ವಿಶ್ವೇಶ್ವರಯ್ಯ ತಾಂತ್ರಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ, ಬೆಳಗಾವಿ

Visvesvaraya Technological University

“Jnana Sangama” Belagavi – 590018

Project Report

On

“PERSONNEL TRACKING SYSTEM IN AN INDOOR ENVIRONMENT”

Submitted in partial fulfilment for the award of degree of
Bachelor of Engineering
By

Bharath L	**1AT20EC023**
Bharath M	**1AT20EC024**
Hemanth N Ganji	**1AT20EC054**
Kavya K A	**1AT20EC071**

Under the Guidance
Of
Somesh B S
Assistant Professor
Department of Electronics & Communication Engineering,
Atria Institute of Technology – Bengaluru

Department of Electronics & Communication Engineering (NBA accredited)
Atria Institute of Technology (NAAC A++ Grade)
Bengaluru-560024
2023-24

Atria Institute of Technology

Bengaluru-560024

Department of Electronics & Communication Engineering

CERTIFICATE

Certified that the project work entitled **“PERSONNEL TRACKING SYSTEM IN AN INDOOR ENVIRONMENT”** carried out by

Mr. **Bharath L - 1AT20EC023**
Mr. **Bharath M - 1AT20EC024**
Mr. **Hemanth N Ganji - 1AT20EC054**
Ms. **Kavya K A -1AT20EC071**

bonafide students of VIII Semester Electronics & Communication Engineering Dept, Atria Institute of Technology, Bengaluru in partial fulfillment for the award of Bachelor of Engineering in Visvesvaraya Technological University, Belagavi during the year **2023-24**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Degree.

Prof. Somesh B S
Asst. Prof
Dept. of E&CE
ATRIA IT, Bengaluru

Dr. Jagadeesh H S
Prof. & Head,
Dept. of E&CE,
ATRIA IT, Bengaluru

Dr. Rajesha S
Principal,
ATRIA IT, Bengaluru

Name of the Examiners Signature with date

1.

2.

DECLARATION

We the students of fourth year, Bachelor of Engineering, Electronics and Communication Engineering, Atria Institute of Technology, Bengaluru-560024, hereby declare that the project work entitled ***"Personnel Tracking System in an Indoor Environment"*** has been carried out at **ATRIA INSTITUTE OF TECHNOLOGY** under the guidance of **Somesh B S** Assistant Professor, Department of Electronics and Communication Engineering, Atria IT, Bengaluru.

We declare that the work submitted in this report is our own and has not been previously submitted for the fulfilment of the B.E degree at the Visvesvaraya Technological University, Belagavi or any other Institution/University.

Place: Bengaluru

Bharath L

Bharath M

Hemanth N Ganji

Kavya KA

ACKNOWLEDGEMENT

The foundation for any successful venture is laid out not just by the individual accomplishing the task, but also by several other people who believe that the individual can excel and put in their every bit in every endeavor he/she embarks on, at every stage in life. And the success is derived when opportunity meets preparation, also supported by a well-coordinated approach and attitude. We express our thanks to **A S Kupparaju and Brothers Charitable Foundation Trust,** the Management of Atria Institute of Technology for giving me this opportunity.

We express our sincere gratitude to **Dr. Rajesha S**, Principal, Atria Institute of Technology, for providing us the required environment and for his valuable suggestion.

Our sincere thanks to **Dr. Jagadeesh H S**, Head of the Dept. Electronics and Communication Engineering, Atria Institute of Technology, for his valuable support and for rendering us resources for this project.

We express our sincere gratitude to **Dr. Keshavamurthy**, Project Co-ordinator Department of Electronics and Communication, Atria Institute of Technology, for providing us the required environment and for his valuable suggestion.

We are indeed indebted to **Assistant Prof. Somesh B S**, Internal Guide for the continued support, advice and valuable inputs during the course of this project work Last, but not the least we would like to thank my family, who has acted as a beacon of light throughout our lives.

Our sincere gratitude goes out to all our comrades and well-wishers who have supported us through all the ventures.

ABSTRACT

Whenever we need to locate someone in a mobile-prohibited area, it becomes challenging to find them within a short period. To address such challenges, the implementation of RFID (Radio Frequency Identification) technology emerges as a viable solution. Previous examinations and advancements in indoor localization have explored technologies like infrared and wireless LAN, among others. RFID, which stands for Radio Frequency Identification, is a fast and reliable means of identifying individuals. However, these methods are often plagued by issues such as limited accuracy and dependence on extensive infrastructure. This project proposes a method that enables indoor location tracking of staff and students by using RFID technology. RFID technology is often used to determine the location of personnel in a reasonable, power-efficient, and user-friendly manner. The indoor location tracking system allows you to track locations with the help of RFID tags and RFID readers in a non-intrusive manner.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER 1

INTRODUCTION

CHAPTER - 1

INTRODUCTION

1.1 BRIEF INTRODUCTION:

In today's institutions, the challenge of quickly locating staff or students when needed is a common issue. Traditional methods like announcements disrupt privacy and workflow. Surveillance cameras have limitations in crowded spaces. Human monitoring is expensive and unreliable. Centrex systems can also fall short. Therefore, institutions seek alternative location detection methods. GPS is useful outdoors but struggles indoor. Enter RFID (Radio-Frequency Identification) technology a discreet, cost-effective, and reliable solution. RFID tags, integrated into wearables, offer real-time tracking within defined areas. Combined with biometrics, it enhances security. RFID has transformed inventory and supply chain management. Now, it promises precise staff and student tracking. Each person wears a unique RFID tag, and the institution is mapped with RFID tracking units. Managed by a microcontroller and IoT communicator, this system guarantees accuracy and reliability without compromising privacy or operations.

1.2 RFID:

RFID is a wireless communication technology. In RFID technology, there are two roles: information carrier called RFID tag and information collector named RFID reader. A RFID tag is composed of two parts: a chip and an antenna. Data is stored in the chip and transferred via the antenna. Protective material like plastics can be placed around the RFID tag to protect it from being damaged. RFID reader is a circuit with external power input. It also produces radio waves actively to communicate with nearby RFID tags. Reader can read information from and write information to RFID tag in a various distance, which ranges from 10 centimeters to 200 meters and depends on the power of both RFID tag and reader. Because of this feature, there are plenty of practical usage of RFID tag and RFID reader. One typical usage is supply chain management, where people can take use of RFID tags attached to goods to track their trace. Those trace information helps people to optimize the supplier, storage and logistics of goods and maximize profit eventually. Second typical of RFID tag is access control.

1.3 WIRELESS LOCAL AREA NETWORK (WLAN):

WLAN is a wireless network at home or office that allows devices to connect to local area network and internet within a limited range. This network allows movement of devices and still connected to the network and internet. Most WLANs based on IEEE 802.11 standards and commonly known as Wi-Fi. Installation and maintenance of WLAN is simple and it allows access to network while moving for users. Therefore, WLAN is popular at home as well as businesses.

1.4 MOBILE PHONE APPLICATION:

Smart phones and mobile phone application are very popular all over the world. Nowadays, the popularity and effectiveness of new system increases as system integrates with mobile phone and application. The scope of the project is not to develop any mobile application but it is an integral part of the proposed system. In this project, we put forward a design of smart parking and vehicle finder based on hybrid WLAN-RFID and mobile phone application. Nowadays in many public places such as malls, multiplex systems, hospitals, offices, market areas there is a crucial problem of car parking. The car-parking area has many lanes/slots for car parking. So to park a car one has to look for all the lanes. Moreover, this involves a lot of manual labor and investment. So, there is a need to develop an automated parking system that indicates directly the availability of vacant parking slots in any lane right at the entrance. It involves a system including infrared transmitter- receiver pair in each lane and a display outside the car parking gate. So the person desirous to park his vehicle is well informed about the status of availability of parking slot. Conventional parking systems do not have any intelligent monitoring system and the parking lots are monitored by security guards. A lot of time is wasted in searching vacant slot for parking and many times it creates jams. Conditions become worse when there are multiple parking lanes and each lane with multiple parking slots. Use of parking management system would reduce the human efforts and time with additional comfort. In the proposed system, the display unit displays a visual representation of the parking and it shows the emptyand occupied slots which help the user to decide where to park their car. The system would not only save time but the software and hardware would also manage the check in and check-outs of the cars under the control of RFID readers/ tags with additional features of automatic billing, Entry exit data logging. The users go through a onetimeregistration process where there are asked to fill in their personal details and an accountis created for them, this account has information about them and also has money in it which they can recharge at kiosks present in the vicinity. In this system, the users are guided to the vacant slot for parking using Video Displays at the entrance of the parking floor, these displays show a visual representation of the parking lot with emptyand occupied slots which are green and red respectively. The user is provided with a tag which he receives on registration, this tag is linked with his prepaid account and includes his personal information, and this tag uses Radio Frequency identification (RFID) technology and is placed on the top of the user's windshield. The parking charges are automatically deducted from the user's account based on the time spent inside the parking area.

1.5 SCOPE OF WORK:

The scope of work for the project involves design and implement a comprehensive system for tracking individuals within indoor spaces. By conducting a thorough literature review, gathering user requirements, and employing advanced system design principles, the project will develop a prototype capable of accurately monitoring personnel movements in real-time. Through rigorous testing and evaluation, the system's effectiveness, efficiency, and usability will be assessed, with the findings documented and presented in a comprehensive report. This project not only addresses the immediate need for indoor tracking solutions but also lays the groundwork for future enhancements and innovations in the field.

1.6 PROBLEM STATEMENT:

Develop an indoor human tracking system that continuously and accurately determines the real-time positions of individuals within various indoor environments like hospitals or colleges.

1.7 AIM:

To implement the personnel tracking system for tracking an individual in an indoor premise.

1.8 OBJECTIVES:

1. Real-Time Location Tracking: Enable real time tracking of personnel within a specific area.
2. Reducing Manual Intervention: To minimize the reliance on manual data entry and paper-based tracking methods. Traditional systems can be time-consuming, prone to errors, and lack real-time updates. By automating data collection and location tracking, you can streamline processes, improve data accuracy, and free up staff for more strategic tasks.
3. To design alert system using Wi-fi system by sending person information to the Mobile. This emphasizes leveraging your existing Wi-Fi infrastructure to create an alert system. By integrating Wi-Fi with your tracking system, you can trigger real-timenotifications on mobile devices when specific events occur. For example, alerts can besent when personnel enter or leave designated areas, encounter safety hazards, or require assistance. This can enhance communication, improve response times, and promote overall safety within our designated zone.

CHAPTER 2

LITERATURE SURVEY

CHAPTER - 2

LITERATURE SURVEY

Paper Title	Authors	Observation
[1]. “*RFID based Indoor Tracking System”* International Engineering Journal for Research & Development. Vol 4, Issue 5.	Anand C Shreyash B Kaushik B Dept of CSE, PRMITR, Amravati, India	A reader and few tags in general are of very littleuse unless, it is connected with a powerful backend. Simply retrieving a serial number does not provide much information. A backend processing unit provides the additional information such as when and where the tag was scanned.
[2]. *“VLC Based Indoor Tracking System with IOT Network for Smart Buildings.”* International Engineering Journal for Research & Development. ISSN: 2321- 0869 (O) 2454-4698 (P), Volume-10, Issue-4, April 2020	S T Lakshmi Bhavyashree Chittamuru Asrija Deepthi J Dept of ECE Amity University, India (April 2020)	This paper aims to develop secure and reliable indoor tracking system. This is designed based on VLC and IOT. Our system has three modules, Transmitter module, Receiver module, IoT module. Transmitter module is with the person it has IR transmitter to transmit the information about that person. Receiver module is with the interior wall mount this get the information from the transmitter and transmit to IoT server. IoT module receives the information from receiver module and stored as a log.

[3]. *“Indoor Location Tracking System Using RFID Technology*” International Engineering Journal for Research & Development. Vol. 3, Issue1, pp: (73-80), Month: January - March 2015.	P Karthika J Harriet Rathna PriyaA Rathinavel Pandian Dept of ECE, Mother Theresa College of Engineering, Tuticorin, India	LCD Display is the output device in the proposed system. The method uses a 20/4 display unit. It simply displays the space to enter the code of the staff needed and then displays the name, designation of staff for conformation and the current location of the staff. It can be further developed by using a computer monitor, which can display further more details.
[4]. *“RFID Scanned Multipurpose tracking system using IoT”* International Engineering Journal for Research & Development. Vol. 4, 2021	Ahamed Irfaan S Arun Kumar K Gnanapraveenkant Dept of ECE, AVS College of Enginerring, Tamil Nadu.	Here, NoSQL database is used for backend storage. Using this as a reference, the implementation of the data storage will be done using any other cloud.
[6]. *“RFID Based Tracking Monitoring System”* IIMT College of Engineering, Greater Noida, U.P., India	Pradeep Kumar, Basanta Mahato Department of Electronics and Communication Department	This system is for tracking or monitoring the students/employees within the campus. Every student/employee will have a RFID card. Each RFID card will be registered on the name of the individual student/employee. A centralized server system will be formed which will keep the record of the students, at what time where is the current location of the student within the campus.

[6]. *“Design and Implementation of RFID based Staff Monitoring System”* Chalapathi Institute of Engineering and Technology, Guntur, AP	M. Sirisha, G. Assistant Professor Prathyusha , N.Vidya,K.Lalith Sai Kumar	In current days, there have been rapid growth in the number of wireless applications based on Radio Frequency Identification (RFID) systems. In this paper, design and implementation of staff monitoring system using RFID technology. The main objective of RFID technology is to monitoring staff attendance accurately and effectively. The attendance system plays a major role in schools and colleges is documented manually. But, this process requires lots of time. RFID Technology is used in schools, colleges, office and stations for different purposes to automatically keep a track of staff. The system also be developed by using GSM (Global System for Mobile communication) technology.
[7]. *“The use of RFID Technology in the patient tracking system”* Master’s Degree Programme in Welfare Technology	Aminullah Nuuri Master’s Thesis	RFID has been deployed in a variety of sectors, including healthcare, however the acceptance and usage of RFID remain a concern. Despite its promise to increase healthcare provider efficiency and patient safety, there are several drawbacks to its use. This thesis was conducted as a scoping review. The process for collecting and analyzing the data was done following the protocols described in literature. Literature searches were conducted using electronic databases

[8]. *"RFID based Tracking System"* International Journal of Advanced Research in Computer and Communication Engineering	Prof. Gaurav G. Narkhede, Akshay Kishor Langade, Kinal Gaurang Mehta, Shubhankar Abhay Kulkarni. Assistant Professor, Electronics and Telecommunications Engineering, MIT COE, Pune, India 1	Radio Frequency Identification (RFID) is coming, and it's bringing a streamlined revolution. When dealing with the tracking device, Radio Frequency Identification (RFID) is the latest phase in the decades, that can be used as an efficient tracker. The development of Tracking System using Radio Frequency Identification (RFID) Technology is Quite new but something that promising. This solution uses RFID technique for monitoring entry and exit of employees with their official assets (E.g. laptops). The Tracking System actually based on external database system that will provide the recorded information about the reader. Since the reader detected by the database, then the tracking system will process the data and will show the result of subject tracking.

CHAPTER 3

HARDWARE AND SOFTWARE USED

CHAPTER - 3

HARDWARE AND SOFTWARE USED

3.1 HARDWARE USED:

1. Arduino UNO:

Arduino UNO is a microcontroller board based on theATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again. "UNO" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

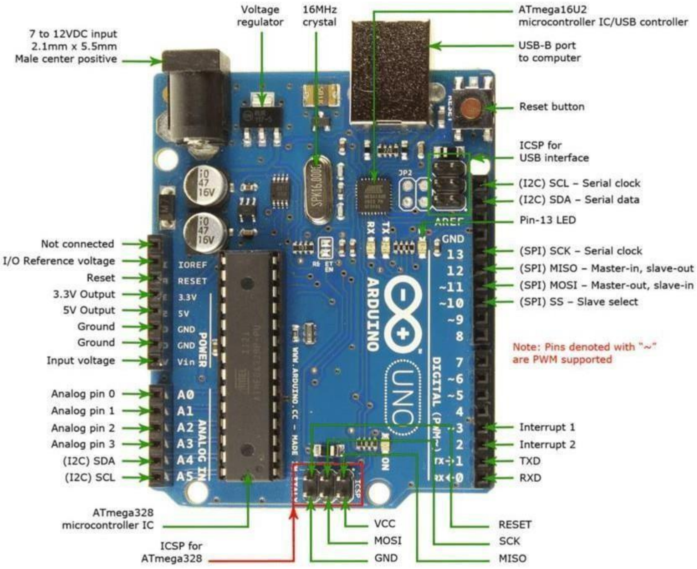


Figure: 3.1.1 Arduino UNO

- **Arduino UNO Specifications:**
  - Microcontroller: ATmega328P
  - Operating Voltage: 5V
  - Input Voltage (recommended): 7-12V
  - I/0 Voltage (limit): 6-20V
  - Digital I/O Pins: 14 (of which 6 provide PWM output)
  - PWM Digital I/O Pins: 6
  - Analog Input Pins: 6
  - DC Current per I/O Pin: 20 mA
  - DC current for 3.3V Pin: 50 mA
  - Flash Memory: 32 KB (ATmega328P) of which 0.5 Kb used by bootloader
  - SRAM: 2 KB (ATmega328P)
  - EEPROM: 1 KB (ATmega328P)
  - Clock Speed: 16 MHz
  - LED-BUILTIN: 13
- **Arduino programming:**

  The Arduino UNO can be programmed with the IDE (Integrated Development Environment). Select Arduino UNO from the Tools Board menu (according to the microcontroller on board). The ATmega328 on the Arduino UNO comes preprogrammed with a boot loader that allows us to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. We can also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

  - On Rev1 boards: connecting the solder jumper on the back of the board (near themap of Italy) and then reusing the 8U2.
  - On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.
- **Warnings:**

  The Arduino UNO has a resettable poly fuse that protects your computer's USB ports from shorts and over current. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

- **Differences with other boards:**

  UNO differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

- **Power:**

  The Arduino UNO board can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and VIN pin headers of the POWER connector. The board can operate on an external supply from 6 to 20 volts. If supplied with less than7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may over heat and damage the board. The recommended range is 7 to 12volts.

  The power pins are as follows:

  - Vin: The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). One can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
  - 5V This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the5V or 3.3V pins bypasses the regulator, and can damage your board.
  - 3.3volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
  - GND: Ground pins.
  - IOREF: This pin on the Arduino board provides the voltage reference with which the microcontroller operates. Configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

- **Memory:**

The ATmega328 has 32 KB (with 0.5 KB occupied by the boot loader). It also has2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

- **Input & Output:**

Each of the 14 digital pins on the Uno can be used as an input or output, using pin mode (), digital write (), and digital read () functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull- up resistor (disconnected by default) of 20-50k ohm. Maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller.

In addition, some pins have specialized functions:

- Serial 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach interrupt () function for details.
- PWM 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog write () function.
- SPI 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- LED 13 There is a built-in LED driven by digital pin 13. When the pin is HIGHvalue, the LED is on, when the pin is LOW, it's off.
- TWI A4 or SDA pin and A5 or SCL pin. Support TWI communication using theWire library.
- The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog reference () function.

- There are a couple of other pins on the board:
- AREF Reference voltage for the analog inputs. Used with analog Reference.
- Reset Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

- **Communication:**

Arduino UNO has a number of facilities for communicating with a computer, another Arduino board, or other microcontrollers. The ATmega328provides UART TTL (5V)serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an inffile is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and1).

A Software serial library allows serial communication on any of the Uno's digital pins. The ATmega328alsosupports I2C(TWI) and SPI communication. The Arduino Software (IDE) includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

- **Automatic (Software)Reset:**

Rather than requiring a physical press of the reset button before an upload, the Arduino UNO board is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nano farad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino Software (IDE) uses this capability to allow you to upload code by simply pressing the upload button in the interface toolbar. This means that the boot loader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the boot loader is running on the Uno. While it is programmed to ignore malformed data (i.e., anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno board contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110ohm resistor from 5V to the reset line.

2. RFID Reader:

Radio frequency identification (RFID) technology is a wireless communication technology that enables users to uniquely identify tagged objects orpeople. The RFID technology is a means of gathering data about a certain item without the need of touching or seeing the data carrier,through the use of inductive coupling or electromagnetic waves.

An RFID system has four basic components:

- A tag which is composed of a semiconductor chip and antenna.
- An interrogator (sometimes called a read/write device), which is composed ofan antenna, a RF electronics module, and a control electronics module.
- A controller (sometimes called a host), which most often takes the form of a PC or a workstation running database and control (often called middleware) software.
- An antenna, which converts electrical power to RF power.

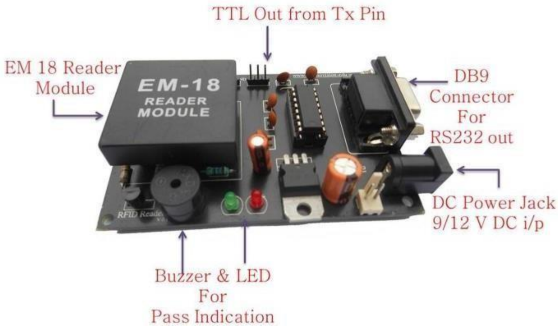

TTL Out from Tx Pin
EM 18 Reader Module
EM-18 READER MODULE
DB9 Connector For RS232 out
DC Power Jack 9/12 V DC i/p
Buzzer & LED For Pass Indication

Figure: 3.1.2 RFID Reader

- **RFID Reader specifications:**
  - Operating frequency: 125kHz
  - It has an on-chip antenna
  - Power supply: 5V
  - Power-up the module and connect the transmit pin of the module to receive pin ofyour microcontroller.
  - Show RFID card within the reading distance and the card number is thrown at theoutput.

4. RFID Tag:

The basic function of an RFID tag is to store data and transmit data to the interrogator (Reader). At its basic, a tag consists of an electronics chip and an antenna encapsulated in a package to form a usable tag, such as a packing label that might be attached to a box. Generally, the chip contains memory where data may be stored and read from and sometimes written. Some tags also contain batteries, and this is what differentiates active tags from passive tags. In our project we use passive tag.

Figure: 3.1.3 RFID Tag

5. Node MCU:

- **Node-MCU ESP8266 Specifications:**
  - Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
  - Operating Voltage: 3.3V
  - Input Voltage: 7-12V
  - Digital I/O Pins (DIO): 16
  - Analog Input Pins (ADC): 1
  - UARTs: 1

- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna

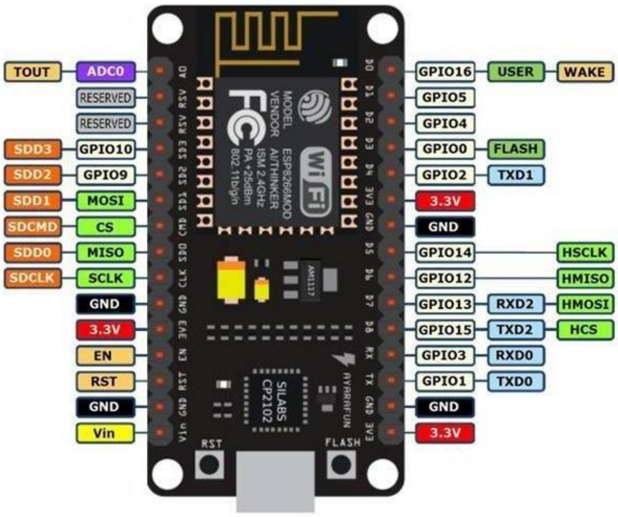


Figure: 3.1.4 Node MCU

- **Pin Description:**

Table 1: Pin Description of NodeMCU

Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	Micro-USB**:** NodeMCU can be powered through the USB port 3.3V**:** Regulated 3.3V can be supplied to this pin to power the board GND**:** Ground Vin**:** External Power Supply
Control Pins	EN, RST	The pin and the button resets the microcontroller

Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 to GPIO16	NodeMCU has 16 general purpose input-output pins on its board
SPI Pins	SD1, CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
I2C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

5. LCD Display:

A **liquid-crystal display** (**LCD**) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in color or monochrome. LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and 7-segment displays, as in a digital clock. They use the same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have larger elements. LCD is used in wide range application including computer monitors, televisions, instrument panels, aircraft cockpit displays, and indoor and outdoor signage. Small LCD screens are common in portable consumer devices such as digital cameras, watches, calculators, and mobile telephones, including smart phones. LCD screens are also used on consumer electronics products such as DVD players, video game devices and clocks. LCD screens have replaced heavy, bulky cathode ray tube (CRT) displays in nearly all applications. LCD screens are available in a wider range of screen sizes than CRT and plasma displays, with LCD screens available in sizes ranging from tiny digital watches to huge, big- screen television sets. Since LCD screens do not use phosphors, they do not suffer image burn-in when a static image is displayed on a screen for a long time (e.g., the table frame for an aircraft schedule on an indoor sign). LCDs are, however, susceptible to image persistence.

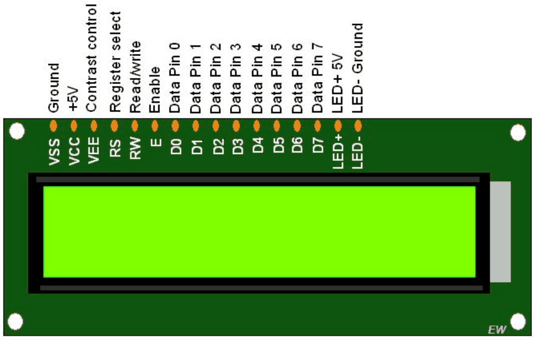


Figure: 3.1.5 LCD Display

- **LCD Specifications:**
  - Brightness (Luminance): measured in millicandles of luminance per square meter, also known as “NITs”.
  - Backlight Longevity: measured in thousands of hours.
  - Temperature Range: Colour Depth
  - Resolution: expressed by the number of columns and rows of pixels.

6. Buzzer:

A **buzzer** is a small yet efficient component to add sound features to our project/system. It is very small and compact 2-pin structure hence can be easily used on breadboard, Perf Board and even on PCBs which makes this a widely used component in most electronic applications. There are two types are buzzers that are commonly available. The one shown here is a simple buzzer which when powered will make a Continuous Beep. sound, the other type is called a readymade buzzer which will look bulkier than this and will produce a Beep. Beep. Beep. Sound due to the internal oscillating circuit present inside it. But the one shown here is most widely used because it can be customized with help of other circuits to fit easily in our application. This buzzer can be used by simply powering it using a DC power supply ranging from 4V to 9V. A simple 9V battery can also be used, but it is recommended to use a regulated +5V or +6V DC supply. The buzzer is normally associated with a switching circuit to turn ON or turn OFF the buzzer at required time and require interval.

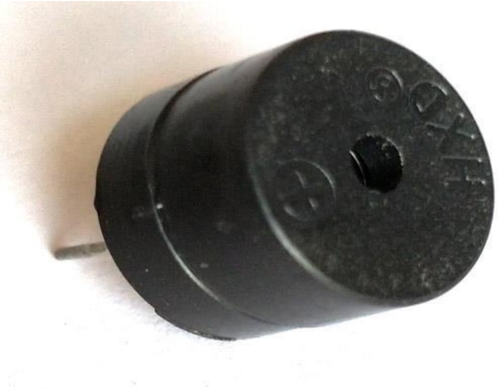


Figure 3.1.6 Buzzer

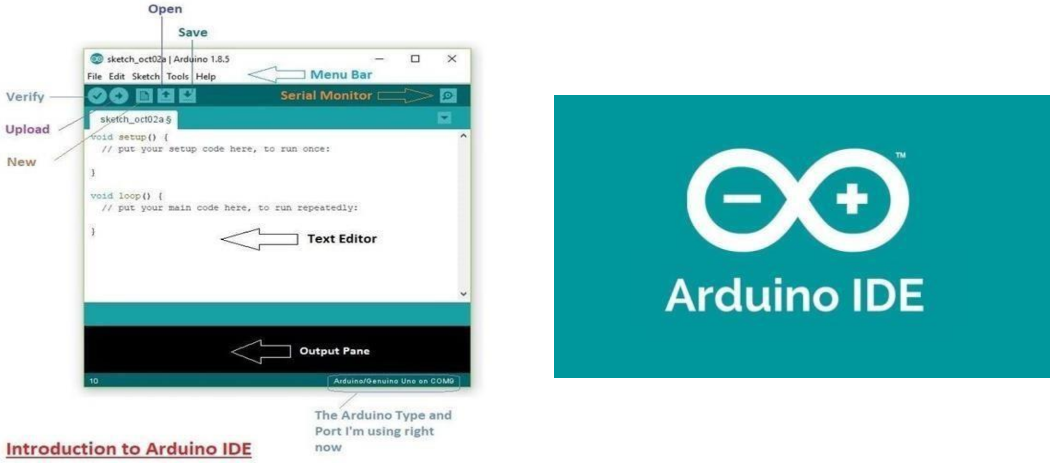
- **Pin Description:**

Table 2: Pin description of Buzzer

Pin Number	**Pin Name**	**Description**
1.	Positive	Identified by (+) symbol or longer terminal lead. Can be powered by 6V DC
2.	Negative	Identified by short terminal lead. Typically connected to the ground of the circuit

3.2 SOFTWARE USED:

1. Arduino IDE:


Open
Save
sketch_oct02a | Arduino 1.8.5
File Edit Sketch Tools Help
Menu Bar
Verify
Upload
New
Serial Monitor
sketch_oct02a §
Text Editor
Output Pane
10
Arduino/Genuino Uno on COM9
The Arduino Type and Port I'm using right now
Introduction to Arduino IDE
Arduino IDE
TM

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Figure: 3.2.1 Arduino IDE

The Arduino integrated development environment (IDE) is a cross-platform application for Windows, macOS, Linux that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards. The Arduino IDE supports the languages C and C++ using special rules of code structuring. It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.

The IDE environment is mainly distributed into three sections:

- Menu Bar
- Text Editor
- Output Panel

Arduino IDE (Integrated development Environment) is fully developed into functionality of full of libraries, as long as programming the Arduino UNO in Embedded C language is possible because Arduino IDE can compile both Arduino code as well as AVR standard code.

- When designing software for a smaller embedded system with the 8051, it is very common place to develop the entire product using assembly code. With many projects, this is a feasible approach since the amount of code that must be generated is typically less than 8 kilobytes and is relatively simple in nature. If a hardware engineer is tasked with designing both the hardware and the software, he or she will frequently be tempted to write the software in assembly language.
- The trouble with projects done with assembly code can is that they can be difficult to read and maintain, especially if they are not well commented. Additionally, the amount of code reusable from a typical assembly language project is usually very low. Use of a higher-level language like C can directly address these issues. A program written in C is easier to read than an assembly program.
- Since a C program possesses greater structure, it is easier to understand and maintain. Because of its modularity, a C program can better lend itself to reuse of code from project to project. The division of code into functions will force better structure of the software and lead to functions that can be taken from one project and used in another, thus reducing overall development time. A high order language such as C allows a developer to write code, which resembles a human's thought process more closely than does the equivalent assembly code. The developer can focus more time on designing the algorithms of the system rather than having to concentrate on their individual implementation. This will greatly reduce development time and lower debugging time since the code is more understandable.
- By using a language like C, the programmer does not have to be intimately familiar with the architecture of the processor. This means that someone new to a given processor can get a project up and running quicker, since the internals and organization of the target processor do not have to be learned. Additionally, code developed in C will be more portable to other systems than code developed in assembly. Many target processors have C compilers available, which support ANSI C.
- All of this is not to say that assembly language does not have its place. In fact, many embedded systems (particularly real time systems) have a combination of C

and assembly code. For time critical operations, assembly code is frequently the only way to go. One of the great things about the C language is that it allows you to perform low-level manipulations of the hardware, if need be, yet provides you the functionality and abstraction of a higher order language.

- In comparison to assembly code, C programs exhibit a higher degree of structure, rendering them more comprehensible and easier to maintain. This inherent modularity of C facilitates code reuse across different projects. By breaking down code into functions, C encourages better software architecture, thereby allowing functions to be easily transplanted from one project to another, thus streamlining development processes. Moreover, the syntax of C closely mirrors human thought processes, enabling developers to focus on designing algorithms rather than getting bogged down in low-level implementation details. Consequently, this not only accelerates development but also reduces debugging time, as the code is more intuitive and understandable.

CHAPTER 4

IMPLEMENTATION

CHAPTER - 4

IMPLEMENTATION

4.1 Circuit diagram:

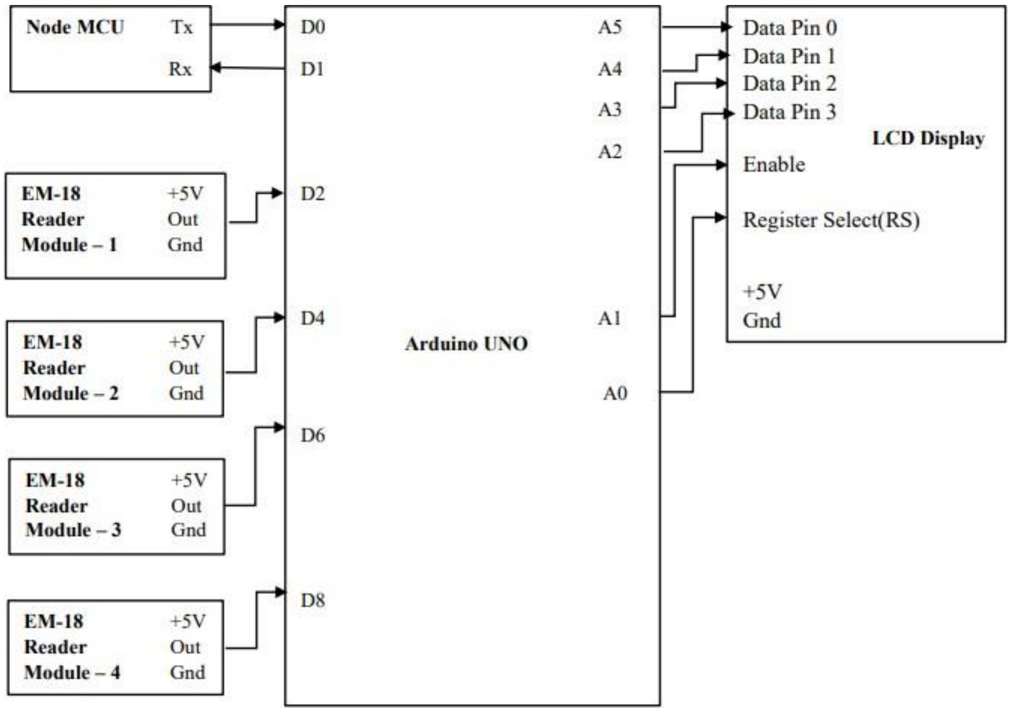


Figure: 4.1.1 Circuit Diagram

For implementing RFID-based indoor tracking using Arduino, NodeMCU, and integrating Telegram for notification purposes:

1. **Hardware Setup:**
   - Obtain RFID reader modules compatible with Arduino.
   - Acquire Arduino and NodeMCU boards.
   - Connect the RFID reader module to the Arduino board. Typically, this involves connecting power, ground, and data pins.
   - Connect the Arduino and NodeMCU boards via serial communication (e.g., USB).
   - Ensure NodeMCU is connected to the internet, either via Wi-Fi or Ethernet.

2. **RFID Tagging:**
    - Assign unique RFID tags to objects or individuals you want to track.
    - Attach RFID tags to the objects or individuals.

3. **Arduino Programming:**
    - Write Arduino code to read RFID tag IDs using the RFID reader module.
    - Store the RFID tag IDs in variables.
    - Utilize appropriate libraries to interface with the RFID reader module.

4. **NodeMCU Programming:**
    - Write NodeMCU code to communicate with Arduino over serial.
    - Receive RFID tag IDs from Arduino.
    - Implement logic for tracking movement or presence based on RFID tag IDs.
    - Integrate Telegram API for sending notifications. You'll need to:
    - Obtain a Telegram Bot API token and chat ID.
    - Use an appropriate library (e.g., "UniversalTelegramBot") to send messages to Telegram.

5. **Integration:**
    - Establish communication between Arduino and NodeMCU over serial.
    - Define protocols for sending RFID tag IDs from Arduino to NodeMCU.
    - Implement logic on NodeMCU to trigger Telegram notifications based on RFIDtag IDs.
    - Ensure NodeMCU has access to the internet for sending Telegram messages.

6. **Testing and Debugging:**
    - Test the system in a controlled environment to ensure RFID readings are accurate.
    - Verify that NodeMCU receives RFID tag IDs correctly from Arduino.
    - Test the Telegram integration by sending test messages.
    - Debug any issues encountered during testing.

7. **Deployment:**
    - Mount the hardware components in the indoor where tracking is required.
    - Ensure proper power supply for Arduino and NodeMCU.
    - Conduct final tests to ensure the system operates as expected in the real environment.

8. **Maintaining and Monitoring:**
    - Regularly monitor the system for any issues.
    - Update software as necessary to fix bugs or add new features.
    - Replace RFID tags or hardware components if they malfunction.

9. **Telegram Bot creation and Notification enabling:**

➢ Set up a Telegram Bot:

- Create a Telegram bot using BotFather.
- Obtain the bot token provided by BotFather.
- Note down the chat ID to which you want to send notifications. You can use your own chat ID or create a group and add your bot to it to obtain the group chat ID.

➢ Install necessary libraries:

- Use the Arduino IDE or Platform IO to program your Node MCU.
- Install libraries such as ESP8266WiFi for connecting to the internet and ArduinoJson for handling JSON data.

➢ Connect Node MCU to the internet:

- Connect your Node MCU to your local Wi-Fi network using the ESP8266 Wi-Filibrary.

➢ Send HTTP requests to Telegram API:

- Use the Wi-Fi Client library to create an HTTP client.
- Craft HTTP requests to send messages to the Telegram bot API. You'll need tomake POST requests to the Telegram Bot API endpoint.

➢ Handle responses and errors:

- Handle responses from the Telegram API appropriately.
- Implement error handling in case of network issues or API errors.

CHAPTER 5

RESULTS

CHAPTER – 5

RESULTS

5.1 Result: Image

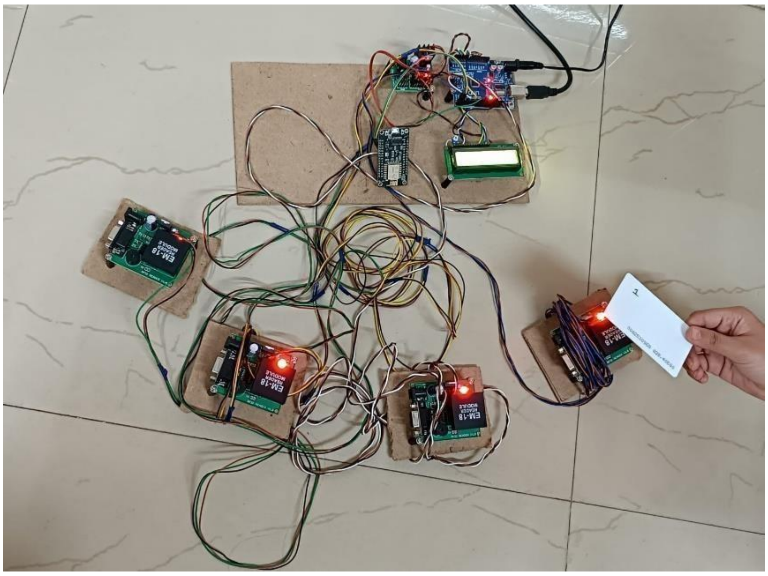


Figure 5.1.1: Tapping RFID card to Room1

When Person 1 taps their RFID card and enters Room 1, they cannot enter another room until they exit Room 1. Upon entry or exit from any room, a notification is sent via Telegram. The system maintains a state for each person to track their current room occupancy status. If Person 1 attempts to enter another room while already inside a room, entry is denied, and they are instructed to exit the current room first. Error handling is implemented to address invalid card taps or communication failures. In the multi-room scenario, each individual, including Person 1 through Person 4, is subject to the same access control rules when tapping their RFID card. The system maintains a dedicated state for each person to track their room occupancy status across all four rooms. Upon entering or exiting any room, the system ensures that individuals adhere to the rule prohibiting entry into another room without first exiting their current room.

5.2 Real Time Working Model Outcomes:

i. Case 1:

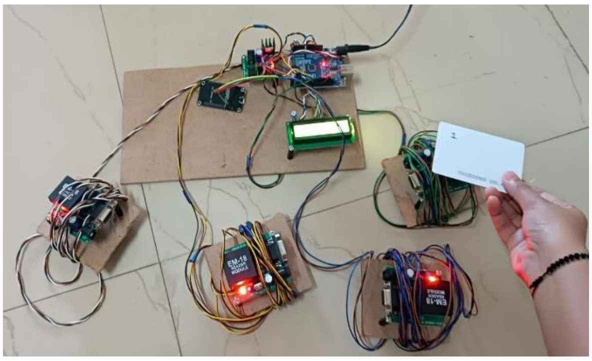


Figure 5.2.1: Entering Room1

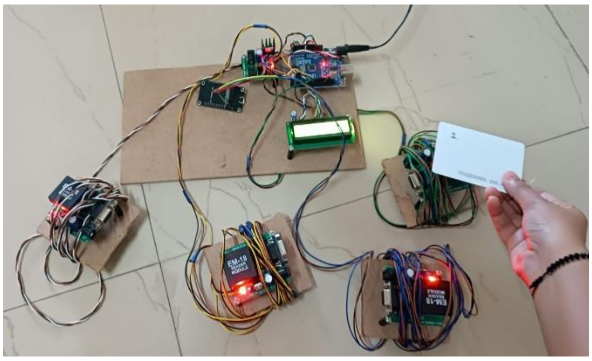


Figure 5.2.2: Exiting Room1

- When an individual taps their RFID tag to enter Room 1, the system registers their entry into that room. The system displays the presence of the individual in Room1.
- In order for the individual to exit Room 1, they need to tap their RFID tag again to signal their departure.
- Without tapping the RFID tag to exit, the system does not allow the individual to move to Room 2, 3, or 4.
- After successfully exiting Room 1 by tapping their RFID tag, the individual can then proceed to tap their RFID tag to enter any of the other rooms (Room 2, 3, or 4).
- Each time they enter a new room, their entry is registered by the system, and their presence is displayed in the respective room.
- Similar to exiting Room 1, the individual must tap their RFID tag to signal their departure from Room 2, 3, or 4.
- Without tapping the RFID tag to exit, the system prevents the individual from entering any other rooms or moving within the indoor environment.

ii. Case 2:

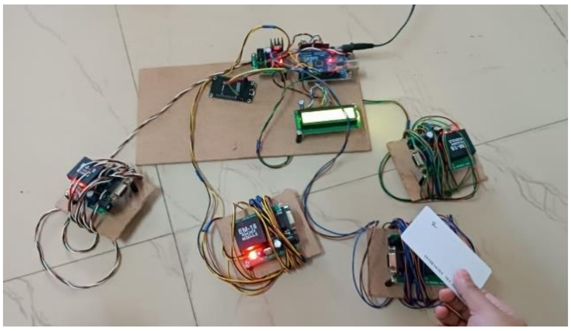


Figure 5.2.3: Entering Room2

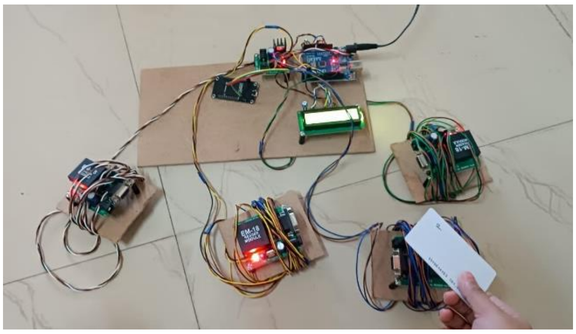


Figure 5.2.4: Exiting Room2

- When an individual taps their RFID tag to enter Room 2, the system registers their entry into that room. The system displays the presence of the individual in Room 2.
- In order for the individual to exit Room 2, they need to tap their RFID tag again to signal their departure.
- Without tapping the RFID tag to exit, the system does not allow the individual to move to Room 1, 3, or 4.
- After successfully exiting Room 2 by tapping their RFID tag, the individual can then proceed to tap their RFID tag to enter any of the other rooms (Room 1, 3, or 4).
- Each time they enter a new room, their entry is registered by the system, and their presence is displayed in the respective room.
- Similar to exiting Room 2, the individual must tap their RFID tag to signal their departure from Room 1, 3, or 4.
- Without tapping the RFID tag to exit, the system prevents the individual from entering any other rooms or moving within the indoor environment.

iii. Case 3:

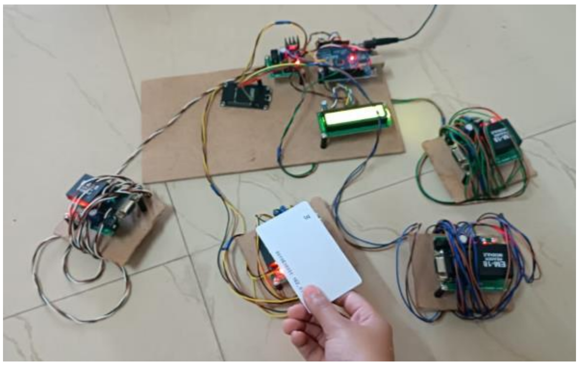


Figure 5.2.5: Entering Room3

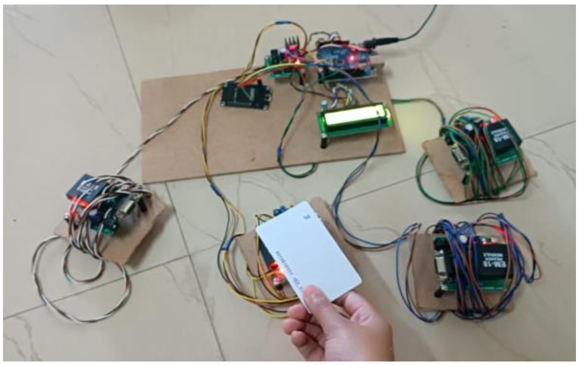
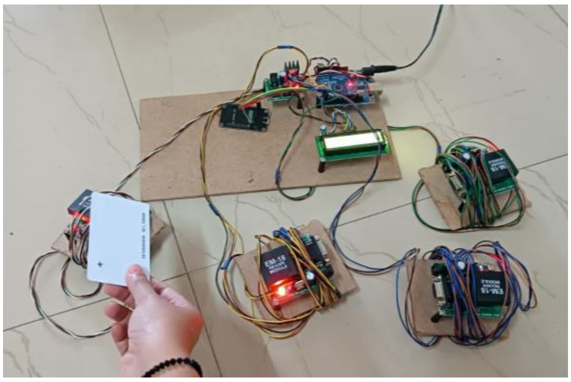


Figure 5.2.6: Exiting Room3

- When an individual taps their RFID tag to enter Room 3, the system registers their entry into that room. The system displays the presence of the individual in Room 3.
- In order for the individual to exit Room 3, they need to tap their RFID tag again to signal their departure.
- Without tapping the RFID tag to exit, the system does not allow the individual to move to Room 1,2 or 4.
- After successfully exiting Room 3 by tapping their RFID tag, the individual can then proceed to tap their RFID tag to enter any of the other rooms (Room 1,2 or 4).
- Each time they enter a new room, their entry is registered by the system, and their presence is displayed in the respective room.
- Similar to exiting Room 3, the individual must tap their RFID tag to signal their departure from Room 1,2 or 4.
- Without tapping the RFID tag to exit, the system prevents the individual from entering any other rooms or moving within the indoor environment.

iv. Case 4:



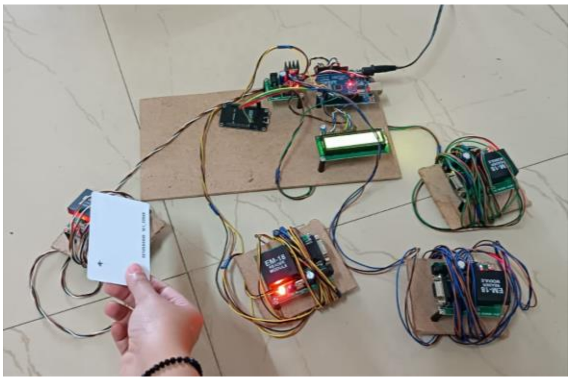


Figure 5.2.7: Entering Room4

Figure 5.2.8: Exiting Room4

- When an individual taps their RFID tag to enter Room 4, the system registers their entry into that room. The system displays the presence of the individual in Room 4.
- In order for the individual to exit Room 4, they need to tap their RFID tag again to signal their departure.
- Without tapping the RFID tag to exit, the system does not allow the individual to move to Room 1,2 or 3.
- After successfully exiting Room 4 by tapping their RFID tag, the individual can then proceed to tap their RFID tag to enter any of the other rooms (Room 1,2 or 3.).
- Each time they enter a new room, their entry is registered by the system, and their presence is displayed in the respective room.
- Similar to exiting Room 4, the individual must tap their RFID tag to signal their departure from Room 1,2 or 3.
- Without tapping the RFID tag to exit, the system prevents the individual from entering any other rooms or moving within the indoor environment.

The system displays entry and exit messages on the central microcontroller's LCD display to indicate when individuals enter or exit rooms. Upon a successful RFID tap, the LCD promptly updates with corresponding entry or exit notifications. This real- time feedback enables efficient monitoring of individuals' movements between rooms. By displaying such messages, the system ensures clear communication of room occupancy status to observers, facilitating smooth and organized access control.

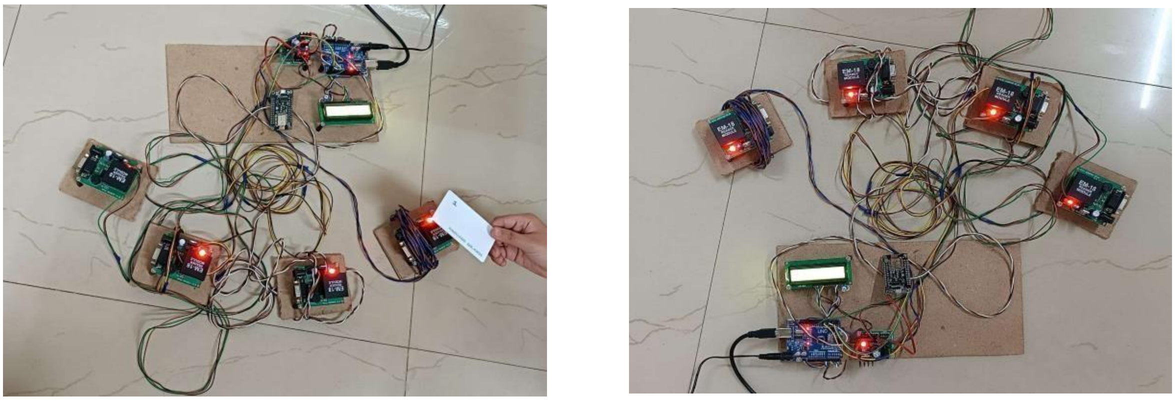
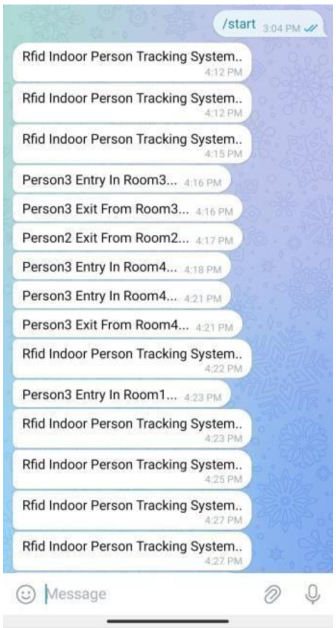


Fig 5.2.9: Different Individual entering and exiting different Room

5.1.1.1 Result: Image 3



/start 3:04 PM

Rfid Indoor Person Tracking System.. 4:12 PM

Rfid Indoor Person Tracking System.. 4:12 PM

Rfid Indoor Person Tracking System.. 4:15 PM

Person3 Entry In Room3... 4:16 PM

Person3 Exit From Room3... 4:16 PM

Person2 Exit From Room2... 4:17 PM

Person3 Entry In Room4... 4:18 PM

Person3 Entry In Room4... 4:21 PM

Person3 Exit From Room4... 4:21 PM

Rfid Indoor Person Tracking System.. 4:22 PM

Person3 Entry In Room1... 4:23 PM

Rfid Indoor Person Tracking System.. 4:23 PM

Rfid Indoor Person Tracking System.. 4:25 PM

Rfid Indoor Person Tracking System.. 4:27 PM

Rfid Indoor Person Tracking System.. 4:27 PM

Figure 5.3.1: Notification in Telegram through NodeMCU

The Telegram bot provides comprehensive notifications detailing the entry and exit data of all individuals within the premises. Each time someone enters or exits a room, the bot promptly sends a message containing relevant information, including the individual's identity and the room they entered or exited. By aggregating this data, the bot offers a comprehensive overview of all movements, facilitating real-time monitoring and access control. These notifications enable efficient tracking of individuals' activities throughout the premises, enhancing security and management capabilities. Moreover, the Telegram platform's accessibility ensures that stakeholders can receive these updates promptly, regardless of their location. This centralized system of notifications empowers administrators to maintain a vigilant and responsive approach to premises management.

CHAPTER 6

APPLICATIONS, ADVANTAGES AND DISADVANTAGES

CHAPTER - 6

APPLICATIONS, ADVANTAGES AND DISADVANTAGES

6.1 APPLICATIONS:

- Track student activities.
- Automate student and staff attendance records.
- Monitor patients in different rooms.
- Libraries have used RFID to switch the barcodes on library thinks.
- RFID technologies area unit currently conjointly enforced in end-user applications in museums.
- Locating landmarks in a city.
- Managing items in a warehouse.
- Locating personal objects.

6.2 ADVANTAGES:

- RFID technology can be used for tracking products or product identification.
- It does not require a line of sight to read the tag.
- RFID reader has a longer read range than barcode reader.
- RFID Tags can store more data than bar code tags.
- RFID readers can simultaneously communicate with multiple tags.

6.3 DISADVANTAGES:

- RFID tag has limited read ranges.
- Interference Issues due to other electronics signals.
- Gets affected in extreme conditions like extreme temperature and moisture.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

CHAPTER - 7

CONCLUSION AND FUTURE SCOPE

7.1CONCLUSION:

In conclusion, the implementation of RFID-based indoor tracking using Arduino and NodeMCU, coupled with Telegram intimation, offers a versatile and efficient solution for various applications ranging from inventory management to personnel tracking. By harnessing the power of RFID technology, real-time monitoring and localization of assets or individuals within confined indoor spaces become feasible. This system not only enhances the efficiency of operations but also improves security measures by providing instant notifications through Telegram, ensuring timely response to any discrepancies or unauthorized movements. Moreover, the integration with NodeMCU facilitates seamless communication between the RFID reader and the Telegram platform, enabling remote monitoring and management capabilities.

Furthermore, the utilization of readily available hardware components such as Arduino and NodeMCU, along with open-source libraries and platforms like Telegram, makes this solution accessible and cost-effective for implementation across various industries and settings. Overall, RFID-based indoor tracking combined with Telegram intimation presents a reliable, scalable, and user-friendly approach to indoor asset and personnel management, promising increased productivity, security, and peace of mind for businesses and organizations. As technology continues to evolve, further enhancements and optimizations can be explored to refine and extend the capabilities of this innovative system.

7.2 FUTURE SCOPE:

- Advancements in RFID technology can lead to tags with longer read ranges and enhanced accuracy, even in challenging environments. This ensures reliable tracking of personnel across larger areas, including outdoor spaces or facilities with complex layouts.
- Future RFID tracking systems can provide real-time monitoring of personnel movements and activities. Integrating RFID data with advanced analytics tools enables organizations to gain actionable insights into workforce behavior, optimize workflow efficiency, and identify potential safety hazards or bottlenecks.
- RFID-based personnel tracking systems can automate attendance tracking and workforce management processes. By seamlessly integrating with HR systems, payroll software, and scheduling tools, organizations can streamline administrative tasks and improve payroll accuracy.
- RFID technology can play a critical role in enhancing safety and emergency response procedures. By providing real-time location information during emergencies, such as fires or evacuations, personnel can be quickly accounted for, and rescue efforts can be coordinated more effectively.
- Future RFID tracking systems should be designed to scale seamlessly as organizational needs evolve. Whether expanding to larger facilities or integrating with new technologies, the system should remain flexible and adaptable to accommodate changing requirements.

BIBLIOGRAPHY

[1] F. Serhan Dani¸s ID and Ali Taylan Cemgil ***"Model-Based Localization and Tracking Using Bluetooth Low-Energy Beacons"***, Galatasaray University, Istanbul, 2017.

[2] Rajashree H S, Sahana B, Surabhi S, T N Apoorva, Dr. Parameshachari B D, ***"Indoor Localization using BLE Technology"***, International Journal of Engineering Research & Technology (IJERT), Vol 6, Issue 13, NCESC - 2018.

[3] Ms. MeghaC K, Mrs.D Hari Priya, ***"Person Tracking in Campus Premises Using Indoor Positioning"***, International Journal of Pure and Applied Mathematics (IJPAM), Volume 118, No.14, pp. 383-386, 2018.

[4] Kais Mekki, Eddy Bajic, Fernand Meyer, ***"Indoor Positioning System for IoT Device based on BLE Technology and MQTT Protocol"***, Proceedings of 5th World Forum on Internet of Things (WF-IoT), pp. 796-801, IEEE April 2019.

[5] Syed Abdullah Fadzli, Sani Iyal Abdulkadir, Mokhairi, Azrul Amri Jamal, ***"Robotic Indoor Path Planning Using Dijkstra's Algorithm with Multi-Layer Dictionaries"***, Proceedings of 2nd International Conference on Information Science and Security (ICISS), pp. 1-4, IEEE December 2022.

[6] Sourabh R Misal, Prajwal S R, Niveditha H M, Vinayaka H M, Veena S, ***"Indoor Positioning System (IPS) Using ESP32, MQTT and Bluetooth"***, Proceedings of 4th International Conference on Computing Methodologies and Communication (ICCMC), ISBN 978-1-7281-4889-2/20, IEEE January 2023.

[7] Vasilis Stavrou, Cleopatra Bardaki, Dimitris Papakyriakopoulos and Katerina Pramatari, ***"An Ensemble Filter for Indoor Positioning in a Retail Store Using Bluetooth Low Energy Beacons"***, PubMed Central (PMC), October 2021.

APPENDIX

❖ Code Snippet:

```
#include<LiquidCrystal.h>
#include <SoftwareSerial.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

const int rs = 14, en = 15, d4 = 16, d5 = 17, d6 = 18, d7 = 19;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

SoftwareSerial Room_1Serial(2, 3);/* (Rx, Tx) */
SoftwareSerial Room_2Serial(4, 5);/* (Rx, Tx) */
SoftwareSerial Room_3Serial(6, 7);/* (Rx, Tx) */
SoftwareSerial Room_4Serial(8, 9);/* (Rx, Tx) */

char CARD_1[]="1500269E9835";
char CARD_2[]="1700A1F593D0";  // INSUFFICIENT BALANCE
char CARD_3[]="1700A204AF1E";
char CARD_4[]="1700A1830D38";

String message="";
char ch;

int a=0;
int b=0;
int c=0;
int d=0;
void RFID_CHECKING();
void setup()
{
  Serial.begin(9600);
  Room_1Serial.begin(9600);
  Room_2Serial.begin(9600);
  Room_3Serial.begin(9600);
  Room_4Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("RFID TRACKING");
  lcd.setCursor(0,1);
  lcd.print("  SYSTEM  ");
  Serial.println("RFID TRACKING SYSTEM...");
  delay(2000);
}
void loop()
{

   MONITORING();

}
void MONITORING()
{
  ROOM1();
  ROOM2();
  ROOM3();
  ROOM4();
}
void ROOM1()
{
   Room_1Serial.begin(9600);
   lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("ROOM1 SCAN YOUR  ");
    lcd.setCursor(0,1);
    lcd.print("CARD......   ");
  //Serial.println("ROOM1...");
    delay(10);
    if(Room_1Serial.available()>0)
    {
      message=Room_1Serial.readString();
      int str_len = message.length() + 1;
      char textmessage[12];
      message.toCharArray(textmessage,str_len);
      Serial.begin(9600);
      Serial.println(textmessage);
      textmessage[12]='\0';
      if((textmessage[11]=='5'))
      {
```

```
  if(a==0)
{
            a=1;
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Person1 Entry");
            lcd.setCursor(0,1);
            lcd.print("In Room1...");
            Serial.println("$Person1 Entry In Room1...#");
             delay(2000);
          }
          else if(a==1)
          {
              a=0;
              lcd.clear();
              lcd.setCursor(0,0);
              lcd.print("Person1 Exit");
              lcd.setCursor(0,1);
              lcd.print("From Room1...");
              Serial.println("$Person1 Exit From Room1...#");
               delay(2000);
          }
      }
      else if((textmessage[11]=='0'))
      {
          if(b==0)
          {
            b=1;
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Person2 Entry");
            lcd.setCursor(0,1);
            lcd.print("In Room1...");
            Serial.println("$Person2 Entry In Room1...#");
             delay(2000);
          }
          else if(b==1)
          {
            b=0;
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Person2 Exit");
            lcd.setCursor(0,1);
            lcd.print("From Room1...");
            Serial.println("$Person2 Exit From Room1...#");
             delay(2000);
          }
      }

      else if((textmessage[11]=='E'))
      {
        if(c==0)
        {
          c=1;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person3 Entry");
          lcd.setCursor(0,1);
          lcd.print("In Room1...");
          Serial.println("$Person3 Entry In Room1...#");
         delay(2000);
        }
        else if(c==1)
        {
           c=0;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person3 Exit");
          lcd.setCursor(0,1);
          lcd.print("From Room1...");
          Serial.println("$Person3 Exit From Room1...#");
         delay(2000);
      }
      else if((textmessage[11]=='8'))
      {
        if(d==0)
        {
          d=1;
          lcd.clear();
          lcd.setCursor(0,0);
```

```
          lcd.print("Person4 Entry");
          lcd.setCursor(0,1);
          lcd.print("In Room1...");

         Serial.println("$Person4 Entry In Room1...#");
         delay(2000);
        }
        else if(d==1)
        {
           d=0;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person4 Exit");
          lcd.setCursor(0,1);
          lcd.print("From Room1...");
          Serial.println("$Person4 Exit From Room1...#");
         delay(2000);

        }
      }

    }
}
void ROOM2()
{
  Room_2Serial.begin(9600);
   lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("ROOM2 SCAN YOUR  ");
    lcd.setCursor(0,1);
    lcd.print("CARD......   ");
  //  Serial.println("ROOM2...");
  delay(10);
    if(Room_2Serial.available()>0)
    {
      message=Room_2Serial.readString();
      int str_len = message.length() + 1;
      char textmessage[12];
      message.toCharArray(textmessage,str_len);
      Serial.println(textmessage);
      textmessage[12]='\0';
```

```
if((textmessage[11]=='5'))
{
   if(a==0)
   {

     a=1;
     lcd.clear();
     lcd.setCursor(0,0);
     lcd.print("Person1 Entry");
     lcd.setCursor(0,1);
     lcd.print("In Room2...");
     Serial.println("$Person1 Entry In Room2...#");
    delay(2000);
   }
   else if(a==1)
   {
      a=0;
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Person1 Exit");
      lcd.setCursor(0,1);
      lcd.print("From Room2...");
      Serial.println("$Person1 Exit From Room2...#");
     delay(2000);
   }
}
else if((textmessage[11]=='0'))
{
   if(b==0)
   {
     b=1;
     lcd.clear();
     lcd.setCursor(0,0);
     lcd.print("Person2 Entry");
     lcd.setCursor(0,1);
     lcd.print("In Room2...");
     Serial.println("$Person2 Entry In Room2...#");
    delay(2000);
   }
   else if(b==1)
```

```
{
            b=0;
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Person2 Exit");
            lcd.setCursor(0,1);

            lcd.print("From Room1...");
            Serial.println("$Person2 Exit From Room2...#");
           delay(2000);
          }
      }

      else if((textmessage[11]=='E'))
      {
        if(c==0)
        {
          c=1;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person3 Entry");
          lcd.setCursor(0,1);
          lcd.print("In Room2...");
          Serial.println("$Person3 Entry In Room2...#");
         delay(2000);
        }
        else if(c==1)
        {
           c=0;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person3 Exit");
          lcd.setCursor(0,1);
          lcd.print("From Room2...");
          Serial.println("$Person3 Exit From Room2...#");
         delay(2000);
        }
      }
      else if((textmessage[11]=='8'))
      {
        if(d==0)
```

```
    {
           d=1;
           lcd.clear();
           lcd.setCursor(0,0);
           lcd.print("Person4 Entry");
           lcd.setCursor(0,1);
           lcd.print("In Room2...");


          Serial.println("$Person4 Entry In Room2...#");
          delay(2000);
         }
         else if(d==1)
         {
            d=0;
           lcd.clear();
           lcd.setCursor(0,0);
           lcd.print("Person4 Exit");
           lcd.setCursor(0,1);
           lcd.print("From Room2...");
           Serial.println("$Person4 Exit From Room2...#");
          delay(2000);
         }
       }
     }
}
void ROOM3()
{
  Room_3Serial.begin(9600);
   lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("ROOM3 SCAN YOUR  ");
    lcd.setCursor(0,1);
    lcd.print("CARD......   ");
  //  Serial.println("ROOM3...");
  delay(10);
    if(Room_3Serial.available()>0)
    {
      message=Room_3Serial.readString();
      int str_len = message.length() + 1;
      char textmessage[12];
```

```
message.toCharArray(textmessage,str_len);
Serial.println(textmessage);
textmessage[12]='\0';
 if((textmessage[11]=='5'))
{
    if(a==0)
    {
      a=1;
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Person1 Entry");
      lcd.setCursor(0,1);
      lcd.print("In Room3...");
      Serial.println("$Person1 Entry In Room3...#");
     delay(2000);
    }
    else if(a==1)
    {
        a=0;
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Person1 Exit");
        lcd.setCursor(0,1);
        lcd.print("From Room3...");
        Serial.println("$Person1 Exit From Room3...#");
       delay(2000);
    }
}
else if((textmessage[11]=='0'))
{
    if(b==0)
    {
      b=1;
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Person2 Entry");
      lcd.setCursor(0,1);
      lcd.print("In Room3...");
      Serial.println("$Person2 Entry In Room3...#");
     delay(2000);
 }
    else if(b==1)
    {
      b=0;
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Person2 Exit");
      lcd.setCursor(0,1);
      lcd.print("From Room3...");
      Serial.println("$Person2 Exit From Room3...#");
     delay(2000);
    }
}
else if((textmessage[11]=='E'))
{
  if(c==0)
  {
    c=1;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Person3 Entry");
    lcd.setCursor(0,1);
    lcd.print("In Room3...");
    Serial.println("$Person3 Entry In Room3...#");
   delay(2000);
  }
  else if(c==1)
  {
     c=0;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Person3 Exit");
    lcd.setCursor(0,1);
    lcd.print("From Room3...");
    Serial.println("$Person3 Exit From Room3...#");
   delay(2000);

  }
}
else if((textmessage[11]=='8'))
```

```
   {
        if(d==0)
        {
          d=1;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person4 Entry");
          lcd.setCursor(0,1);
          lcd.print("In Room3...");
          Serial.println("$Person4 Entry In Room3...#");
         delay(2000);
        }
        else if(d==1)
        {
           d=0;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person4 Exit");
          lcd.setCursor(0,1);
          lcd.print("From Room3...");
          Serial.println("$Person4 Exit From Room3...#");
         delay(2000);

        }
      }
    }
}
void ROOM4()
{
  Room_4Serial.begin(9600);
   lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("ROOM4 SCAN YOUR  ");
    lcd.setCursor(0,1);
    lcd.print("CARD......   ");
   // Serial.println("ROOM4...");
  delay(10);

    if(Room_4Serial.available()>0)
    {
```

```
message=Room_4Serial.readString();
int str_len = message.length() + 1;
char textmessage[12];
message.toCharArray(textmessage,str_len);
Serial.println(textmessage);
textmessage[12]='\0';
 if((textmessage[11]=='5'))
{
    if(a==0)
    {
      a=1;
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Person1 Entry");
      lcd.setCursor(0,1);
      lcd.print("In Room4...");
      Serial.println("$Person1 Entry In Room4...#");
     delay(2000);
    }
    else if(a==1)
    {
        a=0;
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Person1 Exit");
        lcd.setCursor(0,1);
        lcd.print("From Room4...");
        Serial.println("$Person1 Exit From Room4...#");
       delay(2000);
    }
}
else if((textmessage[11]=='0'))
{
    if(b==0)
    {
      b=1;
      lcd.clear();
      lcd.setCursor(0,0);


      lcd.print("Person2 Entry");
      lcd.setCursor(0,1);
      lcd.print("In Room4...");
      Serial.println("$Person2 Entry In Room4...#");
     delay(2000);
    }
    else if(b==1)
    {
      b=0;
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Person2 Exit");
      lcd.setCursor(0,1);
      lcd.print("From Room4...");
      Serial.println("$Person2 Exit From Room4...#");
     delay(2000);
    }
}
else if((textmessage[11]=='E'))
{
  if(c==0)
  {
    c=1;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Person3 Entry");
    lcd.setCursor(0,1);
    lcd.print("In Room4...");
    Serial.println("$Person3 Entry In Room4...#");
   delay(2000);
  }
  else if(c==1)
  {
     c=0;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Person3 Exit");
    lcd.setCursor(0,1);
    lcd.print("From Room4...");
    Serial.println("$Person3 Exit From Room4...#");
     delay(2000);
  }
```

```
       }
      else if((textmessage[11]=='8'))
      {
        if(d==0)
        {
          d=1;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person4 Entry");
          lcd.setCursor(0,1);
          lcd.print("In Room4...");
          Serial.println("$Person4 Entry In Room4...#");
         delay(2000);
        }
        else if(d==1)
        {
           d=0;
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Person4 Exit");
          lcd.setCursor(0,1);
          lcd.print("From Room4...");
          Serial.println("$Person4 Exit From Room4...#");
         delay(2000);
        }
      }
    }
}
```

Explanation of the code:

- LiquidCrystal.h: Library for using LCD displays with Arduino.
- SoftwareSerial.h: Library for serial communication on digital pins.
- Global variables: CARD_1, CARD_2, CARD_3, CARD_4: Arrays storing RFID cardcodes.
    - Message: String variable to store received serial data.
    - Variables a, b, c, d are used as flags to track entry and exit of individuals from each room.

- Function Declarations:
  - RFID_CHECKING(): Function to check RFID card against predefined card codes.
  - MONITORING(): Function to monitor all rooms.
  - ROOM1(), ROOM2(), ROOM3(), ROOM4(): Functions to handle RFID scanning and entry/exit tracking for each room.
- Setup Function:
  - Initializes serial communication, LCD display, and RFID serial ports.
  - Displays a startup message on the LCD and serial monitor.
- Loop Function:
  - Calls the MONITORING() function repeatedly.
- Monitoring Function (MONITORING()):
  - Calls functions to monitor each room.
- Room Functions (ROOM1(), ROOM2(), ROOM3(), ROOM4()):
  - Each function handles RFID scanning for a specific room.
  - When RFID data is available, it reads the data, compares it with predefined cardcodes, and then displays entry or exit messages on the LCD accordingly.
  - Entry and exit actions are determined based on the state of the flags a, b, c, d, which are toggled appropriate