A logo for a university

Description automatically generated



**Module Code & Module Title**

**CC5068NI– Cloud Computing & IoT**

**RFID-bases parking system**

**Assessment Type**

**50% Group Report**

**Semester**

**2024 Spring**

**Group Members**

|  |  |
| --- | --- |
| **London Met ID** | **Student Name** |
| **22067069** | **Fiza Shrestha** |
| **22067084** | **Prithak Babu Shrestha** |
| **22067067** | **Supreem Khadka** |
| **22066752** | **Sujan Panta** |

**Assignment Due Date: 10th May 2024**

**Assignment Submission Date: 9th May 2024**

**Submitted to: Mr. Sugat Man Shakya**

**Word Count: 2137**

*I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded*

# ACKNOWLEDGEMENT

We would like to express our gratitude to our module teachers for their guidance and support and for providing us with an opportunity to work on this project. We would also like to thank our teammates for their contributions and dedication towards this project. Additionally, we express our sincere appreciation towards everyone who have played a role in completion of this project.

# ABSTRACT

RFID based parking system has become significantly productive in the city areas to address the challenges associated with parking management. This solution has become effective in decreasing the search time, traffic congestion, and enhancing the overall user experience. This report contains a structured overview of the development of an RFID-based parking system including the planning, design and implementation steps taken.

Table of Contents

[1. INTRODUCTION 1](#_Toc166184860)

[1.1 Current Scenario 1](#_Toc166184861)

[1.2 Problem Statement and Project as a solution 2](#_Toc166184862)

[1.3 Aim and Objectives 2](#_Toc166184863)

[1.3.1 Aim 2](#_Toc166184864)

[1.3.2 Objectives 2](#_Toc166184865)

[2. BACKGROUND 3](#_Toc166184866)

[2.1 System Overview 3](#_Toc166184867)

[2.2 Design Diagrams 4](#_Toc166184868)

[2.2.1 Hardware Architecture 4](#_Toc166184869)

[2.2.2 Flowchart 5](#_Toc166184870)

[2.2.3 Block diagram 6](#_Toc166184871)

[2.2.4 Circuit diagram 7](#_Toc166184872)

[2.2.5 Schematic diagram 7](#_Toc166184873)

[2.3 Requirement Analysis 8](#_Toc166184874)

[2.3.1 Required Software: 8](#_Toc166184875)

[2.3.2 Required Hardware: 8](#_Toc166184876)

[3. DEVELOPMENT 10](#_Toc166184877)

[3.1 Planning and Design 10](#_Toc166184878)

[3.2 Resource Collection 10](#_Toc166184879)

[3.3 System development 11](#_Toc166184880)

[4. RESULTS AND FINDINGS 14](#_Toc166184881)

[4.1 Test 1 14](#_Toc166184882)

[4.2 Test 2 16](#_Toc166184883)

[4.3 Test 3 18](#_Toc166184884)

[4.4 Test 4 20](#_Toc166184885)

[4.5 Test 5 21](#_Toc166184886)

[5. FUTURE WORKS 23](#_Toc166184887)

[6. CONCLUSION 24](#_Toc166184888)

[7. REFERENCES 25](#_Toc166184889)

[8. APPENDIX 27](#_Toc166184890)

[8.1 Appendix A: Source Code 27](#_Toc166184891)

[8.2 Appendix B: Screenshots of the system 34](#_Toc166184892)

[8.3 Appendix C: Design Diagrams 36](#_Toc166184893)

# TABLE OF FIGURES

[Figure 1. Hardware architecture 4](#_Toc166184894)

[Figure 2. Flowchart 5](#_Toc166184895)

[Figure 3. Block diagram 6](#_Toc166184896)

[Figure 4. Circuit diagram 7](#_Toc166184897)

[Figure 5. Schematic diagram 7](#_Toc166184898)

[Figure 6. Phase 1 of development 11](#_Toc166184899)

[Figure 7. Phase 2 of development 12](#_Toc166184900)

[Figure 8. Phase 3 of development 13](#_Toc166184901)

[Figure 9. Before placing RFID card near the RFID reader 14](#_Toc166184902)

[Figure 10. Green LED failed to glow 15](#_Toc166184903)

[Figure 11. Before placing RFID card near the RFID reader 16](#_Toc166184904)

[Figure 12. Red LED glowed 17](#_Toc166184905)

[Figure 13. Before placing RFID card near the RFID reader 18](#_Toc166184906)

[Figure 14. Position of servo motor before access is granted 19](#_Toc166184907)

[Figure 15. Position of servo motor after access is granted 19](#_Toc166184908)

[Figure 16. Successful compilation of code 20](#_Toc166184909)

[Figure 17. Before placing RFID card near the RFID reader 21](#_Toc166184910)

[Figure 18. Green LED glowed 22](#_Toc166184911)

[Figure 19. Screenshot of the system (ii) 34](#_Toc166184912)

[Figure 20. Screenshot of the system (i) 34](#_Toc166184913)

[Figure 21. Screenshot of the system (iii) 35](#_Toc166184914)

[Figure 22. Work breakdown structure 38](#_Toc166184915)

# TABLE OF TABLE

[Table 1. Test 1 14](#_Toc166184926)

[Table 2. Test 2 16](#_Toc166184927)

[Table 3. Test 3 18](#_Toc166184928)

[Table 4. Test 4 20](#_Toc166184929)

[Table 5. Test 5 21](#_Toc166184930)

[Table 6. Individual contribution plan 37](#_Toc166184931)

# INTRODUCTION

In today's digital age, the IoT has revolutionized how we interact with technology. Connecting devices to the internet allows them to share data and respond to commands in real-time. In our 21st century, IoT is used in almost every field such as Smart Homes, Healthcare, Industries, Smart Cities, Agriculture, Transportation, and many more fields with or without our knowledge.

In our project work, we have used Arduino which is an open-source electronic platform based on easy-to-use hardware and software. The concept of our project work is to create an "RFID Parking System". This project requires us to use the MFRC to read the RFID tag and either allow or deny access to the RFID tags. The need to revolutionize is equally essential as in our country there are many problems regarding parking.

## 1.1 Current Scenario

Nepal is still highly dependent on manual parking management systems. This system has been in use since early years when the number of cars were less in Nepal and parking was not particularly an issue. Till this date, Nepal uses manual-techno parking management system in which parking attendants can be seen holding tickets and collecting money in the parking areas and it requires manual work (ims software, 2022).

The Tangal based outlet of Bhatbhateni Supermarket operates the country’s first multi-level smart parking system. It can accommodate up to 44 four-wheelers and consists of 11 stories (Republica, 2022). Although Nepal is gradually adopting smart parking systems, it will take many more years for the parking systems to develop and improve.

## 1.2 Problem Statement and Project as a solution

The number of vehicles has been increasing day by day and parking has become a major issue since it is a significant cause of traffic congestions and air pollution. Locating available parking spots, especially in central areas like New Road during peak hours can be very difficult and inconvenient.

The RFID based parking system can help guide vehicles to nearest available parking spaces. Features like pre-reservation of parking spaces can be implemented in which only authorized vehicles can be granted access to the parking area using the RFID card. This allows parking operations to run smoothly and reduces the traffic congestions caused by vehicles during the search for empty parking slots. This system also enhances security as only authorized vehicles and individuals are granted access.

## 1.3 Aim and Objectives

### 1.3.1 Aim

The aim of this project is to create a RFID-based parking system, which allows access to authorized vehicles.

### 1.3.2 Objectives

The main objectives of this project are:

* Use of RFID card for access control
* To allow only authorized vehicles in for parking

# BACKGROUND

## 2.1 System Overview

This RFID based parking system allows people to conveniently use the parking system. It can effectively grant or deny access to users with eligible RFID cards and eliminates the need to manually purchase tickets. With the RFID reader installed at the entry and exit points, it can validate each vehicle using RFID tags and control the entry and exit gates with the help of servo motor.

The key features of this prototype that our team has developed are automated access control and efficient gate operation which makes it convenient for users and eliminates the need for manual ticket purchasing and handling.

## 2.2 Design Diagrams

### A close-up of a circuit board Description automatically generated2.2.1 Hardware Architecture

Figure 1. Hardware architecture

### A diagram of a software company Description automatically generated with medium confidence2.2.2 Flowchart

Figure 2. Flowchart

### A diagram of a computer system Description automatically generated2.2.3 Block diagram

Figure 3. Block diagram

### A diagram of a circuit board Description automatically generated2.2.4 Circuit diagram

Figure 4. Circuit diagram

### 2.2.5 Schematic diagram

Figure 5. Schematic diagram

## 2.3 Requirement Analysis

### 2.3.1 Required Software:

* **Arduino IDE 2.3.2:** It is a software used for coding and uploading sketches to Arduino boards. It has features like faster performance, modern editor, responsive interface, autocompletion, code navigation, and live debugger (arduino.cc, 2024).
* **Draw.io:** It is an open-source software that allows users to create flowcharts, diagrams, and other visual representations. It is user friendly and offers a wide range of shapes and symbols for various diagram types. This software also allows users to export their diagrams in various formats such as PNG, PDF, and SVG (draw.io, 2023).
* **Fritzing:** It is an open-source electronics design and prototyping platform that enables users to easily visualize and make prototype circuits (fritzing, 2024).

### 2.3.2 Required Hardware:

* **Arduino uno:** It is a popular microcontroller board based on ATmega328P. Arduino uno contains features like digital I/O pins, analog inputs, clock speed, USB connection, power options, and reset button (arduino, 2024).
* **Breadboard:** It is a versatile construction base used for building temporary prototypes. Breadboards allow users to create and test electronic circuits and are reusable as they do not require permanent connections (Crowell, 2019).
* **Jumper wires:** These are electrical wires having connector pins at each end. It allows users to connect two points of a circuit and are commonly used with prototyping tools such as breadboards. Jumper wires come in three different types which are male-to-male, male-to-female, and female-to-female (HEMMINGS, 2018).
* **Servo motor:** A servo motor is used which is a rotatory actuator. It is an electric motor that provides precise control of position, speed, and torque using a feedback loop system. It can either be a DC motor or an AC motor depending on the application requirements (Electrical4U, 2024).
* **LEDs:** Light Emitting Diode is a semiconductor that emits energy in the form of light when electric current flows through it. The light emitted by LEDs come in different colours (byjus, 2024).
* **RFID reader:** In this project RFID reader is used as a sensor. It is a passive sensor that communicates with RFID tags using radio waves. It can read information from the tag and is also able to write data on them (sparkfun, 2017).
* **Resistor:** A resistor can be defined as a passive two-terminal electrical component that is used for limiting or regulating the flow of current in the circuit (byjus, 2024).
* **Buzzer:** It is a device that converts electrical signals into sound and is used in devices such as alarms and timers. A typical buzzer has a frequency range of around 3300 Hz and operates at a voltage of 3V to 24V DC (elprocus, 2024).
* **I2C Liquid Crystal Display:** It is a module that allows users to connect an LCD to an Arduino board and it includes a built-in potentiometer which allows for adjustment of the contrast (arduinogetstarted, 2024).

# DEVELOPMENT

## 3.1 Planning and Design

For this group coursework, our team decided on developing a RFID based parking system. Initially, the plan was to make simple RFID reader that glows red when access is denied and green when access is granted. But after further research, we enhanced our system so that it can make parking management more efficient.

For this, we added a servo motor which acts as an access barrier and gives access when authorized RFID card is scanned and denies access when unregistered card is scanned. The addition of this device made our project better and more interesting.

## 3.2 Resource Collection

The main resources, which are also mentioned in the background section of this report, were mostly provided by the resource department of Islington College itself. But some resource such as the RFID card, tag, and reader had to be bought by the team-member since it was unavailable in the resource department of the college premises.

## 3.3 System development

**Phase 1:**

Figure 6. Phase 1 of development



At first, we made a connection between Arduino uno and RFID reader (mfrc522) with the help of a breadboard and jumper wires. Then we checked if the RFID was working by using DumpInfo from MRFC522 library in Arduino IDE. The RFID was connected using jumper wires as follows:

* 3.3 V - 3.3 V
* RST - Digital Pin 8
* GND - GND
* MISO - Digital Pin 12
* MOSI - Digital Pin 11
* SCK - Digital Pin 13
* SDA - Digital Pin 10

**Phase 2:**

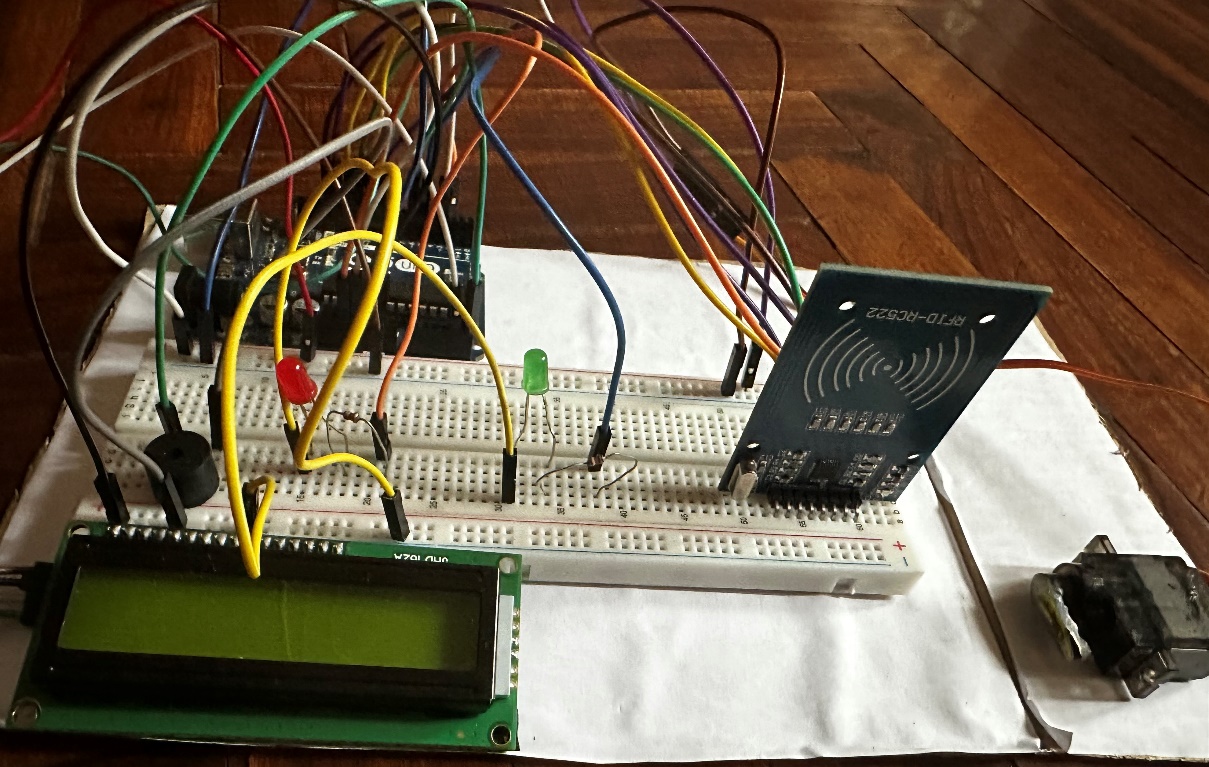


Figure 7. Phase 2 of development

Then other modules were also connected like LCD display, 2 leds, servo motor, and piezo buzzer. Resistors were used with leds to regulate voltage.

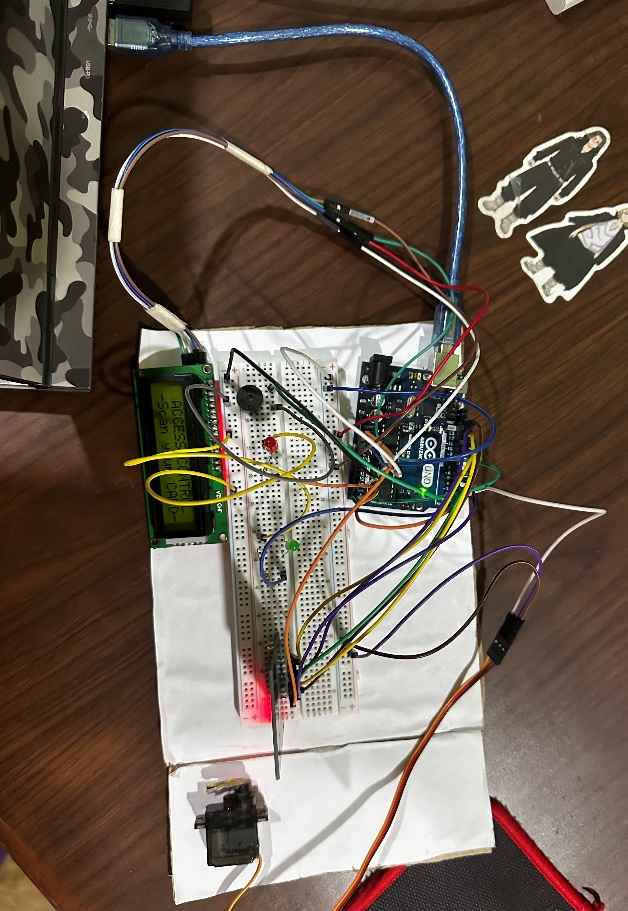
**Phase 3:**

Figure 8. Phase 3 of development

After all the connections were done, we uploaded our code to the Arduino. For uploading the code, we connected the Arduino to the laptop which had the code and then uploaded using Arduino IDE 2.3.2

## RESULTS AND FINDINGS

## 4.1 Test 1

|  |  |
| --- | --- |
| Test | 1 |
| Objective | To check if light glows when access is granted. |
| Action | The authorized RFID card is placed near the RFID reader. |
| Expected result | The LED should glow green. |
| Actual result | The LED did not glow. |
| Conclusion | The test is unsuccessful. |

Table 1. Test 1

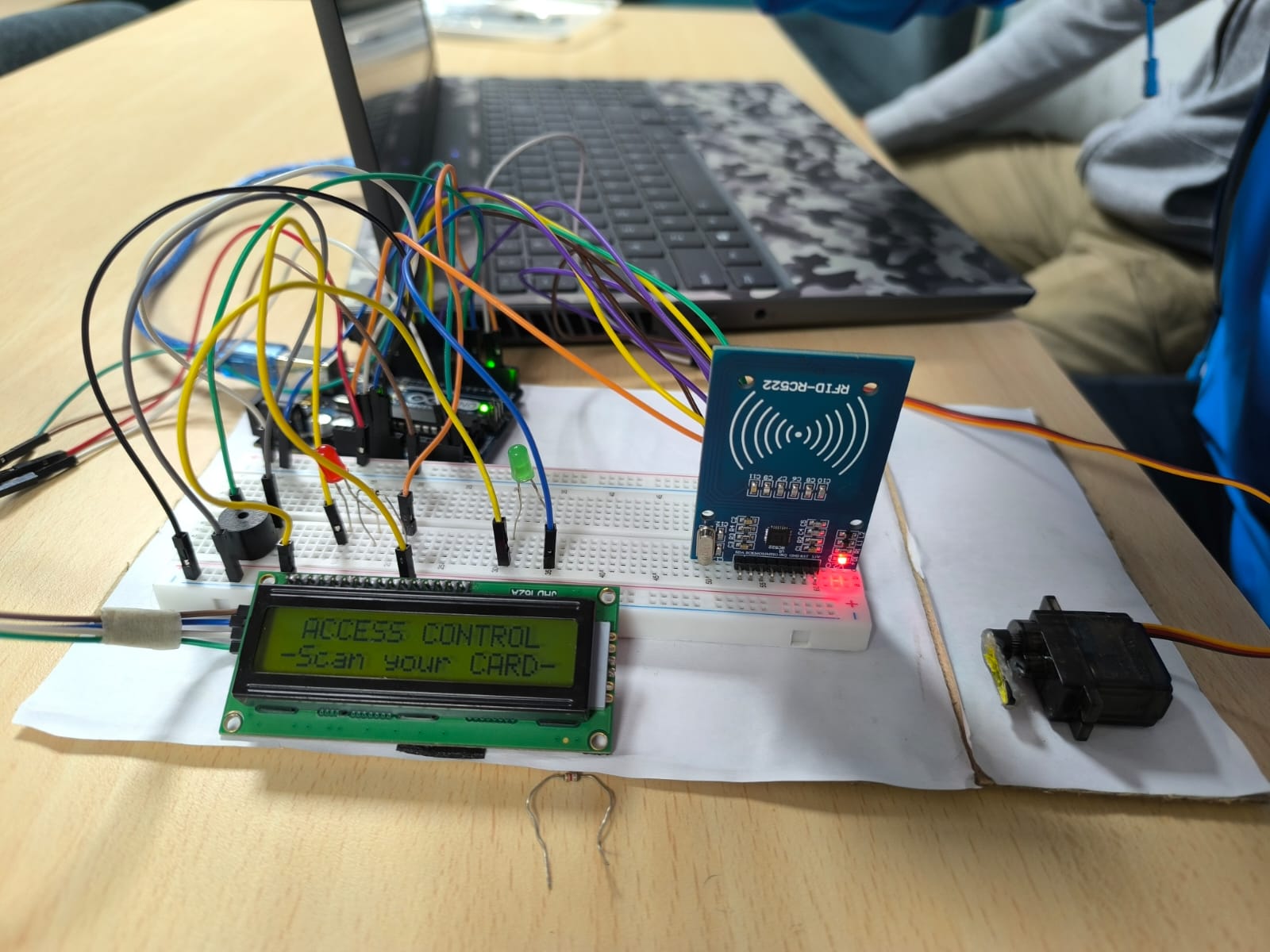


Figure 9. Before placing RFID card near the RFID reader

A hand holding a card over a circuit board

Description automatically generated

Figure 10. Green LED failed to glow

## 4.2 Test 2

|  |  |
| --- | --- |
| Test | 2 |
| Objective | To check if light glows when access is denied. |
| Activity | The unauthorized RFID tag is placed near the RFID reader. |
| Expected result | The LED should glow red. |
| Actual result | The LED did glow red. |
| Conclusion | The test is successful. |

Table 2. Test 2

A circuit board with wires and a screen

Description automatically generated

Figure 11. Before placing RFID card near the RFID reader

A hand holding a key to a circuit board

Description automatically generated

Figure 12. Red LED glowed

## 4.3 Test 3

|  |  |
| --- | --- |
| Test | 3 |
| Objective | To check if the servo motor rotates. |
| Activity | The authorized RFID card us placed near the RFID reader to gain access. |
| Expected result | The servo motor should rotate when access is granted and should not rotate when access is denied. |
| Actual result | The servo motor did rotate when access is granted and did not rotate when access is denied. |
| Conclusion | The test was successful. |

Table 3. Test 3

A circuit board with wires and a screen

Description automatically generated

Figure 13. Before placing RFID card near the RFID reader

A circuit board with wires and a screen

Description automatically generated

Figure 14. Position of servo motor before access is granted

A circuit board with wires and a display

Description automatically generated

Figure 15. Position of servo motor after access is granted

## 4.4 Test 4

|  |  |
| --- | --- |
| Test | 4 |
| Objective | To check if the code executes. |
| Activity | The code was written and compiled. |
| Expected result | The code should compile without any errors. |
| Actual result | The code did compile. |
| Conclusion | The test was successful. |

Table 4. Test 4

A screenshot of a computer

Description automatically generated

Figure 16. Successful compilation of code

## 4.5 Test 5

|  |  |
| --- | --- |
| Test | 5 |
| Objective | To check if light glows when access is granted |
| Activity | The authorized RFID card is placed near the RFID reader. |
| Expected result | The LED should glow green |
| Actual result | The LED did glow green |
| Conclusion | The test is successful |

Table 5. Test 5

A circuit board with wires and a screen

Description automatically generated

Figure 17. Before placing RFID card near the RFID reader

A circuit board with wires and wires

Description automatically generated

Figure 18. Green LED glowed

# FUTURE WORKS

This project work of ours can be further developed. With more time and more effort on this project, we can successfully implement this concept into sectors such as Hospitals, Super Markets, Malls, Theaters, and other places where parking is unavailable due to high traffic or crowds.

A Parking service must be safe and secure for the users to rely on it. With the help of the RFID parking system, we could apply vehicle tracking systems such as GPS to navigate where the vehicle is parked or the find the records where the parking took place. By applying the GPS system, we could also create a system that detects a vacant parking space and notifies us.

RFID-based car parking systems could evolve urban infrastructure by providing not only simplified access, but also valuable data on parking usage, traffic patterns, and user behavior. This data can be processed into useful information analyzing them and using them in machine learning and predictive modeling. These findings allow us to utilize space allocation, forecast future demand, manage traffic flow, and improve the efficiency and user experience of parking facilities.

# CONCLUSION

In conclusion, our project focuses on addressing the pressing issue of parking management in Nepal by introducing an RFID-based parking system. As outlined in the report, manual parking systems currently used in the country contribute to traffic congestion and inconvenience for drivers. Through the integration of Arduino and RFID technology, our project aimed to automate access control, streamline parking operations, and enhance security.

The development process involved meticulous planning, resource collection, and system development phases. Despite encountering challenges during testing, including initial LED functionality issues, our team successfully demonstrated the effectiveness of our system through various tests. These tests validated the functionality of key components such as access control LEDs and servo motor operation.

Looking ahead, there is immense potential for further refinement and expansion of our RFID parking system. Future iterations could explore integration with GPS technology for enhanced vehicle tracking and real-time parking space availability notifications. Additionally, leveraging data collected by RFID-based systems could empower urban planners with insights to optimize parking infrastructure, manage traffic flow, and improve overall user experience.

Ultimately, the RFID-based parking system represents not only a solution to current parking challenges but also a step towards smarter, more efficient urban infrastructure. By harnessing the power of IoT and data analytics, such systems have the capacity to transform how cities manage parking, contributing to improved mobility, sustainability, and quality of life for residents and visitors alike.

# REFERENCES

arduino. (2024). *UNO R3.* Retrieved from arduino: https://docs.arduino.cc/hardware/uno-rev3/

arduino.cc. (2024). *Arduino IDE 2.3.2.* Retrieved from arduino.cc: https://www.arduino.cc/en/software

arduinogetstarted. (2024). *Arduino - LCD I2C.* Retrieved from arduinogetstarted: https://arduinogetstarted.com/tutorials/arduino-lcd-i2c

byjus. (2024). *Light Emitting Diode (LED).* Retrieved from byjus: https://byjus.com/physics/light-emitting-diode/

byjus. (2024). *What is Resistor?* Retrieved from byjus: https://byjus.com/physics/resistor/

Crowell, G. (2019, February 14). *What is a Breadboard?* Retrieved from CircuitBread: https://www.circuitbread.com/ee-faq/what-is-a-breadboard

draw.io. (2023). *draw.io.* Retrieved from draw.io: https://www.drawio.com/

Electrical4U. (2024, April 17). *Servo Motor: Definition, Working Principle, and Applications.* Retrieved from Electrical4U: https://www.electrical4u.com/what-is-servo-motor/

elprocus. (2024). *What is a Buzzer : Working & Its Applications.* Retrieved from elprocus: https://www.elprocus.com/buzzer-working-applications/

fritzing. (2024). *fritzing.* Retrieved from fritzing: https://fritzing.org/download/

HEMMINGS, M. (2018, january 30). *What is a Jumper Wire?* Retrieved from sparkgun education: https://blog.sparkfuneducation.com/what-is-jumper-wire

ims software. (2022, December 4). *What is a parking management system?* Retrieved from ims software: https://imssoftware.com.np/what-is-a-parking-management-system/

Republica. (2022, February 9). *Six groups interested in building smart parking systems at 36 locations in Valley.* Retrieved from myRepublica: https://myrepublica.nagariknetwork.com/news/six-groups-interested-in-building-smart-parking-systems-at-36-locations-in-valley/

sparkfun. (2017, february 23). *USING RFID TAGS AND READERS.* Retrieved from sparkfun: https://www.sparkfun.com/rfid

# APPENDIX

## 8.1 Appendix A: Source Code

#include <LiquidCrystal\_I2C.h>

#include <MFRC522.h>

#include <SPI.h>

#include <Servo.h>

#include <Wire.h>

// Buzzer

const int BUZZER\_PIN = 6;

// LEDs

const int GREEN\_LED = 5; // digital pin of the green led

const int RED\_LED = 2; // digital pin of the red led

// mfrc522 pins

#define RST\_PIN 9

#define SS\_PIN 10

// Instances creation

LiquidCrystal\_I2C lcd(0x27, 16, 2); // Set the LCD address to 0x27 for a 16 chars and 2 line display

Servo s1; // Create Servo instance

MFRC522 mfrc522(SS\_PIN, RST\_PIN); // Create MFRC522 instance

void setup() {

// put your setup code here, to run once:

SPI.begin(); // Initiate SPI bus

Serial.begin(9600); // Initiate a serial communication

// Buzzer

pinMode(BUZZER\_PIN, OUTPUT);

noTone(BUZZER\_PIN);

// LED: Initialize the digital pins as an output

pinMode(GREEN\_LED, OUTPUT);

pinMode(RED\_LED, OUTPUT);

// LCD Display

lcd.begin(); // initializing the LCD

lcd.backlight(); // turn on the back

// mfrc522

mfrc522.PCD\_Init(); // Init MFRC522

delay(4); // optional delay. Some board do need more time after init to be ready

// display to LCD display

// Servo motor

s1.attach(3); // attaching the servo to digital pin 3

s1.write(0); // servo start position

lcd.clear();

lcd.setCursor(1,0);

lcd.print("ACCESS CONTROL");

lcd.setCursor(0,1);

lcd.print("-Scan your CARD-");

}

void loop() {

// put your main code here, to run repeatedly:

// Reset the loop if no new card present on the sensor/reader. This saves the entire process when idle

if ( ! mfrc522.PICC\_IsNewCardPresent())

{

return;

}

// Select one of the cards

if ( ! mfrc522.PICC\_ReadCardSerial())

{

return;

}

Serial.println("UID tag: ");

String card\_id = "";

for (byte i = 0; i < mfrc522.uid.size; i++)

{

Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");

Serial.print(mfrc522.uid.uidByte[i], HEX);

card\_id.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : "0"));

card\_id.concat(String(mfrc522.uid.uidByte[i], HEX));

}

//Serial.print(card\_id.substring(1));

Serial.println();

if (card\_id.substring(1) == "e307b0ac0aa") //change here the UID of the card/cards that you want to give access

{

Serial.println("Authorized Access");

lcd.clear();

lcd.setCursor(1,0);

lcd.print("ACCESS CONTROL");

lcd.setCursor(0,1);

lcd.print("-Access Allowed-");

tone(BUZZER\_PIN, 450);

delay(150);

noTone(BUZZER\_PIN);

s1.write(130);

digitalWrite(GREEN\_LED, HIGH);

delay(5000);

digitalWrite(GREEN\_LED, LOW);

digitalWrite(RED\_LED, HIGH);

tone(BUZZER\_PIN, 450);

delay(150);

noTone(BUZZER\_PIN);

digitalWrite(RED\_LED, LOW);

s1.write(0);

delay(3000);

lcd.clear();

lcd.setCursor(1,0);

lcd.print("ACCESS CONTROL");

lcd.setCursor(0,1);

lcd.print("-Scan your CARD-");

}

else

{

Serial.println("Access denied");

lcd.clear();

lcd.setCursor(1,0);

lcd.print("ACCESS CONTROL");

lcd.setCursor(1,1);

lcd.print("-Access Denied-");

digitalWrite(RED\_LED, HIGH);

tone(BUZZER\_PIN, 300);

delay(500);

digitalWrite(RED\_LED, LOW);

noTone(BUZZER\_PIN);

delay(3000);

lcd.clear();

lcd.setCursor(1,0);

lcd.print("ACCESS CONTROL");

lcd.setCursor(0,1);

lcd.print("-Scan your CARD-");

}

}

## A circuit board with wires and wires Description automatically generatedA circuit board with wires and a screen Description automatically generated8.2 Appendix B: Screenshots of the system

Figure 19. Screenshot of the system (ii)

Figure 20. Screenshot of the system (i)

A circuit board with wires and a display

Description automatically generated

Figure 21. Screenshot of the system (iii)

## Appendix C: Design Diagrams

|  |  |  |
| --- | --- | --- |
| Student Name | Role | Contribution |
| Sujan Panta | Creating Presentation Slides: Analysing the documentation and creating the presentation to present the project work.  Introduction and Testing: Introducing and Testing the Project and the code.  Partial Coding: Helped in completing the code and reviewing it.  Doing Overall Project: Helping with the rest of the minor works. | 25% |
| Prithak Babu Shrestha | Planning and Designing of Project: Planning what assignment is to be prioritized and designing the project.  Assembling Hardware: Connecting Hardware with the help of wires.  Future Scopes/ Works: Planning what the project can do after further development.  Doing Overall Project: Helping with the rest of the minor works. | 25% |
| Fiza Shrestha | Report Formatting: Creating the documents and managing them.  Research of the project: Researching about the project and required hardware and tools.  Background and System Overview: Checking if the assembled components are working properly or not.  Doing Overall Project: Helping with the rest of the minor works.  Circuit Diagram: Drawing the circuit diagram of the Project's system. | 25% |
| Supreem Khadka | Coding: Using the Arduino IDE to write code to run the project.  Idea and research of the Project: Presenting the idea to the group to start the project.  Design of the project: Designing how the project will function and what purpose it will serve.  Result and Findings: Evaluating the project and its best value after implementation.  System Architecture and Flowchart: Operating and maintaining the project system. | 25% |

Table 6. Individual contribution plan

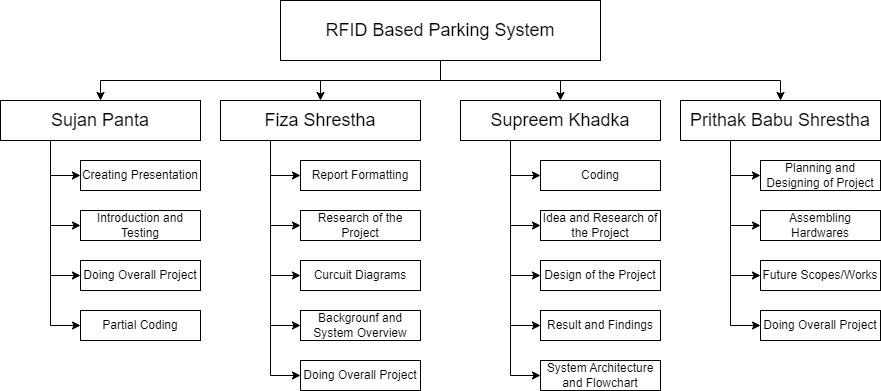


Figure 22. Work breakdown structure