# **1.INTRODUCTION:**

* In this project we try showing how can we collect bus fare(automatically) and add this in our enrollment fee after the semester.
* By this project we got actual calculation of accounts of the branch of Transport.
* Students can’t face fare collection harassment.
* Every students can go their destination freely.
* Without money students can go the bus for this project.(because their money add to the enrollment fee)
* Bus contractor can’t face trouble for collecting fare.
* Outsider of university can’t travel by the bus.
* The director of the branch of transport can handle everything by our mobile application in his/her handphone.

**Final look of the project:**



If we implement this project in our buses ,we think this is the revolutionary changes for our transport branch.

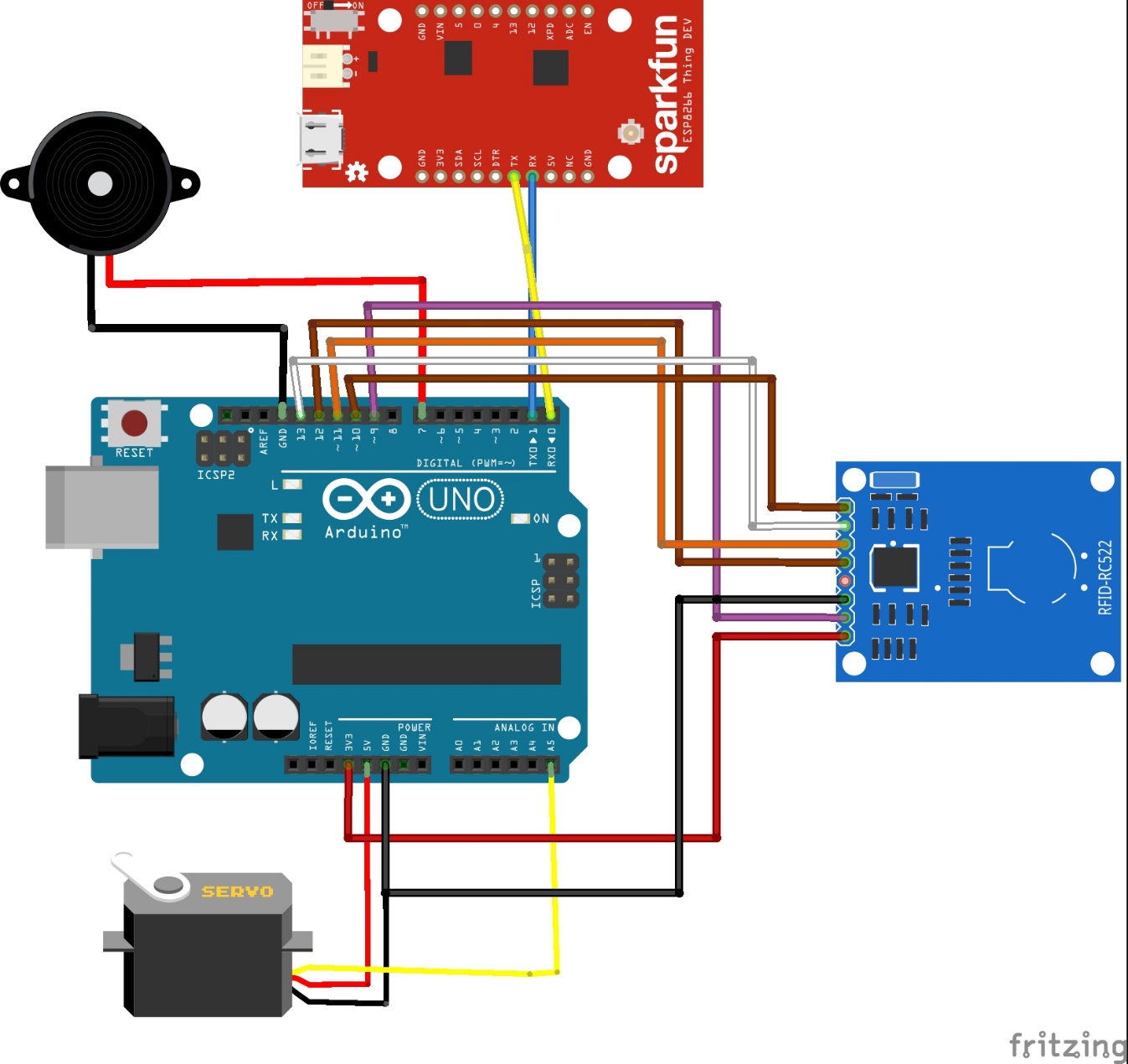
## 2. **Methodology:**

**Equipment:**

* An Arduino Uno.
* A RFID Sensor.
* A Bred board.
* Two Led.
* A Buzzer.
* Some Jumper Wears.

**Setup:**

Frist of all we connecting RFID sensor with Arduino uno by male to female jumper wear .we gives our connection diagram below.



After that we connects our desire led and buzzer with Arduino by using bred board. (Green for acceptable card, Red& buzzer for access denied)

Then we make a sample of bus by using cockpit for attracting the project.

**Code and Logical expression:**

Below we submitted our full code of this project

#include <SoftwareSerial.h>

SoftwareSerial esp8266(2, 3); // RX, TX

#include <SPI.h>

#include <MFRC522.h>

#include <Servo.h>

#define SS\_PIN 10

#define RST\_PIN 9

#define SERVO\_PIN A5

MFRC522 rfid(SS\_PIN, RST\_PIN);

Servo servo;

// My authorized RFID Tags

byte authorizedUID1[4] = { 0xC1, 0xBE, 0x9A, 0x1C };

byte authorizedUID2[4] = { 0x23, 0xE6, 0x9E, 0xE5 };

byte authorizedUID3[4] = { 0xB3, 0xA2, 0x12, 0xE6 };

byte authorizedUID4[4] = { 0x63, 0xE6, 0x34, 0xE6 };

byte authorizedUID5[4] = { 0xF3, 0xEC, 0xDE, 0xE5 };

byte authorizedUID6[4] = { 0x73, 0x70, 0x93, 0xE5 };

byte authorizedUID7[4] = { 0xF3, 0xEC, 0xDE, 0xE5 };

int angle = 0; // the current angle of servo motor

void setup() {

Serial.begin(9600);

Serial.begin(9600);

esp8266.begin(9600);

SPI.begin(); // init SPI bus

rfid.PCD\_Init(); // init MFRC522

servo.attach(SERVO\_PIN);

servo.write(angle); // rotate servo motor to 0°

pinMode(7, OUTPUT);

Serial.println("Tap RFID/NFC Tag on reader");

}

void loop() {

if (rfid.PICC\_IsNewCardPresent()) { // new tag is available

if (rfid.PICC\_ReadCardSerial()) { // NUID has been readed

MFRC522::PICC\_Type piccType = rfid.PICC\_GetType(rfid.uid.sak);

Serial.println("Card found");

if (rfid.uid.uidByte[0] == authorizedUID1[0] && rfid.uid.uidByte[1] == authorizedUID1[1] && rfid.uid.uidByte[2] == authorizedUID1[2] && rfid.uid.uidByte[3] == authorizedUID1[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 1");

changeServo();

delay(2000);

changeServo();

} else if (rfid.uid.uidByte[0] == authorizedUID2[0] && rfid.uid.uidByte[1] == authorizedUID2[1] && rfid.uid.uidByte[2] == authorizedUID2[2] && rfid.uid.uidByte[3] == authorizedUID2[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 2");

changeServo();

delay(2000);

changeServo();

} else if (rfid.uid.uidByte[0] == authorizedUID3[0] && rfid.uid.uidByte[1] == authorizedUID3[1] && rfid.uid.uidByte[2] == authorizedUID3[2] && rfid.uid.uidByte[3] == authorizedUID3[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 3");

changeServo();

delay(2000);

changeServo();

} else if (rfid.uid.uidByte[0] == authorizedUID4[0] && rfid.uid.uidByte[1] == authorizedUID4[1] && rfid.uid.uidByte[2] == authorizedUID4[2] && rfid.uid.uidByte[3] == authorizedUID4[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 4");

changeServo();

delay(2000);

changeServo();

} else if (rfid.uid.uidByte[0] == authorizedUID5[0] && rfid.uid.uidByte[1] == authorizedUID5[1] && rfid.uid.uidByte[2] == authorizedUID5[2] && rfid.uid.uidByte[3] == authorizedUID5[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 5");

changeServo();

delay(2000);

changeServo();

} else if (rfid.uid.uidByte[0] == authorizedUID6[0] && rfid.uid.uidByte[1] == authorizedUID6[1] && rfid.uid.uidByte[2] == authorizedUID6[2] && rfid.uid.uidByte[3] == authorizedUID6[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 6");

changeServo();

delay(2000);

changeServo();

} else if (rfid.uid.uidByte[0] == authorizedUID7[0] && rfid.uid.uidByte[1] == authorizedUID7[1] && rfid.uid.uidByte[2] == authorizedUID7[2] && rfid.uid.uidByte[3] == authorizedUID7[3]) {

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

esp8266.println();

// esp8266.println("Authorized Tag 7");

changeServo();

delay(2000);

changeServo();

} else {

digitalWrite(7, HIGH);

esp8266.print("!");

for (int i = 0; i < rfid.uid.size; i++) {

esp8266.print(rfid.uid.uidByte[i] < 0x10 ? " 0" : " ");

esp8266.print(rfid.uid.uidByte[i], HEX);

}

Serial.println();

delay(500);

digitalWrite(7, LOW);

}

rfid.PICC\_HaltA(); // halt PICC

rfid.PCD\_StopCrypto1(); // stop encryption on PCD

}

}

}

void changeServo() {

// change angle of servo motor

if (angle == 0)

angle = 180;

else //if(angle == 90)

angle = 0;

// control servo motor arccoding to the angle

servo.write(angle);

Serial.print("Rotate Servo Motor to ");

Serial.print(angle);

Serial.println("°");

}

We write this code in the Arduino IDE by Using Arduino code language for uploading code in Arduino uno.

# **3.Implementation:**

**Explanation:**

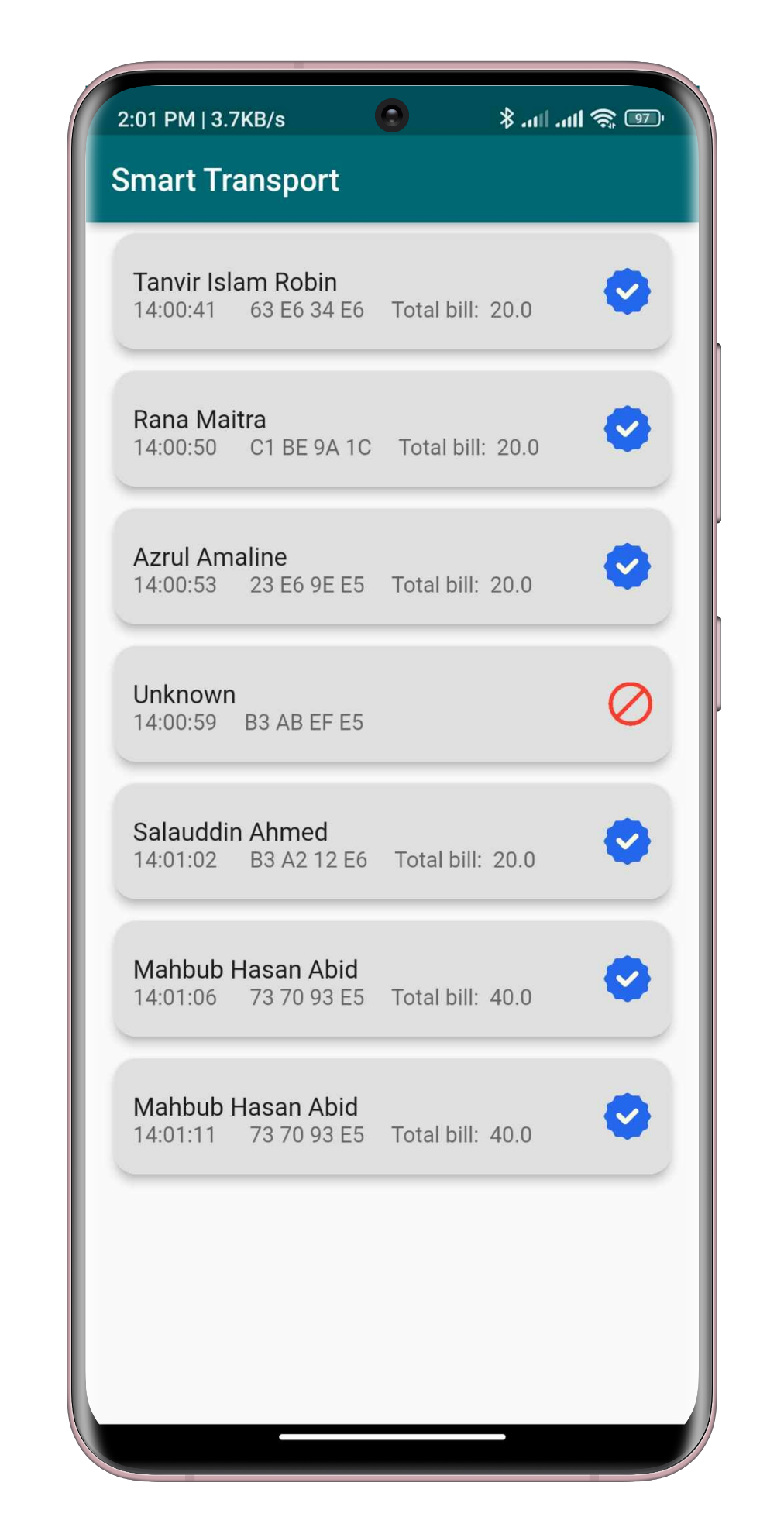
* **Step-1:** We write our code for 3 RFID card we give access with green led and do not give access with red led and buzzer .then upload code in the Arduino uno. Then we need to store the collecting data and the total fare of each student so, we install PLX DAQ-V2 excel spreadsheet and connected this with the code.
* **Step-2:** We cutting and resizing the cockpit with the shape of a little bus.
* **Step-3:** We fixed the RFID sensor side of door.

**Challenge facing:**

* Error handling of code.
* Giving connection between Arduino , bred board and RFID Sensor.
* Making the sample bus.

# 4. **Results and Analysis In My Mobile APP:**

Below we add a report Card of the project.



Here we saw ID 2002052 punch his card total number of 4, so total number of taka 80.(20\*4)

And saw row no. 8 and 9 access denied because in this time punching card is not our Student id card.