



SecureScape



G R O U P 1 5

F I N A L

P R E S E N T A T I O N





Jaxon Topel
Computer Engineering

Introduction & Our Group

- **SecureScape** is a smart, portable security system designed to detect motion, classify threats, and provide real-time alerts in remote areas.



Jaxon Topel
Computer Engineering



Colin Kirby
Computer Engineering



Phillip Murano
Electrical Engineering



Dylan Myers
Electrical Engineering





Phillip Murano
Electrical Engineering

Motivations & Basic Goals

- Many remote areas lack affordable security solutions.
- **SecureScape** offers a portable, motion-detecting security system with real time alerts.
- Ideal for campers, hikers, and rural property owners needing extra security.

Goals :

- Create three portable nodes with IR sensors & Cameras.
- Design a central gadget to manage communication and send alerts.
- Use image processing to detect motion and trigger alarms when a threat is detected

Objectives:

- Establish wireless communication between nodes and the central gadget.
- Enable real-time image processing for motion detection and threat classification.
- Maximize power efficiency for a portable, long-lasting security system.





Phillip Murano
Electrical Engineering



Advanced Goals & Objectives

Advanced Goals :

- Add onboard memory for local data storage.
- Implement user-controlled alarm modes (quiet/loud).
- Enable tamper detection to trigger alarms when nodes are compromised.

Stretch Goals :

- Use recognition algorithms to distinguish between humans and animals.
- Notify users of threat levels based on detected entities.

Advanced Objectives :

- Store capture images and logs for offline access.
- Add a toggle for users to switch between quiet and loud alarms.
- Use motion sensors to detect handling and trigger alerts.

Stretch Objectives :

- Train a model to distinguish between humans and animals.
- Rank threats by severity and send real-time notifications.



Dylan Myers
Electrical Engineering

Specifications

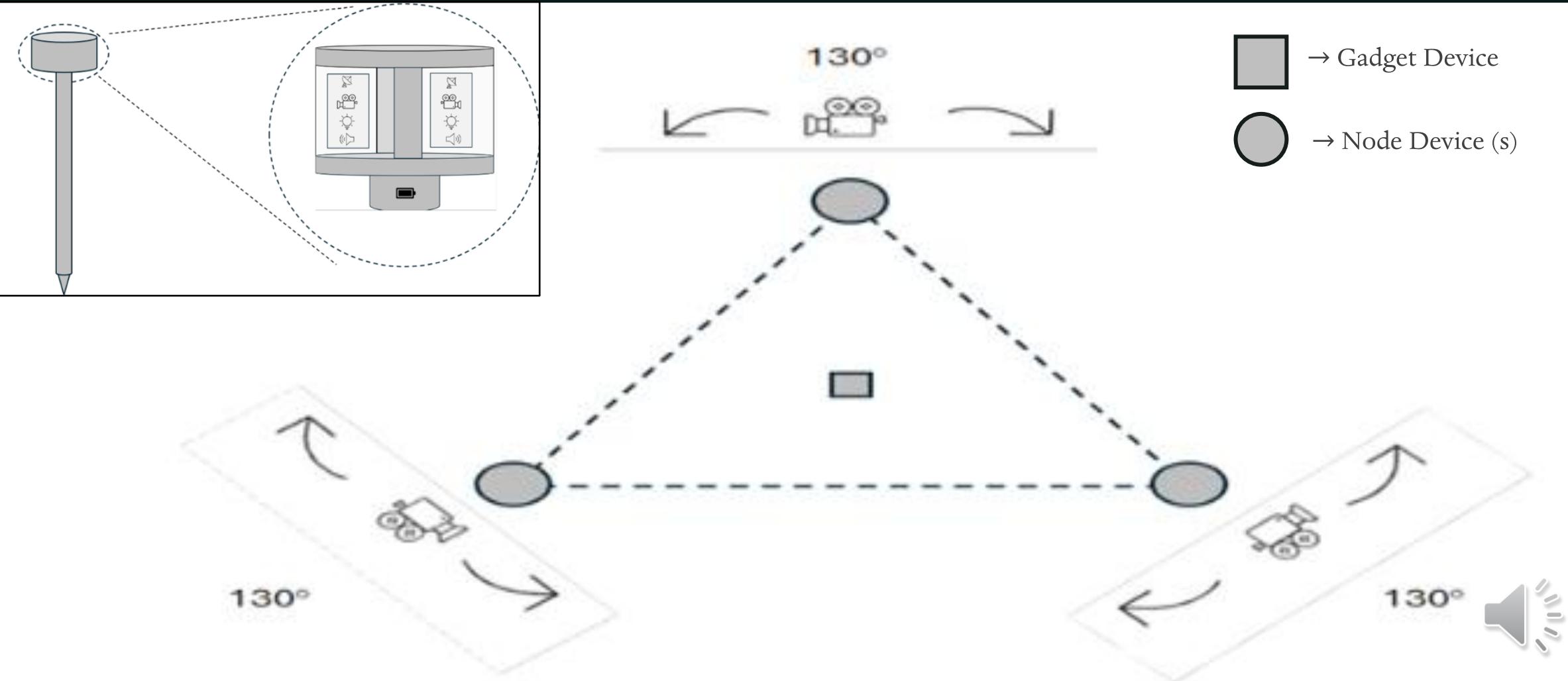
 = Demonstratable Specifications

Metric	Minimum	Maximum	Relevance to project
 Detection Accuracy	60%	~90%	Essential for reliable threat identification.
Image Processing Time	500 ms	2 seconds	Topic one Critical for timely classification of threats.
 Response Time	5 seconds	10 seconds	Important for timely notifications and threat management.
Deployment Time	5 minutes	N / A	Ensures a focus on quick set up in remote locations.
 Detection Range	5 ft	15 – 20 feet	Critical for effective coverage of a secure area.
Mobile App Latency	100 ms	300 ms	Important for smooth UX when controlling the system.



Project Overview

Colin Kirby
Computer Engineering





Phillip Murano
Electrical Engineering

Comparison & Selection of Hardware

- Phillip Murano
- Dylan Myers





Phillip Murano
Electrical Engineering

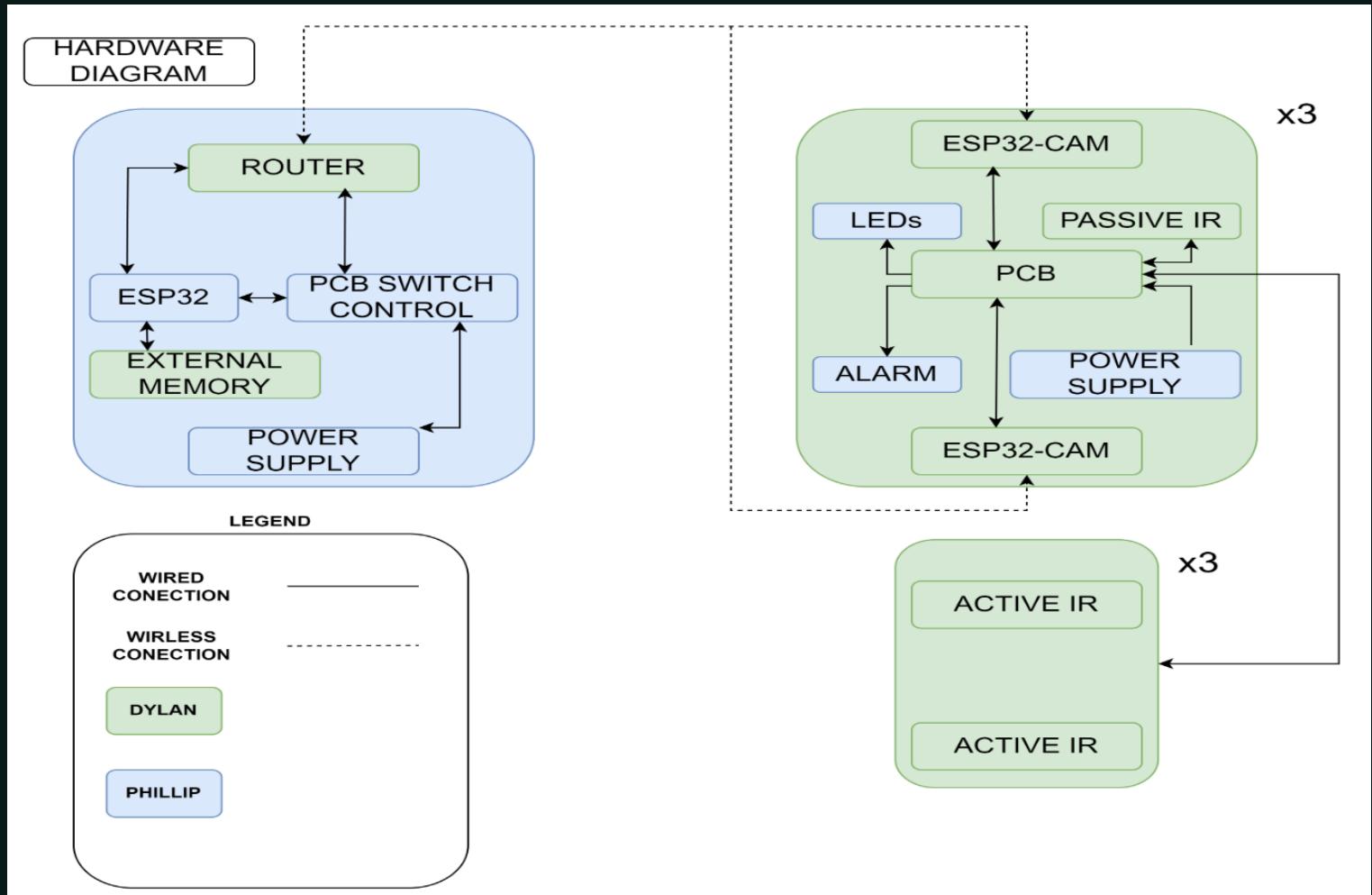


Hardware Diagram

This diagram shows the hardware setup of **SecureScape**, including :

- Nodes
- A central gadget
 - A router

This illustrates how components of our system interact to ensure efficient operation.

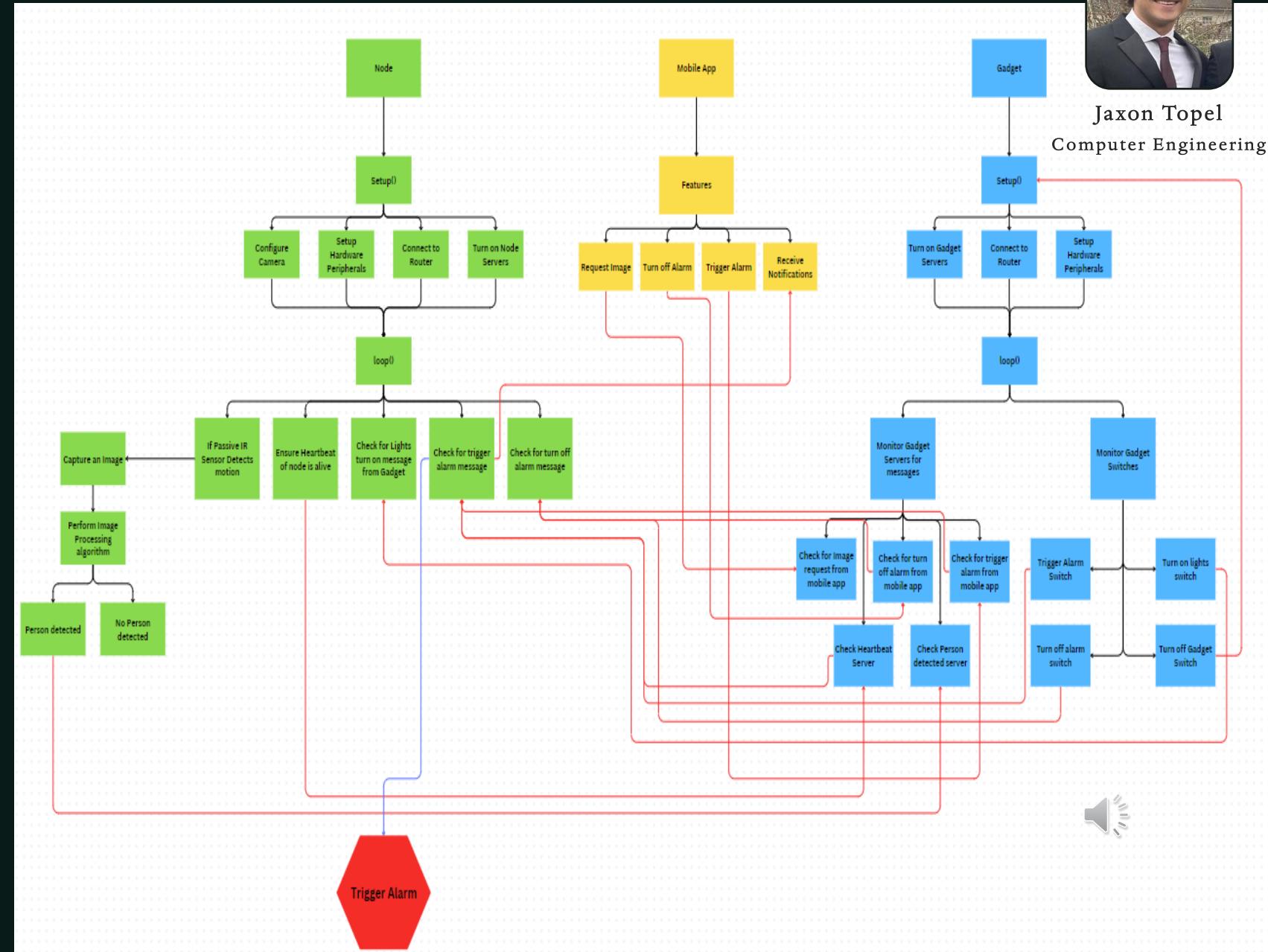


Software Design Flowchart

This chart outlines the software design for CAM nodes & S3 Gadget, detailing their respective processes for :

- Motion detection
- Image processing
- Alarm management
- Communication

with the SecureScape system.



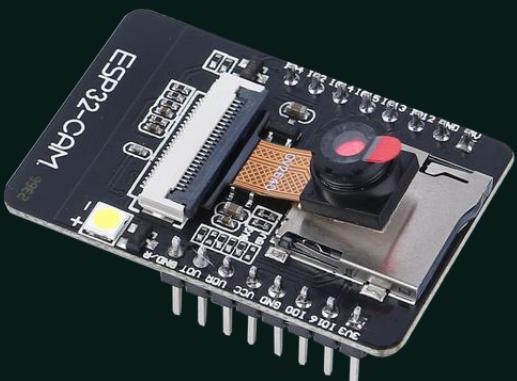
Jaxon Topel
Computer Engineering





Phillip Murano
Electrical Engineering

Camera – ESP32-CAM



Part	ESP32-CAM	Raspberry Pi 4 B 3 B+ Camera Module	WayPonDEV CanMV-K230
Cost per MCU	\$7.67	\$43.47	\$25.99
Video Resolution	480p	1080p	3840 x 2160
Power Consumption	260 mA/h	2.85 watts	185mW-585mW
Board Dimension	4.37 x 3.7 x 1.14 inches	3.54 x 3.54 x 0.39 inches	3.35 x 2.2 x 0.3 inches
IP rating (waterproof)	-	-	67
CRI index	-	-	85





Phillip Murano
Electrical Engineering

LEDs



Part	Samsion	CHNMALITAI	KXZM
Cost / Per	\$7	\$10	\$11
Voltage (V)	5	5	5
LEDs / Length (m)	320 / 1	320 / 1	640 / 2
Wattage per meter (W/m)	6	5	5
IP rating (waterproof)	67	65	65
CRI index	85	90	93

Part	CHANZON	Novelty Place	WOTOFULIN
Cost	\$5	\$6	\$7.29
Quantity	100	100	100
Voltage (V)	2 – 2.2	2 – 2.2	2 – 2.4
Current (mA)	20	20	20
Brightness (mcd)	600-800	3000-4000	8000-12000
Light Beam Angle (Deg)	20	30	20-30





Phillip Murano
Electrical Engineering

Alarm - WEICHUANG

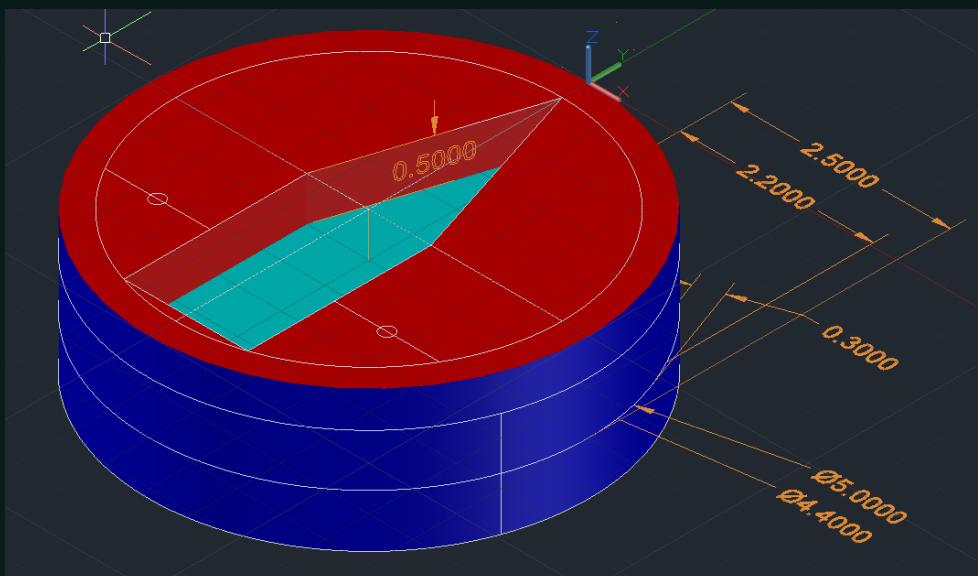
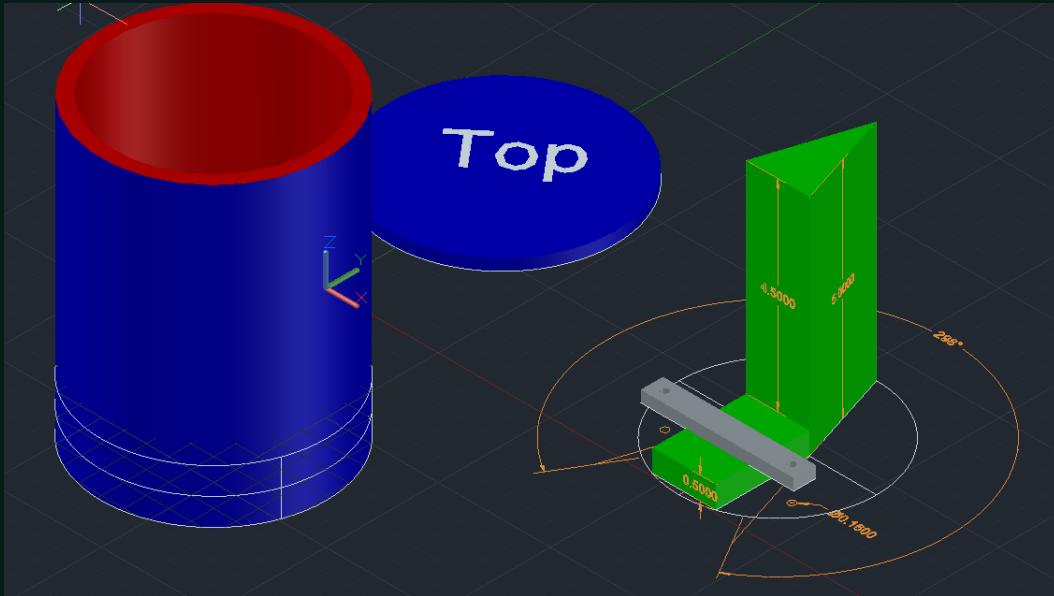


Part	WEICHUANG	BNYZWOT	Jameco
Cost / Per	\$5.49 / 3	\$8.39 / 10	\$7.90 / 2
Range of Voltage (V)	3 - 24	3 - 24	4 - 16
Rated Voltage (V)	12	12	12
Rated Current (mA)	15	12	25
Sound Output (dB)	100	85	95
Frequency Response (Hz)	3300 ± 500	3300 ± 500	2900



Node Structural Design

These CAD designs showcase the structural layout of the **SecureScape** node, including its housing, and key dimensions for integration.



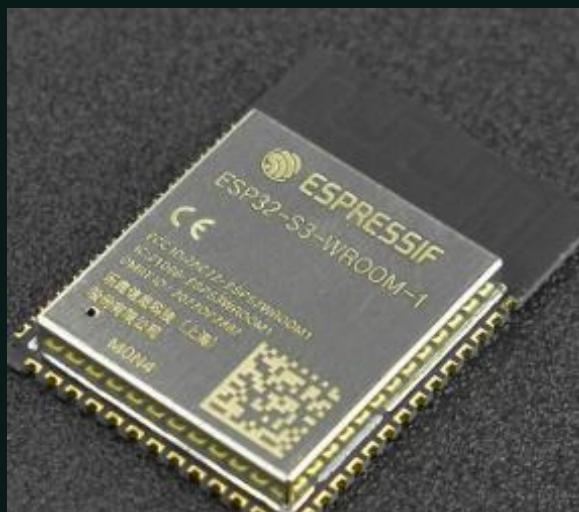
Phillip Murano
Electrical Engineering





Phillip Murano
Electrical Engineering

ESP32-S3 WROOM 1 (GADGET)



Part	ESP32-S3XX	STM32WBA5xx	PIC32MZXW1
Cost	\$5.77	\$4.92	\$9.41
Processor	Dual core LX7 240MHz	Cortex-M33 100MHz	MIPS32 M-Class 200MHZ
Memory (PSRAM/FLASH)	8MB/16MB	128KB/1MB	256-512KB/ 1-2MB
Operating Voltage (V)	3.0-3.6	1.71-3.6	2.97-3.63
GPIO Pin Count	36	20-35	37-62
Connectivity	Wi-Fi / Bluetooth	Wi-Fi / Bluetooth	Wi-Fi





Phillip Murano
Electrical Engineering

DC-DC Converter - AMS1117-3.3V



Part	TPS65090	TPS65023	AMS1117-3.3V
Voltage In Range (V)	6 -17	2.5 - 6	4.75 – 12
Voltage Out Range (V)	1 - 5	0.6 - 6	3.3
Max Current Out (A)	5	1.7	1
Quiescent Current (μ A)	30 - 110	85	10
Step Down converters	3	3	0
LDOs	2	3	1
Cost	\$10.85 / 1	\$3.29 / 1	\$0.153 / 1





Phillip Murano
Electrical Engineering

Memory - SanDisk 128GB Ultra



Part	SanDisk 128GB Ultra	SanDisk 256GB Extreme	TEAMGROUP GO Card
Cost	\$14.20	\$25.54	\$16.99
Capacity	128 GB	256 GB	256 GB
Retrieval	140 Megabytes Per Second	190 Megabytes Per Second	100 Megabytes Per Second
Dimensions	0.04"D x 0.59"W x 0.43"H	0.4)L x 0.5" W	0.04"D x 0.59"W x 0.43"H
Encoding	500	130	50





Phillip Murano
Electrical Engineering

Router – GL.iNet “Shadow”



Frequency limitation:

2.4GHz

Power Input:

5V / 2A

Connectivity:

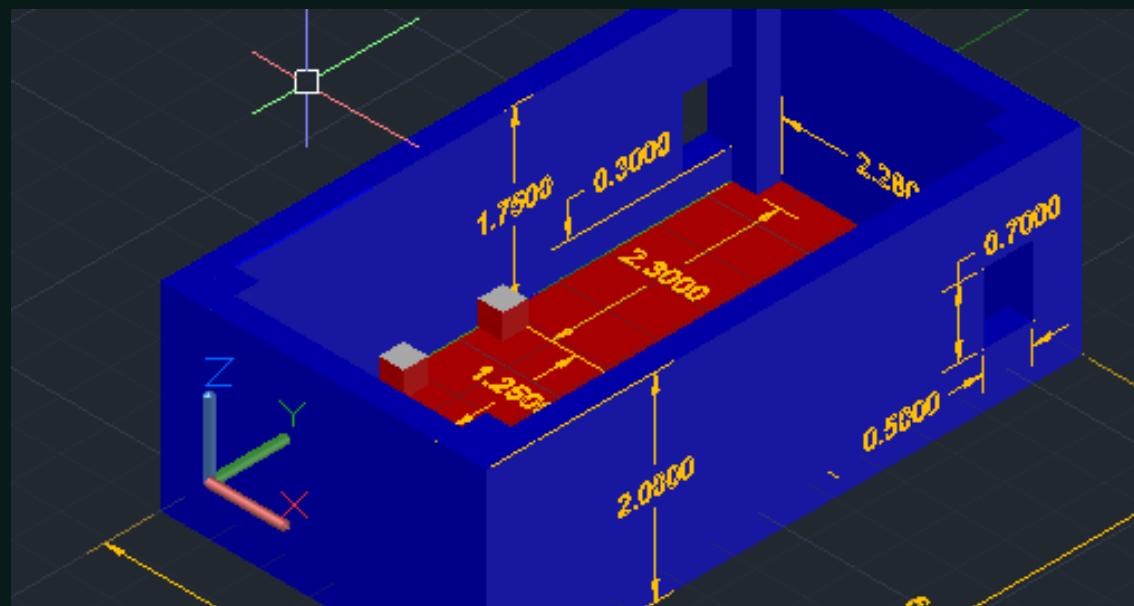
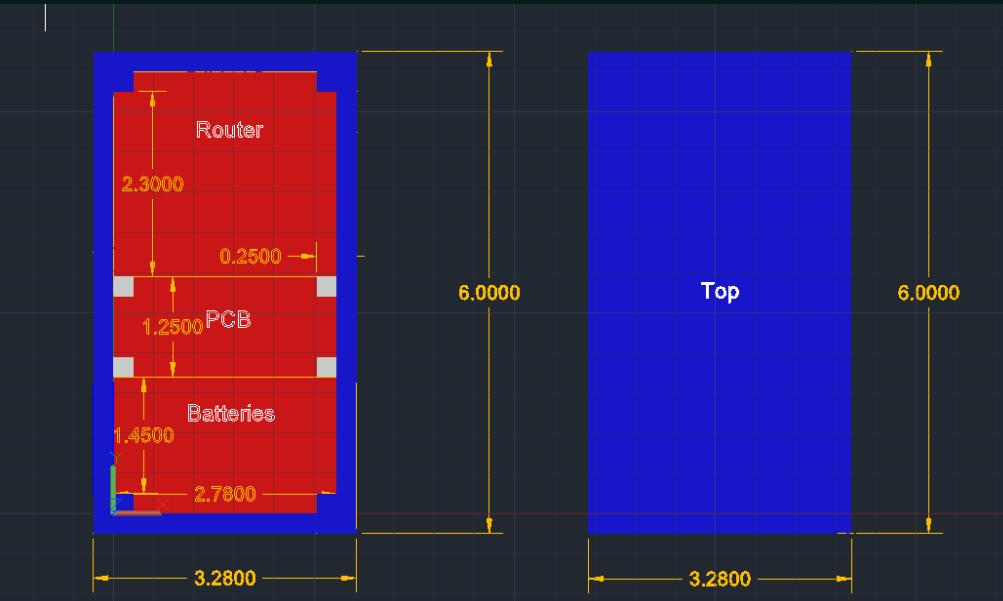
Wi-Fi / Ethernet

Part	GL.iNet GL-SFT1200	TP-Link AC750	GL.iNet GL-AR300M16-Ext Portable
Cost	\$34.90	\$33.99	\$36.90
CPU	SF19A28, Dual-Core @1GHz	IPQ4018, Quad Core@717Mhz	QCA9531, @650MHz SoC
Product Dimensions	4.65" L x 3.35" W x 1.18" H	2.91" L x 0.87" W x 2.64" H	2.28 x 2.28 x 0.98 inches
EAP Support	No	Yes	Yes
Memory	DDR3 128MB	DDR3L 128MB	DDR2 128MB
Storage	SPI NAND Flash 128MB	-	FLASH 16MB
Speed	Max. 300Mbps	Max. 400Mbps	300Mbps



Gadget Structural Design

These CAD designs highlight the structural layout of the **SecureScape** gadget, showcasing the compartmentalization of key components such as the router, PCB, and batteries.

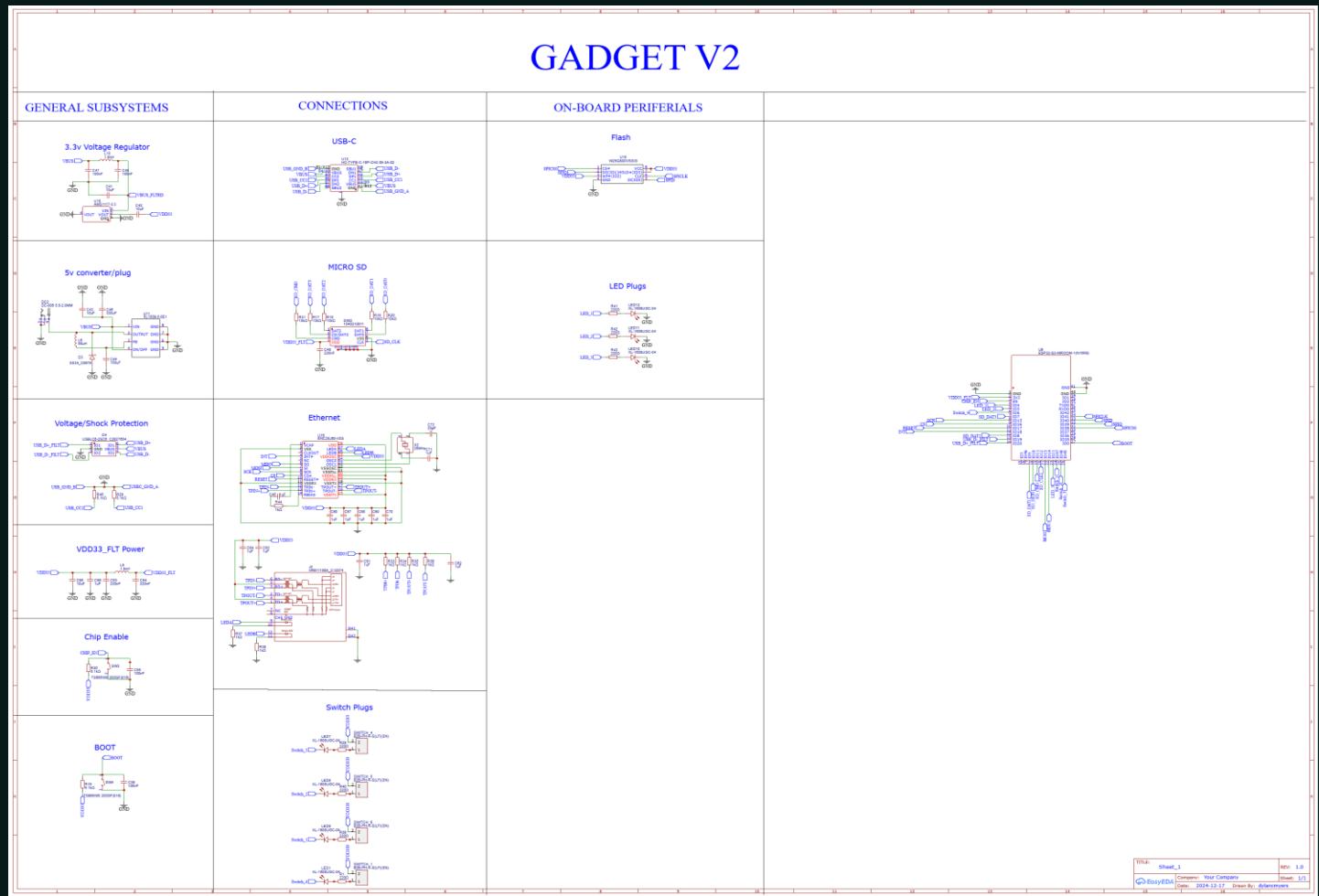


Phillip Murano
Electrical Engineering



Gadget Schematic Overview

LED Plugs
 Switch Plugs
 Ethernet
 5V Converter / Plug
 VDD3p3 Power
 VDDA Power
 Voltage Regulator
 RTC Power
 Voltage Shock Protection
 ...



Dylan Myers
Electrical Engineering

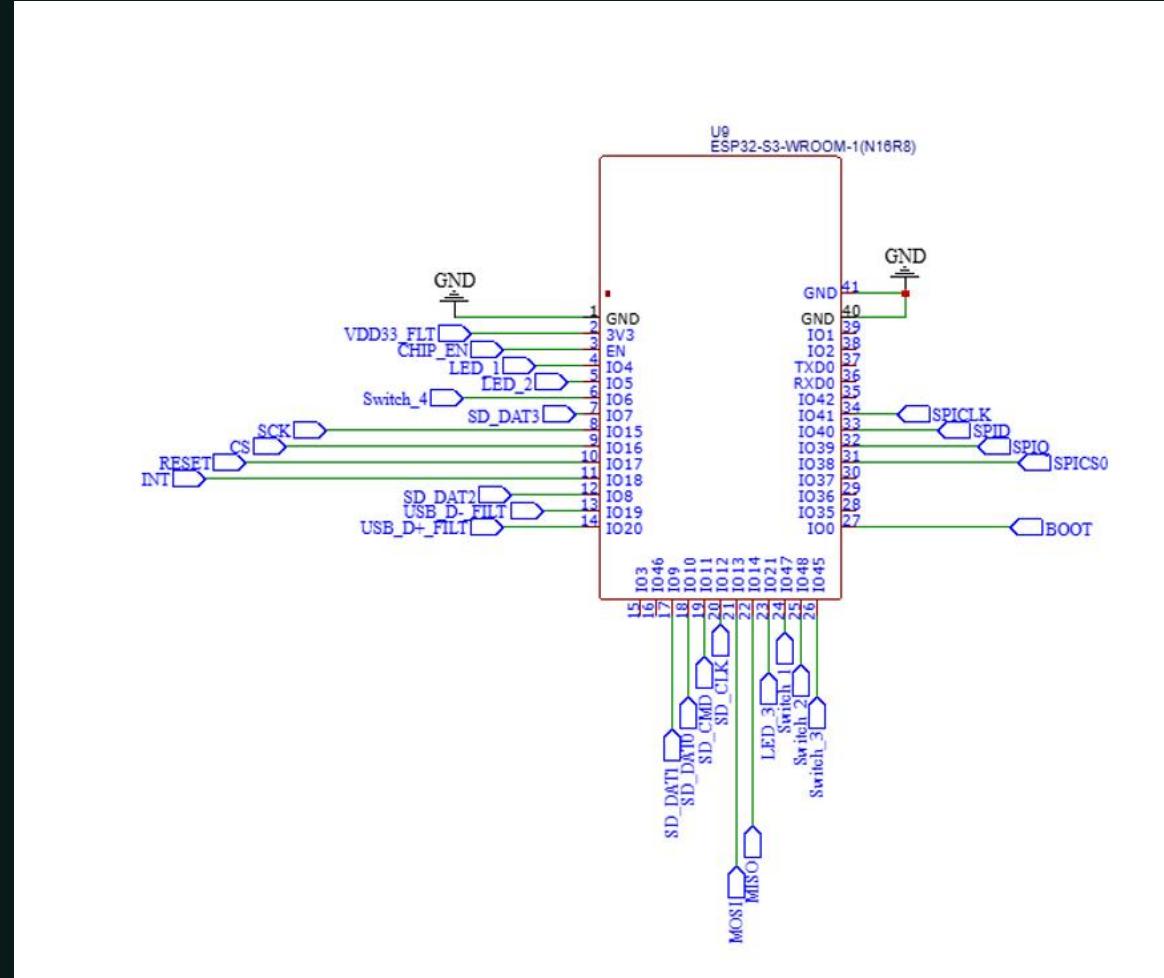




Dylan Myers
Electrical Engineering

ESP-S3- Wroom-1

- Built in Wi-Fi Antenna
- Built in USB 2.0 communications
- Internal 40MHz crystal oscillator
- Universal GPIOs
 - 8 MB PSRAM
- 16 MB Quad SPI flash





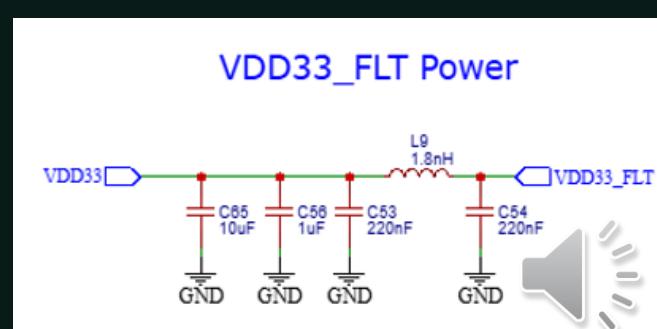
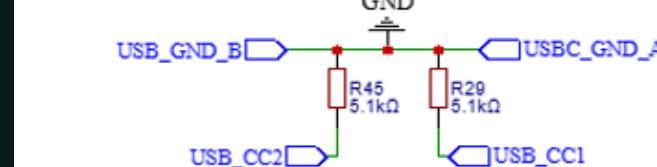
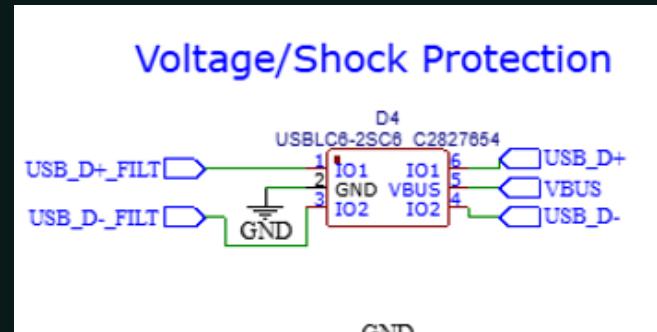
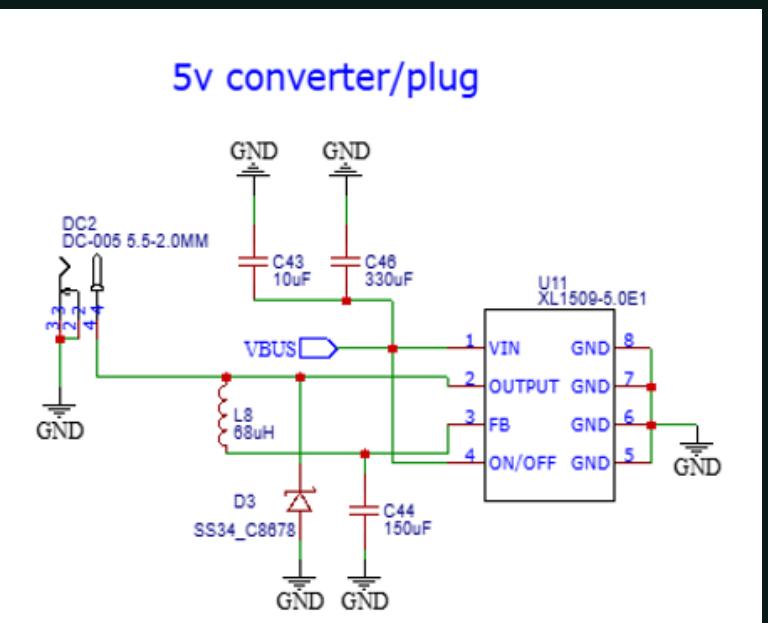
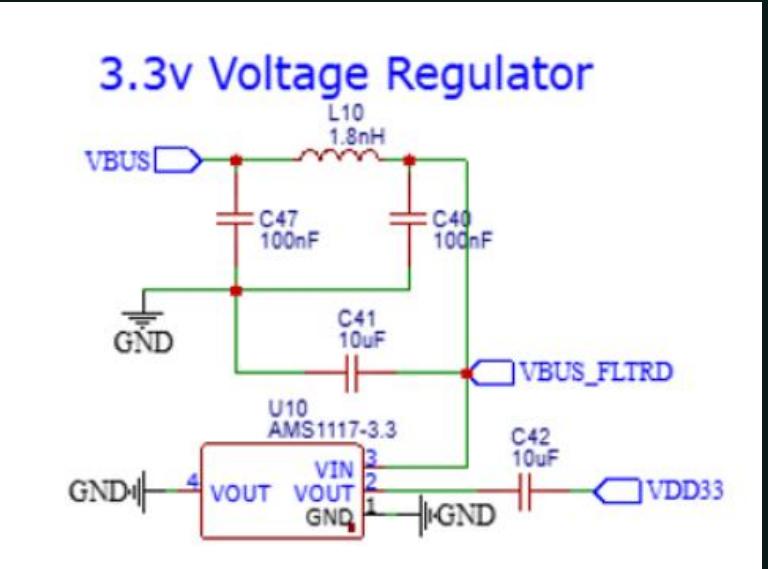
Dylan Myers
Electrical Engineering

Power Regulation

3.3V - Fixed Linear voltage out with adjustable input voltage

5V – Fixed PWM buck
Shock Protection – Path for extreme voltage spikes

Filtration – removes high frequencies



Functional Buttons

Boot - Allows us to put the ESP32-S3 into Boot mode or Flash mode

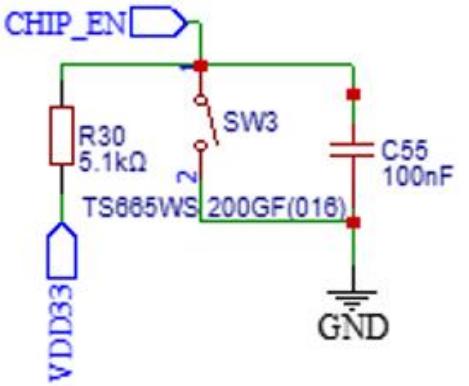
Chip Enable – Chip Enable (EN) pin on the ESP32-S3 acts as a power control or reset pin

Both are pulled high

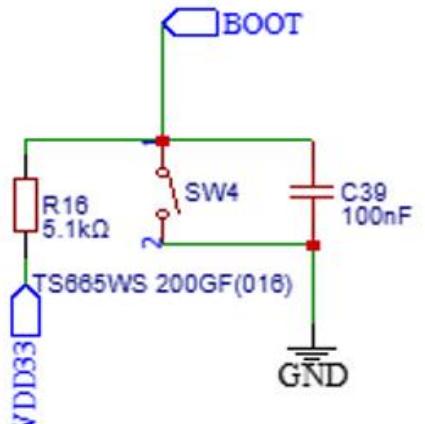


Dylan Myers
Electrical Engineering

Chip Enable



BOOT





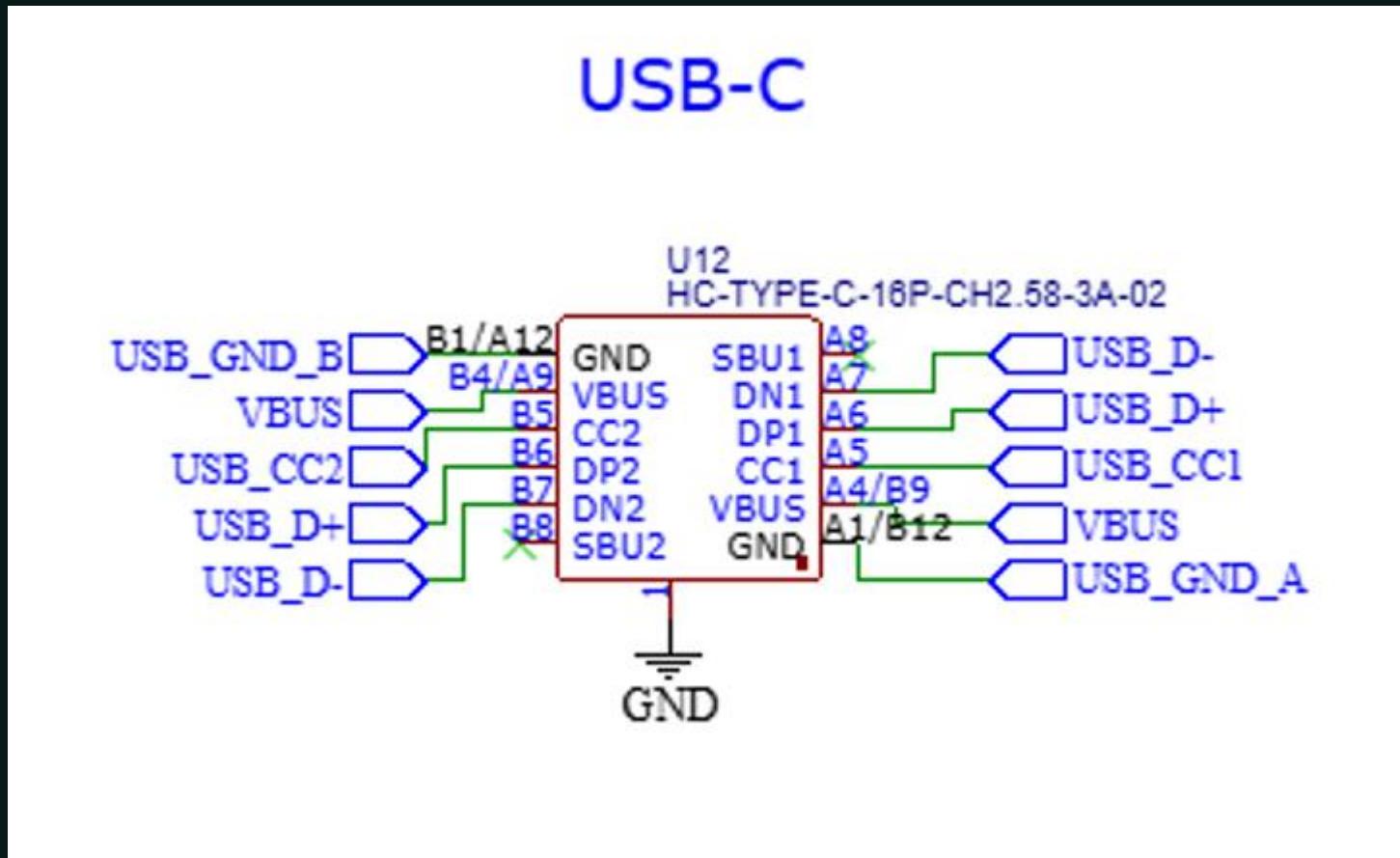
Dylan Myers
Electrical Engineering

USB-C

Used for both power and data upload

Direct connect to the ESP32-S3

ESP32-S3 features a full-speed USB OTG interface along with an integrated transceiver. The USB OTG interface complies with the USB 2.0 specification



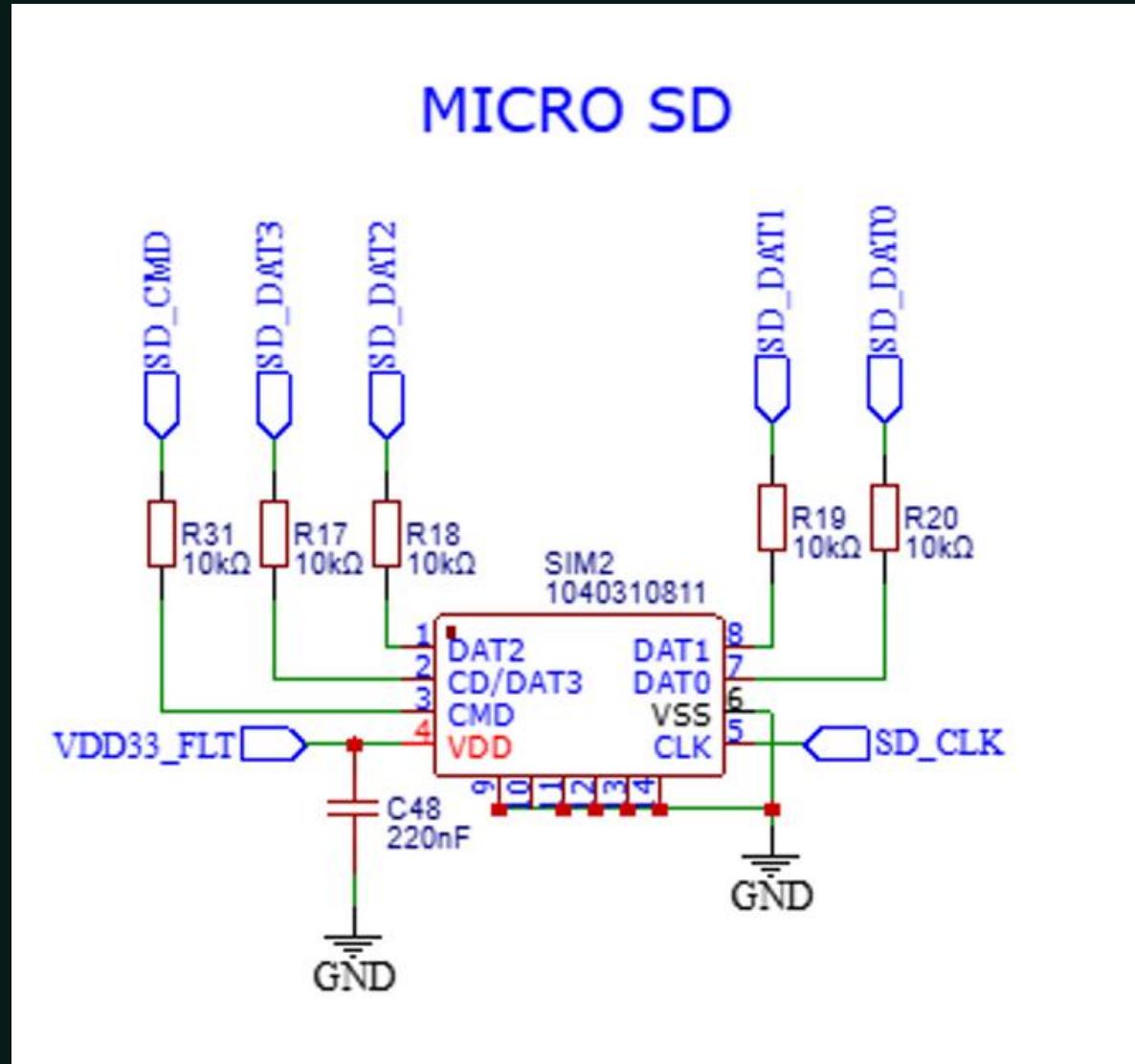


Dylan Myers
Electrical Engineering

Micro SD

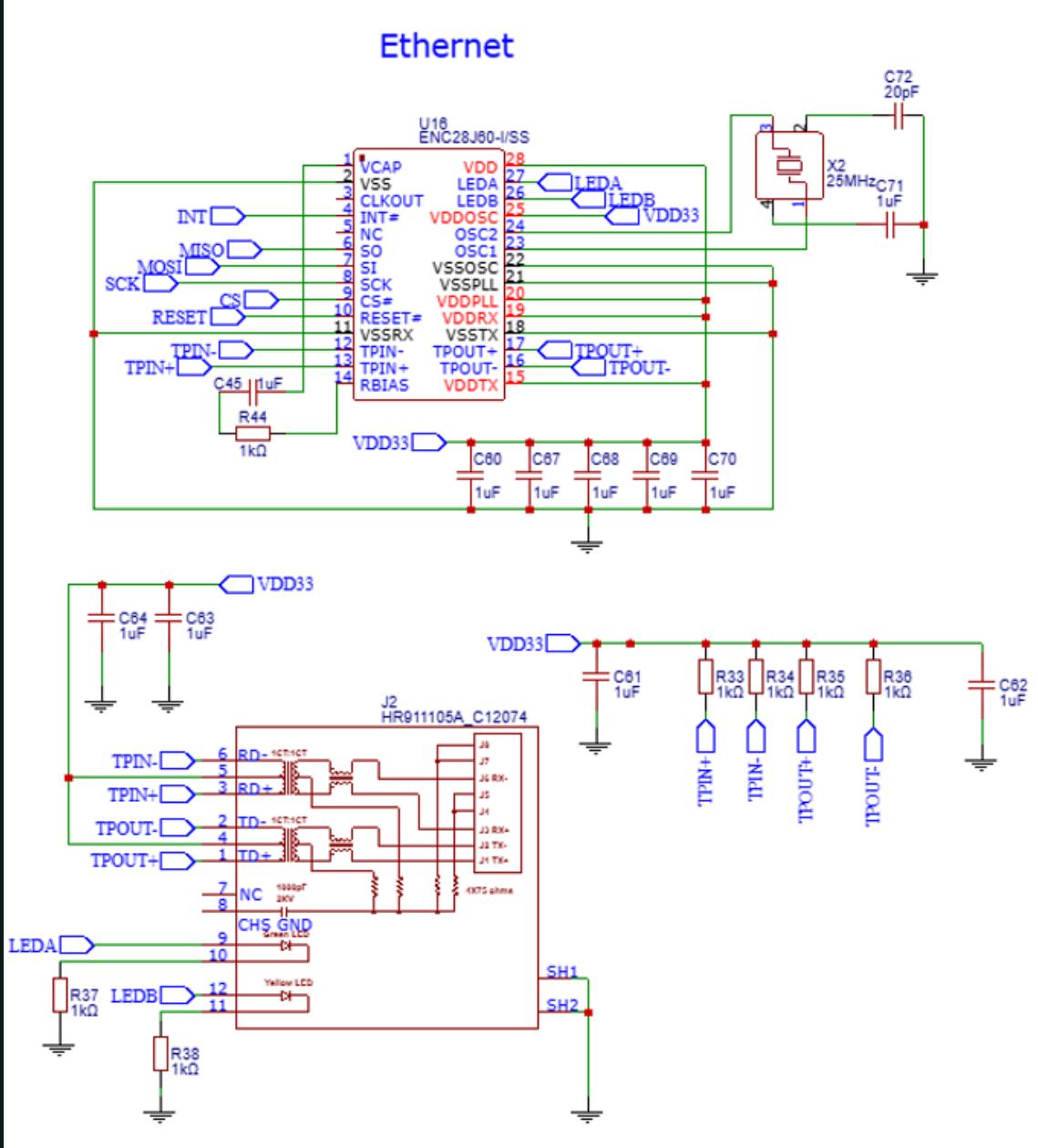
Used for upload /
download of data

Backup for the host data
of the website / App



Ethernet

Used for Backup
connection for the
Wireless LAN



Dylan Myers
Electrical Engineering

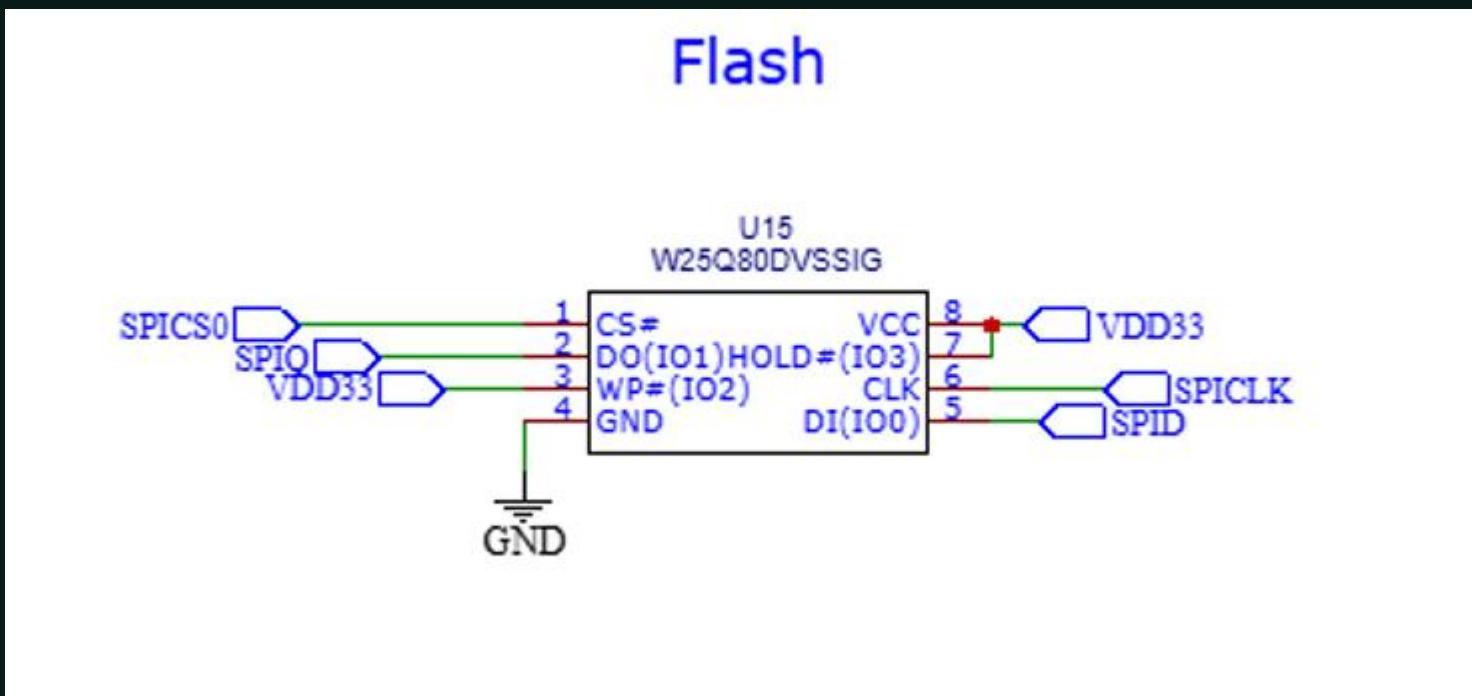




Dylan Myers
Electrical Engineering

Flash

Used for extra memory
for software



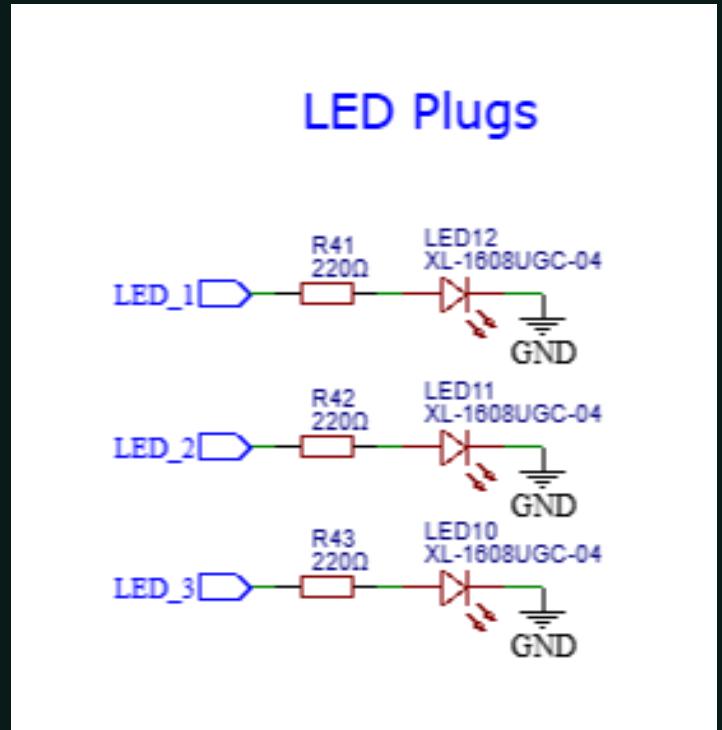
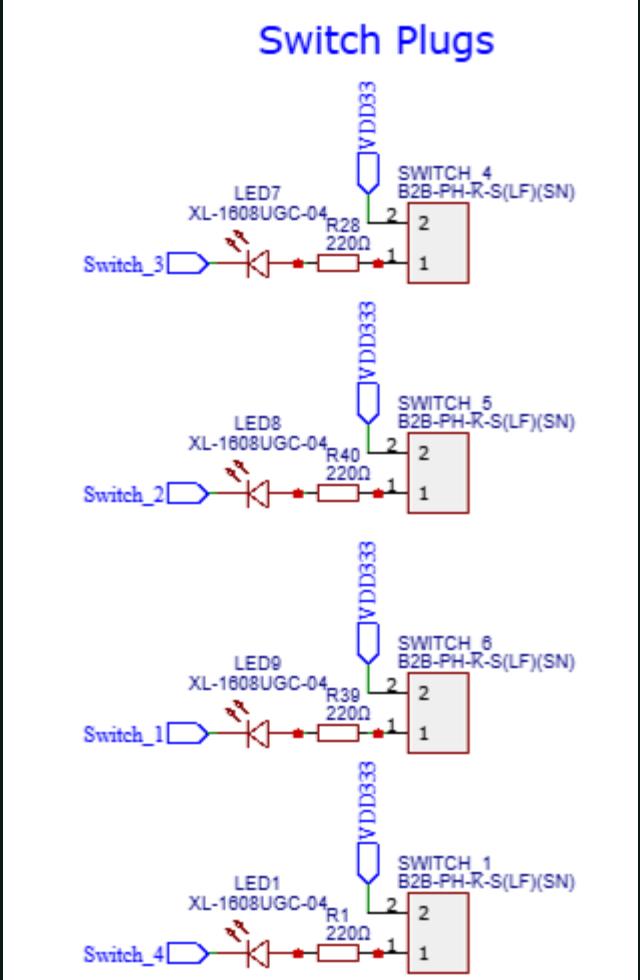


Dylan Myers
Electrical Engineering

Plugs / LEDs

Switch Plugs – will attach to the switches mounted on our housing giving basic functionality to our system.

LEDs – Built in LEDs that will be used for Testing proposes



Node Schematic Overview



Dylan Myers
Electrical Engineering

Power Distribution

USB-C Connector

UART to Serial Converter

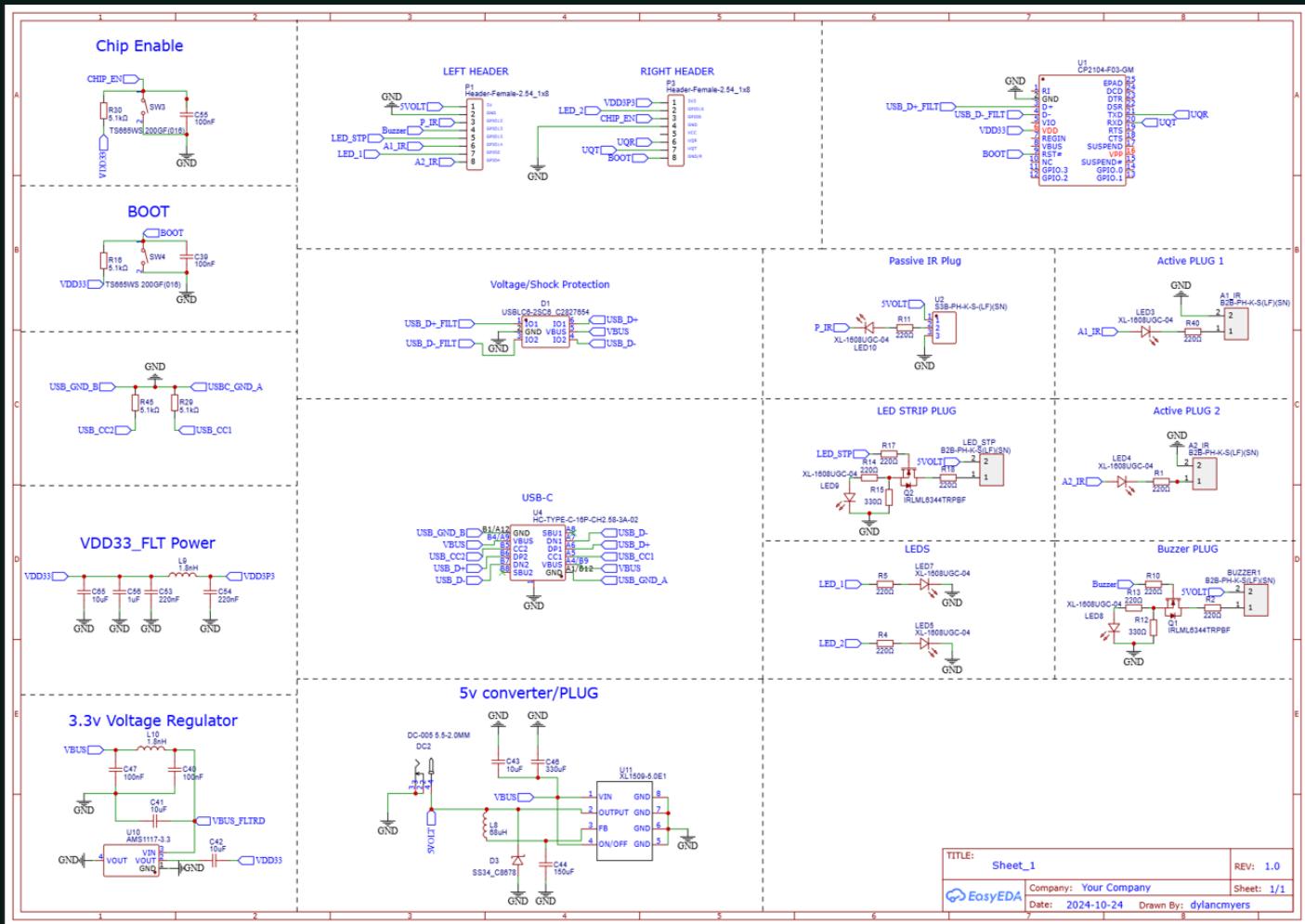
Camera Mounting

Passive IR

LED Strip

Active IR

Buzzer



TITLE:	Sheet_1	REV: 1.0
Company:	Your Company	Sheet: 1/1

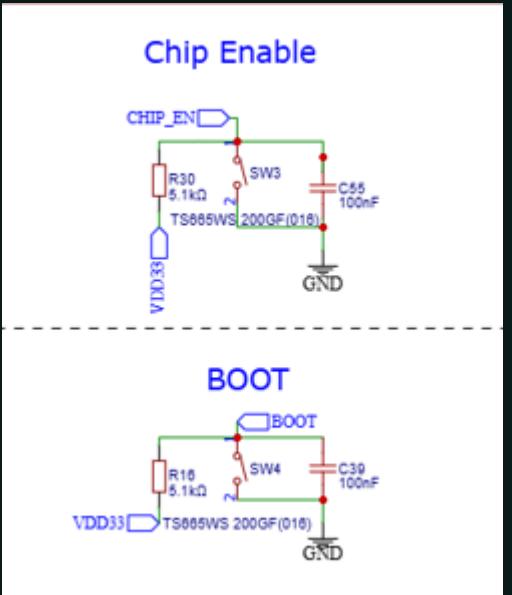




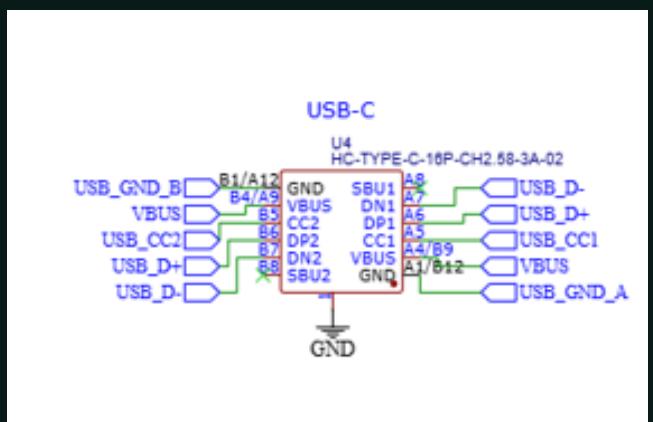
Dylan Myers
Electrical Engineering

Duplicated from Gadget PCB

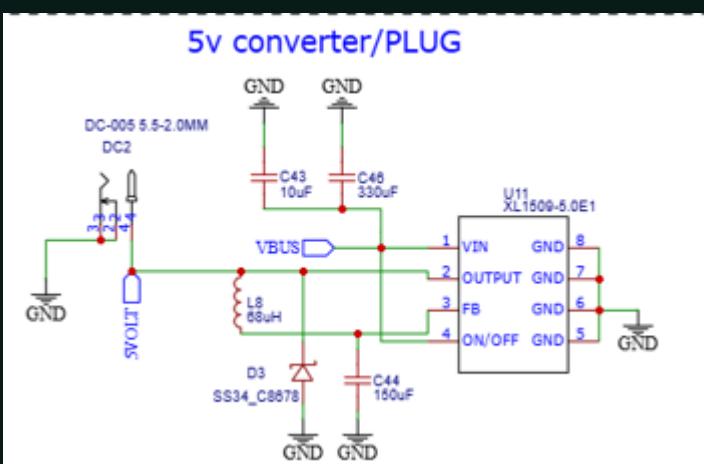
Power Distribution



USB-C Connector



Functional Buttons

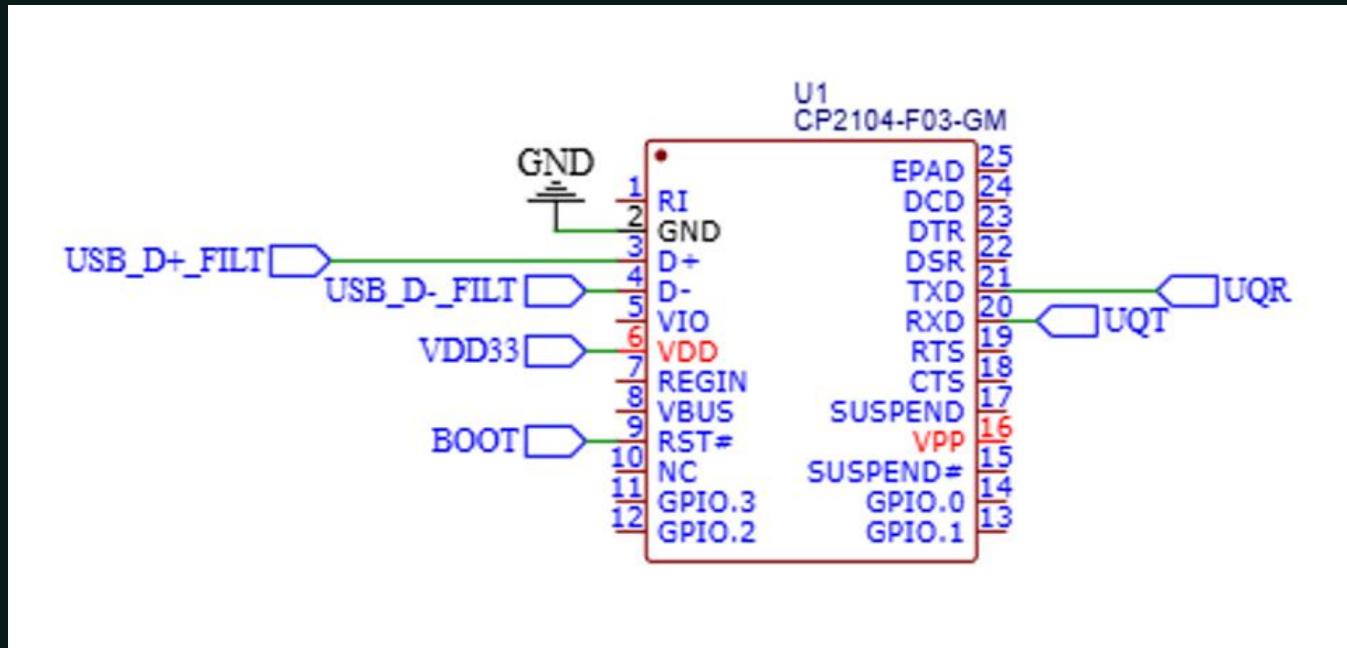




Dylan Myers
Electrical Engineering

USB-TO-UART BRIDGE

Unlike the ESP32-S3
The ESP-Cam doesn't have an
onboard UART to USB
converter

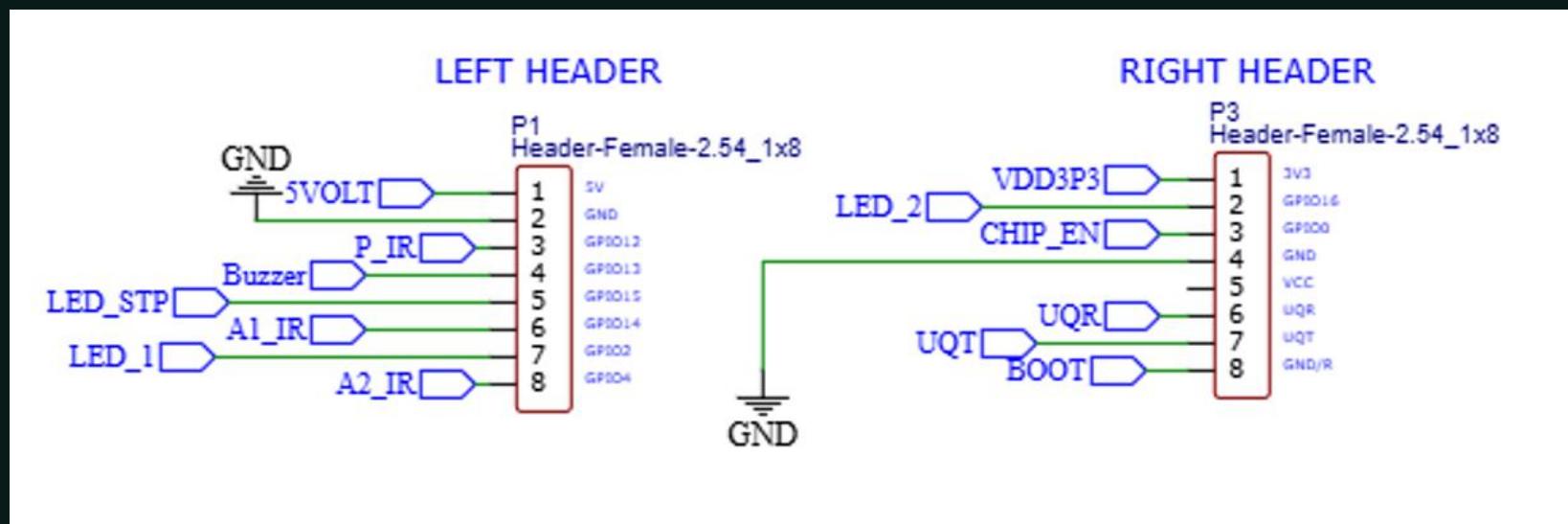




Dylan Myers
Electrical Engineering

Headers

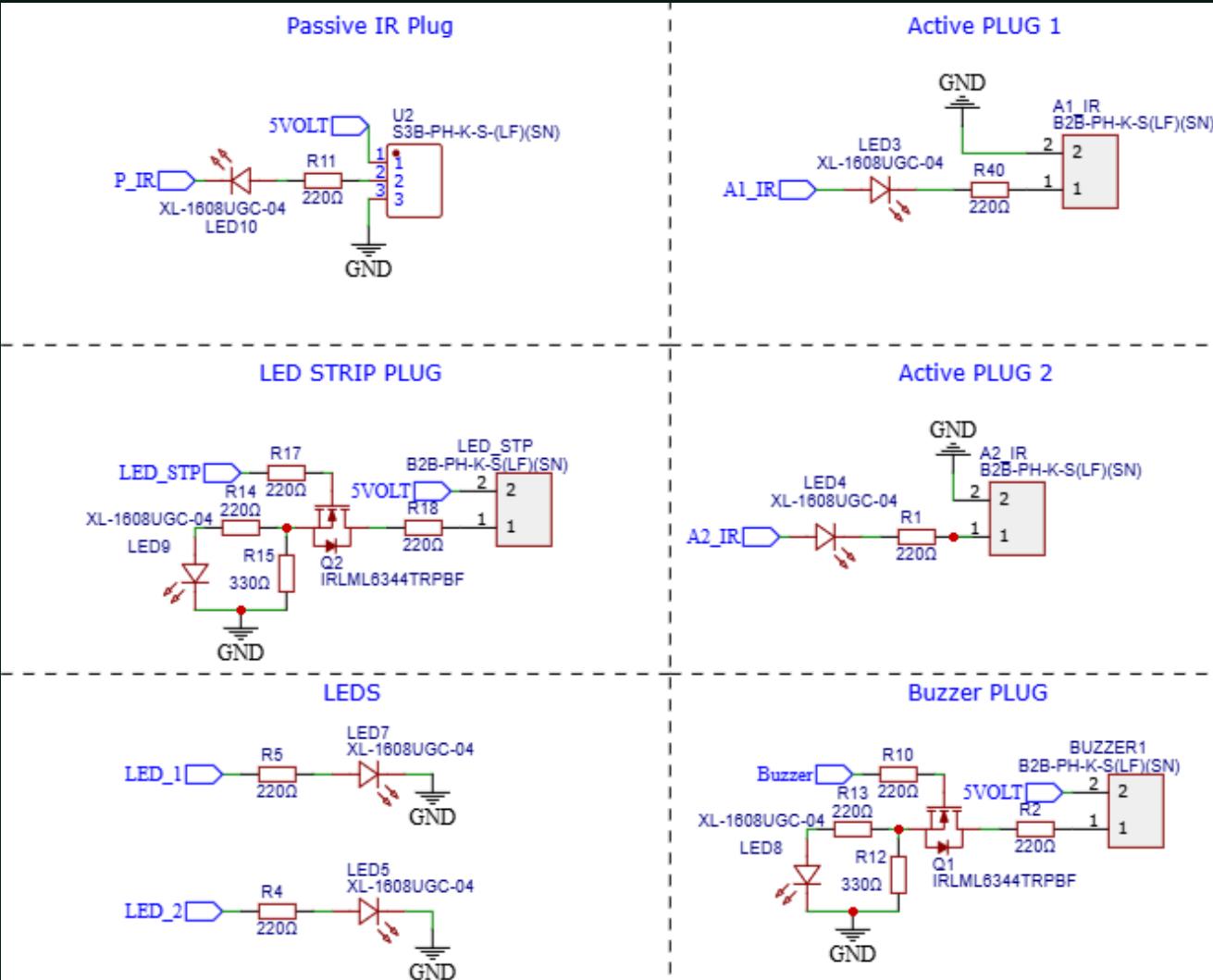
Header pins to connect the
ESP32 – CAM to the PCB





Plugs

Board Triggered – we use
Mosfets to turn on/off the
peripherals





Dylan Myers
Electrical Engineering

PCB Layout & Design

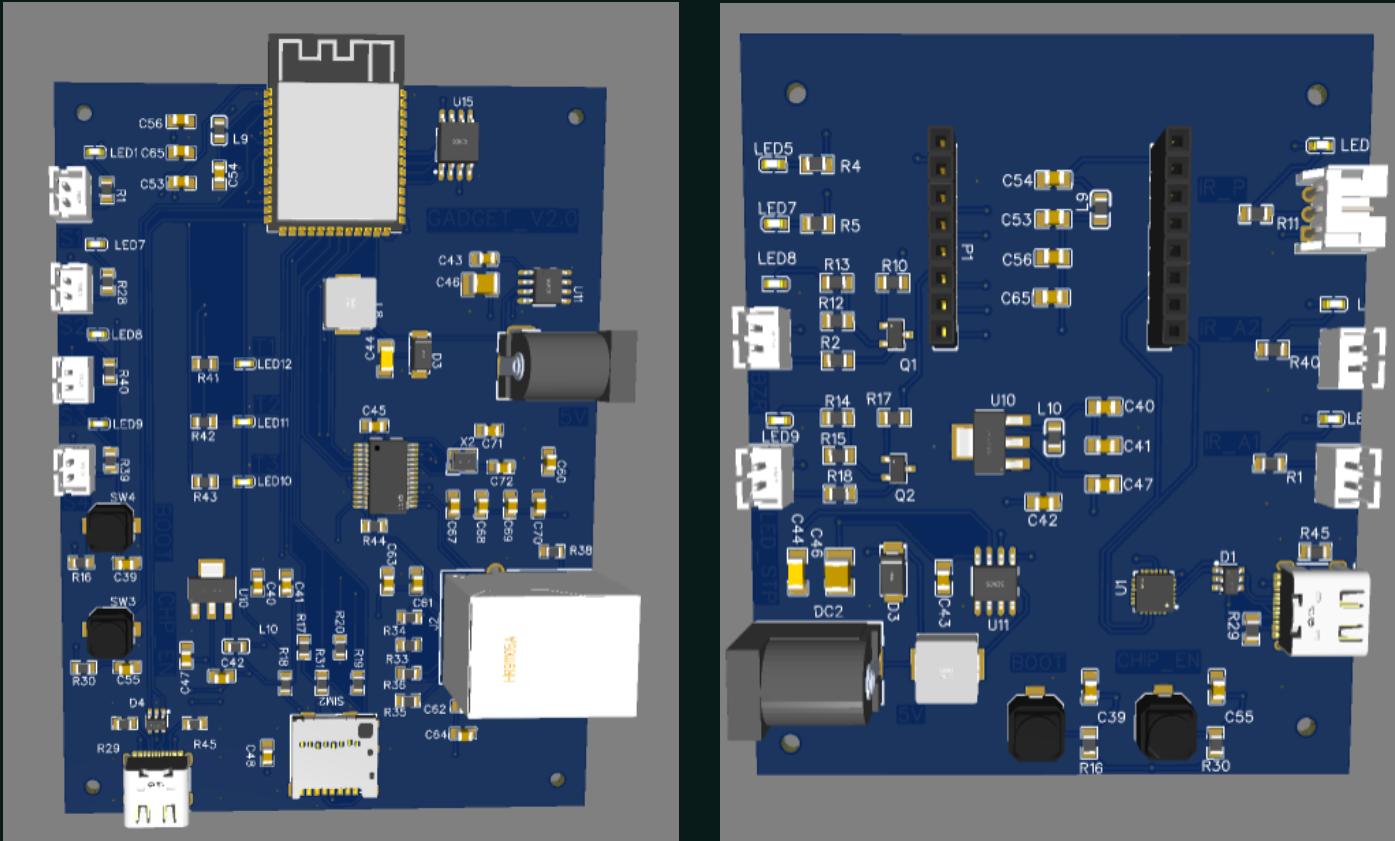
4 Layer Design

Top Layer – Important Signals

Inner layer 1 – Ground

Inner Layer 2 – Power

Bottom Layer – Other Signals



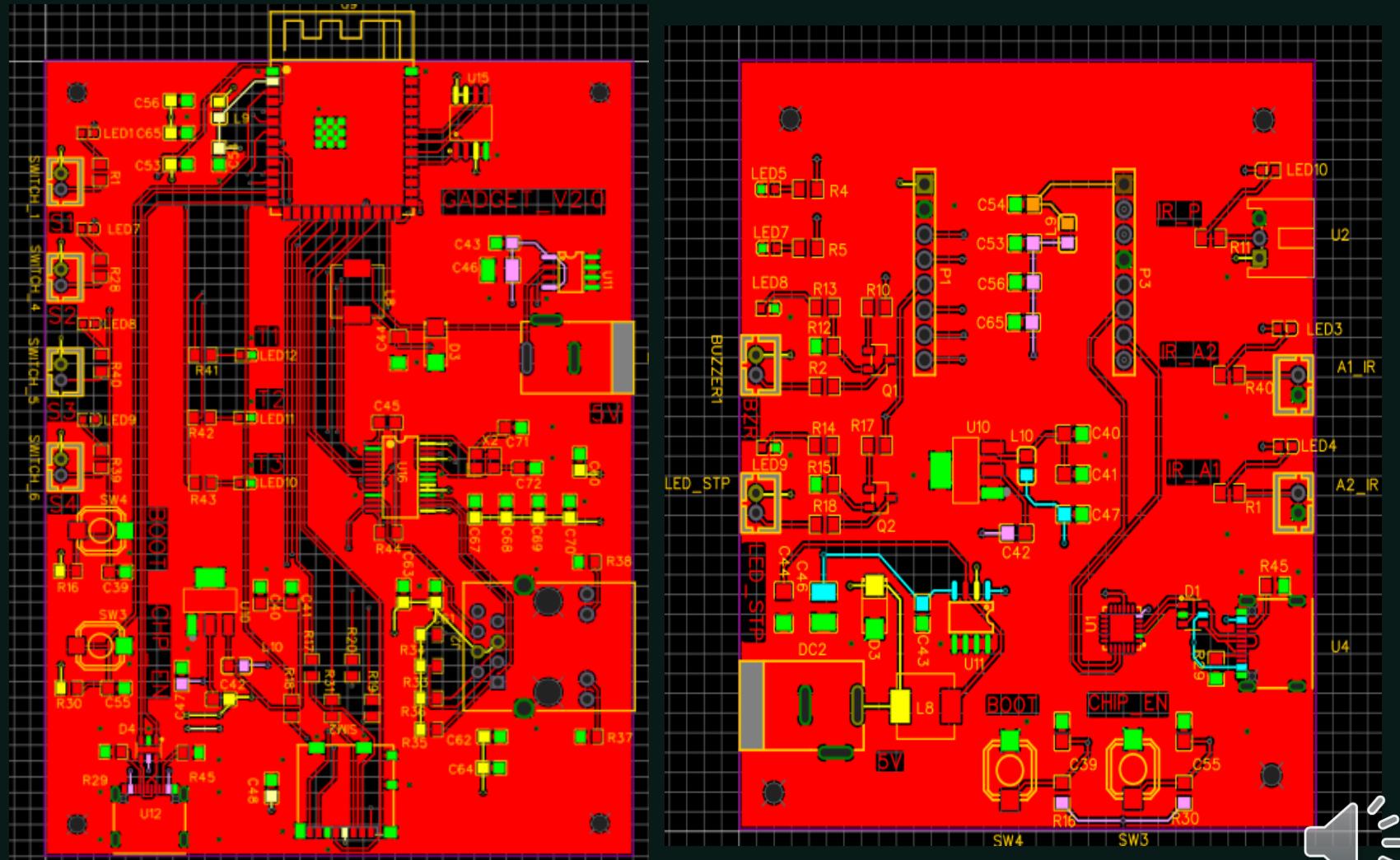


Dylan Myers
Electrical Engineering

Top Layer Data signals

Main Design Drivers

- Data signals are short as possible and are paired correctly
 - All Buttons and connectors are accessible
 - All components are replicable and repairable



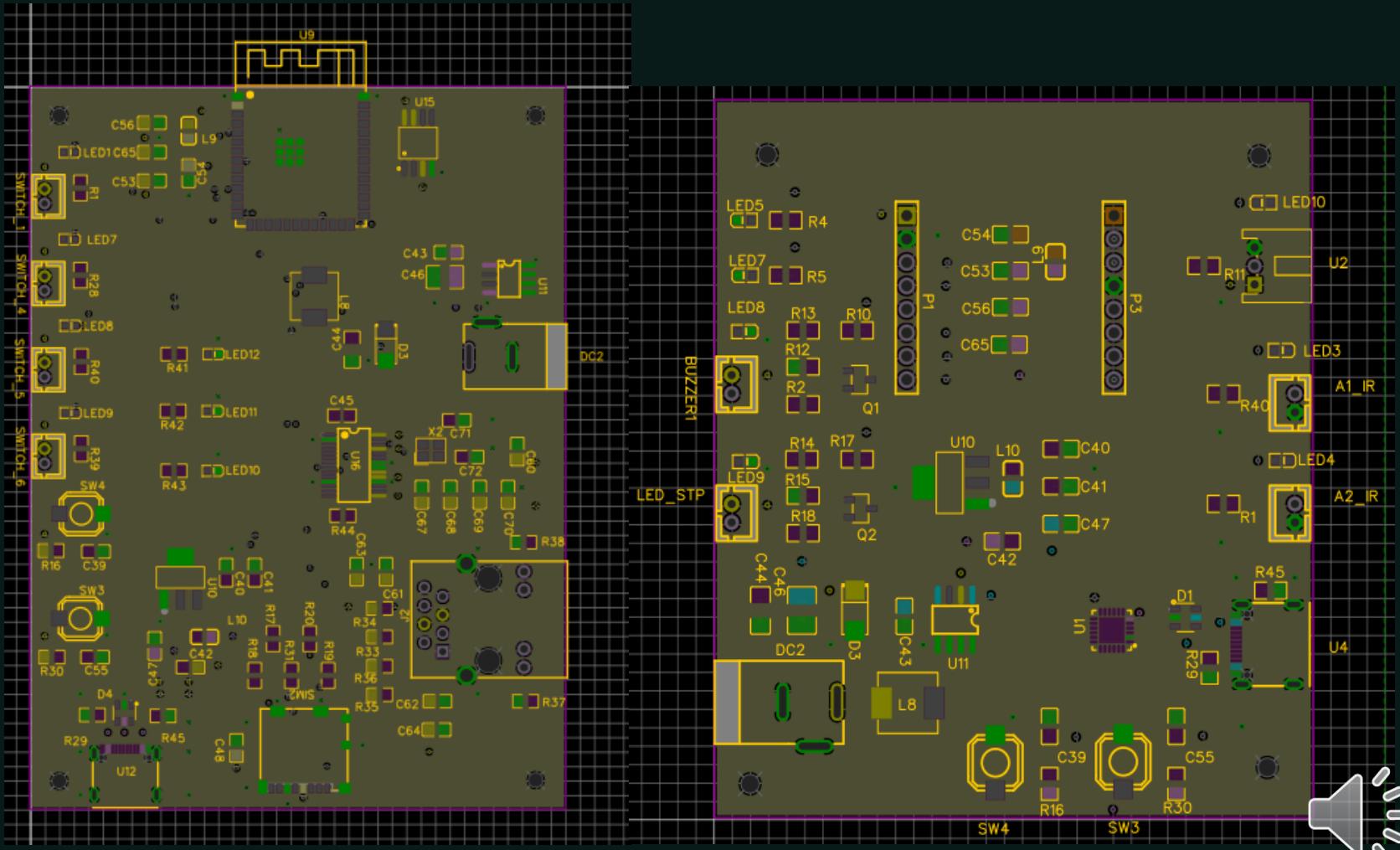


Dylan Myers
Electrical Engineering

Inner Layer 1 Ground

Main Design Drivers

- EMI reduction
- Design simplicity
- Low Ground Impedance





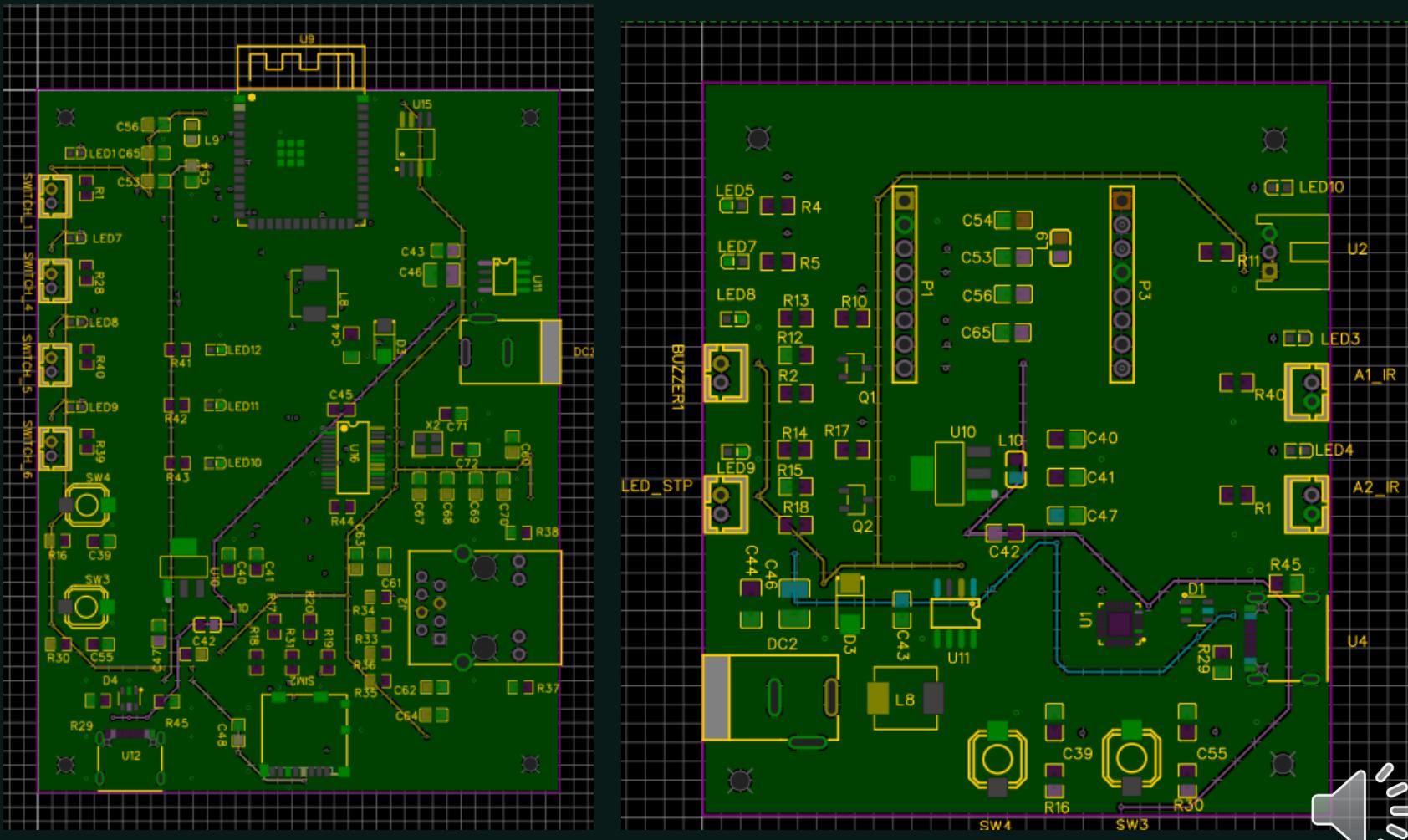
Dylan Myers
Electrical Engineering

Inner Layer 2

Power

Main Design Drivers

- EMI reduction
 - Design simplicity
 - Short Power paths,
limiting Voltage drop
 - Thermal management



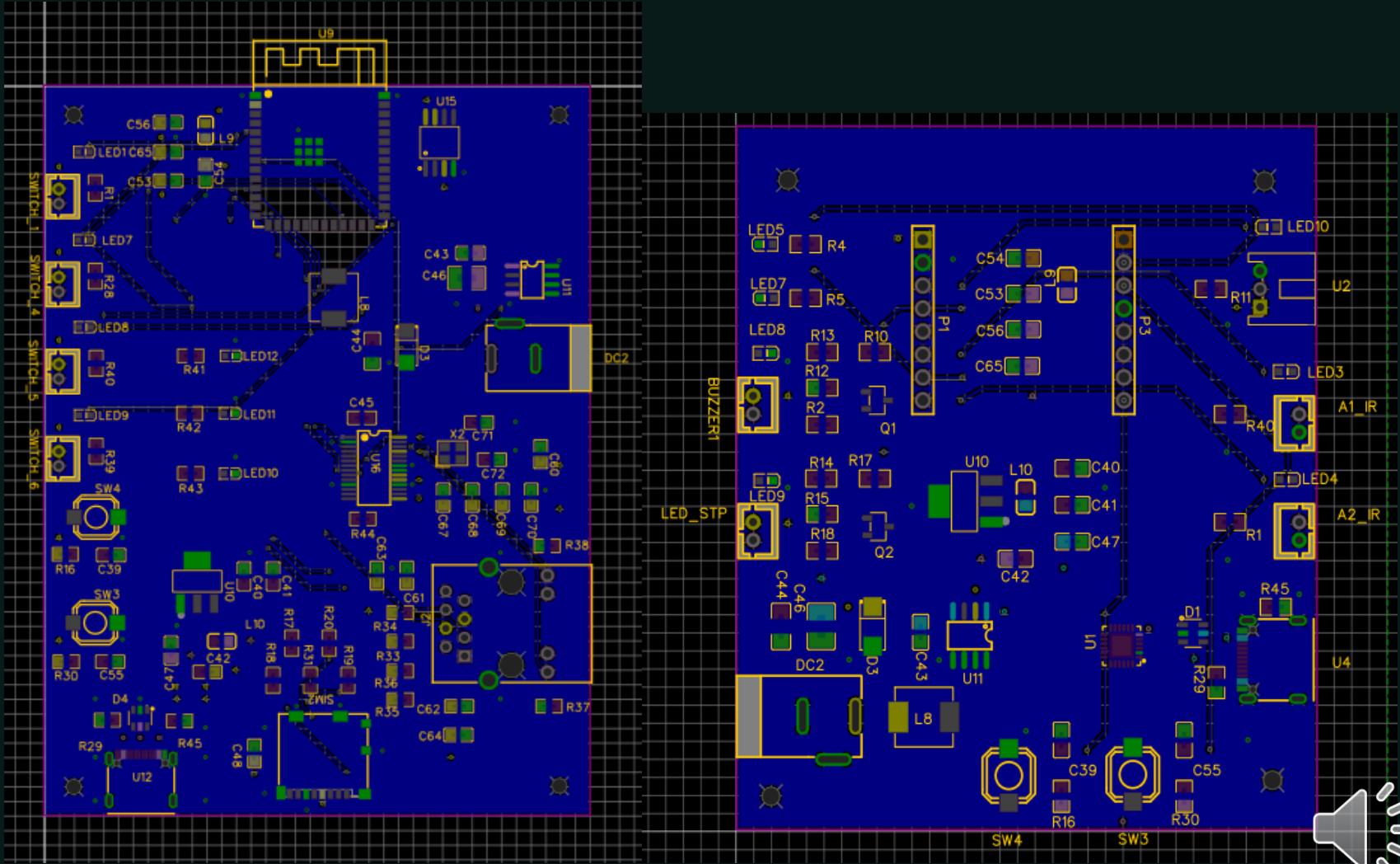


Dylan Myers
Electrical Engineering

Bottom Layer Other signals

Main Design Drivers

- EMI reduction
- Design simplicity





Jaxon Topel
Computer Engineering

Comparison & Selection of Software

- Jaxon Topel
- Colin Kirby





Jaxon Topel
Computer Engineering

Programming Languages Comparison

In terms of available programming languages for our project, we considered :

- C/C++
- Python
- Arduino C++
- JavaScript

The table compares their performance, control, and complexity for our use case.

Language	Performance	Control	Complexity
C / C++	High Efficiency & Low Memory Use.	Full HW control.	High Complexity
Python	Slower Execution.	Less HW control.	Low complexity, high usability.
Arduino C++	Slightly less efficient than C++.	Moderate control with HW libraries.	Simple syntax, beginner-friendly.
JavaScript	Slower execution.	Minimal HW interaction.	Low complexity.





Jaxon Topel
Computer Engineering

Programming Languages Selection



P Y T H O N

- Used for prototyping with extensive image processing libraries.
- Validates camera functionality and image processing before deploying C++.



A R D U I N O C + +

- Chose for embedded development on ESP32 boards.
- Offers HW control, efficient memory management, and access to open libraries.

C R I T I C A L R E Q U I R E M E N T S

- Easy to use.
- High performance.
- Low memory usage.

W H Y B O T H ?

- Python simplifies early testing.
- Arduino C++ ensures reliable operation.





Jaxon Topel
Computer Engineering

Image Processing Libraries Comparison

In terms of available image processing libraries for our project, we considered:

- OpenCV
- TensorFlow Lite
 - YoloV5
 - ESP-WHO

The table on the right compares their performance, power consumption, memory usage, for our use case.

Library	Performance	Power Consumption	Memory Usage
OpenCV	Great for general IP, not optimized for deep learning.	Moderate to High.	High Memory Usage.
TensorFlow Lite	Optimized for edge devices and object detection.	Low to Moderate.	Moderate, can be optimized for lower memory.
YoloV5	High performance for real-time obj detection.	Moderate to High.	High Memory Usage.
ESP-WHO	Optimized for ESP32, best for face detection.	Designed for Low Power.	Low to Moderate, optimized for ESP32 memory.



Jaxon Topel
Computer Engineering

Image Processing Libraries Selection



TENSORFLOW LITE (TF LITE)

- **Extensive Resources:** There is a wealth of online resources and libraries available, enabling us to tailor our machine learning model to suit our task.
- **Low Power and Memory Usage:** TF Lite is designed for efficiency, it ensures minimal power consumption and memory usage.
- **Seamless Integration:** Easily integrates with open-source tools enhancing our ability to increase our model's performance.
- **Edge Impulse:** Edge Impulse is a tool we will be using for easy data acquisition, training, and model deployment.

C R I T I C A L R E Q U I R E M E N T S

- Low power and memory usage.
- Compatibility with object detection goals.
- Support for pre-processing and detection.

W H Y T F L I T E ?

- Using TensorFlow Lite allows us to easily integrate a machine learning model into our system without taking too much power, memory, or other resources away from other tasks.



Jaxon Topel
Computer Engineering

Communication Protocols Comparison

For potential communication protocols, we evaluated :

- HTTP
- WebSockets

The table below compares their performance, power consumption, memory usage, and integrations suitability for our project.

Protocol	Performance	Power Consumption	Memory Usage	Integration
HTTP	Higher Latency, but adequate.	Moderate, short-lived connections.	Low to Moderate, Data Dependent.	Easy, widely supported with many libraries.
WebSocket	Low latency for real-time use.	Higher than HTTP.	Higher than HTTP.	Harder to implement, fewer libraries available.





Jaxon Topel
Computer Engineering

Communication Protocols Selection



H T T P

- Selected for its simplicity and compatibility with ESP32 Libraries.
- Low resource consumption aligns with project efficiency goals.
- Enables easy integration and reliable communication between the nodes and the gadget.

C R I T I C A L R E Q U I R E M E N T S

- Easy to implement with minimal resource consumption.
- Supports reliable data transfer for messages between our nodes, gadget, and mobile app.

W H Y H T T P ?

- Simplifies development while meeting current system performance needs.
- Leaves room for future scalability with more advanced protocols.





Colin Kirby

Computer Engineering

Mobile App Frameworks Comparison

We evaluated these mobile app frameworks for **SecureScape**:

- Flutter
- React Native
- Ionic
- Xamarin.

The table compares their performance, ease of use, cross-platform compatibility, and UI/UX flexibility.

Framework	Performance	Ease of Use	Cross-Platform Compatibility	UI / UX Flexibility
Flutter	High, near-native performance.	Moderate	Excellent (iOS, Android)	High, Customizable UI
React Native	High, Slower than Flutter	High, JavaScript-based	Excellent (iOS, Android)	High, Customizable UI
Ionic	Moderate, uses Web Tech	High, web-based development	Excellent (iOS, Android, PWA)	Moderate
Xamarin	High, near-native performance	Moderate, requires C#	Excellent (iOS, Android, Windows)	 High



Colin Kirby

Computer Engineering



FLUTTER

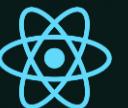
- Near-native performance for real-time image processing.
- Consistent cross-platform support (iOS, Android) with identical UI Code.
- Customizable UI with widget-based architecture.

C R I T I C A L R E Q U I R E M E N T S

- High performance and responsive design.
- Seamless cross-platform development.

W H Y N O T O T H E R S ?

React Native : Easy to use, but slower due to JavaScript bridging.



Ionic : Great for prototyping but lacks real-time performance.



Xamarin : High performance, but too complex and requires C#.





Colin Kirby

Computer Engineering

Mobile App Hosting Comparison

We evaluated these options for SecureScape :

- AWS Lambda
- Firebase
- Heroku
- Digital Ocean

The table compares their performance, scalability, cost, and ease of use.

Hosting Platform	Performance	Scalability	Cost	Ease of Use
AWS Lambda	High, serverless & event-driven	Automatic, scales with demand	Pay-per-use, cost-efficient	Moderate
Firebase	Moderate, real-time optimized	High, auto-scaling	Free for limited use	Easy, minimal setup required
Heroku	Moderate, simplified deployment	High, limited compared to AWS	Free tier, higher costs for scaling	Easy, user-friendly UI
DigitalOcean	High, manual server setup	High, manual scaling options	Low-costs, starts at \$5 / month	Moderate, requires manual set up



Colin Kirby

Computer Engineering

Mobile App Hosting Selection



A W S L A M B D A

- **Scalability** : Automatically scales with demand.
- **Event-Driven** : Ideal for handling intermittent workloads.
- **Cost Efficiency** : Pay-per-use ensures affordable operation.
- **Minimal Management** : Serverless Architecture.

C R I T I C A L R E Q U I R E M E N T S

- High responsiveness for real-time applications.
- Seamless scaling to meet growing user demand.
- Streamlined development with minimal management.

W H Y N O T O T H E R S ?



Firebase : Simple setup but less suitable for resource intensive-tasks.



Heroku : Easy to use but lacks scalability for large-scale deployment.



DigitalOcean : Flexible but requires manual setup.





Colin Kirby

Computer Engineering

Mobile App Database Comparison

We evaluated these options for SecureScape :

- Firebase Firestore
- MongoDB Atlas
- PostgreSQL
- Supabase

The table compares their performance, scalability, ease of use, and security.

Database	Ease of Use	Performance	Scalability	Security
Firebase Firestore	High	High	High	Built-in Authentication
MongoDB Atlas	Moderate	High	High	Requires external Auth
PostgreSQL	Moderate	High	High	Secure, but complex set up
Supabase	High	High	High	Built-in Authentication





Colin Kirby

Computer Engineering

Mobile App Database Selection



FIREBASE FIRESTORE

- **Real-Time Sync** : Keep data consistent across devices and platforms.
- **Scalable** : Handles more nodes and workloads automatically.
- **User-Friendly** : Easy setup and integration simplify development.
- **Secure** : Built-in Authentication reduces effort and ensures data safety.

CRITICAL REQUIREMENTS

- Efficiently manages logs, alerts, and sensor data.
- Ensure responsive behavior for real-time alerts.
- Minimal backend expertise required for integration.

WHY NOT OTHERS?



MongoDB Atlas : Flexible but additional effort for authentication setup.



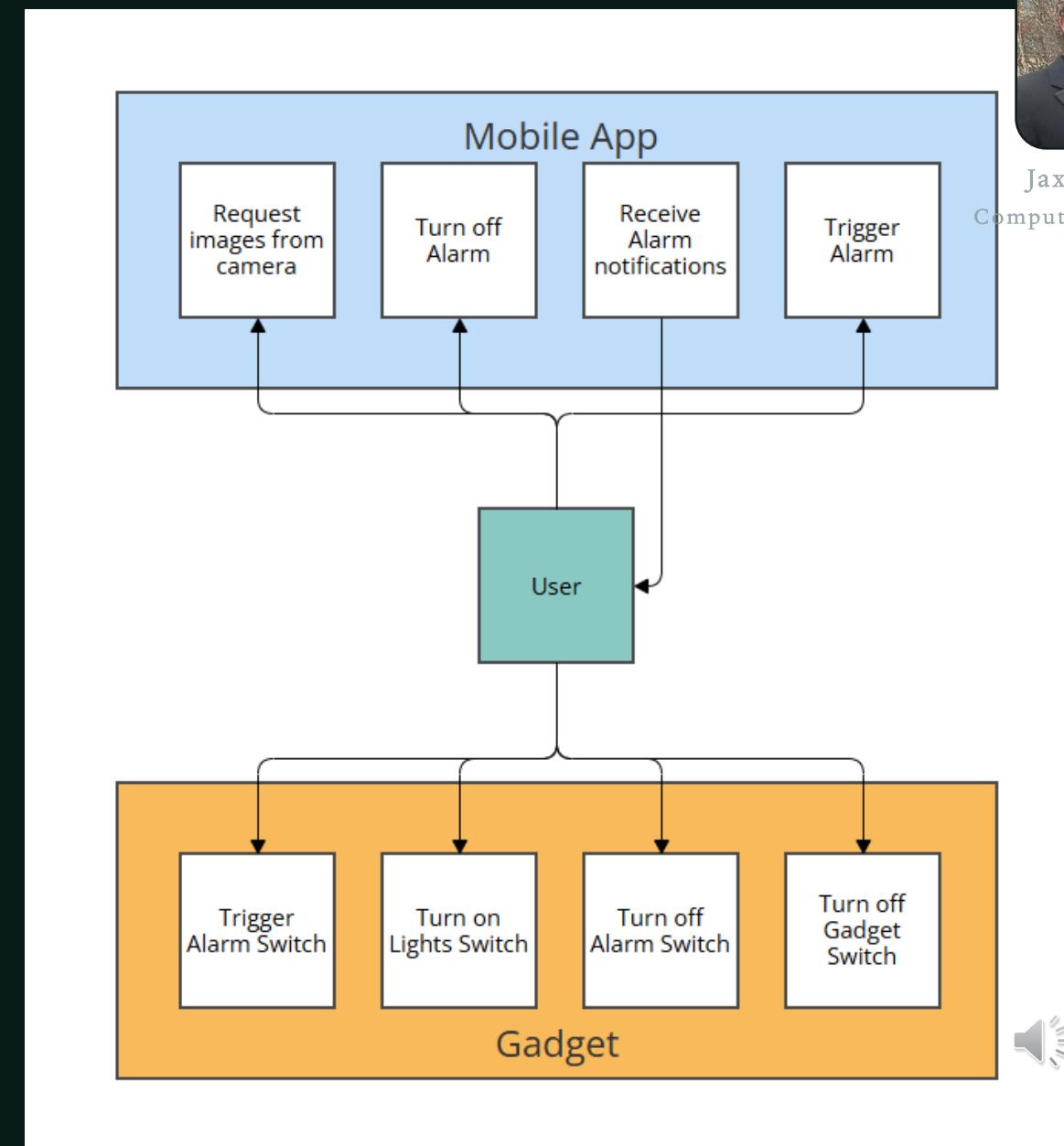
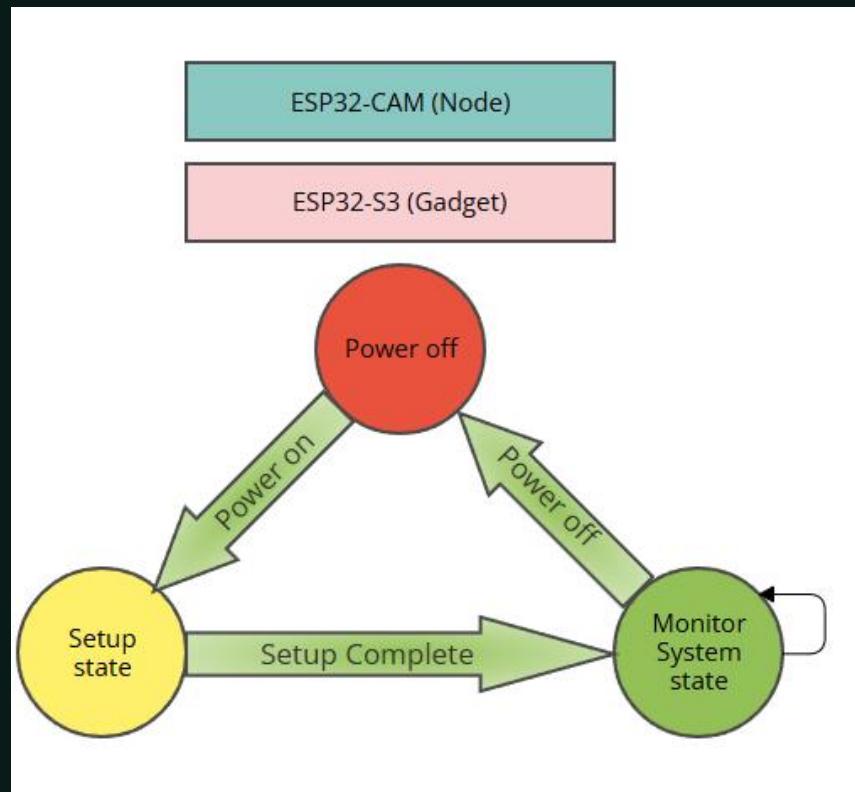
PostgreSQL : Powerful and versatile but too complex for current needs.



Supabase : Provides real-time features but similar to PostgreSQL complexity.



Software State & Use Case Diagrams



Jaxon Topel
Computer Engineering





Backend Class Diagram

ESP32-CAM (Node)

```
main.ino
- setup()
- loop()
- captureAndSendImage()
- handleJpgLo()
- handleJpgMid()
- handleJpgHi()
- checkAlarmNotification()
- checkForTurnOnLights()
- checkForTurnOffLights()
- triggerAlarm()
- sleep()
```

ESP32-S3 (Gadget)

main.ino

- setup()
- loop()
- handleImageUpload()

detectPerson()
called

ObjectDetection.cpp

- initializeModel()
- detectPerson()

Model Instatiated

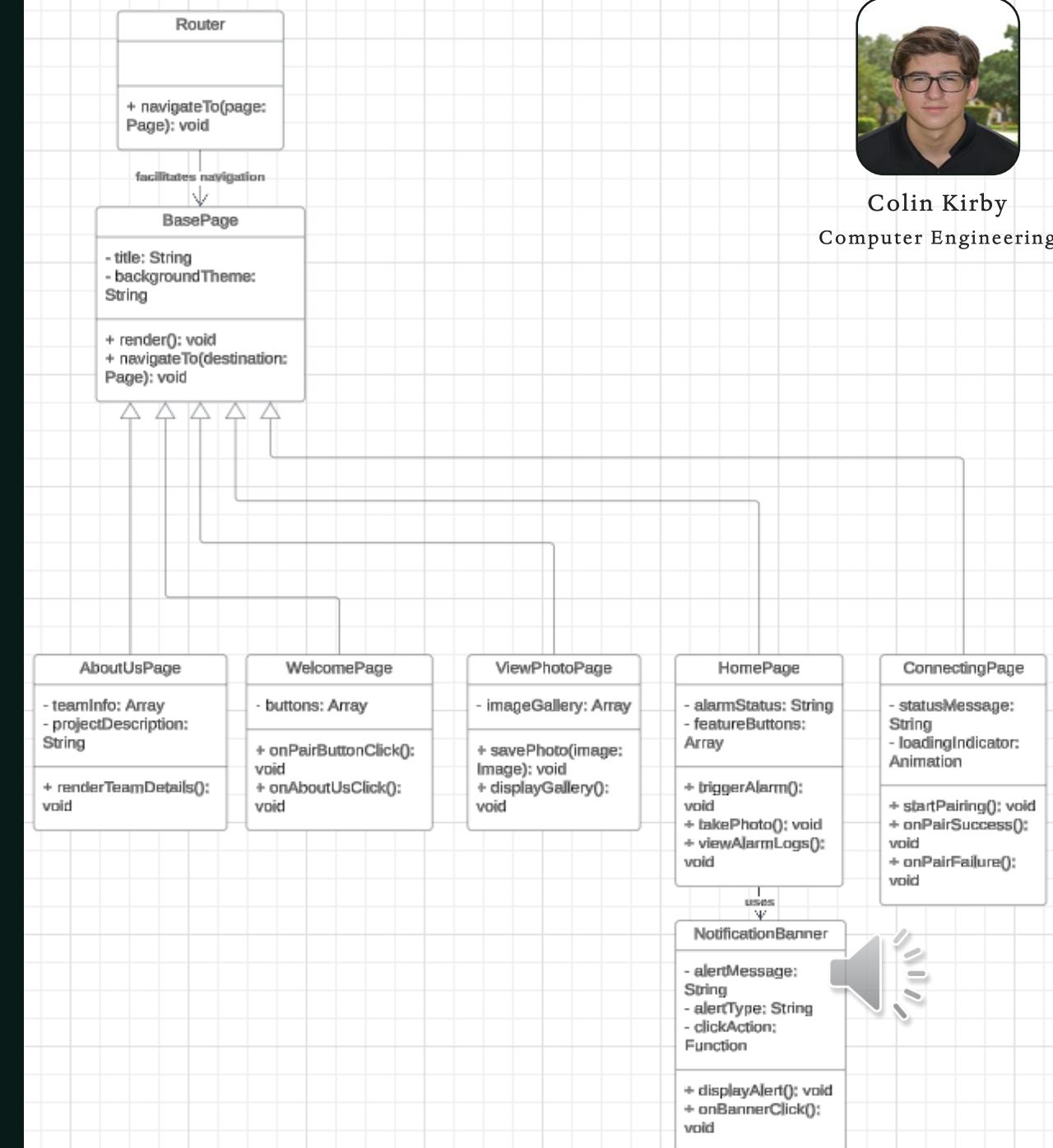
person_detect_model_data.cpp
person_detect_model_data.h

Colin Kirby
Computer Engineering

This diagram details the functions of our **ESP32-CAM (Node)** and **ESP32-S3 (Gadget)**, where the node captures and transmits images, and the gadget processes them using an object detection model to identify people.

Frontend Class Diagram

- Router : manages navigation between pages.
- BasePage : provides common properties like title.
- HomePage : controls alarms and logs.
- AboutUsPage : displays team and project details.
- WelcomePage : handles navigation buttons.
- ViewPhotoPage : manages image storage and display.
- ConnectingPage : handles device pairing status.
- NotificationBanner : shows alerts and warnings.

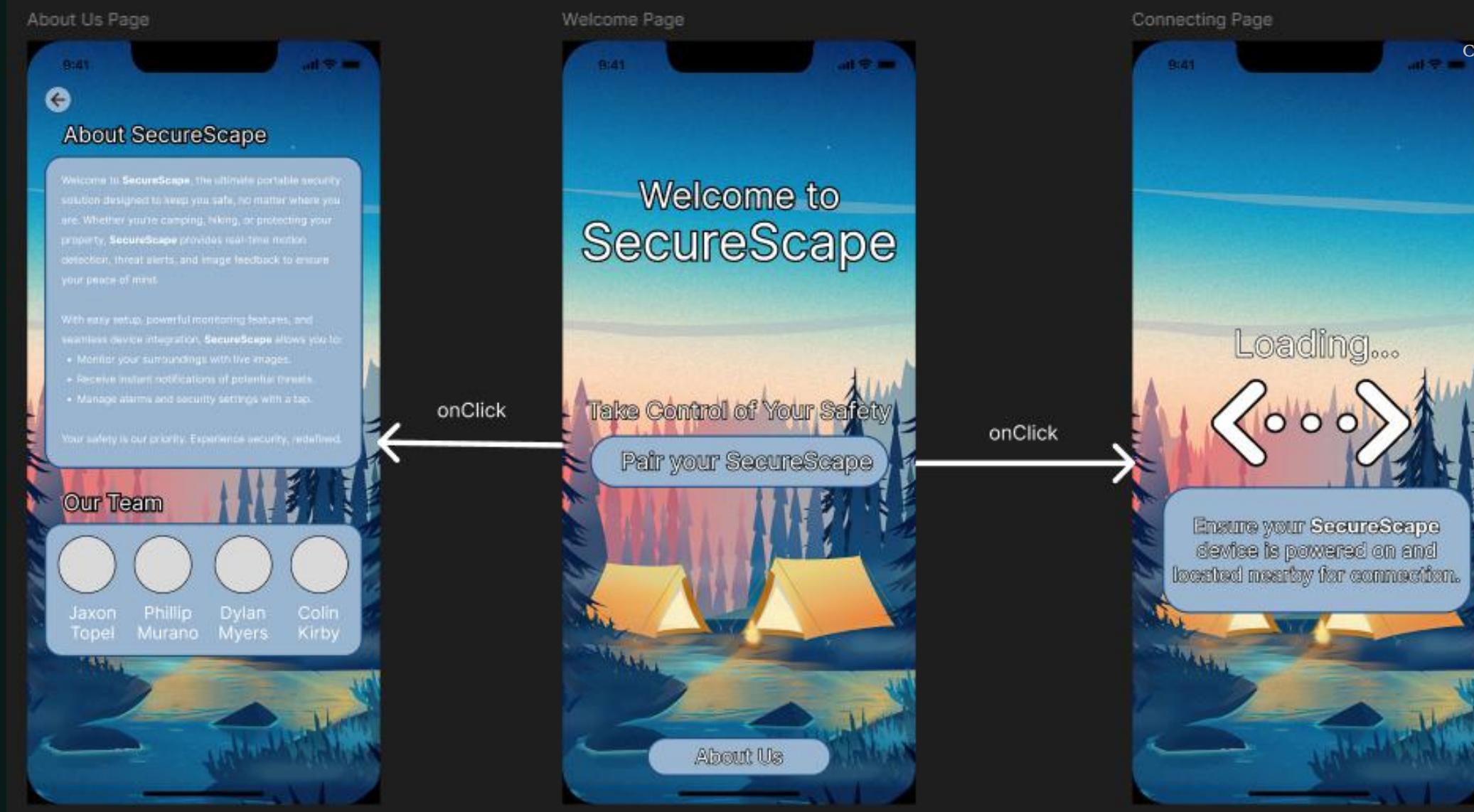


Colin Kirby
Computer Engineering

Mobile App UI - Part 1



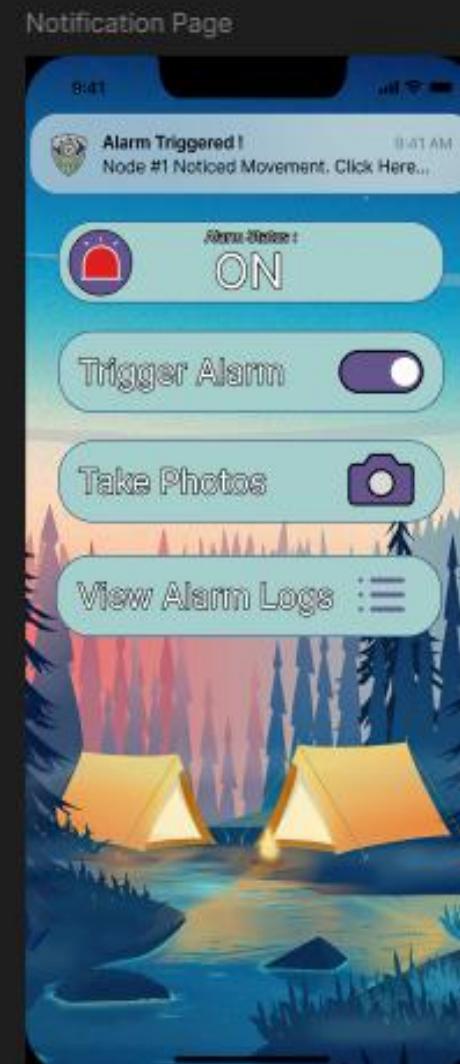
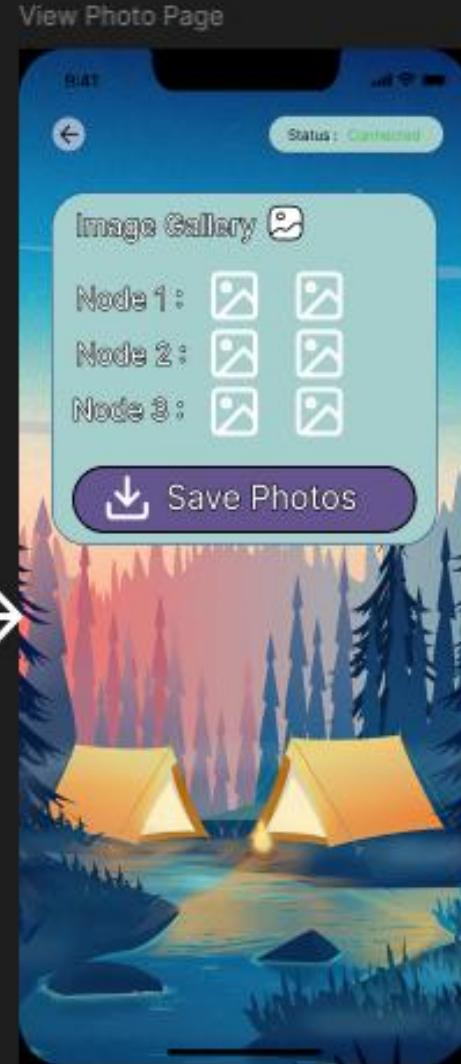
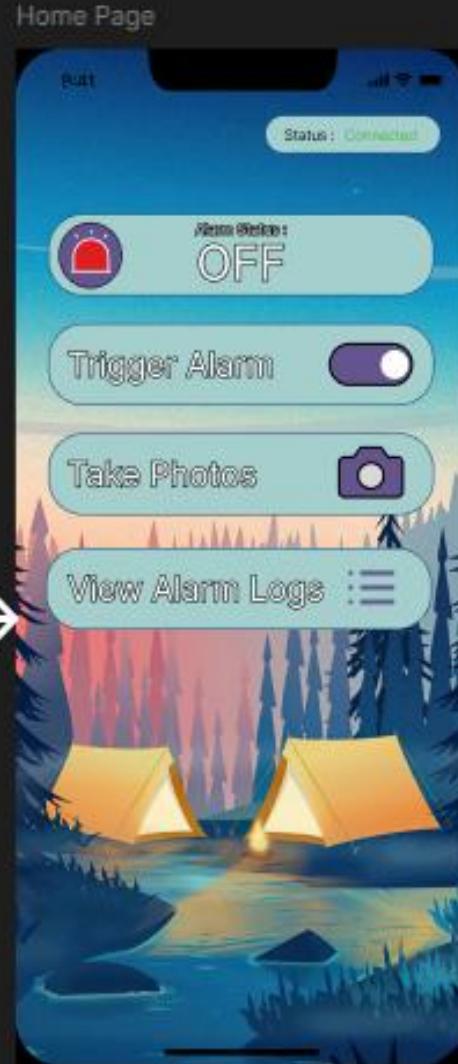
Colin Kirby
Computer Engineering



Mobile App UI - Part 2



Colin Kirby
Computer Engineering





Colin Kirby
Computer Science



Prototyping & Testing





Colin Kirby

Computer Engineering

Software Testing Breakdown

- Software testing for our project will focus on 5 key areas :
 1. **Embedded Code Tests** : Verify correct behavior of ESP32 Devices.
 2. **Communication Network Tests** : Ensure stable data transfer over HTTP.
 3. **Image Processing Algorithm Tests** : Validate accuracy of IP.
 4. **AUR (All Up Round) System Test** : Test system integration without PCB.
 5. **Frontend Tests** : Check responsiveness of mobile app.





Phillip Murano
Electrical Engineering

Embedded/Hardware Peripherals Test

- **ESP32-CAM (Node) & ESP32-S3 (Gadget)** : The software tests are categorized between Node-specific and Gadget-specific tests.
- **Hardware Integration** : The embedded code that will live on our MCU will be in constant communication with our hardware peripherals such as the Passive IR sensor, Switches, and Camera.
- **Testing Approach** : Most of the tests for the embedded code will involve reading the GPIO pin values and verifying successful hardware connections via the serial monitor.





Phillip Murano
Electrical Engineering

Node Peripheral Tests

TEST ID	TEST CASE	EXPECTED OUTCOME	PASS/FAIL CRITERIA	PASS/FAIL	COMMENTS
1	Passive IR Sensor	Passive IR sensor success message observed in serial monitor.	Must see a success message for reading passive IR sensor data in serial monitor.	PASS	We can see monitor detected motion through feedback from the serial monitor.
2	Camera	Success message for capturing and storing image.	Must see success message for image being captured in serial monitor.	PASS	Images were able to be captured and stored correctly.
3	LED	Led Success message seen in serial monitor and visibly seeing lights turn on.	Must see lights turn on and successful serial monitor message.	Not Yet Tested	





Phillip Murano
Electrical Engineering

Gadget Peripheral Tests

TEST ID	TEST CASE	EXPECTED OUTCOME	PASS/FAIL CRITERIA	PASS/FAIL	COMMENTS
1	Triggering Alarm Switch	Serial monitor success message and alarm being turned on.	Must see serial monitor success message and alarm being turned on.	Not yet tested	
2	Turning Off/On Light's Switch	Serial monitor success message and lights turning on/off	Must see serial monitor success message and lights being turned on and off.	Not yet tested	
3	Turning Off Alarm Switch	Serial monitor success message and alarm being turned off	Must see serial monitor success message and alarm being turned off.	Not yet tested	





Colin Kirby
Computer Engineering

Communication Network Tests

TEST ID	TEST CASE	EXPECTED OUTCOME	PASS/FAIL CRITERIA	PASS/FAIL	COMMENTS
1	Gadget Connection	Serial monitor gadget connection to Wi-Fi success message	Gadget successfully connects to router	PASS	Gadget successfully connects to router
2	Node Connection	Serial monitor node connection to Wi-Fi success message	Node successfully connects to router	PASS	Node successfully connects to router
3	Mobile to Gadget Connection	HTTP requests can be sent and received from mobile app to gadget	Messages successfully sent from mobile app to gadget	PASS	Mobile App properly sends HTTP request to Gadget.
4	Node to/from Gadget Connection	Node can send/receive HTTP messages from/to the gadget	Messages being sent successfully between node and gadget	PASS	Messages are successfully being sent between node and gadget



Colin Kirby

Computer Engineering

Mobile Application UI Tests

TEST ID	TEST CASE	EXPECTED OUTCOME	PASS/FAIL	COMMENTS
1	Welcome Page buttons	Buttons provide correct navigation to corresponding pages.	PASS	Navigation through UI is successful
2	Connection Page Transition	Device pairing transitions to home page or shows errors	PASS	Prompts connection to Micro router.
3	Home Page Button	Buttons trigger correct backend actions	Not Yet Tested	
4	About Us Page Rendering	Team info and description displayed properly	PASS	Can successfully view about us page
5	UI Response	UI has "active" feel for user-friendliness	PASS	CSS has been implemented to ensure users know status of connection and app
6	Real Time Updates	Status updates immediately on home page	PASS	Notifications are passed on Real-Time.
7	View Photo Functionality	Views photo from mobile app	PASS	The Capture Request for Images properly functions.



Jaxon Topel
Computer Engineering

Image Processing Algorithm Test Breakdown

- **Detection Accuracy** : The algorithms success will be determined by achieving at least 60% accuracy in person detection.
- **Testing Breakdown** : Four test scenarios evaluate performance under different conditions.
- **Environmental Validation** : Evaluations include, static, dynamic, low-light, and obstructed conditions.

Test Scenarios :

1. **Static Environment Detection** : Evaluates detection in a controlled, well-lit room with no background distractions.
2. **Dynamic Environment Detection** : Assesses robustness in moving backgrounds, such as wind or moving objects.
3. **Low Light & Shadows** : Test performance under dim lighting and directional shadows.
4. **Obstruction & Partial Visibility** : Determines if algorithm can detect partially visible individuals (e.g. behind objects)





Jaxon Topel
Computer Engineering

System Integration Test - AUR (All Up Round) – No PCB

- **Objective** : Evaluate the system's overall functionality and identify potential areas for improvement by integrating all components.
- **Description** : A target person stands in the center of the frame under ideal lighting and moderate background activity. Upon detecting motion :
 1. Passive IR sensor will detect motion, image will be captured.
 2. Once image captured, ESP32-CAM uses tensorflow lite to perform person detection.
 3. If a person is detected, a mobile app notification alerts the user of the threat and gadget sends messages via http to nodes to sound the alarm.
- **Success Criteria** : Seamless operation of all components and accurate notifications for detected threats.





Colin Kirby

Computer Engineering



Administrative Content





Phillip Murano
Electrical Engineering

General Project Budget

The project budget is capped at \$1000, encouraging strategic and efficient selection of components to balance performance and cost-effectiveness.

SECTION	BUDGET
MCUs	~ \$100.00
Cameras	~\$100.00
IR sensors, Alarms, LEDs	~ \$100.00
Power	~ \$150.00
Memory	~ \$50.00
Router	~ \$100.00
Housing Fabrication	~ \$200.00
PCB Fabrication	~ \$150.00
Wires/ Switches/ Misc.	~\$50.00
Total	~ \$1000.00



Bill Of Materials

Dylan Myers
Electrical Engineering



Part	Suggested Part	Quantity	Cost / per	Cost Total (\$)
ESP32 CAM	DORHEA	1	22.99	22.99
ESP32 S3	DORHEA	1	23.42	23.42
Gadget batteries	PAOWANG	1	19.99	19.99
Node batteries	POAWANG	1	19.99	19.99
PCB Node Build	TBD	~3	~29.99	89.97
PCB Gadget Build	TBD	~1	~29.99	29.99
Passive IR	STEMEDU	1	9.98	9.98
Active IR	GIKFUN	1	5.98	5.98
LEDs	CHANZON	1	5.31	5.31
Buzzer	WEICHUANG	3	5.49	5.49
Housing nodes	TBD	3	~24.99	74.97
Housing gadget	TBD	1	~24.99	24.99
SSD	SanDisk 256GB	1	25.44	25.44
Micro Router	GL.iNET	1	31.90	31.90
Gadget Switch	Same Sky	10	.53	5.30
Wires	Sunxeke	1	9.99	9.99
Total Cost				\$ 405.70



Dylan Myers
Electrical Engineering

Progress & Future Milestones

TASK	DEADLINE	STATUS
Image Processing detecting people	1/23/2025	Incomplete
All Hardware Parts Ordered	2/1/2025	Complete
Software Development Complete	2/28/2025	Incomplete
Software Integration with Hardware	3/1/2025	Incomplete
Testing and Revision Phase 1	3/1/2025	Incomplete
Testing and Revision Phase 2	3/15/2025	Incomplete
Testing and Revision Phase 3	4/1/2025	Incomplete
System Integration Complete	4/4/2025	Incomplete
Project Completion	4/4/2025	Incomplete





Phillip Murano
Electrical Engineering

Work Distribution



Phillip Murano
Electrical Engineering



Dylan Myers
Electrical Engineering



Jaxon Topel
Computer Engineering



Colin Kirby
Computer Engineering

Task Description	Primary	Secondary
Frontend Development	Colin Kirby	Jaxon Topel
Embedded Code Development	Jaxon Topel	Colin Kirby
Device Communication	Jaxon Topel	Colin Kirby
Image Processing Algorithm	Jaxon Topel	Colin Kirby
Website Development	Colin Kirby	Jaxon Topel
CAD Development	Phillip Murano	Dylan Myers
Device Power Management	Phillip Murano	Dylan Myers
Node Peripherals	Dylan Myers	Phillip Murano
PCB Design	Dylan Myers	Phillip Murano
Gadget Peripherals	Dylan Myers	Phillip Murano



Thank You!

Jaxon Topel

Dylan Myers

Colin Kirby

Phillip Murano

Questions?



Colin Kirby
Computer Engineering

