

# Project: WiFi Card and Key Fob Reader

This project combines the **RFID RC522** with a **Web Server**. The ESP32 will act as a standalone WiFi hotspot, allowing you to see the "Primary Key" (p) of any scanned card on your phone's browser.

## 1. Connections

The RC522 uses SPI, so wiring is specific. Ensure you use the **\*\*3.3V\*\*** pin, as 5V can damage the module.

ESP32 Pin	RC522 Pin	Description
<b>3.3V</b>	<b>3.3V</b>	Power (Strictly 3.3V)
<b>GND</b>	<b>GND</b>	Ground
<b>GPIO 5</b>	<b>SDA (SS)</b>	Slave Select
<b>GPIO 18</b>	<b>SCK</b>	SPI Clock
<b>GPIO 23</b>	<b>MOSI</b>	SPI Master Out
<b>GPIO 19</b>	<b>MISO</b>	SPI Master In
<b>GPIO 22</b>	<b>RST</b>	Reset
<b>No Connection Needed</b>	<b>IRQ</b>	Interrupt Request

## 2. Libraries that are needed to be downloaded

You need to install this one library from the Arduino Library Manager.

Go to **Tools > Manage Libraries** and install:

1. **MFRC522** by GithubCommunity

## 3. Code

### *How it works*

The ESP32 creates a WiFi network called "ESP32-RFID-Scanner". It hosts a web page that displays the last scanned Card UID. The variable `lastUID` acts as our data holder. When a card is scanned, the ESP32 updates this string and your phone fetches it when you refresh the page.

## Steps to connect

- ✓ Connect your phone to the  
WiFi: **ESP32-RFID-Scanner**  
Password: **123456789**
- ✓ Open your browser and go to **192.168.4.1**

## Code

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

#define SS_PIN 5
#define RST_PIN 22

// a = Aim: Broadcast RFID scans to a mobile web dashboard
MFRC522 rfid(SS_PIN, RST_PIN);
WebServer server(80);

String lastUID = "No card scanned yet";

void handleRoot() {
    String html = "<html><head><meta name='viewport' content='width=device-width, initial-
scale=1.0'>";
    html += "<meta http-equiv='refresh' content='2'></head><body style='font-family:Arial; text-
align:center;'>";
    html += "<h1>RFID Dashboard</h1>";
    html += "<div style='padding:20px; background:#f0f0f0; display:inline-block; border-
radius:10px;'>";
    html += "<h3>Last Scanned UID (p):</h3>";
    html += "<h2 style='color:blue;'>" + lastUID + "</h2>";
    html += "</div><p>Scanning for cards...</p></body></html>";
    server.send(200, "text/html", html);
}

void setup() {
    Serial.begin(115200);
    SPI.begin();
    rfid.PCD_Init();

    WiFi.softAP("ESP32-RFID-Scanner", "123456789");
    server.on("/", handleRoot);
    server.begin();

    Serial.println("WiFi Hotspot Started. IP: 192.168.4.1");
}
```

```

void loop() {
    server.handleClient();

    // Check for new cards
    if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
        lastUID = "";
        for (byte i = 0; i < rfid.uid.size; i++) {
            lastUID += String(rfid.uid.uidByte[i] < 0x10 ? "0" : "");
            lastUID += String(rfid.uid.uidByte[i], HEX);
            if(i < rfid.uid.size - 1) lastUID += ":";

        }
        lastUID.toUpperCase();
        Serial.println("New Card Scanned: " + lastUID);

        rfid.PICC_HaltA();
        rfid.PCD_StopCrypto1();
    }
}

```

## 4. What to expect

- ✓ Your phone will connect to the ESP32 WiFi.
- ✓ When you navigate to **192.168.4.1**, you will see a clean dashboard.
- ✓ Tap an RFID card to the reader; the dashboard will update within 2 seconds (due to the auto-refresh) showing the hexadecimal UID.
- ✓ The Serial Monitor will also display the IDs as a backup.