California State University of San Bernardino

Project Report

Presented May 14, 2021

Household Verification System

Syndicate
Team Members:
Miguel Pacheco
Buntaviret Soun
Nicholas Yniguez

1. Executive Summary

Our first section is the problem description which gives background statistical data on homeowners who do not have security systems, and burglary rates on homes. Next, in order to create a security system, we went through the requirements specification for our project as we did a mock demonstration to see what we would need in order for the system to work properly. The question that came to mind was - what if the homeowner did not want to use a facial scan to unlock the door? Therefore, we decided to have an alternative solution to this by adding a virtual and external keypad to unlock the door. A diagram is given showing what the input and output of the system would look like. Next, a system design is needed to help make sense of the flow of the system. There are two components the system uses for the flow, that being the software and hardware. Since the system design was created, we understand what we needed in full detail. So we separated it into two categories as a need and a want. Finally, after getting everything from the requirements to the design implementation, the system needed to be tested. Our results to design the system came into fruition and it came out very well with the exception of not meeting all of the requirements of our clientele's agreement. With that completed, an economic analysis was needed to see how much our project would cost, including the labor for the engineers as well as the equipment. Next, we included our project management timeline using the Gantt chart that shows each task is done for that specific week and which task was potentially delayed or finished early. Even though we completed our project, there are always improvements to every system design. Therefore, we listed a couple of enhancements for what our system may potentially include if we were to improve it in the future. The famous motto "Team Work Makes The Dream Work!" came to mind when we finished our project, so we listed what each team member did for this project report. Finally, we used a couple of sources from the web to help make our project. So we included the reference section at the end giving the credit to the people who had some form of the same ideas as we do.

2. Problem Description

The reason our team decided to go for this idea of creating a facial recognition lock for homeowners is because a lot of homeowners experience break-ins, and some of those break-ins happen to homeowners who think they live in a safe neighborhood, and never would expect to experience this. We want homeowners to feel a sense of security when they leave their house, or when they go away for a few days. According to [1], 46.9% of the people they surveyed do not have some sort of home security system installed in their home. Also, 34% of burglars use the front door when breaking into a home. Also, according to [2], there are roughly 2.5 million burglaries that happen a year, and 66% of those are home break-ins. With these statistics in mind, we decided to design a facial recognition lock, that will have a database of accepted users that the admin (homeowner) will have access to.

This is an important problem because homeowners want to feel safe when they leave their houses for a day or more, and want to have a better sense of security for their families. The design that we want to create should have multiple images of the user's face, so that when the system is trained, it can detect whether or not the user is an "accepted" user or not. We train the trainer to recognize the users each time a new user is added into the system, or when a user is deleted from the system, so that the system can match the user with the user the trainer thinks it is.

Some constraints we need to take into account when creating our design is that we should be able to help those with certain needs, such as people who are visually impaired and those who can't hear. In our design, we need to take those factors into account because we want the system to be available to a wide range of people, and we can accomplish this by meeting the needs of these constraints. For the visually impaired, we plan to make the text as big as possible, with the display that we chose, and without having the GUI of the system be overly cramped with labels of text that have a big font. We also plan to add background colors to certain frames, such as red backgrounds to signify that the user has been denied, and green backgrounds for those who are accepted. Also, we plan to add braille to the external keypad, so that those who are visually impaired can enter in their passcode without having to rely on someone else always being with them to access the door. Also with the visually impaired, we plan to add a voice function into the design. Meaning that in certain key steps, the system will prompt the user to either enter in their passcode, enter in their information when creating a new user, and when the system goes to the "Home Screen" of the GUI.

3. Requirements Specification

A user friendly interface that allows the option to add or remove data recorded from the database. In the add option, it will have a button to start the facial recognition and then allows the user to input any additional information with the data that has been taken. Once all descriptions are complete, the user may confirm it. It will then go back to the main screen to add or remove data. In the remove option, it will show the list of users in the database that will have a remove button with confirmation afterwards.

The user must stand at a fixed location to allow the camera to recognize the user. For a new user, the user must stand until the program takes a picture of them and adds it to the database. Users may add additional information but a passcode is required to be inputed here. For users already in the database and the face is not recognized, the user will be prompted to use the keypad to unlock the door. When a picture of the user is taken, it is added to the database and stored there, until a user accesses the facial recognition.

There will be an LCD screen that will display the welcome message. When entering the passcode, the password will be shown in the LCD screen with an asterisk for each input. Green LED will be shown when access is granted and a red LED will be shown when access is denied. With this LCD screen, and the incorporation of the green and red backgrounds of the display, we can broaden our reach of the system and provide more help for people who need it.

The result of the device when tested should be such that the facial recognition can recognize the user when the scan/picture of the user's face is being compared to the pictures in the database. Also, the led must flash whether the user is recognized or not, green or red respectively.

4. System Alternatives and Alternative Selection

When we were tasked with generating a requirement specification document for this product, we started by first making an input output diagram, which you can see below. We knew we needed a means of allowing user input such as name, passcode, and pictures. As the output, we needed a lock to disengage and a welcome message on screen and voice operation. With that said, we came up with a design that utilized a 4x4 keypad, a 16x2 character LCD display module, a lock solenoid, a camera module, and a Raspberry Pi for the hardware. However after meeting with our client and discussing the needs of the system, we decided to go with a more user friendly design by upgrading the display to a seven inch touchscreen display. Instead of scrapping the 4x4 keyboard, we decided it would prove useful to modify it to have braille. Another important hardware component to upgrade was the camera module. Although we did not have a specific camera module picked out, we knew it would have to be improved from what we originally had in mind.

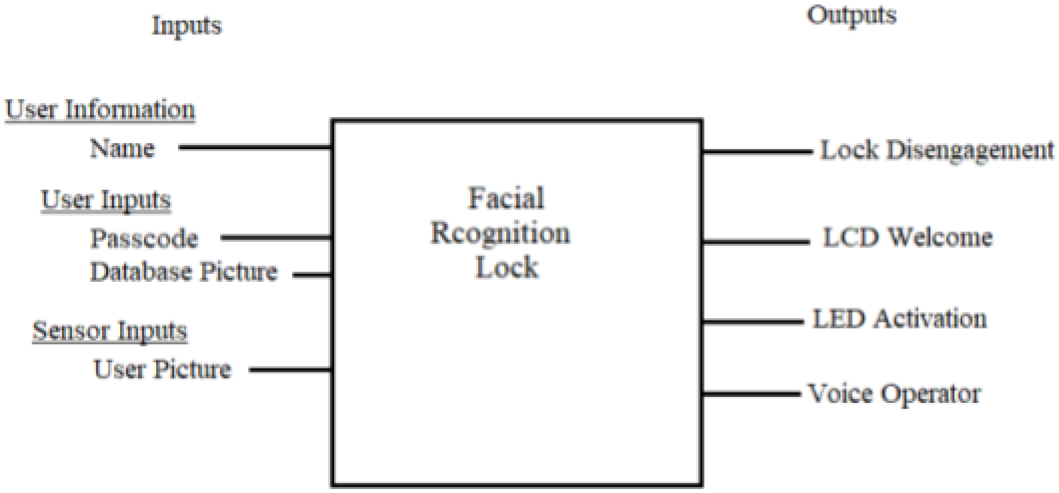


Figure 1: Input and output of the design

These upgrades would allow for a wider variety of people to use the system, such as those with or without disabilities of all sorts. A 16x2 LCD would have proven difficult to use, if not impossible, for a visually handicapped person. The improved display also allowed the use of bright pass/fail cases. The braille helped for those unable to see the screen in the event of failed face recognition. The improved camera camera helped for gaining access into your home when it's dark out, whereas our original camera idea did not account for this. With that said our first design might have been sufficient for general public use, but insufficient for our client. Considering costs, our final decision did come out to be more expensive, however worth it; without these alterations, a wide variety of the public would not be able to use our product. Our final decision for the product did not change the timeline or the projected completion date.

5. System Design

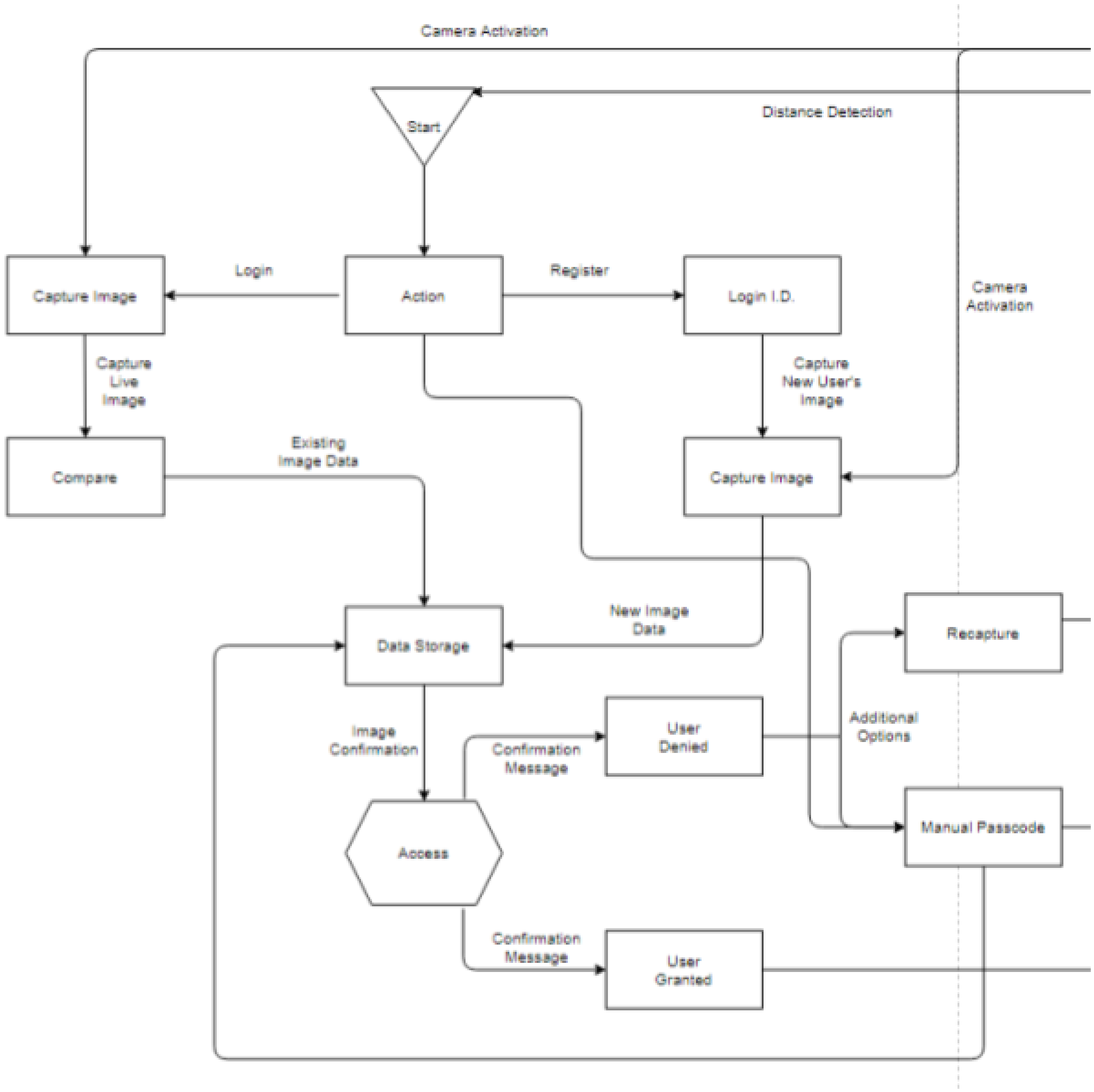


Figure 2: Block diagram for software

Specifications of the Blocks: Although there are 25 blocks in the block diagram, some blocks reference an action that the system will perform, and are thus reusable. A reusable block is independent upon what block comes before or after this block; it will always perform the same action. The blocks in the diagram each have an arrow pointing to what step comes next when accessing the facial recognition lock. The boxes in the figure indicate the action taken. Since this specific block diagram is one containing the entire system, it includes aspects of both hardware and software. Single pointed arrows indicate the control flow, which is the order of steps the flow of the facial recognition happens chronologically. Double-sided arrows indicate that the flow can go both ways, meaning each block with this type of arrow can access data from its corresponding block and vice versa. For example, if the camera is connected to the raspberry pi with a double sided arrow, then data or commands can flow in both directions; the camera can send image data to the raspberry pi, and the raspberry pi can send instructions via signals to the camera.

Action Block: The first block of the diagram is the start block. It is a sensor module that will send a signal to the raspberry pi, powering on the system. It will always be active so that it can detect a user at all times. When the sensor module detects a person, it will wake the system up and the action block will commence. This block is the first part of the user interface. It will display two options on the screen. Whichever option the user picks will dictate which step comes next, and therefore, which block will be activated. The user's options will be Login and Register. If there is no existing user data, the system will only display the Register option. When the register block is activated, the user interface will transition to a registration screen where it will ask for the user's name, and a digit passcode in case the user's picture fails to unlock the system. After this step, the capture image block will be activated. This block is one of the aforementioned reusable blocks. It automatically activates the camera block. The camera block is the camera module, a piece of hardware that captures images to be used to unlock the system. After the camera takes the users picture, it stores the data in the next block, the data storage block. The Register option is now completed, and no block will come after it. The user will then be redirected to the initial screen where they will now see two options: Login and Register.

Data Storage Block: Now that the system contains user data, the user may select the Login option. This will activate the capture image block, a reusable block. Since this block is reusable, it must do exactly what it did in the register option: call upon the camera block. The camera module will activate, and will capture an image of the user. The next block is the compare block which points to the data storage block. The data storage block has a double sided arrow pointing to the raspberry pi. This will allow data to be sent to and from either block. The compare block will commence code that is yet to be written. The purpose of this code is to compare distinctive features in a picture taken to a picture saved in the database. Since the compare block points to data storage, which points to raspberry pi, comparisons can be made there.

Access Block: The access block is next in chronology. One of two results can come from the access block; denied and granted, each of which are their own blocks. The user will have their "live" picture taken with the camera, and the system will compare their picture with the photos in the database. This part is important because it can determine if the user has been accepted, or denied, and whether or not they will have to take extra steps in order to gain access to the lock, when they themselves know they should be given access.

Granted Block: This block is triggered when the user has been accepted as an authorized person when the system compares the user's face with the pictures the owner has in the accepted users database. The system will display an access granted message for those with hearing impairment; an automated voice will emit an access granted message via speaker for those with visual impairments. In the case of a user having total visual impairment, an external keyboard containing braille will be provided. Simultaneously, while displaying an access granted message, the raspberry pi will send a signal to the next block: relay module. This relay module is a piece of hardware that will unlock the locking solenoid,which is the final block in terms of steps. The locking solenoid block will have constant power supplied to it, which is 12 volts in this case. However, it will only unlock when the relay module block sends the required signal to it. After being given access and the lock unlocking, the user may enter the home.

Denied Block: The other result of the access block is the denied block. This block will be reached if the user's face is not recognized with the photos in the database of accepted users. This block will also have two options for the user to select through the user interface: recapture and manual passcode. The denied block is giving the user a chance to either recapture their image, or enter their passcode that they have created as a failsafe.

Recapture Block: If the recapture block is selected, the camera module will be reactivated, and the user will be given another chance to login via facial recognition scan. Again, since the camera block has double sided arrows going to and from the data storage as well as raspberry pi blocks, image comparisons will recommence. If a match is made, the raspberry pi will send the signal to the relay module, unlocking the lock. In the event that the user has not been recognized as an authorized user, they can opt out and take the manual passcode route, as that will make it as a last resort for the user to have access to the lock. The user can choose to go for this option again in the hopes that they can gain access through their recaptured image, rather than having to manually enter their passcode.

Manual Passcode Block: If the manual passcode block is selected, the user will then be prompted to input the passcode they made during the Register block. The user may input their passcode into the external keypad, or the touch-keypad on the display, whichever they would prefer. The manual passcode block sends the data to the raspberry pi, which will then send the unlock signal to the lock via the relay module. This block is selected in the event the user was not given access to the lock, and they chose this option as their other way of verification to gain access to the lock.

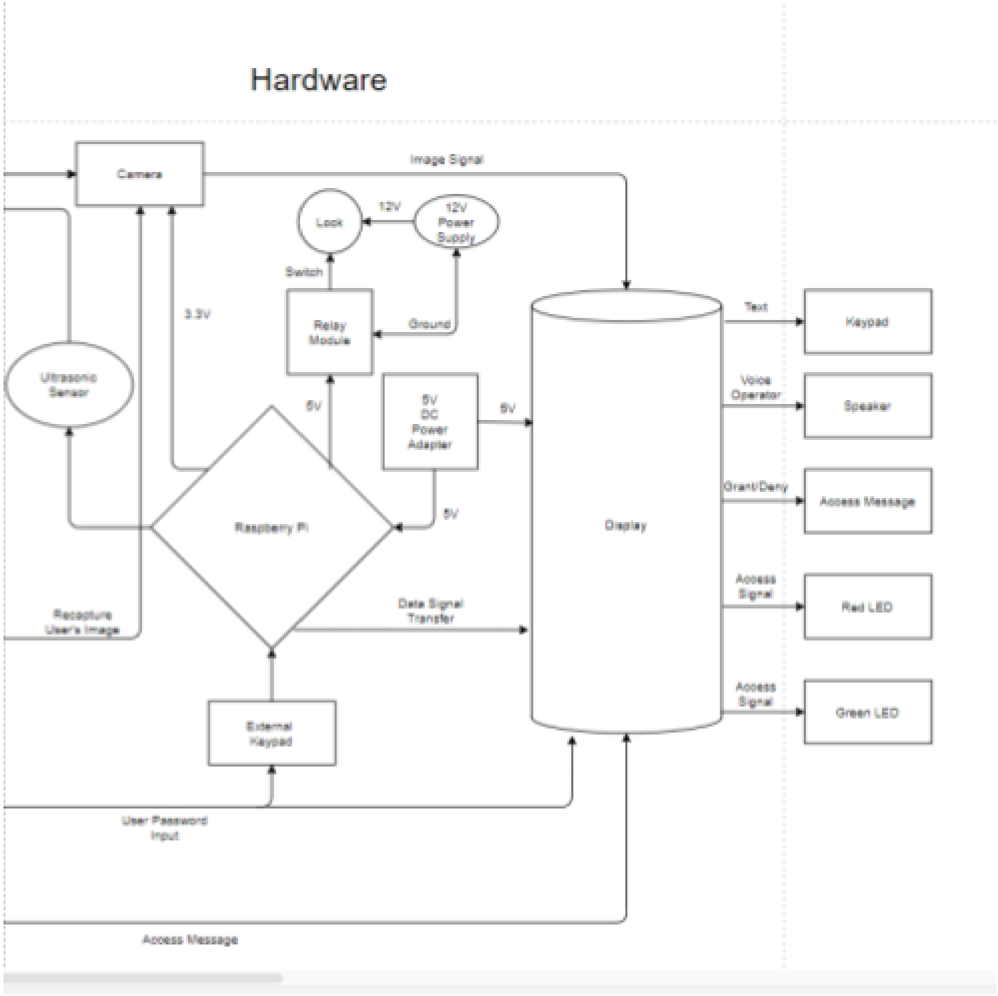


Figure 3: Block diagram for hardware

Camera Block: After being prompted by Capture Image or Recapture block, the camera block will proceed to focus on the image before taking a picture. The camera leads to the Display block, and this will show the users on the display so the user/s can adjust themselves properly before being put into the database. The camera needs a minimum of 3.3 volts to activate, which the Raspberry Pi can give as it has a slot that gives that exact voltage. Therefore, the camera is connected to the Raspberry Pi because it needs power to function.

Ultrasonic Sensor Block: This block is connected to the Raspberry Pi. It activates as soon as an object is within the distance of the product. This leads to the startup block that basically sets up all the given options (login or register) that the user can choose.

Lock Block: The Lock block contains a 12 volts solenoid lock. There needs to be an external power supply of at least 12 volts since the Raspberry Pi only provides a max output of 5 volts. After the user is granted access from the Access block or Manuel Passcode block, it will send a signal to the Raspberry Pi block. The signal will then proceed to the relay module and transfer the message to the lock. The lock will then proceed to unlock itself. It will only stay locked if the user is denied access.

12V Power Supply Block: An external power supply that is solely there to unlock the 12V solenoid lock. This power supply will be active at all times, so that it can unlock the door as soon as it is prompted. A red wire is connected to the lock that gives out the voltage as the black wire is connected from the relay module as ground.

Relay Module Block: The relay module is a separate hardware device used for remote device switching. With it you can remotely control devices, in this case is the lock. It is connected to the Raspberry Pi because it contains the signal from the Access Block. The signal tells the Raspberry Pi to either to activate the switch or keep the switch off. Which then leads the signal to the relay module to provide that switch action.

5V DC Power Adapter: This block is a power supply, and provides the necessary power to turn on the Raspberry Pi and the Display.

External Keypad Block: This block happens as a result of the user choosing the manual passcode route, and opting to use the external keypad rather than the touch-keypad. This external keypad also serves the purpose of helping the people who are visually impaired; it will have braille on the buttons. When the user inputs their passcode on the external keypad, it will send that information to the raspberry pi and compare it with the passcode imputed when the user signed up to determine if they are the accepted user.

Display Block: This block gathers everything and displays them here. All of the software signals use the display block to project the message that the software is trying to display. From start to finish, it shows the user registration with login identification inputs, shows what the camera is projecting, and gives the user the option to choose whether to recapture or manually bring up a virtual keypad to let them input their passcodes. It will then display an access message from the Access Block and will also flash the LED's with the message displayed.

Keypad Block: A virtual keypad on the display to allow the users to input their passcode to access through the door.

Speaker Block: This block is connected to the display to allow the voice operator to speak what is currently on the display. The Speaker block is a must for the visually impaired as it also prompts the user to follow certain steps to navigate through the process of opening the door. This speaker block also ties in with the access message block because the access message will activate, and the speaker block will say that the user has been granted access.

LED blocks: Both the red LED and the green LED block will act like signals for the user. The green LED will display a green color when the user has been given access, and the display screen will turn red when the user is denied access.

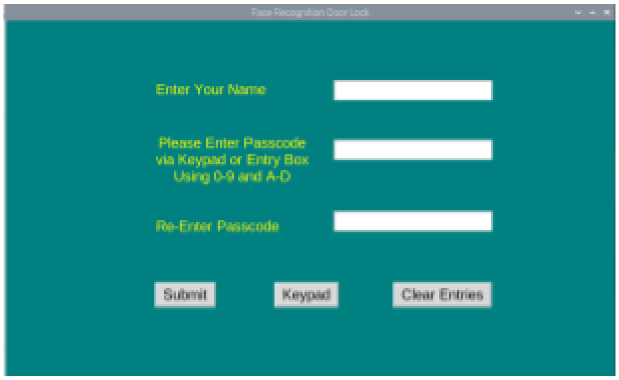
Raspberry Pi: The Raspberry Pi block is the computer for our project. It receives power from the 5V power supply, and sends power to most of the modules in the Facial Recognition Lock, such as the camera, the external keypad, and the relay module. It also sends and receives data from the camera module, the external keypad, and data storage.

6. Detailed Design

The principle of operation is to have the facial recognition lock to recognize the user that is in the database, with the user that is trying to gain access, and allow them to have access to the lock whether it be via passcode or picture compared to person. The plan for this is to have the owner/user have a storage where they can add the people they want to give access to, and can allow the user to either add/delete people from the database.

In implementing the hardware, the system starts by powering on the Raspberry Pi. Once it is powered on the screen will also turn on. After the program is run, the user will be prompted to make a selection to either scan their face, or make administrative changes, which we cover in the software implementation. If the user chooses to scan their face, power will go to the camera to begin the unlocking process. Once this is successfully completed, the Raspberry Pi will send a signal to the relay, which will then send power to the lock solenoid, unlocking the door. The door will automatically lock itself after a predetermined amount of time.

In the software implementation, the system starts by prompting the user to either login, or register, if the user does not actively have accessible entry. From there, if the user is a new user, it will add them to the database so that they can gain access to the lock. When the user enters their login credentials, it will compare them with the database's and determine if they are granted or denied access. Next, if access was granted, the system will display a welcome message and unlock the door. However, if the user was not recognized, the system will prompt them to either enter their passcode manually, or to recapture their face to compare it again, and there will be led's to indicate to the user whether they are granted access or not. We can access the user's profile by pressing on the user that wants to access their profile, but will need to enter in the pin for the specific user. When they enter their profile, they can access the lock directly, just by having their face scanned, or they can enter in their passcode/have their face scanned so that in the next page, they can either change their passcode, or delete the profile in its entirety. When that happens, the user will be deleted from the database, and their pictures will also be deleted. There will also be an admin mode, so that the admin can directly change the information of any user, in the event a user has forgotten their passcode when their face scan has been denied. The admin can change that information by re-entering in their passcode to confirm that the information is to be changed. However, the admin themselves cannot change their passcode or any other information, for security reasons.



Please Enter Passcode via Keypad or Entry Box Using 0-9 and A-D

Figure 4: When the system turns on for first time.

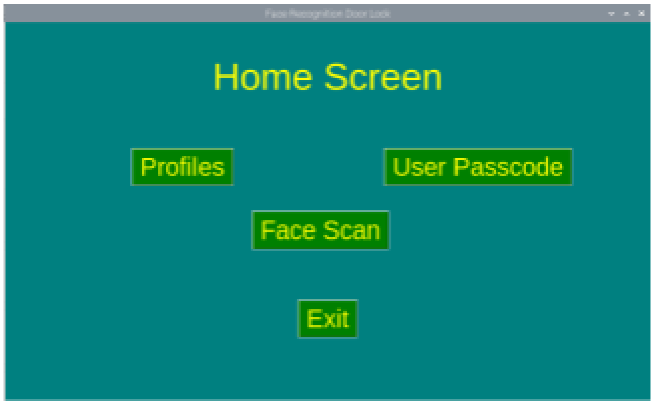
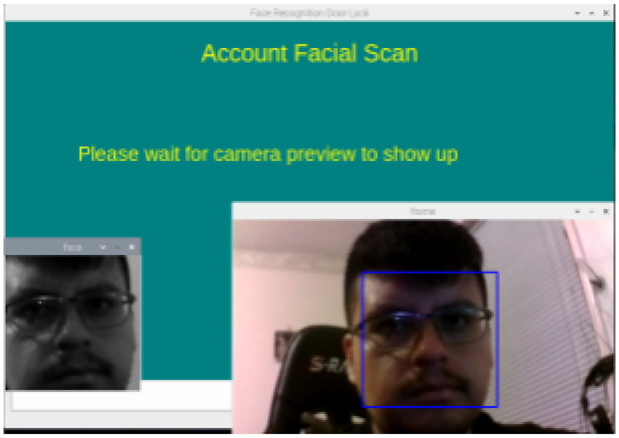
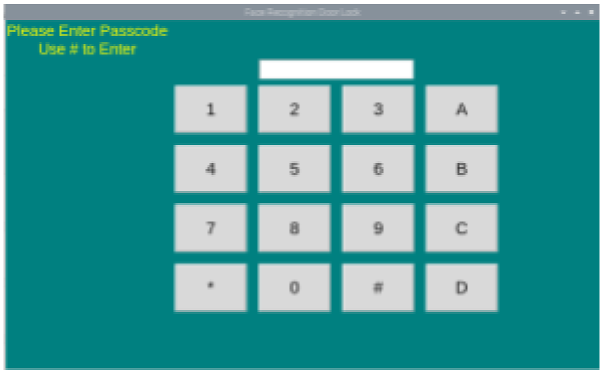


Figure 5: Home Screen



Please wait for camera preview to show up

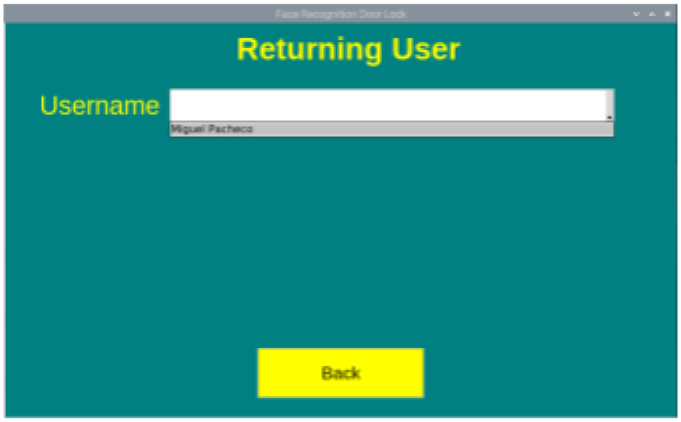
Figure 6: User's pictures being taken



Face Recognition Door Lock

Please Enter Passcode
Use # to Enter

Figure 7: Virtual Keypad



Face Recognition Door Lock

Returning User

Username

Miguel Pacheco

Figure 8: Dropdown menu of users

7. System Test Plan and Results

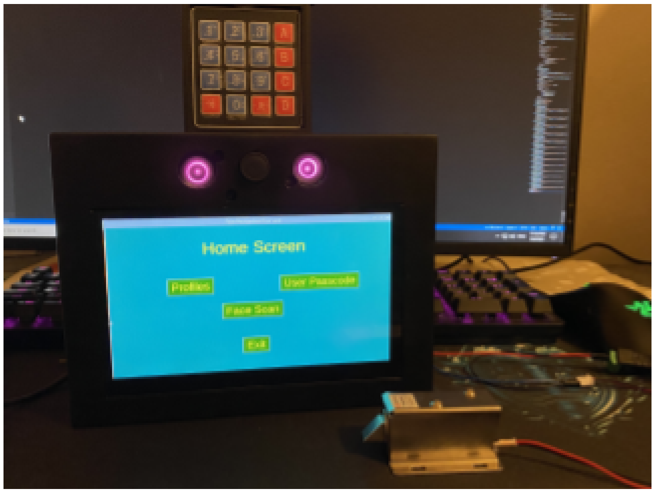


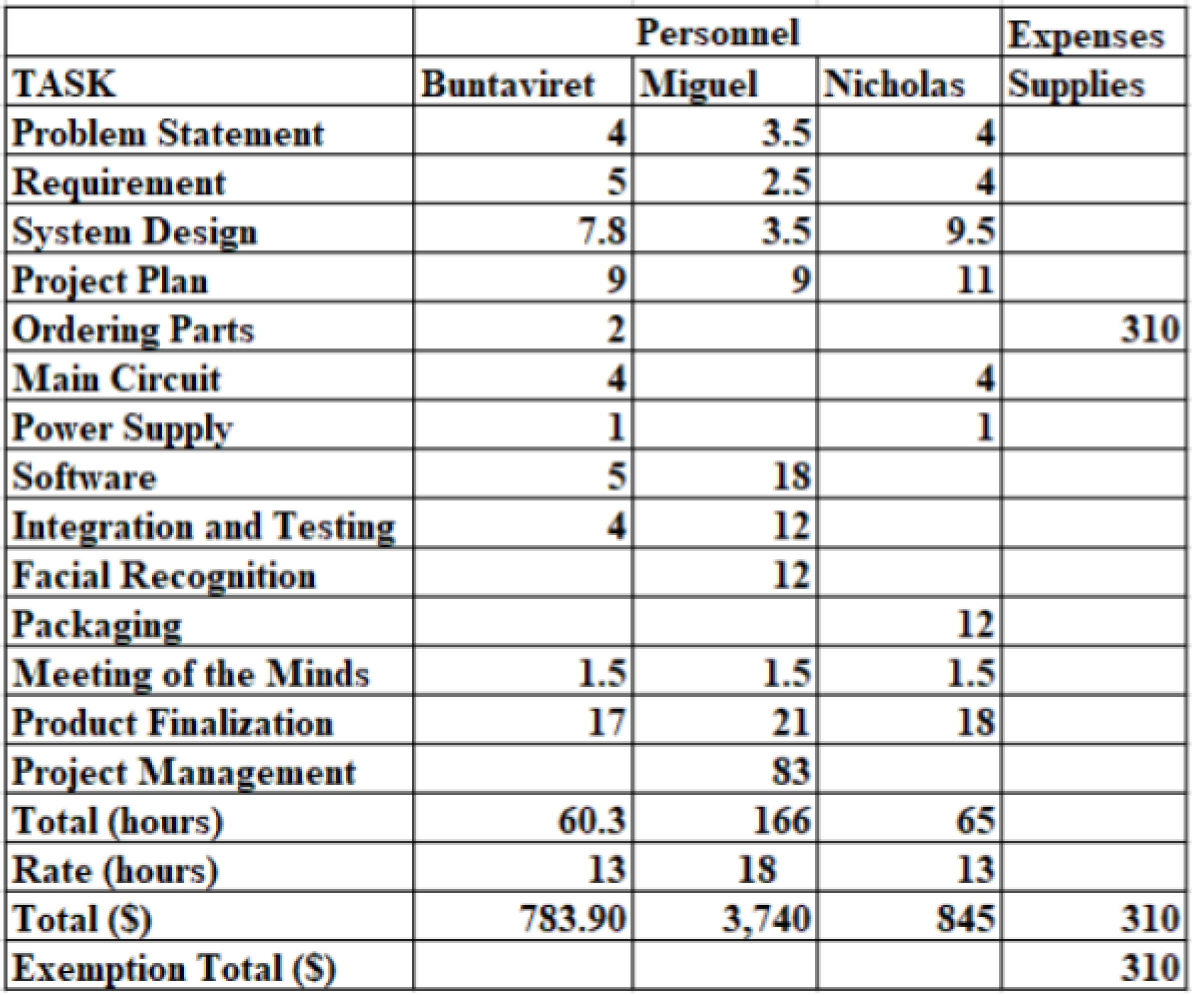
Figure 9: Finished design of the system

To test the system, we decided to test each part individually, to make sure that they work properly. Firstly, we tested the external 4x4 membrane keypad, to make sure that the key presses were being read when running from the pi termal. Next, we tested the solenoid lock by having the lock disengage for 5 seconds, and then re-engage when the 5 seconds have passed. To test the PiCamera, we need to make sure that there are no issues when the PiCamera is taking pictures of the user, and that the PiCamera closes properly each time, so that there are no errors when multiple users are going to have their picture taken in one run. We also test to make sure that the trainer is able to be trained to recognize the user when called in the recognizer file. Next, we tested the solenoid lock with the external keypad, making sure that when the user inputs the correct passcode, the lock disengages for a few seconds, and then re-engages when the time has passed. Then, we made sure that the lock also disengages when the user is recognized by the recognizer, and re-engages when time has passed. The next step was to make sure that the Graphical User Interface (GUI) is able to work with the PiCamera, solenoid lock, and the external keypad. We made sure to test each part separately with the GUI, so that we can see the results of each part. We also made sure that the database works properly with the tkinter module, and that when the user is added to the database, that information can be accessed through the GUI, and can also be manipulated through the GUI. With the GUI, we also made sure that the virtual keypad is able to receive the inputs from the user when they press the buttons on the GUI, and that the information that is grabbed from the virtual keypad can be stored in the database. When everything is tested, we can then incorporate everything together, and test to make sure that the system works.

When we tested the external keypad, we made sure that in the code, we can read the keypresses on the terminal, to make sure that we wired the keypad correctly to the pins on the raspberry pi. Since this was a successful test, we can then move onto the next test, which was testing the solenoid lock. We made sure that the GPIO pins are being read correctly, so that the lock disengages, and this resulted in a pass, so now we were able to move onto the next test. For the PiCamera, we made sure that the ribbon cable is connected properly to the raspberry pi, and configured the camera setting on the raspberry pi. To test the camera module, in the terminal we can use *raspistill -o Desktop/image.jpg* to see the camera preview, before the camera takes a picture. We then tested to make sure that the PiCamera module can take multiple pictures of the user, and get the trainer trained to recognize this user. Once we did that, we then tested the recognizer by using the trainer file that was created when the trainer was trained, and since it was able to recognize the user with a decent confidence level, we were able to move onto the next test, which was testing the solenoid lock with the external keypad. With this test, we made sure that the correct GPIO pins are being called in the code, and when the correct passcode is entered, we can use time.sleep to make the program sleep for a few seconds, to allow the lock to disengage, and once the seconds have passed, the lock should re-engage. With this test being a success, we were then able to move onto the next test which was having the solenoid lock work with the recognizer. Since the recognizer was able to recognize the user, we can add onto the recognizer file we created, to disengage the lock when the user is recognized, and since this test passed, we were then able to move onto the next test. The major test's of the system was with the GUI, as the GUI is what would control everything in our system. We made sure that when a new user is added, the new user's information is added into the database, and their pictures are taken, so that we can add them into the trainer, to retrain it to recognize each distinct user. With this passing, we made sure that the tkinter module is able to have the lock disengage when the user is recognized when doing a facial scan. When this test was completed, the last thing we needed to make sure was that the lock disengages when the user uses the external keypad as means to access the door using the user interface. As there were problems in this test, such as the lock re-engaging right after it disengages, we were still able to show that the lock disengages when the user enters their passcode correctly using the keypad.

The results of all the test's we've done on the system has allowed us to nearly meet all expectations with the clientele. We created an administrative overview that is able to control other users in the database for editorial purposes. Facial recognition scans multiple times to make sure we get the best pictures of the users. The project is able to use a virtual keypad and the keyboard with all the buttons of a regular computer. We have the braille implemented on the external keypad for the visually impaired. Depending on when the user is granted or denied access, a bright green or red color will appear. Lastly, the solenoid lock will automatically lock itself after 5 seconds of unlocking. Now the obstacles that didn't allow us to meet the clientele's expectation is that we considered voice assistance as a must for the visually impaired. However, the pi display keeps turning off and on every few instances when the program is assisting, so we decided to not include it. Also, when using the external keypad to unlock the door, the door immediately locks itself after unlocking.

8. Economic Analysis

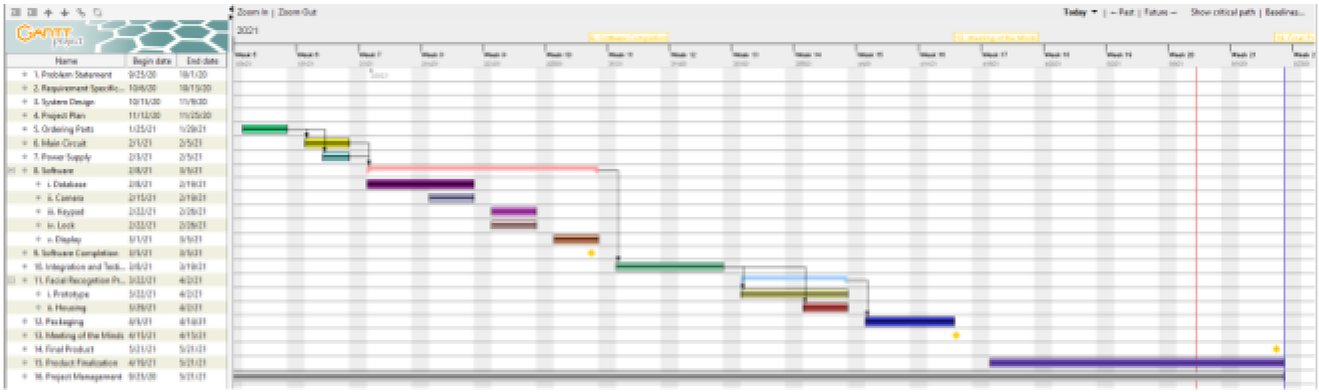


	Personnel			Expenses
TASK	Buntaviret	Miguel	Nicholas	Supplies
Problem Statement	4	3.5	4	
Requirement	5	2.5	4	
System Design	7.8	3.5	9.5	
Project Plan	9	9	11	
Ordering Parts	2			310
Main Circuit	4		4	
Power Supply	1		1	
Software	5	18		
Integration and Testing	4	12		
Facial Recognition		12		
Packaging			12	
Meeting of the Minds	1.5	1.5	1.5	
Product Finalization	17	21	18	
Project Management		83		
Total (hours)	60.3	166	65	
Rate (hours)	13	18	13	
Total ($)	783.90	3,740	845	310
Exemption Total ($)				310

Figure 10: Economic analysis chart

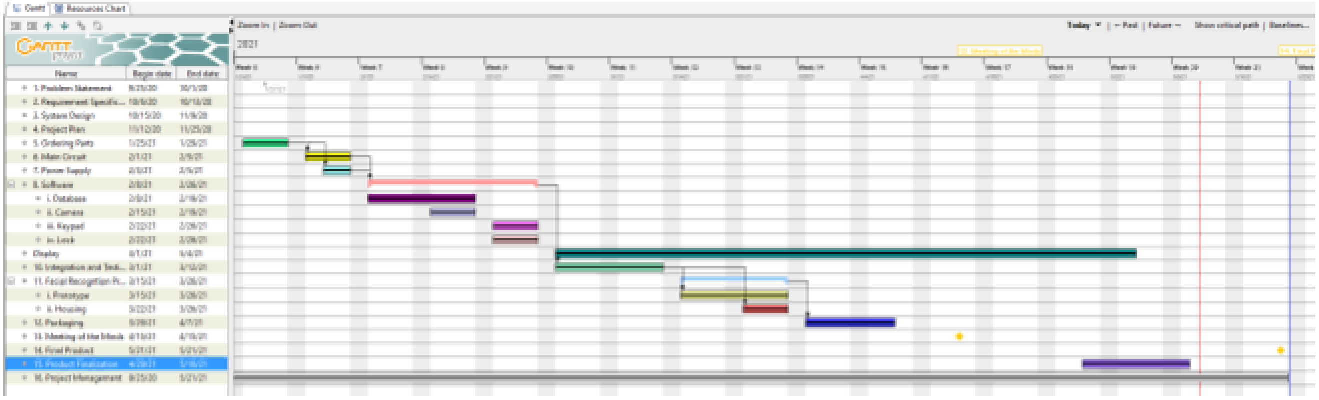
Our group was assigned specific tasks to do. Some others had the same task and some had individual tasks. Our group hourly pay rate was based off of a specific role. Miguel's pay rate is 18 dollars since he was assigned as manager and software engineer. Due to being a manager, Miguel had the most responsibility as he was our group leader. Therefore, his pay rate is much higher than the other two members. Nicholas and Buntaviret's pay rate was only 13 dollars since they were just labeled engineers. The personnel columns show the hours each member put in for each of the tasks. The expense column shows the price of the supplies that were bought. However, since this was a school project and not the actual contractors being hired, we decided to exempt the total labor cost of all the personnels involved. This leaves the total cost of the project coming out to be only the cost of the supplies, at 310 dollars.

9. Project Management


Zoom In | Zoom Out
Today ▾ | ← Past | Future → Show critical path | Baselines...
2021
GANTT project

Name	Begin date	End date
1. Problem Statement	9/25/20	10/1/20
2. Requirement Specific...	10/6/20	10/15/20
3. System Design	10/15/20	11/9/20
4. Project Plan	11/12/20	11/25/20
5. Ordering Parts	1/25/21	1/29/21
6. Main Circuit	2/1/21	2/5/21
7. Power Supply	2/3/21	2/5/21
8. Software	2/8/21	3/5/21
i. Database	2/8/21	2/19/21
ii. Camera	2/15/21	2/19/21
iii. Keypad	2/22/21	2/26/21
iv. Lock	2/22/21	2/26/21
v. Display	3/1/21	3/5/21
9. Software Completion	3/5/21	3/5/21
10. Integration and Testi...	3/8/21	3/19/21
11. Facial Recognition Pr...	3/22/21	4/2/21
i. Prototype	3/22/21	4/2/21
ii. Housing	3/29/21	4/2/21
12. Packaging	4/5/21	4/16/21
13. Meeting of the Minds	4/15/21	4/15/21
14. Final Product	5/21/21	5/21/21
15. Product Finalization	4/19/21	5/21/21
16. Project Management	9/25/20	5/21/21

Figure 11: Expected schedule of the project


Gantt
Resources Chart
Zoom In | Zoom Out
Today ▾ | ← Past | Future → Show critical path | Baselines...
2021
GANTT project

Name	Begin date	End date
1. Problem Statement	9/25/20	10/1/20
2. Requirement Specific...	10/6/20	10/13/20
3. System Design	10/15/20	11/6/20
4. Project Plan	11/12/20	11/25/20
5. Ordering Parts	1/25/21	1/29/21
6. Main Circuit	2/1/21	2/5/21
7. Power Supply	2/3/21	2/5/21
8. Software	2/8/21	2/26/21
i. Database	2/8/21	2/19/21
ii. Camera	2/15/21	2/19/21
iii. Keypad	2/22/21	2/26/21
iv. Lock	2/22/21	2/26/21
Display	3/1/21	5/4/21
10. Integration and Testi...	3/1/21	3/12/21
11. Facial Recognition Pr...	3/15/21	3/26/21
i. Prototype	3/15/21	3/26/21
ii. Housing	3/22/21	3/26/21
12. Packaging	3/29/21	4/7/21
13. Meeting of the Minds	4/15/21	4/16/21
14. Final Product	5/21/21	5/21/21
15. Product Finalization	4/26/21	5/18/21
16. Project Management	9/25/20	5/21/21

Figure 12: Actual schedule of the project

When creating the Gantt chart for our project, we thought that creating the Graphical User Interface (GUI) would be an easy task, and the integration would be the hardest part of the project. As seen from the expected schedule, we had high hopes that we can create a user interface easily, and can have time to spare to make improvements in the system. In reality, we were wrong about how difficult it would be to create the GUI for our project, and how difficult it would be to integrate the camera, solenoid lock, and external 4x4 membrane keypad with the GUI. Because of this, we delayed a few tasks until we could get a “working” prototype of the GUI, so that we could test the face scan, external keypad, and solenoid lock. Even though this caused us to delay certain tasks because of the GUI, we were able to go ahead with other tasks that did not rely on the GUI. Nicholas and Buntaviret were able to go ahead and complete the encasing, and wire up everything that is needed for our project, while Miguel kept on working on the GUI, so that the user interface is user friendly. By the time Buntaviret and Nicholas finished the encasing of the system, Miguel had a working prototype of the GUI we are going to use, and was given the system to make sure that the important key steps such as accepted, denied, lock disengaging, and many other things were working properly. Once Miguel made sure that they were working properly, he continued to improve the GUI by incorporating the external keypad with tkinter, so that the entry label can be updated with every keypress the user presses on the keypad. As seen in the actual schedule, we skipped over the display task and started on the integration and testing of the functions that we already had, such as the camera taking pictures of the user, the system recognizing the user, the keypad unlocking the solenoid lock when the passcode is entered correctly, and the solenoid lock unlocking when the user is recognized by the recognizer.

One key discrepancy we had when creating the GUI was finding a way to incorporate the external keypad with tkinter, so that with each keypress on the keypad, the entry label on the user interface would update, so that the user can see that the passcode they are inputting is of correct length. The problem that came up was finding a way to get the entry label to be updated each time the user presses on the keypad, and have the solenoid lock disengage when the passcode is correctly entered for the user. This problem took a while to solve because we had to research how to get the entry label to update with the keys of the external keypad, and find a way to implement this. Miguel found a way to get the entry label to update with each keypress, and get the solenoid lock to disengage when the user correctly enters their passcode. This discrepancy did not affect the timeline of our project, since we had tested beforehand that the keypad worked correctly, and the solenoid lock disengages when a passcode is entered correctly, and tested the keypad with the solenoid lock when we made the encasing and wired everything.

10. Summary and Future Work

In implementing this product, our team learned many useful skills that will serve us in the future. Some of these skills include using OpenCV for our face recognition, MySQL for building the database, Tkinter for building the graphical user interface, TinkerCAD for building the 3-dimensional model that would be our encasing, and many more skills. However, there are a few things that we overlooked, and may have been able to do differently. That said, our team may still implement these changes as future work to enhance the product. Perhaps the biggest drawback to our design is the fact that a power outage will render a door employing this product permanently locked until power is restored. This can be fixed by modifying the lock solenoid to include a deadbolt mechanism that can be unlocked via key. It would not require much alteration to the design, since the lock solenoid is not inside the encasing, and it would not require additional power or software to operate. Another future enhancement we want to implement into our design is utilizing Haar Cascade to enable the trainer to recognize when a user is wearing eyeglasses. Doing so would not require any changes to our block diagram, but would require us to make changes to the code portion.

Some other future enhancements we wish to incorporate into this product include the ability to scan your face on a mobile device instead of on the facial recognition device, receiving notification updates on entry attempts, as well as having the database in a separate location. Having the database in a separate location would circumvent successful hacks on the system, strengthening the overall security of the product. All of these future enhancements would require a major overhaul to the code to function properly, but we would not need to incorporate a wifi receiver, since Raspberry Pi 4B has this feature. However, since we would need this feature, the computer may require additional power to operate properly. This was also an issue when attempting to incorporate voice direction in key steps, stemming from lack of sufficient power.

11. Contributions of each individual member

a. **Miguel Pacheco:** Completed Problem Description, completed Project Management, completed References, part of Summary and Future Work, part of System Test Plan and Results, part of Detailed Design, part of Executive Summary.

b. **Nicholas Yniguez:** Completed System Alternatives and Alternative Selection, completed System Design, part of Summary and Future Work, part of System Test Plan and Results, part of Detailed Design, part of Executive Summary.

c. **Buntaviret Soun:** Completed Requirement Specifications, part of System Test Plan and Results, completed Economic Analysis, part of Detailed Design, part of Executive Summary.

References

1. Covington, T. (2021). *Burglary Statistics & Research from the BSJ and FBI: The Zebra*. Burglary Statistics & Research from the BSJ and FBI | The Zebra. https://www.thezebra.com/resources/research/burglary-statistics/.
2. *Burglary Statistics: The Hard Numbers*. National Council For Home Safety and Security. (2019, December 19). https://www.alarms.org/burglary-statistics/#:~:text=%E2%80%8BThere%20is%20one%20burglary,a%20witness%20or%20physical%20evidence.