

FOR NODE MCU

```
#include <Wire.h>
```

```
#include <PN532.h>
```

```
#include <PN532_I2C.h>
```

```
#include <NfcAdapter.h>
```

```
#include <ESP8266WiFi.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
void connectWIFI(); void
```

```
displayEntry(); void
```

```
displayExit(); void
```

```
readTAG();
```

```
void invalid(); void
```

```
getMeterCount(); void
```

```
calculateDistance(); void
```

```
detectAmount(); void
```

```
sendSMS();
```

```
String tagId;
```

```
WiFiClient client;
```

```
PN532_I2C pn532_i2c(Wire);
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
NfcAdapter nfc = NfcAdapter(pn532_i2c);
```

```
int count; float m_value; float  
scount, ecount, distance; float  
amount, bal = 500;
```

```
String  MakerIFTTT_Key;  String  
MakerIFTTT_Event;  
char *append_str(char *here, String s) {  
    int i = 0;  
    while (*here++ = s[i]) { i++; }; return here - 1;  
} char *append_ul(char *here, unsigned long u) {  
char buf[20]; return append_str(here, ultoa(u,  
buf, 10));  
} char post_rqst[256]; char *p;  
char *content_length_here;  
char *json_start; int comp;
```

```
void setup() {  
    Serial.begin(9600);  
    Wire.begin(8); nfc.begin();  
    lcd.init();  
    lcd.begin(16, 2);  
    lcd.backlight();  
    WiFi.disconnect();  
    delay(500); connectWIFI();  
}
```

```

void loop() {  lcd.setCursor(0,
0);  lcd.print("Scan Your
Card");  if (nfc.tagPresent()) {
lcd.clear();

    readTAG();  if (tagId == "53
9C 2C 4C") {    count++;    if
(count == 1) {

        Serial.println("Passanger Entered");
getMeterCount();    scount = m_value;

        Serial.print("Starting Meter Count : ");
Serial.println(scount);    displayEntry();

    } else if (count == 2) {
Serial.println("Passanger Left");
getMeterCount();    ecount =
m_value;

        Serial.print("Exit Meter Count : ");    Serial.println(ecount);    calculateDistance();
detectAmount();    displayExit();    sendSMS();    amount = 0;    count = 0;

    }

    } else {    invalid();

    }

}

}

void connectWIFI() {

```

```
WiFi.begin("Anish Samuel", "anisharon");

while ((!(WiFi.status() == WL_CONNECTED))) {

    delay(300);    lcd.setCursor(0,
0);    lcd.print("Waiting for");
lcd.setCursor(0, 1);
lcd.print("Connection ...");
    }    lcd.clear(); lcd.setCursor(0,
0); lcd.print("Wifi connected");
lcd.setCursor(0, 1);
lcd.print("Ready to Go ...");
Serial.println("Wifi
Connected"); delay(1000);

    lcd.clear();
}
```

```
void displayEntry() {
lcd.setCursor(0, 0);
lcd.print("Hi");
lcd.setCursor(2, 1);
lcd.print("Anish Samuel");
delay(1000);

    lcd.clear();    lcd.setCursor(0,
0);    lcd.print("Entry Meter");
lcd.setCursor(0, 1);
```

```
lcd.print("Count : ");  
lcd.setCursor(8, 1);  
lcd.print(int(scount));  
delay(1000);  
    lcd.clear();  
}
```

```
void displayExit() {  
    lcd.setCursor(0, 0);  
    lcd.print("Thank You");  
    lcd.setCursor(2, 1);  
    lcd.print("Anish Samuel");  
    delay(1000);  
    lcd.clear(); lcd.setCursor(0,  
0); lcd.print("Exit Meter");  
    lcd.setCursor(0, 1);  
    lcd.print("Count : ");  
    lcd.setCursor(8, 1);  
    lcd.print(int(ecount));  
    delay(1000);  
    lcd.clear(); lcd.setCursor(0,  
0); lcd.print("Distance:");  
    lcd.setCursor(9, 0);  
    lcd.print(distance);  
    lcd.setCursor(0, 1);
```

```
lcd.print("Cost: Rs. ");  
lcd.setCursor(10, 1);  
lcd.print(amount);  
delay(1000);  
    lcd.clear(); }
```

```
void readTAG() { NfcTag tag  
    = nfc.read(); tagId =  
    tag.getUidString();  
    Serial.print("Tag : ");  
  
    Serial.println(tagId);  
}
```

```
void invalid() {  
    lcd.setCursor(0, 0);  
    lcd.print("Sorry,");  
    lcd.setCursor(0, 1);  
    lcd.print("It is invalid");  
    delay(1000);  
    lcd.clear();  
}
```

```
void    getMeterCount()    {  
    Wire.requestFrom(8,    3);  
    m_value = Wire.read();
```

```
Serial.print("Count Received : ");  
Serial.println(m_value);  
}
```

```
void calculateDistance() { distance =  
(ecount - scount) / 10;  
Serial.print("Distance Travelled : ");  
Serial.println(distance);  
}
```

```
void detectAmount() {  
    amount = distance * 2;  
    Serial.print("Amount Detected : ");  
Serial.println(amount); bal = bal -  
amount;  
    Serial.print("Balance : ");  
    Serial.println(bal);  
}
```

```
void sendSMS() {  
    if (client.connect("maker.ifttt.com", 80)) {  
        MakerIFTTT_Key = "fQgtuWqFb1wuzODAOQ8NIQGgKkxhOPC66Bm1dEl4Xk0";  
        MakerIFTTT_Event = "fare";  
        p = post_rqst; p = append_str(p, "POST /trigger/"); p =  
append_str(p, MakerIFTTT_Event); p = append_str(p,
```

```

"/with/key/");  p = append_str(p, MakerIFTTT_Key);  p =
append_str(p, " HTTP/1.1\r\n");  p = append_str(p, "Host:
maker.ifttt.com\r\n");  p = append_str(p, "Content-Type:
application/json\r\n");  p = append_str(p, "Content-Length:
");  content_length_here = p;  p = append_str(p,
"NN\r\n");  p = append_str(p, "\r\n");  json_start = p;  p
= append_str(p, "{\"value1\":\");  p = append_str(p,
String(amount));

p = append_str(p, "\",\"value2\":\");
p = append_str(p, String(distance));  p =
append_str(p, "\",\"value3\":\");  p =
append_str(p, String(bal));  p =
append_str(p, "\"}");

compi = strlen(json_start);  content_length_here[0]
= '0' + (compi / 10);  content_length_here[1] = '0' +
(compi % 10);  client.print(post_rqst);

Serial.println("SMS Sent");

}

}

```


