SecureScape

GROUP 15

FINAL PRESENTATION

Introduction & Our Group

Jaxon Topel
Computer Engineering

- **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 a portable node with IR sensors & Camera.
- 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.

Specifications

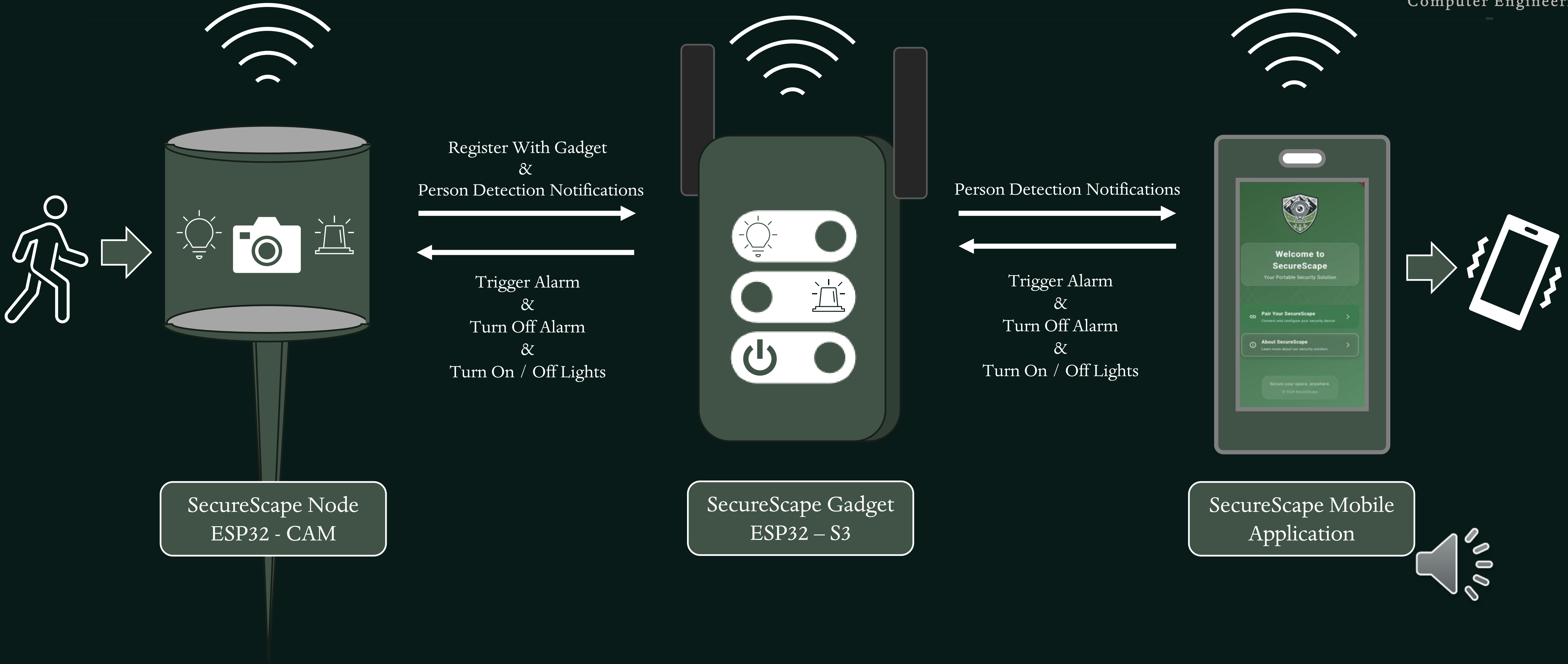
Phillip Murano
Electrical Engineering

☆ = Demonstratable Specifications

Metric	Guarantee	Relevance to project
☆ Detection Accuracy	60% detection	Essential for reliable threat identification.
Image Processing Time	3 seconds	Topic one Critical for timely classification of threats.
☆ Response Time	10 seconds	Important for timely notifications and threat management.
Deployment Time	5 minutes	Ensures a focus on quick set up in remote locations.
☆ Detection Range	5 feet	Critical for effective coverage of a secure area.
Mobile App Latency	3 seconds	Important for smooth UX when controlling the system.

Quick Project Overview

Colin Kirby
Computer Engineering



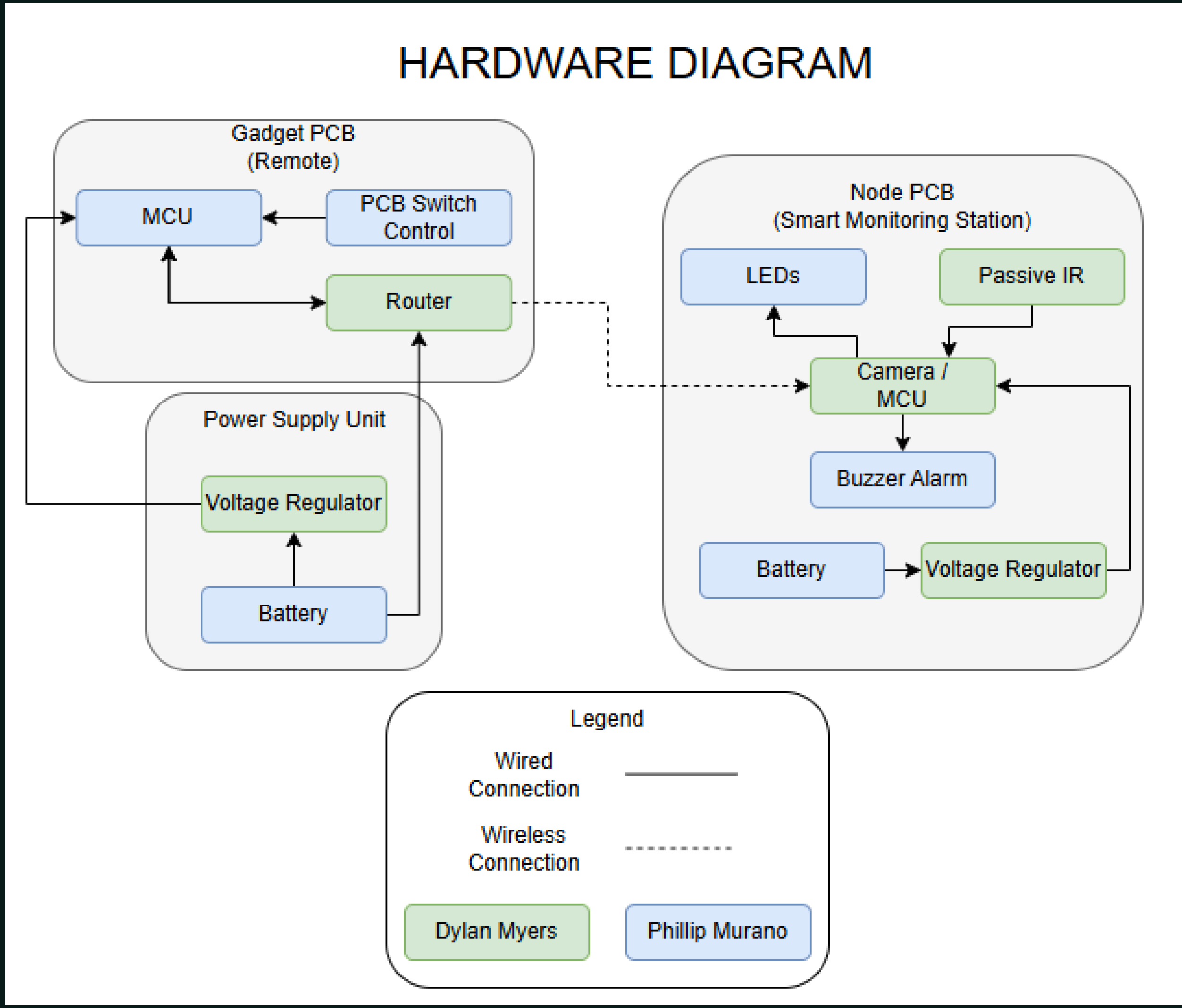
Hardware Diagram

Phillip Murano
Electrical Engineering

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.

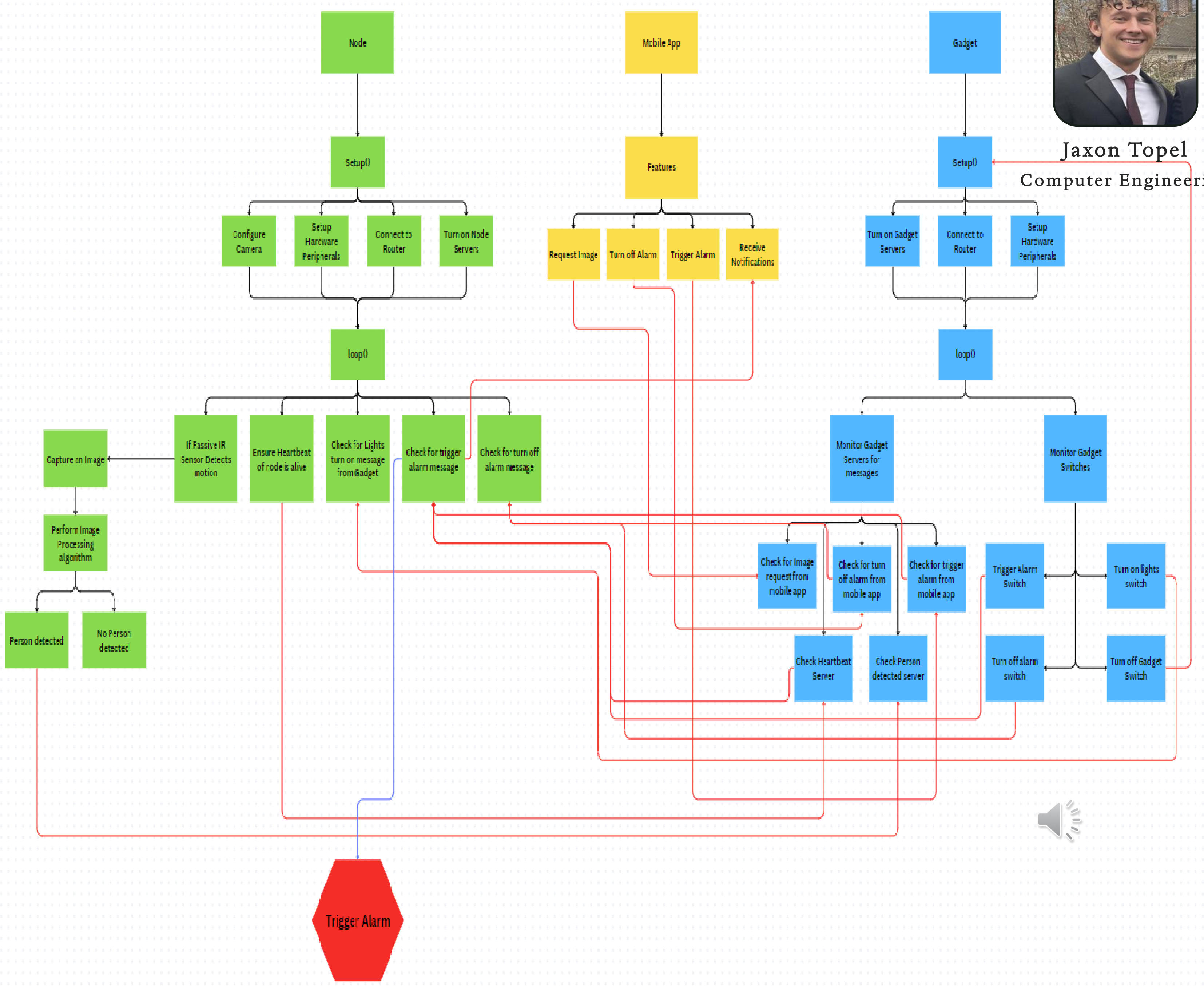


Software Design Flowchart

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

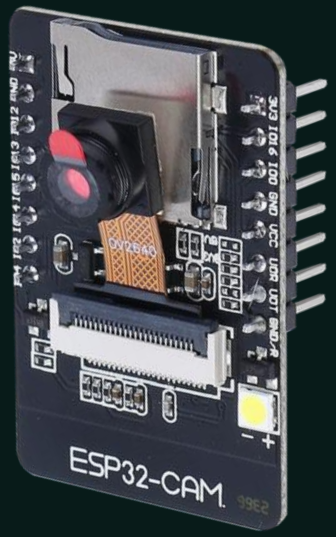
Comparison & Selection of Hardware

- Phillip Murano
- Dylan Myers

Camera Comparison – ESP32-CAM

Phillip Murano
Electrical Engineering

Category	ESP32-CAM	Raspberry Pi 4 B 3 B+ Camera Module	WayPonDEV CanMV-K230
Unit Cost	$7.67	$43.47	$25.99
Video Resolution	640 x 480	1920 x 1080	3840 x 2160
Power Draw (W)	~ 0.86 W	2.85 W	0.19 – 0.59 W
Board Size (in)	4.37 x 3.7 x 1.14	3.54 x 3.54 x 0.39	3.35 x 2.2 x 0.3
IP Rating	N/A	N/A	67
CRI (Index)	N/A	N/A	85



LED Comparison – Samsion

Phillip Murano
Electrical Engineering

Category	Samsion	CHNMALITAI	KXZM
Unit Cost	$7.00	$10.00	$1.00
Voltage (V)	5.0	5.0	5.0
LEDs Density (LEDs/m)	320	320	320
Power Draw (W)	6.0	5.0	5.0
IP Rating	67	65	65
CRI Index	85	90	93

Alarm Comparison - WEICHUANG

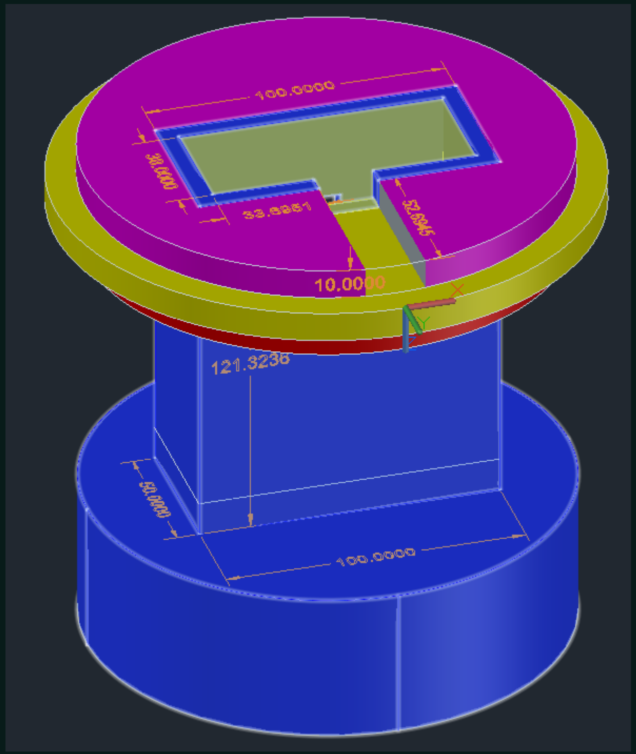
Phillip Murano
Electrical Engineering

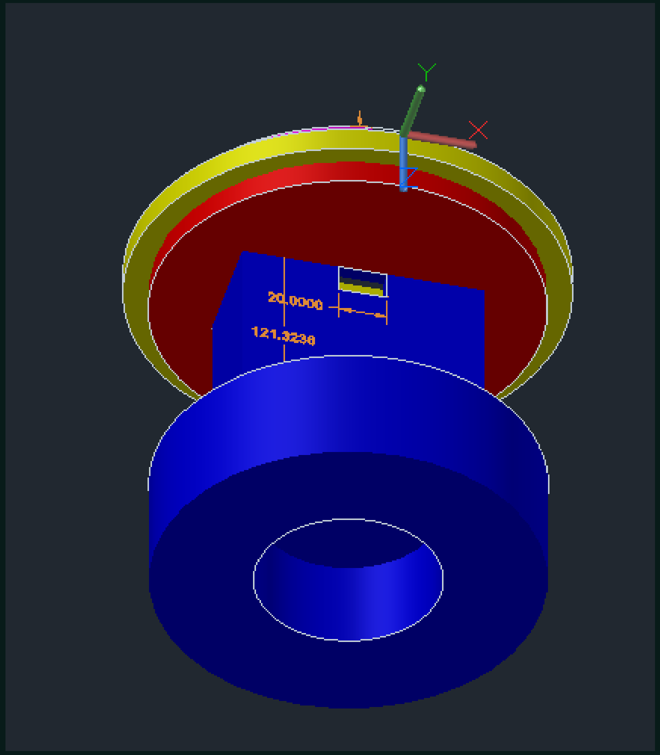
Category	WEICHUANG	BNYZWOT	Jameco
Unit Cost	\$5.49 (Pack of 3)	\$8.39 (Pack of 10)	\$7.90 (Pack of 2)
Voltage Range (V)	3.0 - 24.0	3.0 - 24.0	4.0 - 16.0
Rated Voltage (V)	12.0	12.0	12.0
Rated Current (mA)	15.0	12.0	25.0
Sound Output (dB)	100.0	85.0	95.0
Frequency (Hz)	3300 ± 500	3300 ± 500	2900

Node Structural Design

Phillip Murano
Electrical Engineering

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

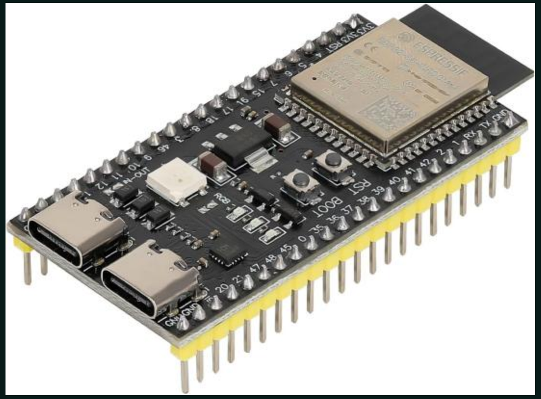




MCU Comparison – ESP32-S3

Phillip Murano
Electrical Engineering

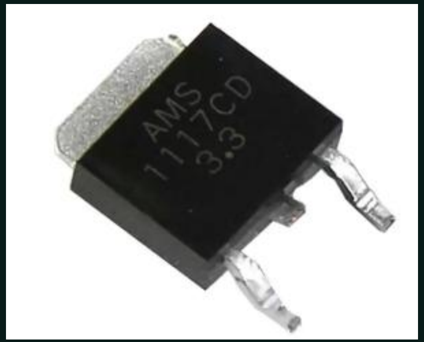
Part	ESP32-S3	STM32WBA5xx	PIC32MZXW1
Unit Cost	$5.77	$4.92	$9.41
Processor	Dual core LX7 @ 240MHz	Cortex-M33 @ 100MHz	MIPS32 M-Class @ 200MHZ
Memory (RAM/FLASH)	8MB / 16MB	128KB / 1MB	512KB / 2MB
Voltage (V)	3.0 - 3.6	1.71 - 3.6	2.97 - 3.63
GPIO Pins	36	20 - 35	37 - 62
Connectivity	Wi-Fi + Bluetooth	Wi-Fi + Bluetooth	Wi-Fi

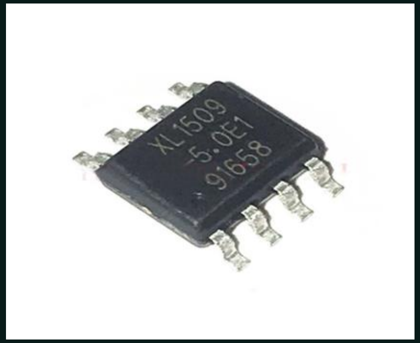


Voltage Regulator Comparison – For ESP32-S3

Phillip Murano
Electrical Engineering

Category	AMS1117-3.3V	TPS65023	TPS65090	XL1509-5.0E1
Input Voltage Range (V)	4.75 – 12	2.5 - 6	6 - 17	7.0 - 40
Output Voltage Range (V)	3.3	0.6 - 6	1 - 5	5.0
Max Current Out (A)	1.0	1.7	5.0	2.0
Quiescent Current (µA)	10	85	30 - 110	2.0 - 10
Step-Down converters	0	3	3	0
LDO Regulators	1	3	2	1
Unit Cost	$0.68	$3.29	$10.85	$1.21





Router Comparison – For ESP32-S3 Networking

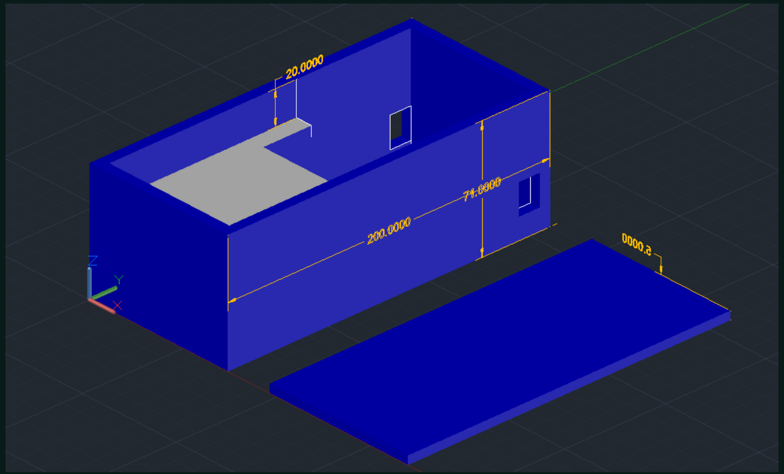
Phillip Murano
Electrical Engineering

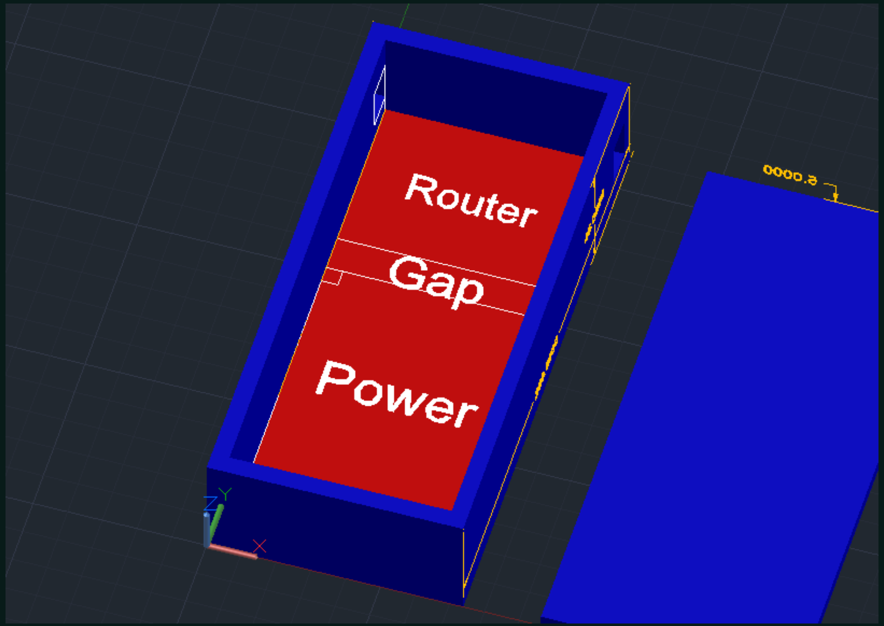
Category	GL.iNet GL-SFT1200 (Shadow)	TP-Link AC750	GL.iNet GL-AR300M16-Ext Portable
Unit Cost	$34.90	$33.99	$36.90
CPU	SF19A28, Dual-Core @ 1GHz	IPQ4018, Quad Core @ 717Mhz	QCA9531, @ 650MHz SoC
Dimensions (in)	4.65 x 3.35 x 1.18	2.91 x 0.87 x 2.64	2.28 x 2.28 x 0.98
EAP Support	No	Yes	Yes
Memory (RAM)	DDR3 128MB	DDR3L 128MB	DDR2 128MB
Storage	SPI NAND Flash 128MB	-	FLASH 16MB
Wi-Fi Speed	Max. 300Mbps	Max. 400Mbps	300Mbps

Gadget Structural Design

Phillip Murano
Electrical Engineering

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

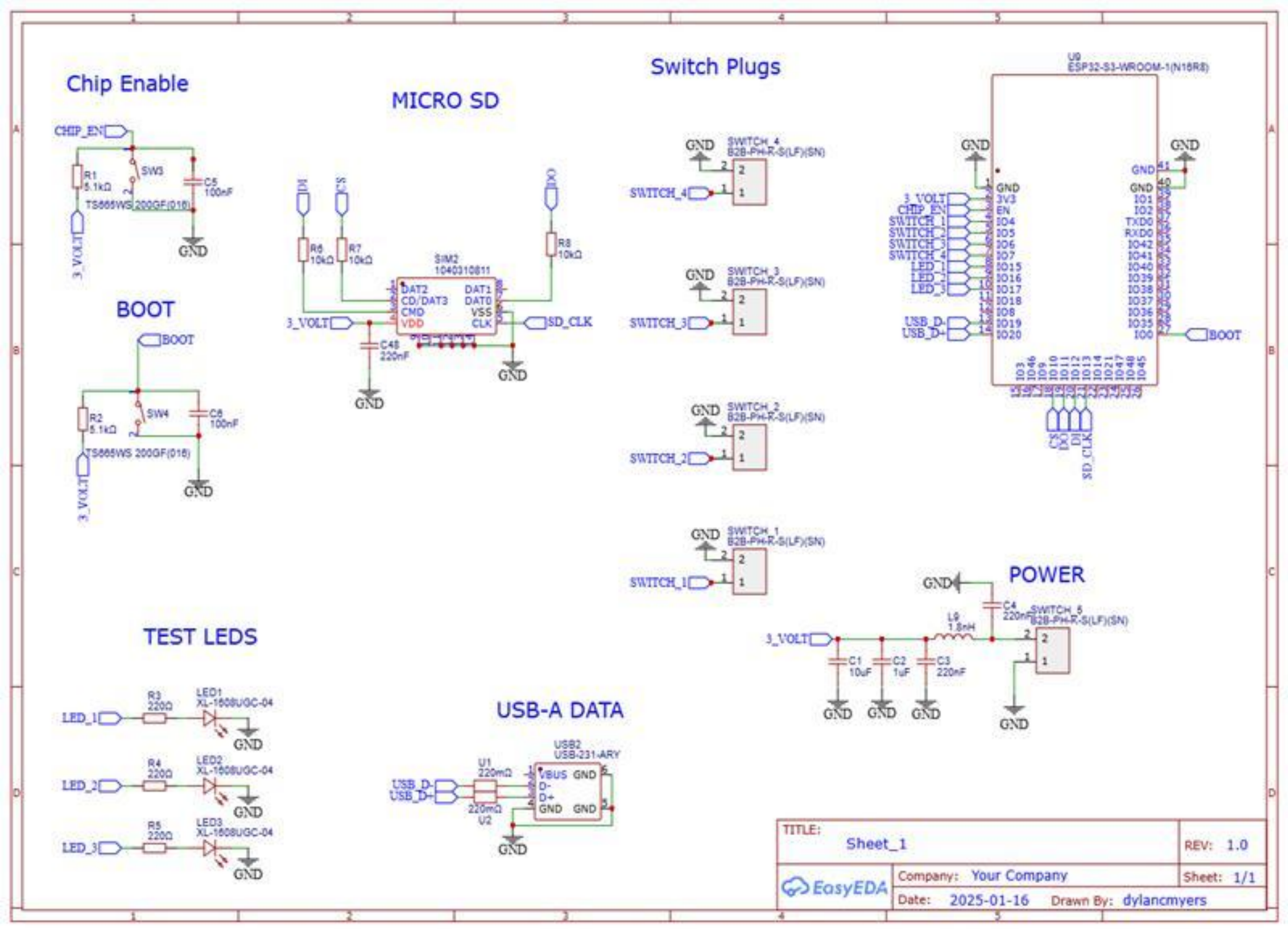




Dylan Myers
ctrical Engineering

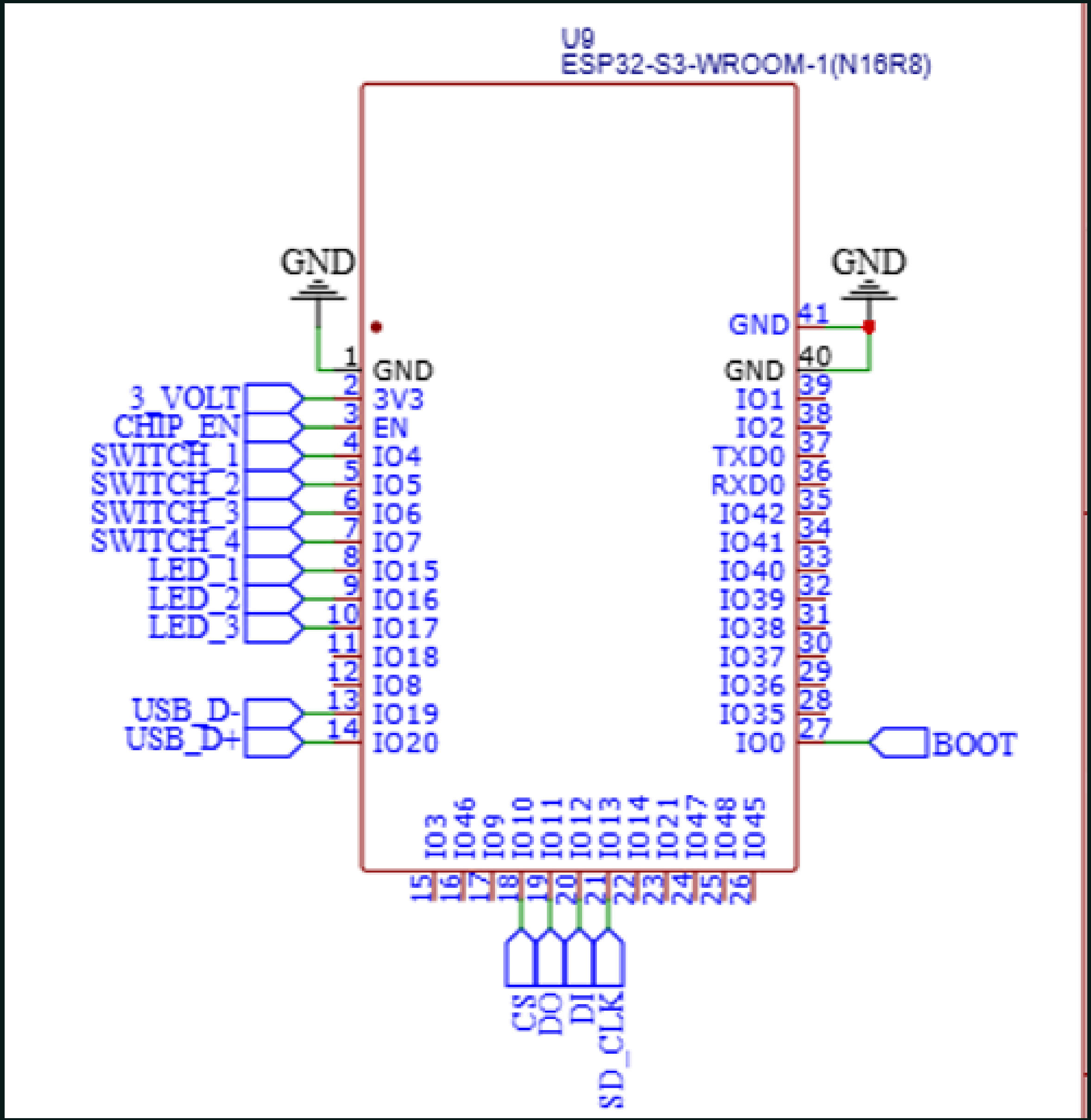
Gadget Schematic Overview

- LED Plugs
- Switch Plugs
- Bott/Chip enable buttons
- Micro SD slot
- USB-A connector



ESP-S3- Wroom-1

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



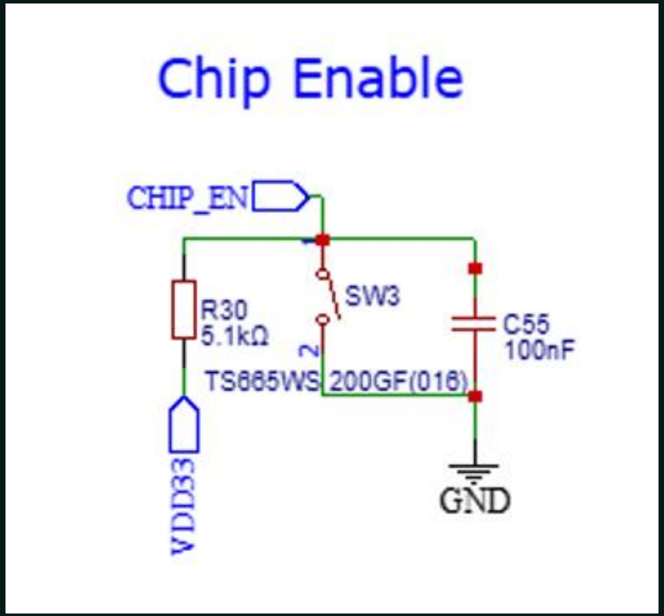
Dylan Myers
Electrical Engineering

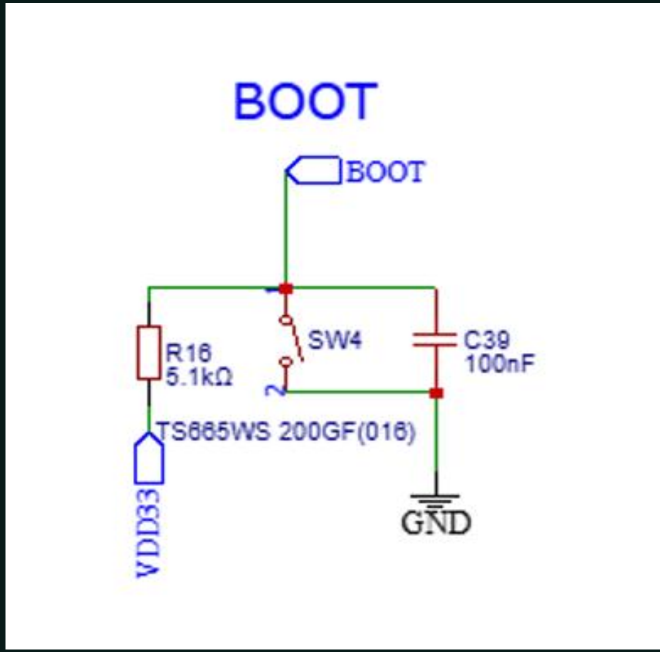
Functional Buttons

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





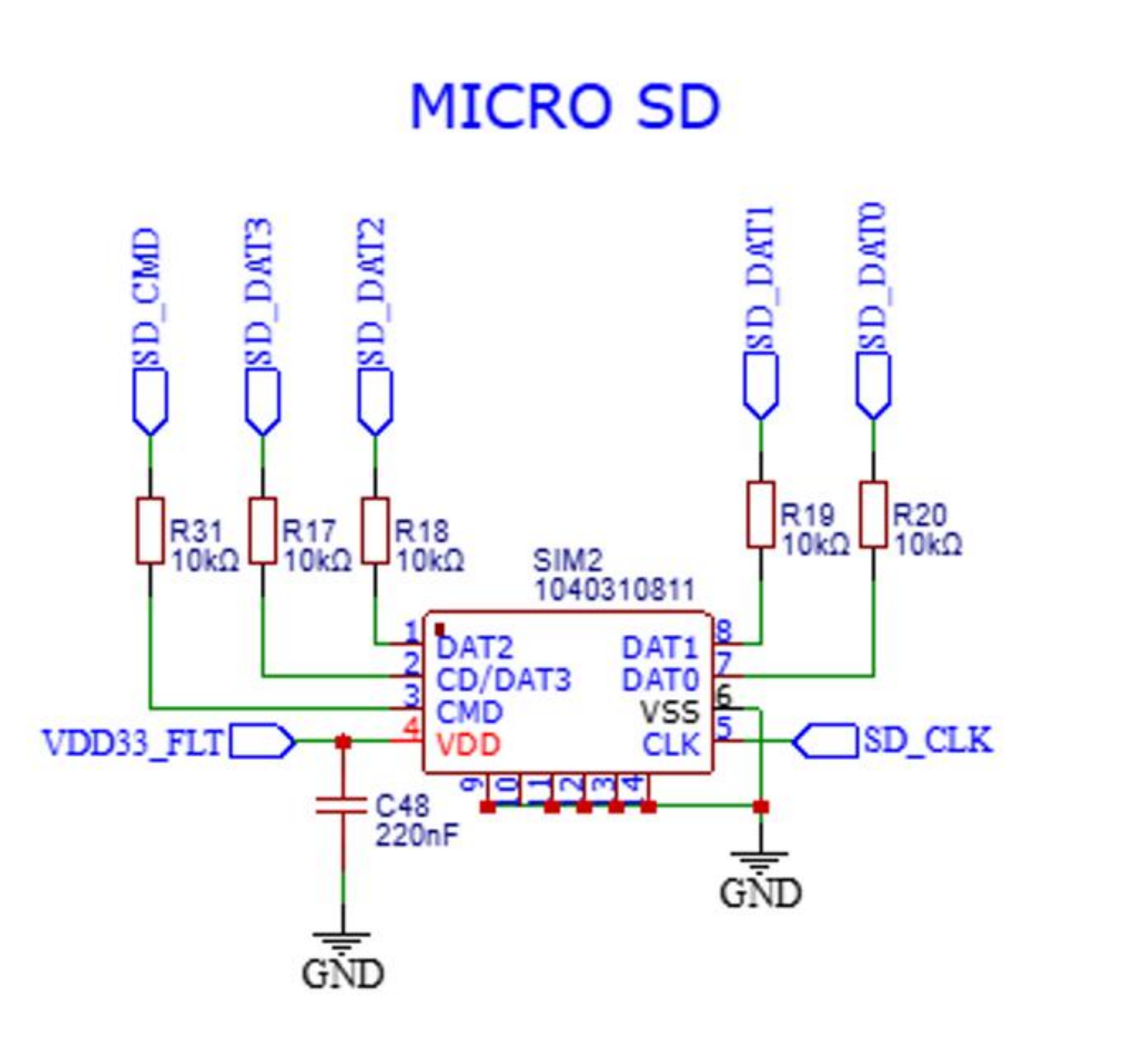
Dylan Myers
Electrical Engineering

Dylan Myers
Electrical Engineering

Micro SD

Used for upload / download of data

Backup for the host data of the website / App

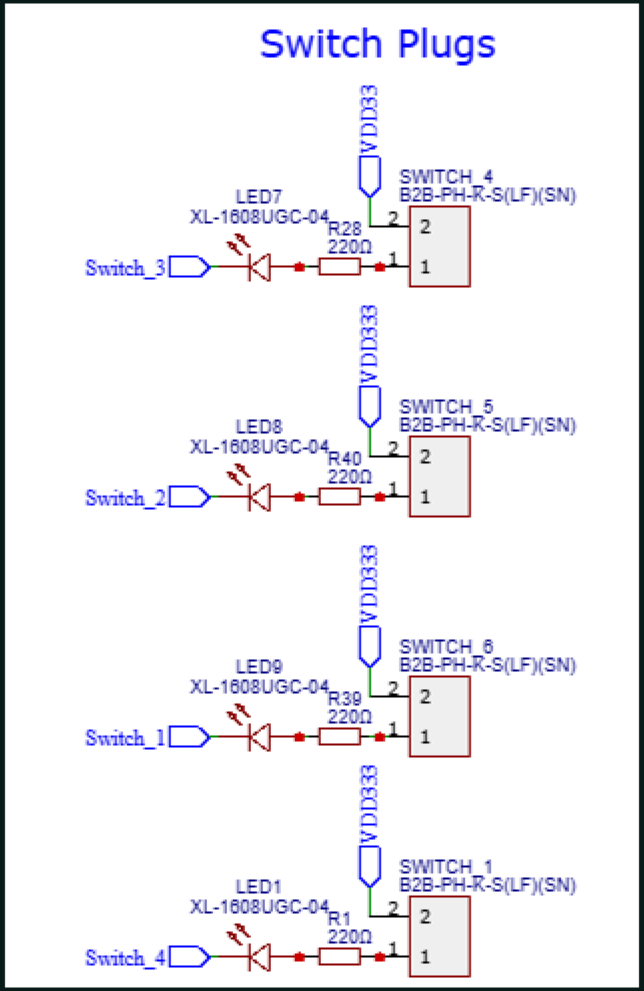


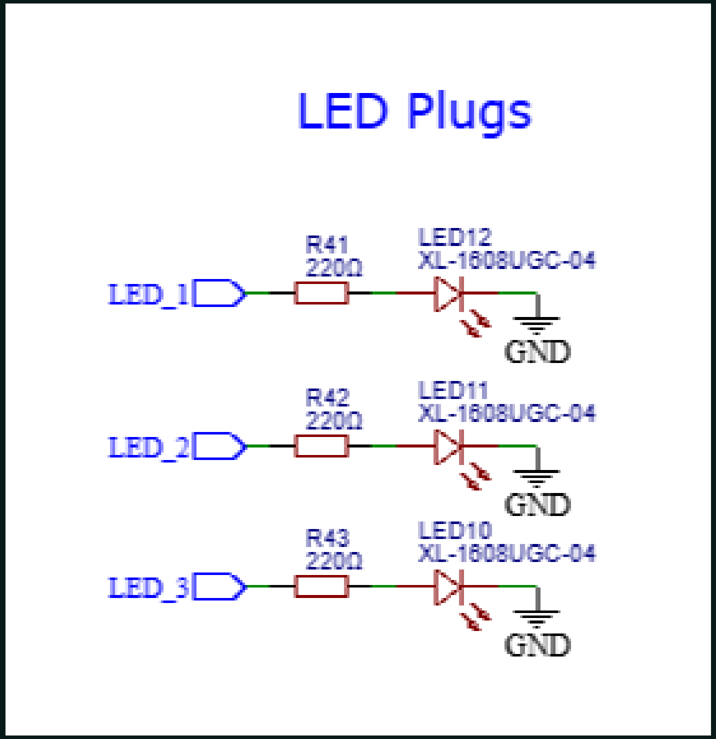
Dylan Myers
Electrical Engineering

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

Power Distribution

USB-C Connector

UART to Serial Converter

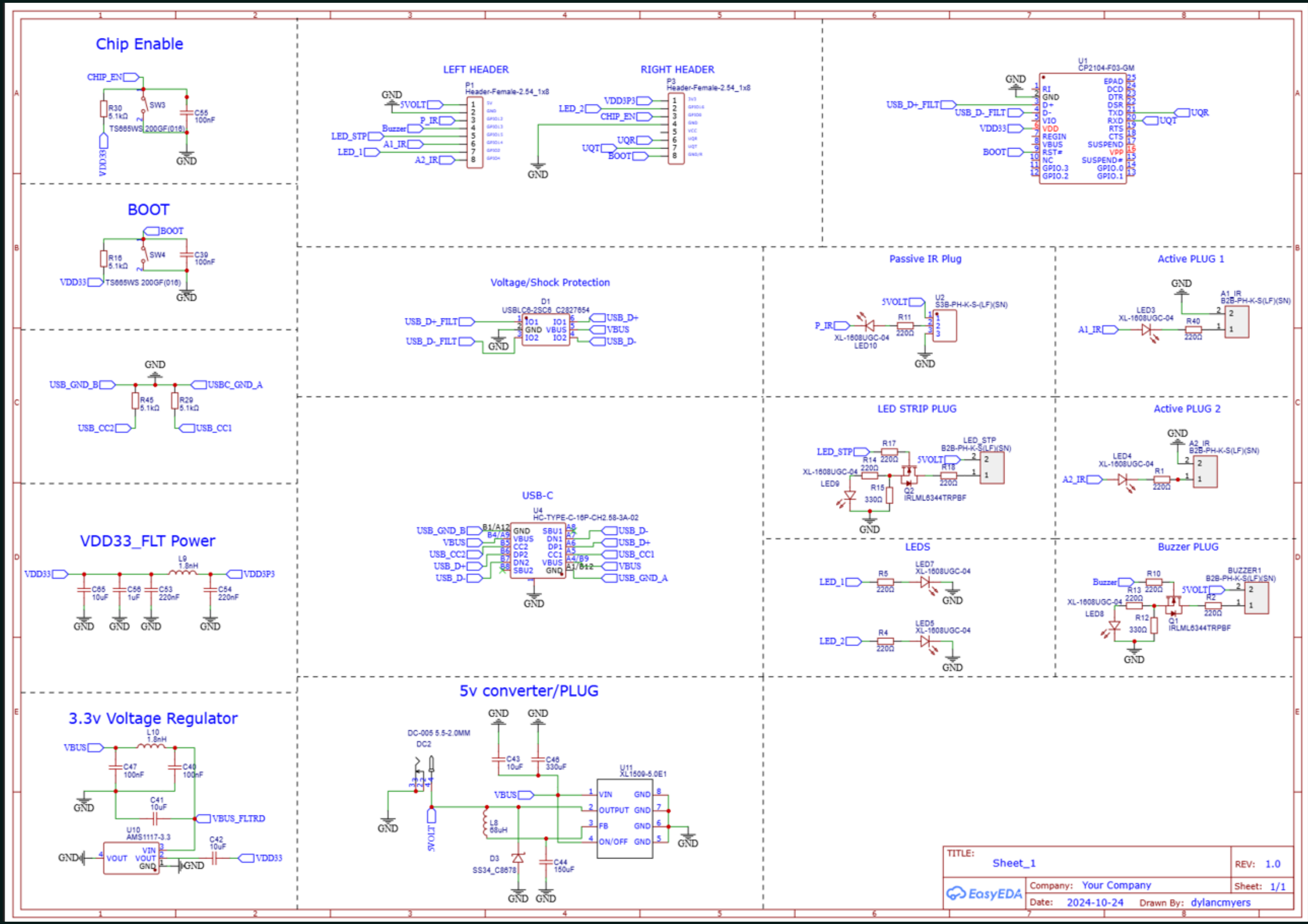
Camera Mounting

Passive IR

LED Strip

Active IR

Buzzer



Dylan Myers
Electrical Engineering

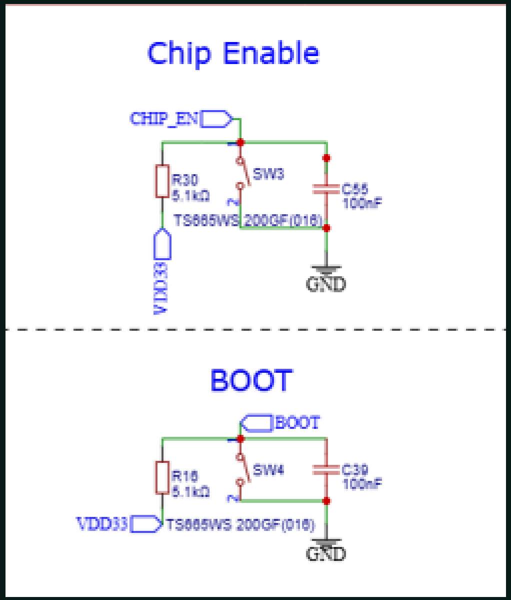
Duplicated from Gadget PCB

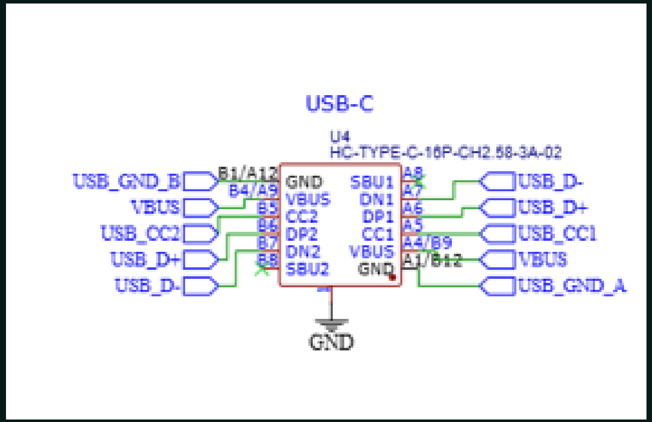
Dylan Myers
Electrical Engineering

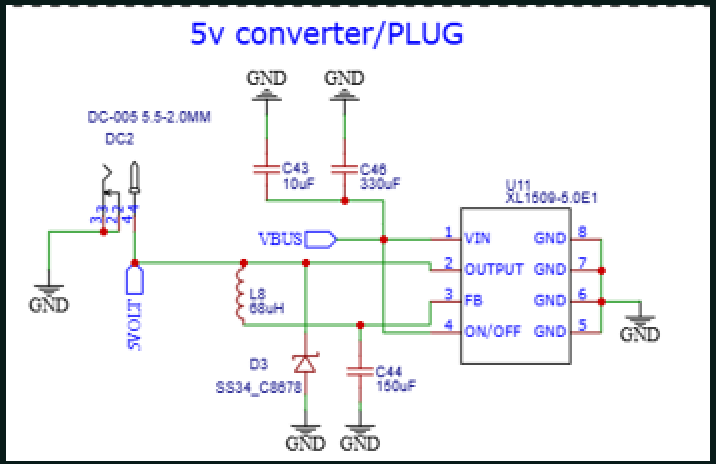
Power Distribution

USB-C Connector

Functional Buttons





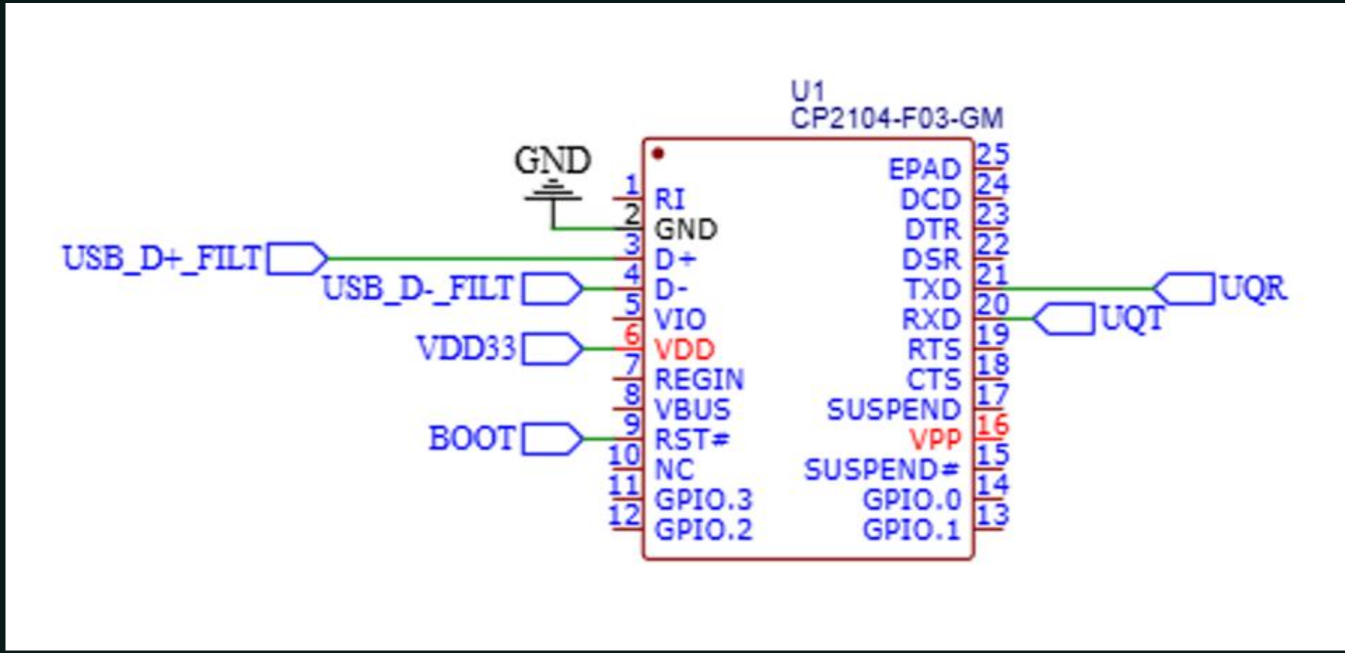


USB-TO-UART BRIDGE

Dylan Myers
Electrical Engineering

Unlike the ESP32-S3

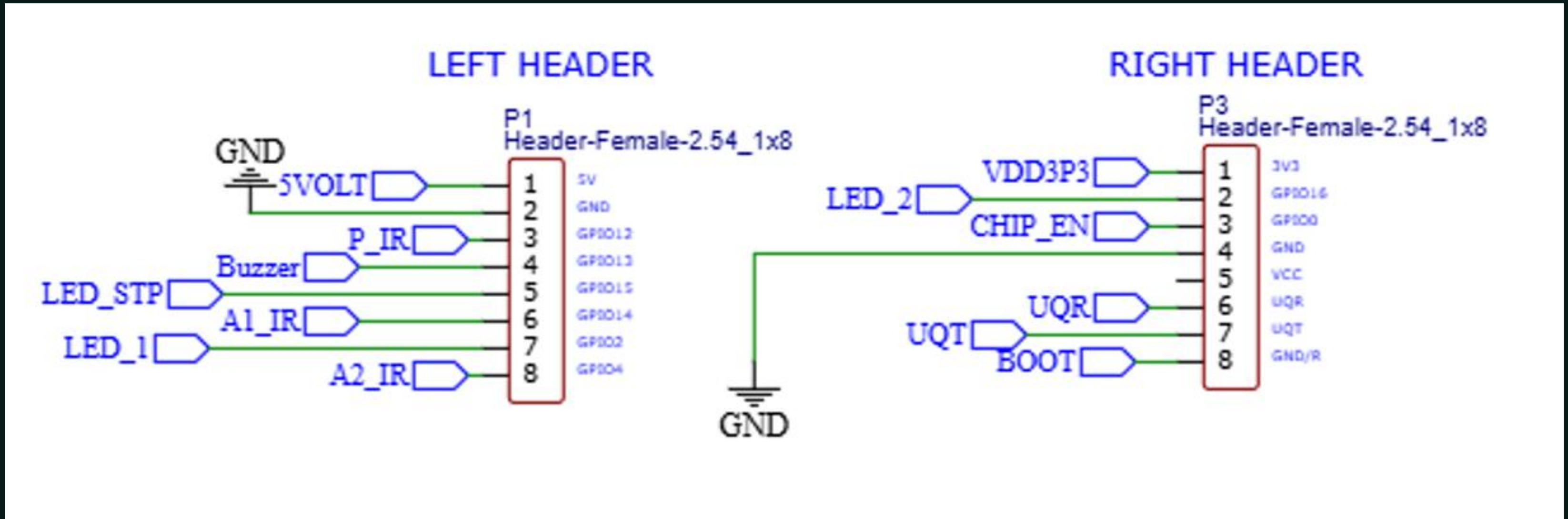
The ESP-Cam doesn't have an onboard UART to USB converter



Dylan Myers
Electrical Engineering

Headers

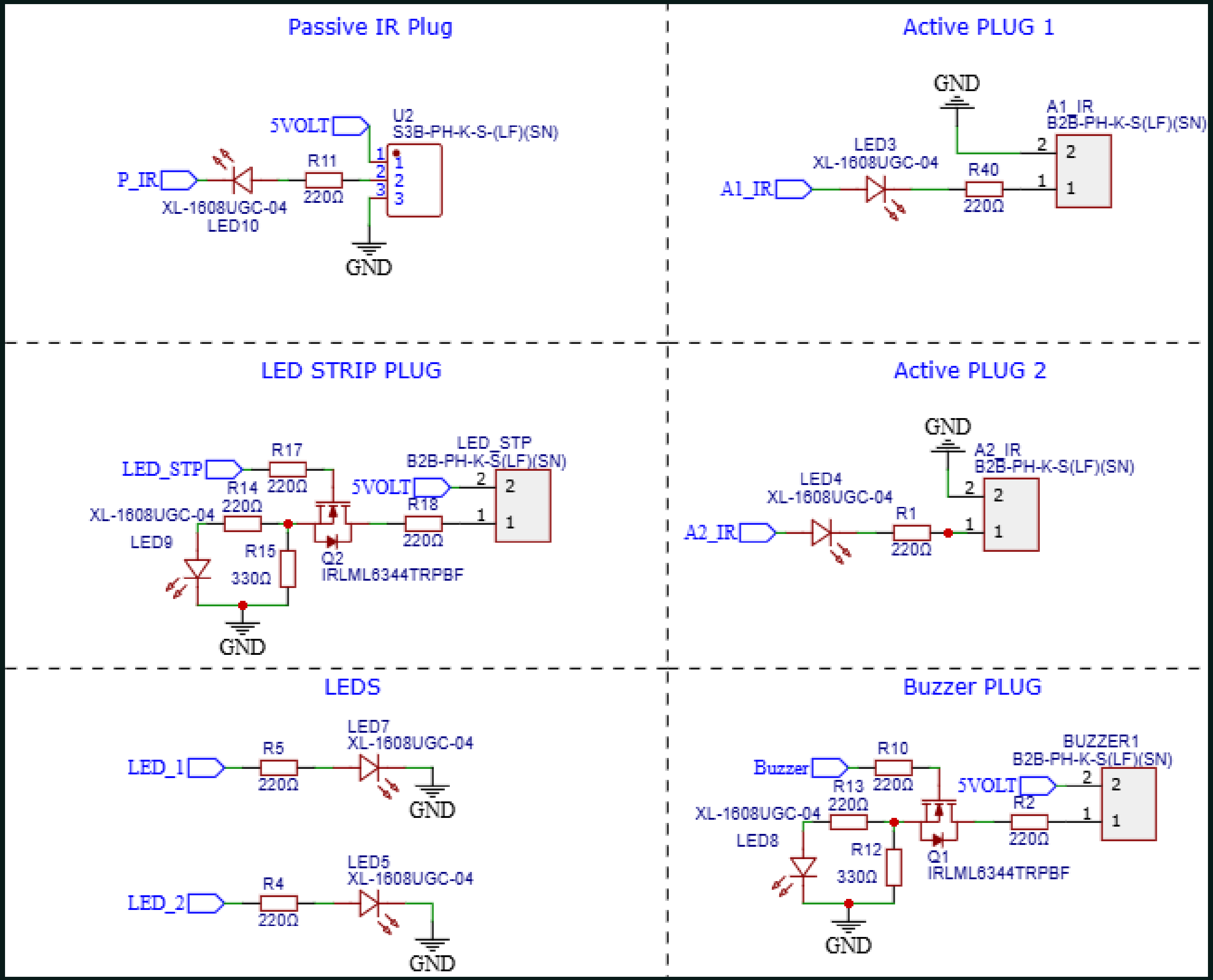
Header pins to connect the ESP32 – CAM to the PCB



Dylan Myers
Electrical Engineering

Plugs

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



Power Schematic Overview

Power Distribution

USB-A Connectors

5V Converter / Plug

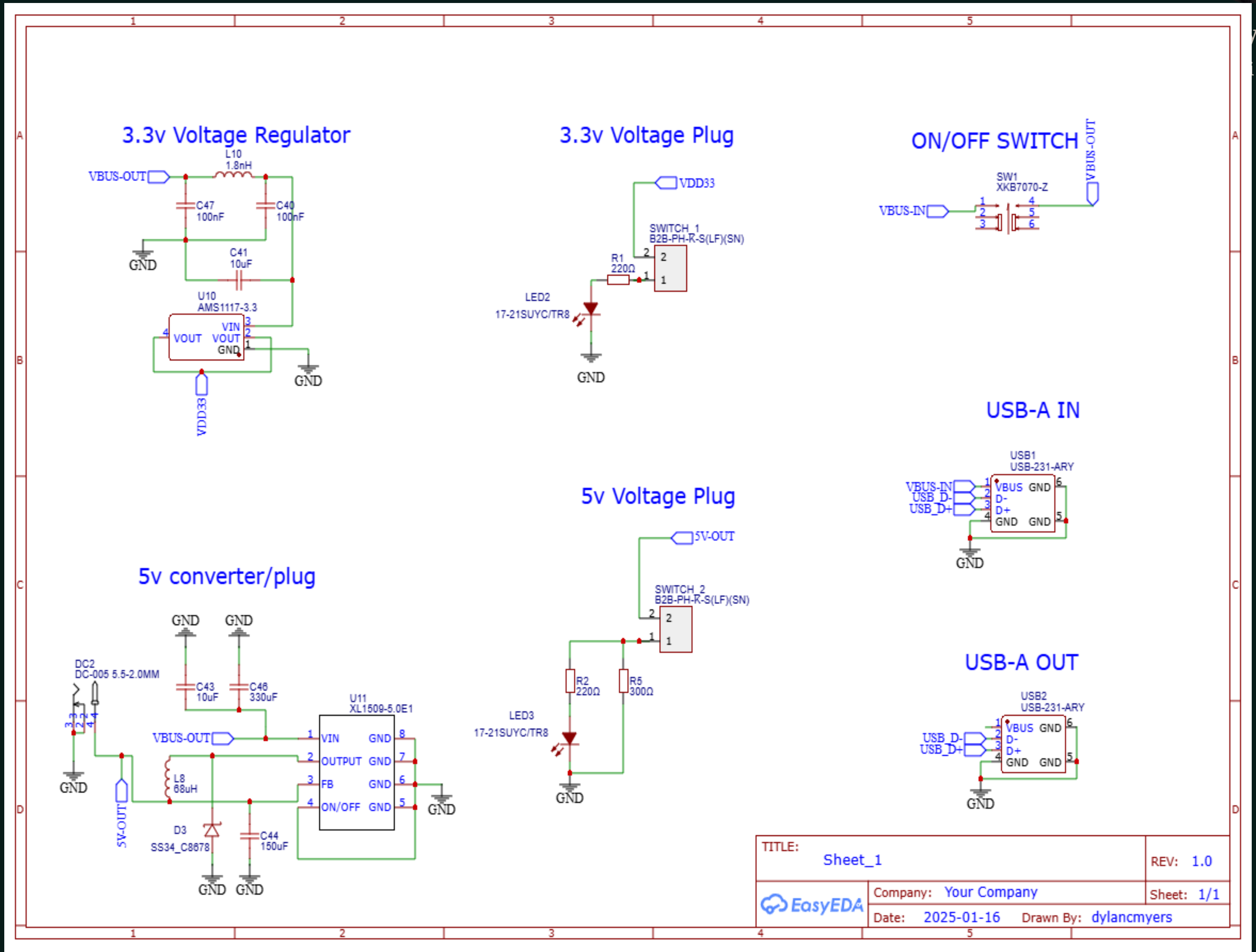
VDD3p3 Power

VDDA Power

Voltage Regulator

RTC Power

Voltage Shock Protection



ylan Myers

ical Engineering

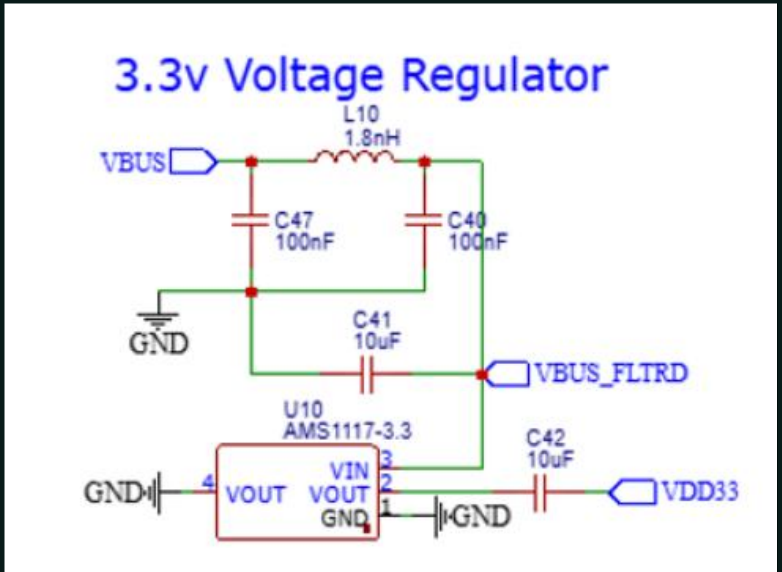
Power Regulation

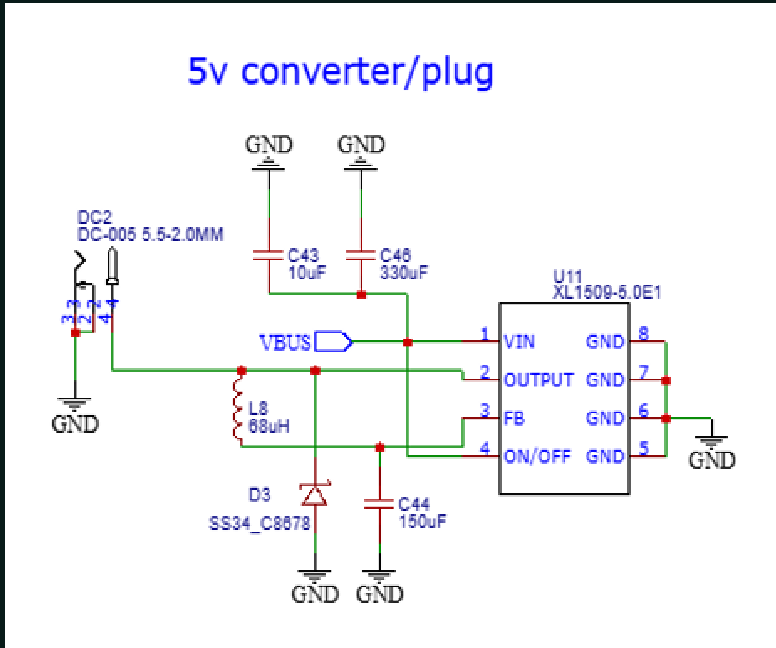
3.3V - Fixed Linear voltage out with adjustable input voltage

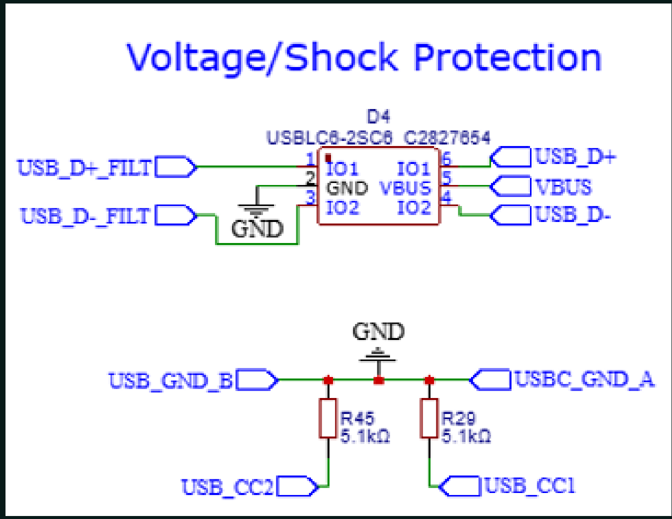
5V – Fixed PWM buck

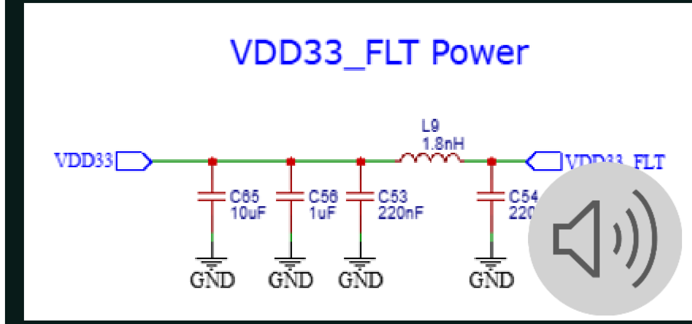
Shock Protection – Path for extreme voltage spikes

Filtration – removes high frequencies









Dylan Myers
Electrical Engineering

PCB Layout & Design Gadget

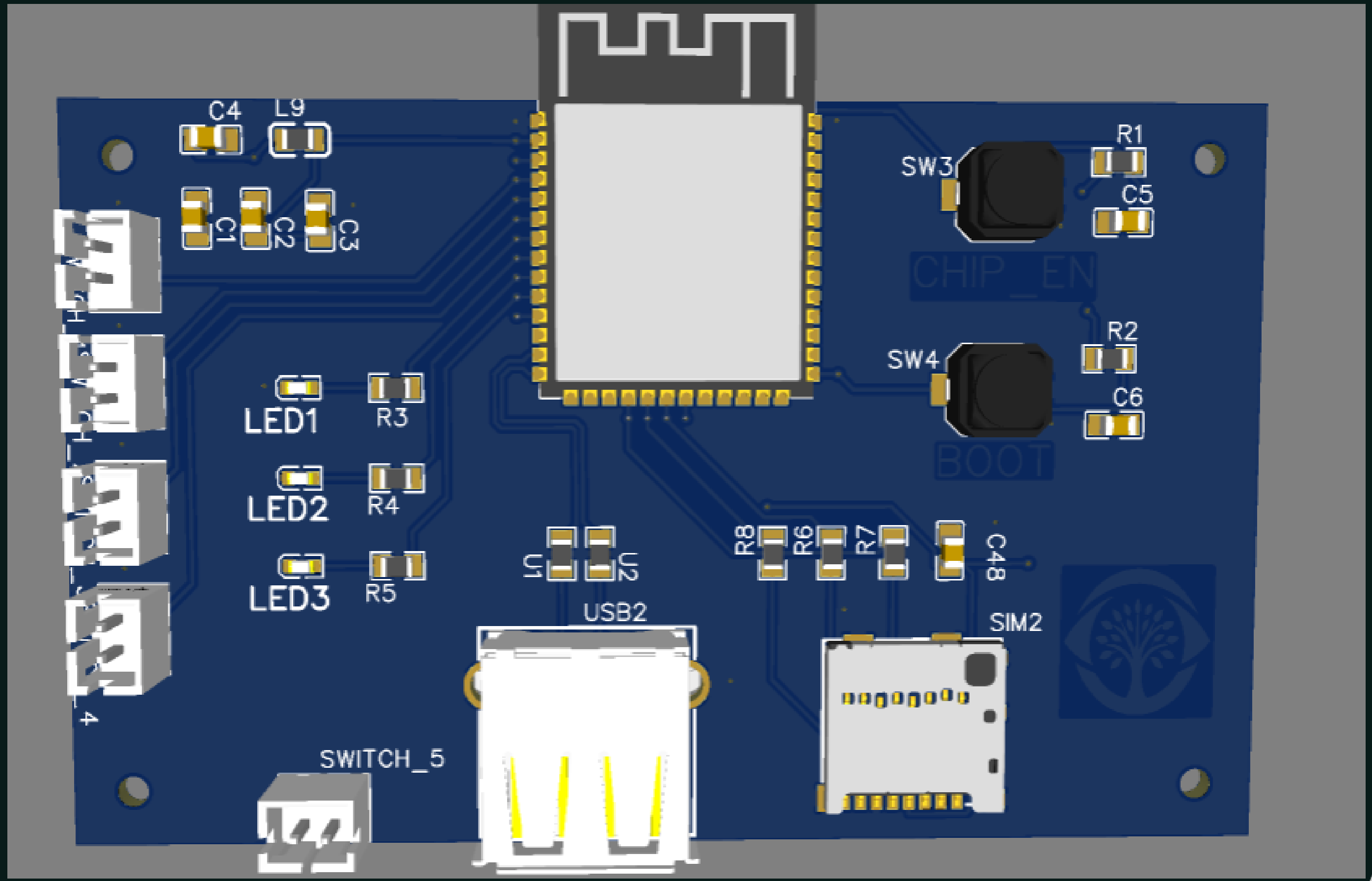
Dylan Myers
Electrical Engineering

2 Layer Design

Power Board Separated for testing

Top layer: Signals

Bottom Layer: Power



PCB Layout & Design Node

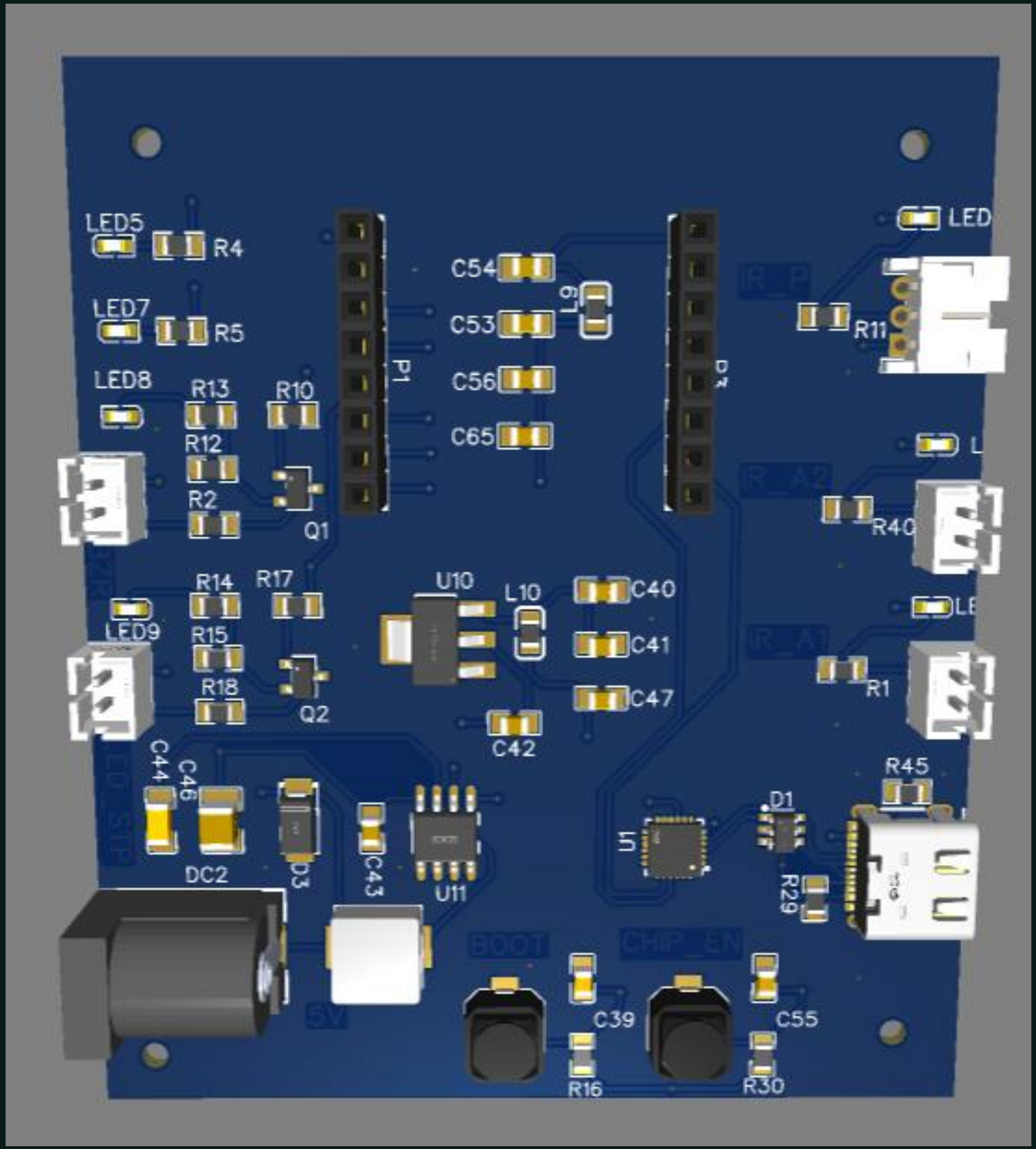
4 Layer Design

Power Board kept on Board for stability

Top layer: Signals

Middle 1: Ground

Bottom Layer: Power



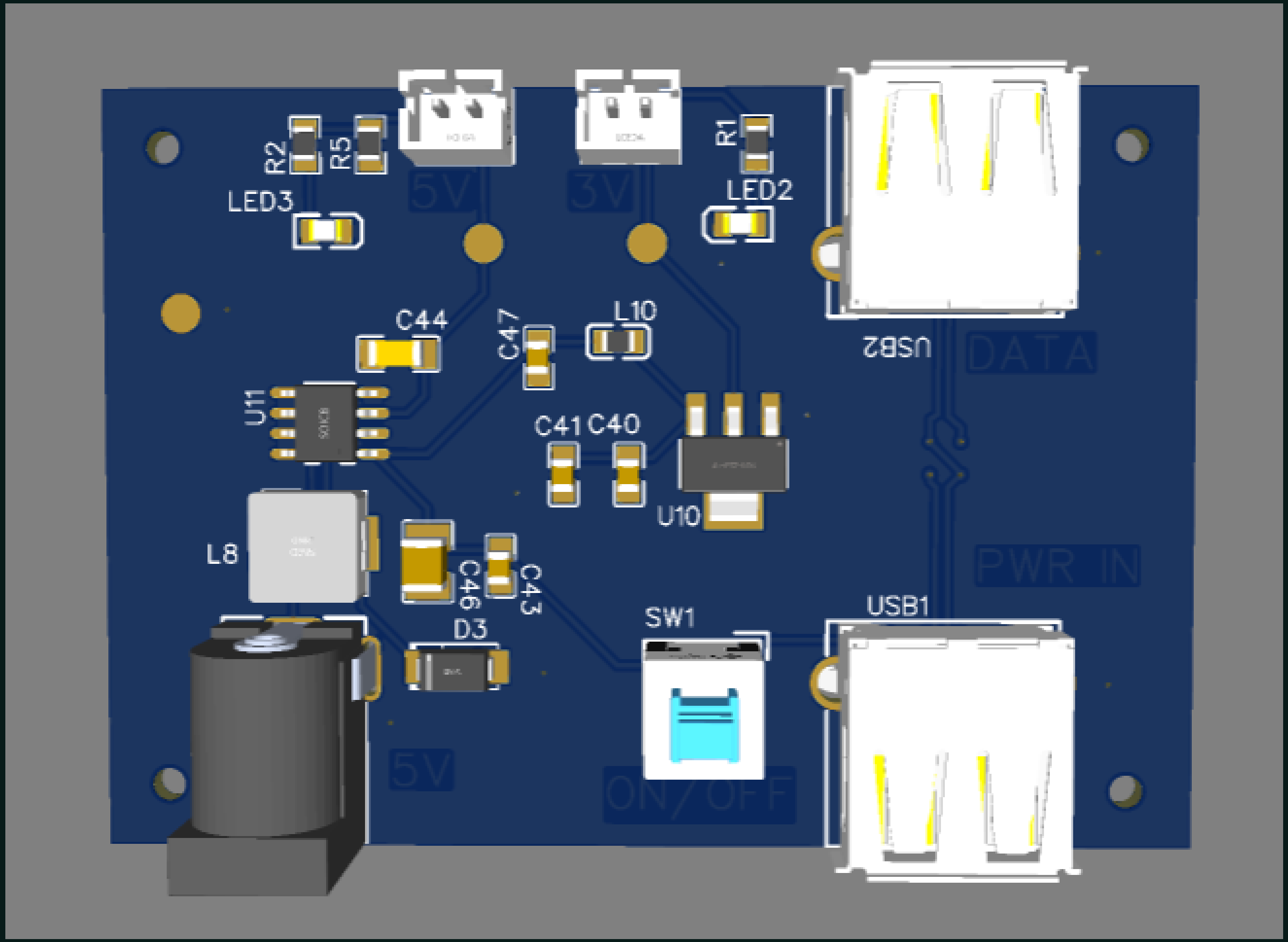
Dylan Myers
Electrical Engineering

Dylan Myers
Electrical Engineering

PCB Layout & Design Power

2 Layer Design

- Able to be used in line with the data
- Works with both the Gadget and Node



Reworks and Revisions

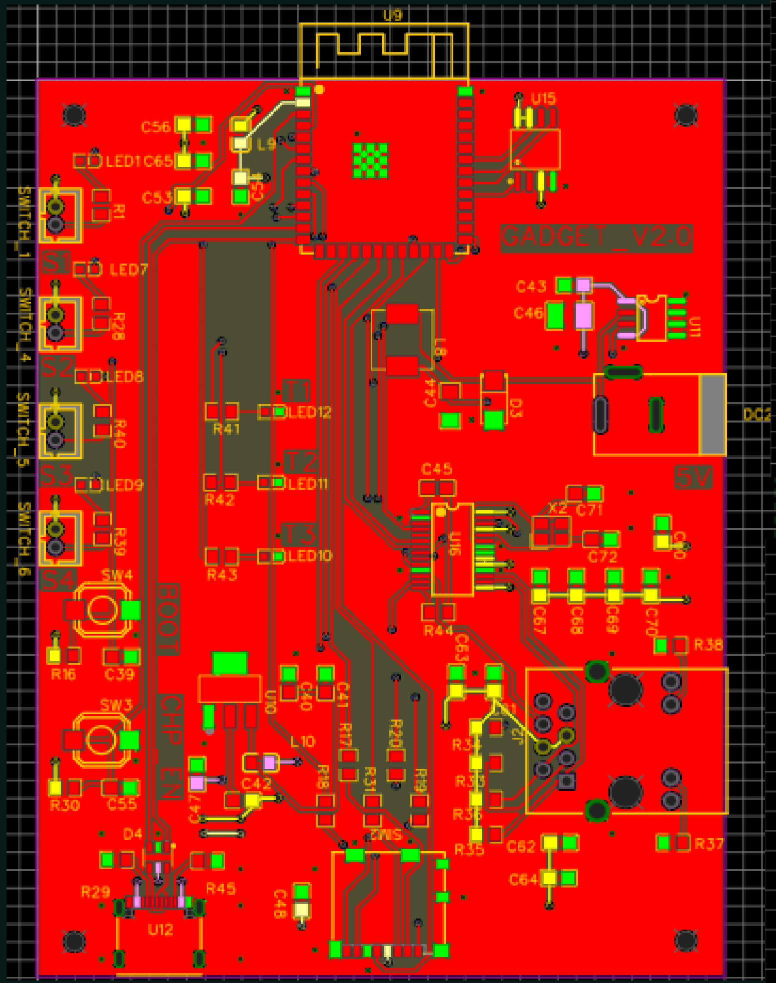
Dylan Myers
Electrical Engineering

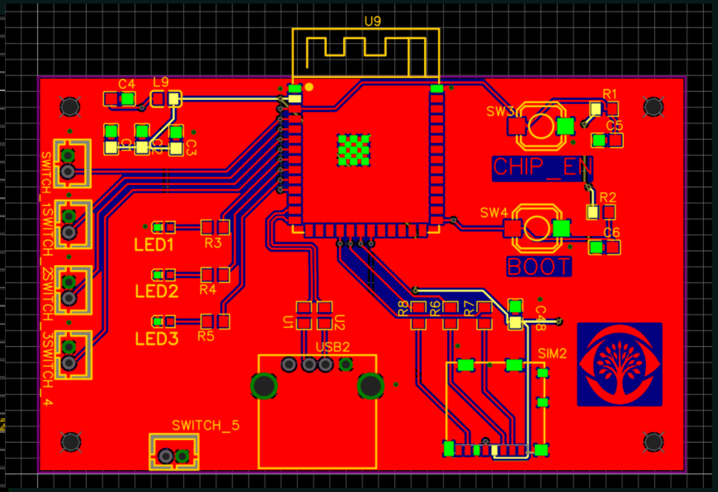
Gadget:

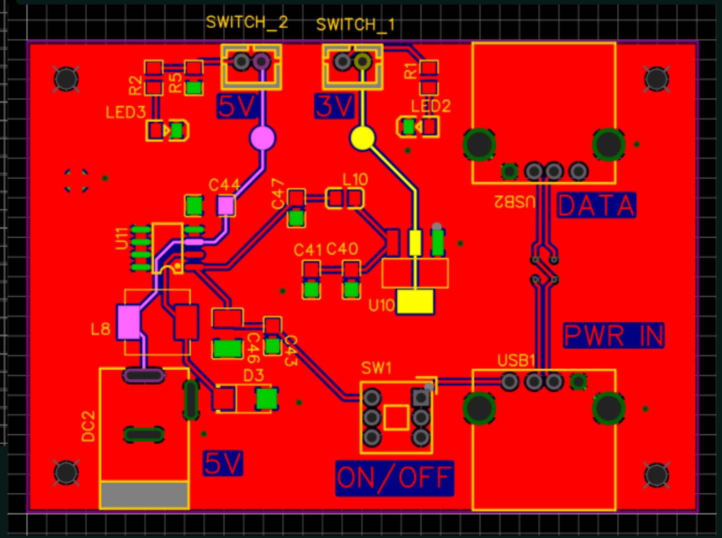
- Split off power to separate board
- Removed components as they became irrelevant to the project
- Extra grounding

Node:

Reworked power outputs







Jaxon Topel
Computer Engineering

Comparison & Selection of Software

- Jaxon Topel
- Colin Kirby

Programming Languages Comparison

Jaxon Topel
Computer Engineering

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.

Programming Languages Selection

Jaxon Topel
Computer Engineering

PYTHON

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

ARDUINO C++

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

CRITICAL REQUIREMENTS

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

WHY BOTH?

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

Image Processing Libraries Comparison

Jaxon Topel
Computer Engineering

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.

CRITICAL REQUIREMENTS

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

WHY TF LITE?

- 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.

Communication Protocols Comparison

Jaxon Topel
Computer Engineering

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.

Communication Protocols Selection

Jaxon Topel
Computer Engineering

HTTP

- 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.

CRITICAL REQUIREMENTS

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

WHY HTTP?

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

Mobile App Frameworks Comparison

Colin Kirby
Computer Engineering

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

Mobile App Framework Selection

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.

CRITICAL REQUIREMENTS

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

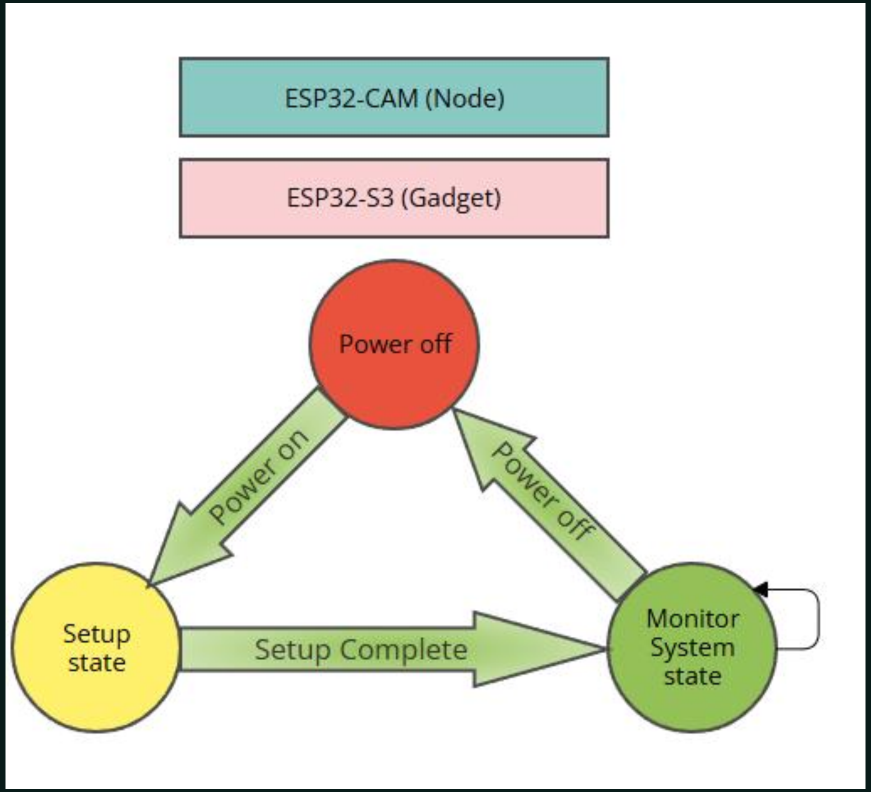
WHY NOT OTHERS?

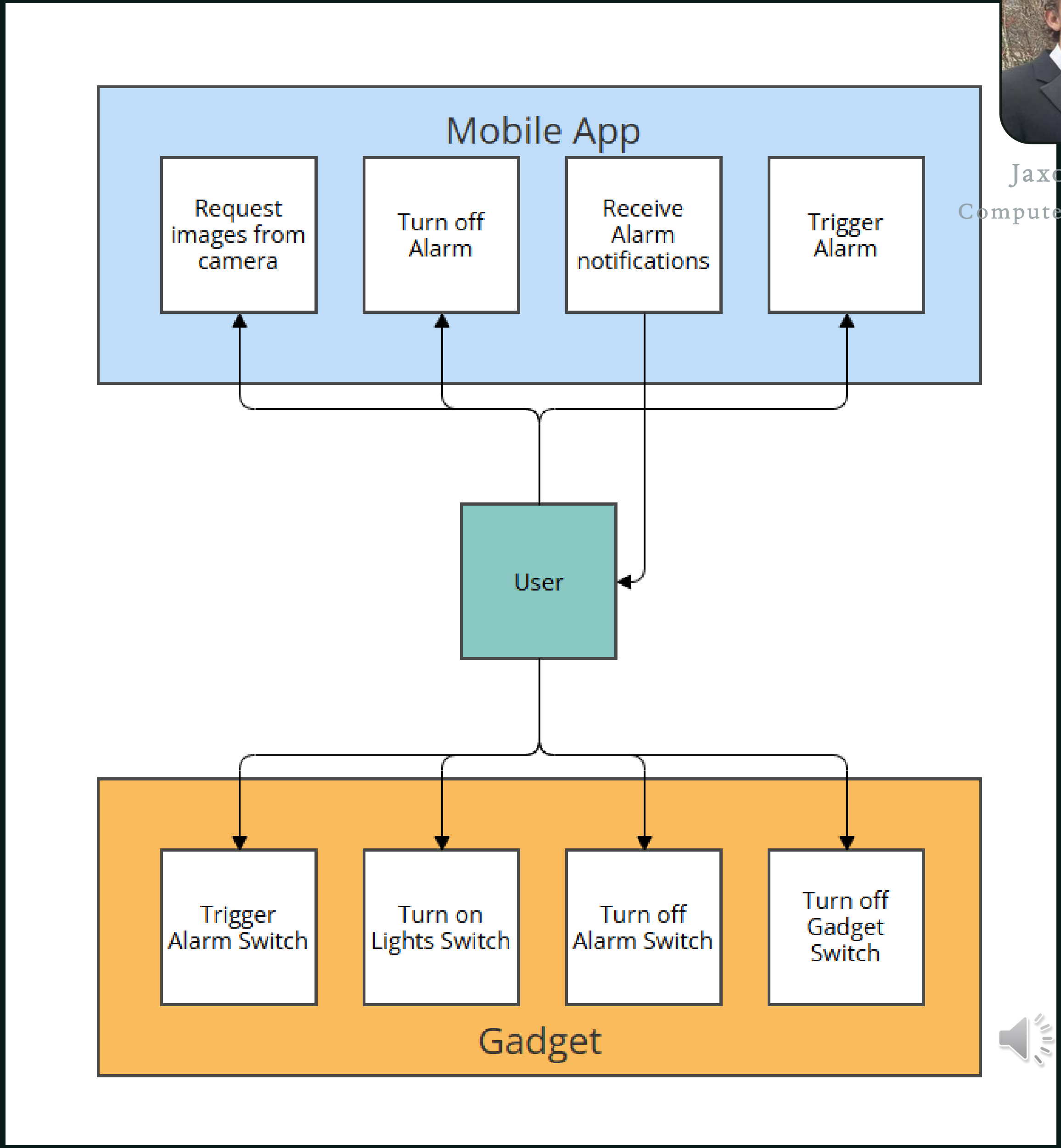
React Native : Easy to use, but slower due to JavaScript bridging.

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

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

Software State & Use Case Diagrams

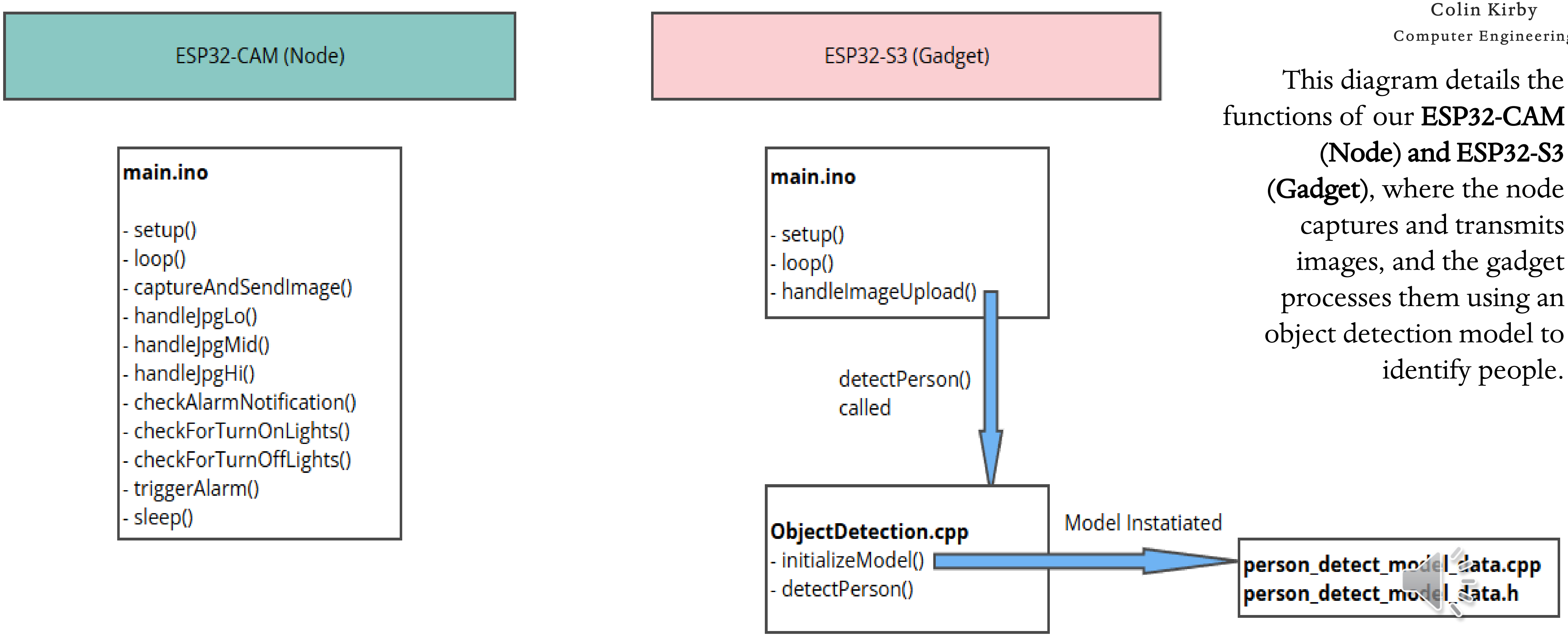




Jaxon Topel
Computer Engineering

Backend Class Diagram

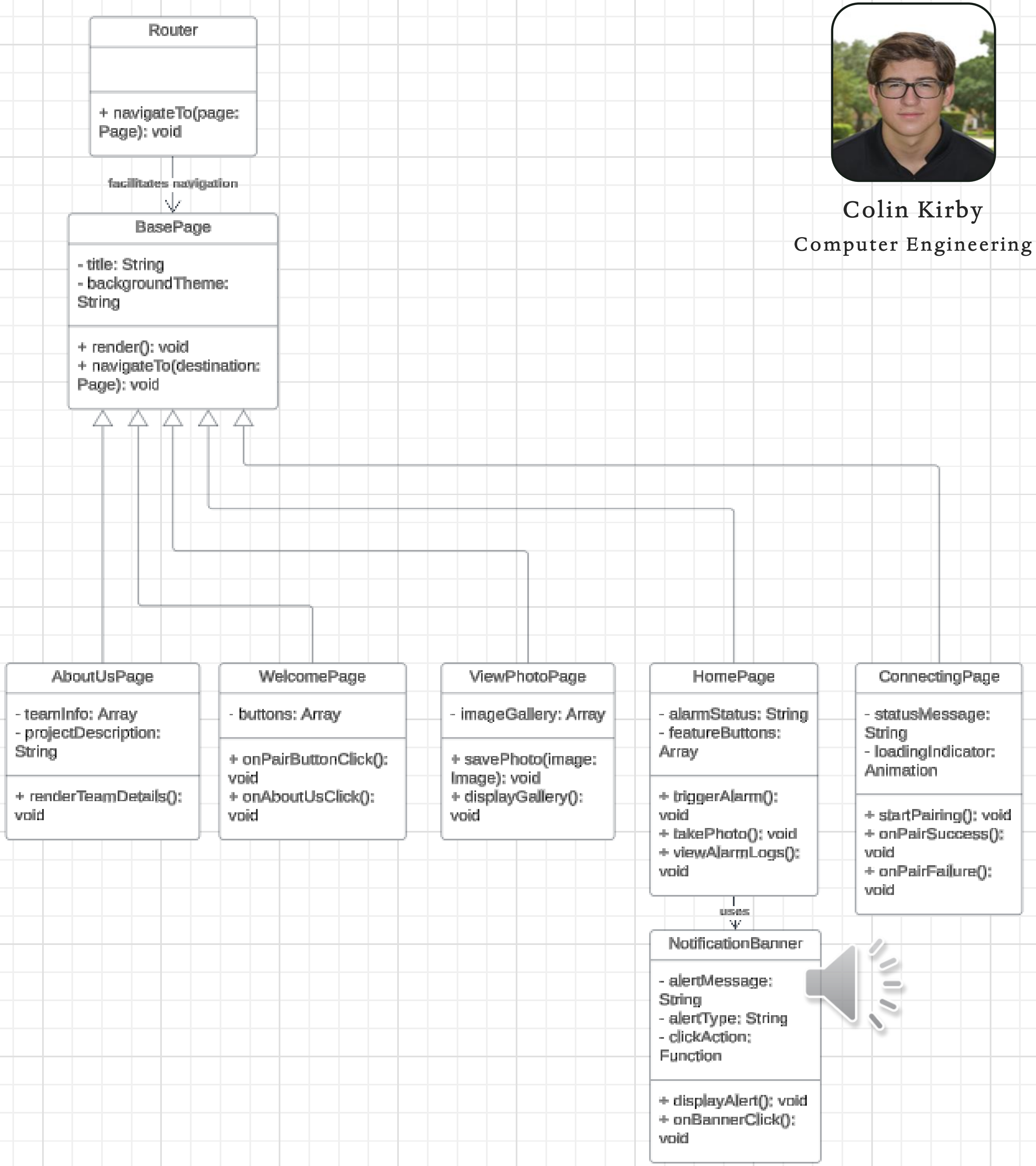
Colin Kirby
Computer Engineering



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

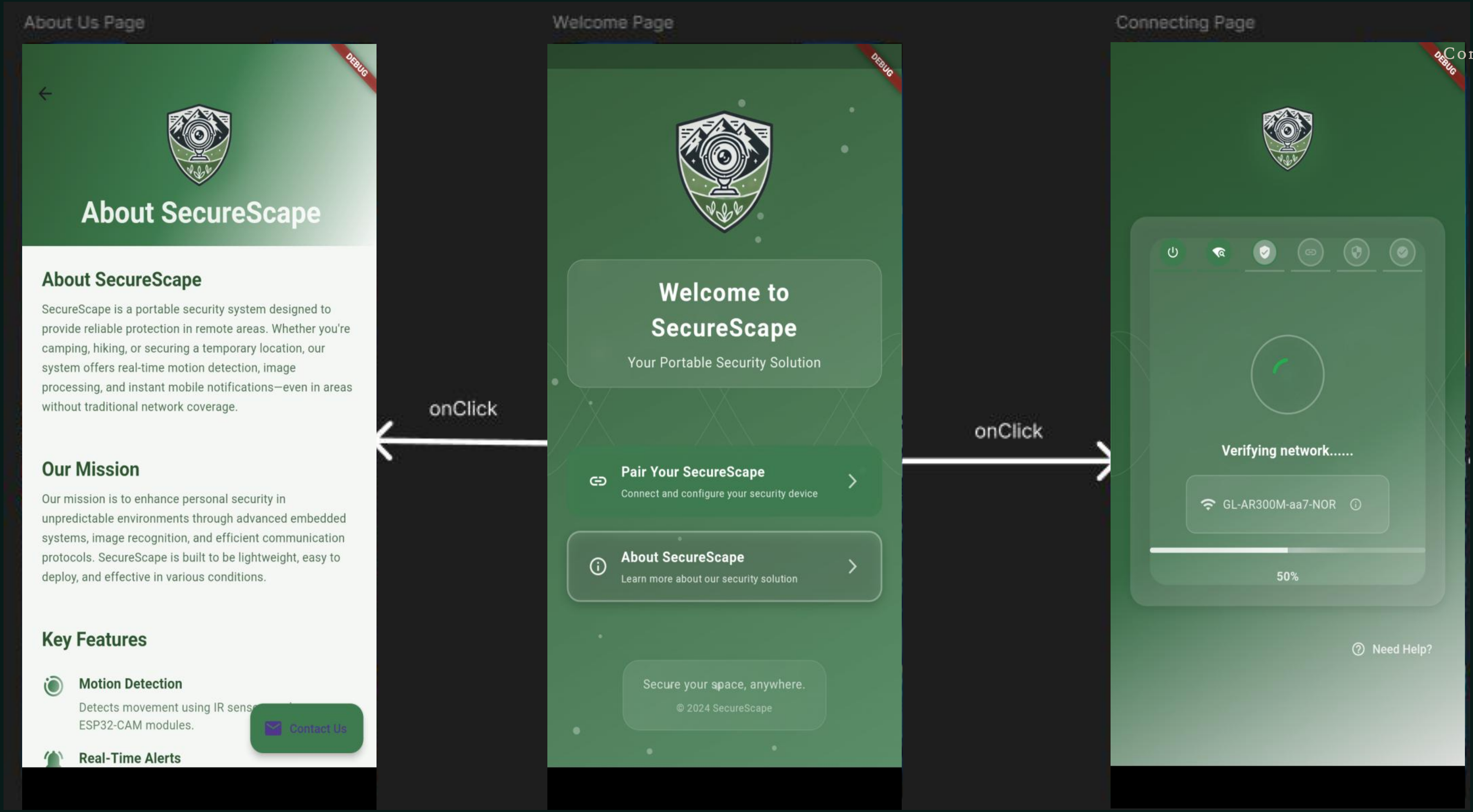
- **Router** : manages navigation between pages.
- **BasePage** : provides common properties like title.
- **HomePage** : controls alarms and logs.
- **AboutUsPage** : displays team and project details.
- **WelcomePage** : handles navigation buttons.
- **ViewPhotoPage** : manages image storage and display.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



Colin Kirby
Computer Engineering

Mobile App UI - Part 1

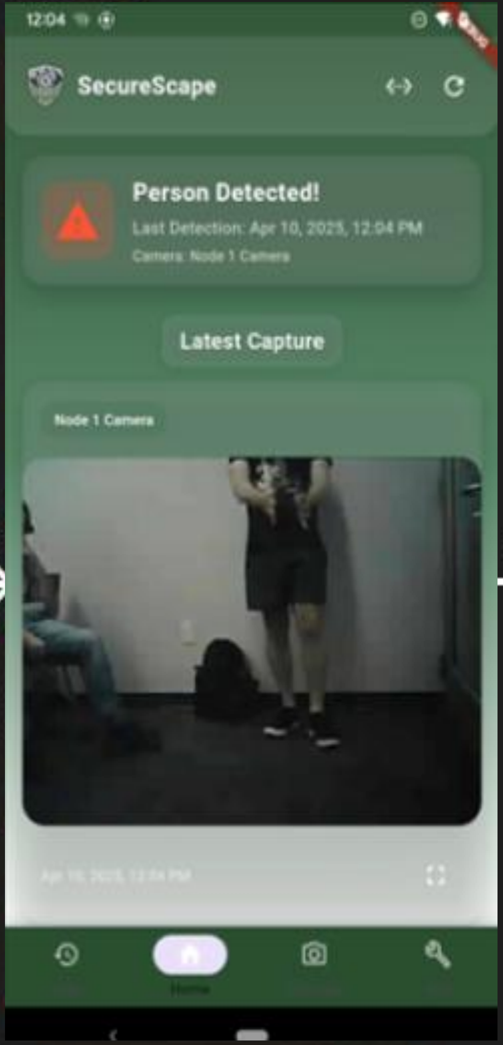
Colin Kirby
Computer Engineering



Mobile App UI - Part 2

Colin Kirby
Computer Engineering

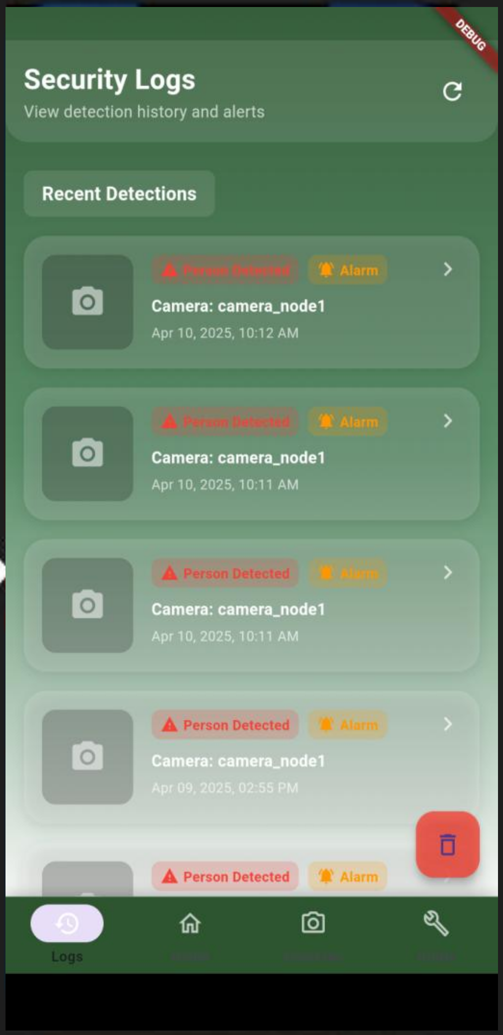
Home Page



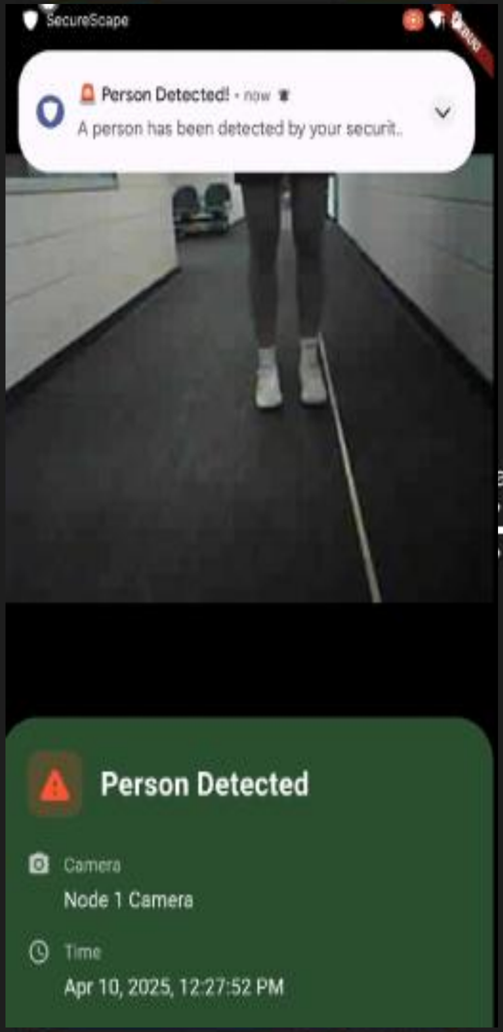
afterConnect

afterTakePhoto

View Photo Page



Notification Page



afterNotification

Colin Kirby
Computer Engineering

Prototyping & Testing

Software Testing Breakdown

Colin Kirby
Computer Engineering

- 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.

Embedded/Hardware Peripherals Test

Phillip Murano
Electrical Engineering

- **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, buzzer, 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.

Node Peripheral Tests

Phillip Murano
Electrical Engineering

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.	PASS	Lights successfully turned on.

Gadget Peripheral Tests

Phillip Murano
Electrical Engineering

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.	PASS	Successfully able to trigger alarm from our gadget switches.
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.	PASS	Successfully able to turn both on and off the LED strip with switches.
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.	PASS	Successfully able to turn off alarm from our gadget switches.

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

Mobile Application UI Tests

Colin Kirby
Computer Engineering

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	PASS	Allows user to navigate successfully through application.
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.

Image Processing Algorithm Test Breakdown

Jaxon Topel
Computer Engineering

- **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)

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

Jaxon Topel
Computer Engineering

- **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

General Project Budget

Phillip Murano
Electrical Engineering

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	34.32	34.32
Node batteries	POAWANG	1	34.32	34.32
PCB Node Build	JLCPCB	1	74.80	74.80
PCB Gadget Build	JLCPCB	1	69.36	69.36
Passive IR	STEMEDU	1	9.98	9.98
PCB Ccomponents	LCSC	1	56.05	56.05
LEDs	CHANZON	1	5.31	5.31
Buzzer	WEICHUANG	3	5.49	5.49
Housing nodes	In House Build	3	0.00	0.00
Housing gadget	In House Build	1	0.00	0.00
Micro Router	GL.iNET	1	31.90	31.90
Gadget Switchs	Same Sky	10	.53	5.30
Wires	Sunxeke	1	9.99	9.99
Total Cost				$ 383.23

Progress & Milestones

Dylan Myers
Electrical Engineering

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

Work Distribution

Phillip Murano
Electrical Engineering

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	Philip Murano	Dylan Myers
Node Peripherals	Dylan Myers	Phillip Murano
PCB Design	Dylan Myers	Phillip Murano
Gadget Peripherals	Dylan Myers	Phillip Murano

Thank You!

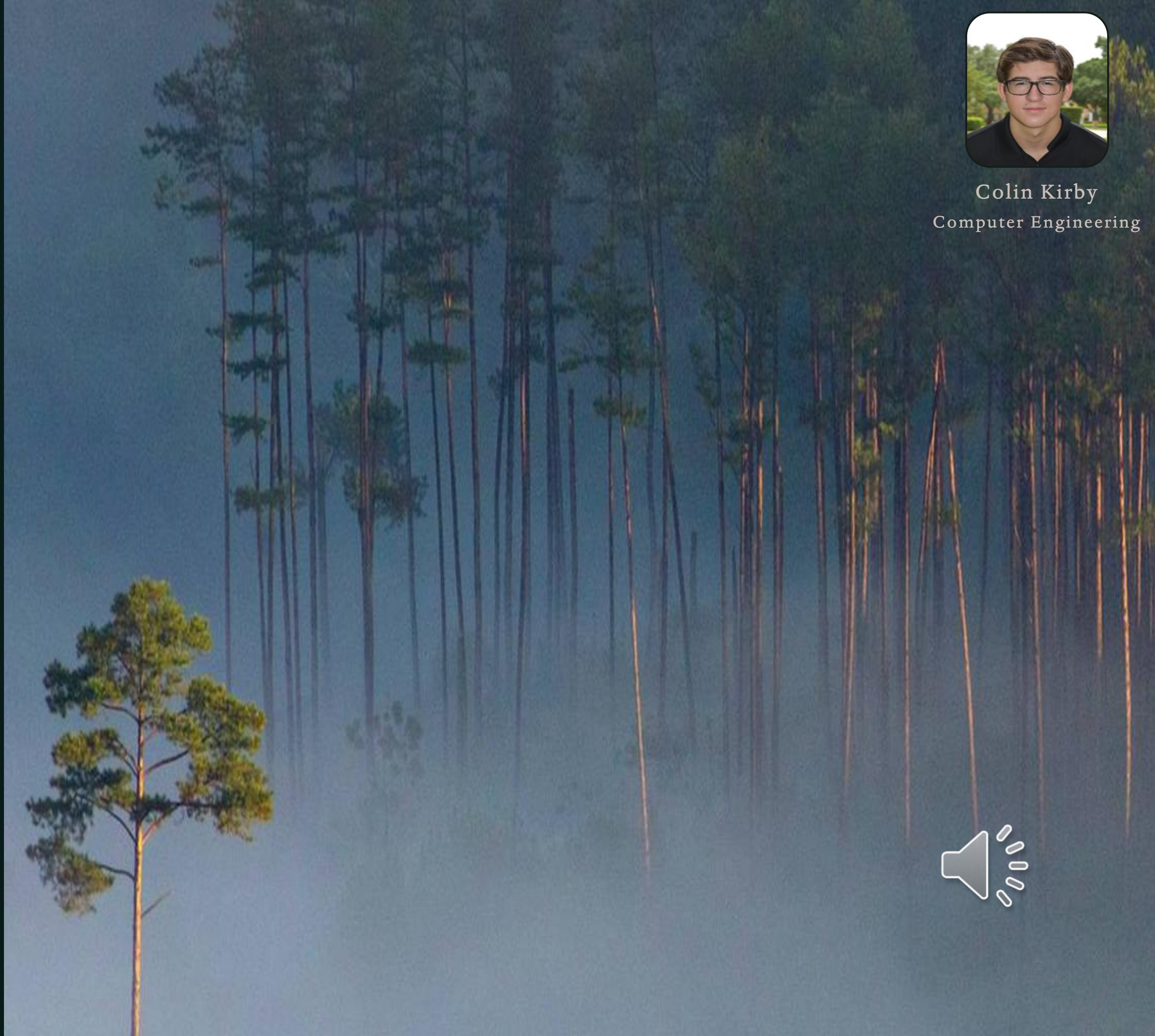
Jaxon Topel

Dylan Myers

Colin Kirby

Phillip Murano

Questions?



Colin Kirby
Computer Engineering