Project Final Report

EYERIS LIGHT CONTROLLER

ENSC 491 - 63

—

Jared Alano-Gray (CPEN)

Tarik Alauddin (CPEN)

Carson Lewis (ME)

Chance Schacle (ME)

—

Dan Harmon (Advisor)

—

4/21/2021

Executive Summary

The development of the Eyeris Light Controller (ELC) is an autonomous lighting system that provides a consistent and thorough level of light in a commercial space such as a classroom or office. Using this consistent level of light, ELC creates a comfortable, Eco-friendly, and healthy environment. The ELC utilizes radio frequency (RF) to wirelessly communicate with light sensors and LEDs. ELC can adjust manufactured light levels based on the presence of ambient light to a user defined preset.

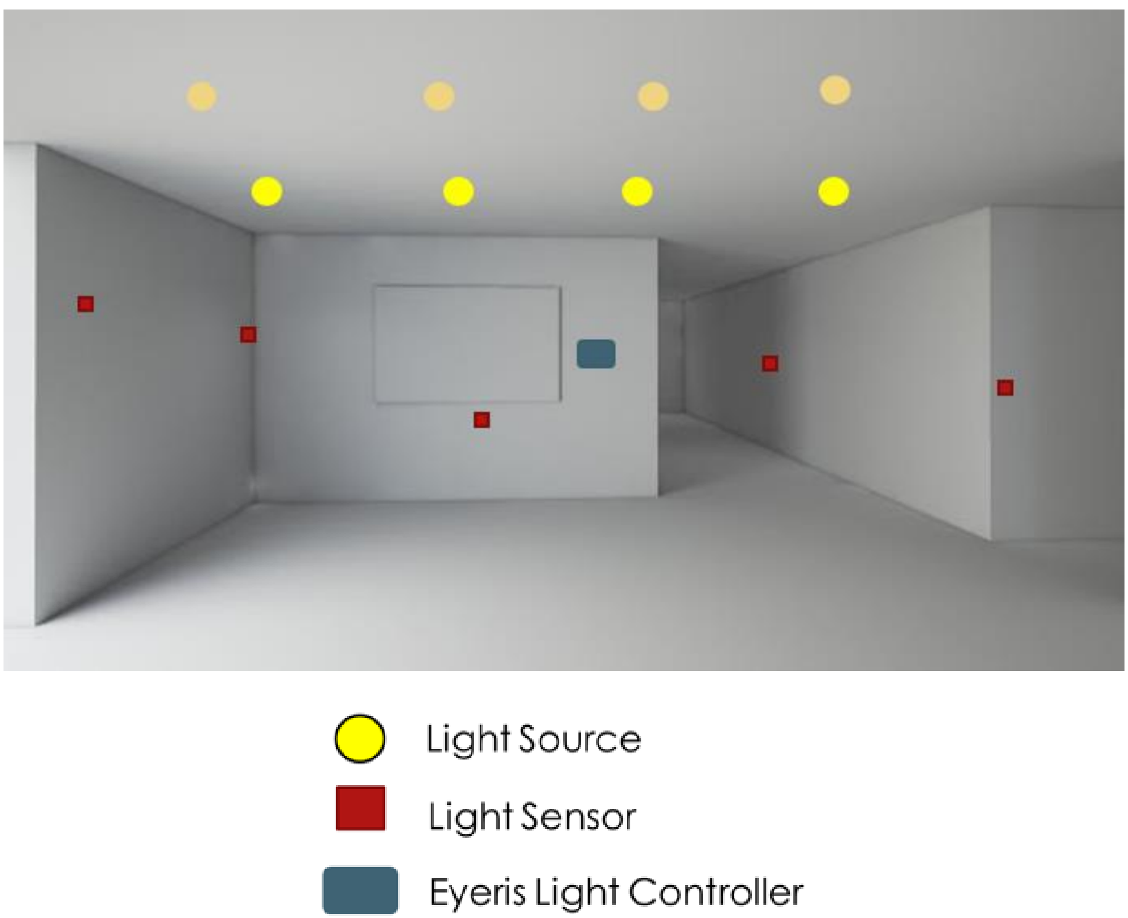
Ample part research was key to deciding on compatible and appropriate component selection. Increasing the amount of time allotted to research was beneficial to the design decisions that were made and provided a better understanding of what direction to take. Although the schedule had been modified, responsibilities among the design team remained unchanged. With the goal of designing an industry-competitive product, several major design decisions were made. First, the means of communication between the components and the controller will ultimately guide the project. Wireless communication proved to be the obvious choice over a physical wired connection mainly due to usability and ease of implementation. The means by which the controller would wirelessly communicate with components favored RF over Bluetooth communication on the account of range, connectivity, and multiple device communication. The selection of the Arduino Uno microcontroller proved vital to the project's usability and simplicity. Ultimately, with significant schedule changes and drastic delays in parts deliveries both prototype models had to be developed concurrently. However, we accomplished our goal by creating a functional proof-of-concept model and have allowed for a more refined marketable model to be developed and produced.

Table of Contents

Content	*Page*

PROJECT DESCRIPTION

The goal of this project is to make a controller that will automatically maintain light levels for a given commercial space to reduce eyestrain, prioritize natural light, as well as reduce electricity usage from lighting during the day. The Eyeris Light Controller (ELC) would be the bridge between current technologies and allow for automatic light adjustment based on a user preset. This controller will automatically dim the light sources in the room based on the presence of natural light. Using light sensors, the Eyeris Controller will be able to take in data regarding the levels of light at different positions in a space and adjust the corresponding light sources to a user-set level by communicating with the controller via Radio Frequency (RF).



SCOPE AND RESEARCH

We spent a majority of our research time finding electronic components that best suited our needs. In order to do so, we made use of decision matrices.

For Decision Matrices: On a scale of 1 to 3 (worst -> best)

	Non-Selected Components
	Selected Components

See Appendix for relevant data sheets.

	Memory	Programmability	Support	Power Efficiency	Compatibility	Cost	Score
Arduino Uno R3	1	3	3	2	3	2	14
TM4C123GH6PM	3	1	2	2	2	2	12

Ultimately, we chose the Arduino Uno R3 because of its community support, simple programming experience and compatibility with various parts that are vital to our project.

Wireless Communication vs. Wired Communication

	Usability	Ease of Implementation	Data Security	Application to Our Project	Score
Wireless	3	2	1	3	9
Wired	1	3	3	1	8

For the purposes of our project, implementing wireless communication allows for remote sensors to be placed freely about a room and increases the usability of our project.

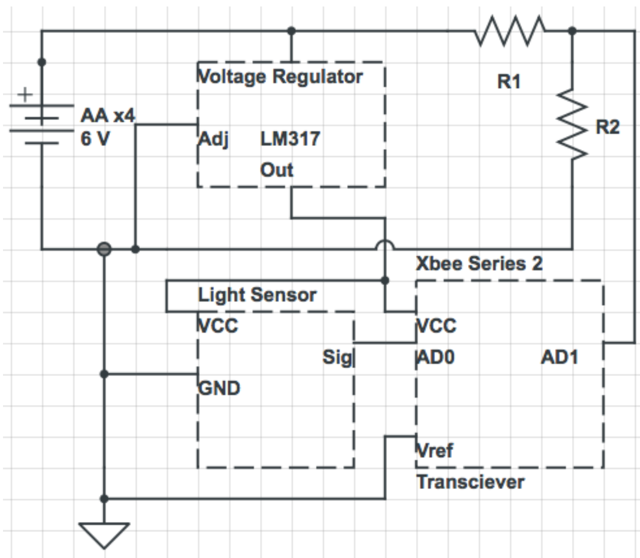
Bluetooth vs. Xbee Series 2 vs. NRF24

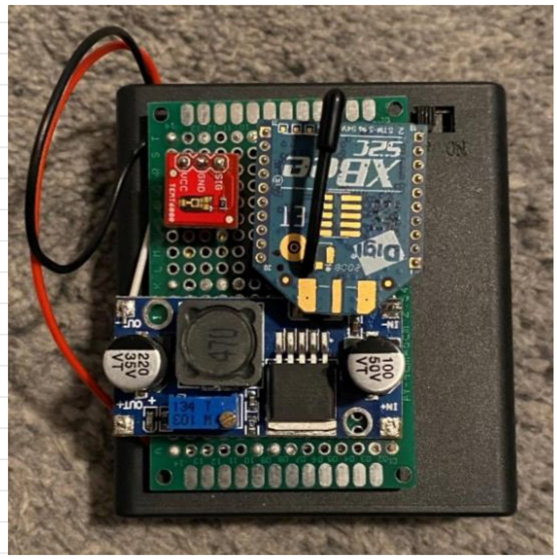
	Ease of Pairing	Range	Connectivity to Multiple Devices	Cost	Simplicity	Power Efficiency	Score
Bluetooth	3	1	1	3	3	2	13
Xbee Series 2	3	3	3	1	3	3	16
NRF24	3	3	2	2	1	3	14

From our options regarding wireless communication, we decided to use the Xbee Series 2 modules. These modules can be paired with an adapter shield and communicate using 2.4 GHz RF frequencies. They provide an easy method of interfacing multiple sensors and gathering data with our Arduino Uno R3. The Xbee Series 2 also has sleep mode capabilities and is very energy efficient which should allow for long battery life.

Computer Engineering

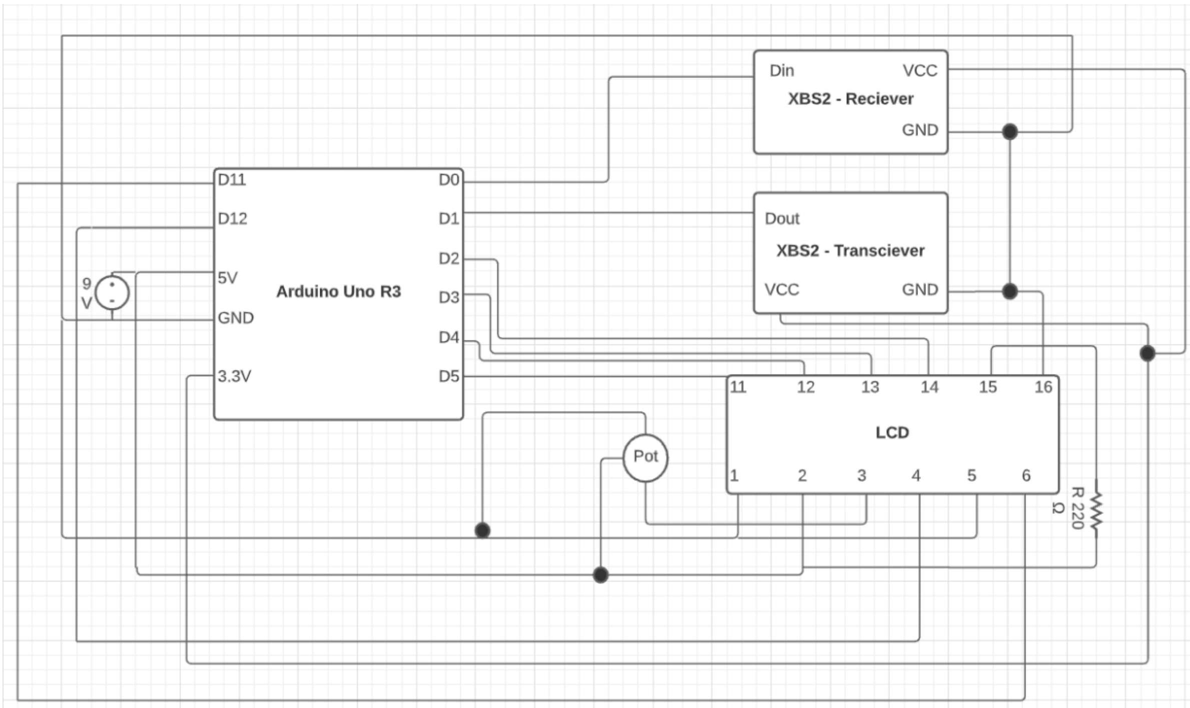
Remote Sensor Units:





The remote sensor units were designed to measure the ambient light as well as monitor the battery life. They were designed with the intent of being placed anywhere in a room and are capable of transmitting to the central unit from up to 300ft away.

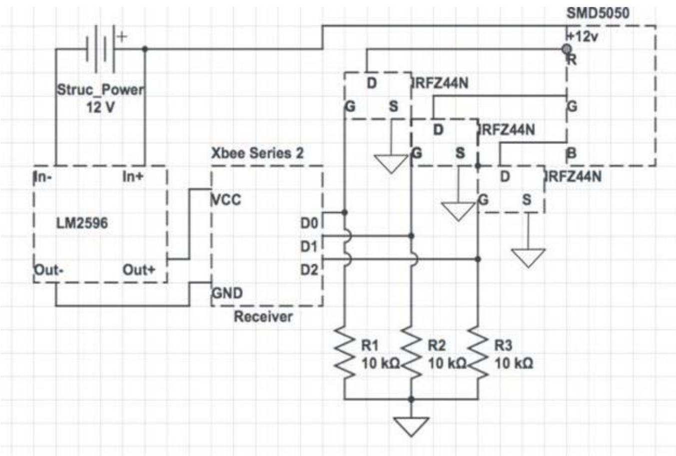
Central Control Unit:

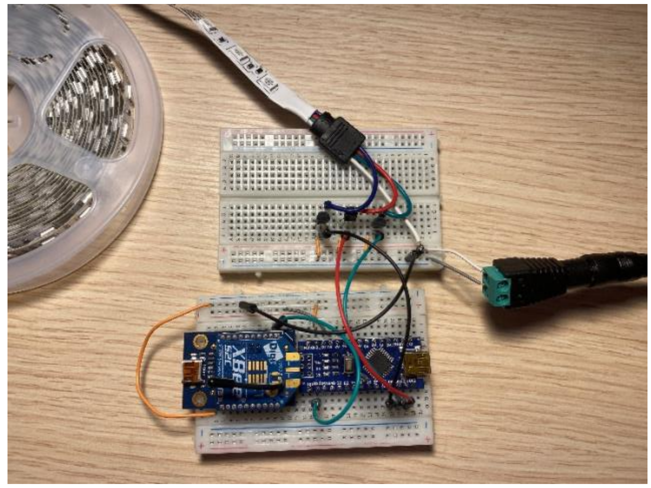


Externally: The central control unit was designed for user-friendly interaction with the system. The user may set their desired light levels and monitor the entire system using the keypad and lcd screen.

Internally: The system communicates with the remote sensor units to gather ambient lighting levels, calculates the new lighting requirements, and updates the artificial lighting accordingly.

Lighting Unit:





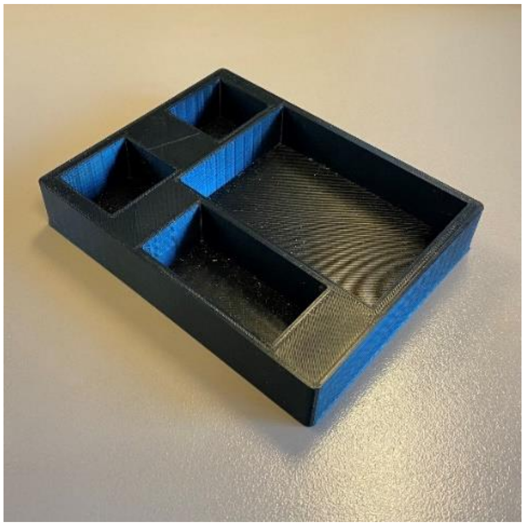
The lighting unit shown above was implemented on Model 1. The Xbee receives the desired output level from the ELC main unit that is then fed to LED strip. It slowly fades to the desired output incrementing one PWM value (0-255) once every 10ms to create a seamless and near unnoticeable change to the appropriate light level. Additionally, this slow transition and delay mitigates the effect of a rapid-error feedback loop.

ENGINEERING WORK AND TASKS CONT.

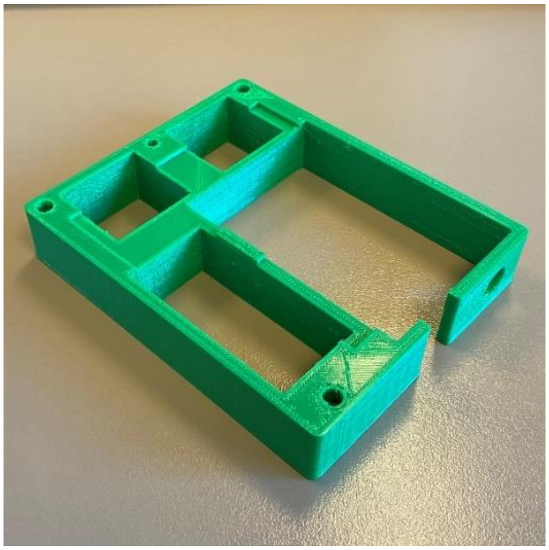
Mechanical Engineering

All throughout the process, the Mechanical Engineer team compiled CAD models of components to compare to the housing model for reference on optimization of layout and sizing. Difficulties lied within the communicative process between the CPEN and ME teams as models were everchanging due to new information/ideas presenting themselves.

Enclosure Model

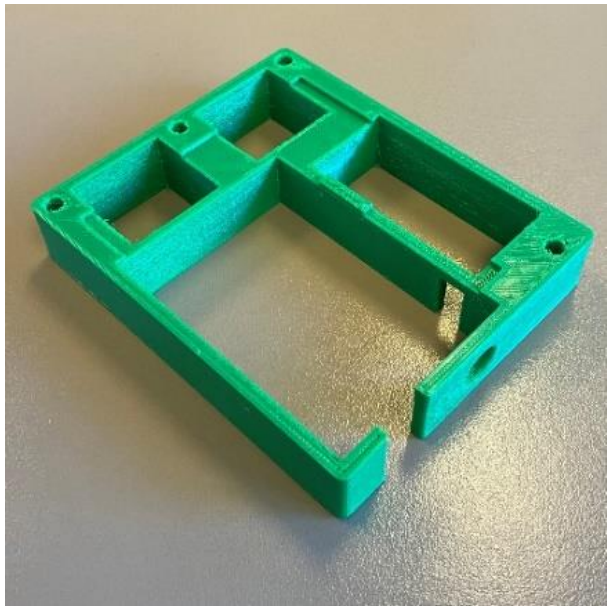


First revision: Rough concept that solely considered sizing of components on the main level of the enclosure



Second revision: Included holes where necessary for electrical components and fastening to other components of enclosure. Began correcting sizing with tolerances and inaccuracies inherent with the 3D-printing process.

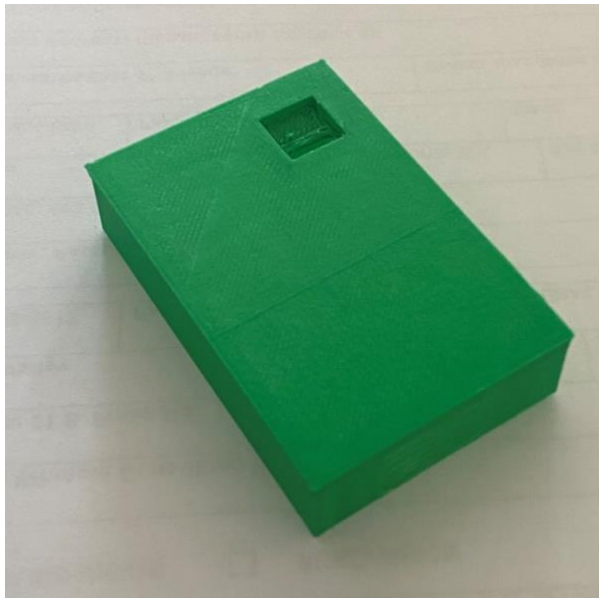
ENGINEERING WORK AND TASKS CONT.



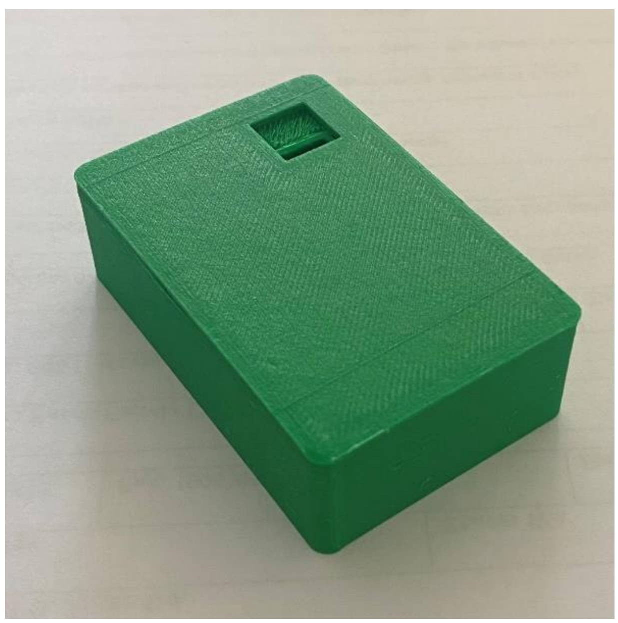
Third revision: Finalized the layout of components and corrected any last-minute tolerancing issues.

Final Product

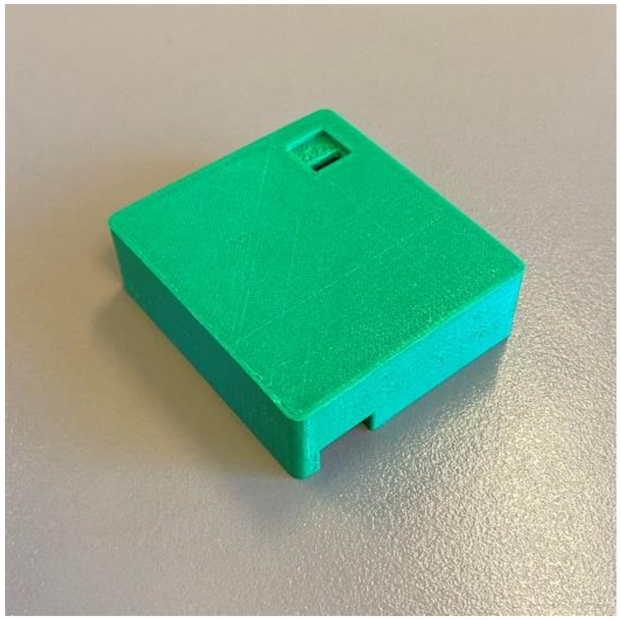
Sensor Housing Model



First revision: Rough model for various component sizing/conceptualization.



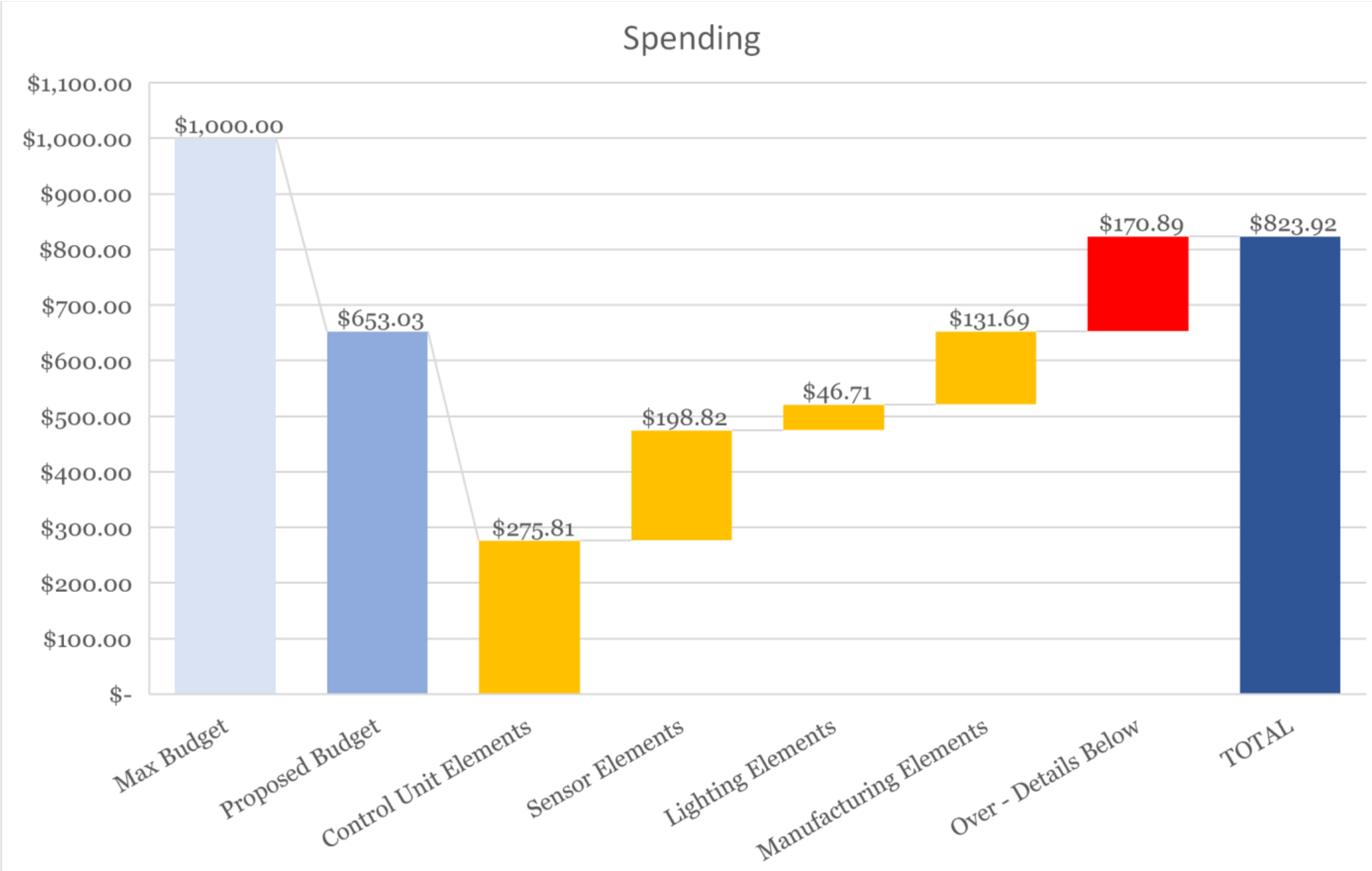
Second revision: Increased wall sizing to allow for better structural integrity. Furthered understanding of 3D-printing process.



Third revision: Included hole for power switch as well as more appropriate sizing against utilized components.

Final product

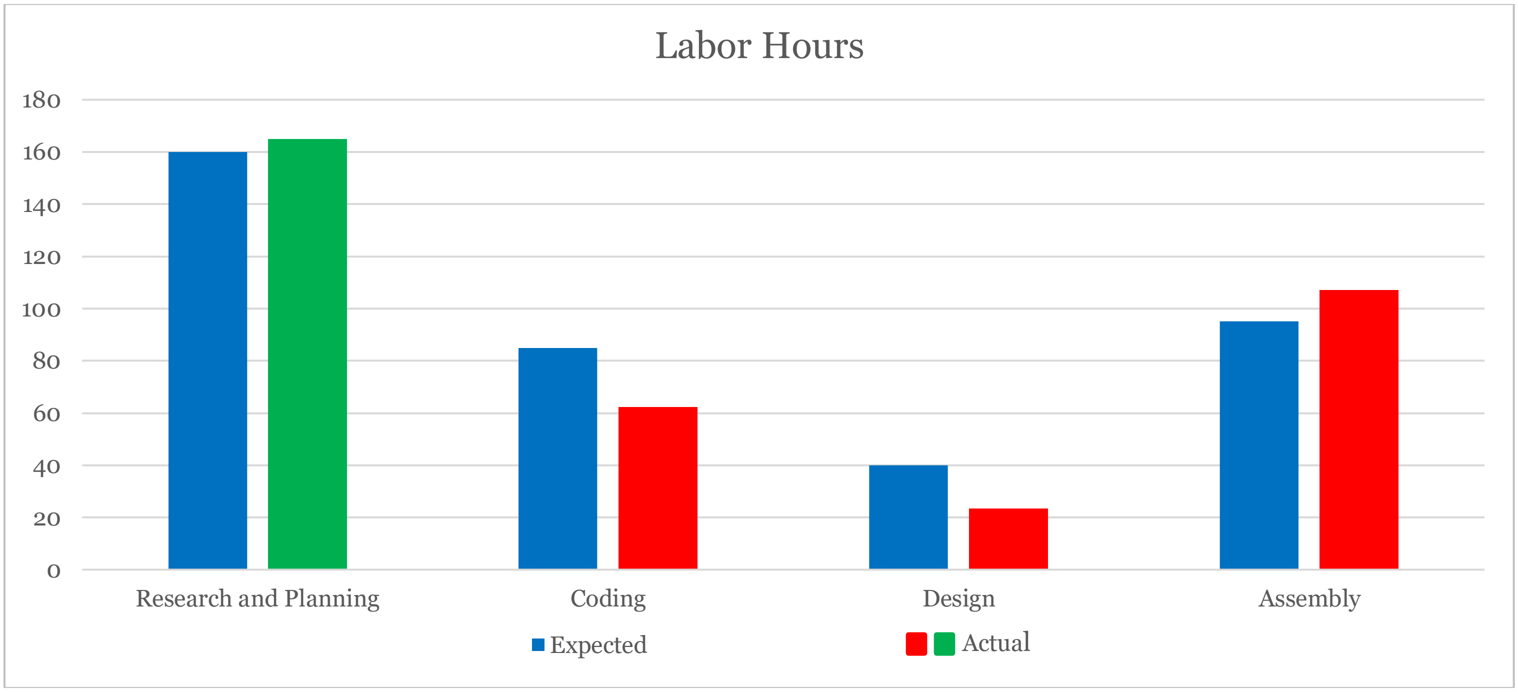
BUDGET



Depicted by the waterfall chart above, our spending went over our initially proposed budget but remained below our total allotted funds for this project. These extra costs were distributed among a number of different categories:

- Additional components (adapters, cables, etc.) which, in turn, improved efficiency and productivity.
- Replacements due to parts missing in transit (buttons, power supplies, assembly items).
- Finally, unforeseen additional costs in miscellaneous items that were not specific to the Controller System such as the Model 1 enclosure, Velcro, additional LED strips, etc.

Several ordered components went undelivered: buttons, voltage regulators, Velcro, and ABS Spool.



Green indicates within +/- 10% margin, Red indicates outside.

The chart depicts how our executed labor hours stacked up to the estimated hours for each phase of the project. For the initial phase, Research and Planning, we were slightly over the expected budget but within a decent margin (green). However, for both the Coding and Design phases of the project we fell well under budget (red) due to significant parts delays and roadblocks that this delay caused. After receiving the required components to move forward with development, it is evident that significant time was required to make up for the parts delay which then carried us well over our timing budget (Red) in the final phase of the project. This correction brought us closer to the overall expected labor hours required for this project but ultimately, we finished under the total time by a fair margin (under by 6%). Finishing under the hours budget demonstrates that our estimates were correct despite setbacks and if the project were to have reimbursable labor, we would have remained under budget building trust with our client.

EXECUTED CONTINGENCY PLANS

Lead Times

The project required many different parts from many different sources to be shipped so that the final product could be made. In theory the parts would be delivered in a timely manner but how it worked out led to many of the parts either not arriving or arriving quite late. To mitigate this the team barrowed parts to start working on the project sooner.

Manufacturing

The use of a 3D printer was a major component in making of the physical product and as such the dimensions of the parts being printed needed to be precise as to not waste material or time. When we started, we started printing test designs without the actual parts and as such used the manufacturer dimensions for the parts or cad models if available. When the parts arrived, they were either not the size stated or different parts entirely to combat this problem we measured the parts we received and designed around those dimensions.

Electronic Design

Electronic design was able to continue through the increasing lead times but was ultimately setback by receiving incorrect parts. Electronic redesigns were also necessary as component functionality had not met our researched standards. Additionally, our hopes to maximize the capabilities of our wireless network proved to be more difficult than we initially estimated, however we found efficiency through simplicity.

MARKET

Based on constraints, Eyeris has successfully breeched the “proof of concept” phase, but there are still user interface (UI) and ergonomic refinement needed before going to market.

As the total cost of the prototype Eyeris model exceeds $375, it would be less than optimal to follow the same practices in a mass-produced version. Each component that would be selected for the bulk production model would serve an individual purpose, rather than have additional features that are disregarded because they are extraneous to the concept. Space would be more optimized for this reason, thus creating a smaller physical footprint and lowering costs for components and plastic enclosure.

Along these same lines, there would be more practical improvements made. Instead of the costly and time-consuming 3D-printing process for each unit, if there were enough models projected for production, the best line of action would be to produce a similar enclosure via plastic injection molding. If this is the direction chosen, costs upfront would be increased, but overhead would remain low in the long run as well as cost per unit diminishing as more are produced. Although nothing has been quoted by approved manufacturers, we can expect cost of productions to be lowered to anywhere in the realm of $50 per unit depending on volumes produced.

TAKEAWAYS

- **Communication is key.** We learned that better communication helps to provide the best guidance for everyone on the team.
- **Projections tend to be inaccurate (but are a good basis).** We learned that often our projections were wrong, and that is okay. That some tasks run smoother or are cheaper than we thought, and others take a little more time and maybe more capital.
- **The scope may change as the project progresses.** As we progressed through our project, we gained a better understanding of the problem that our solution solves. As a result, the scope of the project needed to change to fit a list of evolving requirements.
- **Start early, fail early, fail fast.** We learned the most after giving our best attempt and failing… many times. Finding many ways to not solve our problems ultimately guided us to our best solutions.
- **Controlling the narrative is key.** Explaining the events from your perspective in an honest, cordial, and clear manner. Why you did the things you did, and why that worked out for you (or maybe why it did not).
- **Accept feedback and carefully consider its implications.** Asking for feedback can be super useful when 'the well has run dry'. It is important to understand how that feedback will affect the overall goal.

APPENDIX AND REFERENCES

Appendix A

Research References (Datasheets):

Arduino Uno R3
Xbee Series 2

Appendix B

Component List (Comprehensive):

	Item	Cost	Quantity	Subtotal	Link
Control Unit	Arduino Uno	23.00	2	46.00	Arduino Uno Rev3
	Mini Breadboards (6 Pack)	9.00	2	18.00	Breadboards - 6 Pack
	9V Battery (8 Pack)	12.00	1	12.00	9V Battery
	Buttons	1.98	2	3.96	Buttons
	LCD Screen	25.00	1	25.00	LCD
	USB C to USB 2.0	7.00	1	7.00	Mac Adapter
	USB to USB 2.0	9.00	1	9.00	Windows Adapter
Sensors	Xbee Modules	23.00	6	138.00	Xbee Modules
	Xbee Adapter	30.00	6	180.00	Xbee Adapter
	USB to Mini USB	7.00	1	7.00	Micro USB
	USB C to Mini USB	7.49	1	7.49	USB C
	Light Sensor	4.25	6	25.50	Light Sensor
	Voltage Regulator	0.35	6	2.10	LM317
	Motion Sensor (5 Pack)	9.00	1	9.00	PIR Sensor
	AA Batteries (48 Pack)	15.49	1	15.49	Batteries
	Battery Holders (2 Pack)	6.00	3	18.00	Holders
Manufacturing	ABS Spool (Black)	25.50	2	51.00	Printing Materials
	Painters Tape	5.64	1	5.64	Tape
	Scraper	13.30	1	13.30	Scraper
Lights	LED's (SMD 5050)	11.00	2	22.00	RGB LED
	Power Supply	12.00	1	12.00	12V Supply
	MOSFET (Logic Level)	1.09	3	3.27	MOSFET
	Voltage Regulator	11.00	1	11.00	12V-3V Potentiometer
Testing Enclosure	Clear Plastic Display Box	27.99	1	27.99	Display Box
	Construction Paper	12.55	1	12.55	Paper
	Velcro	12.46	1	12.46	Velcro
New Additions	24V Adapter	24.99	1	24.99	24V Adapter
	24V Transistors	11.49	1	11.49	Transistors
	LCD Screen v2	16.32	2	32.64	LCDScreenv2
	ALITOVE LED Strip Lights	34.99	1	34.99	LEDs

APPENDIX AND REFERENCES

Appendix C

Component List (Controller System Only):

	Item	Cost	Quantity	Subtotal	Link
Control Unit	Arduino Uno	23.00	2	46.00	Arduino Uno Rev3
	9V Battery (8 Pack)	12.00	1	12.00	9V Battery
	Buttons	1.98	2	3.96	Buttons
	LCD Screen v2	16.32	2	32.64	LCDScreenv2
Sensors	Xbee Modules	23.00	6	138.00	Xbee Modules
	Light Sensor	4.25	6	25.50	Light Sensor
	Voltage Regulator	0.35	6	2.10	LM317
	Motion Sensor (5 Pack)	9.00	1	9.00	PIR Sensor
	AA Batteries (48 Pack)	15.49	1	15.49	Batteries
	Battery Holders (2 Pack)	6.00	3	18.00	Holders
Lights	Voltage Regulator	11.00	1	11.00	12V-3V Potentiometer
	24V Adapter	11.99	1	11.99	24V Adapter
	24V Transistors	11.49	1	11.49	Transistors
	ALITOVE LED Strip Lights	34.99	1	34.99	LEDs

Appendix D

Github Repository (Relevant Code):

Eyeris Light Controller (ELC)