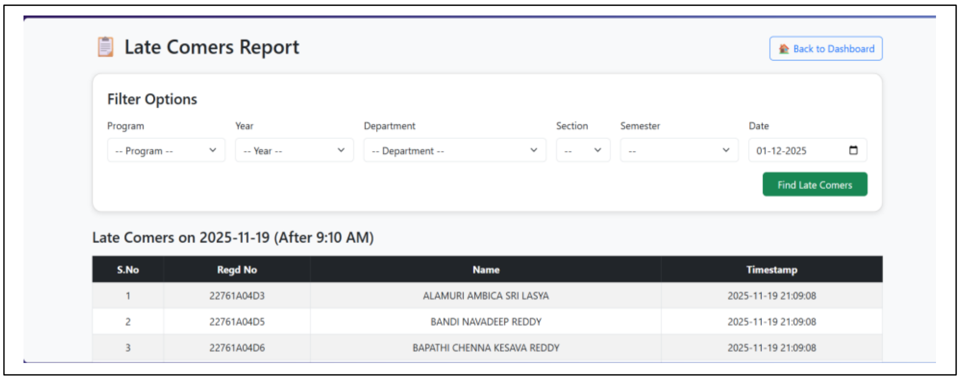

Figure 5.15. Late comers report

CHAPTER – VI

CONCLUSION&REFERENCES

The Smart RFID Attendance System with ESP8266 and web portal integration successfully demonstrates an efficient, modern, and automated method of tracking attendance. By combining RFID technology with Wi-Fi connectivity, the system enables quick, contactless identification and real-time data transmission to a centralized online platform. The use of the ESP8266 ensures reliable network communication, while the web portal provides an accessible interface for administrators to monitor records, generate reports, and manage user information remotely.

This project not only reduces manual errors and time consumption associated with traditional attendance systems but also enhances security, accuracy, and convenience. Its scalable design allows easy expansion in schools, offices, and industries. Overall, the system represents a practical and smart solution for digital attendance management in today's connected world.

REFERENCES

[1]. Joseph D. I., Emmalia A., and Akh F. (2018). RFID and IoT for Attendance Monitoring System. Materials Science, Engineering and Chemistry(MATEC) Web of Conferences, Vol 164, Published 23, April 2018.doi: 10.1051/matecconf/ 201816401020.

[2]. Ajay J., Aman A., Arpit S. and Poonam J. (2021). RFID-Based Attendance System. International Journal for Modern Trends in Science and Technology, Vol. 07, Issue 01, January 2021, pp.-40-43. doi: 10.46501/IJMTST0701009.

[3]. Ms.G.T. Bharathy, Ms.S. Bhavanisankari, T. Tamilselvi (2021). Smart Attendance Monitoring System Using IoT and RFID. International Journal of Advances in Engineering and Management(IJAEM). Vol. 03, Issue 06, June 2021, pp: 1307- 1313. doi: 10.35629/5252- 030613071313.

[4]. Hariom K. and Narayan V. (2022). RFIDBased Attendance Monitoring System Using ESP8266 Wi-Fi Module and IoT Enabled Mobile Application. NeuroQuantology, Vol. 20, Issue 09, September 2022.doi: 10.14704/nq.2022.20.9.NQ44681.

[5]. Prine Ana, Ukoette Jeremiah Ekah, Emmanuel Oyojta. (2022). IOT-based biometric attendance system for CRUTECH. International Journal of Science and Research Archive (IJSRA), Issue 29, January 2022.doi: 10.30574/ijsra.2022.5.1.0035.

[6]. Deki Juniasyah, Novi Herawadi, Danang Ade Muktiawan (2023). Design of A Laboratory Assistant Presence System Using Rfid Sensor and Web-Based Esp8266 Microcontroller. 9th Proceeding International Conference on Information Technology and Business (P-ICITB) Issue 2023, in Press.

[7]. Anurag Shrivastava, S. J. Suji Prasad, Ajay Reddy Yeruva, P. Mani, Pooja Nagpal & Abhay Chaturvedi (2023). IoT-Based RFID Attendance Monitoring System of Students using Arduino ESP8266 & Adafruit.io on Defined Area. Cybernetics and Systems: An International Journal, Vol. 03, Issue 10, October 2020.doi: 10.1080/01969722.2023.2166243