Dwight Look College of ENGINEERING
TEXAS A&M UNIVERSITY

Team 28:
Smart Caller ID for Landlines
Bi-Weekly Update 1

Amy Chen
Scott Kevil-Yeager
Matthew Hebrado

Sponsor: Dr. Tod Cox

TA: Dalton Cyr and Rohith Kumar

Project Summary

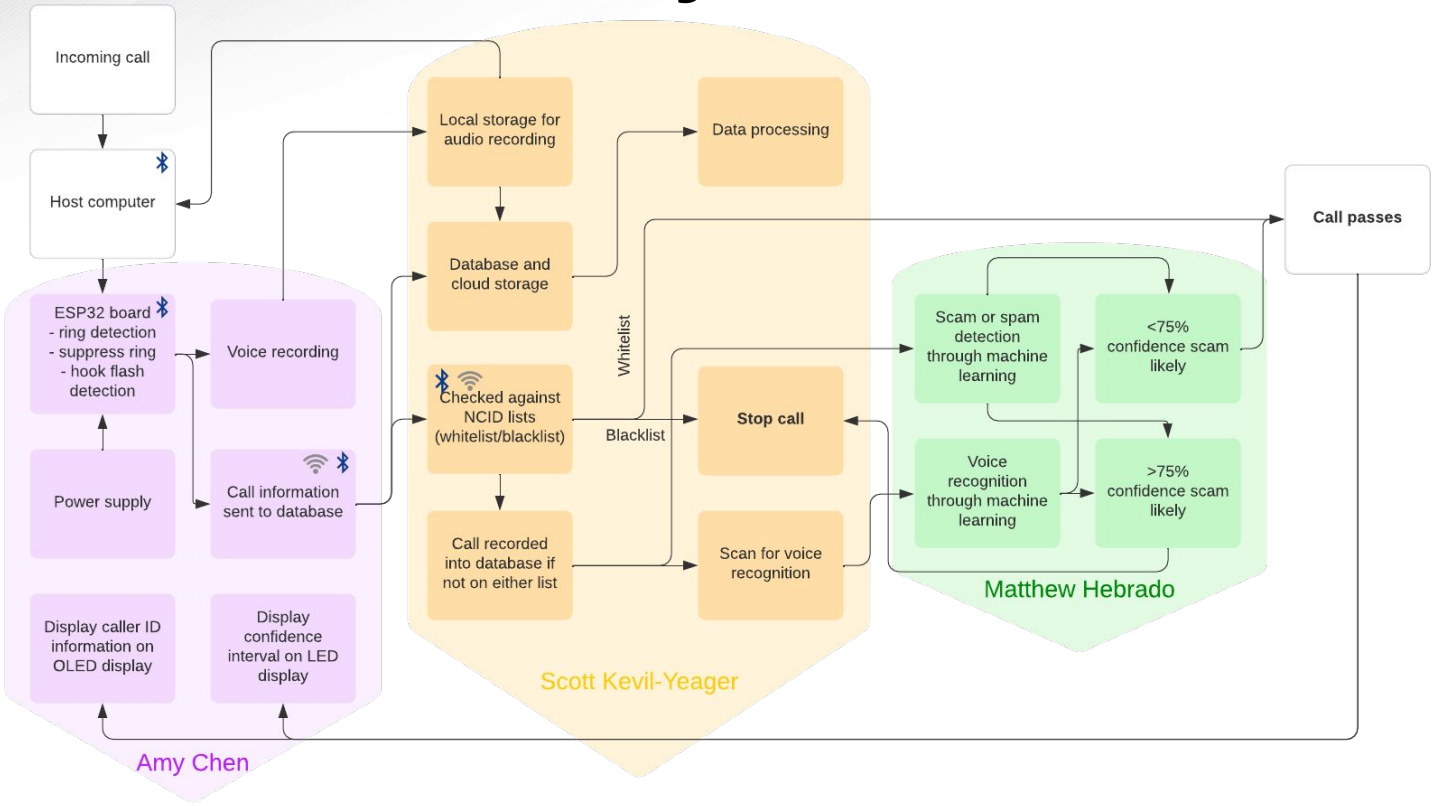
The Problem

- Many older residents in the U.S. still rely on a landline for telephone communications and many times, they are scammed out of money. This process can go on for months before someone notices.

The Solution

- The goal of our project is to create a device with a secure caller ID/scam protection with a call screening process using NCID's built in blacklist and whitelist functionality.
- Our project will also provide vocal signature matching to ensure that when family members or friends call they are recognized regardless of what number they call from.

Subsystems


Incoming call
Host computer
ESP32 board
- ring detection
- suppress ring
- hook flash
detection
Voice recording
Power supply
Call information
sent to database
Display caller ID
information on
OLED display
Display
confidence
interval on LED
display
Amy Chen
Local storage for
audio recording
Data processing
Database and
cloud storage
Checked against
NCID lists
(whitelist/blacklist)
Whitelist
Blacklist
Stop call
Call recorded
into database if
not on either list
Scan for voice
recognition
Scott Kevil-Yeager
Call passes
Scam or spam
detection
through machine
learning
<75%
confidence scam
likely
Voice
recognition
through machine
learning
>75%
confidence scam
likely
Matthew Hebrado

Major Project Changes

- Redistribution of subsystem parts between Amy and Scott
- The voice mapping needs to be improved to account for untrained/unknown voices that will be passed through the system

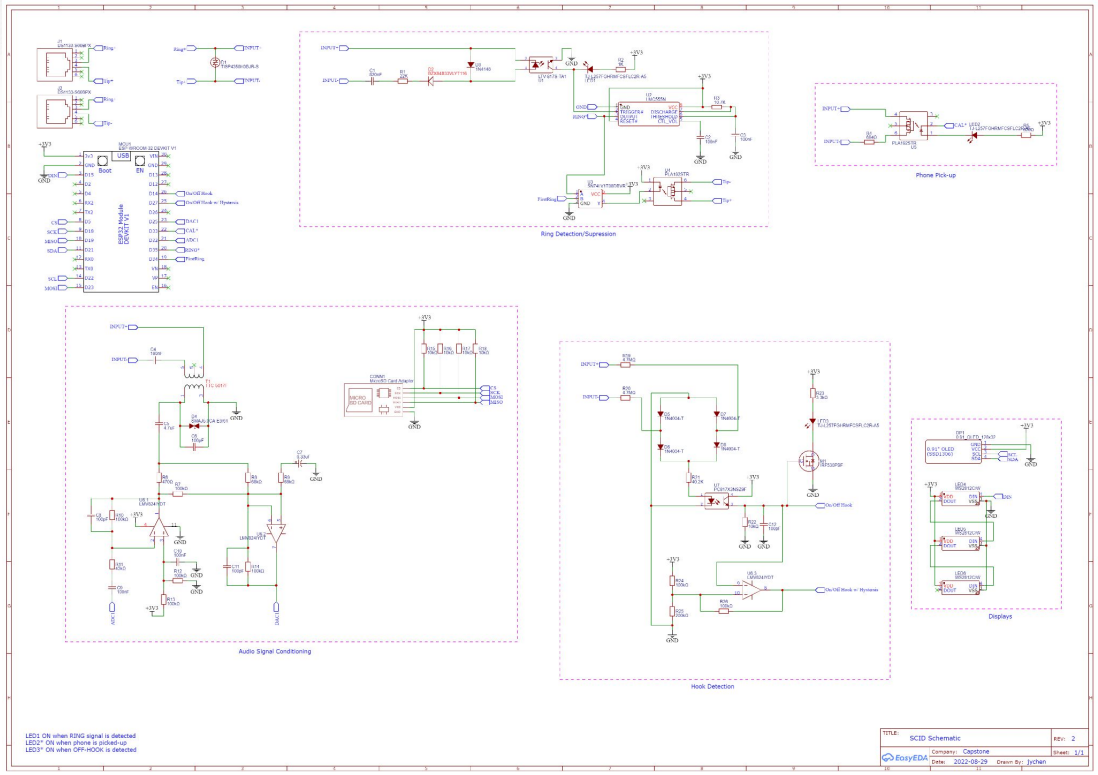
Project Timeline

	8/29	9/5	9/12	9/19	9/26	10/3	10/10	10/17	10/24	10/31	11/7	11/14	11/21	11/28	Team Member	
Verify Hardware Subsystems															Amy	Key
Verify Database Subsystems															Scott	Ongoing
Verify ML Subsystems															Matthew	Complete
PCB assembly															Team	Planned
Database and ML Subsystem Integration															Team	Behind
Hardware and Database Subsystem Integration															Team	
Debugging and Validation																

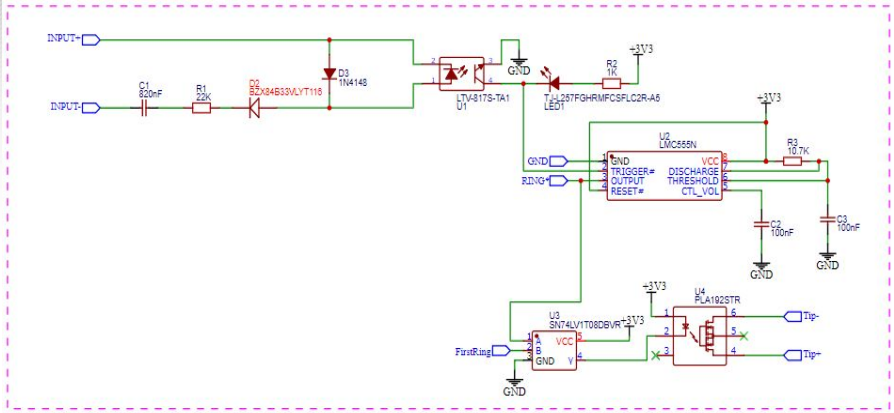
Hardware Subsystem

Accomplishments since 403 40 hrs of effort	Ongoing progress/problems and plans until next presentation
• Design PCB • Ordered PCB	• Test code for displays again on ESP32 • Outline C code for state machine on ESP32 • Begin testing PCB

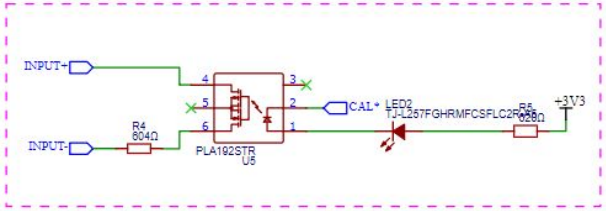
Hardware Subsystem



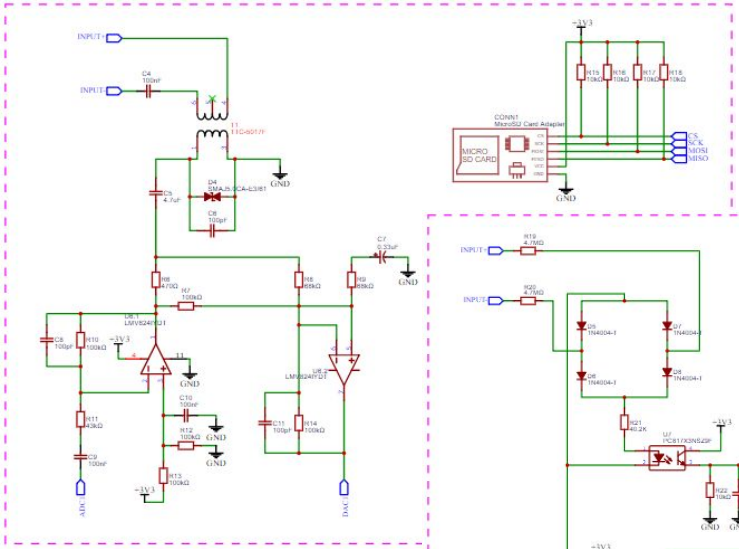
Hardware Subsystem



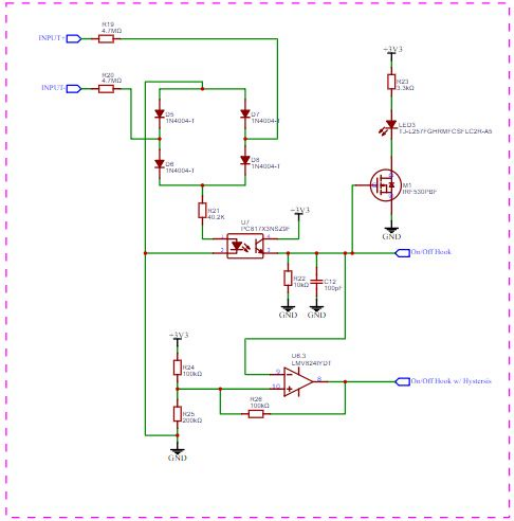
Ring Detection/Supression



Phone Pick-up

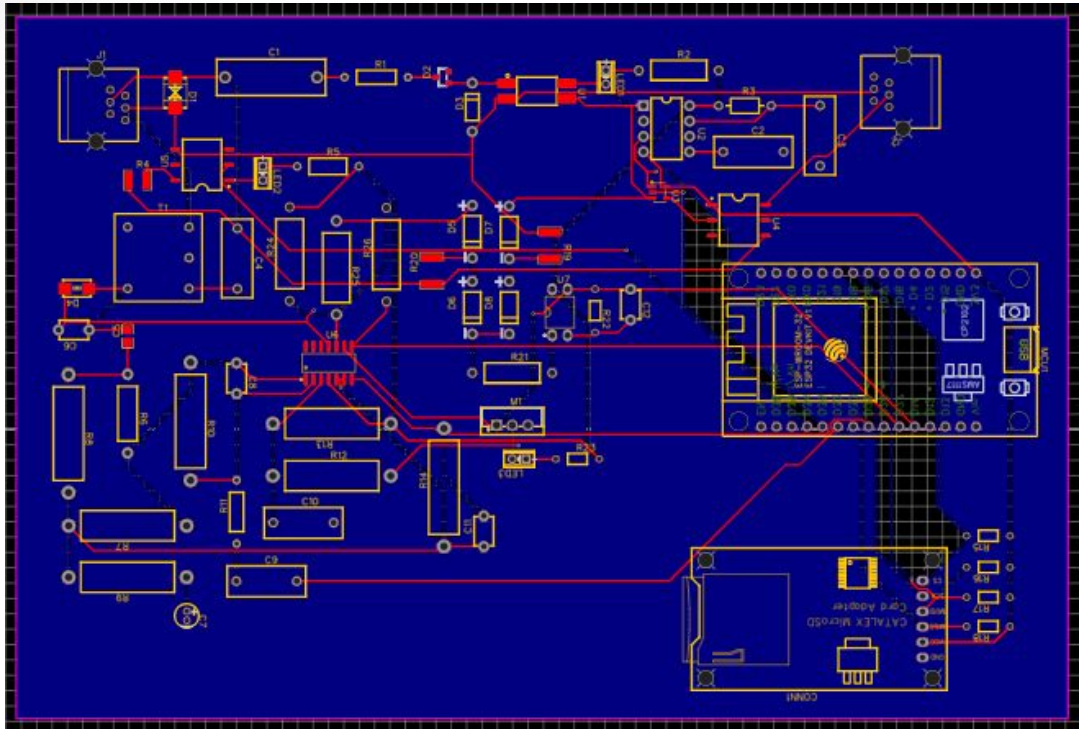


Audio Signal Conditioning



Hook Detection

Hardware Subsystem



Database and Data Processing

Accomplishments since 403	Ongoing progress/problems and plans until next presentation
• Implemented audio recording subsystem that will be integrated into the hardware and database subsystems	• Wireless transmission of audio recordings by ESP32 to local storage and microSD card to simulate how host computer may receive data • Planning integration of subsystem with hardware and machine learning subsystems (including state machine of possible hardware inputs into ESP32)

Database and Data Processing

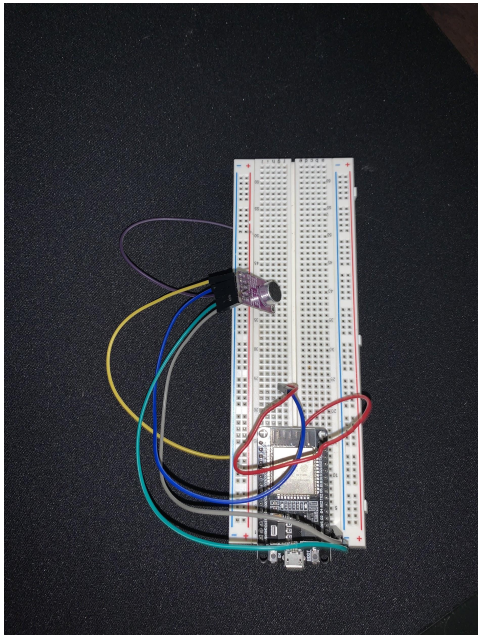
User interface:

```
 ****************** LOGIN ******************
What action would you like to take:
list recent [c]alls
[u]pload
e[X]it app
[?] Help (this info)
```

```
What action would you like to take:
upload [f]older
upload f[i]le
[r]eturn to previous page
e[x]it app
[?] Help (this info)

> f
Input the path to the folder (using / to separate files):
```

ESP32 bread boarded microphone:



Database and Data Processing

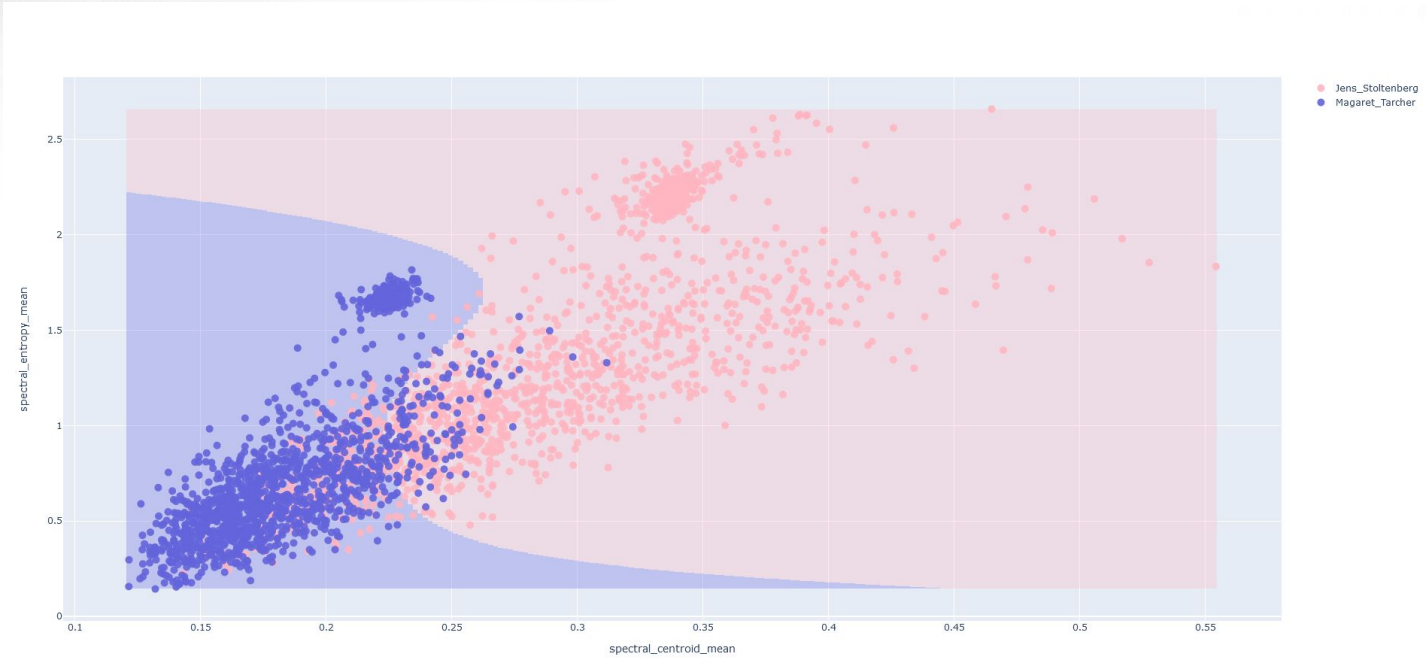
Future plans:

- Integration of database with ML algorithm; incoming and outgoing call voice prints stored to document variable to keep records of incoming and outgoing calls
- Integration with hardware audio recording through the ESP32 microcontroller allowing direct recording from handset, as well as a state machine for every hand set state (on-hook, off-hook, call waiting, etc.)
- Allow users to set a file directory that will hold recording files that will be automatically uploaded to the database, improve naming scheme of recordings to ensure easy look up and uploading

Machine Learning

Accomplishments since 403 33 hrs of effort	Ongoing progress/problems and plans until next presentation
● Further testing on voice mapping ○ Upon testing unclassified voice samples the program is falsely matching them with known voices	● Find a way to determine whether the voice sample is a known voice or not ● Test and run code off of a Raspberry Pi

Machine Learning



Parts Ordering Status

Name	Quantity	Status
820nF	1	Ordered
100nF	5	Ordered
4.7uF	1	Ordered
100pF	1	Ordered
0.33uF	1	Ordered
100pF	3	Ordered
MicroSD Card Adapter	1	Received
TISP4350H3BJR-S	1	Received
BZX84B33VLYT116	1	Received
1N4148	1	Received
SMAJ5.0CA-E3/61	1	Received
1N4004-T	4	Received
0.91_OLED_128x32	1	Received
DS1133-S60BPX	2	Received
TJ-L257FGHRMFCSFLC2R-A5	3	Ordered
WS2812C/W	3	Received
IRF530PBF	1	Received
ESP-WROOM-32 DEVKIT V1	1	Received
22KΩ	1	Ordered
1KΩ	1	Ordered
10.7KΩ	1	Ordered

Name	Quantity	Status
604Ω	1	Ordered
620Ω	1	Ordered
470Ω	1	Ordered
100kΩ	5	Ordered
68kΩ	2	Ordered
43kΩ	1	Ordered
10kΩ	5	Ordered
4.7MΩ	2	Ordered
40.2KΩ	1	Ordered
3.3kΩ	1	Ordered
100kΩ	2	Ordered
200kΩ	1	Ordered
TTC-5017F	1	Received
LTV-817S-TA1	1	Received
LMC555N	1	Received
SN74LV1T08DBVR	1	Received
PLA192STR	2	Received
PC817X3NSZ9F	1	Received
LMV824M/TR	1	Received
PCB	1	Ordered

Execution Plan

	8/31	9/7	9/14	9/21	9/28	10/5	10/12	10/19	10/26	11/2	11/9	11/16	11/23	11/30	Team Member	
Status Update 1															Team	Key
Design PCB															Amy	Ongoing
Order PCB															Amy	Complete
Generate an SVM															Matthew	Planned
Train and test SVM on known dataset															Matthew	Behind
Local storage receives recordings															Scott	
Test and run code on Raspberry Pi															Matthew	
Outline C Code for state machine															Amy	
Test display code on ESP32															Amy	
Train and test SVM on unfamiliar dataset															Matthew	
Status Update 2															Team	
Assist in helping capture FSK encoded CID/CWID															Amy	
ESP32 Captures incoming FSK encoded CID															Scott	
Begin writing code for state machine															Amy	
Create filesystem for storing all known callers (white/blacklist)															Matthew	
Generate accurate voice mappings from test data to known data															Matthew	
Test PCB															Amy	
Edit and order 2nd PCB															Amy	
Write state machine for possible states															Scott	
Interface with Database															Matthew	
Status Update 3															Team	
Recieve audio files from database															Matthew	
Handset properly records through ESP32															Scott	
Status Update 4															Team	
Test 2nd PCB															Amy	
Intergrate with Database															Amy	
Send SVM out to database/onto host machine															Matthew	
Miscellaneous integration															Team	
Integration Checks from ML Subsystem															Matthew	
Status Update 5															Team	
Final Validation															Team	
Final Design Presentation															Team	
Final System Demo															Team	
Virtual Project Showcase Video															Team	
Final Report															Team	

Validation Plan

Validation Plan				
Test	**Detail**	**Data**	**Status**	**Responsible Student**
Device powers on	Turns on Raspberry Pi and ESP32	Turns on	Complete	Amy Chen
Display powers on	Displays caller ID information		Complete	Amy Chen
Ring detect	LED lights up when detection occurs		WIP	Amy Chen
Suppress ring	Suppresses ring after initial ring		WIP	Amy Chen
Detect off-hook/on-hook	LED lights up when detection occurs		WIP	Amy Chen
Detect hook flash on ESP32	Detect hook flash in firmware		WIP	Amy Chen
Arduino IDE	Set up Arduino IDE		Complete	Amy Chen
Decode CID/CWID on ESP32	Decode CID/CWID information in firmware		WIP	Amy Chen
Decode DTMF and FSK on ESP32	Decode DTMF and FSK in firmware		WIP	Amy Chen
OLED program	Code for OLED display		Complete	Amy Chen
WS2812B program	Code for LED light		Complete	Amy Chen
Control WS2812B	Test code on LED light		Complete	Amy Chen
WiFi program	Code for WiFi STA Mode		Complete	Amy Chen
Implement WiFi program on ESP32	Load code onto ESP32	Successfully connects and disconnects to WiFi	Complete	Amy Chen
Bluetooth program	Code for serial to serial Bluetooth		Complete	Amy Chen
Implement Bluetooth program on ESP32	Load code onto ESP32		WIP	Amy Chen
Write a recording	Write a recording as WAV file and store in SD card		WIP	Amy Chen
Play a recording	Open a WAV file from SD card play the recording		WIP	Amy Chen
Implement audio code on ESP32	Load code onto ESP32		WIP	Amy Chen
Retrieve file from database	The file will be in the given or created directory that the user has input		Complete	Scott Kevil-Yeager
UI works as expected, allowing users to input test folder directories	UI works as expected, allowing users to input test folder directories		Complete	Scott Kevil-Yeager
Upload folder	Files in given directory will be counted, processed, named, and uploaded to the database automatically		Complete	Scott Kevil-Yeager
Listen to recording	Properly allows the playback of recording audio through the host machine, this assumes that the host machine will have a speaker		Complete	Scott Kevil-Yeager
Error checking	If a folder directory or file directory is incorrectly given then a message is given and the user is prompted for another input		Complete	Scott Kevil-Yeager
Delete recording in database	Given a valid name the function removes a single entry from the database		Complete	Scott Kevil-Yeager
Delete local recording	If a folder path and file name are given then the function will delete the local file		Complete	Scott Kevil-Yeager
pyAudioAnalysis	Removes periods of silence in recordings to reduce file size		Complete	Scott Kevil-Yeager
Local storage receives recordings			WIP	Scott Kevil-Yeager
ESP32 Captures incoming FSK encoded CID			WIP	Scott Kevil-Yeager
Write state machine for possible states			WIP	Scott Kevil-Yeager
Handset properly records through ESP32			WIP	Scott Kevil-Yeager
feature extraction on a wav file	uses pAA to do feature extraction on a wav file and prints the names of all features extracted	log of all features extracted from a given wav file	Complete	Matthew Hebrado
genereate files used to train SVM	take a source file and split it into 1 sec intervals	several wav files are produced that are 1 sec long	Complete	Matthew Hebrado
graph feature comparisons	based on the feature extraction graph is generated that displays a comparison of the two speakers		Complete	Matthew Hebrado
create SVM classification file	does feature extraction on all files in a directory and creates SVM file		Complete	Matthew Hebrado
run tests/predictions from known speakers		~80% accuracy accross the board	Complete	Matthew Hebrado
run tests/predictions from unknown speakers			WIP	Matthew Hebrado
code runs on pi			WIP	Matthew Hebrado
send file to database			WIP	Matthew Hebrado
recieve file from database			WIP	Matthew Hebrado

Thank You!