

Wifi, NFC, knock sequence enabled lock device

**By
Yujia Dong
Boyu Li
Jiexin Lyu**

**Final Report for ECE 445, Senior Design, Spring 2016
TA: Vivian Hou**

**May 2016
Project No.64**

Abstract

Home security and privacy matters more and more in modern life. Carrying multiple keys around, or be present when giving access are sometimes inconvenient, especially when you have many guests coming in and out or have multiple rental properties. We want to solve this problem by designing a set of lock and key (your smartphone) that allows user to remotely lock and unlock the doors/drawers from their smartphone and manage user access, track usage activity.

This project intends to realize a smart lock device that allow users to open their door without a physical key in multiple ways, and ensures safety and convenience at the same time. For this device, we have three different methods for users to unlock. First one is to unlock with the NFC chip built in user's smartphone, and user can unlock the door by placing their NFC enabled smartphone within 10 cm (4 inches) of the lock. The second method is unlock with wifi connection and mobile app. The mobile app allows user to manage user access, track usage activity, and unlock their lock with wifi connection. When the mobile app connects to the network and sends a data pack to the microcontroller, the microcontroller will check whether the encryption matches and decide whether to unlock the door. Finally, user can also unlock by knocking a preset knock sequence, the vibration is determined by piezo buzzer and the time intervals between each knock are processed by microcontroller. There is also LED indication whether the user has knocked correct sequence, If there are multiple failed attempts, the lock will produce alarm sound with piezo buzzer.

Table of Content

1. Introduction	3
1.1 Motivation and Objectives	3
1.2 Functions and Features	3
1.3 Installment	4
2. Design	5
2.1 Block Diagram	5
2.2 Circuit Schematics	6
2.3 Block Description & Design	7
2.3.1 Power Supply	7
2.3.2 Microcontroller	8
2.3.3 Sensor	9
a. Piezo Buzzer	9
b. Wifi Sensor	9
c. NFC Sensor	10
2.3.4 User Interface	11
2.3.5 Alert	13
2.3.6 LED	14
a. LED Matrix	14
b. LED Indicator	14
2.3.7 Lock	14
3. Design Verification	16
3.1 Verification for each component	16
3.1.1 Power Supply	16
3.1.2 Microcontroller	16
3.1.3 Sensor	16
a. Piezo Buzzer	16
b. Wifi Sensor	16
c. NFC Sensor	16
3.1.4 User Interface	16
3.1.5 Alert	17
3.1.6 LED	17
a. LED Matrix	17
b. LED Indicator	17
4. Cost	18
4.1 Parts Costs	18
4.2 Labor	19
5. Conclusion	20
5.1 Accomplishments	20
5.2 Uncertainties	20
5.3 Ethics Consideration	20
5.4 Future work	20
Reference	
Appendix	

1. Introduction

1.1 Motivation and objectives

Safety and privacy is a big issue for all of us, especially when we consider the case of our home. Traditionally we use a physical key to open the door, but carrying physical “key” itself could be a problem. What if someone lost the key? What if the key is locked inside the unit? Our motivation is to build a keyless lock device that could be opened by three different approaches: special knocking pattern, Wifi or NFC.

With our device users could open the door in three different ways without a physical key. All of the unlock method are relatively independent so users have more options of unlocking the door. There are many scenarios where this device could be practical:

1. When host want to remotely open the door for someone, he/she could open it using web app through Wi-Fi.
2. When host want to share/revoke access, he/she could either share the secret knocking pattern, or hand out a set of username and passcode through the web app.
3. For host himself, he could use NFC card to open the door fast and conveniently.
4. In case that any of the above unlock method is not working, there are always other options, ie, if someone forgets the cellphone, or phone is out of power, or the NFC card is lost, he/she could always unlock by knocking a specific pattern.
5. Easy for host to manage several rental properties, the Airbnb hosts for example.

1.2 Functions and features

The device should achieve following functionalities and features:

1. Could recognize valid knock from user and be able to determine if the knocking pattern matches the preset knocking pattern
2. Could communicate with the web app through mobile phone and unlock per user request
3. Could communicate with NFC device (smartphone NFC chip, NFC card, etc.)
4. Could track unlock activities to increase security
5. Could make an alert sound when people knock the door incorrectly three times
6. Should have user-friendly UI and LED instruction that users could follow easily

1.3 Installment

The device is small and could be implemented inside a normal house door easily. The system consists of mainly six physical parts: power supply, microcontroller, sensor, alert, LED and locking solenoid, and a user interface which is the web application. The physical unit should be inserted inside a door. More specifically, microcontroller (except the switch button) and power supply should be invisible to the user; LED, alert and sensor unit (including buzzer, Wi-Fi sensor, NFC sensor) should be implemented either inside or outside the door depending on functionalities of each component; lock solenoid should be placed partially outside the door to perform locking mechanism. We will discuss the detail implementations for each component in the following sections.

2. Design

2.1 Block Diagram

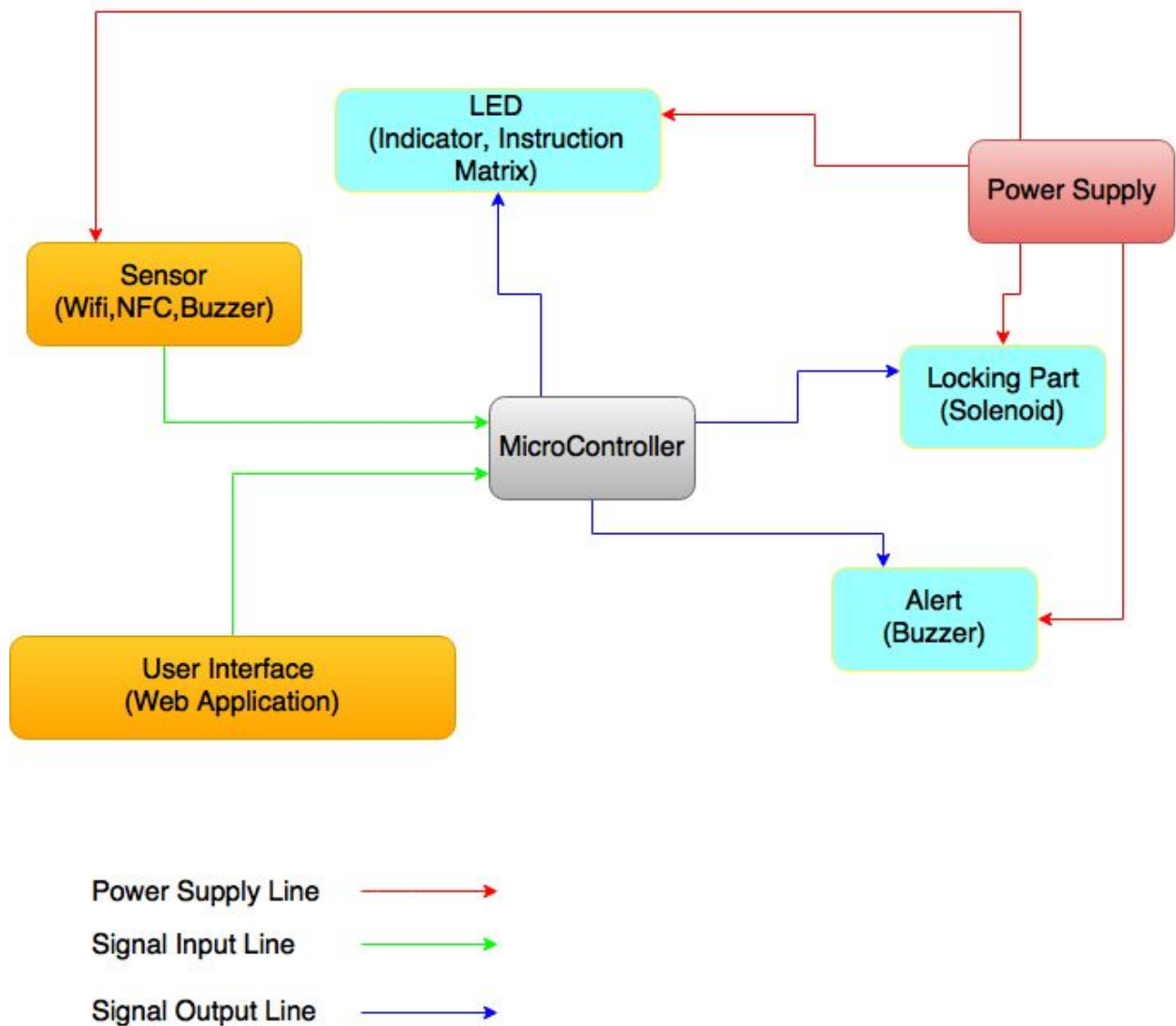


Figure. 1 Block diagram of the WIFI, NFC, knock sequence enabled device

There are five 7 modules in this design: power supply module, sensor module, user interface module, microcontroller, locking part, LED module and alert module. Figure 1 above shows a block diagram and their relations.

2.2 Circuit Schematics

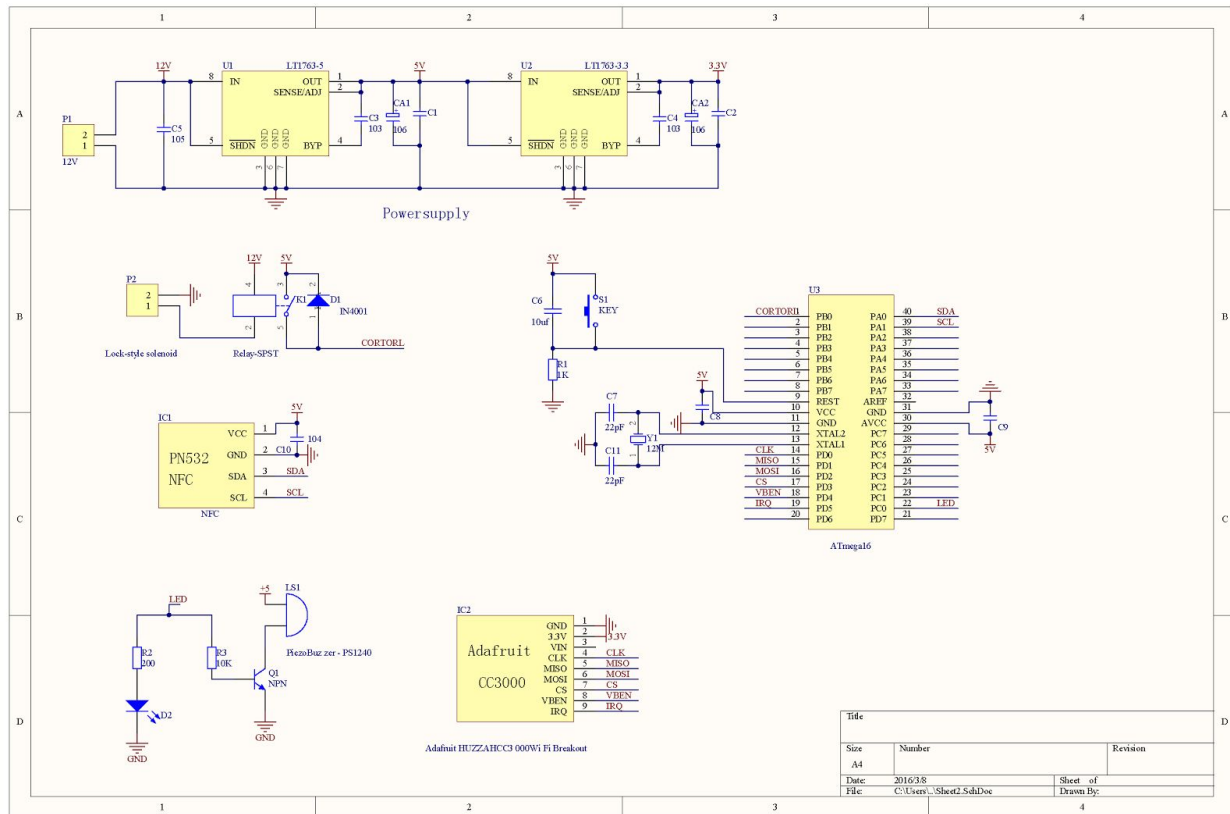


Figure. 2 PCB Schematics for the WIFI, NFC, knock sequence enabled device

2.3 Block Description & Design

The overall system consists of six parts: power supply, microcontroller, sensor, alert, user interface and locking solenoid. We will talk about the technical details about each component.

2.3.1 Power Supply

Description: For the power source, we use home power system by plugging in a wall outlet. The power for the device will come from a 12V power adapter. Since each electrical component

needs different input voltage, we need to use voltage regulator to output 3.3V, 5V and 12V voltages separately from power supply.

Input	12V DC from wall outlet
Output 1	3.3V DC +/- 0.3V to NFC module
Output 2	5V DC +/- 0.3V to Wifi module
Output 3	12V DC to solenoid lock via voltage amplifier

Table 1. Power Supply Overview

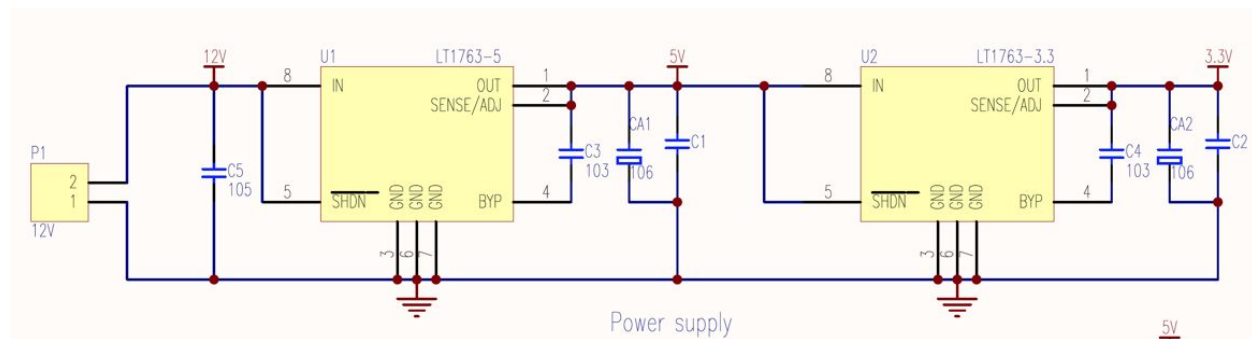


Figure 3. Power Supply Circuit Schematics

2.3.2 Microcontroller

Description: The microcontroller is a programmable bootloader chip from Arduino Uno. It is responsible for communicating and controlling every other electrical components and performing logical decisions. Specifically, the microcontroller controls the on and off state of the solenoid lock, two LED indicators, alert buzzer.

In this project, the microcontroller has 5 tasks. First, every time it detects a valid knock above threshold, it will turn on the green LED, and after listening for the knock sequence, it determines whether the sequence matches the preset secret sequence. If true, then it will quickly flashes the green LED 5 time; if false, it will quickly flashes the red LED 5 times. Second, if the user fails to knock the correct sequence above 3 times, the microcontroller will turn on the alert buzzer for 10 seconds. Third, it sends HTTP request to the web server and pulls data from a specific webpage (from the web app) and check for the special character to determine whether to unlock

the door or not. Fourth, there will be a switch button that controls the system either under knocking mode or Wifi/NFC mode. The microcontroller will get this input and decide which unlock mode user is using. Finally, it reads the data from the NFC tag user uses to swipe the door and verifies the information in the tag to determine whether or not to unlock the door. We are using Arduino IDE to program the chip and a standalone arduino without Arduino Uno board with the rest of the components on a PCB.

Design Components: Atmega328 programmable bootloader microcontroller chip, tactile button switch, 16 MHz crystal clock, 2 220 Ohm resistor, 1 10K Ohm resistor, 2 22 pF capacitor.

Circuit Diagram:

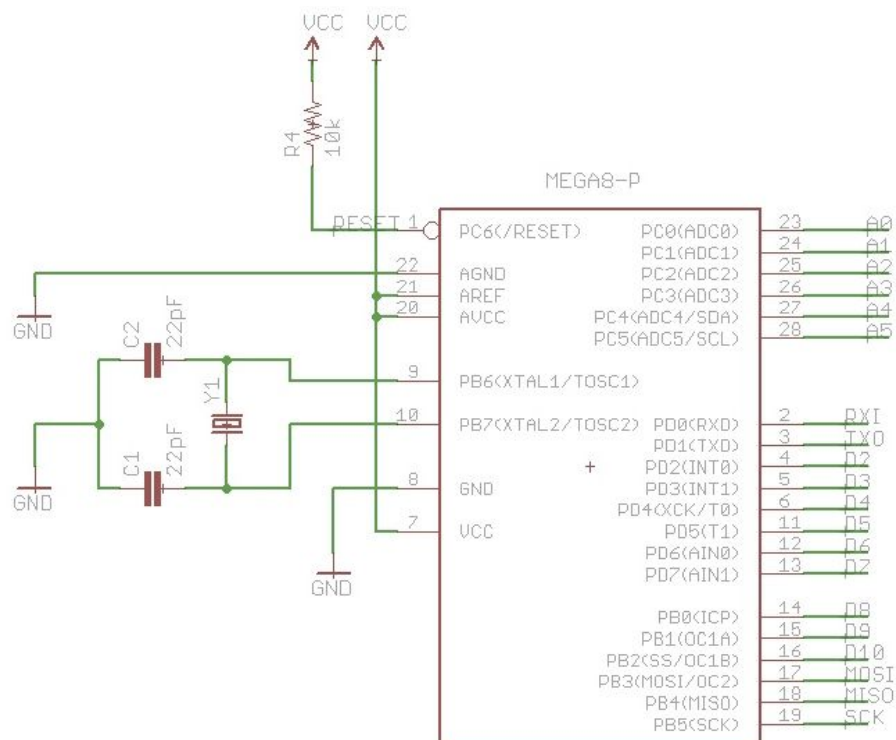


Figure 4. Microcontroller Schematic

Pin Layout:

Digital Pin 0: default unused
 Digital Pin 1: default unused
 Digital Pin 2: to NFC (PN532_SCK)
 Digital Pin 3: to NFC (PN532_MOSI)
 Digital Pin 4: to NFC (PN532_SS)
 Digital Pin 5: to NFC (PN532_MISO)

Digital Pin 6 : to Lock Solenoid
 Digital Pin 7 : to red LED Indicator
 Digital Pin 8 : to Wifi (WINC_RST)
 Digital Pin 9 : to Wifi (WINC_IRQ)
 Digital Pin 10: to Wifi (WINC_CS)
 Digital Pin 11: unused

Analog Pin 0: knock sensor
Analog Pin 1: button switch (used as digital input)
Analog Pin 2: to alert signal (used as digital output)

Analog Pin 3: to green LED indicator
Analog Pin 4: to LED Matrix (DataIn)
Analog Pin 5: to LED Matrix (CLK)

2.3.3 Sensor

The sensor part will receive external signal and send signals to the microcontroller. This part includes piezo buzzer, Wi-Fi sensor, and NFC sensor.

a. Piezo Buzzer

Piezo Buzzer is part of the knocking pattern module. It is responsible for listening to user knocks and transfer knocking signals to microcontroller. Whenever it detects a vibration that cause a voltage peak above threshold voltage, it will set pin signal to HIGH for microcontroller.

b. Wi-Fi Sensor

Description: Wifi sensor is responsible for all the internet related functionalities. It will connect to web server and reads corresponding data from the web page and send corresponding signals to microcontroller once the user has requested to unlock the door.

Algorithm and code snippet:

1. Connect to the internet in setup function.

```
// attempt to connect to Wifi network:
while ( status != WL_CONNECTED) {
  Serial.print("Attempting to connect to SSID: ");
  Serial.println(ssid);
  // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
  status = WiFi.begin(ssid, pass);

  // wait 10 seconds for connection:
  delay(1000);
}
```

2. Read specified web page. If the special character is found on the web page, then send unlock request to microcontroller.

```

while (client.available()) {
  //Serial.println("enter reard");
  char c = client.read();
  //Serial.write(c);
  //pinMode(6, OUTPUT);
  digitalWrite(6, LOW);
  if (c == '~') {
    Serial.print("yes");
    digitalWrite(6, HIGH);
    delay(3000);
  }
  else {
    digitalWrite(6, LOW);
  }
}
}

```

3. Read the web page every 5 seconds.

```

if (millis() - lastConnectionTime > postingInterval) {
  httpRequest();
}

```

c. NFC Sensor

Description: NFC (Near Field Communication) sensor is responsible for NFC communication, pairing, verifying and send correct signals to microcontroller. NFC is a set of short-range wireless communication technologies designed to offer lightweight and secure communication between two devices. In this project, we uses the passive communication method as shown in figure 5, here the NFC tag acts as an initiator that generate RF fields at 13.56MHz and starts the NFCIP. The target, which is the the PN532 breakout board responds to initiator command in load modulation.

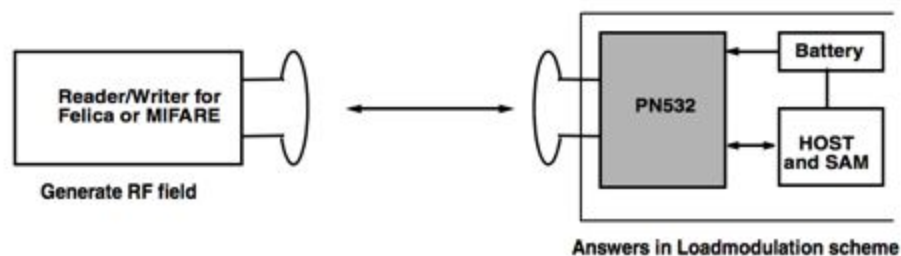


Figure 5. NFC card operating mode with passive communication method

Algorithm and code Snippet:

NFC module will wait for the NFC communication with a NFC card for 1 second in the loop. So every 1 second the function will timeout and go to the next line. If it does receive a NFC communication, it will send unlock request to microcontroller.

```
success = nfc.readPassiveTargetID(PN532_MIFARE_ISO14443A, uid, &uidLength, 1000);  
if(success){  
    digitalWrite(6,HIGH);  
    delay(3000);  
    digitalWrite(6,LOW);  
}
```

2.3.4 User Interface

Through Wi-Fi connection, the user can use their mobile phone to log into the web app to send unlock signal (a special character that will be read by wifi module) to the microcontroller. And as long as the microcontroller reads the matching encoded character, it will set specific digital pin to high and retract the solenoid which unlocks the door.

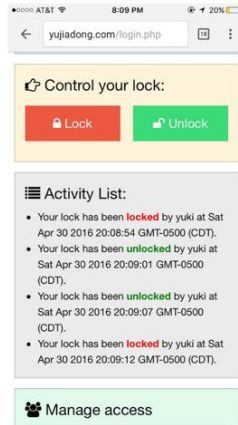
Besides the functionality to control locks, users can also use the app to monitor activities of the lock, for example who has unlocked the door at what time. As illustrated in Figure 6 below.

The app also allows user to manage access of their lock. On the app, if the user is logged in as host, he or she will see an interface where the host can pick any of their friends and families and give them access or revoke their access. It is also possible to schedule the access for specific day and time for extra safety. As shown in Figure 7 below, once the host adds a guest to list, he/she created a set of username and password for the guest. These information is sent to the guest's email address automatically through PHP request, and the guest can then use the given username and password the in email to log in to the app and control the lock.

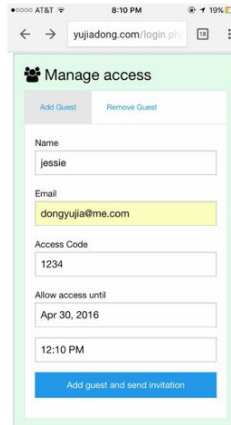
The web app is built with HTML and PHP, so it can be easily accessed through any web browser on any platform including desktop and mobile phone.



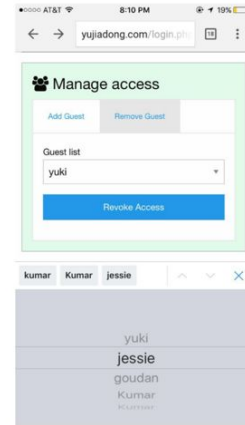
Display User info and lock address



24/7 Activity Log

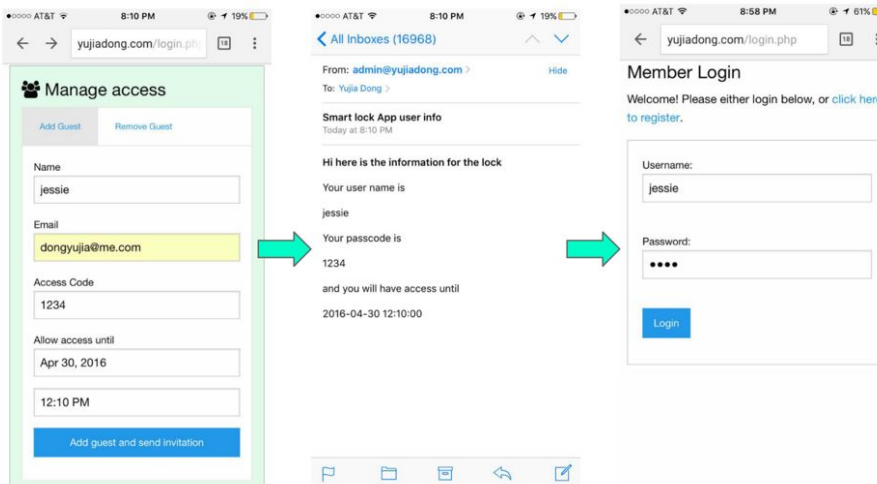


Grant access and set access period



Remove guests from access list

Figure 6. Web application screenshots



Host can send invitation to guest with their username, password and access time period via email.

Then guest can log in to the interface with the username and pw in the email.

Figure 7. How to give other people access to the lock

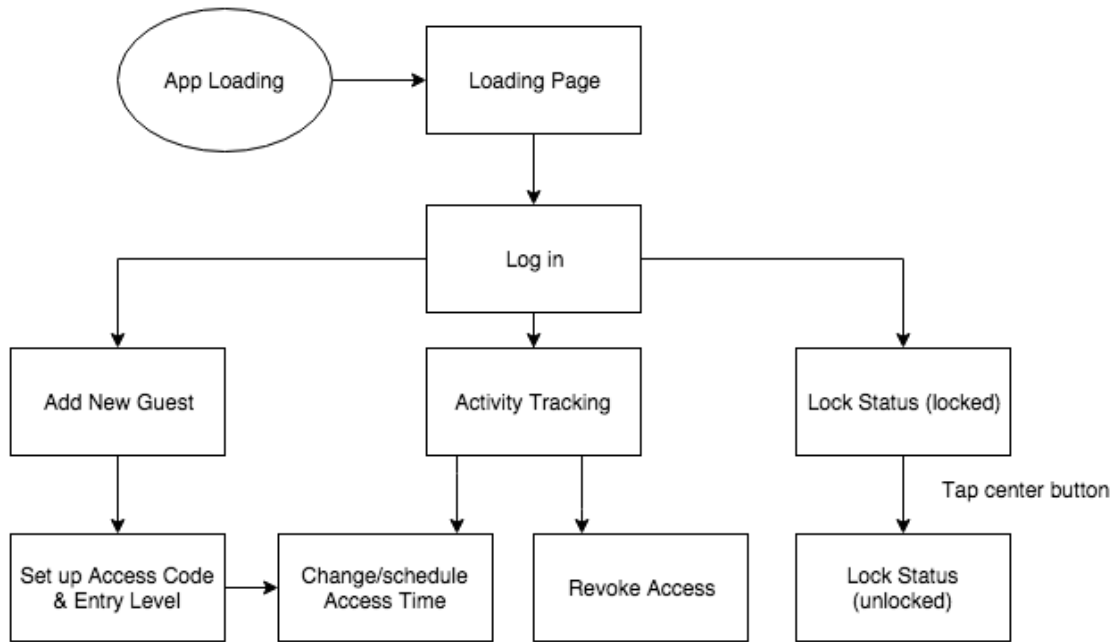


Figure 8. Web application flow chart

2.3.5 Alert

Description: The Alert Unit contains a buzzer that will go off under situation that the system thinks there are someone tries to open the door while he may not be granted with access, ie. knock the door incorrectly for more than three times. The on and off states is controlled by the microcontroller chip analog pin 5 (used as digital pin). When the control signal is at logic-1, the speaker sounds. When the control signal is at logic-0, the speaker stays quiet. The buzzer is driven by 5VDC, an NPN transistor and a 10k resistor.

Circuit Diagram:

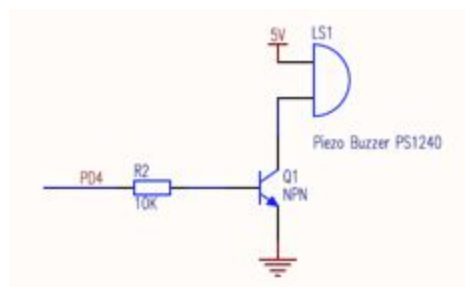


Figure 9. Alert Unit Circuit Schematic

2.3.6 LED

a. LED Matrix

Description: The LED Matrix constantly displays instructions for users to use the device. It constantly displays text about how to switch from knocking and Wifi/NFC mode, ie. press the button to enter knocking mode, release button to unlock with NFC card/web app. This makes sure that the lock only listens for knock when user intentionally pressing the button.

b. LED Indicator

Description: In this design, the green LED is LTL-4233, and the red LED is LTL-4223. The on and off states is controlled by the microcontroller. When the buzzer detects a knock, the green LED will lights up. After listening to the complete knock sequence, the microcontroller will compare the sequence with the preset secret sequence, if there is a match, the green LED will quickly flashes 5 times. Otherwise, the red LED will quickly flashes 5 times to indicate a failure.

2.3.7 Lock

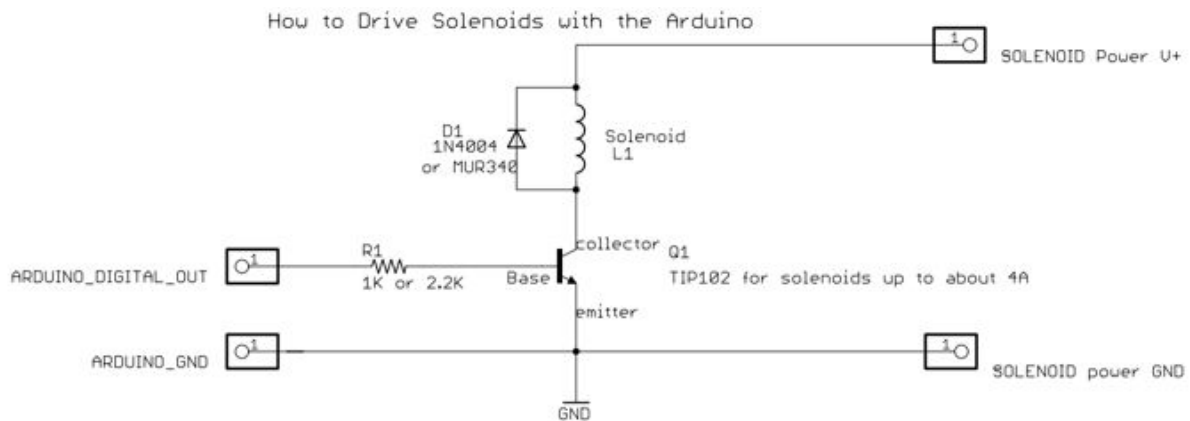


Fig 10. Lock-style solenoid driving unit

Description: The Fig 10 is the schematic of the lock module. As indicated in the datasheet of the lock, the lock open if the voltage across is within the range of 9 ~12 V. Therefore, we decide to connect the positive side of the lock to 12 V. For the purpose of closing and opening the lock based on the voltage level of control signal, a transistor is used. The transistor, in this module, is BJT TIP102 from Fairchild. TIP102 is desirable, since it has high DC current gain $G=1000$ and high maximum collector current $I_C = 8\text{ A}$, and since the lock would draw 1 A when it is activated. A 1 k Ω resistor is connected between the output port of microcontroller and the base of TIP102

to draw heat from TIP102 due to the high current through TIP102. Diode 1N4004 is in parallel with the lock so that prevent reverse current to damage the lock.

For a BJT transistor,

$$\begin{aligned}I_E &= I_C + I_B \\I_C &= \beta * I_B\end{aligned}$$

As indicated in the datasheet of MSP430, the current coming out from MSP430 is 1 mA.

Therefore

$$I_C = \beta * I_B = 1000 * 1 \text{ mA} = 1 \text{ A}$$

From the above calculation, we see that the collector current is 1 A, which satisfies the requirement of the lock.

Besides, from the database of TIP102, we its base-emitter cut-off voltage $V_{BE(On)} = 2 \text{ V}$, which fit our requirement about controlling. That is when control signal is at logic-1, the lock open, and when control signal is at logic-0, the lock close.

3. Design Verification

For detail R&V tables please refer to the Appendix.

3.1 Verification for each component

3.1.1 Power Supply

1. Input voltage is 12V
2. Output voltage are 3.3V and 5V

3.1.2 Microcontroller

1. Input voltage is 5V
2. Button controls two operation mode: Knocking or Wifi & NFC.
3. Communication with all the other components, including buzzers, Wi-Fi breakout, NFC breakout, LED, and alarm.
4. Unlock the lock solenoid

3.1.3 Sensor

1. Piezo Buzzer

1. Input voltage 5V from microcontroller digital pins
2. Can detect user knock
3. Communicate with microcontroller

2. Wi-Fi Sensor

1. Input voltage 5V
2. Connect to the internet
3. Read web page data
4. Communicate with Microcontroller

3. NFC Sensor

1. Input Voltage 3.3V
2. Read NFC card data
3. Communicate with Microcontroller

3.1.4 User Interface

1. User can log in and logout
2. First time user (host) can register
3. Button on webpage controls lock signal that read by Wi-Fi module

4. Host can add new guest and specify access time, and guest can receive corresponding email
5. User can remove guest from list

3.1.5 Alert

1. Input from microcontroller analog pin 5 (used as digital pin)
2. Make a high-low sound for three times when bootup the device
3. Make sound when knock sequence is incorrect for 3 times

3.1.6 LED

1. LED Indicator

1. Input voltage 5V from microcontroller
2. Green LED will light on when on knocking mode
3. Green LED will blink once when receive a valid knock
4. Green LED will flash for a few seconds when the knocking pattern is correct
5. Red LED will flash for a few seconds when the knocking pattern is incorrect

2. LED Indicator

1. Input voltage 5V from microcontroller
2. Display instruction for users about switch between knocking and NFC & Wifi modes.

3.1.7 Lock

1. Input voltage 9V
2. Communicate with microcontroller
3. Unlock when digital pin 6 is HIGH

4. Costs

4.1 Parts

Table 2. Parts Costs

Details	Number	Manufacturer	Retail Cost (\$)	Bulk Purchasing Cost (\$)	Actual Cost (\$)
Arduino Uno board	2	TI	30	30	30
Arduino bootloader-programmed chip (Atmega328)	4	TI	25	20	50
Piezo Buzzer - PS1240	2	Adafruit	2	N/A	4
PN532 NFC/RFID controller breakout board - v1.6	1	Adafruit	40	N/A	40
Lock-style Solenoid - 12VDC	1	Adafruit	15	N/A	15
ATWINC1500 WiFi Breakout	1	Adafruit	35	35	35
PCB	2	N/A	30	N/A	30
Electrolytic Capacitors	4	Lelon	0.11	0.06	0.44
8*8 Bi-Color LED matrix	1	Adafruit	5	5	5
Schottky Diode 1N5825	1	Microsemi Corporation	0.96	0.89	0.96
NPN Transistor	1	N/A	0.5	0.5	0.5

LED SSL-LX3044ID-5V	2	Lumex	0.29	0.12	0.58
Ceramic Capacitors T356Axxxx	4	KEMET	0.49	0.32	1.56
Metal-Film Resistors 271-xxx-RC	10	Xicon N/A	0.15 N/A	0.001	1.5
AWG Wire	multiple	N/A	N/A	N/A	N/A
7805 Voltage regulator	1	N/A	0.5	0.5	0.5
78033 Voltage regulator	1	N/A	0.5	0.5	0.5
16MHZ Clock Crystal	1	N/A	2	2	2
Tactile Button Switch	2	Adafruit	2.5 * 20 pack	2.5 * 20 pack	2.5
Total					220.04

4.2 Labor

Each of us will devote decent amount of time on this project, the estimated amount of time we are going to work on this is:

Table 3. Labor costs

Name	Hours/week	Cost
Yujia Dong	8hrs/week	\$500/week
Boyu Li	8hrs/week	\$500/week
Jiexin Lyu	8hrs/week	\$500/week
Total	24hrs/week	\$1500/week

We estimate the hourly rate per person is 25\$. So engineering cost for the team per week with a multiplier of 2.5 should be $2.5 \times 24 \times 25 = \1500 . We worked on this project for 12 weeks, so the total labor cost would be $\$1500 \times 12 = \18000 .

5. Conclusion

5.1 Accomplishments

We succeeded in complete the promised functions in our design including: sending stable control signals on controlling the opening and closed status of lock with the corresponding correct knock sequence input from the user, recognizing the corresponding ID read from the NFC tag with our NFC module, pulling data efficiently from the web server and send that data to microcontroller with our Wifi module, sending indicating LED signals and beeping sound, displaying knock instructions on the bi-color 8*8 LED matrix, and also have an efficient working circuit with low power cost.

5.2 Uncertainties

We have already implemented the three functions for the lock device, but there are some concerns with the wifi module functionalities. So far, our approach for the wifi module is actively pulling data from the web server by sending HTTP request every 5 seconds, this is power consuming and not quite responsive, that is, there is a delay between user hit the button on web page till the data gets read by the lock. We want to work on this problem by using an ethernet shield and micro SD card, using ajax for sending and receiving data between the Arduino web server and web page. By using this approach, the microcontroller can be passively interrupted by the web page instead of actively polling information, and user can get a response immediately after pressing the button on webpage.

5.3 Ethical Consideration

We consult and follow the IEEE code of ethics^[6] to conduct ourselves in an ethical manner of good engineers . Below are a few of the points that are directly relevant to our project.

1. To accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;

Our project aims to create a safe electric lock under low cost, which will help to create a safe and convenient living environment for our users, and their families and friends. We'll take our

responsibilities to make the lock as reliable as possible in order to create this environment for our users.

2. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;

While working on this project, we are open to accept all the comments and criticism from our peer workers and teaching assistants. Based on their feedback and recommendations, we modified our circuit and implementation.

3. to avoid injuring others, their property, reputation, or employment by false or malicious action;

We will make sure that the electrical lock we developed in this project will create a safe living environment for the users and their families and guests and will not have malicious action under any situation to result in any injuries of the property and health of our guests.

5.4 Future Work

We want to improve the security of the lock more, by adjusting the voltage of the piezo buzzer for a louder alert sound, adding another sensor inside the door. In the case when door is jammed or someone is intentionally holding the door, although the slug is out of the solenoid, which indicating the door is locked for the device, the door is not actually locked. Our solution is to add another sensor inside door on the other side to detect the attachment of the slug and the sensor, to make sure the slug is actually inserted into the door and therefore the door is securely locked. We will also try to make the device easier install for homeowners, and make an actual cost and saving market analysis and use the statistics to attract a larger group of customers.

References

- [2] Buzzer circuit diagram, Available: <http://www.eng.utah.edu/~cs5789/handouts/piezo.pdf>
https://product.tdk.com/info/en/catalog/datasheets/ef532_ps.pdf Accessed: 02/16/2016
- [3] Read and write NFC tags with Arduino, Available:
<http://www.allaboutcircuits.com/projects/read-and-write-on-nfc-tags-with-an-arduino/>, Accessed:
02/20/2016
- [4] Schematics of Wi-Fi Shield, Available
<https://www.arduino.cc/en/uploads/Main/arduino-wifi-shield-schematic.pdf>, Accessed:
02/28/2016
http://www.atmel.com/Images/Atmel-42487-ATWINC1500B-MU_Datasheet.pdf, Accessed:
03/05/2016
- [5] Solenoid Circuit Diagram, Available:
http://playground.arduino.cc/uploads/Learning/solenoid_driver.pdf, Accessed: 02/29/2016
- [6] IEEE Code of Ethics, Available: <http://www.ieee.org/about/corporate/governance/p7-8.html>,
Accessed 03/01/2016
- [7] Adafruit Wifi Breakout:
<https://www.adafruit.com/products/1510>
- [8] Adafruit PN532 RFID/NFC Breakout and Shield
<https://learn.adafruit.com/assets/26820>
- [9] NFC Security
<http://events.iaik.tugraz.at/RFIDSec06/Program/papers/002%20-%20Security%20in%20NFC.pdf>
- [9] Buzzer Schematic
<https://startingelectronics.org/beginners/circuits/arduino-buzzer/>
- [10] Piezo Buzzer Data Sheet
https://product.tdk.com/info/en/catalog/datasheets/ef532_ps.pdf

Appendix: R&V Table

Requirement	Verification
Power Supply Unit <ol style="list-style-type: none"> Description: The power supply will provide power for the whole system. The input power will be 12V from 4 AA batteries. Functionality: The power supply for the system, providing 3.3V, 5V, 12V. Technical Requirements: <ol style="list-style-type: none"> 12V input from batteries Use LT1763-5 power regulator to convert 12V input to 5V output. Use LT1763-3.3 power regulator to convert 12V input to 3.3V output. 	<ol style="list-style-type: none"> 12V input verification: Use multimeter, voltage mode . Connect red probe to battery positive black probe to battery negative and test the voltage. (3 points) 5V output verification: <ol style="list-style-type: none"> Use multimeter, voltage mode. Connect red and black cable from multimeter to OUT(pin1) and GND(pin3, pin6 or pin7) of LT1763-5. (3 points) 3V output verification: <ol style="list-style-type: none"> Use multimeter, voltage mode. Connect red and black cable from multimeter to OUT(pin1) and GND(pin3, pin6 or pin7) of LT1763-3.3. (3 points)
Micro-Controller <ol style="list-style-type: none"> Description: The micro-controller is Atmega 328 programmable bootloader. We are using Arduino library to do the coding. Functionalities: The Micro-Controller Unit is responsible for controlling every part of the system. Knock and unknock control: Read input from Buzzer/Wifi/NFC and send output to UI/Solenoid. Technical Requirements: <ol style="list-style-type: none"> Operating voltage: 5+-0.5V Can communicate with Knock Unit Can communicate with Wifi breakout Can communicate with NFC breakout Can communicate with LED Matrix 	<ol style="list-style-type: none"> 1.a Connect Arduino Uno with power source 1.b Turn on multimeter, setup it up to measure voltage 1.c Connect GND and power of Arduino Uno with probes from multimeter 1.d Verify that the voltage is between 5+-0.5V 2 Please refer to the Knock Unit Verification 3 Please refer to the Wifi Unit Verification 4 Please refer to the NFC Unit Verification 5 Please refer to the LED Matrix Unit Verification (3 points)
Knocking Unit	<ol style="list-style-type: none"> Piezo buzzer functionality test

The Knocking Unit consists of a piezo buzzer to detect knocking vibration; a solenoid for lock/unlock mechanism; two LEDs to indicate knocking status; TP120 Power Darlington Transistor to amplify the voltage to solenoid. The system is controlled by micro-controller.

1. Piezo Buzzer

- a. **Description:** Piezo elements work in two modes: If apply voltage, the piezo crystals deform and produce sound. If deform the crystals, they produce a voltage. To use them as a 'knock' sensor, we need to firmly couple the piezo element to the surface that is being knocked.
- b. **Functionality:** Users make knocks on the piezo buzzer, LEDs will indicate whether the knocking pattern is received and correctly, received but incorrect or not received. If received and correct, the solenoid will unlock.
- c. **Technical Requirements:**
 1. Driven by 5+-0.5V power supply
 2. When piezo buzzer detects a knock, the green LED will flash

2. LED Indicator

- a. **Description:** We have two basic LEDs: one red and one green. They are 5mm LED with a red and green lens. They have a typical forward voltage of 2.0V and a rated forward current of 20mA.
- b. **Functionality:** The LED indicators are used for the visual feedback of the correctness of the lock and when each knock is received and when the knock sequence is being evaluated.
- d. **Technical Requirements:**
 1. 1.8-2.2VDC forward drop
 2. Max current: 20mA
 3. Suggested using current: 16-18mA
 4. Luminous Intensity: 150-200mcd

1.a. Solder a pair of 30cm leads to the piezo buzzer.

1.b. Connect it between Arduino analog pin 0 and the ground. Also attach the 1M ohm resistor between Analog pin 0 and the ground.

1.c. Plug Arduino into computer with USB cable and open the Serial Monitor window in the Arduino development environment.

1.d. With the Arduino powered on, tap the piezo buzzer and there is a message says "knock" each time a tap is detected. Stop for a second or two and there should be a message says "knock failed". And if the right knock sequence is tapped, the serial monitor should say "door unlocked". The default knock is 5 consecutive, evenly distributed knock, which can be changed by the user. (4 points)

2. LED Indicator test

2.a. Connect the red LED to digital pin 4 and green LED to digital pin 5 with their corresponding 560 ohm resistors in line.

2.b. After powering the circuit, the green LED lights up.

2.c. Every time the tap is detected the green led would dim.

2.d. After tapping the correct sequence the green led would blink a few times.

2.e. Tapping the wrong sequence would blink the red one. (3 points)

3. Lock-style Solenoid functionality test

3.a. Wire the solenoid with Arduino and power transistor and a diode. The detailed schematic is given in the next section. To connect it, we need a 1 NPN Transistor P2N2222A, 1 Rectifier Diode (1N4001 or similar) and 1 2.2k ohm resistor (1/4 watt).

<p>3. Lock-style Solenoid</p> <p>a. Description: The lock-style solenoid is basically electromagnets. When the coil is energized, the slug is pulled into the center of the coil. This makes the solenoid able to pull from one end. Normally the lock is active so you can't open the door because the solenoid slug is in the way.</p> <p>b. Functionality: It does not use any power in this state. When 9-12VDC is applied, the slug pulls in so it doesn't stick out anymore and the door can be opened.</p> <p>c. Technical Details:</p> <ol style="list-style-type: none"> 1. Voltage 12VDC 2. Draws 650mA at 12V, 500 mA at 9V when activated 3. Designed for 1-10 seconds long activation time 	<p>3.b. Then power the circuit and tap the correct sequence (preset in the code).</p> <p>3.c. The solenoid style lock should retract and unlock.</p> <p>3.d.If not, then should check the connections and the polarity of the diode. (3 points)</p>
<p>Wifi Unit</p> <p>a. Description: The Wifi unit consists of a Wifi breakout board Adafruit ATWINC 1500 WiFi Breakout. It will connect directly to micro-controller.</p> <p>b. Functionality: Wi-Fi can communicate with micro-controller and User Interface(Mobile App), including remote unlock the door.</p> <p>c. Technical Requirement:</p> <ol style="list-style-type: none"> 1. Operating Voltage: 5+-0.5V 2. Wifi Internet connection 3. WebServer creation and connection 4. Lock/Unlock message transmission 	<p>1. 5V power verification:</p> <ol style="list-style-type: none"> a. Connect Wi-Fi breakout to power source and Micro-controller. b. Turn on multimeter and set to voltage mode c. Connect pin1(VIN) and pin2(GND) of Wifi breakout with red and black probes of multimeter separately. d. Verify the Voltage is 5+-0.5V (3 points) <p>2. Pin Layout and Internet Connection verification</p> <ol style="list-style-type: none"> a. Connect Wifi breakout with power and micro-controller: <ul style="list-style-type: none"> • Vin - this is the power-in pin. Connect to 3.3 - 5.5VDC • GND - ground for signal and power • SCK - SPI clock input, 3V or 5V compliant • MISO - SPI data out from module, 3.3V line level • MOSI - SPI data into module, 3V or 5V compliant • CS - Connect to Digital Pin8 of Micro-controller • EN - Enables the entire module, by default tied low with a 100K resistor. Tie to 3-5V to keep the module on all the

	<p>time, connect to a ground signal to disable the module</p> <ul style="list-style-type: none"> • IRQ - Interrupts from the module, connect to your microcontroller's INT input line. 3.3V logic level • RST - Module reset, by default tied low with a 100K resistor. Pull high to bring out of reset. • Wake - wake input signal, used to wake up the module (not used in existing code, but available if you can figure it out!) 3-5V logic in • CFG - allows you to select between SPI (default) or UART data transport. Since we don't have any UART code, keep disconnected • RXD/TXD - UART data transport pins. Since we don't have any UART code, keep disconnected <ol style="list-style-type: none"> On Arduino programming studio run internet testing program, load it to micro-controller The output should be a list of available internet. On Arduino programming studio run internet connection program, insert internet SSID and Password to desired position in script. Load the program to micro-controller. Verify the wifi breakout has connected to the internet by reading the output from output monitor. <p>(3 points)</p> <p>3. WebServer Creation and Connection</p> <ol style="list-style-type: none"> Connect Wifi controller with power and micro-controller Make sure that Wifi breakout is connected to internet(Verification 2) On Arduino Programming Studio run the program for webserver creation, create an web ip address. Load the program to micro-controller. On mobile device/or web browser. Connect to the webserver with ip address(on wifi part, connect to the webserver just created) If use web browser the output will be
--	---

	<p>some verification text. For mobile app output, refer to the User Interface Unit (3 points)</p> <p>4. Lock/Unlock Message transmission verification</p> <ol style="list-style-type: none"> Connect Wifi controller with power and micro-controller Make sure that Wifi breakout is connected to internet(Verification 2) Make sure that the Wifi breakout has created a valid webserver that other devices could connect to. On Arduino Programming Studio run test program for webpage setup. Create a button that if user press then Wifi will send a high signal to micro-controller. On the user side, first connect to the webserver via web browser or mobile app. Then press the button(or other kind of user input). Verify that the solenoid could unlock. (3 points)
<p>Near Field Communication (NFC) Unit</p> <ol style="list-style-type: none"> Description: The NFC part consists of PN532 and 4050 level Shifter Functionality : Bluetooth connection between ios devices and the lock Technical Detail <ol style="list-style-type: none"> Battery Supply Volatge: 3.3V Digital Supply Current: 25mA 	<p>1. NFC responding verification</p> <ol style="list-style-type: none"> Connect Arduino digital pin 2 with 4050 pin 9 Connect Arduino digital pin 3 with 4050 pin 11 Connect Arduino digital pin 4 with 4050 Pint 14 Connect 3.3 Vin with Arduino 3.3V Connect SCK to 4050 pin 10 Connect MISO to Arduino pin 5 Connect MOSI to 4050 pin 12 Connect SSEL to 4050 pin 15 Run the program on the Arduino and use the NFC chip approach the board, it should have the message send out (3 points) <p>2. NFC unlock verification</p> <ol style="list-style-type: none"> Correctly connect the NFC chip with the Arduino and the solenoid Use the chip approach NFC receiver, the solenoid should unlock (3 points)

<p>Alert Unit</p> <p>a. Description: The alert part consist of piezo buzzer, a 10k resistor and transistor</p> <p>b. Functionality: the buzzer will ring when it receive the signal from the micro controller.</p> <p>c. Technical detail: 5V</p>	<p>1a. Connect the one pin of the buzzer with the 5V voltage</p> <p>2a. Connect another pin with the collector pin of the transistor</p> <p>3a. Connect the emitter pin of transistor with the ground.</p> <p>4a. Connect base pin of transistor with digital pin 6 of the Arduino through 10K resistor (3 points)</p>
<p>User Interface Unit</p> <p>a. Description: The most important part of the interface module is the mobile app. This is the key interface where user performs a lot of tasks on. It relies on the information and data received from the wifi module.</p> <p>b. Functionality: Its function includes: lock and unlock the door, grant and revoke access from guests, monitor door status and guest activities.</p>	<p>a. Download the mobile app from Google Play Store and install it on an Android phone. (This could also be an web interface runs on a web server)</p> <p>b. Log in the app/web page with a specific username and password</p> <p>c. Tap the button says “unlock” and the lock should retract and there is a message displayed says “door is unlocked”.</p> <p>D. Later when door is locked by itself, the message on screen should change its status say “door is locked”.</p> <p>E. There is also an activity list that keeps track when the lock status has changed at what time. The two status changed above should be added to the list and displayed on the webpage/app. (3 points)</p>
<p>8. LED Matrix</p> <p>a. Description: The LED Matrix consists of a 8*8 LED Matrix with backpack.</p> <p>b. Functionality: It is served as a unser communication board. Will output message/instruction to user. Connect with micro-controller</p> <p>c. Technical Requirements:</p> <ol style="list-style-type: none"> 1. Operating Voltage 5V 2. Could output some bitmap 	<p>1. 5V power verification:</p> <ol style="list-style-type: none"> a. Connect Wi-Fi breakout to power source and Micro-controller. b. Turn on multimeter and set to voltage mode c. Connect pin1(VIN) and pin2(GND) of Wifi breakout with red and black probes of multimeter separately. d. Verify the Voltage is 5+-0.5V (2 points) <p>2. Bitmap output verification:</p> <ol style="list-style-type: none"> a. Connect to microcontroller and power b. On arduino programming studio, run the program to test bitmap output. Load the program to led matrix

	c. Verify that the LED matrix can show the bitmap. (2 points)
--	--