

## Automatic Attendance Tracker

### Project Overview

This Arduino-based project will be focused on making an attendance checking system to efficiently take attendance in a SMART classroom.

Course Connections	21 <sup>st</sup> Century Skills	CTE Alignment
Middle School Science High School Physics High School Engineering	Communication, Critical Thinking, Collaboration	Energy, Environment, and Development Pathway

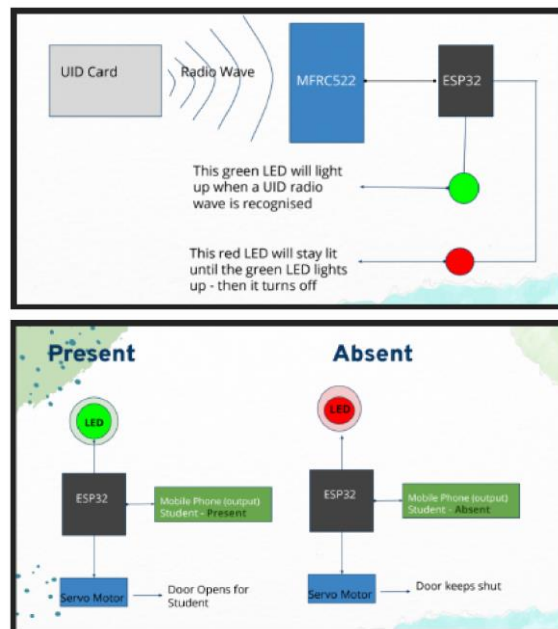
### Objective

Students will make an attendance checking system to efficiently take attendance in a SMART classroom.

### Materials

1. ESP32 Microcontroller
2. MFRC522 Module
3. UID Card
4. 2 LED's, 1 green and 1 red
5. Servo Motor
6. Breadboard
7. Wires (Male to female)
8. USB Cable
9. Computer

## Block Diagram



## Application

Traditional attendance can be time-consuming, with teachers roll calling or using a seating chart to ensure students are present. This process is known to shift the focus and learning in the classroom. An automatic attendance checker is not time-demanding and allows teachers to focus on instruction. The automatic attendance checker allows for a smooth learning process and greater security in the classroom. This system could also be used for restroom breaks. When a student leaves the classroom and how long it took for the restroom break.

## Outline/Schedule

### Part 1: Introduction to Automatic Attendance Checker (complete by Day 1)

- Introduce the problem of traditional attendance taking in the classroom and how it can be time-consuming and distracting from instruction.
- Introduce the solution of an automatic attendance checker and explain the benefits of using it.
- Discuss the materials needed for the project, draw the block diagram, explain to them how the system works.
- Assign team tasks, distribute materials, and plan the schedule.
- Record video clips for important steps.

**Part 2: Programming the ESP32 Microcontroller (complete by Jun. ??)**

- Introduce the ESP32 Microcontroller and its capabilities.
- What issues one will run into when programming. ie syntax errors...
- introduce some problems in programming and have students figure out what the solution is
- Review the basics of programming with Arduino and ESP32.
- Install the MFRC522 library and show the connection between ESP32 and MFRC522 module.
- Demonstrate how to program the ESP32 to read RFID tags with the MFRC522 module and output the results to the serial monitor.
- Allow team time to program their microcontrollers and test them with the RFID module.
- Record video clips for important steps.

**Part 3: Adding a Servo Motor (complete by Jun. ??)**

- Introduce the Servo Motor and how it can be used to control the opening and closing of a door.
- Install the ESP32Servo library.
- Demonstrate how to connect and control a Servo Motor with the ESP32 Microcontroller.
- Review what problems one can run into when connecting wires and motors into the system
- Allow teams time to add a Servo Motor to their project and test it.
- Record video clips for important steps.

**Part 4: Adding LEDs and Blynk Integration (complete by Jun. ??)**

- Introduce the use of LEDs to indicate the status of the attendance checker.
- Demonstrate how to connect and control LEDs with the ESP32 Microcontroller.
- Mention to students the problem one may run into due to polarity of leds
  - Reminder for LED's polarity
- Install the Blynk library and how it can be used to display the attendance status on a mobile app.
- Allow teams time to integrate LEDs and Blynk into their project and test it.
- Record video clips for important steps.
- (Optional for Restroom Break) Add timer for when student is on restroom break

**Part 5: Finalizing and Testing the Project (complete by Jun. ??)**

- Design the testing plan based on the project objectives and requirements.
- Discuss the troubles that students may encounter and how to troubleshoot.
- Allow teams time to finalize their project and test it according to the testing plan provided.
- Have teams present their projects and demonstrate their functionality.
- Record video clips for important steps.

- Prepare the project presentation and make the video demo.

### Reflection

Have students reflect on their experience with the project and what they have learned about programming, robotics, and Arduino.

- Discuss how STEM skills and knowledge can be applied to real-world problems and applications.
- Allow time for students to provide feedback on the project and suggest improvements for future iterations.

### Sample Code

[Sample Code](#)