Voice Recognition Door Lock

ECE 445 Second Design Document

Team 1: Rahul Krishnan, Rohil Hatalkar, Vaidotas Marcinkevicius
TA: Dhruv Mathur
Spring 2020

Table of Contents

1) Introduction 2
  1.1 Objective 2
  1.2 Background 2
  1.3 Visual Aid 3
  1.4 High-level Requirements List 3

2) Design 3
  2.1 Block Diagram 3
  2.2 Power System 5
    2.2.1 AC/DC Converter 5
    2.2.2 Voltage Regulator 5
  2.3 Control System 6
    2.3.1 Microcontroller 6
    2.3.2 Ultrasonic Sensor 7
    2.3.3 WiFi Module 7
    2.3.4 Microphone 8
    2.3.5 SD Card 9
    2.3.6 Circuit Schematic 10
    2.3.7 State Diagram 11
  2.4 Locking System 12
    2.4.1 Servo Motor 12
    2.4.2 Door Lock 12
  2.5 Software 13
    2.5.1 Mobile Application 13
    2.5.2 Cognitive Services APIs 14
  2.6 Tolerance Analysis 15

3) Project Differences 16
  3.1 Overview 16
  3.2 Analysis 17

4) Cost and Schedule 18
  4.1 Cost Analysis 18
    4.1.1 Labor 18
    4.1.2 Parts 18
    4.1.3 Sum of Costs 19
  4.2 Schedule 19

5) Discussion of Ethics and Safety 20

6) Citations 22

1) Introduction

1.1 Objective

Classic lock-and-key mechanisms come with a variety of inconveniences including the inability to fit a key into the door lock or unlock the door when one's hands are full. People also often lose their keys. A Lost and Found Survey conducted by Pixie discovered that on average Americans spend 2.5 days a year looking for misplaced items. When asked what items they misplace at least once a week, keys at 28% were among the most commonly lost items **[1]**. We aim to replace the traditional physical key with a "smart" lock that can be trained to identify a user's voice and unlock the door when the user says a specific keyphrase. This solution will save the user time and provide greater convenience when compared to using a physical key to unlock the door. The voice recognition door lock will consist of a control system, locking system, and a mobile application for the user enrollment phase. The control system will be responsible for sensing if someone is nearby using an ultrasonic sensor, listening to the voice and keyphrase to identify if the person is verified, and interfacing with the mobile application using a WiFi module. The locking system will consist of a servo motor and door lock, which should unlock the door only when a verified person says their keyphrase. The user will also be able to unlock the door using the physical key in case of a power outage. Finally, the mobile application will serve as the connection between the user and Microsoft Azure's Cognitive Services APIs. The mobile application will allow new users to create a profile and set up a keyphrase. When creating a new user, the application will use the Cognitive Services APIs to maintain a profile unique to the user's voice **[2]**.

1.2 Background

There are not many existing "smart" lock solutions that use voice recognition to open a door, but they are beginning to emerge in the market. These solutions interface with a voice assistant system (such as Google Home or Apple's Siri) that the user must purchase separately in order to use the voice recognition feature of the door lock **[3]**. We aim to create an all-in-one system that can be utilized with only the locking mechanism and a smartphone. The August Smart Lock, which can be controlled with Amazon Alexa, Google Home, or Apple's Siri Homekit is the most similar product on the market. This smart lock kit comes in multiple components and costs a total of $360, making it a convoluted and expensive solution. By incorporating everything into one system, we aim to keep costs low and make it easy to install so that anyone can afford to purchase our lock.

The Spring 2018 solution to this problem used facial recognition to unlock the door, but by using voice recognition, the lock will be functional even in low-light settings where a camera with facial recognition would not be able to operate. The user can also be positioned anywhere around the door instead of having to stand directly in front of a camera. Facial recognition technology also poses greater concerns to an individual's privacy. One of the main concerns is the lack of federal regulation around the use of facial recognition technology and how these databases of photos can be used or shared without permission **[4]**.

1.3 Visual Aid

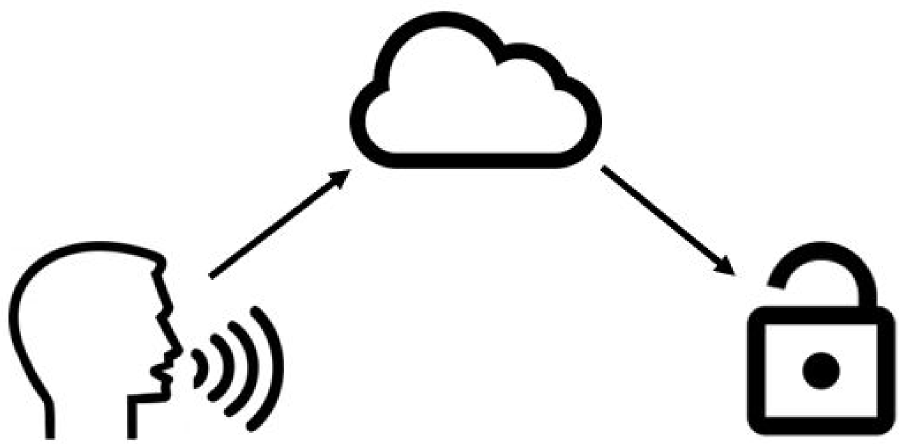

Figure 1. Visual Aid of the Voice Recognition Door Lock

1.4 High-level Requirements List

- Voice recognition algorithm must accurately identify the speaker 95% of the time.
- Must be able to identify if the user is verified and unlock the door in under 5 seconds after the user has spoken.
- All team members must be able to lock and unlock using their unique key phrases.

2) Design

2.1 Block Diagram

The system has four main components: a lock system that is responsible for locking and unlocking the door, a control system that is responsible for capturing the voice audio from a user, a Microsoft Azure server that will receive and analyze the voice message and return the appropriate response, and a mobile application that will serve as the interface and enrollment process for users. The locking mechanism will house the servo motor which will be connected to the deadbolt lock. There are two ways that the door can be unlocked: via voice control or a physical key. If the voice control method is used, the door will unlock upon receipt of an accepted voice message and return to a locked state after ten seconds of inactivity, which will be determined by the ultrasonic sensor. The microcontroller will trigger the servo motor to move the deadbolt. The other scenario allows for the user to manually lock/unlock the door with a key. In this scenario, the movement of the deadlock will move the servo motor as well. Since we will be using an analog feedback servo motor, we will be able to detect the active position of the servo motor to keep the movement consistent. If the deadbolt is in the unlocked state as measured by the servo motor analog read, the microcontroller will wait ten seconds for any activity and then lock the door. The third part of the system lies in the mobile application and server side application. Xamarin will be used to create the mobile application, as it is a Microsoft product and will allow for easy connectivity between the Cognitive Services APIs and the application creation process.

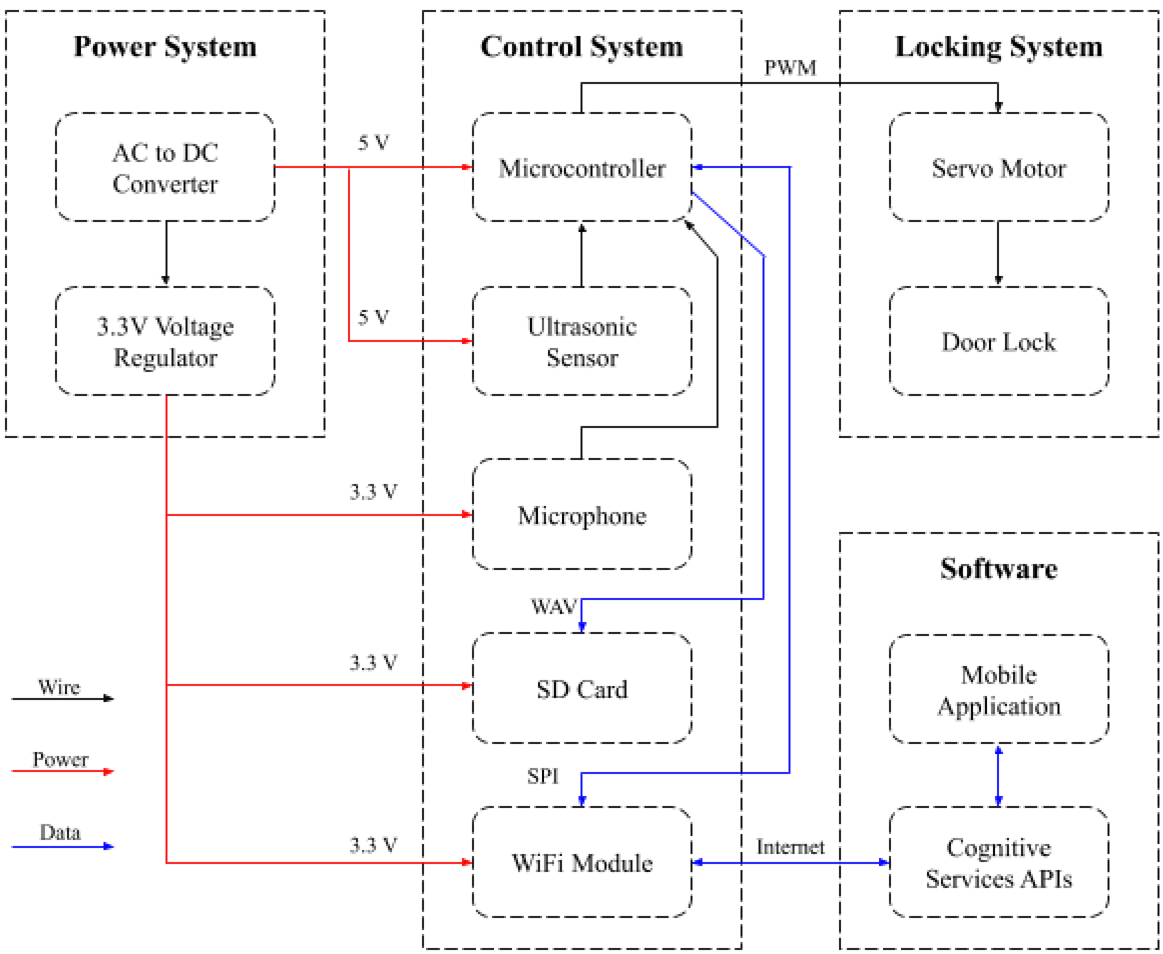

Figure 2. Block Diagram of the Voice Recognition Door Lock

2.2 Power System

2.2.1 AC/DC Converter

Power from a wall outlet and an AC/DC converter will be used to power the voice recognition door lock. A portable power solution is not needed because the door lock will never have to be moved and the wall outlet will provide a stable source of power. The design requires 5VDC to power the microcontroller and ultrasonic sensor. The PSA10F-050(P)-R AC/DC wall mount adapter takes an input voltage of 90-264VAC and outputs 5VDC. U.S. wall outlets draw 120-240VAC power so this adapter will suffice. The microcontroller will consume about 80mA of current, the servo motor will use about 150mA of current when it is active, while the WiFi module and ultrasonic sensor each consume about 30mA of current each. The microphone only consumes 0.5mA of current, so the total current requirement is a little short of 300mA.

Requirement	**Verification**
1. Outputs 5VDC +/- 5% provided an input of 120-240VAC; 2. Provides at least 300mA of current.	1. Connect the adapter to a DMM to ensure that the voltage is 5V +/- 5%; 2. Connect the adapter to a DMM to ensure that the current is at least 300mA.

2.2.2 Voltage Regulator

One voltage regulator is needed to step down the voltage from 5V to 3.3V to power the microphone and WiFi module. The TPS75233QPWPREP voltage regulator takes a maximum input voltage of 5V, steps the voltage down to 3.3V with a tolerance of +/- 2% and supplies an output current of 2A. Since the only components that will need 3.3V are the microphone and WiFi module, the voltage regulator needs to supply about 40mA, so the voltage regulator will have more than enough current.

Requirement	**Verification**
1. Provides 3.3V +/- 5% from a 5V source;	1. A. Connect a power supply to the voltage regulator and feed in 5V; B. Measure the output voltage using an oscilloscope to see if it stays within 3.3V +/- 5%;

2. Provides at least 40mA of current.	2. Connect the voltage regulator to a DMM to ensure that the current is at least 40mA.

2.3 Control System

2.3.1 Microcontroller

The microcontroller will be connected to the ultrasonic sensor, microphone, WiFi module and SD card. The ATmega328P will read data from the ultrasonic sensor to determine if someone is in front of the door in order to begin listening for the keyphrase. The microcontroller will take audio from the microphone and convert it to WAV format and store it on the SD card. After the recording is saved, it will be uploaded to the Microsoft Cognitive Services API, which will verify the identity and keyphrase and return a boolean response. If the person is verified, the microcontroller will send a PWM signal to the servo motor to unlock the door and will light an LED green. Otherwise, the light will flash red indicating the keyphrase has been rejected. The system will continue to listen until the ultrasonic sensor detects no change in setting for ten seconds, after which, the servo motor will be set to the locked position and the system will sleep.

Requirement	**Verification**
1. Must be able to accurately read input from ultrasonic sensor;	1. A. Connect the ultrasonic sensor to the ATmega328P; B. Stand in front of the sensor and check the input from the microcontroller;
2. Must be able to communicate with the WiFi module over SPI protocol;	2. A. Connect the WiFi module to the ATmega328P; B. Use the Arduino serial monitor to send and verify correct transmission of data between modules;
3. Must be able to convert raw analog input from microphone into WAV format;	3. A. Receive input from microphone; B. Run our WAV file generation code on the analog input; C. Play back the generated WAV file on a PC and listen for any distortions or analyze visually;

4. Must be able to store at least thirty seconds of audio on SD card in WAV format.	4. A. Generate thirty second audio clip; B. Write to SD card; C. Read from SD card and compare differences.

2.3.2 Ultrasonic Sensor

An ultrasonic sensor on the door lock will be used to detect if a person is nearby in order to begin listening for the keyphrase. It should be able to detect objects within one meter of the locking system and communicate with the microcontroller when an object is detected. An ultrasonic sensor relies on sound waves and its accuracy is not as affected by surrounding conditions compared to an infrared sensor **[5]**. An infrared sensor would not be as accurate in dark lighting conditions and the product has to be reliable and secure during all times of day. The HC-SR04 ultrasonic distance sensor has a ranging distance of 2cm to 4m, which is more than sufficient for the project.

Requirement	**Verification**
1. Must be able to detect a person up to one meter away.	1. Walk back and forth one meter in front of the ultrasonic sensor and actively monitor the output being sent to the microcontroller to confirm detection.

2.3.3 WiFi Module

The WiFi module will be connected to the microcontroller and will be used to communicate with Microsoft's API. ESP8266 WiFi Module will send an audio file of WAV format to the API and receive a binary signal from the API indicating if the door should be unlocked or stay locked. The WAV file will be at most 960KB given a 30 second duration, 16-bit depth, and 16kHz sample rate. This WiFi module should be capable of a fairly large bandwidth of 54Mbps and the 802.11 b/g/n WiFi modules are capable of streaming between 1-6.75MB/s, which is more than enough for an audio application **[6]**.

Requirement	**Verification**
1. Must be able to send 960KB WAV files to the API;	1. A. Generate a WAV file of 30 seconds; B. Send the WAV file to the API and check if the program is running to

2. After the WAV file is sent, it must receive a binary signal from the API.	verify; 2. Monitor the Arduino serial port to see if a binary response has been received from Microsoft's servers.

2.3.4 Microphone

The Electret Microphone Breakout will have to accurately record the user when they speak so that their identity and keyphrase can be verified by the mobile application. There will be ambient noise as the door lock will be outside, so the microphone has to be of high quality, so the recording is clear. Since the human voice ranges from 50-70dB when talking at distances of 0-2 meters **[7]**, and outdoor noise levels vary from 30-60dB depending on the location **[8]**, the microphone must have good frequency response for frequencies between 100Hz and 3kHz at levels of 40-70dB so that we can discern human speech from background outdoor noise.

Requirement	Verification
1. Must have an acceptable frequency response for 100Hz to 3kHz;	1. A. Connect the microphone to the 3.3V reference voltage and a DMM; B. Play a sonic sweep from 100Hz to 3kHz 1 meter away from the microphone and actively monitor the DMM to make sure the voltage is fairly consistent;
2. Must have a sensitivity of at least 3.3V/Pa.	2. A. Connect the microphone to the 3.3V reference voltage and a DMM; B. Play sounds of varying intensity from 0.1Pa to 1Pa and ensure that there is no clipping at low intensities and high intensities of sound.

2.3.5 SD Card

The SD card will store the generated WAV file. The microcontroller only has 32KB of flash memory so an SD card is essential to the functionality of the product. The analog data captured from the user will be converted in real time by the WAV file generation program and stored on the SD card. The newly generated WAV file will be uploaded to Microsoft's servers for analysis. A WAV file with a 30 second duration, 16-bit depth, and 16kHz sample rate will require 960KB of space, so the 8GB SanDisk MicroSD card should have more than enough storage space. Assuming we hold five 6-second clips on the SD card at a given time, it will need to have capacity for at least 30 seconds of audio data. This is an overestimate for how much capacity we actually need, but will allow for future functionality improvements.

Requirement	**Verification**
1. The SD card must be able to store 960KB of audio data when converted to WAV format.	1. A. Convert a 30 second voice clip to WAV format; B. Write the WAV file to the SD card; C. Play the WAV file from the SD card to verify that it is complete and uncorrupted.

2.3.6 Circuit Schematic

Below is the circuit schematic for the control system of the voice recognition door lock. Some components use 5V power (microcontroller and ultrasonic sensor), while all the other components use 3.3V power, which is stepped down from 5V with a voltage regulator. The SD card adapter, WiFi module, and servo motor are connected to the ATMega328P microcontroller chip as outputs, while the microphone break-out and ultrasonic sensor are connected as inputs. The microcontroller is vital to the project as it will be acting as an information bus and controlling/routing all of the signals within the control system.

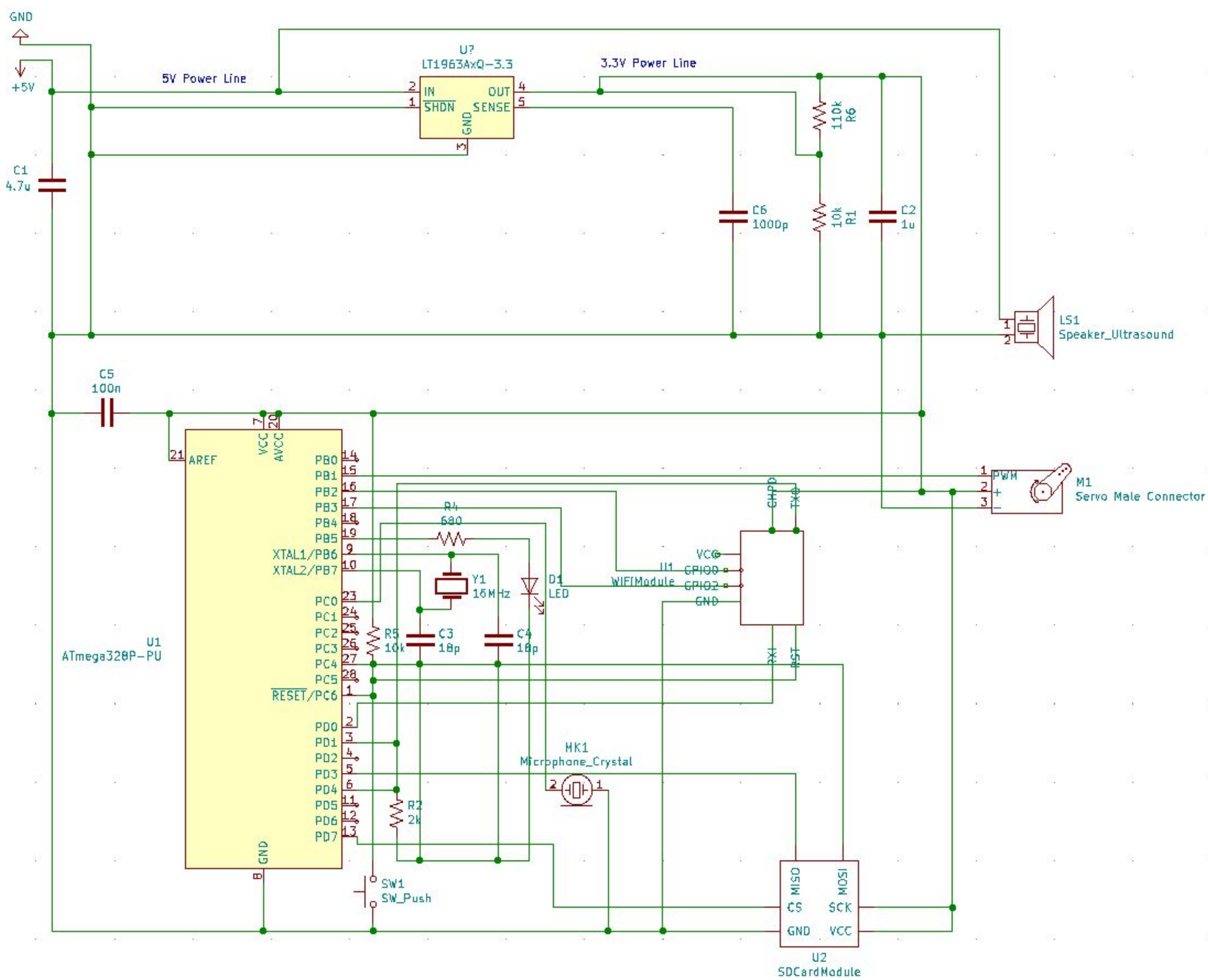

Figure 3. Circuit Schematic of the Control System

2.3.7 State Diagram

This state diagram displays an overview of the control system functionality. The lock state functions as the dormant state of the system. In this state, the ultrasonic sensor will monitor the area around the door and trigger the activation of the system when it detects motion at a distance of 1m away. Once motion is detected, the system will move to the listen and receive state where it will begin to capture audio, send a WAV file through WiFi and wait to receive a positive signal. If a positive signal is not received, the state will continue to listen and receive until the ultrasonic sensor has detected ten seconds of no motion change. If a positive signal is received, the door will unlock and wait ten seconds before returning to the locked state.

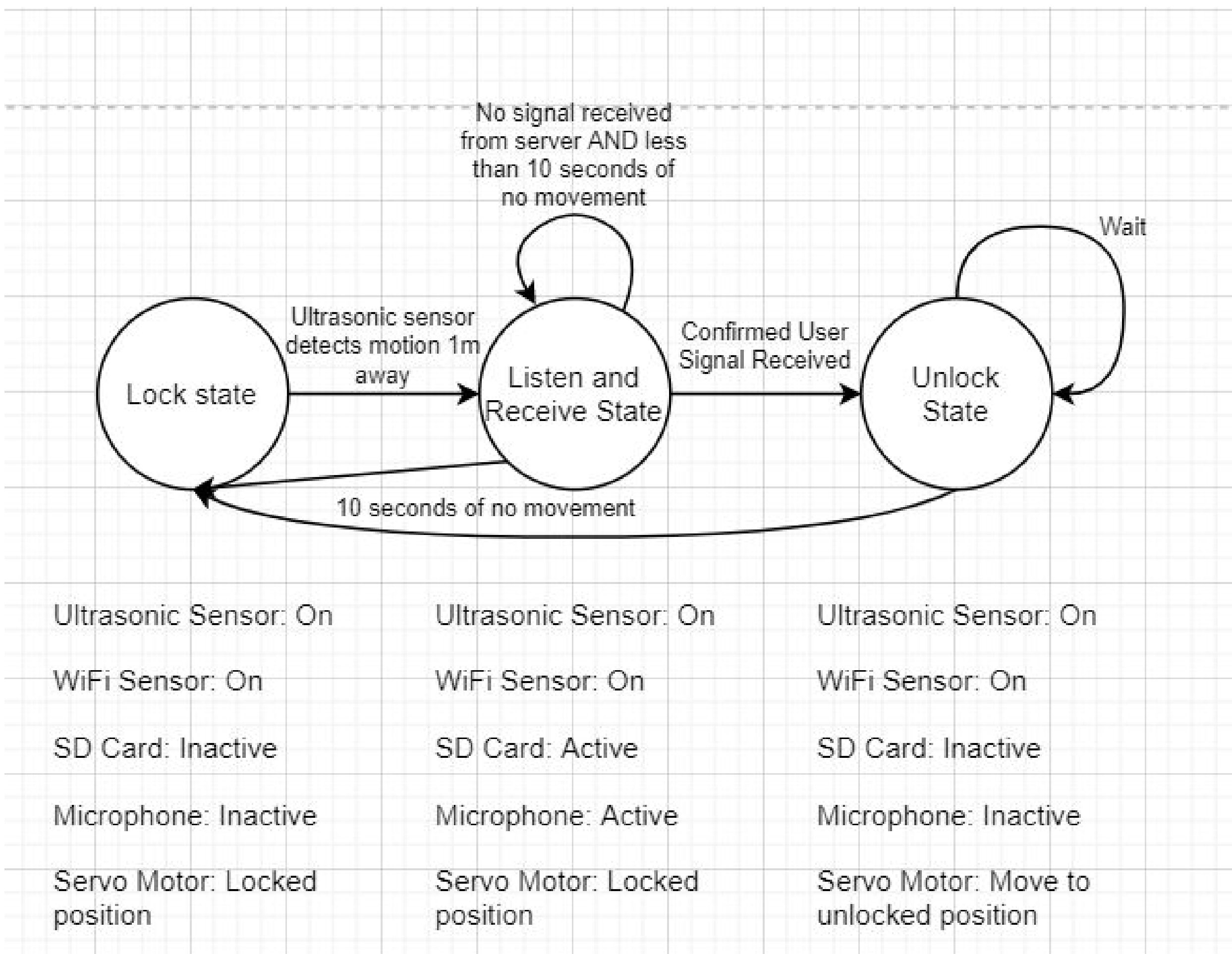

Figure 4. State Diagram of the Control System

2.4 Locking System

2.4.1 Servo Motor

The servo motor will be used to unlock and lock the door. The servo will receive a PWM signal from the microcontroller and rotate the correct angle to unlock or lock the door. The servo will power off when not in use so the user can unlock or lock the door with minimal resistance using a key. An Analog Feedback Servo will be used to allow the position of the servo to be read. This feature is important for our implementation of the door lock because the case where the user locks or unlocks the door manually has to be handled. If no action triggers the ultrasonic sensor for ten seconds, the servo motor will be powered on and set to the lock PWM signal. The servo must also have enough torque to turn the deadbolt in a door lock.

Requirement	**Verification**
1. Must be capable of a 90° rotation within 300ms of receiving a PWM signal;	1. A. Use a signal generator to generate PWM signals of increasing duty cycle; B. Record the corresponding revolution angles;
2. The microcontroller must be able to read the servo position;	2. A. Cut power to the servo motor; B. Manually move the servo motor 30°; C. Turn the power back on and check that the position read is correct.
3. Servo must have at least 0.5 Nm of torque.	3. A. Attach a force meter to the end of the servo and send a maximum PWM signal; B. Check the force meter to ensure that the servo is capable of producing a minimum of 0.5 Nm of force when activated.

2.4.2 Door Lock

The door lock will be incorporated with a servo motor so the user can still lock or unlock the door with a physical key. If power is lost in the building, the voice recognition door lock will be unusable, so users must still be able to enter their home. The lock must have a keyhole as an entry alternative, but the deadbolt should be able to move with the servo or the key. The

AmazonBasics Exterior Door Knob With Lock and Deadbolt meets our criteria and it should be easy to assemble the servo motor to it.

Requirement	**Verification**
1. Must be able to lock and unlock with a traditional key; 2. Must be able to configure the door lock with a servo.	1. Use the key to check if the door can be locked and unlocked; 2. Assemble the servo to the door lock and send PWM signals to check if the servo can unlock and lock the door.

2.5 Software

2.5.1 Mobile Application

The mobile application will interface with Microsoft Azure's Cognitive Services APIs to achieve the enrollment phase of the system. The Speaker Recognition APIs will be used to assemble and analyze the voices of users. Initially, a user will login to the application with some username/password and add a recording of themselves saying a specific keyphrase, which will be used to verify the speaker for unlocking the door. We will specify that the keyphrase be a maximum of three seconds in duration. Microsoft Azure's Speaker Verification API will be used to create profiles and store unique voice signatures from the keyphrases. This information will be stored securely on Microsoft's servers and the user controls how long the data should be retained. The mobile application will be able to create, update, and remove enrollment data for individual speakers through API calls. The mobile application will be built using Xamarin as it is owned by Microsoft and can be easily incorporated with the Cognitive Services APIs.

Requirement	**Verification**
1. Must be capable of recording and sending a keyphrase to Microsoft's API;	1. A. Record a keyphrase using the app and hit the submit button; B. Access Microsoft's API and verify that a voice signature has been extracted from the keyphrase.

2.5.2 Cognitive Services APIs

The Speaker Verification API will run on Microsoft's servers once pinged by the WiFi module. The API extracts voice features and the keyphrase from the given voice recording and compares these features with elements from the enrollment data for the speaker. Once the WAV file has been sent from the SD card to Microsoft's API, the main function returns three components on execution: result, confidence, and phrase. A user will be considered verified and the door will unlock if the result is accepted (voice matched to the user), and the confidence level of verification is high.

Requirement	**Verification**
1. Must correctly respond after analyzing the received message with at least 95% accuracy.	1. Check that a user can unlock the door if and only if they are a valid user and they say their keyphrase. Run multiple simulations to compare accuracy.

For the free service, 20 pings can be sent every minute, so for testing, we will be sending voice signals to the API in the following manner:

1. (t=0): Ultrasonic sensor is triggered and microphone starts listening.
2. (t=3): Three second clip is sent (0-3s).
3. (t=6): Six second clip is sent (0-6s).
4. (t=9): Six second clip is sent (3-9s).

…

After the first three second clip is sent, six second clips will be sent to the server every three seconds. Since the keyphrase will be under three seconds (mandated in the app design), this method will assure that the keyphrase is captured even if the user does not speak immediately after walking up to the door. This process will continue until the ultrasonic sensor detects no change in motion for ten seconds. By using a constant stream of voice samples, we can make our verification as close to real-time as possible, and give the Microsoft Cognitive Services API more time to process and respond, while still maintaining the three second minimum length requirement for verification.

2.6 Tolerance Analysis

While all features of the project are important to the success of our product, the speed of operation is the most crucial. This depends on a lot of different variables, such as the strength and speed of our WiFi module, the speed of the user's WiFi, the SD card read/write times, and the Microsoft Cognitive Services API response time. Since our high-level requirements state that we must be able to unlock the door in under five seconds after the user has spoken their keyphrase. We must be able to send our audio file, analyze it using the voice recognition API, and receive a boolean signal in under five seconds. This could be impossible if our WiFi module is slow or if the WiFi signal is not strong enough.

Since our voice signal will be first stored on the SD card, it will have to be read by the microcontroller, streamed to the WiFi module, and then sent to the home WiFi router, from where it will be sent to Microsoft's server for analysis. The size of the WAV files will start at three seconds (96KB) for the first ping to the server, but then increase to six seconds (192KB) as explained in the previous section **[9]**.

Reading a 96KB to 192KB audio file from the 8GB SanDisk MicroSD card should take between .001 seconds to .002 seconds, since the SD card we chose has a read speed of 98 MB/s.

$$Read\ Time\ for\ 96KB\ File = \frac{0.096MB}{98MB/s} = 0.00097959\ seconds \approx 0.001\ seconds$$

$$Read\ Time\ for\ 192KB\ File = \frac{0.192MB}{98MB/s} = 0.0019592\ seconds \approx 0.002\ seconds$$

The WiFi module communicates with the microcontroller using SPI protocol at the rate of the microcontroller's 16MHz clock frequency, which means it will take 48 ms to send a 96KB signal at 16MHz and 96 ms to send a 192KB signal at 16MHz. The WiFi module has a maximum bandwidth of 6.75MB/s, which should be more than enough to transmit a WAV file. It also takes <2ms to wake up and starts sending packets every 0.4ms to the router, where each packet is 2304 bytes **[6]**.

Thus, the maximum time it would take to transmit from the WiFi module to the router is 62ms for the six second files. This means our overall time from reading the audio file to sending it out to the server is between:

$$.001\ seconds + 0.048\ seconds + 0.002\ seconds + .0167\ seconds = 0.0677\ seconds\ (best\ case)$$

$$.002\ seconds + 0.096\ seconds + 0.002\ seconds + .0333\ seconds = 0.1333\ seconds\ (worst\ case)$$

This range is sufficient and should allow the product to be successful and have a total operation time under five seconds, which will be discussed in our project differences analysis.

3) Project Differences

3.1 Overview

The Spring 2018 solution to the door lock problem went with a facial recognition approach. The design involves a facial recognition system, lock system, and PCB. The ultrasonic sensor detects if a user is standing in front of the door to activate the microphone for listening. If a human voice command is heard, the camera begins taking pictures of the user, so the facial recognition system can make a decision on unlocking the door. The facial recognition system runs on a PYNQ FPGA board and includes a convolutional neural network algorithm to test the images and determine if the identification was a success or failure. If the identification is a success, the microcontroller sends the unlock signal to the motor to unlock the door.

The fundamental difference between the Spring 2018 solution and our solution is the use of facial recognition rather than voice recognition to determine if the door should be unlocked. One of the main advantages voice recognition has over facial recognition is in usability. Voice recognition allows the user to stand anywhere near the microphone and just say their keyphrase. In the case of facial recognition, the user has to stand directly in front of the camera, and in a very specific spot. The user cannot have their face tilted to the side or stand a variable distance from the camera, while the voice recognition unlocking method allows the user to be standing anywhere within a certain radius of the door lock, making our lock easier to use. Another problem with the camera arises in dark lighting conditions, where it is difficult to see facial features, which results in low quality images and the facial recognition system producing an inaccurate result or in the worst case making the system completely unusable in complete darkness. Our solution is also significantly cheaper (about five times) to manufacture, since we do not utilize an FPGA, but our costs would increase slightly with a large user base due to the usage of a third-party API.

While our solution offers simpler, cheaper, and more comfortable usage, it is also faster than the previous solution in their worst case scenario and only a fraction of a second slower in their best case scenario. On average, the voice recognition door lock performs slightly faster than the original facial recognition door lock. Calculations for the comparison of operation times are discussed below in the analysis section.

3.2 Analysis

The problem with traditional door locks is the inconvenience of using a physical key when one's hands are full or the inability to even find the key in the first place. Our voice recognition door lock provides a fast and convenient way to unlock a door. Our design has operational improvements over the original design in that it can function in all lighting conditions and will provide a marginal speed improvement on average. The Spring 2018 solution estimated an operation time of 3.78 ± 0.45 seconds. We will now break down our solution to find its approximate operation time:

The user will speak their keyphrase into the microphone in three seconds. The storage space for a three second mono audio signal with 16-bit depth and 16kHz sample rate is:

$$Audio\ File\ Size\ =\ \frac{16 * 16{,}000 * 1 * 3}{8} = 96KB$$

The WAV file generation program will read from the analog pin and convert the audio signal to a WAV file in real time, so this will take zero seconds in processing time.

In the tolerance analysis, we discuss the calculations for the time it takes for the microcontroller to read the WAV file from the SD Card, transfer it to the WiFi Module, the WiFi Module to send it to the home router, and from there stream the WAV file over the internet to Microsoft Azure's servers. This time was estimated to be between 0.0677 to 0.1333 seconds.

As the Speaker Recognition API within Cognitive Services is rather new, Microsoft does not provide metrics in their Resources page as they have provided for their other services within Cognitive Services. In order to approximate the response time for the Speaker Recognition API we looked at the Translator Text API and found that the typical response time was on average 225ms for text within 100 characters **[11]**, so we will assume a similar response time for our estimate.

We are unable to measure the exact time it would take the servo to unlock the door, but the Analog Feedback Servo has a speed of 0.21 sec/60°, so we approximate it will take 0.315 seconds to unlock the door given a 90° angle of rotation.

Given the operational analysis of our solution above, the approximate operation time for this voice recognition door lock design is:

$$3 + 0 + (0.0677 + 0.1333)/2 + 0.225 + 0.315 = 3.6405 \pm 0.1005\ seconds$$

This operation time meets our high-level requirements of being under five seconds and should be sufficient for the user to comfortably open the door in about the same time they would be able to if they were using a physical key.

4) Cost and Schedule

4.1 Cost Analysis

4.1.1 Labor

We estimate the cost of labor for three people will be around $37,500 assuming a reasonable salary of $50/hour, while working 10 hours/week for 10 weeks.

$$3 \cdot \frac{\$50}{hour} \cdot \frac{10\ hours}{week} \cdot 10\ weeks \cdot 2.5 \ = \ \$37,500$$

4.1.2 Parts

Description	**Manufacturer**	**Part #**	**Quantity**	**Cost**
ATMega328P Microcontroller Chip	Microchip Technology	ATMEGA328P-PU	1	$2.08
PSA10F-050(P)-R AC/DC Wall Mount Adapter	Phihong USA	PSA10F-050(P)-R	1	$12.56
TPS75233QPWPREP Voltage Regulator	Texas Instruments	TPS75233QPWPREP	1	$10.49
HC-SR04 Ultrasonic Distance Sensor	SparkFun	HC - SR04	1	$3.95
ESP8266 WiFi Module	SparkFun	WRL-13678	1	$6.99
Electret Microphone Breakout	SparkFun	CEM-C9745JAD462P2.54R	1	$6.95
Micro SD TF Card Adapter Module	Electropeak	DAT-01-003	1	$1.99
16GB MicroSD Card	SanDisk	SDSQUAR-016G-GN6MA	1	$5.99
Analog Feedback Servo	Adafruit	1404	1	$14.95
Exterior Door Knob	AmazonBasics	B07GF58FQ3	1	$22.99

With Lock and Deadbolt				
Total				$68.81

4.1.3 Sum of Costs

Sum of Costs = Labor Cost + Parts Cost = $37,500 + $68.1 = $37,568.81

4.2 Schedule

Week	**Rahul**	**Vaidas**	**Rohil**
1	Design Review Order Parts	Design Review Order Parts	Design Review Order Parts
2	Design and Order PCB Place Machine Shop Order	Design and Order PCB Place Machine Shop Order	Design and Order PCB Place Machine Shop Order
3	Assemble Locking System	Solder PCB	Assemble Locking System
4	Program Microcontroller	Configure and Test Sensors	Develop Mobile App
5	Test Generation of WAV File and SD Card Transfer	Configure and Optimize WiFi Module Performance	Test Communication with Cognitive Services API
6	Testing and Debugging	Testing and Debugging	Testing and Debugging
7	Work on Presentation and Practice Demo	Work on Presentation and Practice Demo	Work on Presentation and Practice Demo
8	Mock Demo	Mock Demo	Mock Demo
9	Final Demo	Final Demo	Final Demo
10	Presentation Final Paper	Presentation Final Paper	Presentation Final Paper

5) Discussion of Ethics and Safety

While we do not anticipate many safety hazards while working with this project, we will be working with 120VAC when installing the system, so we must take extra precautions such as insulating gloves and proper lock-out protocols when building and installing our system. We must ensure that the system can be safely installed by every user that purchases our product. We are also using servos to power the locking mechanism, and the servos can be a safety hazard to the user and cause damage to property if they do not function properly, so we must ensure that the servos rotate the exact amount that is needed and do not overheat during operation. Since we will be utilizing the servo very intermittently, this should not be a large issue, but we will actively monitor the temperature of components to ensure that the operating temperatures do not get too high.

Ethics is a much larger issue with our project, because we will be handling personal data and will need to store it somewhere to verify user identity. We plan on using the user's home WiFi to send the voice recordings to Microsoft's servers in order to minimize the threat to privacy. By having a reputable company such as Microsoft with years of experience handling personal data, we can bring greater trust to our product. Microsoft states that users control how long the information is retained, so our mobile application will have a feature where users can remove all their data. We will also make sure we receive the appropriate permissions from users who buy the product.

The specific issues that we will face from the IEEE Code of Ethics are:

5. to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems;

9. to avoid injuring others, their property, reputation, or employment by false or malicious action **[12]**;

We are abiding by the first rule by simply creating this system because we are creating a new intelligent system that will help user's gain functionality and save time in their everyday lives. The second rule concerns safety, which we have discussed above.

The specific issues that we will face from the ACM Code of Ethics are:

1.6 Respect privacy.

1.7 Honor confidentiality.

2.9 Design and implement systems that are robustly and usably secure.

3.3 Manage personnel and resources to enhance the quality of working life.

3.7 Recognize and take special care of systems that become integrated into the infrastructure of society **[13]**.

We plan to tackle these issues by not storing any of the user's private and confidential data on our own systems, and instead leaving the data handling to a reputable company such as Microsoft as mentioned earlier. We plan to maintain security by using WiFi, which is encrypted and give the user more control over their own data as they also have the SD card, which the WAV files will be temporarily stored on. The second two issues once again touch on safety and improvement of everyday life, which we have already discussed above.

Overall, if we stick to the design choices that we have made and keep these ethical issues in mind, we believe that our product will not pose any ethical or safety threats to the user. By following these codes of ethics, we will ensure that our product is suitable, safe, and ethical.

6) Citations

[1] “Lost and Found: The Average American Spends 2.5 Days Each Year Looking For Lost Items Collectively Costing U.S. Households $2.7 Billion Annually in Replacement Costs .” *PR Newswire*, Cision, 2 May 2017, www.prnewswire.com/news-releases/lost-and-found-the-average-american-spends-25-days-each-year-looking-for-lost-items-collectively-costing-us-households-27-billion-annually-in-replacement-costs-300449305.html.

[2] “Speaker Recognition API: Microsoft Azure.” *Speaker Recognition API | Microsoft Azure*, azure.microsoft.com/en-us/services/cognitive-services/speaker-recognition/.

[3] Wollerton, Megan. “Is It Safe to Control a Smart Lock with Your Voice?” *CNET*, CNET, 30 Mar. 2017, www.cnet.com/news/controlling-locks-with-your-voice-good-idea-or-bad-idea/.

[4] Martin, Nicole. “The Major Concerns Around Facial Recognition Technology.” *Forbes*, 25 Sept. 2019, www.forbes.com/sites/nicolemartin1/2019/09/25/the-major-concerns-around-facial-recognition-technology.

[5] Burnett, Roderick. “Ultrasonic vs Infrared (IR) Sensors – Which Is Better?” *MaxBotix*, 27 Nov. 2017, www.maxbotix.com/articles/ultrasonic-or-infrared-sensors.htm.

[6] “IEEE 802.11TM WIRELESS LOCAL AREA NETWORKS.” *IEEE 802.11, The Working Group Setting the Standards for Wireless LANs*, www.ieee802.org/11/.

[7] “Voice Level at Distance.” *Engineering ToolBox*, www.engineeringtoolbox.com/voice-level-d_938.html.

[8] “Outdoor Ambient Sound Levels.” *Engineering ToolBox*, www.engineeringtoolbox.com/outdoor-noise-d_62.html.

[9] Herman, Michael, et al. “Audio File Size Calculator.” *Colin Crawley*, www.colincrawley.com/audio-file-size-calculator/.

[10] “Serial Communication.” *SparkFun*, SparkFun Electronics, learn.sparkfun.com/tutorials/serial-communication/all.

[11] “Request Limits for Translator Text.” *Microsoft Azure*, Microsoft, docs.microsoft.com/en-us/azure/cognitive-services/translator/request-limits.

[12] “IEEE Code of Ethics.” *IEEE Code of Ethics*, IEEE, www.ieee.org/about/corporate/governance/p7-8.html.

[13] “ACM Code of Ethics and Professional Conduct.” *Code of Ethics*, ACM, www.acm.org/code-of-ethics.