Secure Mailbox with Mobile Connectivity

ECE 445 Design Document - Spring 2023

Team 26

Neehar Sawant, Avadh Patel, Roshun Navin

TA: Vishal Dayalan

1. Introduction

1.1 Problem:

Mail is one of the biggest forms of communication in today's world and sees the transfer of sensitive information such as bank statements and checks to millions of households throughout the nation. While apartment buildings have keys which are only possessed by a mailman and tenant that unlocks a unit's mailbox, many single family and townhomes have a conventional mailbox which can be easily opened by anyone. This means that millions of Americans do not have any means of securely storing their mail as they often do not pick their mail up immediately, leaving mail unprotected and easily stolen for long periods of time. As seen in the WFMY local news article, checks in the mail are frequently stolen and authorities warn citizens to not leave sensitive material in mailboxes overnight and instead hand them directly to the mail carrier. Currently, electronically locking boxes exist which let packages be stored outside of your door securely, but a similar solution does not exist for mail that is both secure and can be mounted in place of an existing mailbox.

1.2 Solution:

Our solution is to create a mailbox that will have two primary features. The first feature will be the ability to automatically lock once mail is placed inside of it. The second feature allows the user to schedule a time period for the mailbox to remain unlocked in the case of multiple deliveries happening throughout the course of a day. Any time the mailbox is opened, closed, or mail is placed inside the mailbox the user will be notified. These notifications will be sent to the user via a mobile application. Additionally this mobile application will allow the user to interact with the mailbox. In the application the user will be able to manually unlock/lock the mailbox and set a schedule for the mailbox to remain unlocked. If there is no schedule set, the mailbox will act under the assumption that only one mail delivery will occur that day and will lock after mail is placed into the mailbox for the first time and the door is closed.

When it comes to the implementation, the mailbox will use a magnetic contact sensor to determine when the mailbox is opened and then send a signal to the PCB that sets a ready state and waits for the close signal. Once the door is closed, the magnetic contact sensor will send another signal to the PCB which in turn instructs the lock to close. The PCB will also interface with the mobile application, via the ESP32 which has built in wireless capabilities, and will send unlock and lock signals to the PCB to control the actions of the locking mechanism. This system will also be used for unlocking the mailbox during a scheduled window which can be controlled in the app. In order to notify the user if mail is present in the mailbox, multiple ultrasonic sensors will be used to detect if mail is covering any one of them. This information will be sent back to the PCB and then be sent to the mobile application to alert the user of mail. A small, low powered, motor will be used to lock and unlock the mailbox. It will be attached to a metal extension to secure the lock.

1.3 Visual Aid:

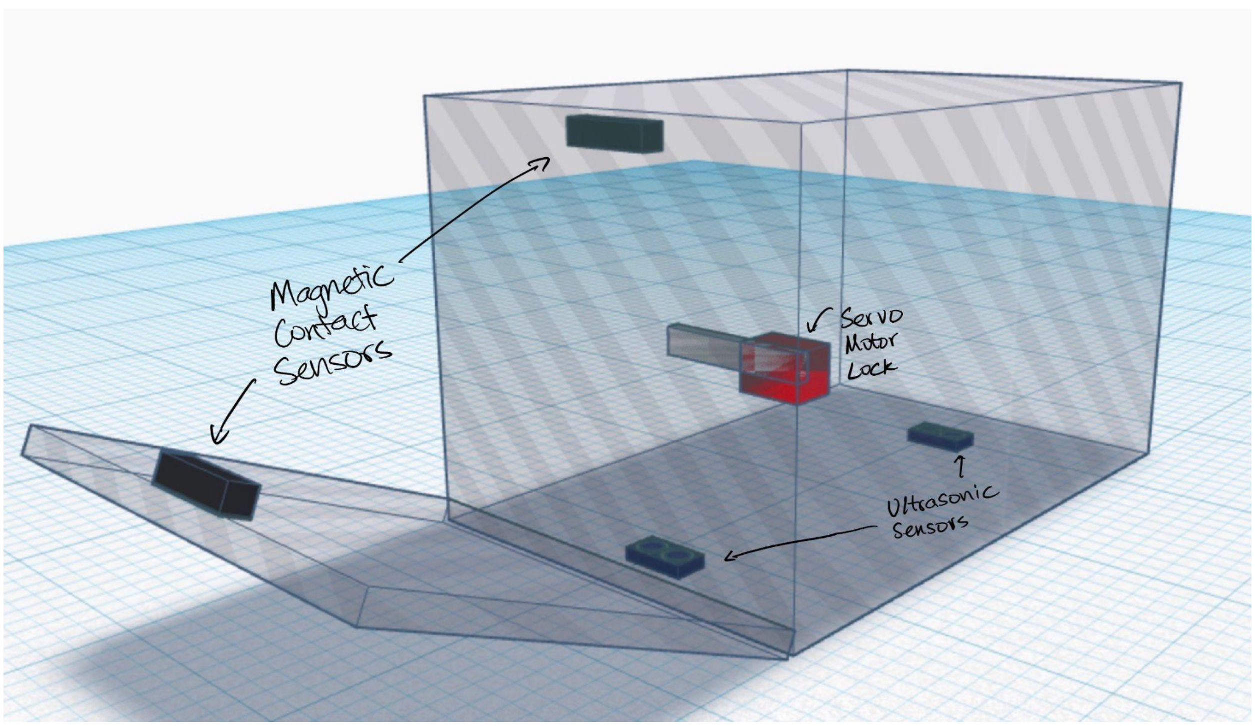

Figure 1: Full Mailbox with Labeled Sensors

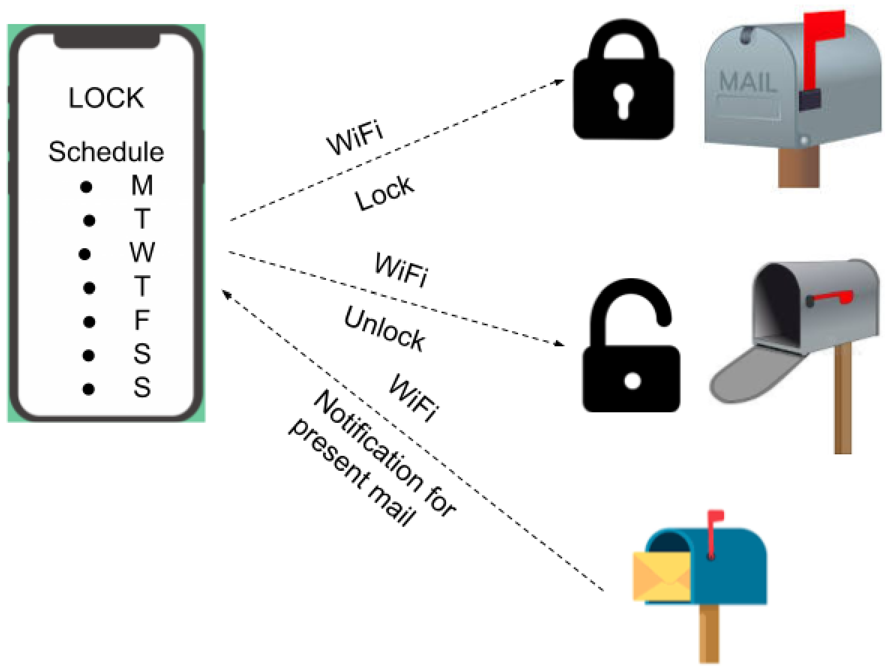

Figure 2: High Level Diagram

1.4 High-level Requirements List:

For our project to be successful, we will implement these requirements:

1. The mailbox must automatically lock within 10 seconds of the door being closed if mail is placed inside and there is no schedule set to leave it unlocked.
2. The mailbox must lock and unlock within 5 seconds of pressing the corresponding button on the application.
3. The mailbox must send a notification within 30 seconds of an action being made on the mailbox. This includes opening and closing the mailbox as well as whether mail is present.

2. Design

2.1 Block Diagram:

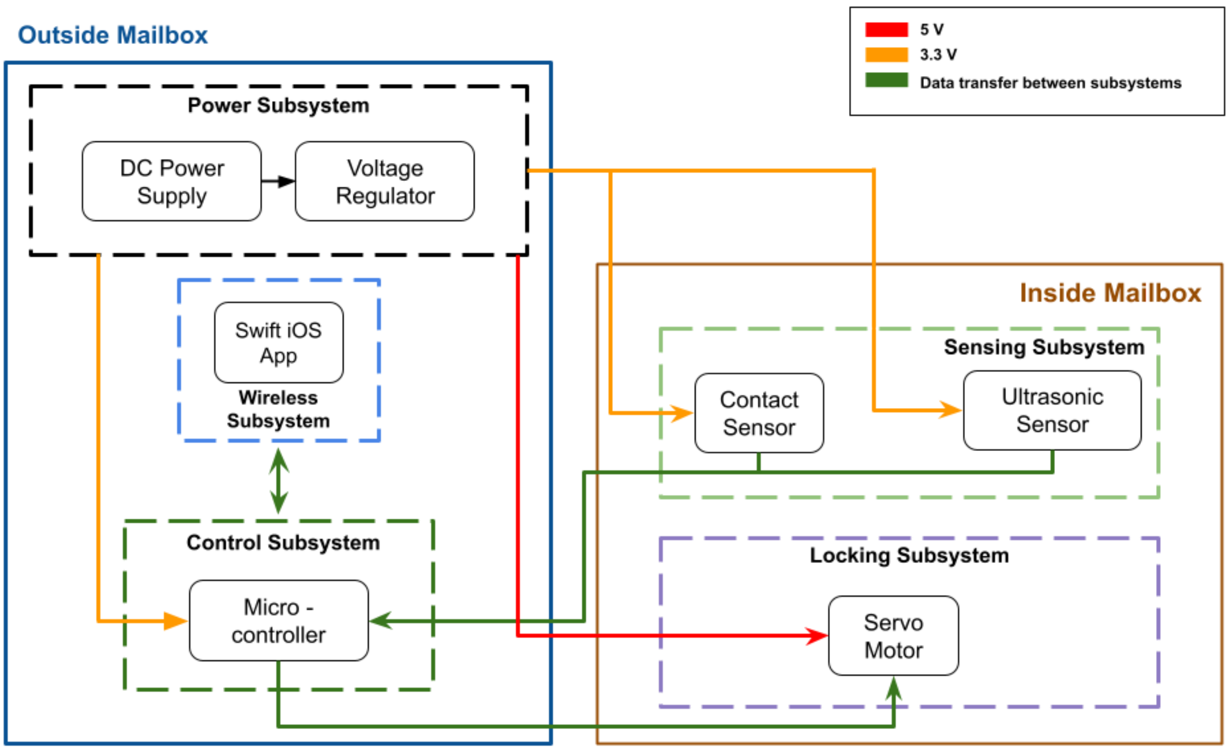

Figure 3

2.2 Physical Design:

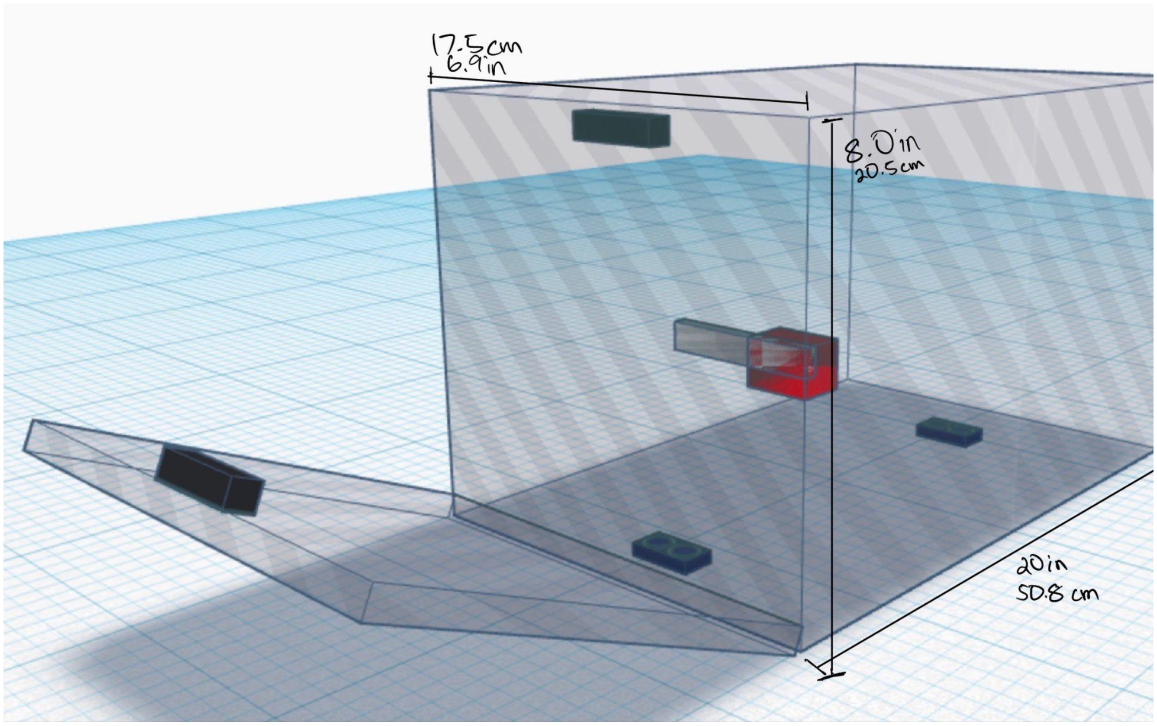

Figure 4:

Mailbox Measurements (Meets Federal Requirements)

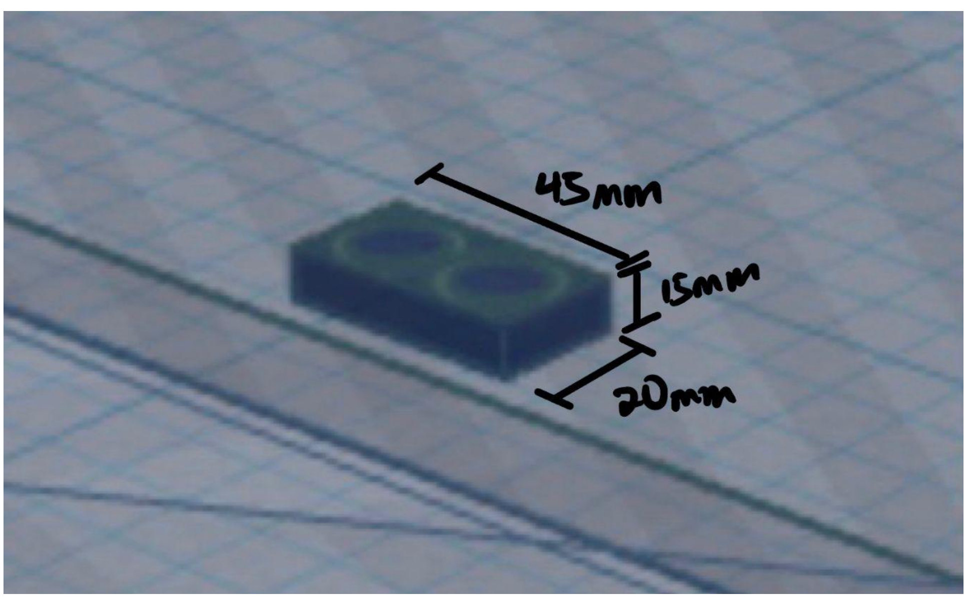

Figure 5:

Ultrasonic Sensor Measurements - Used to detect if mail is present

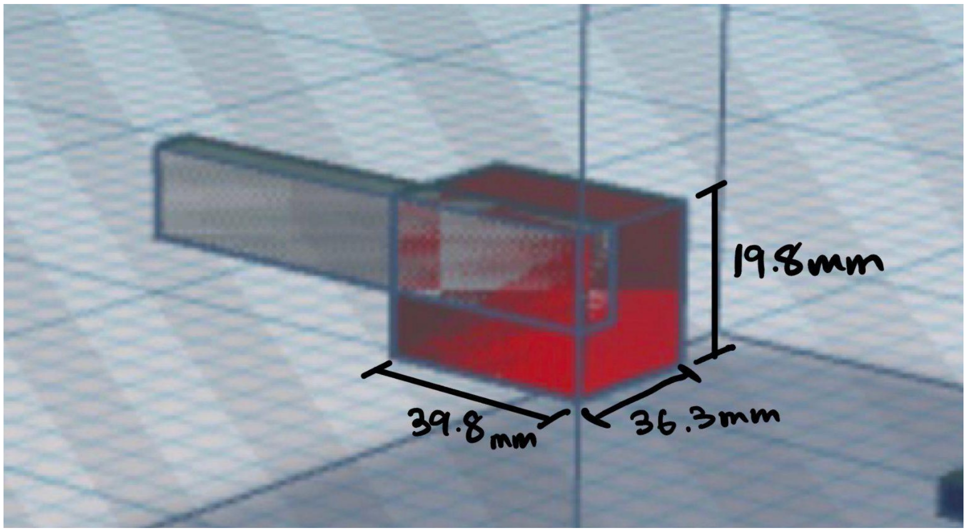

Figure 6:

Servo Motor Measurements - Used for locking Mailbox

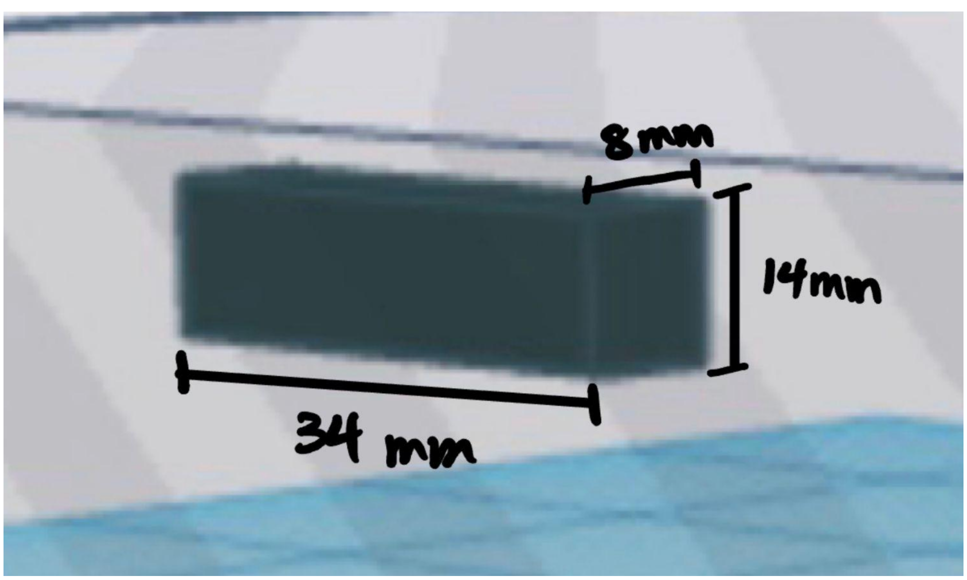

Figure 7:

Magnetic Contact Sensor - Used for detecting if the mailbox door is open or closed

2.3 Subsystems:

2.3.1 Power

This subsystem will provide the needed power to each component of the circuit. Using a standard 12 V connection. There will need to be voltage and current regulation to prevent damage to components on the PCB. An initial 12 V will need to be reduced to 3.3 V for the digital components but the original 12 V will be needed to drive the motors. To produce the proper power for the control unit, a combination of a low pass filter, and a voltage regulator should be used. Voltage regulators create a distinct voltage across them making it easy to have an accurate measurement of the output voltage. However they can continue some noise so having a low pass filter prior to the voltage regulators makes the circuit have excellent noise protection. Some protection diodes should be added on the motor power side to limit the feedback from the motors as this can cause damage to the other components in the circuit. A current limiting fuse will need to be attached as well.

This subsystem will supply the necessary power to each component of the mailbox. This will be done using a 12 V connection. To ensure that the components on the PCB are not damaged, voltage and current regulation are necessary. The initial voltage of 12 V will be decreased to 3.3 V for the ultrasonic sensors and the contact sensor. It will be decreased to 5 V for the WiFi module and the servo motor. A low pass filter and a voltage regulator combination will be used to generate the appropriate power for the control unit.

Requirements	Verification
• Uses a 12 V connection and regulate voltage and current draw from an outlet • The proper voltages and currents	• The voltage to the Motor and Wireless subsystem is 5 V $\mp$ 0.25 V measured by multimeter • The voltage for the digital components

needed for each other sub system will be given.	is 3.3 V ∓ 0.165 V measured by multimeter

2.3.2 Control

The control subsystem will receive data from the wireless subsystem and sensing subsystem. It will then communicate back with the wireless subsystem and the locking subsystem in order to ensure that the mailbox is locked or unlocked appropriately. A microcontroller (ESP-WROOM-32 Development Board) will be used in the control subsystem in order to properly communicate with all of the other subsystems. The microcontroller will connect has built in Wi-Fi connectivity which will allow it to connect to a server in order to communicate with the mobile application. Anytime the microcontroller receives signals from the sensing subsystem it will communicate with the Mobile Application in order for the user to be notified as well as the locking subsystem if needed. Additionally, data from the Mobile Application will be sent to the Microcontroller if the mailbox needs to be locked/unlocked manually or if a scheduled delivery is in place. In either of these cases the Microcontroller will communicate with the locking subsystem with the appropriate signal.

Requirements	Verification
● Can establish a stable WiFi connection ● Send data to and from mobile application ● Take data from contact and ultrasonic sensors and send signals to the locking system	● Check router website and ensure microprocessor is connected ● Setup remote host and test if pings can be sent and received over WiFi ● Manually code logic to test functionality of locking mechanism given current input of sensors

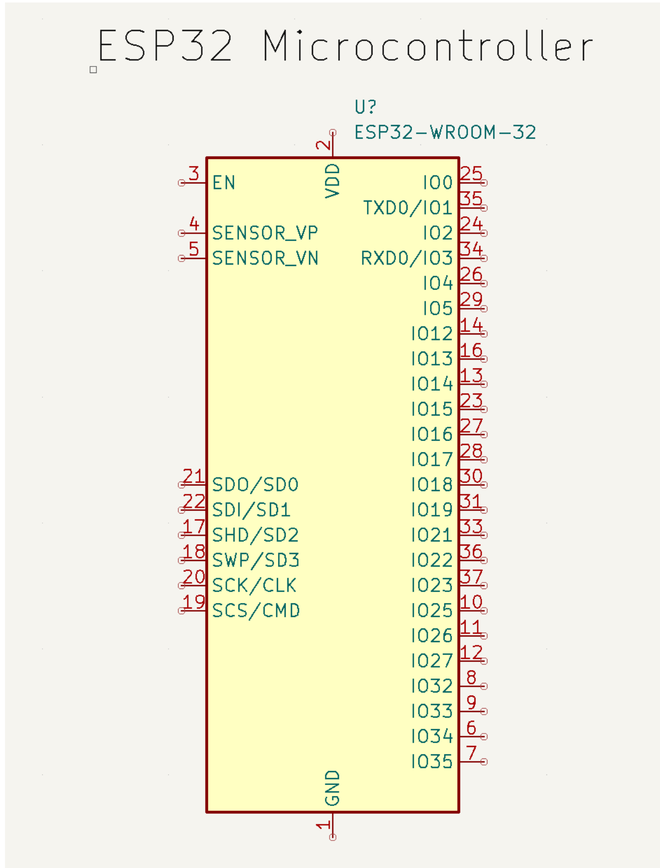

Figure 8:

ESP32 Microcontroller Circuit Schematic

2.3.3 Wireless

In order for the user to interact with the mailbox, we will be leveraging swift to make an IOS application which will be able to control how the mailbox behaves. In order to maximize the range and usability of our mobile app, we will be using WiFi as the means of communication between the application and mailbox. In the mailbox's default state, the door will automatically lock after being opened and closed the first time. This will trigger a notification to be sent to the user from the application alerting them that the mailbox has been opened and closed and that their mailbox has been locked. Another notification will be sent as well which will inform the user if mail is present inside the mailbox. The application will allow the user to manually lock and unlock the mailbox as well. Once the mailbox has been manually unlocked for the user to retrieve their mail and then they close the door again, the mailbox will revert back to its default state where the door is unlocked and is waiting for mail. The user also has the option to schedule when the mailbox will be unlocked, giving them greater flexibility and security for when they are expecting multiple packages. Thus if a schedule has been set, the mailbox will use the interval from the user as the default behavior on those days, defaulting back to the original functionality on days that no schedule has been set.

In order to communicate between the mailbox and the application, WiFi is going to be used to send and receive data. A cloud based server will be deployed which will be connected to the ESP32 microcontroller. The home WiFi network of the mailbox will be preset by the user allowing for connectivity to outside applications. The microcontroller and swift application will be given access to the cloud server. The swift application will use a service such as AWS Amplify and the Swift SDK to create backend data storage and transfer to send user inputs through the cloud and deliver low latency reaction and notifications too and from the mailbox. We decided that swift would be the most optimal development language for our application as it supports integration with cloud services and its development is widely supported and documented.

Requirements	Verification
- Mailbox locks or unlocks within 5 seconds of hitting the corresponding button on the application - Application sends notification to user within 30 seconds of action on mailbox(Open, Closed, Mail Present) - Mailbox unlocks and locks as given by a user inputted schedule	- View data in cloud server and ensure they are being received - Ensure data is being sent and received by cloud server - Set a schedule which calls for an unlock period of 2 min and monitor the lock's behavior

Mobile Application Sketch:

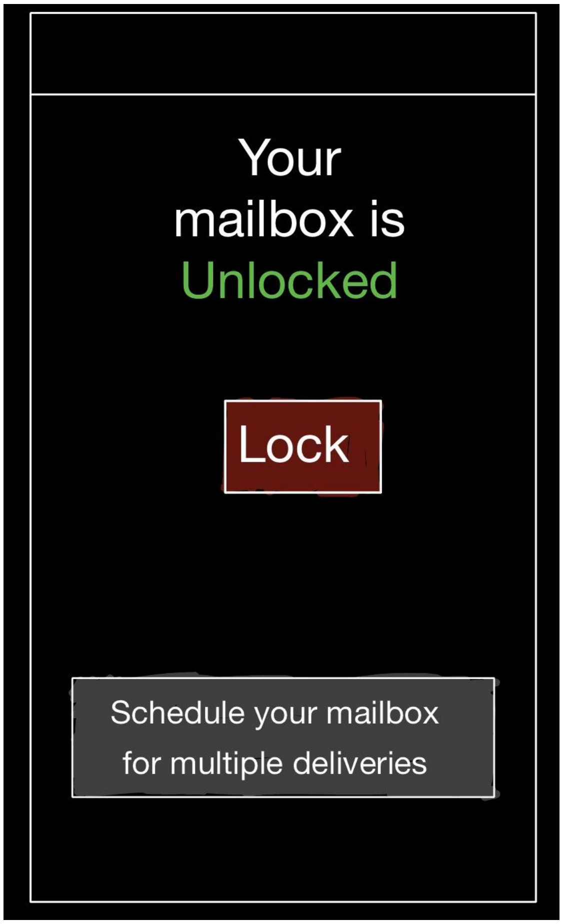

Figure 9: Unlocked State

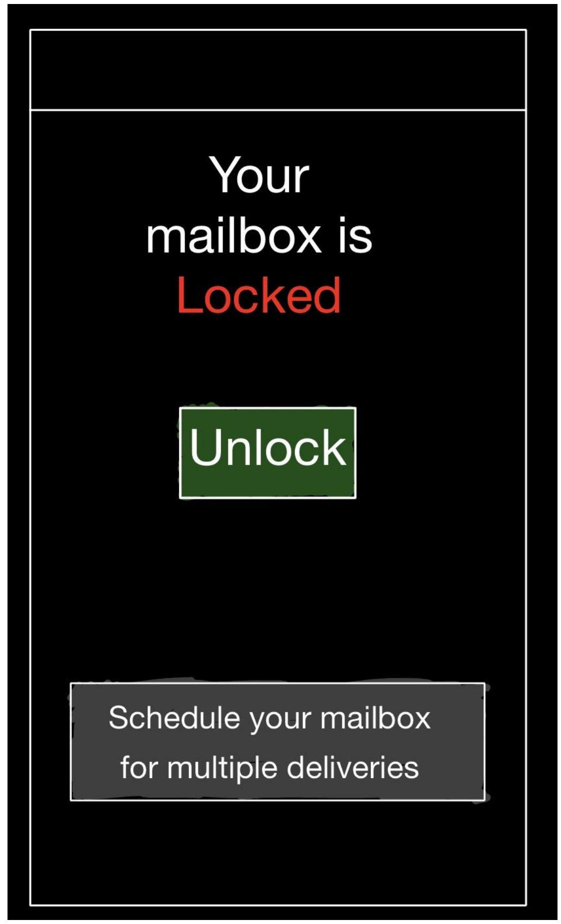

Figure 10: Locked State

This is one of two screens that will be displayed when the user launches the application. The app will show the current state of the mailbox along with a button that will change the state of the mailbox. Also shown will be a button that will take the user to a new page in the application to input their desired mailbox schedule. Figure 4 above shows the application in the unlocked state and Figure 5 above shows the application in the locked state.

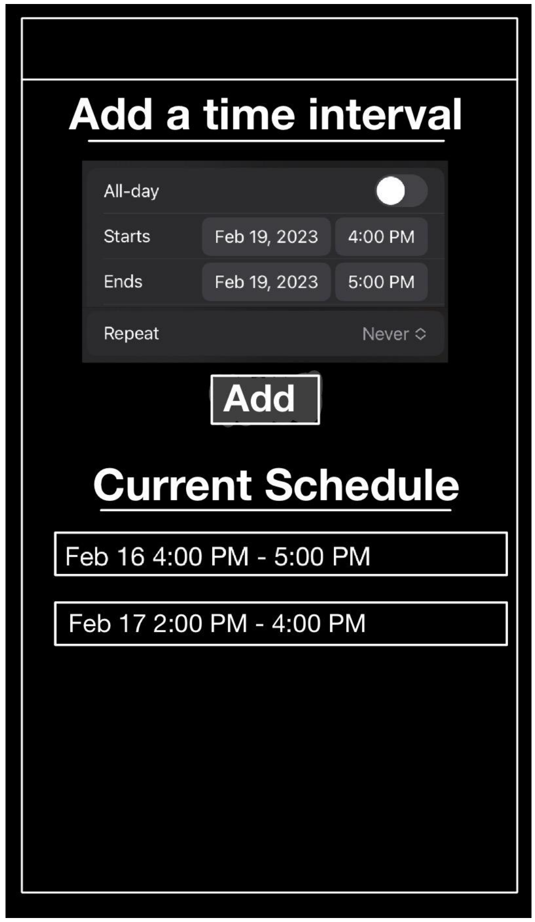

Figure 11: Scheduling

This is the second screen that will be displayed when the user presses the schedule button in the application. This screen allows you to enter the date and time range the user wants the mailbox in the unlocked state and add it to the schedule. The user is also able to see the current schedule they have set as shown in Figure 6.

2.3.4 Sensing

The sensors from our system will have two main components: a magnetic contact sensor (COM-13247) and two ultrasonic sensors (SEN-15569). The magnetic contact sensor will be placed on the door and on the roof of the mailbox to determine whether the door is closed. The ultrasonic sensors will be placed at the bottom of the mailbox to determine if there is mail in the mailbox. Both of these components are powered by 3.3V from the Power Subsystem.

The magnetic contact sensor and the ultrasonic sensor will both send data to the control system to control both the locking mechanism and wireless subsystem for notifications. Any time that the two contact parts of the magnetic contact sensor go from being together to separated or separated to together it will send a signal to the microcontroller notifying that the mailbox has been either opened or closed. Based on the type of signal that is being sent to the control system, it will be able to figure out whether the door has been closed or opened. This will allow the control subsystem to communicate with the wireless subsystem anytime the door has been opened or closed so that the user can be notified. The ultrasonic sensors will be used to detect whether or not there is mail. The ultrasonic sensors will send a signal to the control system anytime the door has been closed. If the data from that signal indicates that there is something placed over either of the ultrasonic sensors, the control system will be able to determine that there is mail inside the mailbox. Once again, this will allow the control subsystem to communicate with the wireless subsystem so that the user can be notified if there is mail in the mailbox.

Requirements	Verification
• Magnetic contact sensor sends the correct signal when it is either separated or brought back together and ensures that mailbox door is fully closed • Ultrasonic Sensor sends a signal to the	• Unit test the magnetic contact sensor with the microcontroller to verify that it sends the correct signals • We will check the output signal time of the ultrasonic sensor and make sure that it is between 117.65 µs and

control subsystem if an object is picked up between 2 and 20 cm from the sensor	1,176.47 µs in order to ensure that there is an object between 2 and 20 cm from the ultrasonic sensor.

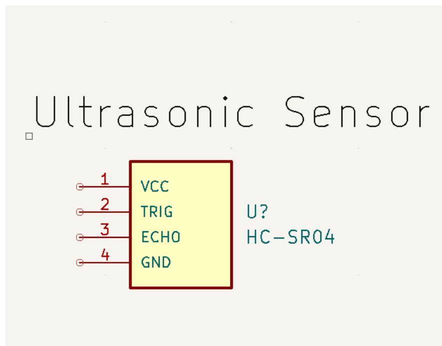

Figure 12:

Ultrasonic Sensor Circuit Schematic

2.3.5 Locking

A servo with a connector (HS-311 Standard Economy Servo) will be mounted inside the mailbox which works in conjunction with a piece mounted to the mailbox door. The locking subsystem will receive its information from the microcontroller after taking into account the status of the magnetic contact and ultrasonic sensors from the sensor subsystem as well as the manual unlock and lock of the mobile app from the wireless subsystem.

Requirements	Verification
• Servo Motor is able to move into the correct positions to lock and unlock the mailbox • The control signals from the microcontroller to locking mechanism need be delivered within 1 second	• Test the servo motor to ensure that it always moves to the correct position based on the signal • Unit test servo motor with control subsystem and time the amount of time it takes to move the servo to the correct position

2.4 Tolerance Analysis

When using ultrasonic sensors, we must ensure that mail is being detected in an accurate manner. If this is not being done accurately, mail will not be secured properly inside of the mailbox. An ultrasonic sensor can detect the nearest object to it within 2 to 400 cm. Our mailbox will be a height of 20.5 cm, so if our ultrasonic sensor detects an object between 2 and 20 cm away from it, then we will know that there is mail in the mailbox. An ultrasonic sensor outputs the amount of time it takes a sound wave to bounce back to it if there is an object within 400 cm of it. The equation for the amount of time a sound wave will take to bounce back to an ultrasonic sensor is the following: T = (2 * D) / v, where T is time, D is the distance between the ultrasonic sensor and the object, and v is the speed of sound. The reason it is 2 * D is because we have to account for the sound wave to make two trips: one to the object and one back to the ultrasonic once it bounces off the object. The speed of sound is around 340 m/s which is 0.034 cm/µs. We will be testing to see if there is an object within 2 and 20 cm. For 2 cm the time it will take is T = (2 * 2) / 0.034 = 117.65 µs. For 20 cm the time it will take is T = (2 * 20) / 0.034 = 1,176.47 µs. This means we will have to continuously monitor our ultrasonic sensor and if we ever see a time output between 117.65 µs and 1,176.47 µs, we will know that mail is in the mailbox.

3. Cost and Schedule

3.1 Cost Analysis

Labor:

The average starting salary for ECE graduates is about $92,000/year or $46/hour. Using the estimate that we will each work 100 hours on this project each during this semester, we can estimate the total cost for our labor.

$$\frac{\$46}{hour} \times\ 100\,hours\ =\ \frac{\$4{,}600}{person} \times 2.5\ \times 3\,people\ \ = \$34{,}500\ total\ labor\ costs$$

Parts:

Description	Manufacturer	Part Number	Quantity	Price	Total
Ultrasonic Distance Sensor	Digi-Key	HC-SR04	3	$4.38	$13.14
Magnetic Contact Sensor	Digi-Key	COM-13247	2	$3.95	$7.90
Servo Motor	HiTec	HS-311 (31311S)	1	$13.54	$13.54
Power Supply	ALITOVE	5050 3528	1	$11.99	$11.99
Microcontroller	HiLetgo	ESP-WROOM-32	1	$10.99	$10.99
3D Printed Mailbox	Illinois Maker Lab	NA	1	$20	$20
					PARTS TOTAL: $77.56

Grand Total:

$$\$34,500\ total\ labor\ costs\ +\ \$77.56\ total\ parts\ costs\ =\ \$34,577.56\ total\ costs$$

3.2 Schedule

Dates	Tasks
2/19 - 2/25 Design Document 2/23 Team Contract 2/24	Neehar: ● Start on PCB Design Avadh: ● Order rest of the parts necessary Roshun:

	● Begin developing models of mailbox for 3D printing
2/26 - 3/4 Design Review	Neehar: ● Work on PCB Design, unit testing parts as they arrive Avadh: ● Start making Swift app Roshun: ● Start 3D printing mailbox housing, start making Swift App
3/5 - 3/11 First Round PCBway 3/7 Teamwork Evaluation 3/8	Neehar: ● Finalize PCB Design, construct and test prototype Avadh: ● Begin WiFi connection, begin creating AWS instance Roshun: ● Front end development of Swift App, work on 3D printing
3/12 - 3/18	SPRING BREAK
3/19 - 3/25	Neehar: ● Finalizing first round of PCB, integrating sensors/parts Avadh: ● Finalize WiFi connection and connect to AWS instance Roshun: ● Connect mobile application to AWS instance
3/26 - 4/1 Second Round PCBway 3/28 Progress Report 3/29	Neehar: ● Addition of second round PCB and testing Avadh: ● Testing and working on prototype Roshun:

	• Testing and working on prototype
4/2 - 4/8	Neehar: • Finalize complete PCB Design Avadh: • Demo preparation, fixing last bugs Roshun: • Demo preparation, fixing last bugs
4/9 - 4/15	Neehar: • Finalize testing bugs in hardware subsystems Avadh: • Finalize testing Roshun: • Finalize testing
4/16 - 4/22 Team Contract 4/16 MOCK DEMO WEEK	Neehar: • Practice and prepare for mock demo Avadh: • Practice and prepare for mock demo Roshun: • Practice and prepare for mock demo
4/23 - 4/29 FINAL DEMO WEEK	Neehar: • Prepare for final demo, work on final report Avadh: • Prepare for final demo, work on final report Roshun: • Prepare for final demo, work on final report
4/30 - 5/6 FINAL PRESENTATION	Neehar: • Prepare for final presentation

WEEK	Avadh: ● Prepare for final presentation Roshun: ● Prepare for final presentation

4. Ethics and Safety

An ethics issue may arise if an individual were to gain access to the user's phone and use the application to get access to their mail. This can become a risk to the user as someone who gets ahold of their phone will potentially be able to get access to the sensitive documents in their mail as well. In the IEEE Code of Ethics Section 1.1, it is noted that our product must "protect the privacy of others". An individual would be able to compromise the privacy of the user if they were intentionally misusing the product and deceiving the user. A solution to this potential privacy issue would be authenticating the user before being allowed to use the application.

The reliability of our product could result in violations of the ACM Code of Ethics Section 2.9 which states that you must "design and implement systems that are robustly and usably secure". A safety issue that may arise is if one of the components in our product malfunctions. This has the potential of either locking the door permanently or always leaving it unlocked. If it is locked permanently, the user will not be able to safely access their mail. If it is always unlocked, the user will have a false sense of security that their mailbox is locked when it is not. To ensure that this does not occur, we will be testing our product under a plethora of different scenarios to make sure that it does not malfunction.

Another important pillar of the IEEE Code of Ethics is Section 1.5 which partly states that we must be "honest and realistic in stating claims". In order to be transparent with the user regarding the functionality of our product and the possible privacy implications regarding misuse and component failure, our product will not guarantee safety of mail or items placed in the mailbox.

5. Citations

"Magnetic Door Switch Set." *COM*, https://www.sparkfun.com/products/13247.

"Ultrasonic Distance Sensor - HC-SR04." *SEN-15569 - SparkFun Electronics*, https://www.sparkfun.com/products/15569.

Grainger Engineering Office of Marketing and Communications. "Salary Averages." *Electrical & Computer Engineering | UIUC*, https://ece.illinois.edu/admissions/why-ece/salary-averages.

Rivera, Tanya. "Rash of Mailbox Thefts: Stop Putting Checks in Your Mailbox." *wfmynews2.Com*, 17 Feb. 2022, https://www.wfmynews2.com/article/news/local/2-wants-to-know/stop-putting-checks-in-your-mailbox-thieves-steal-checks-wash-change-amount-cash-dont-put-mail-in-blue-usps-mailboxes-overnight-post-office/83-fc6b3146-5565-4a32-8a58-e22b2bab835d.

"HS-311 Standard Economy Servo." *HiTec*, https://hitecrcd.com/products/servos/analog/sport-2/hs-311/product.

Conn, Marla. "Build an IOS Application." *Amazon*, 2021, https://aws.amazon.com/getting-started/hands-on/build-ios-app-amplify/.

"IEEE Code of Ethics." *IEEE*, https://www.ieee.org/about/corporate/governance/p7-8.html.

"ACM Code of Ethics." *ACM*, https://www.acm.org/code-of-ethics.