Automated Toll Gate System

Name of Student	Roll No	Slot (Theory)	Course Code
Dogiparthi Aasrith	21BAI1702	TA2	BECE352E
Yarraguntla Kalyan Chakravarthy	21BAI1759	TA2	BECE352E
Hanuma Vihari G	21BRS1022	TA2	BECE352E

Introduction:

The goal of the Arduino Uno tollgate system project is to automate the toll collection procedure at toll plazas. To build an effective and user-friendly system, it combines a number of components, including an LCD display, an IR sensor, an RFID scanner, and a keypad. The tollgate barrier is moved by a servo motor, and further features include an RFID scanner for vehicle authentication, an IR sensor for detecting the presence of cars, an account-recharging keypad, and an LCD display for user feedback such as .

Objectives:

Reduce waiting times and human interference by automating the toll collection process.

Utilize RFID technology to securely and reliably authenticate vehicles.

Use infrared sensors to detect the presence of vehicles to increase efficiency and safety.

Permit consumers to simply refill their toll accounts through the use of a keypad interface.

Present pertinent data on an LCD screen, such as the balance of your card and the status of your transactions.

Components Used:

- Arduino Uno
- Servo Motor
- RFID Scanner
- IR Sensor
- Keypad
- LCD Display

System Architecture:

The following modules make up the architecture of the tollgate system:

Arduino Uno: Microcontroller board for interfacing with other components and controlling system operations

RFID Scanner Module: Authenticates vehicles by reading RFID tags/cards.

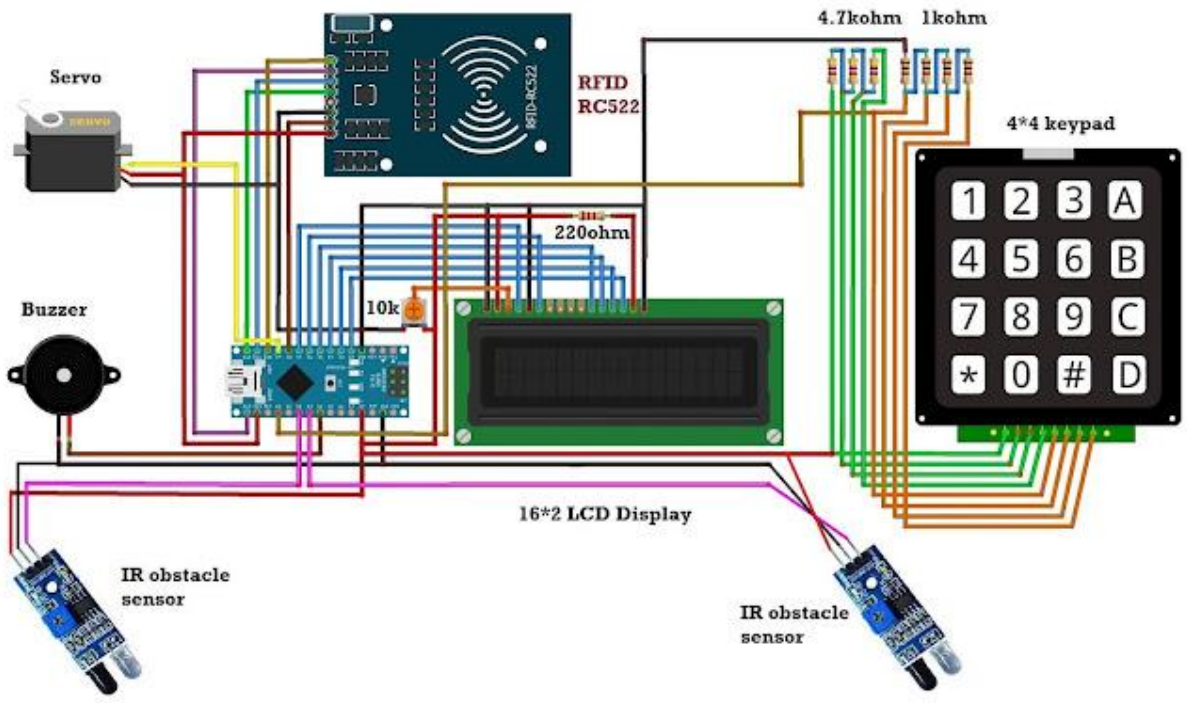
IR Sensor Module: Moves the tollgate barrier when it detects the presence of a vehicle.

Keypad Module: Enables users to enter additional data and the recharge amount.

LCD Display Module: Gives customers visual feedback on instructions, transaction status, and account balance.

Servo Motor Module: Controls the tollgate barrier's movement to permit or prohibit vehicle passage.

Circuit Diagram:

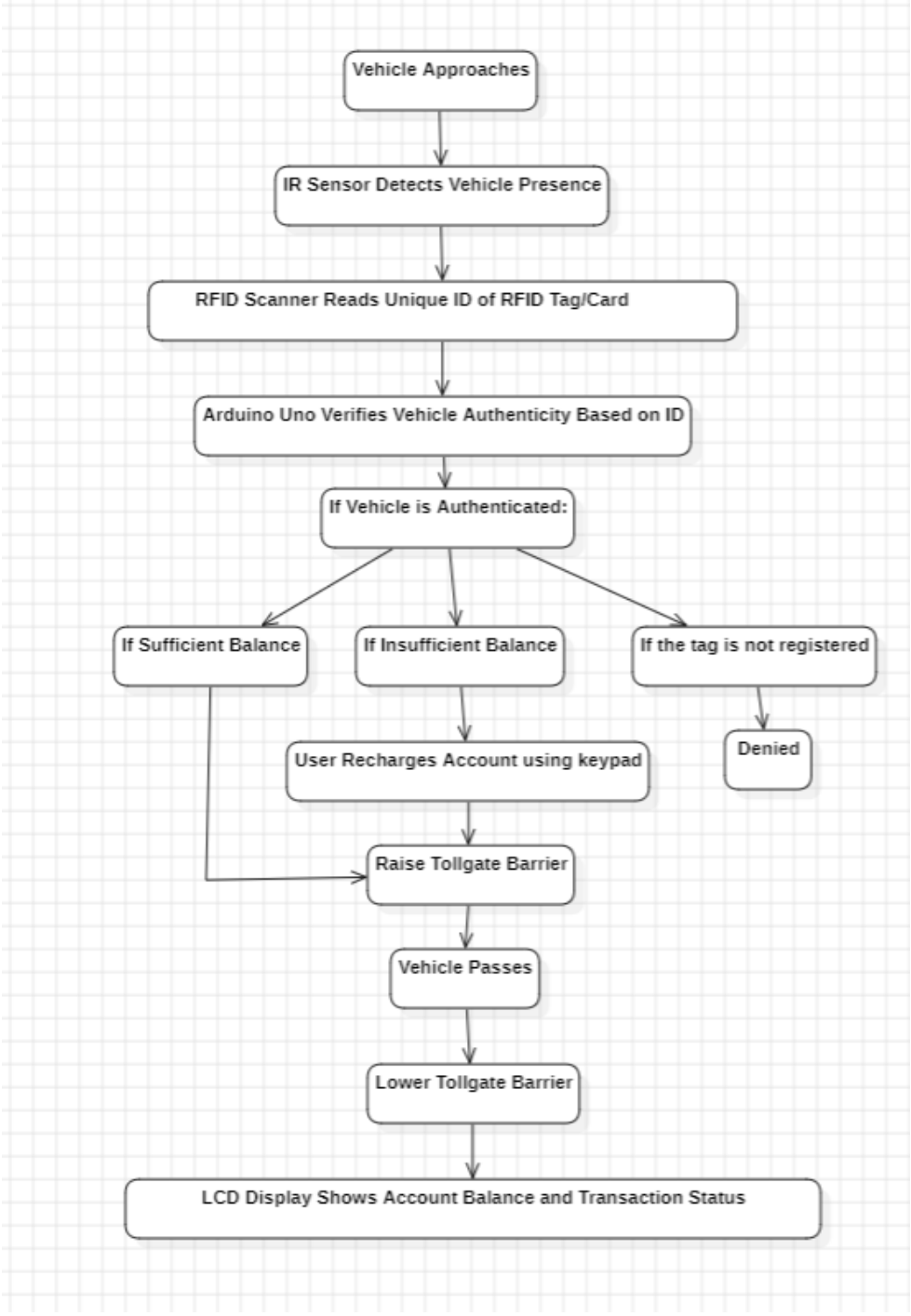


Workflow:

- Vehicle approaches the tollgate.
- IR sensor detects the presence of the vehicle.
- RFID scanner reads the unique ID of the RFID tag/card attached to the vehicle.
- Arduino Uno verifies the authenticity of the vehicle based on the scanned ID.
- If the vehicle is authenticated, Arduino Uno checks the account balance.

- If the balance is sufficient, the tollgate barrier is raised using the servo motor, and the vehicle is allowed to pass.
- If the balance is insufficient, the user can recharge their account using the keypad.
- After successful payment or recharge, the LCD display shows the updated account balance and confirms safe passage.

Flow Chart:



Advantages:

Automation: Increases tollgate efficiency and decreases manual intervention.

Security: RFID technology is used to securely authenticate vehicles.

Convenience: The keypad interface makes it easy for users to recharge their accounts.

Real-time Feedback: The LCD display gives you immediate information about the status of transactions and account balance.

Codes

Code to find the RFID tokens

```
#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN         8
#define SS_PIN          10

MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup() {
  Serial.begin(9600);
  while (!Serial);
  SPI.begin();
  mfrc522.PCD_Init();
  delay(4);
  mfrc522.PCD_DumpVersionToSerial();
  Serial.println(F("Scan PICC to see UID, SAK, type, and data blocks..."));
}

void loop() {
  if ( ! mfrc522.PICC_IsNewCardPresent()) {
    return;
  }
  if ( ! mfrc522.PICC_ReadCardSerial()) {
    return;
  }
  mfrc522.PICC_DumpToSerial(&(mfrc522.uid));
}
```

To open gate based on the above conditions

```
// 72 7E B1 55 insuff
// F3 AE 38 FB successful
// 40 05 15 59 denied
#include <SPI.h>
#include <MFRC522.h>
#include <OnewireKeypad.h>
#include <Servo.h>

#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

Servo servo;
int servoPos = 0;

#define sensorPin1 A2
#define sensorPin2 A3
#define buzzerPin 6

int senVal1 = 0;
int senVal2 = 0;

#define RST_PIN 8
#define SS_PIN 10

int card1Balance = 10000;
int card2Balance = 400;

#define num 7
char Data[num];
byte data_count = 0;

String num1, num2, card, card2;
int a, b;
char Key;

bool recharge = true;

MFRC522 mfrc522(SS_PIN, RST_PIN);

int state = 0;
```

```
char KEYS[] = {
  '1', '2', '3', 'A',
  '4', '5', '6', 'B',
  '7', '8', '9', 'C',
  '0', '#', 'D', '*'
};

OnewireKeypad <Print, 16 > KP2(Serial, KEYS, 4, 4, A0, 4700, 1000);

void setup () {
  lcd.init();       // initialize the lcd

  lcd.backlight();
  Serial.begin(9600);

  servo.attach(9);

  servo.write(90);

  pinMode(sensorPin1, INPUT);
  pinMode(sensorPin2, INPUT);
  pinMode(buzzerPin, OUTPUT);

  KP2.SetKeypadVoltage(5.0);

  SPI.begin();
  mfrc522.PCD_Init();

  lcd.setCursor(0, 0);
  lcd.print(" Automatic toll");
  lcd.setCursor(0, 1);
  lcd.print("colection system");
  delay(3000);
  lcd.clear();
}
void loop()
{

  if (recharge == 0)
  {
```

```
    reCharge();
  }
  else
  {
    lcd.setCursor(0, 0);
    lcd.print("   Welcome!!!");
    sensorRead();
    rfid();
    KeyPad();
    if (senVal1 == 0)
    {
      servoDown();
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Vehicle detected");
      delay(1000);
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Put your card to");
      lcd.setCursor(0, 1);
      lcd.print("the reader......");
      delay(2000);
      lcd.clear();

    }
    else if (senVal2 == 0 && state == 1)
    {
      servoUp();
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Have a safe");
      lcd.setCursor(0, 1);
      lcd.print("journey");
      delay(1000);
      lcd.clear();
      state = 0;
    }
  }
}

void servoDown()
```

```
{
  servo.attach(9);
  // for (servoPos = 90; servoPos <= 180; servoPos += 1)
  // {
  //   servo.write(servoPos);
  //   delay(15);
  // }
  servo.write(180);
}

void servoUp()
{
  servo.attach(9);
  // for (servoPos = 180; servoPos >= 90; servoPos -= 1)
  // {
  //   servo.write(servoPos);
  //   delay(15);
  // }
  servo.write(90);
}

void sensorRead()
{
  senVal1 = digitalRead(sensorPin1);
  senVal2 = digitalRead(sensorPin2);
}

void rfid()
{
  if ( ! mfrc522.PICC_IsNewCardPresent())
  {
    return;
  }
  if ( ! mfrc522.PICC_ReadCardSerial())
  {
    return;
  }

  String content = "";
  for (byte i = 0; i < mfrc522.uid.size; i++)
  {
```

```
  content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
  content.concat(String(mfrc522.uid.uidByte[i], HEX));
}
content.toUpperCase();

if (content.substring(1) == "F3 AE 38 FB")
{
  if (card1Balance >= 500)
  {
    lcdPrint();
    card1Balance = card1Balance - 500;
    lcd.setCursor(9, 1);
    lcd.print(card1Balance);
    delay(2000);
    lcd.clear();
    state = 1;
  }
  else
  {
    card = content.substring(1);
    LcdPrint();
    lcd.setCursor(9, 1);
    lcd.print(card1Balance);
    lcd.print(" Tk");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Please Recharge");
    delay(1000);
    lcd.clear();
    state = 0;
  }
}
else if (content.substring(1) == "72 7E B1 55")
{
  if (card2Balance >= 500)
  {
    lcdPrint();
    card2Balance = card2Balance - 500;
    lcd.setCursor(9, 1);
    lcd.print(card2Balance);
```

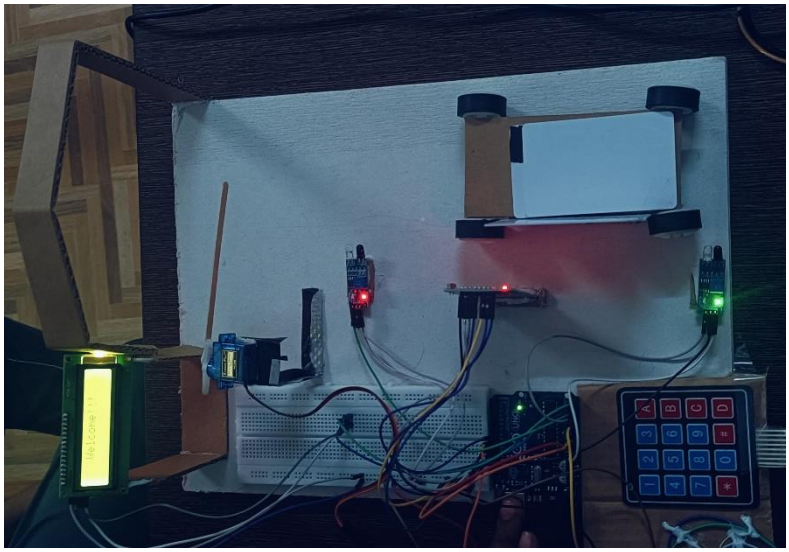
```
    delay(2000);
    lcd.clear();
    state = 1;
   }
   else
   {
    card = content.substring(1);
    LcdPrint();
    lcd.setCursor(9, 1);
    lcd.print(card2Balance);
    lcd.print(" Tk");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Please Recharge");
    lcd.clear();
    delay(1000);
    state = 0;
   }
  }

  else   {
   digitalWrite(buzzerPin, HIGH);
   lcd.setCursor(0, 0);
   lcd.print("Unknown Vehicle");
   lcd.setCursor(0, 1);
   lcd.print("Access denied");
   delay(1500);
   lcd.clear();
   digitalWrite(buzzerPin, LOW);
  }
}

void KeyPad()
{
  byte KState = KP2.Key_State();

  if (KState == PRESSED)
  {
   Key = KP2.Getkey();
   if (Key)
```

```
    {
      if (Key == 'A')
      {
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Recharging Mode.");
        lcd.setCursor(0, 1);
        lcd.print(".................");
        delay(1500);
        lcd.clear();
        recharge = 0;
      }
    }
  }
}

void clearData()
{
  while (data_count != 0)
  {
    Data[data_count--] = 0;
  }
  return;
}

void reCharge()
{

  lcd.setCursor(0, 0);
  lcd.print ("Enter the amount");

  byte KState = KP2.Key_State();

  if (KState == PRESSED)
  {
    Key = KP2.Getkey();
    if (Key)
    {
      if (Key == '#')
      {
        if (card == "72 7E B1 55")
```

```
 {
  num1 = Data;
  card2Balance = num1.toInt() + card2Balance;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Your current");
  lcd.setCursor(0, 1);
  lcd.print("balance: ");
  lcd.setCursor(9, 1);
  lcd.print (card2Balance);
  lcd.print(" Tk");
  delay(3000);
  clearData();
  lcd.clear();
  recharge = 1;
 }
 else if (card == "")
 {
  num2 = Data;
  card1Balance = num2.toInt() + card1Balance;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Your current");
  lcd.setCursor(0, 1);
  lcd.print("balance: ");
  lcd.setCursor(9, 1);
  lcd.print (card1Balance);
  lcd.print(" Tk");
  delay(3000);
  clearData();
  lcd.clear();
  recharge = 1;
 }
}
else
{
 Data[data_count] = Key;
 lcd.setCursor(data_count, 1);
 lcd.print(Data[data_count]);
 data_count++;
}
```

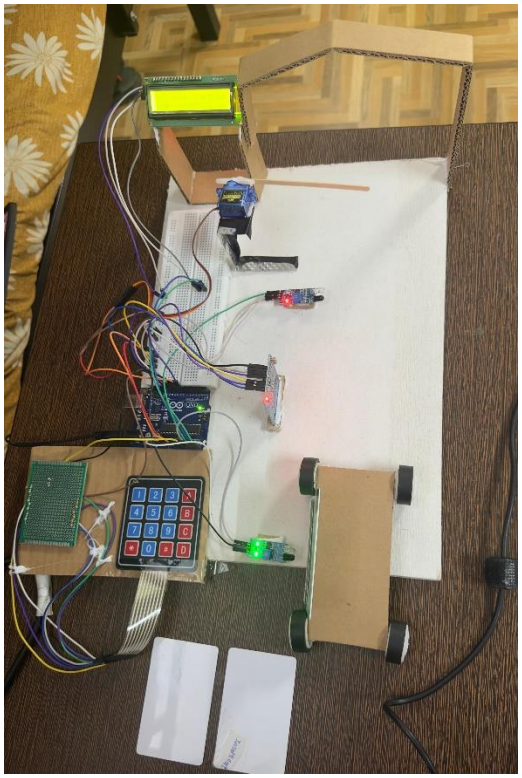
```
   }
  }
}
void lcdPrint()
{
  digitalWrite(buzzerPin, HIGH);
  delay(200);
  digitalWrite(buzzerPin, LOW);
  delay(100);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("  Successfully");
  lcd.setCursor(0, 1);
  lcd.print(" paid your bill");
  delay(1500);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Your Remaining");
  lcd.setCursor(0, 1);
  lcd.print("balance: ");
}

void LcdPrint()
{
  digitalWrite(buzzerPin, HIGH);
  delay(200);
  digitalWrite(buzzerPin, LOW);
  delay(100);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("  Your balance");
  lcd.setCursor(0, 1);
  lcd.print(" is insufficent");
  delay(1500);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Your Remaining");
  lcd.setCursor(0, 1);
  lcd.print("balance: ");
}
```

Image of real Project/simulation

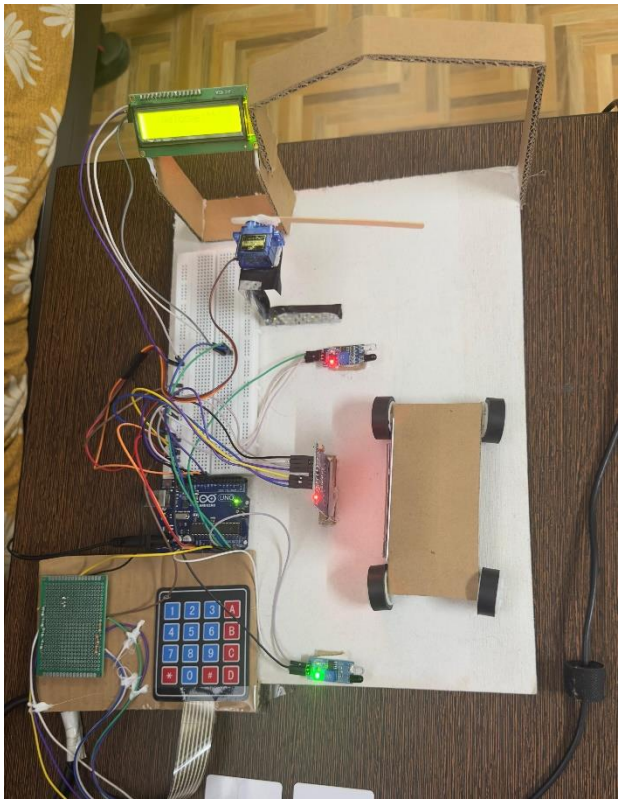


There are three types of vehicles passing the tollgates

The IR sensor detects the vehicles and displays a welcome message in LCD



After vehicle passes through IR senor and moves RFID scanner

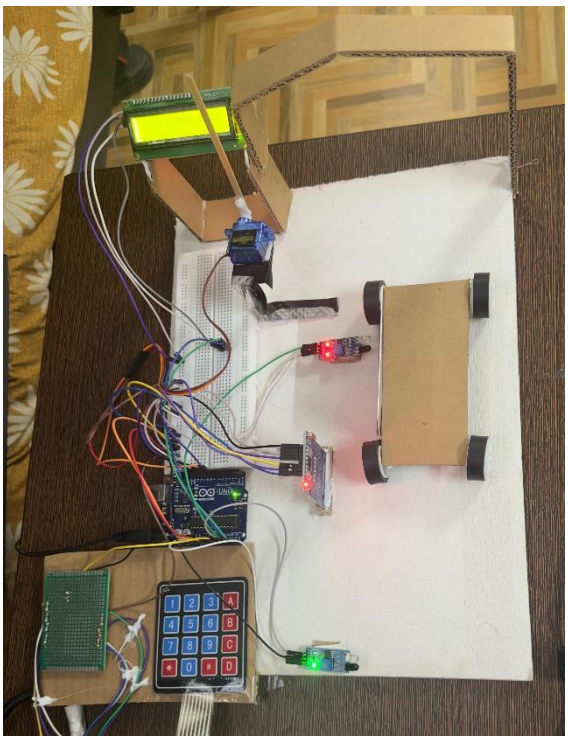


There are three types of vehicles

<u>Sufficient Balance</u>

And the balance is shown on LCD

The tollgate opens when the bill is paid successfully and detected at next IR sensor



Insufficient Balance

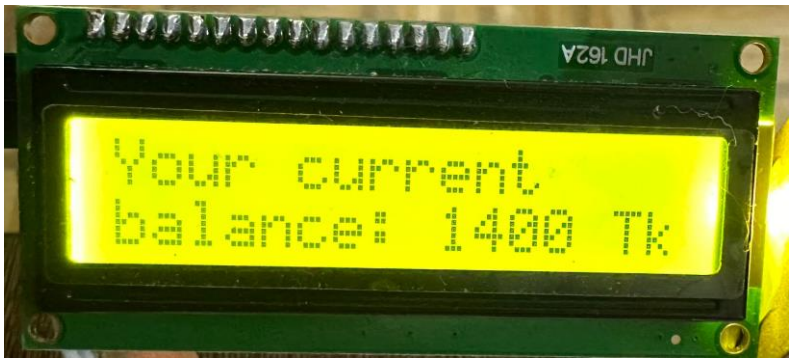
After the card gets scanned if there is less amount you need to recharge the card by using keypad.

The amount needed to pass the tollgate is 500. So, recharge through keypad

Enter A key in keypad to recharge

Enter the amount through keypad

The updated balance is shown as below



After the recharge same process is repeated as shown above for sufficient balance.

Not registered

After the vehicle gets scanned if the vehicle is not registered it shows denied as below

And the vehicle is not passed through tollgate.

Conclusion:

The tollgate system using Arduino Uno offers an efficient and user-friendly solution for toll collection at toll plazas. By leveraging RFID technology, IR sensors, keypad, and LCD display, the system provides secure authentication, convenient payment options, and real-time feedback to users. It has the potential to revolutionize tollgate operations and improve overall transportation infrastructure.