Building a Security System Using Arduino

**Project Description**

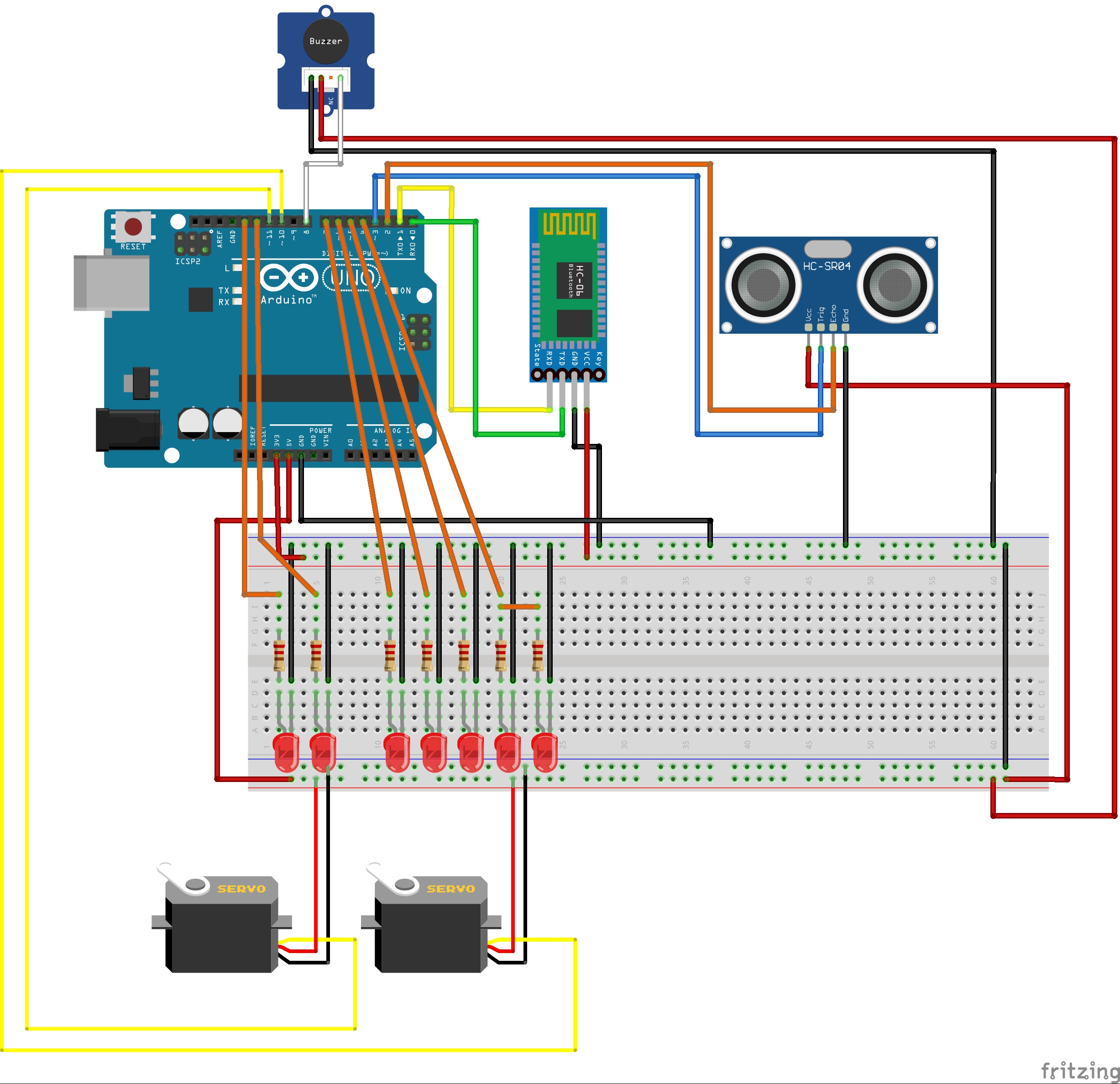
Smart Secure is an easy and innovative way to protect your home. This project is an alarm system used to sense, alarm, and notify the homeowner when an intruder is near. In further detail, the main components of our security system are servos, an ultrasonic sensor, and an alarm. These three pieces of hardware all tie together when then ultrasonic sensor is turned on and within its radius. The sensor sends out a soundwave as a specific distance (measurement). If it detects a change in the sound wave distance (someone is standing in the way) it knows that something is in the way hence the change in distance, and it will sound the security alarm. Access the alarm system via the **House Control** app or on a computer (changing a simple symbol in the code). Utilize the app to turn on your home lights, open your garage door or to open your front doors.

**IDE Used: Arduino IDE**

House Dimensions: 20” x 20” x 2.7”

Note: App is directly downloaded to phone from laptop. The app is only compatible with Android.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Device** | **Use(s)** | **QTY.** | **Cost** | **Price** |
| Alarm Buzzer | When an intruder tries to enter the house | 1 | Own | $2.53 |
| Servo Motor | Operate door | 2 | Own | $10.00 |
| Ultrasonic | To detect if someone is in the house when it is locked | 1 | Own | $2.61 |
| Bluetooth module | To control lock and contact phone | 1 | Own | $5.39 |
| LEDs | Turn on lights in various rooms | 7 | $1.75 | $1.75 |
| Arduino IDE | To program | 1 | Own | $32.18 |
| Foam Core House | Location of sensors | 1 | Own | $1.75 |
| Breadboard | Wiring devices into | 1 | Own | $10.95 |
| Resistors (220Ω) | To control voltage to the LEDs | 7 | $0.35 | $0.35 |
| **Total** | | | $2.10 | $67.51 |



|  |  |
| --- | --- |
| **Device** | **Port** |
| Echo (sonar) | 2 |
| Trigger (sonar) | 3 |
| Hall LED (1 and 2) | 4 |
| Garage LED | 5 |
| Dining LED | 6 |
| Kitchen LED | 7 |
| Buzzer | 9 |
| Garage Door | 10 |
| Front Lock | 11 |
| Living LED | 12 |
| Bathroom LED | 13 |

# User Manual

## Available Services

|  |  |  |
| --- | --- | --- |
| **Service** | **Device(s)** | **Use** |
| Turning on Lights | - LED | By pressing a button on your app, you can turn on and off the lights in specific rooms. When you’re leaving your house, just press one button to turn all remaining lights off. |
| Front door lock | - Servo motor | When you have left the house or coming back in with your hands full you can unlock your door without having to pull out your keys |
| Intruder Alarm | - Ultrasonic Sound Sensor  - Alarm Buzzer | An alert is sent to your phone when the sensor picks up movement when your house is locked. Sets off an alarm when there is movement in the house when it is locked |
| Garage Door | - Servo motor | To open and close your garage door on your phone |

## Connecting to the Home

1. Download our app, “House Control”
2. Turn on your phone’s Bluetooth
3. Select the home and enter in the password (0827 for ours)
4. Open the app
5. Press connect and select your home

## Using the App

1. On the top left of the screen, click the “Connect” button
2. From the menu, select your house name. The default name is “House”
3. If you connect successfully. The text on the screen will change from “Not Connected” to “Connected”
4. If the connection is unsuccessful, disconnect and reconnect. Check your Bluetooth connection and make sure you are connecting to the correct device.
5. Press anyone of the buttons to activate a function and press it again to turn it off.

**Difficulties**

* Configuring the Bluetooth module and giving it a password

*Bought and connected it to a TTL to USB which is made specifically for*

*configuring wireless devices.*

* Figuring out how to draw the wire diagrams

*When referencing other wire diagrams, finding and utilizing a program that configures the hardware for this specific project (buzzer and Bluetooth module) was not found.*

* What resistor for the LEDs

*Research by finding tutorials and reading through forums of what works*

* Sending signals to the Arduino

*The Arduino was receiving signals but in the occurrence of receiving new data, the device would not get rid of the previous data that was recorded in order to start reading the new data. To fix this we moved where it cleared the previous data and how it did so.*

* Figuring out how the garage door will work

*As we did not have a roof attached, the garage door was difficult to open. Since the servo could only rotate to a certain degree, we could not manipulate the servo to open and close the door. When adding a pulley system, the door was opened but not fully. We determined that we needed another circular disk beneath the first one in order to create enough room (distance) for the string to completely stretch out. [The door still does not open fully]*

* Figuring out how the front door lock will work

*As we did not have physical ‘door knobs’ on our model, fixating a lock on the door became an issue. As we did not have an actual lock, using the servo like a switch was the best alternative. The servo can rotate to a certain degree. We used this to our advantage as a lock as the door only open in one direction*