# Dawson College

# Electrical Engineering Technology Department

# Introduction to Internet of Things

## Project Name:

### Smart Alarm Gate

## Team Members:

### Sunit Patel (ID: 2339661)

### Eric John Sinocruz (ID: 2438740)

### May 12, 2025

## 2. Project Description:

**Charlie**, a museum owner, wants to showcase his ruby without a case on top of it to show its beauty to the visitors. However, leaving it open can introduce to many problems like thievery.

To make sure the ruby stays safe, while still being able to show its precious stone to the world, we designed a ***Smart Alarm Gate*,** it uses an ***ultrasonic*** sensor to monitor how close someone can get to the ruby. When the system detects an object (e.g. hand) approaching the ruby, it will activate a ***Micro Servo*** to rotate which acts like a gate that will automatically close and activate an ***LED*** and ***Buzzer*** which will notify the guards. It also has an ***LCD*** screen which will warn the individual when they are too close.

This helps the museum owner attract more visitors and making more profit while also keeping his jewelry safe.

**How it works:**

* The **ultrasonic sensor** measures the distance between the visitors and the ruby.
* If the visitors get to closed to the ruby, it will trigger the **Micro Servo** which will lower the gate, it will trigger the **Buzzer**, and the **LED** will start to flash red to alert any nearby guards. It will also display a message on the **LCD** screen.

**Final assembly diagram:**

A close-up of a barrier

AI-generated content may be incorrect.

**Figure 1 – Project Diagram**

## 3. Circuit Diagram:

**Inputs:**

* Ultrasonic sensor (HC-SR04)

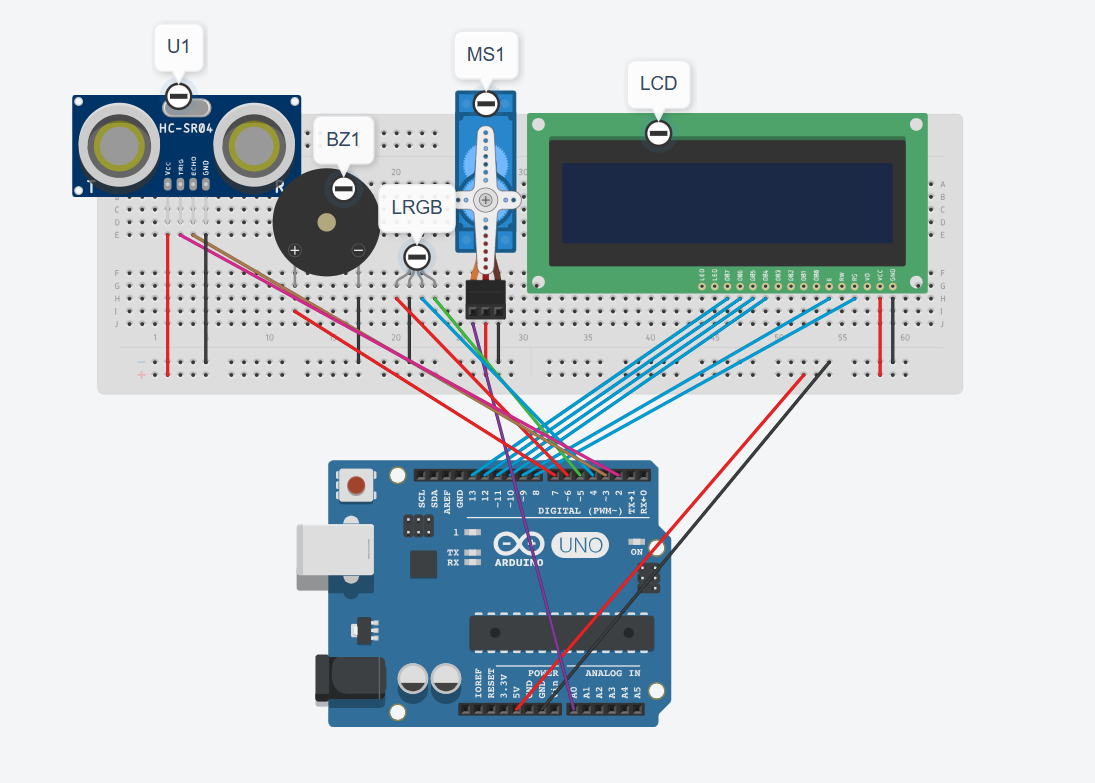
**Outputs:**

* LED
* Buzzer
* LCD
* Micro Servo

This table below will help you understand the hardware connection.

Table 1 – Hardware Connections

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Part** | **Arduino Uno Pin** | **Notes** |
| **U1** | **HC-SR04 Ultrasonic Sensor** | **Trig → D2 (Pink) Echo → D3 (Brown)** | * **Vcc to the + terminal on the breadboard** * **GND to the – terminal on the breadboard** |
| **BZ1** | **Piezo Buzzer** | **Positive → D7 (Red) Negative → GND (Black)** | * **Vcc to the + terminal on the breadboard** * **GND to the – terminal on the breadboard** |
| **LRGB** | **LED RGB** | **Red → D6 (Red)**  **Cathode → GND (Black)**  **Blue → D4 (Blue)**  **Green → D5 (Green)** | * **There’s only the cathode which goes to – terminal on the breadboard** |
| **MS1** | **Micro Servo** | **Signal → A0 (Purple)** | * **Power to the + terminal on the breadboard** * **Ground to the – terminal on the breadboard** |
| **LCD** | **LCD 16 x 2** | **RS → D8 (Blue)**  **EN→ D9 (Blue)**  **D4 → D10 (Blue)**  **D5 → D11 (Blue)**  **D6 → D12 (Blue)**  **D7 → D13 (Blue)** | * **Vcc to the + terminal on the breadboard** * **GND to the – terminal on the breadboard** |
| **—** | **+5 V Rail** | **(red)** | **Feed Vcc to the + rail on the breadboard.** |
| **—** | **GND Rail** | **(black)** | **Connect the GND to – rail on the breadboard to connect all the GND together** |

****

*Figure 2. Smart Alarm gate Circuit built on Tinkercad*

## 4. Code documentation

**Uses 3 external libraries**

* NewPing.h - Ultra Sonic Sensor
* LiquidCrystal.h - LCD 16 x 2
* Servo.h - Micro Servo

**Setup function**

* Sets up LCD screen
* Starts LCD screen
* Sets up motor for servo
* Sets up initial position for servo
* Prints startup message on LCD (e.g. “Getting ready...”)
* Configures pins as either INPUTS or OUTPUTS

**Loop function**

* Measures distance using Ultra Sonic Sensor
* Triggers LED
* Triggers Buzzer
* Triggers Servo
* Display different messages on LCD

Function descriptions:

**void** **setup()**

* Initializes LCD, servo motor, and other pins, and sets up their communication with the computer.
* Displays initial message on LCD
* Sets Servo at initial position

**void** **loop()**

* Measures distance with Ultra Sonic Sensor in “cm “
* Prints messages to serial monitor on Arduino app
* If statement (distance is lower than 15)
* Displays “! Warning!” message on LCD
* Triggers LED and Buzzer, blinking and beeping 5 times,
* Servo is positioned at 90 degrees
* Else statement (otherwise)
* Displays “Do not Touch” message on LCD
* Only Green LED is on
* Servo is positioned at 0 degrees

Sonar.ping\_cm() - function to help sensor measure distance in “cm”

Void flashRedLEDandBuzzer (int times, int interval) - function that lets LED and buzzer to flash and beep X times at X interval (e.g. 5 times, 500ms delay)

**How sensor works in code:**

* Trigger pin receives a signal
* Echo pin uses that signal to calculate the distance between the sensor and an object
* The time it takes for the signal sent from the trigger pin to return is how distance is calculated

How output devices work in this project:

**LCD**

* Used to display warning and alert messages depending on the state

**RGB LEDs**

* Green light means that there is no object detected
* Red light means that is an object detected, which will trigger the alarm

**Buzzer**

* Works alongside Red LED
* Turns on when there is an object detected / alarm is triggered

**Servo**

* Works alongside Red LED and Buzzer
* Acts as a gate in our project
* Closes when object is detected
* Opens when no object is detected

## 5. Ethics, Privacy, or Security Disclaimer:

This project was **primarily designed for safety purposes, in our case, to protect valuable and beautiful jewelry in a museum setting. With a few adjustments, it can also be adapted for home use. Instead of activating a gate, the system could trigger automatic locks on doors to help prevent threats such as robberies or home invasions.**

**We prioritize user safety and privacy. The Smart Gate does not collect or store any personal data. It simply detects the presence of objects using an ultrasonic sensor, which means there are no privacy concerns. As a result, the Smart Gate is a user-friendly and secure device that helps protect property while keeping your personal information completely private**

## 6. Resources

**Basics to controlling a servo:** [**https://docs.arduino.cc/tutorials/generic/basic-servo-control/**](https://docs.arduino.cc/tutorials/generic/basic-servo-control/)

**Basics to using an LCD 16 x 2:** <https://dawsoncollege-lea.omnivox.ca/cvir/dtrv/ReadDocumentTravail.aspx/10_-_Visualization.docx?idtravail=d20f12c6-3269-494f-bd38-98abb9be55d1&iddocumenttravail=ad80e28e-bddd-405e-b2f8-45a7ddd98381&C=DAW&E=P&L=ANG&Ref=20250512224425&SID=e3cc2c46-164e-4700-a577-2ce7a224894d&Info=YmhBZTE2T3NWRWpLNFBGMzVONmY2NlV2TVdTaXZCMWpxckZYaitYQSt6bTNyYUkwNmZwM3VObWUyY24zakVZVDBUSnIzZDFwaGlHeW9jaHdzNGVWM01YZ0tJcnRBSFJCbDl1YkZNZ0dqR0swYnBiREJUMWU4bmJORXVNNEljZktoSzFOMkZxLy9uR285enJ5RUhucVJOSjdiS21zSmZ6aW0rSHA2YTZVWEl3PQ__>

**Basics of Arduino**: <https://docs.arduino.cc/language-reference/en/variables/data-types/int/>

**Basics to using an ultra-sonic sensor**: <https://dawsoncollege-lea.omnivox.ca/cvir/dtrv/ReadDocumentTravail.aspx/6_-_Libraries.docx?idtravail=2189861d-49be-4240-a9b6-2a9fcced95f5&iddocumenttravail=044ae804-42d3-4131-848f-5959f0782b6d&C=DAW&E=P&L=ANG&Ref=20250512224806&SID=e3cc2c46-164e-4700-a577-2ce7a224894d&Info=MFBwMzIyQUl0MzA1ZEN3aWlVYWloME1KS3N1NVQzdDV4ajc3RWpmT2tzNGpsSzJ0cGpRd2lCU0xVbU9hNW1UT0pvK25GWWNteFpDVXJWT0llWlNUdllROWoweHZNanZPZHdBalBHdjYvWnZQNndqenZ1MmZHZlo0blNiVjU0RFNwdStZeExNWURkQTQ3MHFZbkJmbnpGMjNWcDlTOG5OMk9FZ3Y1T2N2RWo4PQ__>

**Circuit Diagram** built with Tinkercad :

<https://www.tinkercad.com/things/ipvSpPueFIU-programming-the-arduino-simulator/editel?returnTo=https%3A%2F%2Fwww.tinkercad.com%2Fdashboard>