Multi-Party, Multi-Factor Authentication Lockbox

By

Shelby Doty

Noah Hill

Akshay Sundaram

Final Report for ECE 445, Senior Design, Fall 2022

TA: Zhicong Fan

5 December 2022

Project No. 17

Table of Contents

1 Introduction

1.1 Problem

Governments and government agencies, banks, hospitals, or companies may have rooms, safes, or vaults requiring controlled access to protect their contents. Contents may include sensitive information, security and surveillance equipment and their controls, critical facility infrastructure equipment for telecommunications or power distribution, or hazardous materials. Restricted areas may include server rooms, data centers, or rooms housing industrial control systems. These areas and their contents are prone to physical security attacks such as severance of critical cables, theft of communication equipment, or theft of data servers. Security attacks may be conducted by a malicious insider, resulting in devastating data breaches. According to IBM Security's "Cost of a Data Breach" report in 2020, 10% of malicious data breaches between August 2019 and April 2020 were the result of a physical security compromise while 7% were caused by a malicious insider. The 2021 "Cost of a Data Breach" report by IBM Security noted that the average time to identify and contain a breach caused by physical security compromise was 292 days while a breach caused by a malicious insider took on average 306 days to identify and contain. Existing methods to protect physical systems from malicious insiders include auditing, job rotation, and separation of duties. Auditing access to a restricted area is reactive and does not prevent unauthorized access from occurring. Job rotation and separation of duties only limit prolonged access to certain areas or physical systems.

1.2 Solution

Multi-factor authentication (MFA) is an electronic authentication method used to grant an individual access to an application or place only after successfully presenting multiple factors for verification purposes. Multi-party authorization (MPA) requires multiple individuals to authorize access to an application or place. An example of multi-party authorization usage occurs in banks when one accesses a lockbox. This requires both a bank official and the lockbox owner to act together to open the lockbox.

We propose an electronic lock mechanism providing a proactive approach to physical access control by employing both MFA and MPA methods. Access is granted only when a configurable number of individuals (multi-party) successfully authenticate with an inherence factor and a possession factor (multi-factor). The inherence authentication factor will be a fingerprint, while a one-time numerical token received via SMS serves as the possession factor. This locking mechanism would be applicable to a lockbox in a bank, for example, which already requires multiple parties to authorize access. However, the inherence authentication factor used for authentication in this design, the fingerprint, is not easily lost or misplaced as lockbox keys are.

1.3 Final Design

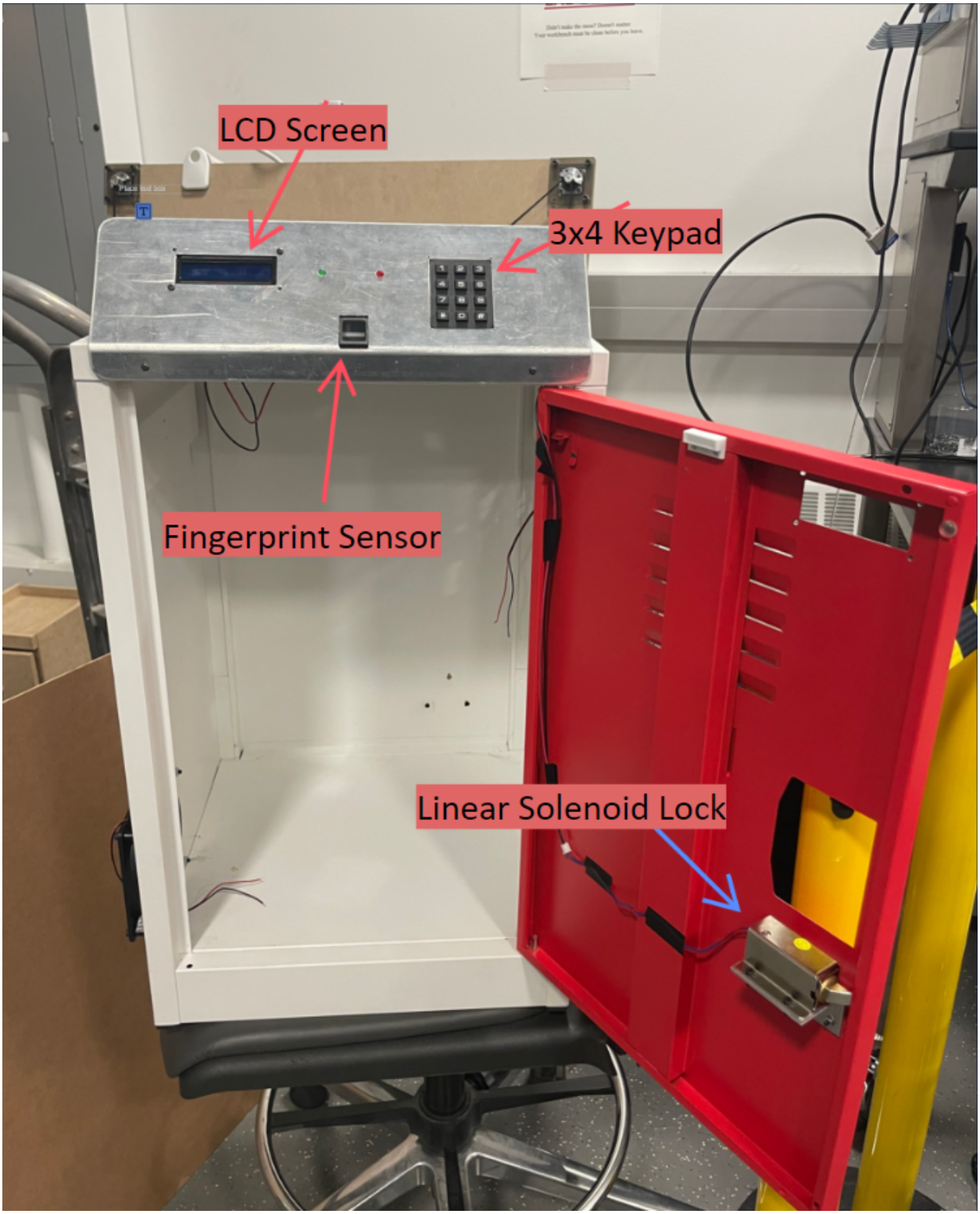


Figure 1: Completed Lock Design

1.4 High-level requirements list

1. **Inherence Factor Authentication**
    a. The system must accurately identify and communicate successful and unsuccessful biometric authentication attempts.
2. **Possession Factor Authentication**
    a. The system must generate and send a token to phones over a SMS API.
3. **Wireless Connectivity**
    a. The system must connect to a WiFi network when powered on.

2 Design

2.1 Block Diagram

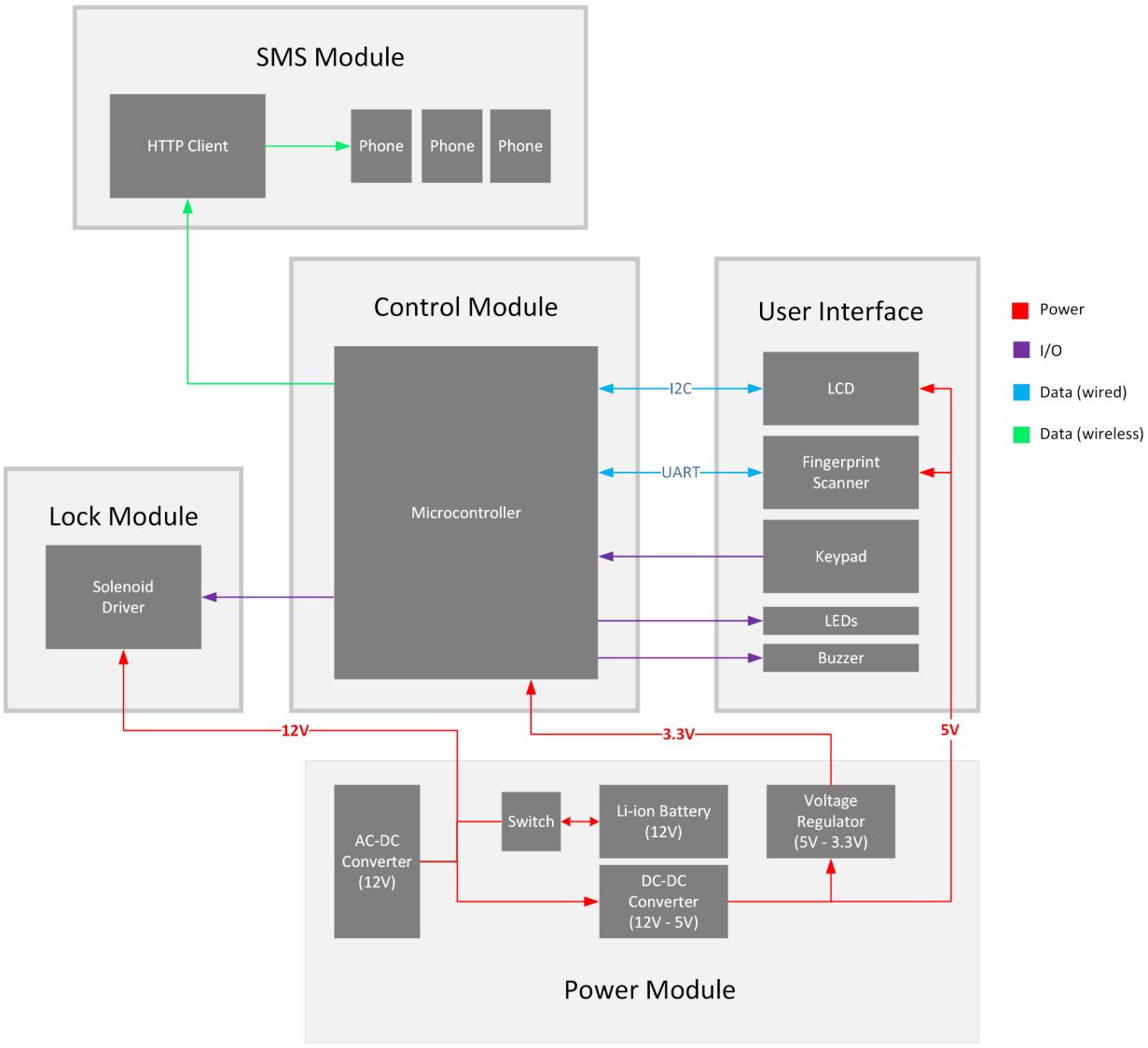


Figure 2: Block Diagram

The critical subsystems depicted in the block diagram include the power supply, control module, web interface, authentication interface, and lock. The power supply includes a battery-backup system, ensuring an uninterrupted source of power for all components in the event the system is unplugged from a regular utility power source. The control module is responsible for seamless communication between all other subsystems as it monitors the status of authentication attempts, engages and disengages the lock, and provides user feedback. The web interface allows the device to be easily connected to a wireless network and contains a client capable of generating and sending one-time tokens via SMS. The authentication interface

consists of the devices utilized for authentication, including a fingerprint sensor and keypad for entering a one-time token. Lastly, the lock subsystem contains the lock hardware and status indicators for user feedback.

2.2 Subsystem Overview

2.2.1 Control Module

The control module receives data from the user interface subsystem via the fingerprint sensor and tactile keypad and sends data to the web interface subsystem over WiFi. The control module controls the LCD screen from the user interface and the solenoid from the mechanical relay & lock status indicator for locking/unlocking. The control module consists of a microcontroller, the ESP32, which uses Wi-Fi connectivity and acts as a TCP client to provide the TCP server with data regarding user identity and authentication success/failure when an authentication attempt is made. Upon successful biometric authentication, the ESP32 microcontroller and the user authenticating receive a one-time token sent via SMS to be input on the tactile keypad. Access is granted/denied depending on whether the user inputs the correct token generated by the TCP server.

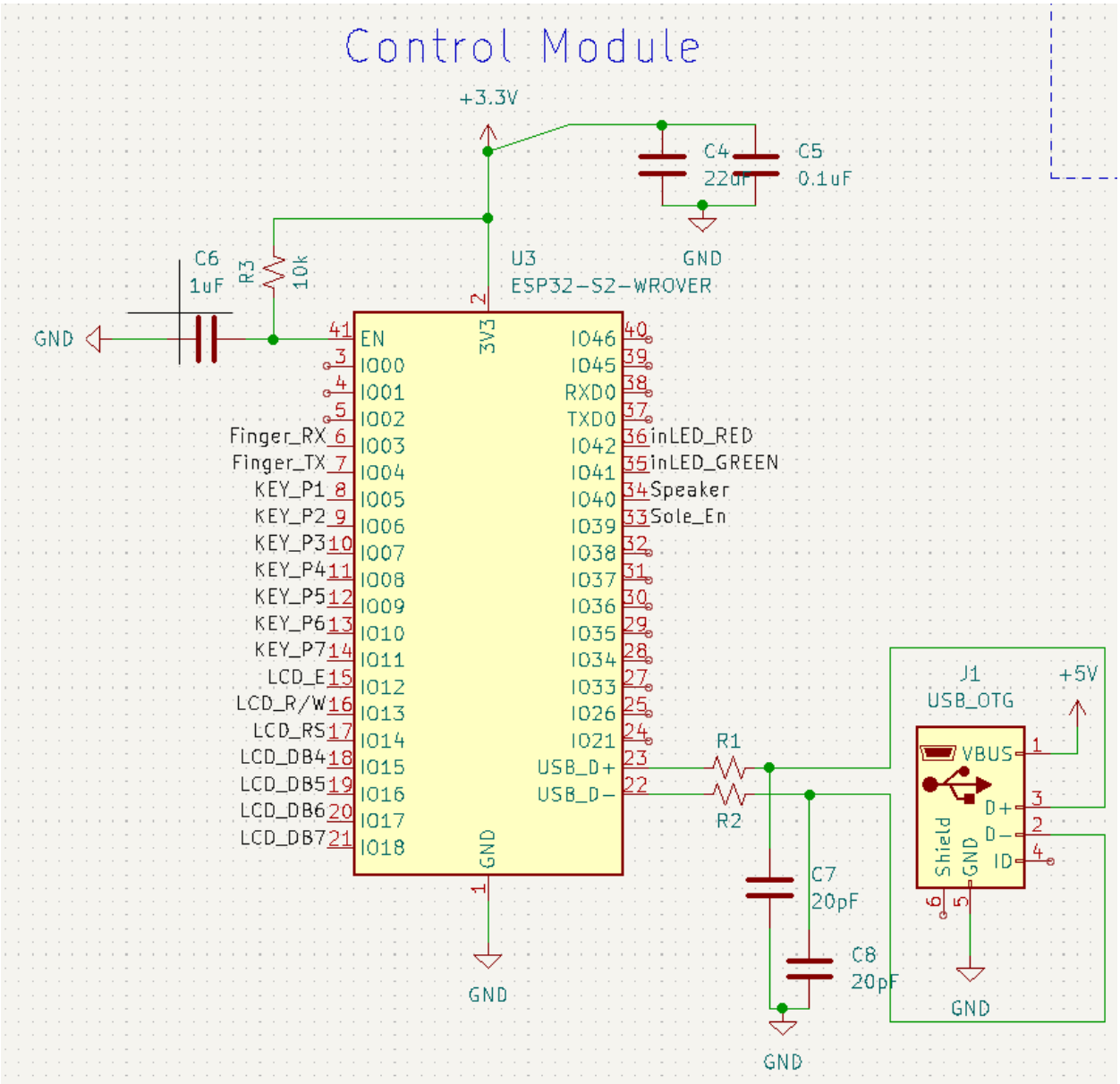


Figure 3: Control Module Schematic

2.2.2 User interface

This subsystem consists of the fingerprint sensor module for gathering biometric data, an LCD screen to display warnings and instructions, and a tactile keypad. The AS608 optical fingerprint sensor module stores biometric data, collects and renders fingerprint images, and matches fingerprint scans with those in storage. A LCD2004 character-type liquid crystal display provides user feedback regarding system status, authentication success/failure messages, remaining successful authentications before unlock, etc. The 3x4 tactile keypad allows users to enter a one-time token received via SMS.

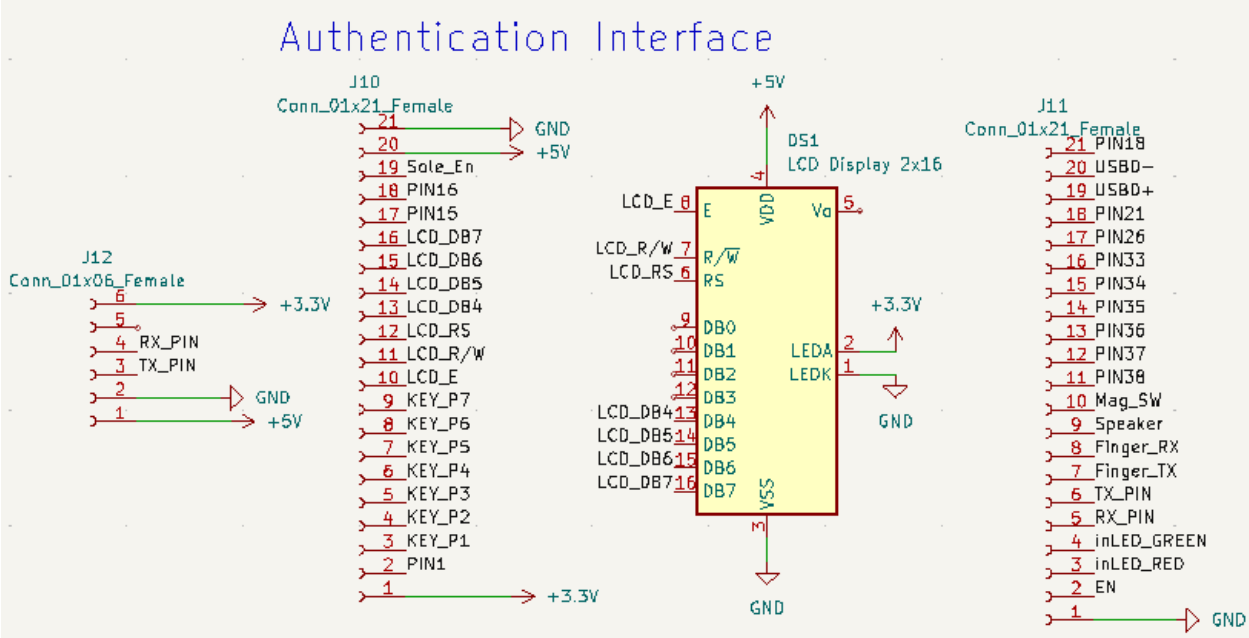


Figure 4: Authentication Interface Schematic

2.2.3 Mechanical relay & lock status indicator

This system will be controlled by the control unit and responsible for securely locking and unlocking the lockbox using a 12V linear solenoid. LEDs indicate to the user when the lockbox is locked or unlocked. A speaker emits a beeping noise if the lockbox door is open longer than a predetermined amount of time. A magnetic contact switch completes a closed circuit when the lockbox door is shut, sending a signal to the control module to emit a sound from the speaker if necessary. Each of these components were chosen because of their relatively cheap costs, and their ability to perform their function while not requiring an excessive amount of power to operate.

A critical feature of our project is engaging and disengaging the locking mechanism which is a simple linear solenoid. We needed to design a circuit that uses a GPIO pin from the ESP32 to control when the solenoid lock opens and closes. Since the solenoid operates at 12V, we could not simply connect it directly to an ESP32 since that can provide at most 3.3V. After researching how to control something like this, we found 2 viable options:

1. Use a relay which takes in the 12V input and 3.3V signal from the ESP32 and if the ESP32 sends a high signal it completes the circuit using an embedded physical switch mechanism that activates the solenoid and opens the lockbox

2. Use a NMOS where the gate voltage comes from the ESP32 and the supply voltage is 12V wired in series with the solenoid such that when the ESP32 outputs a high voltage, current flows through the solenoid, and opens the lockbox.

I decided to choose option 2 since we already have multiple NMOSs from the ECE210 lab kit and none of us have ever used a relay switch before.

To see if this implementation would work or not, we simulated the circuit in LTSpice as shown in Figure 5 below.

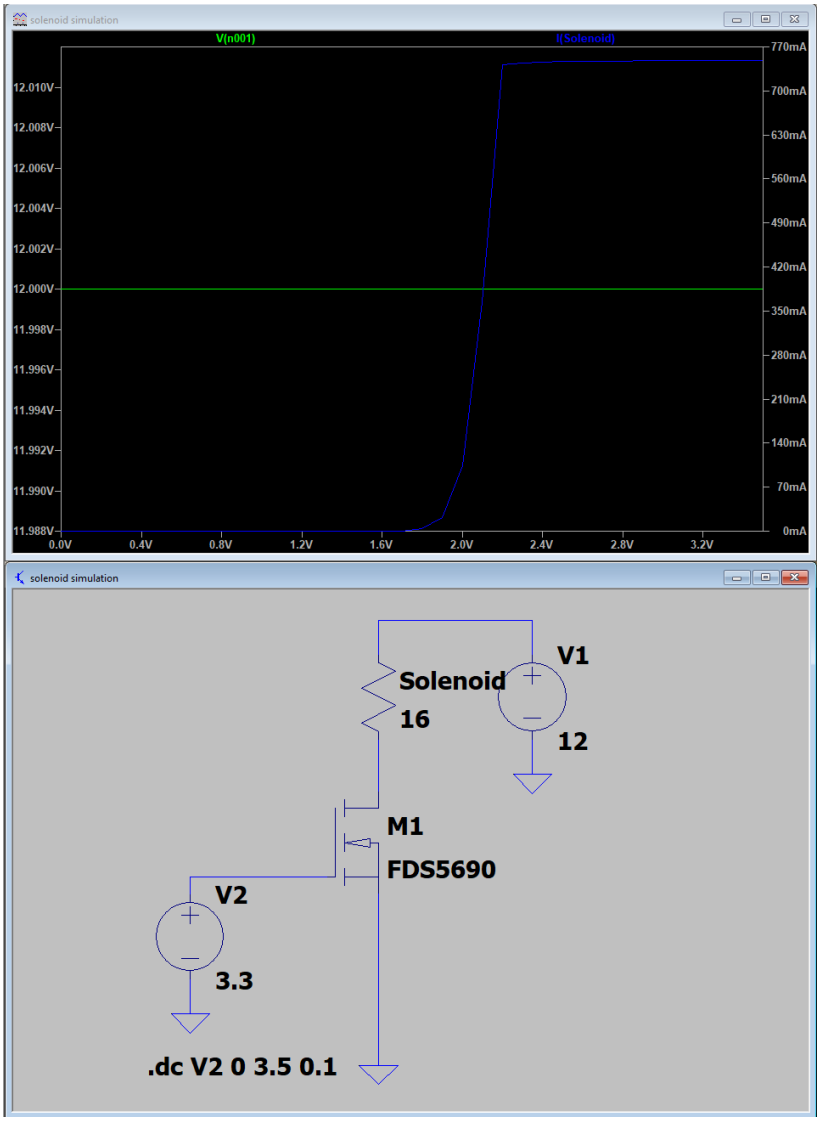


Figure 5: LTSpice Simulation for NMOS

According to the datasheet for the linear solenoid, it has an internal resistance of 16 ohms, so we represented it as a simple resistor with a value of 16 in this simulation. We then incrementally increased the gate voltage and at around 2.2V current peaked through the resistor.

Since the ESP can output up to 3.3V at a GPIO pin then this circuit will be effective at locking and unlocking the lockbox.

Next, we built this circuit on a breadboard and used a couple lab bench power supplies to generate both the 12V and 3.3V and found that it did not unlock the solenoid at all. The solenoid only moved to the unlock position when we increased the gate voltage to 7V, more than double what a GPIO pin can supply. Since this option was a failure, we decided to use a similar BJT circuit as shown in Figure 6.

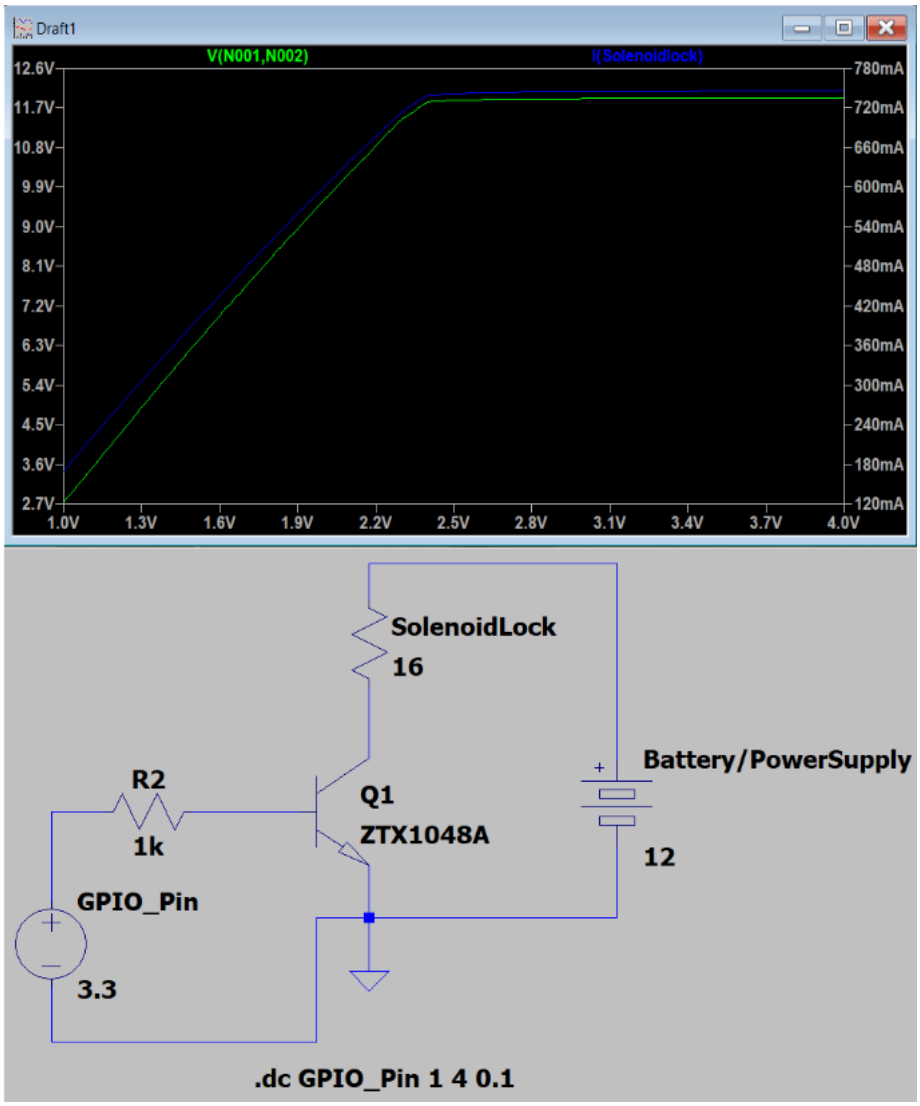


Figure 6: LTSpice Simulation for BJT

After getting an appropriate BJT to use, we created this circuit on a breadboard and it was successfully able to actuate the solenoid.

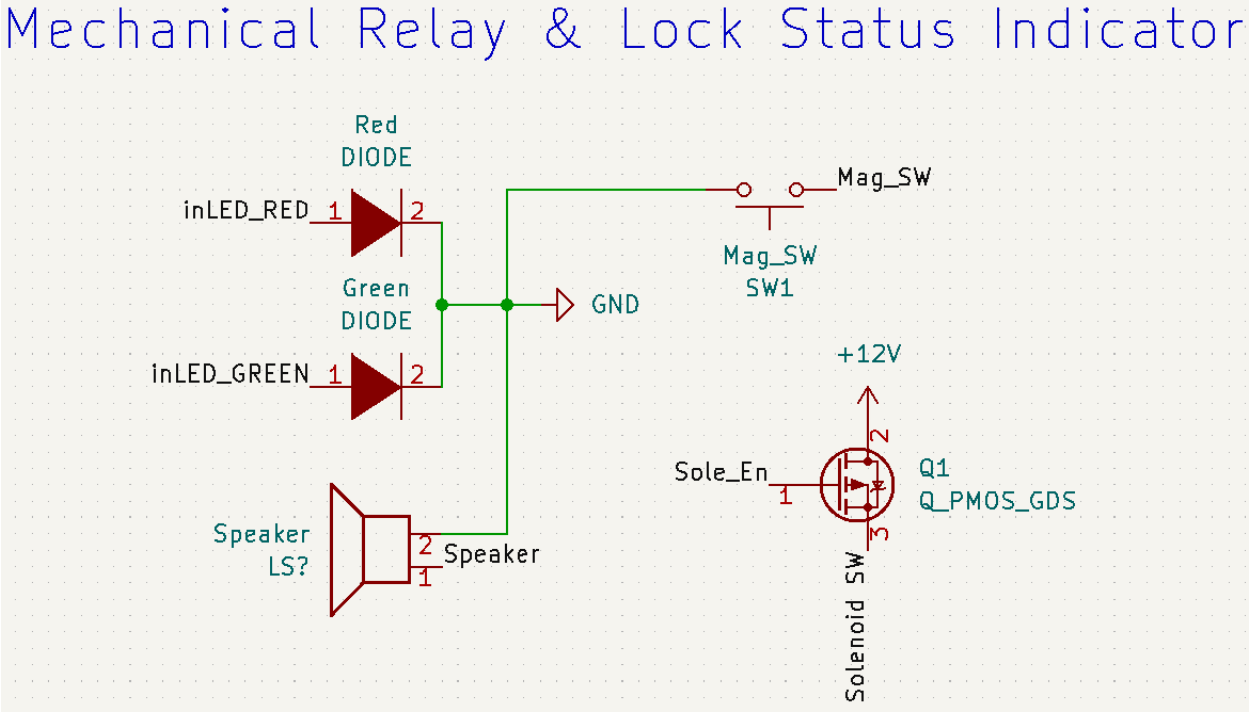


Figure 7: Mechanical Relay & Lock Status Indicator Schematic

2.2.4 Power Supply

The power subsystem plays a crucial role, powering the control unit, user interface, and the mechanical relay & lock status indicator subsystems. As stated previously in section 1.4, our lockbox will be primarily powered via an AC/DC wall outlet power supply, but also have a back-up battery so it remains functional if it loses power from the wall outlet. The Talent Cell 6000mAh 12V DC Li-ion battery pack can charge and power components at the same time. This functionality is compatible with our design because the lockbox must remain plugged in and while it is plugged in it should charge the backup battery and power all of the lockbox components. Included with the Talent Cell battery pack is a AC/DC 12.6 1A charger which satisfies our need to charge the battery pack, and/or power our other subsystems. Since the charger comes with the battery pack, there is no risk for us picking out the wrong one which could cause it to be damaged. Furthermore, the Talent Cell has a built in BMS with overcharge, discharge, and short-circuit protection.

Many of our project's components operate at different voltages. For example, the solenoid switch requires 12V, the LCD display requires 5V, and the ESP32 microcontroller, fingerprint sensor, and LEDs require 3.3V. To create these 3 different voltages, my initial idea was to use 2 linear voltage regulators; one that steps down 12V to 5V and another that steps down 5V to 3.3V. Figure 8 shows the schematic for this initial design.

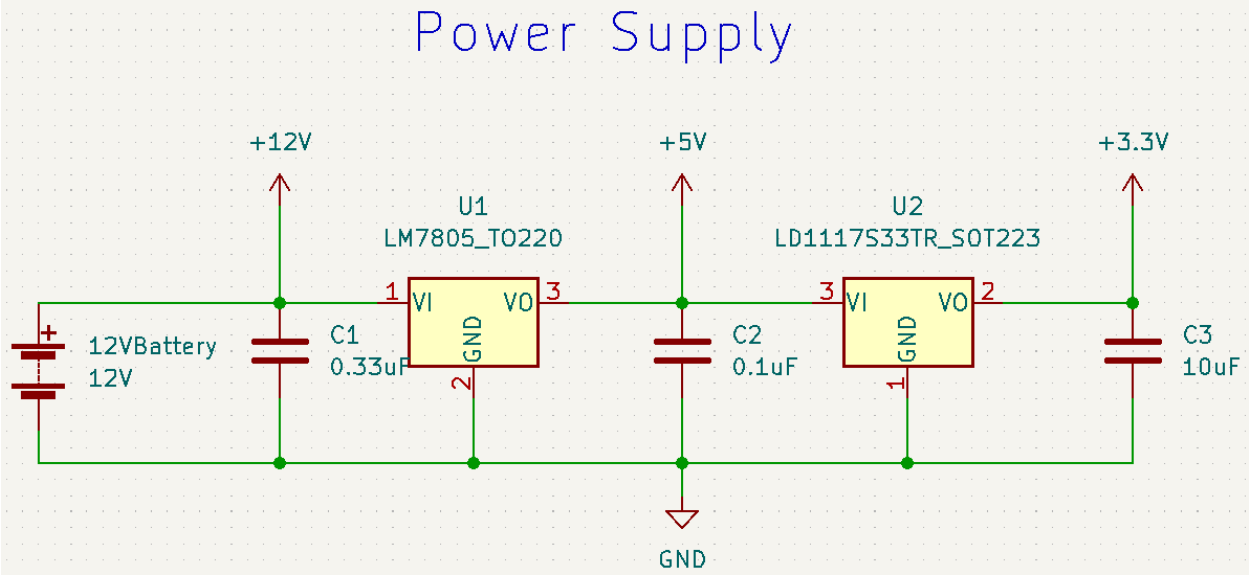


Figure 8: Power Supply Schematic Rev. 1

During our design review, we were informed that this is a very inefficient design since there will be significant power losses when stepping down from 12 to 5V using a linear voltage regulator. Using the equation:

[9]

$$P = IV$$

Where P is power in Watts, I is current in Amps, and V is voltage in volts, we can calculate that the LM7805 voltage regulator will dissipate (12V - 5V)*(0.8A) = 5.6W [0.8A is the maximum theoretical current drawn by all our components]. Furthermore, using the equation:

[10]

$$T_J = T_A + (R_{\theta JA} * P_D)$$

Where T_J is the temperature of the chip-junction, T_A is the ambient room temperature (°C), R_θJA is the junction to ambient thermal resistance given by the datasheet (°C / W), and P_D is the power dissipated (W). T_J = 25°C + (19°C / W) * 5.6W = 131°C which is extremely hot, just 20°C below the rated maximum for the LM7805.

I needed to redesign this circuit and with some guidance from Jason Paximadas and Zhicong Fan, they recommended we use either a buck converter or a switching power supply to step down from 12V to 5V since both options are much more thermally efficient. Figure 9 shows a sketch of those potential schematics.

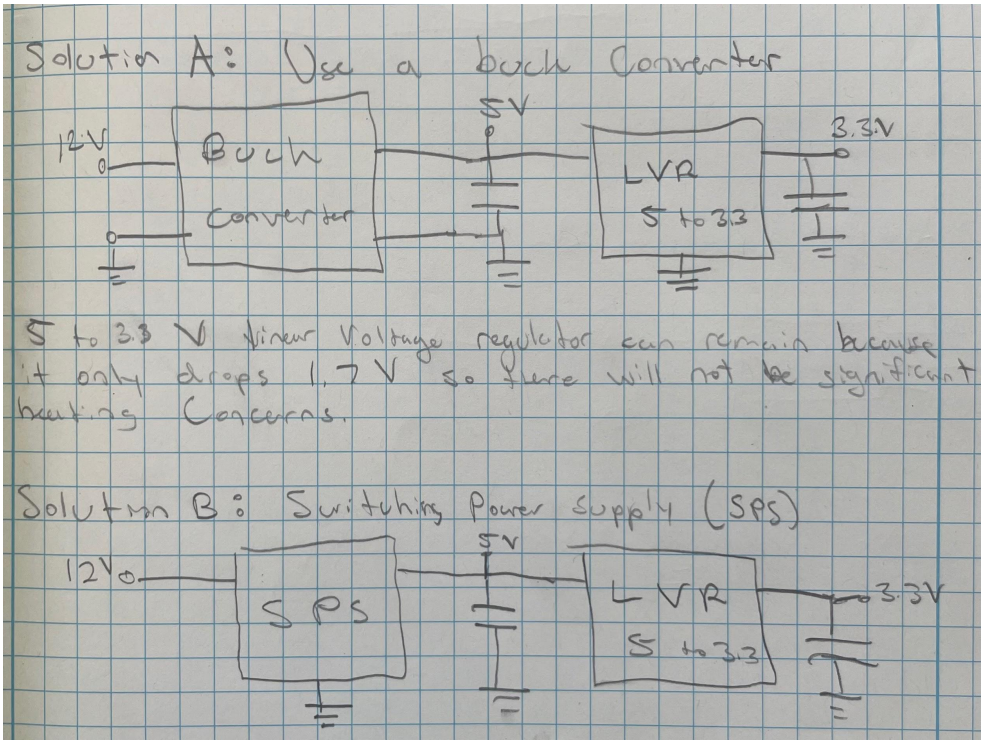


Figure 9: Buck Converter vs. Switching Power Supply

After careful consideration, we chose to use a switching power supply since it has a similar footprint to a linear voltage regulator. Figure 10 shows the final circuit we designed for the power supply.

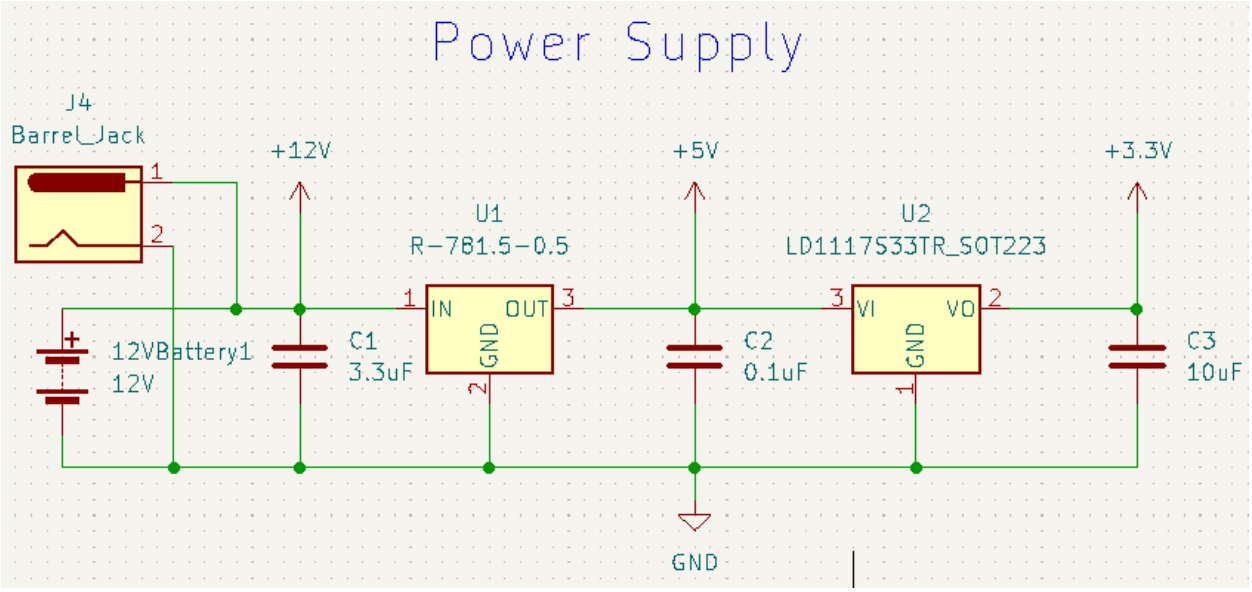


Figure 10: Power Supply Circuit Rev. 2

2.2.5 SMS Module

The SMS module was originally designed to be a web application interface. However, we realized that there was no real need for a web application as we displayed what needed to be shown to the user on the LCD display and the web application was rather redundant. The SMS module needs to have the ESP32 connect to Wi-Fi on bootup. Then, when fingerprint scans are successfully authenticated, the SMS module creates a message from a given Twilio phone number and sends it to the phone number previously configured with the scanned fingerprint. The message created contains a simple four digit pin code randomly generated by the randint python function. When fingerprint scans are unsuccessful, the SMS module goes through the same process of creating and sending a message but this time the message says that the authentication has failed and the door remains locked. The random four digit pin code that was generated is also saved to the ESP32 controller as well so that it can check for keypad authentication. Phone numbers need to be correctly configured on the ESP32 and registered on Twilio which is done through the user console on their website.

As can be seen in Appendix A, Section 5, the SMS module was fully functional and passed all of its verification tests. As mentioned, the web application was removed from our design so all of the verification tests are simply testing communication between the ESP32, the SMS API, and our phones. Additionally, our user software flowchart in Appendix B, Section 1 and admin software flowchart in Appendix B, Section 2, are all still accurate from our original design with the exception of the ESP32 microcontroller handling all the responsibilities the original TCP server was supposed to.

2.3 Tolerance Analysis

Supplying the system with adequate and uninterrupted power is imperative in ensuring the proper functionality of authentication methods, continuous communication with users, and preservation of biometric data. A standby uninterruptible power supply (UPS) will turn on to power the system when utility supply voltage has fallen below a predetermined level. When the UPS is not powering the system, a rectifier will convert AC to DC to charge the UPS.

The uninterruptible power supply will be composed of protected 18650 lithium ion batteries due to their high energy density and high voltage. To estimate the amount of power required to facilitate the lock system, a “worst-case” tolerance analysis was performed. Design components requiring power were placed at their tolerance limits to calculate the range of potential wattages demanded by the load.

Four key components were considered:

- AS608 optical fingerprint sensor module
- LCD2004 character-type liquid crystal display
- Micro DC motor / Push-pull solenoid

- ESP32-S2-MINI-1 microcontroller

Voltage requirements for the fingerprint sensor module range from 3.6 - 6.0V while current drawn ranges from 120 - 150mA.

Voltage requirements for the display range from 4.5 - 5.5V while current drawn ranges from 2 - 5mA.

Voltage requirements for the motor/solenoid vary from 6 - 12V while current ranges from 40 - 300mA.

The ESP32 microcontroller input voltage varies from 3.0 - 3.6V and current drawn varies from 68 - 310mA.

Using P = IV, the greatest and least possible load power consumptions by these components were determined. These components, at minimum, will consume 0.855W 20%. At most, they will dissipate 5.6435W 20%. The additional margins provide a more encompassing tolerance estimate regarding the variation of the necessary power supply, increasing the probability of continuous power in the presence of solenoid voltage spikes or lithium-ion self-discharging.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

As of November 2, 2022, the average hourly pay for an electrical engineering graduate in the United States is approximately $33.98 an hour [8]. It is estimated that each person on the team will work 15 hours per week for 10 weeks.

$$Labor\ cost\ per\ person\ = \frac{\$33.98}{hour} * 2.5 * 150\ hours\ = \ \$12,742.50$$

$$Total\ labor\ cost\ = \frac{\$12{,}742.50}{person} * 3\ people\ = \ \$38,227.50$$

3.1.2 Parts

Description	Manufacturer	Quantity	Cost/Unit ($)	Link
ESP32-S2-WROVER	Espressif Systems	1	3.95	Link
3X4 Matrix Keypad Module	SparkFun Electronics	1	4.95	Link
ZFM-20 Fingerprint Identification Module	Zhiantec	1	49.95	Link
LCD2004 Module	SunFounder	1	10.99	Link
6000mAh Li-ion Battery Pack	TalentCell	1	34.99	Link
LD-20MG Digital Servo	LewanSoul	1	15.98	Link

MP001162 Linear Solenoid	Multicomp Pro	1	5.56	Link
SW-MAG REED FLANGE MOUNT	NTE Electronics, Inc	1	4.11	Link
R-785.0-0.5 12-to-5V switching regulator	Recom Power	1	8.24	Link
LD1117AS33TR 5 to 3.3V Regulator	Stmicroelectronics	1	0.95	Link
Red 623nm LED Indication	Lite-On Inc.	1	0.36	Link
Green 568nm LED Indication	Kingbright	1	0.34	Link
BUZZER MAGNETIC 3V 12MM TH	CUI Devices	1	1.27	Link
1uF Capacitor	TDK	5	0.062	Link
10uF Capacitor	TDK	2	0.096	Link
22uF Capacitor	TDK	1	0.154	Link
0.1uF Capacitor	TDK	4	0.021	Link
20pF Capacitor	MURATA	4	0.002	Link
0.33uF Capacitor	TDK	2	0.397	Link
Total cost: $136.23				

3.1.3 Grand Total

$$\begin{aligned} Grand\ total &= total\ labor\ cost + total\ parts\ cost \\ &= \$48,964.50 + \$136.23 \\ &= \$49,100.73 \end{aligned}$$

3.2 Schedule

Week	Project Component	Noah	Shelby	Akshay
9/26	Design Document	Finish draft of PCB design	Compile power supply & lock parts required	Database structures, Research python modules
10/3	Design Review PCB Reviews	Revise PCB design	Connect user interface module, test requirements	Web application development, Testing
10/10	PCB Order I Teamwork Evaluation I	PCB Order I design complete	Test power supply requirements with different locks	Web application development, Testing
10/17	Devise lock box layout	Plan out how to assemble all parts onto locker	Plan out how to assemble all parts onto locker	Wi-Fi connectivity, Fingerprint

				scanner data transfer to server
10/24	Revist Requirements & Verification	Verify components	Verify components	Debugging microcontroller connection
10/31	PCB Order II Individual Progress Reports	Assemble all components	Assemble all components	Review and optimize code, Debugging
11/7	Debugging and Finalization	Assemble all components	Assemble all components	Verify code works with ESP32, Debugging
11/14	Mock Demonstration			
11/21	**Fall Break**			
11/28	Final Demonstration	Prepare for demo	Prepare for demo	Prepare for demo
12/5	Final Presentation Final Papers Teamwork Evaluation II	Complete paper and evaluation	Complete paper and evaluation	Complete paper and evaluation

4 Conclusions

4.1 Accomplishments

All three of our high level requirements described are met by our device. The inherence factor authentication requirement is met by the fingerprint sensor successfully storing fingerprint templates on the memory. Additionally, the fingerprint templates previously saved in the enrollment process are accessible to authenticate a person's scanned fingers and are linked by a fingerprint ID number to connect to that person's phone number. The fingerprint sensor also is able to send success and failure messages in the console and the LCD display. The possession factor authentication requirement is met by the phone number's successfully sending four digit pin codes to the verified phone numbers. Phone numbers are verified on Twilio API and the ESP32. The SMS module also sends success and failure notifications based on fingerprint authentication. The wireless connectivity requirement is met by the ESP32 successfully connecting to a Wi-Fi network and can be verified by sending success or failure messages in the console of an IDE. Additionally, the Wi-Fi connection is necessary in order to send SMS messages through the API.

4.2 Uncertainties

Some uncertainties that we had while designing our final system included issues connecting to Wi-Fi. We tested through multiple different Wi-Fi sources including hotspots and

public Wi-Fi but it seemed that the ESP32 would not connect to Apple iPhone hotspots or networks that require additional authentication to connect such as IllinoisNet. We did get our system to connect to Wi-Fi using an Android hotspot, but this could be looked at in the future. Other challenges we had included soldering on the micro USB port since it was so small. We would have liked to use a larger cable such as a regular USB port. Additionally, interfacing the i2c communication with the LCD was slightly troublesome due to the microPython package used causing the cursor to move very slowly. Additionally, one of the pins would be turned on incorrectly on bootup causing the LCD display to break, but this was fixed using a hardware switch component.

4.3 Ethical Considerations

We will make sure “to treat all persons fairly and with respect”[1] when conducting ourselves as a team both between ourselves and with all others. We will also be sure to “hold paramount the safety, health, and welfare of the public”[1] when designing our lockbox in accordance with the IEEE and ACM Code of Ethics.

Since this will be a lockbox that opens using biometric data, a fingerprint, it must be able to both secure the contents, but equally important it must keep the biometric data secure. In alignment with sections 1.6 and 1.7 of the ACM code of ethics and professional conduct which state, “Computing professionals should protect confidentiality except in cases where there is evidence of the violation of law, of organizational regulations, or of the Code. ”[2], we will make sure that data is transferred by secure methods and data is never made public in order to preserve the integrity of our lockbox. Fingerprint data will also be kept on the hardware instead of the web server to keep the biometric data more secure to cyber attacks.

Regarding our battery pack, we will need to be safe in the lab when working with this. Our battery pack will follow the guidelines from the General Battery Safety[3] document on the ECE445 course website. We will also look through the battery’s manuals for the proper safety and handling procedures. We have chosen to implement a prebuilt battery pack with a built in BMS to avoid costly mistakes that may happen when designing our own. Batteries have hazards such as battery acid burn, flammability, and electric shock. In order to prevent injuries to users, we need to make sure the battery is safely secured onto the lockbox. Additionally, we will practice proper wall outlet safety by purchasing an AC/DC Converter and avoiding overheating components.

4.4 Future Work

Our final design and product, while functional, has much room for improvement. Some future work that could be done includes verifying phone numbers during the enrollment process. As it functions currently, any ten digit number combination could be inputted when the system

asked the user for a phone number. However, that phone number must also be registered on the Twilio API in order to send SMS messages. In this case, the phone number and fingerprint template stored together would be invalid and there is no way of reentering phone numbers. Another improvement would be to have some sort of temperature control on the system so that in the case of sensitive chemicals or frail historical document preservation, the temperature of the system does not interact with the items stored and is overall safer. One more thing that could be done is being able to check the system status which is currently unavailable. This could be done by checking if the linear solenoid is in its on or off state and send that signal to the ESP32 signaling locked or unlocked. Then the system status could be sent in an SMS message to all the users enrolled. One more future improvement we could implement if this system were to be implemented in the real world is having a master key or master password in order to reset the machine. Currently, configurations stored cannot be deleted unless plugged into the ESP32 and we could easily implement this using delete template and delete file functions.

5 References

[1] "IEEE Code of Ethics." IEEE, IEEE Policies, Section 7 - Professional Activities (Part A - IEEE Policies), . [Accessed 15 September 2022]

[2] "ACM Code of Ethics and Professional Conduct." Association for Computing Machinery, ACM, . [Accessed 15 September 2022].

[3] "General Battery Safety." Safe Practice for Lead Acid and Lithium Batteries, [Online]13 April 2016, [Accessed 15 September 2022].

[4] IBM Security. (2020). *Cost of a Data Breach Report 2020.*

[5] IBM Security. (2021). *Cost of a Data Breach Report 2021.*

[6] Texas Instruments, "uA7800 Series Positive-Voltage Regulators", LM7805 datasheet, May 1976 - Rev. May 2003, [Accessed 15 September 2022].

[7] Texas Instruments, "LM3940 1-A Low-Dropout Regulator for 5-V to 3.3-V Conversion ", LM3940 datasheet, May 1999 - Rev. Feb. 2015, [Accessed 15 September 2022].

[8] ZipRecruiter, "Graduate electrical engineer salary." [Online]. Available: https://www.ziprecruiter. com/Salaries/Graduate-Electrical-Engineer-Salary

[9] "Douglas College Physics 1207." Section 4.4 Electric Power and Energy - includes Heat energy. [Accessed 2 November 2022].

[10] A. Vassighi and M. Sachdev, *Thermal and power management of Integrated Circuits*. New York, NY: Springer Science+Business Media, 2006. [Accessed 2 November 2022]

Appendix A: Requirements and Verification Tables

1. Control Module

Requirements	**Verification**	**Verification Status**
1. Upon successful multi-factor authentication of all configured parties, the microcontroller must disengage the lock within 2 seconds.	1a. Engage lock 1b. Begin authentication process 1c. Once the last successful factor of authentication has been entered, begin a timer 1d. When the lock obtains a signal to disengage, stop the timer 1e. Ensure the time is less than 2 seconds	Yes
2. Upon successful biometric authentication, the microcontroller must determine to which phone number an SMS must be sent.	2a. Begin biometric authentication process 2b. When a fingerprint scan is successfully matched to a template stored in the fingerprint database, return associated phone number	Yes

2. User Interface

Requirements	**Verification**	**Verification Status**
1. LCD displays each button press when typing in the 4 digit token, phone number, all other input requests.	1a. Visually verify that the correct numbers show up when pressed.	Yes

3. Power Module

Requirements	Verification	Verification Status
1. Battery remains within the rated temperature while operating under maximum possible load (~700mA)	1a. Connect an amp meter between the power source and the power input on the PCB 1b. Perform a routine opening and closing sequence 3 times in a row 1c. Use a laser thermometer gun to monitor the battery temperature, making sure it does not go above 125°F.	Yes
2. The device must be able to remain powered even if unplugged from the wall outlet	2a. Connect AC-DC adapter, battery, and PCB to barrel jack splitter. 2b. Visually verify that the LCD backlight turns on. 2c. Unplug adapter from wall and verify that the LCD screen did not reset and remained constantly powered.	Yes

4. Lock Module

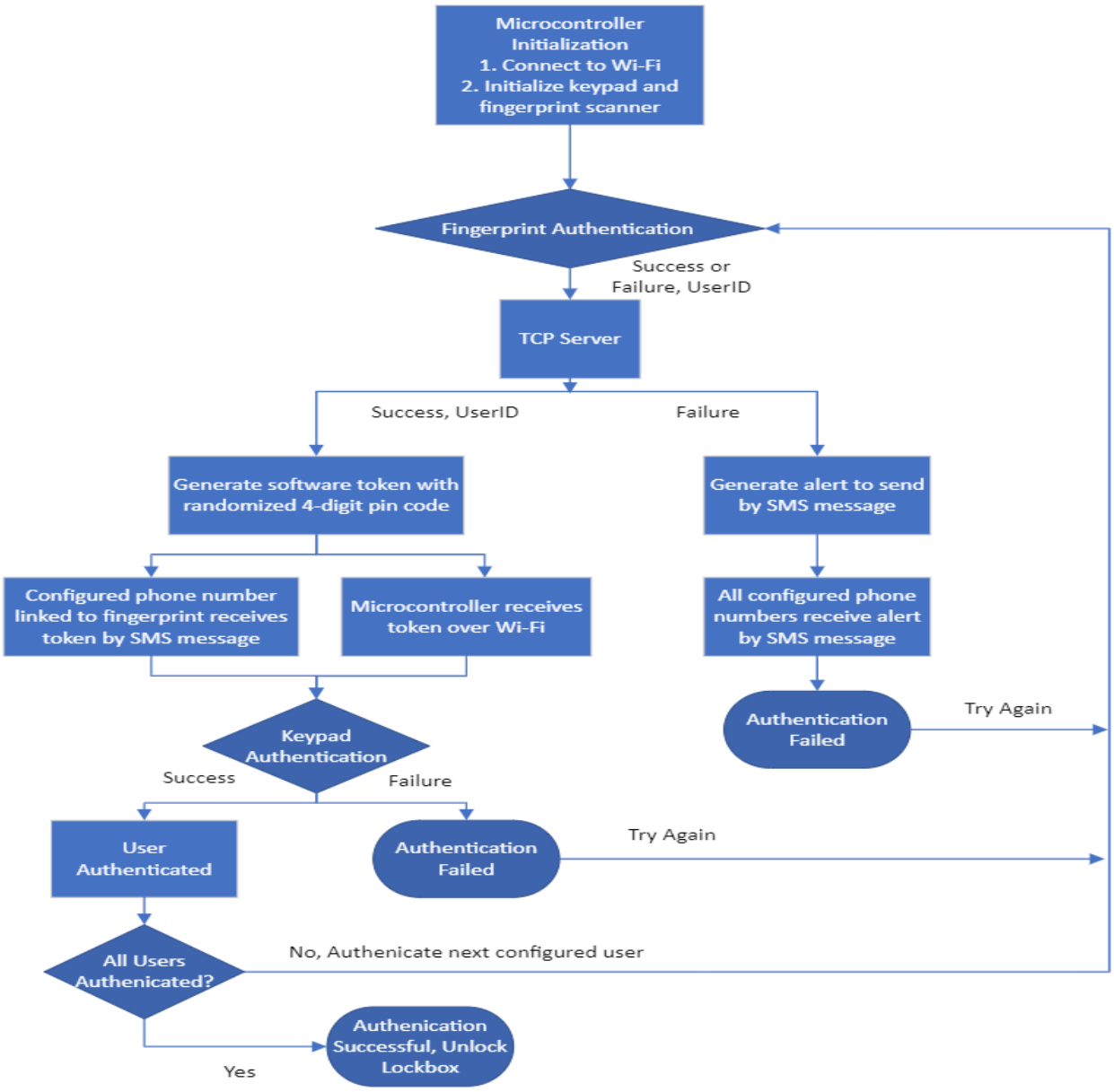
Requirements	Verification	Verification Status
1. Must drive the solenoid with a 3.3V input signal for a solenoid VDC in the range of 9V - 12V.	1a. Power solenoid with 12V and turn on 3.3V input signal to ensure the lock disengages. 1b. Repeat this process, lowering the solenoid VDC with each test.	Yes
1. Solenoid lock securely closes the door and will not open unless signaled to do so.	2a. Physically pull on the door and attempt to open it when the door is supposed to be locked. 2b. Visually verify the door does not open. 2c. Repeat steps when the door is signaled to be unlocked and make sure it opens.	Yes

5. SMS Module

Requirements	Verification	Verification Status
1. Able to securely connect Wi-Fi.	1a. Check on the router (hotspot) that ESP32 is securely connected to Wi-Fi on boot.	Yes
1. Able to securely store at least 3 distinct user configurations.	2a. Display the number of enrollees on LCD display when authenticating. 2b. During enrollment, be able to set up a fingerprint template linked with a phone number that can later be accessed for each person.	Yes
2. Able to generate a software token containing a randomized 4-digit pin code.	3a. Verify that the 4-digit pin code sent to the phone is able to successfully authenticate when entering code in the keypad.	Yes
3. Able to send accurate SMS messages to configured phone numbers in under 10 seconds.	4a. Successfully POST an HTTP request to the SMS API in order to send a failure message when fingerprint authentication fails under 10 seconds. 4b. Successfully POST an HTTP request to the SMS API in order to send a token upon successful fingerprint authentication in under 10 seconds.	Yes

Appendix B: Software Flowcharts

1. User Software Flowchart


Microcontroller Initialization
1. Connect to Wi-Fi
2. Initialize keypad and fingerprint scanner
Fingerprint Authentication
Success or Failure, UserID
TCP Server
Success, UserID
Failure
Generate software token with randomized 4-digit pin code
Generate alert to send by SMS message
Configured phone number linked to fingerprint receives token by SMS message
Microcontroller receives token over Wi-Fi
All configured phone numbers receive alert by SMS message
Authentication Failed
Try Again
Keypad Authentication
Success
Failure
User Authenticated
Authentication Failed
Try Again
All Users Authenicated?
No, Authenicate next configured user
Yes
Authenication Successful, Unlock Lockbox

2. Admin Software Flowchart