

Wirelessly Charged Nursery Light

FINAL REPORT

TEAM 22

CLIENT/ADVISOR: CHENG HUANG

JOSH HOLLOWAY / TEAM LEAD

THOMAS YOUHN / ELECTRICAL LEAD

WILLIAM SNYDER / TESTER

LOGAN FARMER / FACILITATOR

ALEXANDROS PSOMAS / PRODUCT DESIGNER

WYATT RAYL / COMPONENT DESIGN

sdmay24-22@iastate.edu

<https://sdmay24-22.sd.ece.iastate.edu>

Table of Contents

Table of Contents	2
1 Introduction / Background	4
1.1 Problem Statement	4
1.2 Inteded Users and Uses	4
1.3 Work in Context	4
2 Revised Design	5
2.1 Summary of Requirements	5
2.2 Development Standards & Practices Used	5
2.3 Security Concerns and Countermeasures	6
2.4 Design Evolution	6
3 Implementation Details	8
3.1 Detailed Design	8
3.1.1 Design Context	8
3.1.2 Design Complexity	8
3.1.3 Materials	9
3.1.4 Physical Components	9
3.1.5 Features	10
3.1.6 Circuit Design	10
3.2 Description of Functionality	11
3.3 Notes on Implementation	11
4 Testing	12
4.1 Unit Testing	12
4.2 Interface Testing	12
4.3 Integration Testing	13
4.4 System Testing	13
4.4.1 Circuit Design	13
4.4.2 Software Design	13
4.4.3 Physical Design	13
4.5 Regression Testing	13
4.6 Acceptance Testing	14
4.8 Results	14
4.8.1 Touch Sensor Results	14
4.8.2 Arduino Nano Every Results	15
4.8.3 LED Strip Results	15
4.8.4 Battery Charging Unit Results	16
4.8.5 Light Diffusal Testing	16
4.8.6 Hardware in Product Housing Results	17
5 Broader Context	18
6 Conclusions	19
6.1 Review Progress	19
6.2 Application Value	19

6.3 Potential Future Steps	20
7 Appendices	21
7.1 Operation Manual	21
6.1.1 Adujsting the Arduino Code	21
7.1.2 Uploading the Arduino Code	21
7.1.3 Using the Light	22
7.1.4 Charging the Light	24
7.1.5 Changing any Hardware	24
7.2 Alternative/Initial Versions of Design	26
7.3 Other Considerations	28
6.4 Code	28
7.5 References	30

1 Introduction / Background

1.1 PROBLEM STATEMENT

The hardest thing for a parent to do is let their young child out of their sight. Ensuring safety at all times is difficult, and every night you want to ensure a safe environment for your child. This project's goal is to bring an easy-to-use, safe, sensory nursery light that is wirelessly charged - ensuring no loose cords to help keep the child safe throughout the night.

1.2 INTENDED USERS AND USES

The intended users of this product are parents or caregivers of small children. It will most likely be kept inside a nursery room or just a small child's bedroom. The main scenario where our product would be most beneficial would be either late at night or during bedtime and the parent or caregiver quickly comes into the room to care for the baby or child. Whether that be a diaper change, quick bottle feeding, reading bedtime stories, or putting them back to bed. The 10-minute timer should be plenty of time to get these tasks done and slowly turn off not to disturb the child. The timer also allows the user to leave the room or area where the light is left and not worry about having to walk back to turn it off. The portability gives the user the ability to bring the light wherever, like to a changing table, nightstand, or chair in the room to have the light where it is most needed and the wireless charging capability allows for easy charging without having to fumble with a cable to get it charged; It can just be placed on any wireless charger as someone walks by.

1.3 WORK IN CONTEXT

When looking at other nursery and night lights available on the market today, there is a large variety of styles available and many come with a couple of similar features. For example, there are a lot of options for different sizes and shapes. These include simple cylinders or spheres or more intricate designs like different animals and other non-standard shapes. While most other options have multiple lighting levels and touch controls like our design, there are a few features that are found in other products that we will not have. These include having different color options for the lights, a remote control, and some even have apps to control them. There are a few features we will have that seem less prevalent in the market today. Firstly is the wirelessly charging aspect. There are a few options with proprietary wireless charging docks whereas our design will work with any Qi charger that it can fit. Also, our design uses an Arduino Nano which means that it is completely programmable and can be programmed to add more features and tweak some of the options that we will have.

Pros	Cons
Wireless Charging	No Remote Control
Completely Programmable	No Lighting Color Options
Auto Power Off	No App Control

2 Revised Design

2.1 SUMMARY OF REQUIREMENTS

Functional Requirements:

- Design must be battery-powered and charged wirelessly.
- Design must be able to alternate between two brightness levels by tapping on it (constraint).
- Light must automatically dim and shut off after ten minutes (constraint).
- At all times when the light is turned on/off it must do so smoothly as to not wake the baby.

Resource Requirements:

- Project must utilize Arduino and remain programmable.
- Project must use a Lithium Polymer battery.

Qualitative Aesthetic Requirements:

- Design must be soft and easily approachable for children.
- Design must emit amber or other warm levels of light.
- Design is not required to be commercialized or mass-manufactured.

Performance Requirements:

- Battery life lasts longer than three days during standard use (constraint).

2.2 DEVELOPMENT STANDARDS & PRACTICES USED

Qi2 - The Qi2 wireless charging standard is required to ensure we approach the wireless charging aspect of the nursery light properly and ensure proper efficiency and alignment.

IEEE 1855 - Standard for Fuzzy Markup Language, or Arduino code. As this project will remain programmable and be implemented with an Arduino Nano, this standard is needed.

IEEE 370 - Standard for printed circuit boards, which is planned to be utilized in this design as a shield to the Arduino Nano.

IEEE 1679.1 - Standard guide for characterization and usage of Lithium-Based Batteries in stationary applications, we plan to use a Lithium Polymer Battery so this standard is needed.

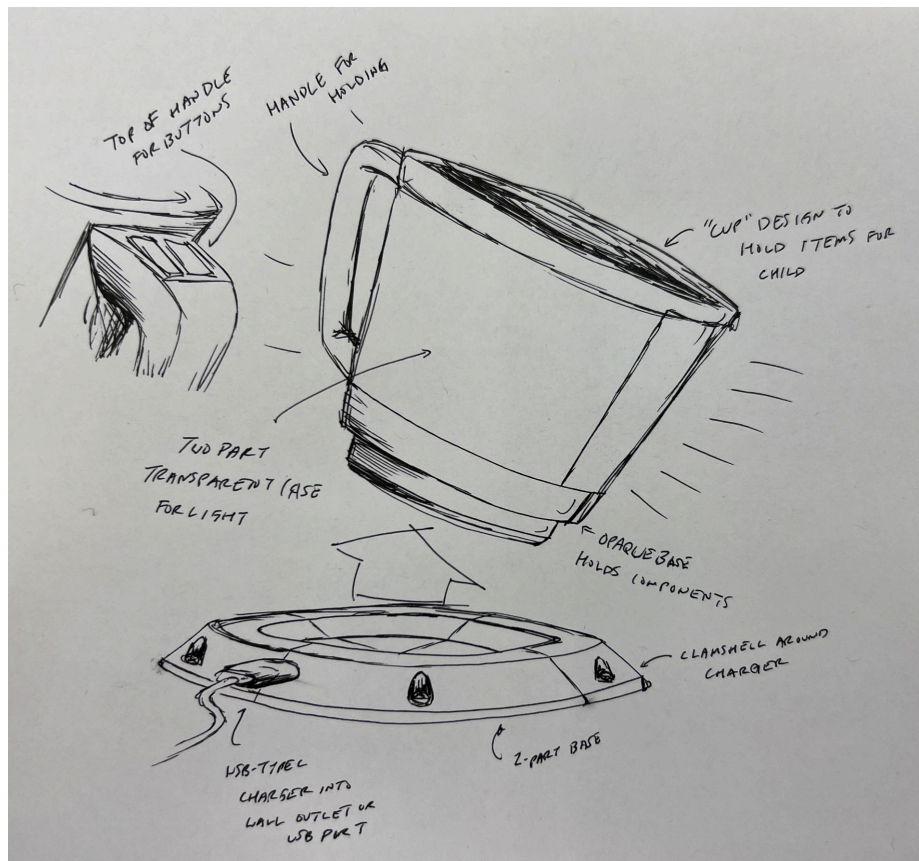
IEEE 2700 - Sensor parameters for touch-based applications, helpful for the touch sensors applied to this project for power/brightness level.

2.3 SECURITY CONCERNS AND COUNTERMEASURES

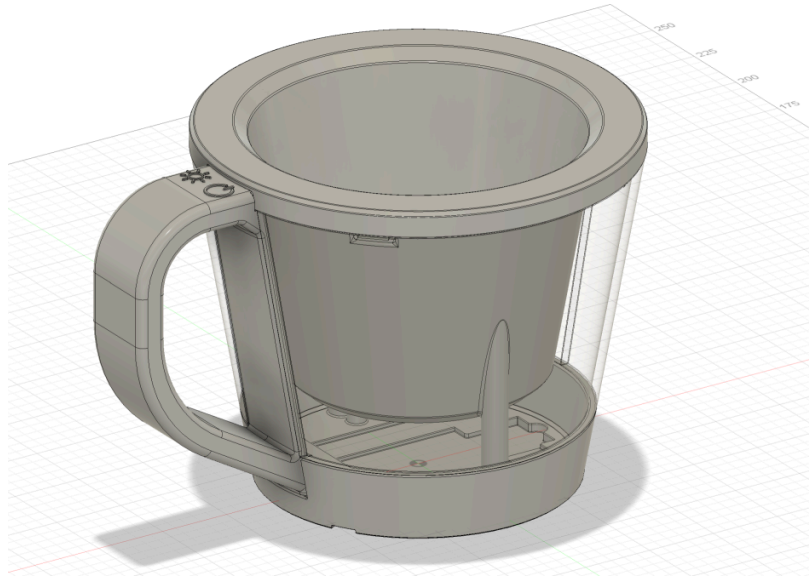
This design inherently avoids cybersecurity risks and vulnerabilities, however, most concerns regarding the design were heat management and design safety due to the nature of being used around children. Heat management is regulated by the open-air design and easily removable outer covers to reduce enclosed heat buildup; and the overall design is soft and bottom-weighted to reduce tipping, falling, and other gravity-induced dangers posed to the child. Also, the design has no sharp edges to create a soft and approachable piece that should not harm any users.

2.4 DESIGN EVOLUTION

Using our foundational basis from 491, our design went from sketch to rough block out to functional prototype over a few short months. Our circuit design detailed in Section 3.1 remained largely similar, however, our focus changed to our product housing over the spring semester and this design had the most evolution. We started with a rough layout sketch that detailed all functionality and where we wanted our design to go.



From this, we were able to develop a rough layout of general shapes and components for review for our client. After confirmation, we used this layout and our known hardware to develop our final CAD files for printing and implementation.



This design features an enclosed bottom puck that fits all relevant hardware with a connected handle with slots for both touch sensors. This handle/puck connection is permanent, allowing disassembly for showcase without separating the circuit. The inner cup attaches by hardware and locks into the handle, reducing any risk of the handle separating from the main body or leaving the product feeling loose. Finally, the exterior glass sections are magnetized and printed in FDM transparent filament, which diffuses the light from our LED strips.



3 Implementation Details

3.1 DETAILED DESIGN

3.1.1 DESIGN CONTEXT

For the design of our nursery light, we are creating the housing and circuit that will be used. The housing will include the base shell for all the electronics as well as a cover for the light to deter damage. Our circuitry will mostly use Arduino to control the light's finer aspects, including its ability to dim over time and run on a timer.

3.1.2 DESIGN COMPLEXITY

This project can be split into 5 main challenges.

1. Dimmable LED using MOSFETs
2. Creating a PWM and Dimming Timers in Arduino
3. Battery Life and Choice
4. Wireless Charging
5. Physical Design

Focusing on them individually, each one uses varying amounts of engineering principles. The first challenge is finding a way for the Arduino to control the brightness of an LED without it being directly connected to the LED strips. This is because the Arduino cannot output the current needed to light the strips in the brightness levels wanted. To solve this issue, a MOSFET was used, one of the core blocks of electrical engineering design. This MOSFET gate will act as an input for the Arduino pin while the battery provides the current and voltage in series with the LEDs. This MOSFET will then act as a switch allowing current through when the Arduino pin is high and cutting it off when it is low. A proper MOSFET had to be selected and currents had to be calculated to not overload the battery's life. This looked at the basic square law models and made sure that the threshold voltage was correct for Arduino's output pin voltages.

Looking at the dimming of the LED would be a different problem entirely. Dimming the LED can be done by lowering the flicker frequency. The easiest way to do this would be through pulse wave modulation (PWM). Using the Arduino timer, the output pin's frequency can be raised or lowered, and the LED's frequency will be raised or lowered. This of course must still be higher than the human eye can see to make sure that the flicker is not bothersome. The lower the frequency, the dimmer the LEDs will be. This PWM also has to be able to be adjusted so that it will lower over set intervals, set by another timer, and changed due to buttons being pressed on the design. This means that the PWM must be adjustable through the Arduino software. This PWM must also be present when turning on the light since it needs to have a smooth turn-on process.

Another challenge will be getting the correct battery life for the project. The goal of the project is to have a working battery life of about three days. The main draw for the battery will be from the LEDs as they need the most current. Next, it would be the draw from the Arduino. The battery must have the life to sustain the current drawn from it over a period of 10 minutes (the approximated dimming time). This 10-minute period will have different levels of dimming so the current draw will vary. Still, this must be

calculated and to do this, the current draw at max brightness needs to be tested and gathered to pick out the correct life for the battery. Along with that, there needs to be a way to boost the battery voltage to 5 V to run both the Arduino and the LEDs. After that, the dimming currents should be tested to select a battery with the correct AmpHrs.

The next major issue is the wireless charging. This change is going to be done through magnetic induction. This needs to be researched and the correct coils need to be found. These coils then need to be tested to ensure that the input from a wall outlet will output the proper voltage to charge the battery without too much current flowing through it. The heat must also be managed as the induction coils can output a lot of heat as waste. Managing these two will lead to making sure that the charge rate is at a comfortable amount, meaning that it only takes a little bit of time to charge the battery while at the same time not making the current so high that it could rise to dangerous temperatures.

The final big part of the project is creating the physical housing for the light. This housing needs to be able to hold the LEDs, Arduino, circuit components, and the wireless charging coil. With that, the coil must also be able to interact with the other coil to start the charging process. The goal is to also have it fit comfortably and easily so that small adjustments will continue the changing process accidentally. The design needs to be able to withstand any heat that might be output by the circuit without any damage. Finally, to help warm the light, a clear filament will be used to further dim the LEDs to not bother the children using the lights.

3.1.3 MATERIALS

Housing will mostly be additive plastic with metal fasteners. This reduces lifetime due to constant wear on the places the fasteners are applied but as the product uses LEDs and is wireless, it should last a long time without needing to be opened up. We will use a mix of opaque and translucent plastics for our housing as well.

3.1.4 PHYSICAL COMPONENTS

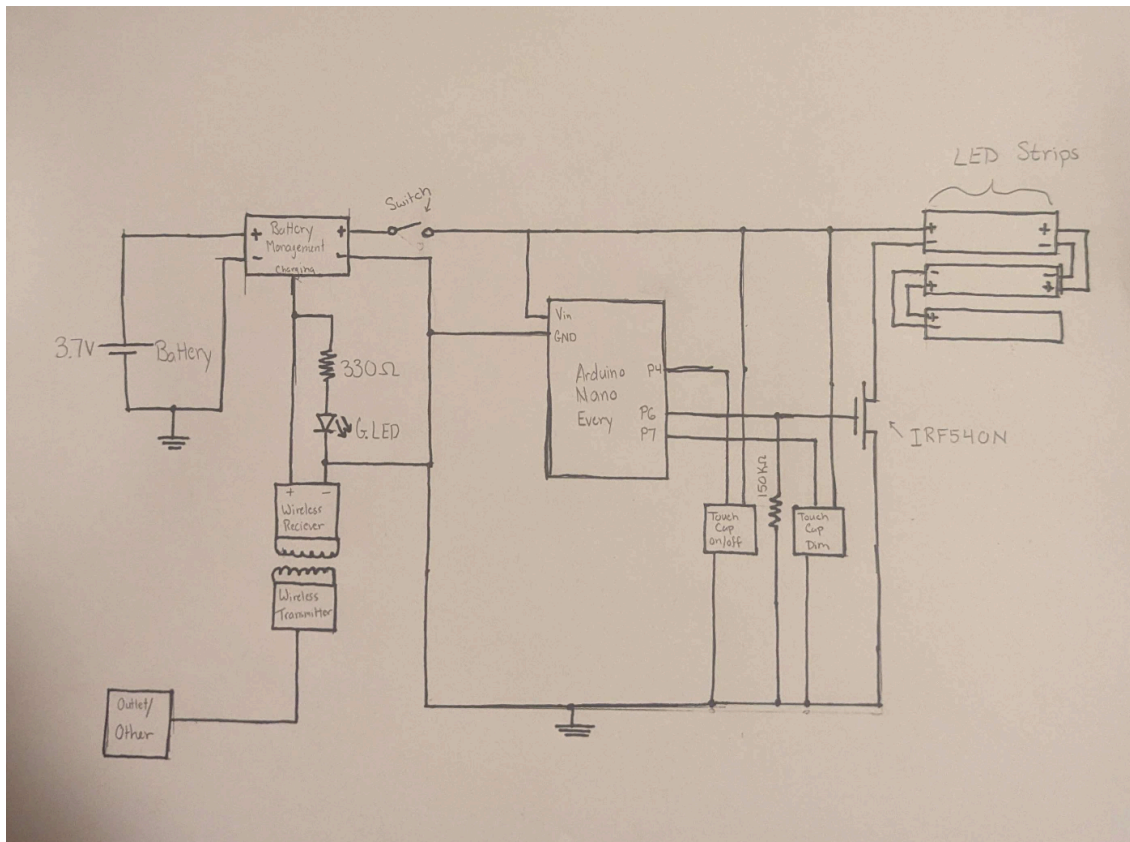
We will utilize the following physical components to make our circuit:

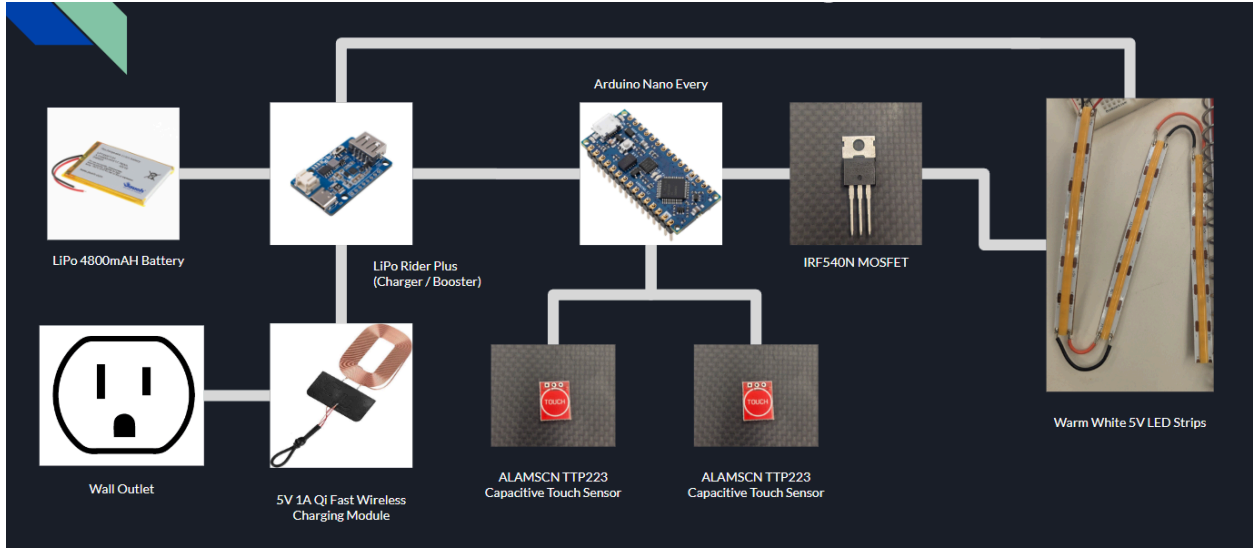
- Arduino
- Li-Po battery
- Battery charging/protection module
- Touch sensors
- Fasteners
- Housing
- Wireless receiver
- Wireless charging board
- Solder/solid core wiring

3.1.5 FEATURES

Our design features must hold to the requirements listed in section 1.2, while also following these basic tenants: Must hold battery, Arduino, and receiver. It also must have space for the touch sensors as well as something to diffuse the light from the LEDs.

3.1.6 CIRCUIT DESIGN





3.2 DESCRIPTION OF FUNCTIONALITY

Our product was designed with specific functionality in mind. We were given a list of functionality requirements from our client which includes being wirelessly charged, battery-powered, having adjustable brightness levels, auto dim and auto shut off, and smooth turn on and turn off. The requirements are self-explanatory but to achieve these functionalities we required a battery so our product could be picked up and moved, a wireless charging receiver so that it can charge on any Qi charging pad, a microcontroller, and code to control the brightness of the light, smooth transitions, and auto shutoff.

3.3 NOTES ON IMPLEMENTATION

To implement these functionalities we used an Arduino Nano Every as our microcontroller. We picked this because we were familiar with how to code it and it was small and convenient. Using the Arduino Nano we were able to program it to use a pulse wave modulation (PWM) circuit to change the brightness of our LED light. We were also able to use the Arduino Nano to work with our two capacitive buttons to turn on the device and change the brightness settings having a smooth transition between brightness levels. We also added a battery management system to incorporate the battery with our wireless receiver.

4 Testing

The testing plan can be broken into six major parts, as seen in the sections below. Each of these tests focuses on a specific portion of the project. It starts with testing the physical design and then moving into the user interface. Once those are done, the focus shifts to integrating the two previous tests and ensuring they work together. With that finished, the testing moves onto the system as a whole and ensures that anything added does not damage the old process. Once these checks are done, the final tests make sure that the final prototype will pass all of the conditions given in the project procedure.

4.1 UNIT TESTING

1. Touch sensor
 - Tested by using a breadboard to connect a voltage source to an LED through the touch sensor and testing what materials and thickness can be used with it.
2. Light strip
 - Tested by connecting to a power source and measuring the current. Then, we will change the supplied voltage and read the change in current.
3. Wireless charger
 - Tested by connecting the transmitter to power using a USB cable, and holding the receiver over the transmitter while measuring the voltage and varying the distance between the coils.
4. Arduino Nano Every
 - Tested by connecting to the downloadable Arduino IDE, test run with the provided “Blink” code to ensure onboard LED functions as well as uploading functionality on the main board.

4.2 INTERFACE TESTING

1. Arduino -> Touch sensor interface
 - We will test the touch sensors through different materials like plastic to see which materials we want to make the shell out of. This information will help us ensure the sensor is always working properly and protected.
2. Arduino -> Light strip interface
 - We will test the ability to change brightness and dim the light over time in order to fulfill the requirements given to us to ensure optimal usage.
3. Wireless charger -> Lithium-polymer battery interface

- We will test the connection between the battery and the circuit to bring the usage amount as low as possible to ensure the light can be used as much as possible between charges.

4.3 INTEGRATION TESTING

The main integration for this project is the wirelessly charging battery circuit to the Arduino system, connecting to the touch sensors and the light strip. By completing integration tests, we know the individual circuits work, and combining them requires tests on voltage and current levels being input and output from the Arduino board itself. These voltage levels need to be consistent with the chosen battery. Along with that, the current draw from the battery should allow for up to three days of regular use. Everything must be connected to the Arduino, and the functionality should continue to work. This testing will be done through smaller modules, building on each previous part. These modules will be compiled into the working script that will allow the battery to be charged, turn on the LEDs, adjust the brightness, have a self-sleep mode, and so on.

4.4 SYSTEM TESTING

Our project will separate system testing into three different areas and individually test each section before integration. These sections and their respective strategies are listed below:

4.4.1 CIRCUIT DESIGN

To test the circuit design, we will measure the current draw and the temperature of all the components. To do this, we will use a temperature sensor and a multimeter to meet our target numbers: a temperature of 70 degrees and an overall current draw of 1.3 Amps.

4.4.2 SOFTWARE DESIGN

To test the software, we will use buttons and an LED strip to make sure the code is working properly. What the software needs to do is turn on the lights, turn off the lights, dim the lights over 10 minutes, have different settings for the lights, and be able to switch between them

4.4.3 PHYSICAL DESIGN

To test the physical design, we will make sure that the design can fit all of the components, be able to be held for easy movement, have a space for a wireless charger, and be easy to assemble.

4.5 REGRESSION TESTING

To make sure that new additions keep the old functionalities, we will make sure to test all of the features after each thing is added to make sure that they still work. If the new features affect an older one, then we will have to review the new functions to find out what is causing it before moving on to the next thing. The critical features we will need to ensure do not break are the touch sensor and the lights. The touch sensor is the user's interface to use the nursery light. If that function ever stops working, then the other functions won't be able to work at all. So when writing code or adding components, we will make sure to check every function still works, and we will not move on until we know that they do.

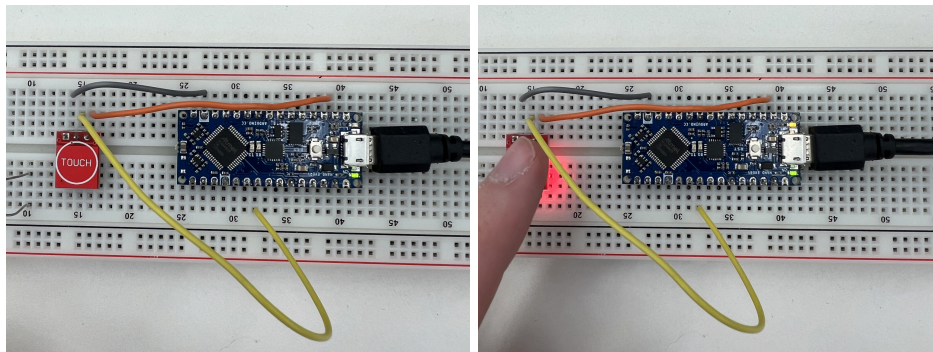
4.6 ACCEPTANCE TESTING

Throughout the testing process, we aim to present and involve the client in ensuring all data meets standards and double-check that the requirements remain the same as the original meeting. The goal is to ensure that throughout the original testing procedure, we do not delve down a path that is not desired and also ensure that the client starts seeing tangible results.

- We will test to ensure that battery life lasts on normal use up to three days, minimum.
- We will test that the outer housing is easy to use and assemble.
- We will test programmability and the ability to easily scale values for operation changes.
- We will test heat output to ensure the light is “safe.”
- We will ensure charging can function from 0% to 100%.
- We will test the individual functions of all components.

4.8 RESULTS

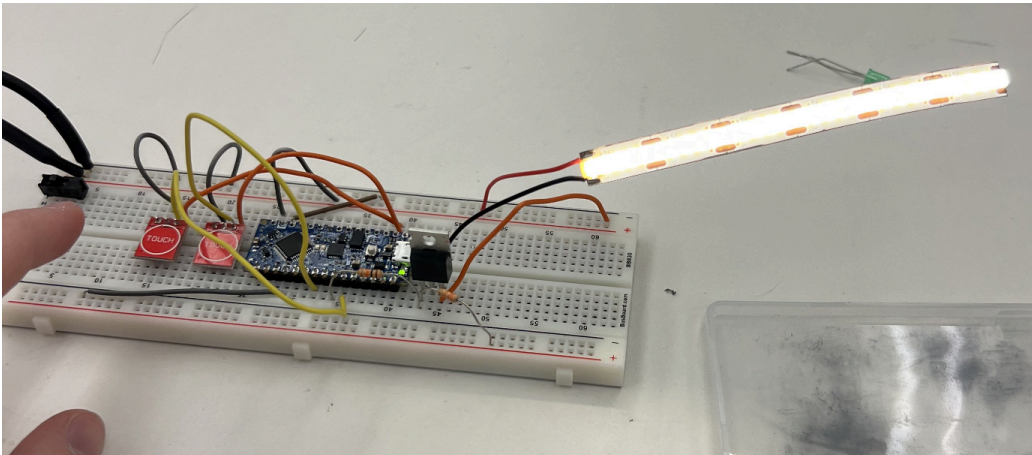
4.8.1 TOUCH SENSOR RESULTS



To test the functionality of the touch sensor, we developed a simplistic code system that, upon pressing the button, turned on the onboard LED (seen top right with a yellow glow) for one minute before turning it off. This doubled as a test for the Arduino’s timer system we would need later.

We also concluded that the sensor functions with a 0.125” layer on top of it, allowing us to conceal it within the project casing while retaining functionality. We can wire two touch sensors in parallel to the Arduino’s built-in 3.3 V power supply, which sends a digital signal to a digital receiver pin when pressed to start the built-in code.

4.8.2 ARDUINO NANO EVERY RESULTS



The next test was finding the currents that would be drawn by the Arduino Nano and making sure that the outputs were working as intended. These need to be found so the overall current draw can be calculated to find the correct battery. Along with that, the output voltages of the I/O were also found.

This test also doubled as a total current test. This is because the Arduino Nano's current draw will vary based on the load. Testing this was very simple. The structure connects a DC power supply to the Arduino Nano's power supply and then the positive input of the LED strip. The negative port of the LED is then connected to a MOSFET's drain. The source of this nMOS is connected to the ground, and its gate is connected to a voltage divider powered by the PWM wave from the Nano.

With this test structure, the I/O pin voltage was found to be 4.6 V which is close to 5 V which is what the datasheet provided. This value is used to make sure that the voltage divider provides an accurate voltage and the MOSFET will be on for the selected brightness levels. After that, the load was disconnected from the Nano and its base current draw was tested to be 27 mA. The load was once again added for three small LED strips and the current drawn by the Nano was found to be 30 mA. This current is much smaller than the LED currents found in the next section which is great as this should not be drawing much power.

4.8.3 LED STRIP RESULTS

The final test needed for battery calculations was the current drawn by the LED strips. These strips consume the vast majority of the power for this circuit. Getting these currents will give a much better idea of what battery could power this structure for 3 days. The difference between this test and the previous one was that the total current was found for each brightness level since as the PWM changes, the current controlling MOSFET will change the current and therefore the brightness of the LED strips.

Running this test once again used the DC source to provide 5 V to the circuits and a multimeter to test the currents. The light is planned to have 4 brightness levels and then the off state. During the off-state, it will only draw the 30 mA for the Nano as found in the previous section. During the other sections it will draw current from both the LEDs and the Nano's current will change with the load. This test was run a few times with the brightness levels varying to see what values would lead to a noticeable change in brightness and have a total power that would be feasible for a LiPo battery to supply. The first test was found to have currents of 1.989 A at max, 1.36 A at the next level, 0.8 A while dimming, and 0.43 A at the lowest level. This was done using the full strip and the total current for 3 days in high usage would be 7.95 Ahrs. This is much too high, but it was used as a benchmark as this test used the full LED strip.

After that, the LED strips were cut into smaller unit lengths such as the one seen in the figure in section 5.8.2. The test was then run for a single unit length. This results in values of 300 mA, 234 mA, 166 mA, and 99 mA. The total current draw for this was 2.19 Ahrs. This value is much smaller than the previous one and would work for many LiPo batteries. The issue is that one LED would not provide enough light, so to solve this issue, two more unit strips were added and the current was tested once again. These currents ended up being 730 mA, 556 mA, 385 mA, and 210 mA and had a total current draw of 4.58 Ahrs at high usage. This was an acceptable value in terms of current draw and it provided enough light to be useful in a dark room, so this LED design was chosen and a battery with 4.8 Ahr life was ordered.

4.8.4 BATTERY CHARGING UNIT RESULTS

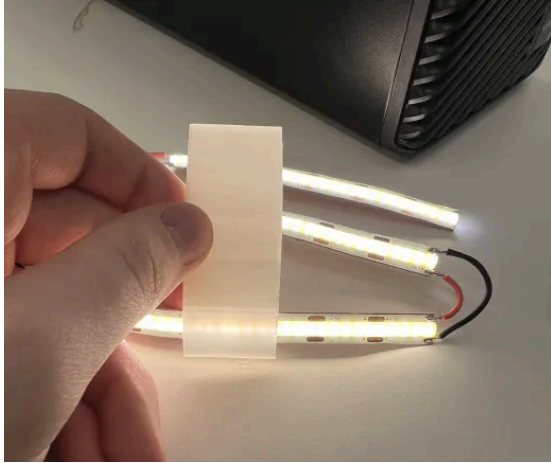
Once the battery arrived, the testing could begin. The main goal of this testing is to find out how long it would take to charge the battery to full from an empty battery. Along with this, some other important tests were making sure that there would not be an overcharging issue with the LiPo battery during magnetic charging. Before finding the charging rate, the charging circuit had to first be made, and this was simple due to the battery management model having a charging input that goes directly to the LiPo battery. With that, the circuit is simply connecting the wireless receiver to the charging port and the ground. This worked as expected, but it was noticed that it was hard to show that charging had started due to the LED on the BMM being within the structure and unable to be seen. To fix this, a simple green LED was added in parallel to show that the light is currently being charged.

With the circuit finalized, the next step is to look at the current supplied during charging and see how long it would take to reach full. The current was measured to be about 550 mA, with it varying from around 535 mA to 565 mA. Assuming that the average current will be 550 mA, then the full time to charge the 4800 mAh battery will be 8.73 hours. Though this is a longer time, the assumption is that the light will only be used during the night and therefore can be left on the charger during the day. This will provide enough time to fully charge the battery if needed but should be more than enough in the long run. During the current testing, the battery was left connected for a long time to make sure that the battery's overcharge protection was working.

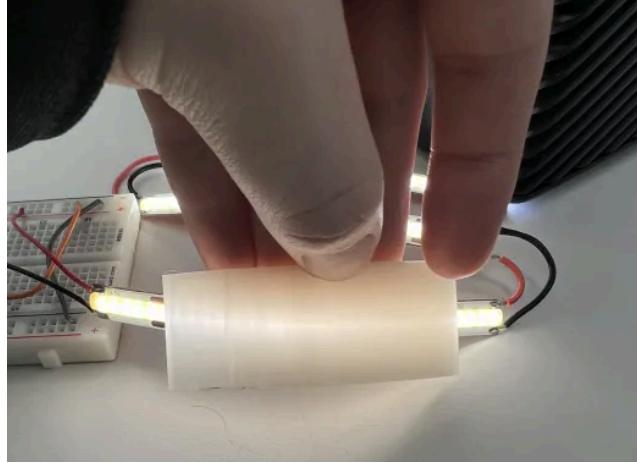
With those found and tested, the final test was to make sure that the charging works through the light's housing. Once the housing was finished, the charging was tested again. These tests showed that the charging was indeed working and providing the same average current meaning that no changes needed to be made to the charging portion to increase charging speed.

4.8.5 LIGHT DIFFUSAL TESTING

To test the thickness of our transparent FDM sections for the exterior of the nursery light, we made a small rounded section that increased overall thickness by 1mm ranging from 2 to 6 mm. Upon testing, we discovered that the 2mm section worked best, as well as that the orientation of the print lines had an effect on the diffusing properties of the plastic. When orientated normally to the direction of the light strip, the lights barely diffused if at all. However, when the print lines were parallel to the light strip they diffused very well, showcasing how we wanted to orientate the project.



Print lines normal to light strip



Print lines parallel to light strip

4.8.6 HARDWARE IN PRODUCT HOUSING RESULTS

When developing our project's casing, we built it after already knowing our finalized circuit and all the hardware that would need to fit. This alongside previous testing of heat management and our touch sensor capabilities allowed us to build out a fully functional prototype with relative ease thanks to our preparation. To test we inserted both touch sensors in each cavity in the upper handle and then tested the touch capacitive nature of the final result to ensure the project functioned. Also, we roughly tested the layout by inserting each piece of larger hardware (charging coil, battery, and proto board) to ensure our internal layout would hold everything it needed.

5 Broader Context

Our project may seem very simple and unassuming but nursery lights are a big part of having a child. Our project will address the societal needs of parents who are looking for a safe lasting nursery light. There are a lot of cheap nursery lights on the market but we aim to do what none of the others can do we are going to make the light movable unconnected from the wall so you can bring it to any room in the house or even out of the house if you go on vacation or to the grandparent's house. using wireless technology we will create a wireless charging station for this nursery light to be placed to charge. Then when it's fully charged parents won't have to worry about it for three days. Using the integrated battery and auto turn off timers our nursery light will have plenty of battery to last whatever event is keeping you from the wall.

Area	Description
Public health, safety, and welfare	Our product is made with safety in mind. we will design the light so that the diode will never get hot, giving peace of mind to the parents. The wireless feature also reduces the chances of wires strangling young children.
Global, cultural, and social	The goal of this product is to be universal and simple, using simple design choices that allow it to effortlessly fit into the average home.
Environmental	Our design will impact the power consumption of a household, by reducing overall power consumption due to the wireless charging causes it only sometimes to be plugged in and taking charge.
Economic	Our product is closed, meaning the final result will be used by our client and our client alone without moving to a secondary market. Because of this, the only economic impact we impose is paying for parts before assembly.

6 Conclusions

6.1 REVIEW PROGRESS

Over two semesters, our dedication to schedule, preparation, and communication has helped us to develop a well-rounded prototype that functions within all given constraints of the project. For semester 1, our focus was dedicated to building and testing our circuit as well as isolating any potential issues in hardware functionality. After developing and testing all variables and ensuring our project functions with all hardware-based constraints, we were able to transition our focus to the other side of project development: rapid prototyping with a design focus. Using previous experience with CAD modeling and additive manufacturing, we were able to design, review, test, and present a functioning project housing that holds and displays our project with a streamlined and effective design.

Our team was able to learn and develop skills that we would have needed to learn, test, and apply in the future including circuit design to given constraints, rapid prototyping, and product design/research.



6.2 APPLICATION VALUE

Our design is a unique product in a bland product space. In our research, we found that nursery lights have been bland and stale for years never really changing or advancing in any way. There are only a few different types currently readily available a wall plug light that turns on when it is dark and off when it is not dark, a table lamp that can be turned on or off, or a table lamp with two brightness settings. And they all have various designs and looks but all do the same thing. Our product changes the game adding new functionality that solves problems with current nursery light designs. Our design adds a battery so it can be picked up and moved around the room. With its convenient handle and basket design it is easy to carry and hold anything you might also need to carry in the middle. With the convenient wireless charging receiver imbedded in the base of the device, it can be charged anywhere throughout the house using just a Qi power wireless charging pad. Our design improves more than the life of the parent but also the children. We have incorporated a smooth transition system that can seamlessly switch between the 5 brightness settings our product supports, this allows you to set the light to an optimal brightness for your room. Also, we have added an automatic turn-off after 10 minutes that uses the smooth dim feature to turn off slowly without waking up the child.

6.3 POTENTIAL FUTURE STEPS

Throughout our design process, there have been some suggested ideas for future improvements or features that could be added. A big improvement that could be made which is available in some other similar products in the market is having internet and app connectivity. This would allow the user to change the settings remotely from a mobile device or be connected to a smart home network or baby monitors. Another idea that was proposed by our professor was to add voice controls. This would allow the user to control the product with their voice from around the room.

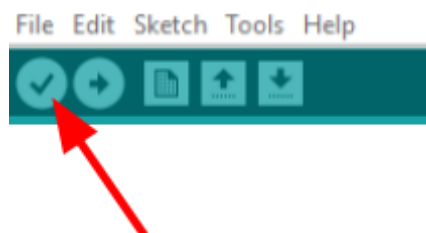
7 Appendices

7.1 OPERATION MANUAL

7.1.1 ADJUSTING THE ARDUINO CODE

To change the code for the Arduino follow the steps below. In general, the only aspects made to be changed are the brightness levels, the auto-shutdown time, and the turn-on/off time. These can all be adjusted by following the steps below, but not all of them need to be adjusted at the same time. In the case of only adjusting one setting, the other steps can be skipped. For example, if only the turn on/off time was to be changed, then do steps 1, 4, and 5. Skip 2 and 3 as they are not relevant to that portion.

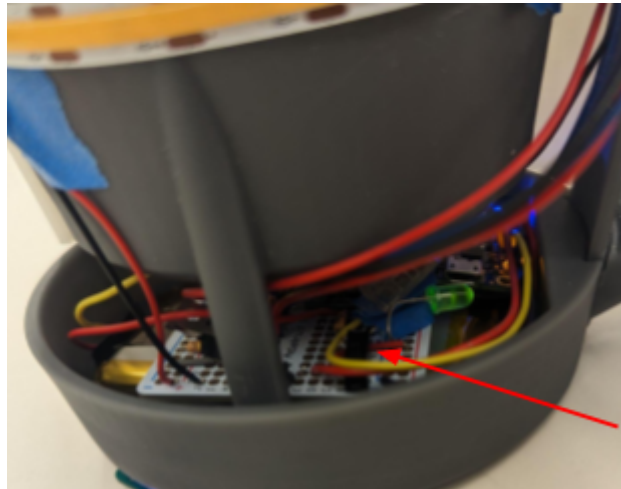
1. Open Arduino IDE and import the code found in section 6.4 of the Report.
2. To change any brightness levels, locate the array titled “brightnessLevel” and change a value within the curly braces (the new value should be between 0 and 255).. *Note: Changing brightness levels can affect the operational battery life
 - a. To add new brightness levels, simply change the 5 in the brightnessLevel square brackets to the number of total brightness levels and add numbers within the curly braces until the number of separate values equals that of the number in the square brackets.
 - b. Next, move down to the equation for the level variable and change the value after the “%” to be the number placed within the square brackets.
3. To change the duration the light is on, find this statement “if (millis() - startTime >= (1 * 60000)),” and change the value 60000 to the length until the automated turn-off. *Note: The number provided is in milliseconds so 60000 is 10 minutes. Changing this value will also change the total battery life.
4. To change the time it takes for the light to turn on or off, find the for loops
 - a. Choose “for (int i = 0; i < brightnessLevel[level]; i++),” for changing the turn-on time.
 - i. Within that loop, change the value within the delay block. The length of the turn-on time will be that value times the number of brightness levels. *Note: The value in the delay block is in milliseconds.
 - b. Choose “for (int j = brightnessLevel[level]; j > 0; j--),” for changing the turn off time
 - i. Within that loop, change the value within the delay block. The length of the turn-on time will be that value times the number of brightness levels. *Note: The value in the delay block is in milliseconds.
5. Once the changes are made, save the file and check the code in the Arduino IDE using the button shown in the figure below.



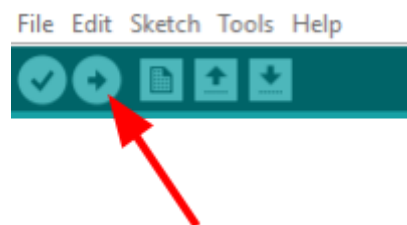
7.1.2 UPLOADING THE ARDUINO CODE

This is for uploading any code to the Arduino due to it being changed in the previous set of instructions. Before doing this, make sure to have a micro USB cable as it is needed to upload any code to the Arduino. Also, have an Allen wrench to unscrew mounts.

1. While the light is off, gently take off the opaque coverings. This can be done by lightly pulling on them, as they should be disconnected from the magnets holding them in.
2. Once the light is off, find the switch in the figure below and flip it.



3. Next, turn over the light, and using an Allen wrench, unscrew the two central bolts holding in the top mount (The others are for the handle and should not need to be unscrewed)
4. Once unscrewed, gently lift the top mount off of the handle. Make sure not to pull on it too hard, as the wires connecting the lights will still be attached to the base and can be broken if pulled too hard.
5. Plug in the micro USB to the Arduino port and open the Arduino IDE on a computer.
6. Pull up the code used for the light and press the button in the figure below to upload the new code.



7. When the upload is finished, reattach the top mount and flip the switch back to power the Arduino and the rest of the circuit.
8. Finally, put the covers back and ensure they are held in place by the magnet mounts.

7.1.3 USING THE LIGHT

Follow the steps below to use the main functionality of the light.

1. To use the light, place a finger over the power symbol on the top of the handle in the image below.



2. Wait for the slow turn-on of the light. The length of this depends on the time delay in the code and the number of brightness levels. This should be a couple of seconds.
3. Once it is at max brightness, then the light level can be adjusted by placing a finger over the brightness button shown below.



4. The light will then turn off after the set amount of time, by default, this length is 10 minutes.

*Note: If the light is not turning on, make sure to flip the switch below into the correct position, and if that does not work try charging the battery.

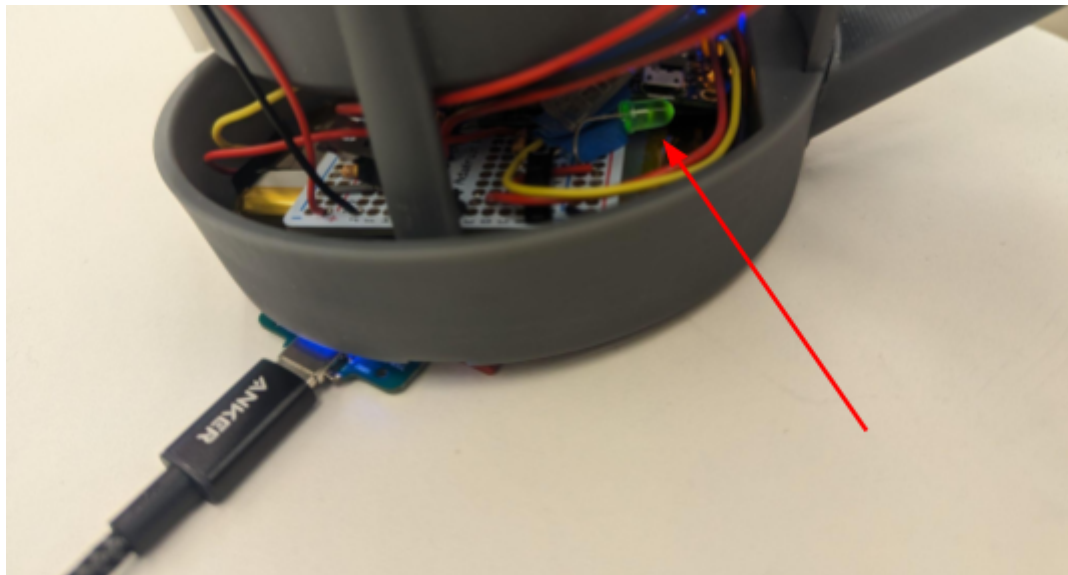
*Note: If the light is being moved around, objects can be carried within the top mount for convenience.



7.1.4 CHARGING THE LIGHT

Follow the steps below to charge the light if it is needed.

1. Place the light onto a Qi wireless charging transmitter.
2. Take off the opaque light covers and ensure that the green LED shown below is on.



3. Wait until the light is charged. This will take about 8 hours and 40 minutes from a dead battery

7.1.5 CHANGING ANY HARDWARE

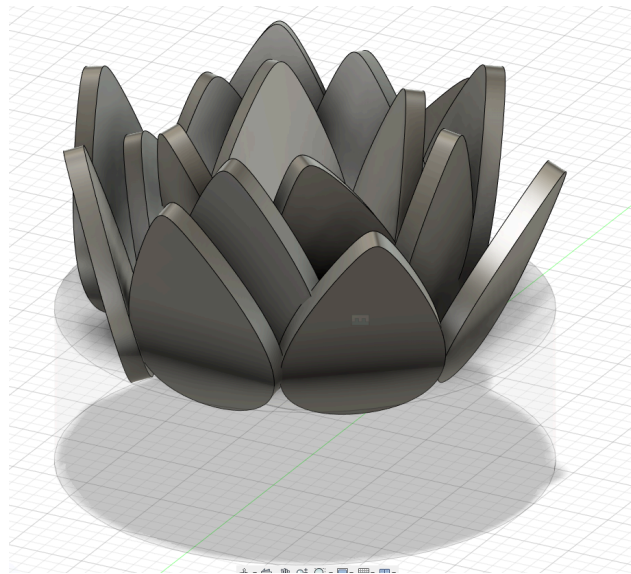
None of the hardware on the light should need to be changed, but if the necessity arises, the following steps can help fix it.

1. While the light is off, gently take off the opaque coverings. This can be done by lightly pulling on them as they should be disconnected from the magnets holding them in.
2. Once the light is off, find the switch in the figure below and flip it.

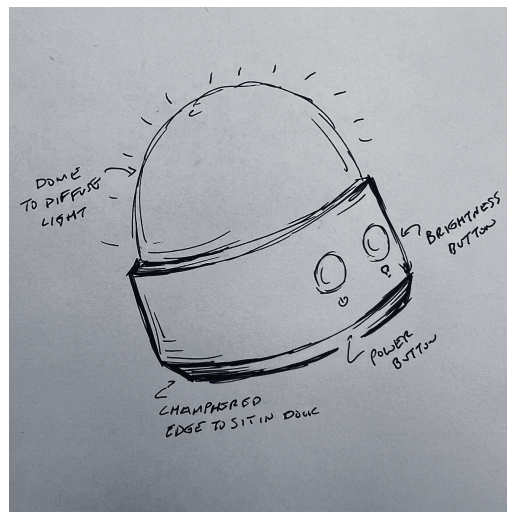


3. Next, turn over the light, and using an Allen wrench, unscrew the two central bolts holding in the top mount.
4. Once unscrewed, gently lift the top mount off of the handle. Make sure to not pull on it too hard as the wires connecting the lights will still be attached to the base and can be broken if pulled too hard.
5. Using the Allen wrench again unscrew the bolts holding in the prototyping breadboard.
6. Take out and replace any components that need to be using a soldering iron.
7. Reattach the breadboard and screw the bolts back in.
8. Reattach the top mount to the handle and screw in the bolts on the bottom of the light.
9. Flip the switch back to the starting position
10. Finally, put the covers back and make sure that they are held in place by the magnet mounts.

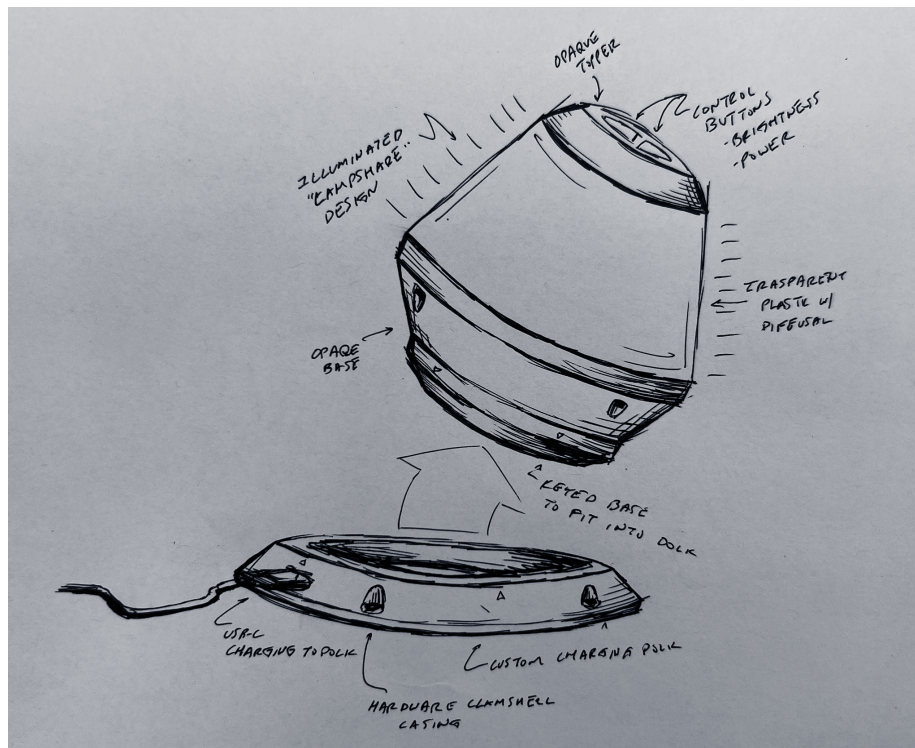
7.2 ALTERNATIVE/INITIAL VERSIONS OF DESIGN



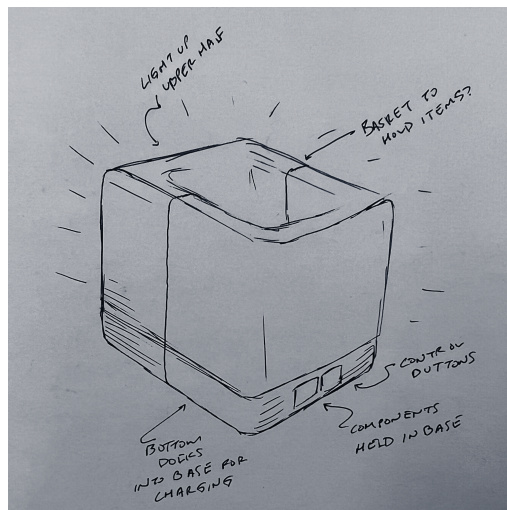
This is the first design we created, it was designed to look like a lotus flower and the petals would open up to turn on the light. This idea was scrapped due to the complex design and the need for more safety that would come with petals. However, as a brainstorming idea, this one helped us turn the gears to try and design something new.



Our second design addressed the issues of both complexity and safety with more rounded edges; and with a box design it was easier to handle and move around with. However this design has an issue of light diffusion as most of the light will project on the ceiling.



Another initial design, this one took inspiration from lampshades to develop a familiar yet different portable light device. Due to the lack of easy places to put buttons for device control, this design was scrapped however by inverting the idea we were able to progress into our “mug” design.



Another initial design, this came after discussions with our client delved into the desire for the ability to carry items along with the lamp. This created the above rudimentary “basket” design, to which the question arose of how it is carried if heavy things could be transported. The addition of a handle also brought us the

ability to change button placement to somewhere easier to access both while stationary and while being carried, which led to our current design choices.

7.3 OTHER CONSIDERATIONS

Throughout this project, we have learned how to apply our course learnings into a group project that mimicked our future working environments in a testing capacity. Firstly, we learned how to work in a group on a long-term project and goal. Most projects throughout our college career have been a semester-long at most so having one last a full year has brought new challenges is making sure everything stays on schedule to be productive the whole year. Another thing we have learned is how to work on a project for a client instead of just some written requirements. This brings in a new aspect of a project where it is not just important to meet the written requirements but also to please a client who the product is for and has their own opinion which is the most important. Finally, we have learned more about sourcing and putting together components on our own. Generally, for college projects the components and information is all given so having to research and find our own products is more realistic to real-world projects for our future careers.

7.4 CODE

```
#define ONBUTTON 4
#define BRIGHTBUTTON 7
#define LED 6
#define LIGHT 13

int level = 0;
bool light = false;

void setup() {
  pinMode(LIGHT, OUTPUT);
  pinMode(LED, OUTPUT);
  pinMode(ONBUTTON, INPUT);
  pinMode(BRIGHTBUTTON, INPUT);

  Serial.begin(9600);
}

void loop() {
  int onstate = digitalRead(ONBUTTON);
  int brightstate;
  int brightnessLevel[5] = {250, 190, 127, 65, 0};

  long startTime;

  if (onstate == 1 && light == false) {
    for (int i = 0; i < brightnessLevel[level]; i++) {
      analogWrite(LED, i);
      delay(60);
    }
  }
}
```

```

}

light = true;
long startTime = millis();
Serial.write("On button \n");

while (light == true) {
  brightstate = digitalRead(BRIGHTBUTTON);
  analogWrite(LED, brightnessLevel[level]);

  //Changes brightness state when button is pressed
  if (brightstate == 1) {
    Serial.write("Change light val \n");
    level = (level + 1) % 5;
    delay(200);
  }

  // Time check to turn off
  if (millis() - startTime >= (1 * 600000)) {
    for (int j = brightnessLevel[level]; j > 0; j--) {
      analogWrite(LED, j);
      delay(60);
    }
    light = false;
  }

  //Delay
  delay(250);
}
else {
  analogWrite(LED, 0);
}

delay(250);
}

```

7.5 REFERENCES

- [1] Jauch Battery Solutions,
https://www.jauch.com/downloadfile/5bf529a13146c840dc7ef8d1b712be114/4800mah_-_lp735977jh_1s1p_2_wire_70mm.pdf (accessed Nov. 30, 2023).
- [2] TNo106 n-channel enhancement-mode vertical DMOS FET data sheet,
<https://ww1.microchip.com/downloads/en/DeviceDoc/TNo106-N-Channel-Enhancement-Mode-Vertical-DMOS-FET-Data-Sheet-20005932A.pdf> (accessed Nov. 30, 2023).
- [3] 5V/2.4a USB type C - mm.digikey.com,
https://mm.digikey.com/Volume0/opasdata/d220001/medias/docus/338/106990290_Web.pdf (accessed Nov. 30, 2023).
- [4] TTP223-BA6 TTP223N-BA6 tontouch - LCSC,
https://datasheet.lcsc.com/szlcsc/TTP223-BA6_C80757.pdf (accessed Nov. 30, 2023).
- [5] Arduino® Nano every, <https://docs.arduino.cc/resources/datasheets/ABX00028-datasheet.pdf> (accessed Nov. 30, 2023).