LED Installation and Programming in Glow: Symphony of Life

Sven van Schie

Content

Introduction

Symphony of Life is an interactive light installation created for the Glow 2024 light festival in Eindhoven. The installation features five interactive, two-meter pillars that respond to touch, creating a dynamic symphony of light and sound. The primary focus of Symphony of Life is on the interaction between the visitor and the artwork, allowing the audience to engage with the installation in a unique, immersive way.

Each pillar consists of a metal base that houses the power supply and speakers, with a metal rod running vertically from the bottom to the top, where LED strips are mounted. At the top of the acrylic pillar, a metal cap holds a ring of sensors, enabling the interactive element of the installation.

This document outlines the research conducted, designs made, and steps taken to develop the functional lighting bars inside the acrylic tubing, which form the backbone of the installation. The challenges encountered, such as power management, memory optimization, and hardware validation, will also be discussed.

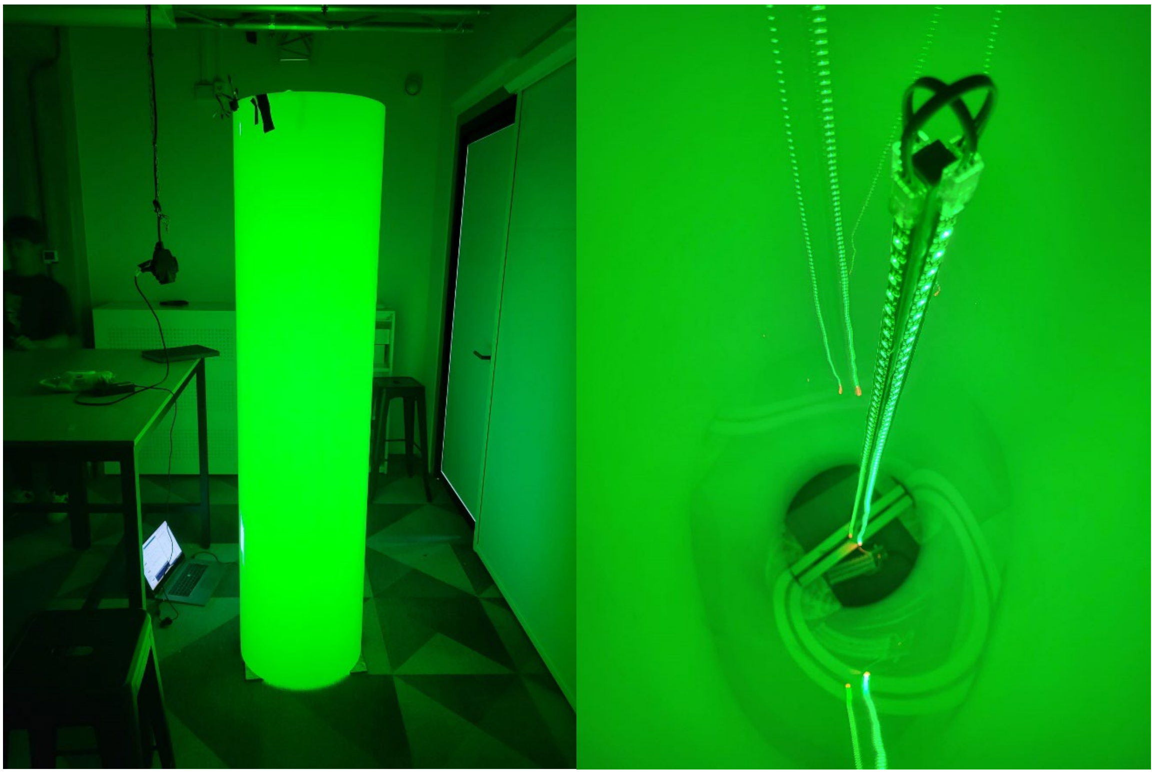
Prototype

Before starting on creating a 2-meter version, research was conducted on which type of light would be used for the 2-meter pillars. After this research was completed, LED strips were chosen due to their ability to provide more uniform illumination throughout the tube.

After a lighting solution had been selected, a two-meter prototype could be created. This prototype was made with three primary objectives:

1. Provide a functional base for testing and developing LED animations specific to the two-meter format.
2. Validate the hardware components and their performance in a full-scale prototype.
3. Offer a visual representation of the final product to guide further development.

After setting these objectives, a prototype could be assembled, test animations could be made, and hardware components could be tested in a two-meter environment. With the completion of this first prototype, a few things were discovered that required an update.

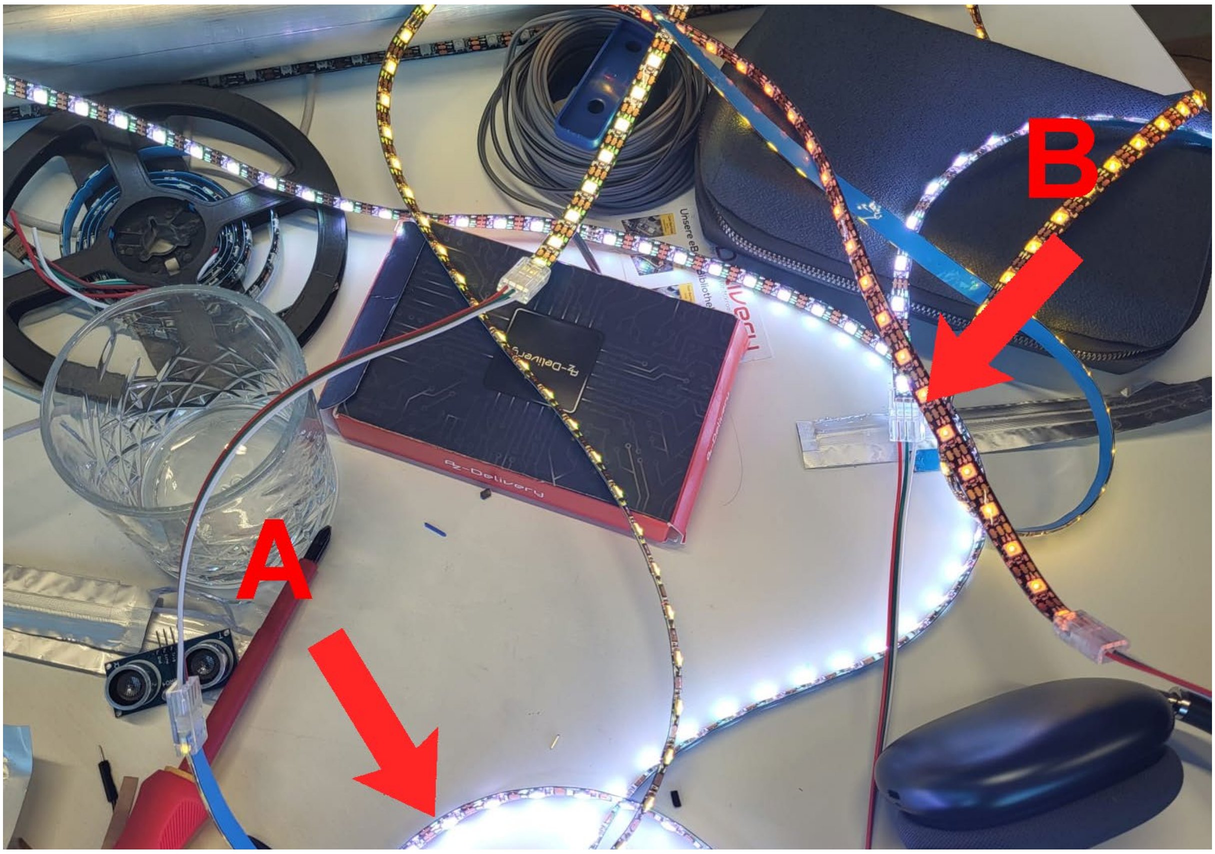


Note. Photograph of the inside and outside of pillar with green light.

Power

It was quickly discovered that the power fall-off on the LED strips was greater than accounted for. The LED strips would start bright yellow and gradually become more orange towards the end of the last strip. This was deemed unacceptable, as colour would be an important part of the visual experience.

The quickest and most straightforward solution was to equip both the beginning and the end of the LED strip with power supply connections. This meant inserting the data signal into one side while having the voltage and ground come from both sides. This solution proved to be sufficient, with only a minimal brightness fall-off throughout the LED strips.



Note. Brightness-fall off between start of LED strip (A) and end of LED strip (B).

Not enough light & pillar size

After getting a clear visual on the amount of light produced by four LED strips, it was decided that they would provide insufficient light for an effective visual experience. Because of this, the number of LED strips had to be increased from four to eight. This also meant changing the bar upon which the strips were mounted.

Because of the amount of space between the LED bar and the acrylic pillar surrounding the bar, it was discovered that sufficient light diffusion was present for the LED strips to not have to be placed exactly next to each other. To make assembly easier, the decision was made to use a 50mm circular rod. This provided enough space for the LED strips, with added space in between for easy assembly, and good light distribution because of its shape.

Timing

To get a better understanding of the amount of time needed to update the LED strips, flags were used to test the strips' on and off states. It was important to measure this updating time to ensure that no further optimizations were possible and to verify that the current lighting code didn't block any other parts of the program.

After measuring the amount of time needed to update the LEDs, a discovery was made. It appeared that the function for updating the LEDs blocked all other code necessary, including the code checking the sensor values. To resolve this issue, a function was wrapped around the code to add a short delay to any updates sent to the LED strips.

Memory

One concern with the number of LEDs now present inside the pillar was the amount of memory needed to save all the RGB values inside an array.

An unfortunate realization with the library used (FastLED) was that the library uses the RGB array for two separate use cases. First, it uses the CRGB array to initialize each LED. This means that when you have an LED strip of 100 LEDs, you would need a CRGB array with 100 indexes. The second use of this array is to send the CRGB values to the individual LEDs based on their index within the array. A problem with this method is that it makes it programmatically impossible to initialize a large number of LEDs by copying and pasting CRGB values. It would have been preferable to split these arrays into a singular array for the LED index and a second array for any CRGB values needed for animations, saving space on CRGB values.

Because of this, research was done into trying to reduce the size of the array or the memory placement programmatically, without changing the library. This problem of the CRGB array taking up a large amount of memory was found on an internet forum, where a user suggested: "Look into PROGMEM for array storage and F() for serial printing constant literals." (Blackfin, 2019). Unfortunately, it was deemed unfit for our solution because our CRGB could not be defined as a constant, needing continuous updating based on touch.

A second solution was discussed by moving the array from the smaller SRAM to Flash memory. This was useful during compilation, as it saved program space, but the microcontroller would still crash at runtime when the array was copied from Flash into SRAM during the program's execution.

After concluding that this problem could not be fixed programmatically without changing the library—a decision against which was made due to limited time—the decision was made to fix this problem by modifying the hardware. Instead of having 8 separate LED strips, each with their own index in the CRGB array, the data cable would be split in two, resulting in a copied animation on the two groups of four LED strips. Because of the circular design of the pillar, the relative simplicity of the animations, and the amount of diffusion, this would not be visible to an outside viewer.

Animations

The user experience team provided a colour palette and two lighting animations, ready for further processing. However, in their current state, the code running these animations was not suitable for integration with the sensor and server code.

Two primary issues needed to be addressed. First, the code quality did not meet the standards required for the project. There was no consideration for memory usage, and multiple instances of the delay() function were present, potentially blocking the rest of the program. Second, the switching between animations was abrupt, which was deemed undesirable for an outdoor viewing experience. Due to these issues, both a code refactor, and animation transition improvement were necessary.

To resolve the transition issue, the blend function from the FastLED library was utilized. This function takes two colours as input and computes a new colour based on a fractional blend between them. By incorporating this function, it became possible to adjust each frame of the animation to gradually move toward the colours of the next animation, creating a smooth and visually pleasing transition.

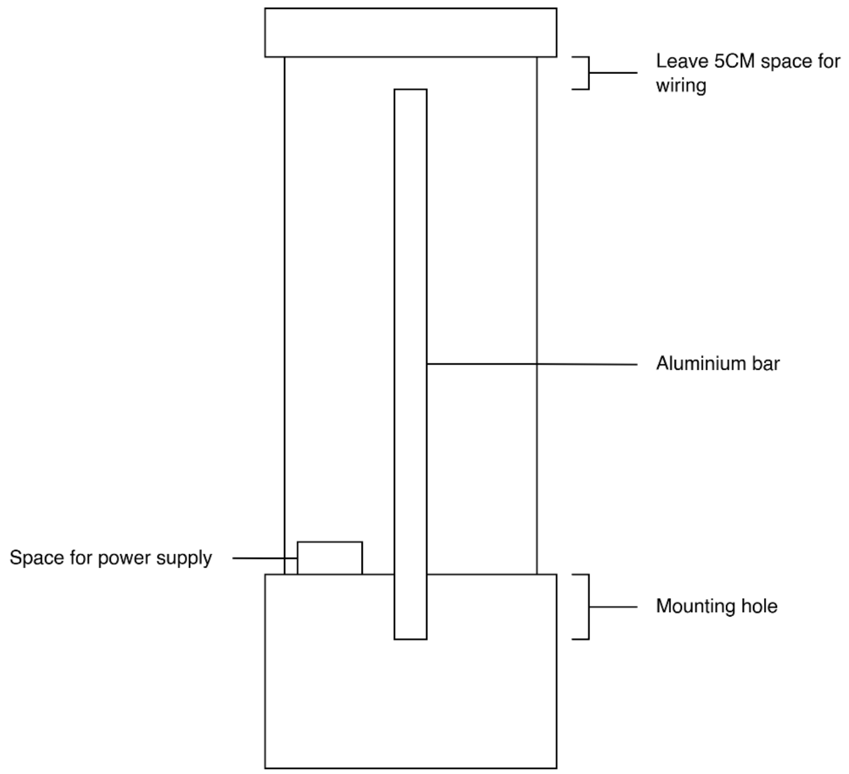
Requirements V1

After creating this prototype, the first set of requirements could be sent to the outside party responsible for creating the metal top and base.

The following requirements were sent:

Inside the 2-meter tube, an aluminium bar will be mounted, upon which the LED strip will be placed. The top and bottom have several extra requirements to fit this aluminium bar:

1. The bar should be firmly mounted to the bottom but **should always be removable**. A rectangular hole in the bottom with space for the bar might be a simple solution.
2. The bar should be removed from the top by **5 cm** to leave space for wiring.
3. There should be space in the bottom of the pillar for a power supply.
4. The aluminium bar has a width of **1.5 cm**.



Note. Image of requirements sent to outside party.

Requirements V2

After receiving the first version of the metal top and bottom, a second set of requirements were sent to the outside party. These requirements consisted of both changes to the existing structure as well as changes based on new developments to our prototype (the new circular lighting bar, for example). It also included general feedback on the structure.

The following requirements and changes were sent to the outside party, ordered by additions to the existing structure, necessary revisions, and potential worries with the design:

Additions

After reviewing the pillar, a few possible additions came to mind to improve functionality and aesthetics.

Side cable holes

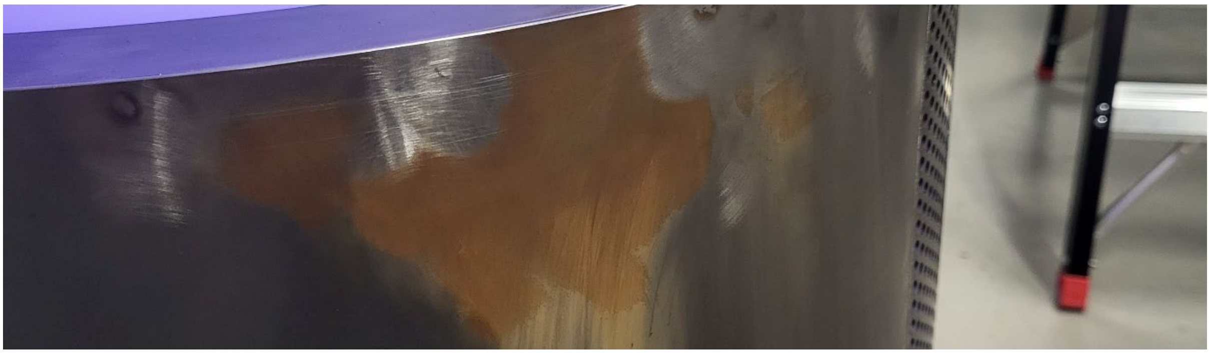
Multiple cables need to enter the base of the pillar, including power, DMX, and serial cables. In the current base design, a hole has been found on the bottom plate, but it has not been verified whether this will be our cable entry point or if an additional hole will be needed on the side of the base.

Pinboard mounting spot

A pinboard will be placed in the top part of the pillar. Creating a mounting spot with screw holes is an effective way to securely attach the pinboard to the inner plate of the top part, reducing unnecessary movement of the electronics.

Paint

The base and top may require paint to cover the "prototype spots," although this is still under review.

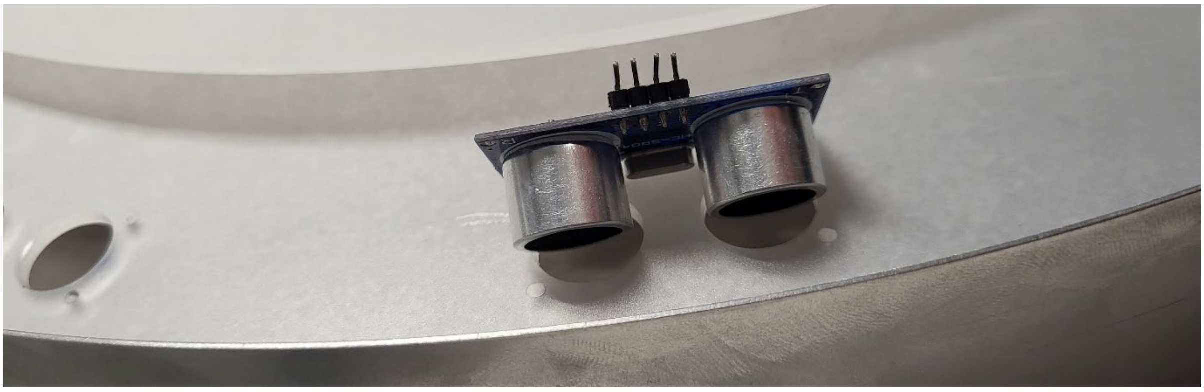


Necessary revisions

After close inspection and some changing requirements, it was concluded that several revisions will be necessary to the existing top and base design.

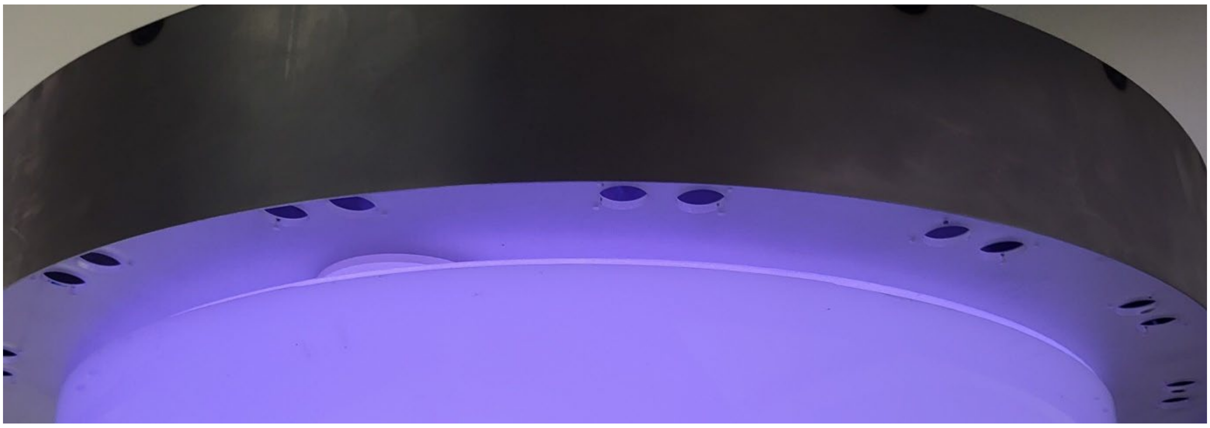
Sensor hole size

The current sensor holes are too small for the sensors to fit properly.



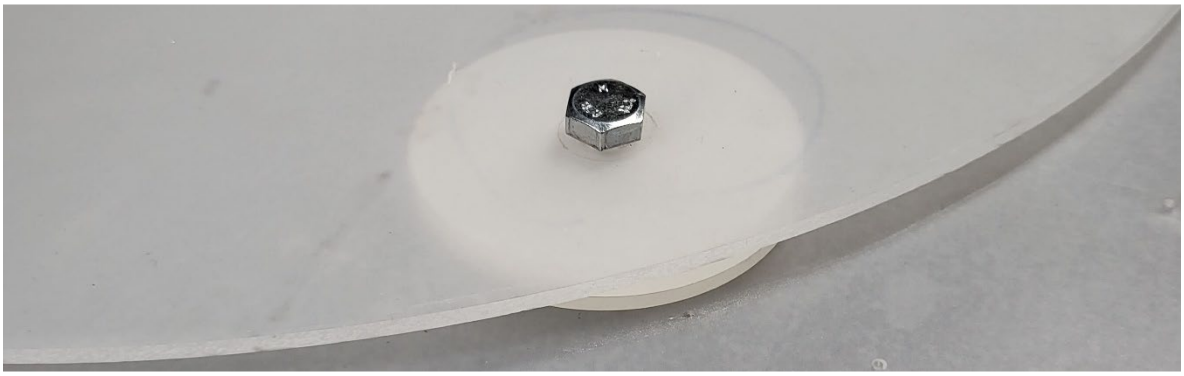
Top inner plastic circle size

The inner circle of the top part does not fit inside the pillar, though it comes very close.



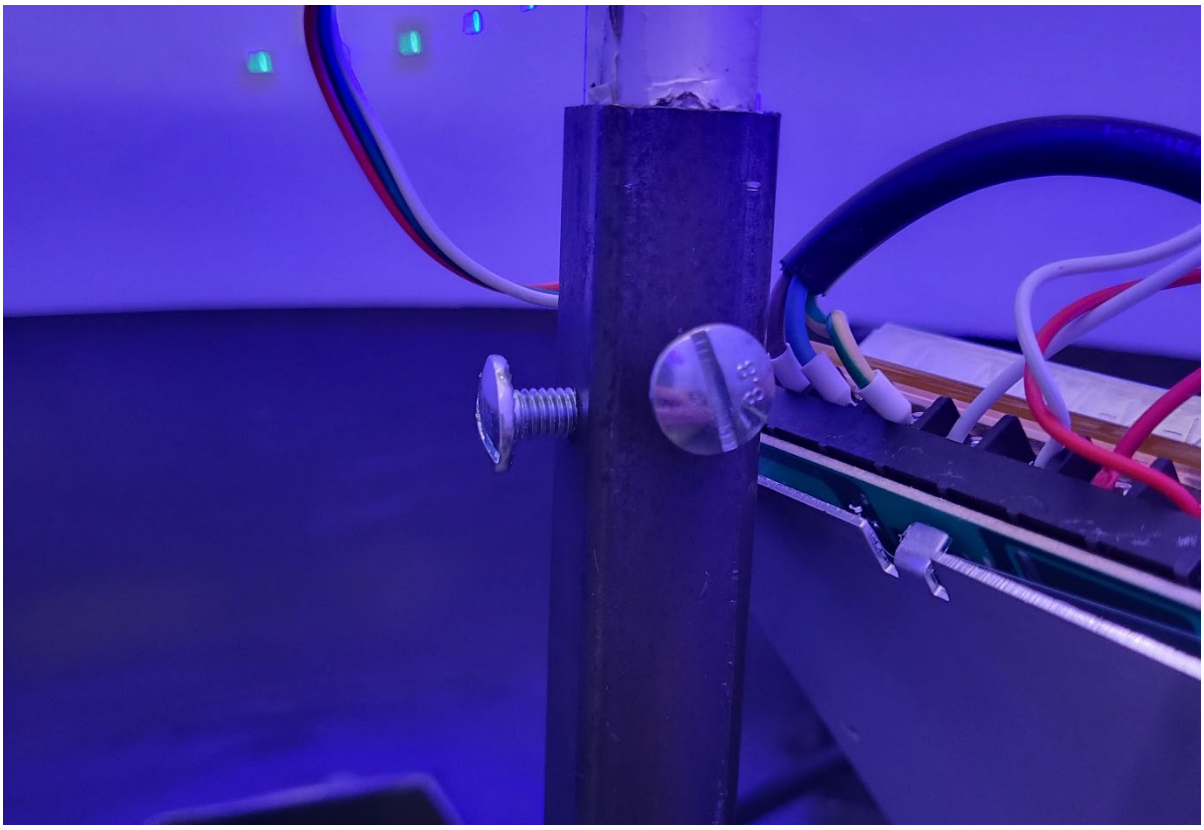
Removability top inner part

Removing the inner plastic part of the metal top is currently challenging, making quick fixes difficult. Simplifying the design, if possible, would improve accessibility for maintenance.



Circular light mounting

The current rectangular lighting mounting bar will be replaced with a larger circular one, necessitating a change to the hole in the bottom. Exact dimensions are still under consideration.



Worries and possible solutions

Several concerns were raised regarding the safety and aesthetics of the current design. Possible solutions have been proposed, but we seek professional advice on the feasibility of these solutions or any alternative recommendations.

Pillar falling out of base

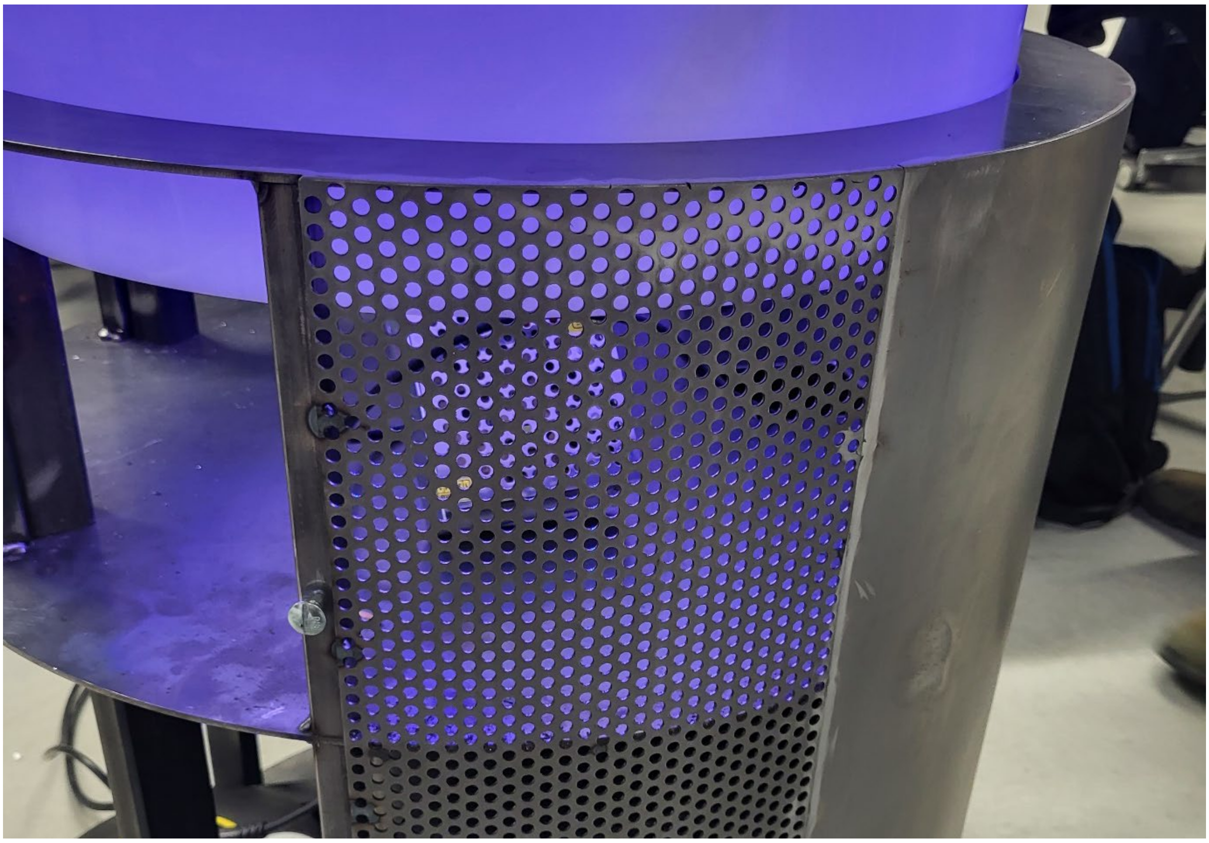
With the current base design, the pillar is too easily removable, raising concerns that it might fall out of the base if accidentally pushed.

A possible solution could be increasing the depth of the pillar inside of the base.

Water/light permeating

A concern was raised about light showing through the grating on the second part of the base. Additionally, water could run through this section, potentially damaging electronics such as the power supply.

Options include removing the grating only on the second layer of the base or closing it off from the inside.



Top part falling

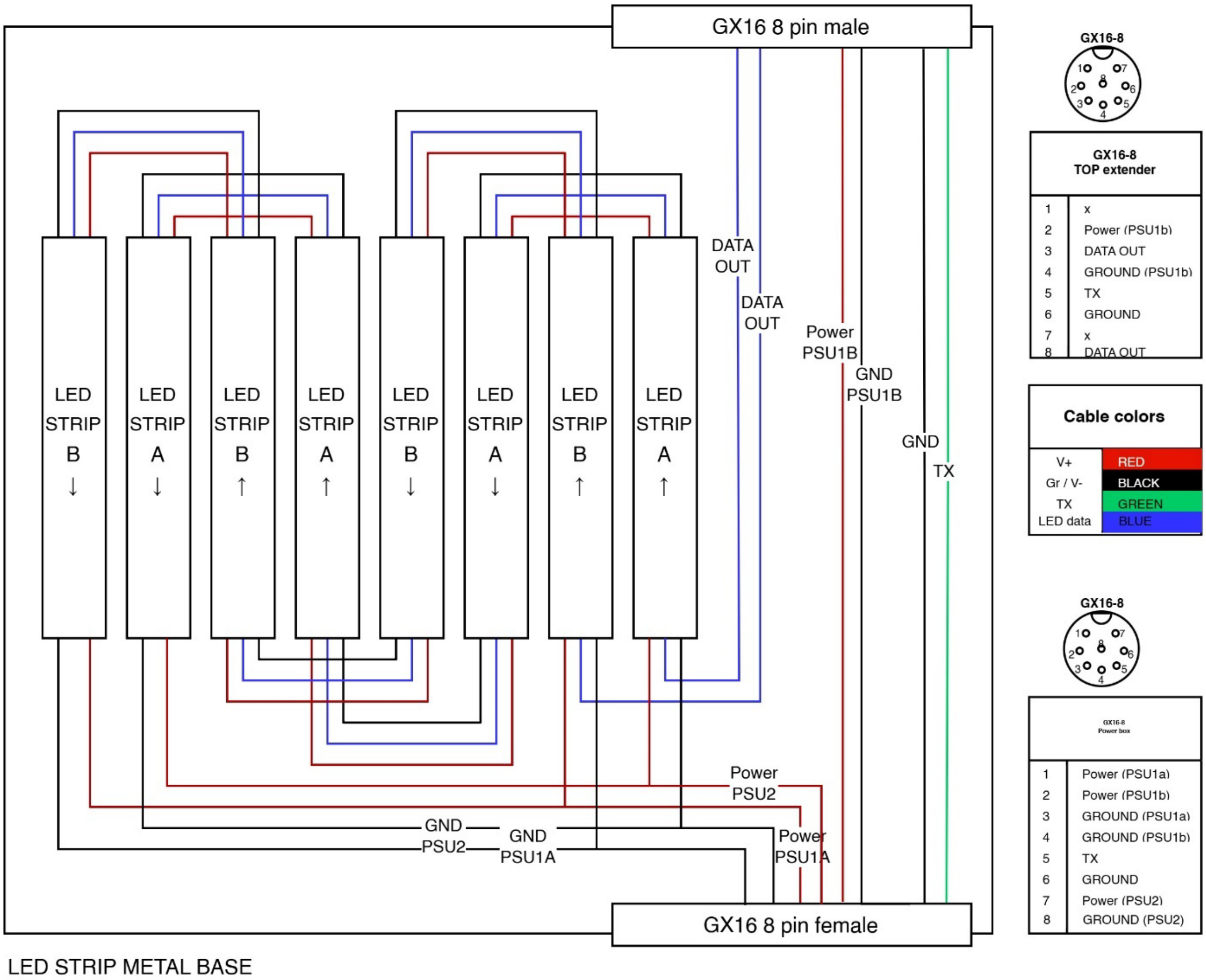
A final concern was raised about the top part easily falling off the pillar.

Increasing the depth or thickness of the top's inner section within the pillar could resolve this issue.

Design

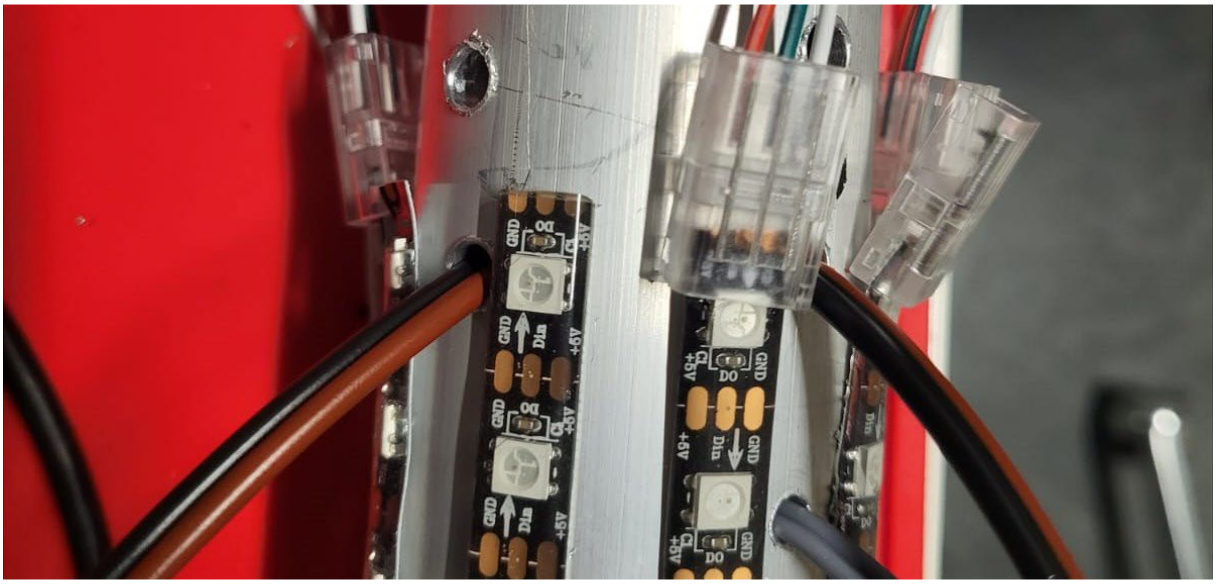
After a second set of requirements was sent to the external party, a design for the LED bar needed to be created. This design was essential to optimize the workflow and facilitate easier knowledge transfer, enabling team members from different groups to seamlessly step in when needed.

Given the arrangement of the LED strips, it was decided that both data and power would be supplied from the bottom. To simplify transportation and setup, GX16-8 connectors were chosen to be soldered between the top, bottom, and middle sections, providing easy plug-and-play functionality.



Note. Hardware overview of LED strips mounted on circular bar

To ensure the cables could easily reach the LED strips, it was decided to drill five holes into the bottom of the tube. This would allow the four power cables and two data cables to pass through the bottom and into the tube, reaching the LED strips. Once inside, the cables would be soldered to smaller wires connected to the GX16-8 connectors.



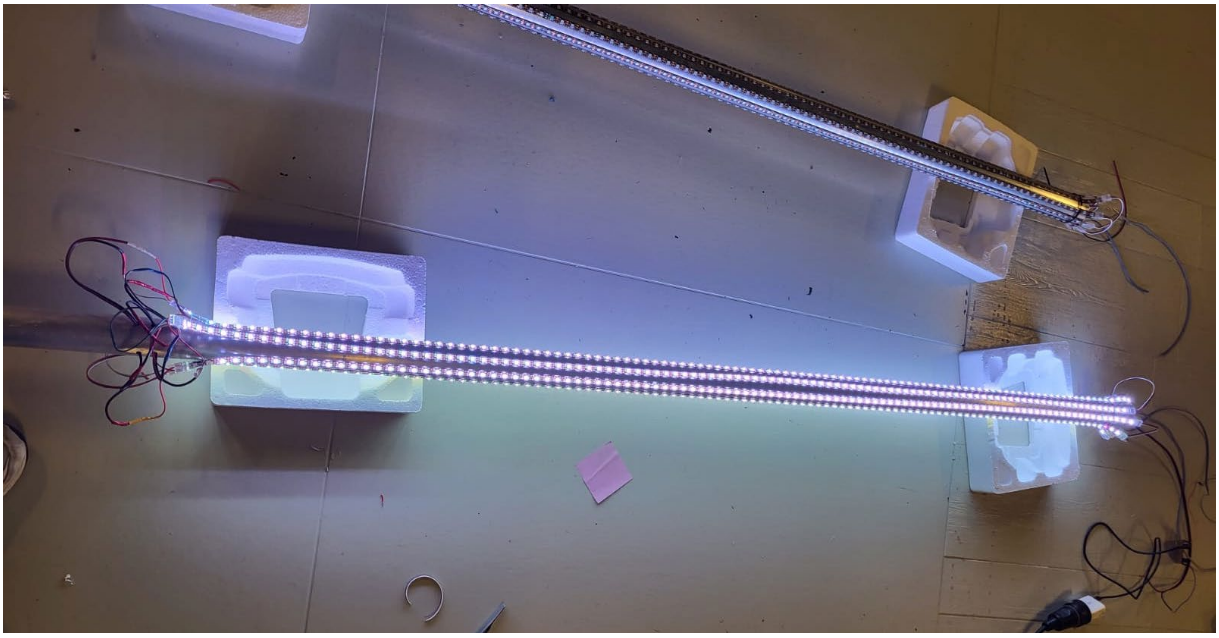
Note. Holes inside the tube, allowing wires to pass through reaching the LED strips.

Realisation

After completing the design and a functional prototype, the production of all LED bars could begin. The process involved the following steps:

1. Drilling holes into the tube.
2. Connecting eight LED strips together using connector clips.
3. Running the power and data cables through the tube and the drilled holes.
4. Adhering the LED strips to the tube.
5. Soldering the power and data cables to the LED strips.
6. Soldering the GX16-8 connectors to the wires at the top and bottom.

Once these steps were completed, a functional LED bar was ready to be mounted to the top and bottom of the pillar.



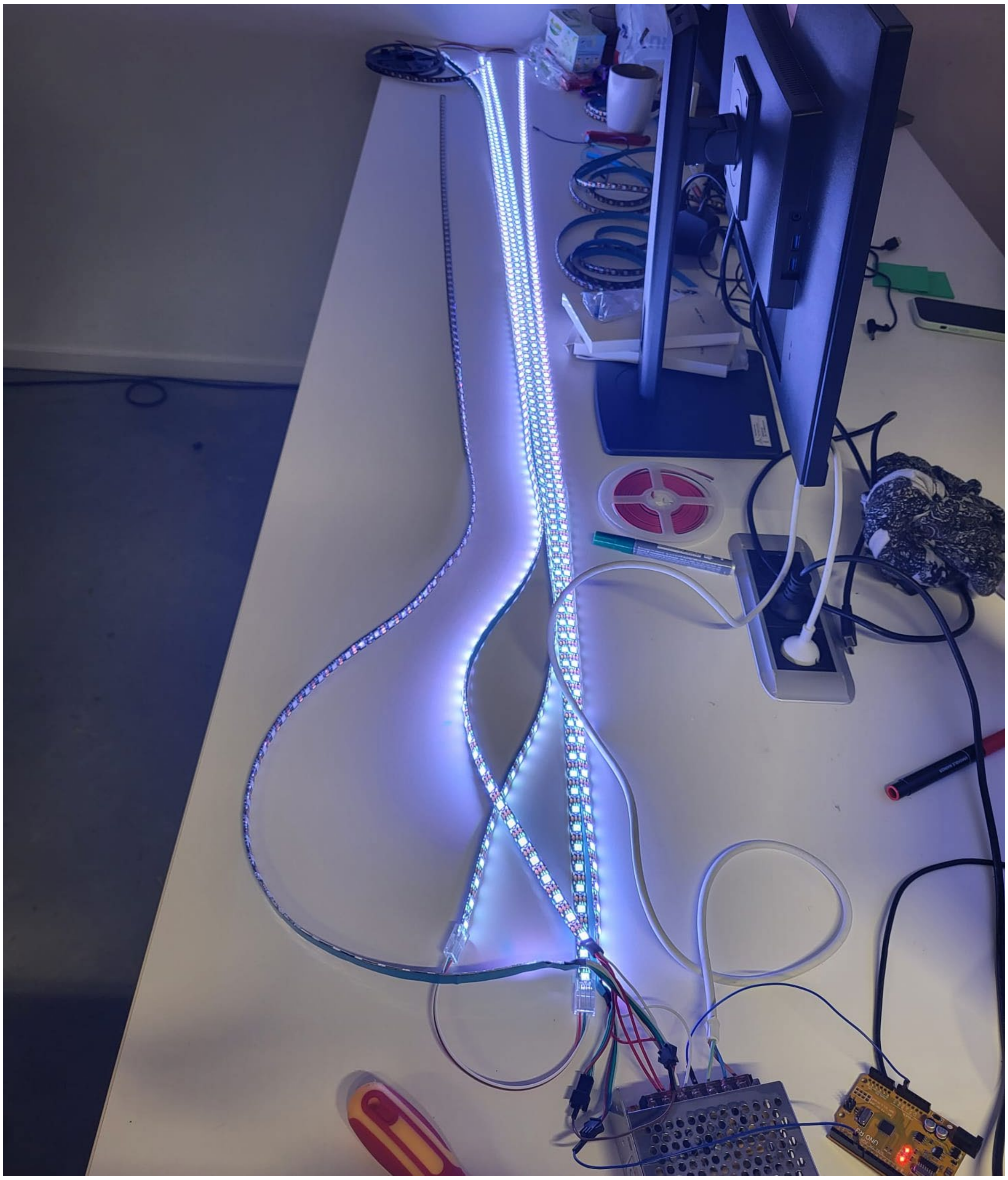
Note. LED bars in the assembly process, before being mounted onto the pillar structure.

Hardware validation

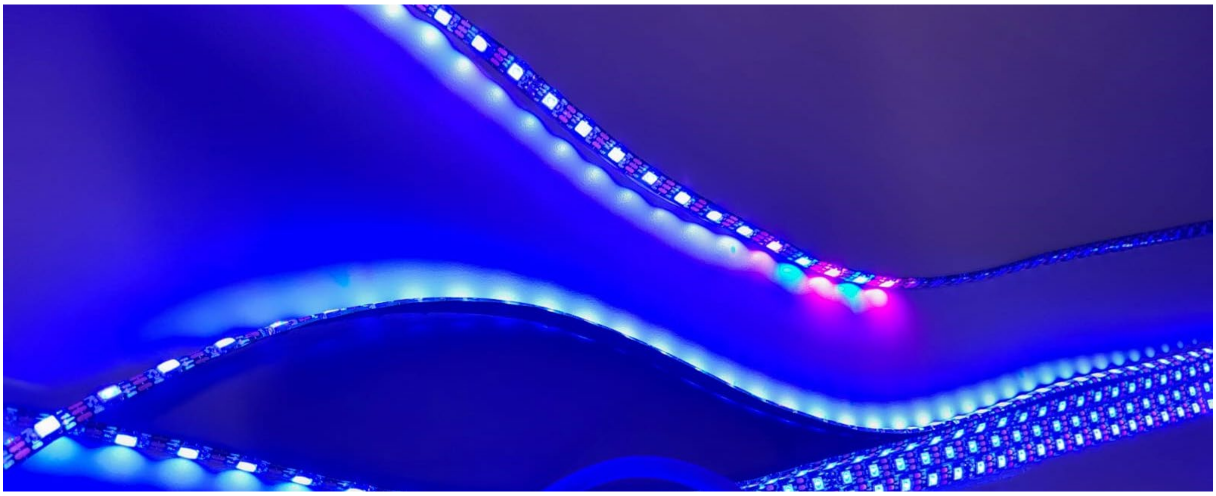
During realisation, it was crucial to validate the functionality of each hardware component, at each step of the assembly process.

The first step involved testing the LED strips. Although most strips were brand new, verifying their functionality was essential, especially for those previously used in earlier prototypes. Given the number of strips involved, defective sections were cut away, and the remaining portions were soldered together to form new, functional strips. These were tested again to ensure reliability.

This step was critical because removing and replacing individual LEDs after they had been adhered to the bar proved to be a challenging and time-consuming task.



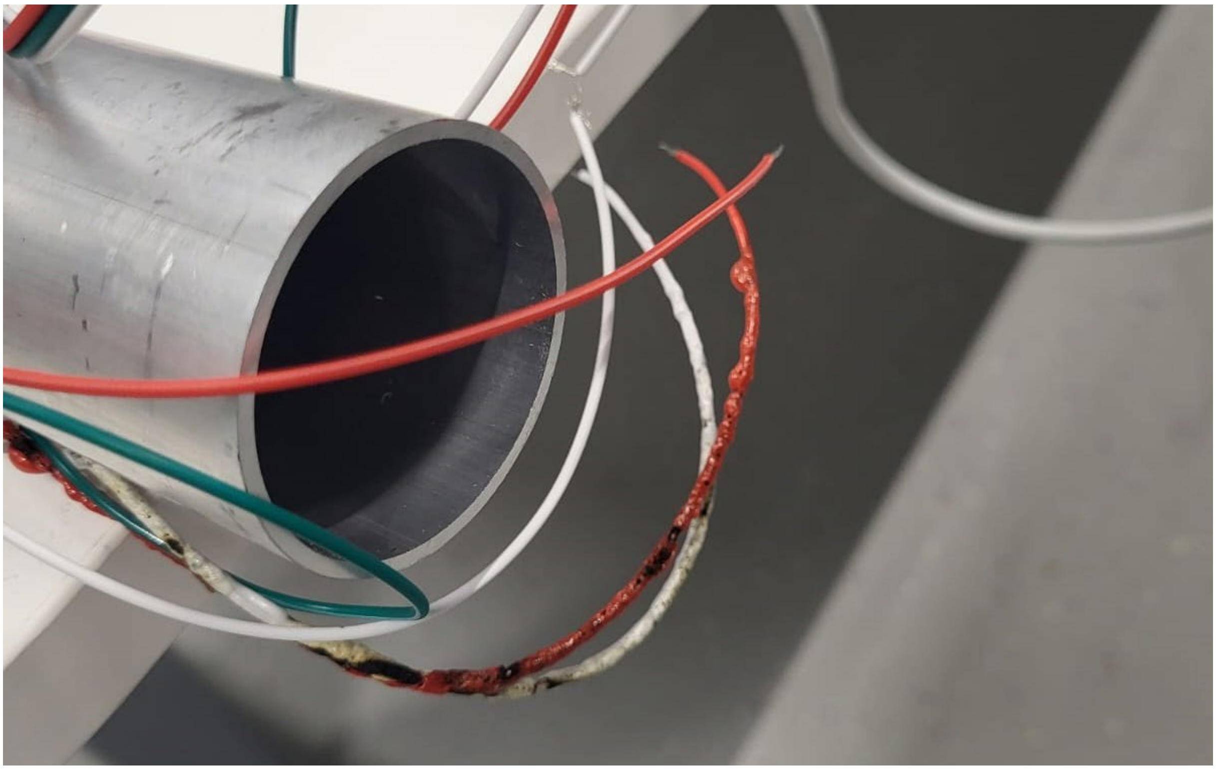
Note. *LED strips being validated before being mounted on the circular bar.*



Note. Broken LED strips that require removal.

After validating the eight LED strips required for a single lighting bar, they were assembled onto the bar itself. The next step in the assembly process involved soldering the necessary wires to the GX16-8 connectors. Once this process was completed for one side, the connections were verified using a handheld digital multimeter.

This validation step was crucial. For example, if the ground and voltage wires were accidentally switched, the wiring could overheat, emit smoke, or even catch fire—a problem encountered during the creation of an earlier prototype.



Note. Wires burned through because of improper wiring

After validating the wiring with a multimeter, the setup was connected to a power supply at the bottom and a single Arduino at the top. If it functioned as expected, the final GX16-8 connector at the top could be soldered onto the bar. Additional multimeter testing was then performed to ensure proper functionality.

Once these steps were completed, the lighting bar was fully prepared for final assembly onto the actual pillar structure.

Note. LED bar mounted onto the pillar structure.

Conclusion

The development of the interactive light installation for Symphony of life, a project for the Glow 224 light festival, presented a rewarding and educational experience, offering insight into both the hardware and software challenges associated with large-scale interactive installations. From addressing power fall-off issues and memory optimization to refining the structural components of the pillars, the project has progressed significantly since the initial prototype. The integration of hardware design and hardware validation was crucial to ensuring the installation's reliability and user experience.

In terms of user interaction, further refinement could be done to the sensor system to improve responsiveness and accuracy. The current animations present inside the pillar were not the first chosen and could be updated to provide a more dynamic and interactive experience.

From a structural standpoint, the current design is functional, it is however currently too easy to break. The top of the pillar breaks easily under pressure while the connectors connecting the LED strips together don't provide a consistent connection, causing sometimes more than half of the LED strips to fail.

The server connection between the pillars could also be improved. There is currently very little interaction between pillars and playing with light and sound depending on pillars touch could enhance the projects interactivity even further.

In conclusion, while the interactive light installation for Symphony of Life has made significant progress, there are still areas for improvement. Enhancing the sensor system, updating animations for a more dynamic experience, and strengthening the structural design will help address current limitations. Overall, the project has been a valuable learning experience and a great stepping stone toward creating more innovative and immersive interactive installations in the future.

References

Blackfin. (2019, November). Look into PROGMEM for array storage and F() for serial printing constant literals. [comment on forum post about CRGB memory reduction] Arduino forum. https://forum.arduino.cc/t/suggestions-on-reducing-memory-requirements-using-crgb/621326