

# ROOMIFY: A SMART ROOM SYSTEM

ECE 445 DESIGN DOCUMENT - SPRING 2026

---

## **Project # 67 - *Roomify***

**Benjamin Chang, Warren Lam, Owen Wang**

Professor: Craig Schultz

TA: Lukas Dumasius

# Contents

<b>Contents</b>	<b>2</b>
<b>1. Introduction</b>	<b>3</b>
1.1 Problem	3
1.2 Solution	4
1.3 Visual Aid	5
1.4 High-Level Requirements List	5
<b>2. Design</b>	<b>6</b>
2.1 Block Diagram	6
2.2 Physical Design	7
2.3 Subsystem Overview and Requirements	9
2.3.1 Infrared Transceiver System	9
2.3.2 Repeater System	10
2.3.3 Roomify Display Interface System	11
2.3.4 Power Systems (Box and Repeater)	12
2.3.5 Box + Logic System	15
2.3.6 Mobile Device Control System	16
2.4 Tolerance Analysis	18
2.4.1 Infrared Signal Transmission	18
2.4.2 Box Power System	19
<b>3. Cost and Schedule</b>	<b>20</b>
3.1 Cost Analysis	20
3.2 Schedule	21
<b>4. Ethics</b>	<b>22</b>
4.1 Impact and Engineering Considerations	22
4.2 Lab Safety Manual	24
<b>References</b>	<b>26</b>

# 1. Introduction

## 1.1 Problem

Room devices such as LED lights, music systems, TVs, and other decorations are hard to coordinate, leading to unsynchronized room atmospheres and an excess of handheld remotes. People interested in room decor face a fragmented ecosystem where each device operates independently. This makes it difficult to create cohesive environments with a unified mood (e.g., slow music with flashing LED lights), and the need for multiple remotes and is inconvenient for managing room environments.

For college students and apartment residents, budget and room modification restrictions make purchasing and installing integrated smart home systems unrealistic. In addition, existing products like Google Home and Amazon Echo only support specially designed devices, making it impossible to integrate use with ordinary room decor devices and appliances. These smart home-compatible devices are also much more expensive and harder to find. The lack of solutions for smart room systems indicates a need for an affordable, flexible solution that can synchronize home devices without requiring special hardware.

As social media use rises, room decor has become increasingly popular as people post their setups. For example, the hashtag #HomeDecorTikTok has over 10 billion views, making the problem of improving room ecosystems extremely relevant [1]. Solving this problem with a cohesive, extendable room system would reduce remote clutter, enhance room lifestyle, and establish a new standard for integrated rooms. Because people spend a lot of their time at home, the ability to easily create personalized room environments without purchasing expensive smart devices would be very impactful.

## 1.2 Solution

We propose *Roomify*, a centralized room control system where users can easily operate devices in their room like LED lights, TVs/projectors, aroma diffusers, Spotify, or other remote-controlled decorations and displays. For example, users will be able to press a single button on the *Roomify* screen and put on the cooking channel, turn the LED string lights yellow, diffuse lavender, and play jazz music. *Roomify* introduces an extendable, cohesive room system that coordinates independent devices and removes the need for multiple remotes.

*Roomify* will appear like a vinyl player with a hinged wooden box, with a square capacitive touch RGB TTL TFT display driven by an ESP32-S3 board. The core functionality of the board will be controlling the display, infrared (IR) signal receiving and decoding to store device remote codes, omni-directional IR transmission to transmit device remote codes, and WI-FI communication (to make Spotify API calls). In addition, we will design “repeaters” placed throughout the room to relay IR signals to devices not in direct line of sight.

By copying and storing IR remote codes as a “virtual remote”, users can map buttons on the *Roomify* touchscreen display and their phone (web-app) to IR signals that interact with devices in the room. To accomplish this, *Roomify* will have an "Add Remote" mode. For example, a user would add the "red" button on a LED string light remote by aiming the remote at the *Roomify* box. After *Roomify* decodes the signal, users can label and save the button code data. The user could then repeat the process for other buttons on the remote, or for other remotes. Any device that uses infrared remote signaling or has an API (like Spotify) will be usable with *Roomify*.

After adding remotes to *Roomify*, users will be able to create quick-start presets (e.g. green lights with Christmas music and a Snoopy Gif display on the square display screen). When the user selects a preset, *Roomify* will transmit the necessary IR signals in all directions. Aside from presets, users can also use the virtual *Roomify* remotes to change individual device settings.

*Roomify* is an extensible, centralized room system that allows for full control over room decor, lights, and sound, allowing students and apartment owners to easily set cohesive room atmospheres.

## 1.3 Visual Aid

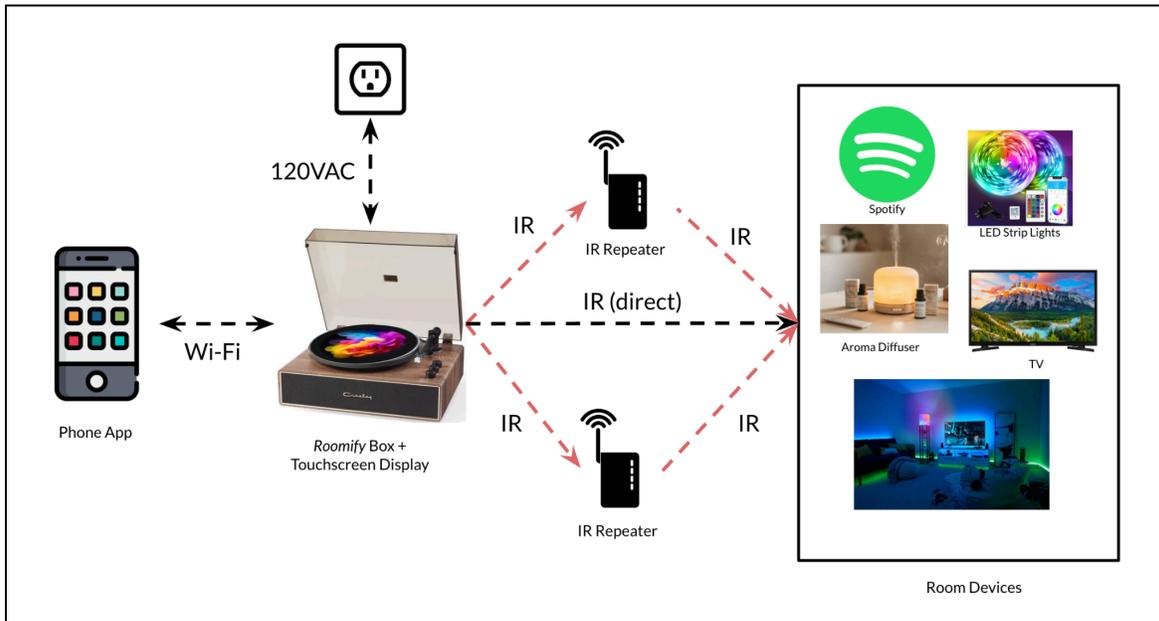


Figure 1: Visual Aid

## 1.4 High-Level Requirements List

1. **Infrared Remote Control Transmission.** *Roomify* should be able to decode, store, and retransmit at least 10 unique IR commands for at least three different devices with accuracy  $\geq 95\%$  compared to using the original remote.
2. **Preset Execution.** *Roomify* should be able to operate multiple devices in a user-selected preset. Execution should finish within 1 second, measured from user input to the correct activation of all devices in the preset. The preset should include at least one device in the following categories: IR transmission (LED lights), API calls over Wi-Fi (Spotify), and *Roomify* screen display.
3. **Range and Coverage.** *Roomify* must provide reliable omnidirectional infrared coverage capable of controlling three separate IR-based devices within a 5-meter radius across a  $\geq 300^\circ$  horizontal field using no more than three infrared repeaters.

## 2. Design

### 2.1 Block Diagram

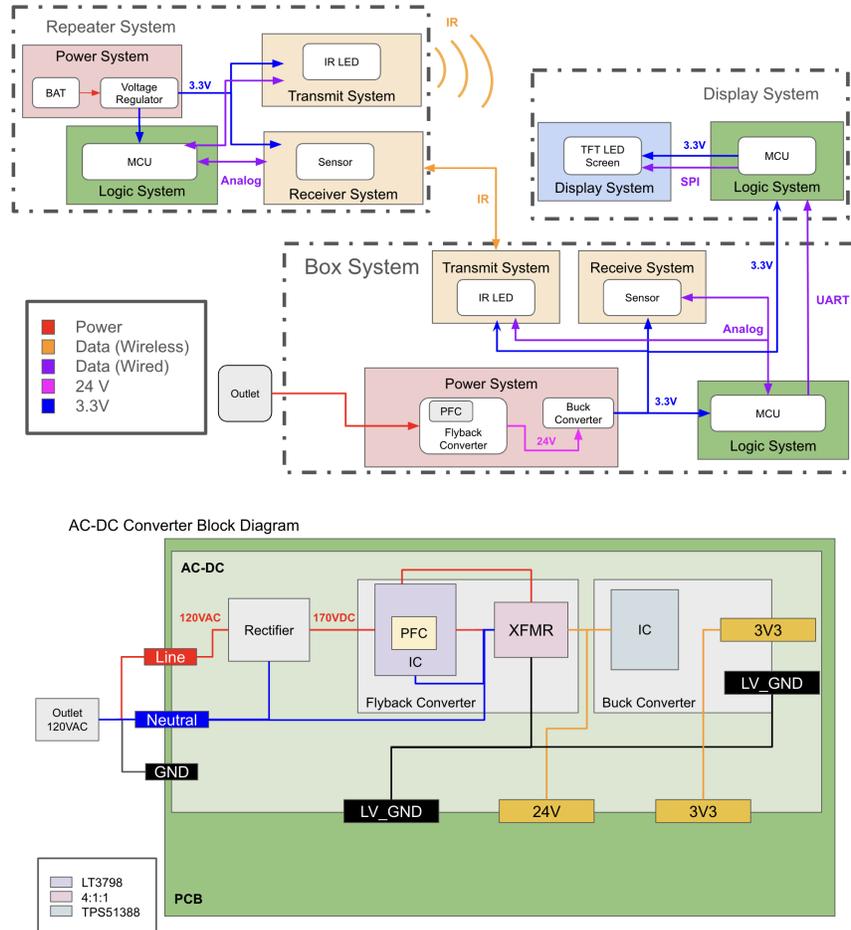
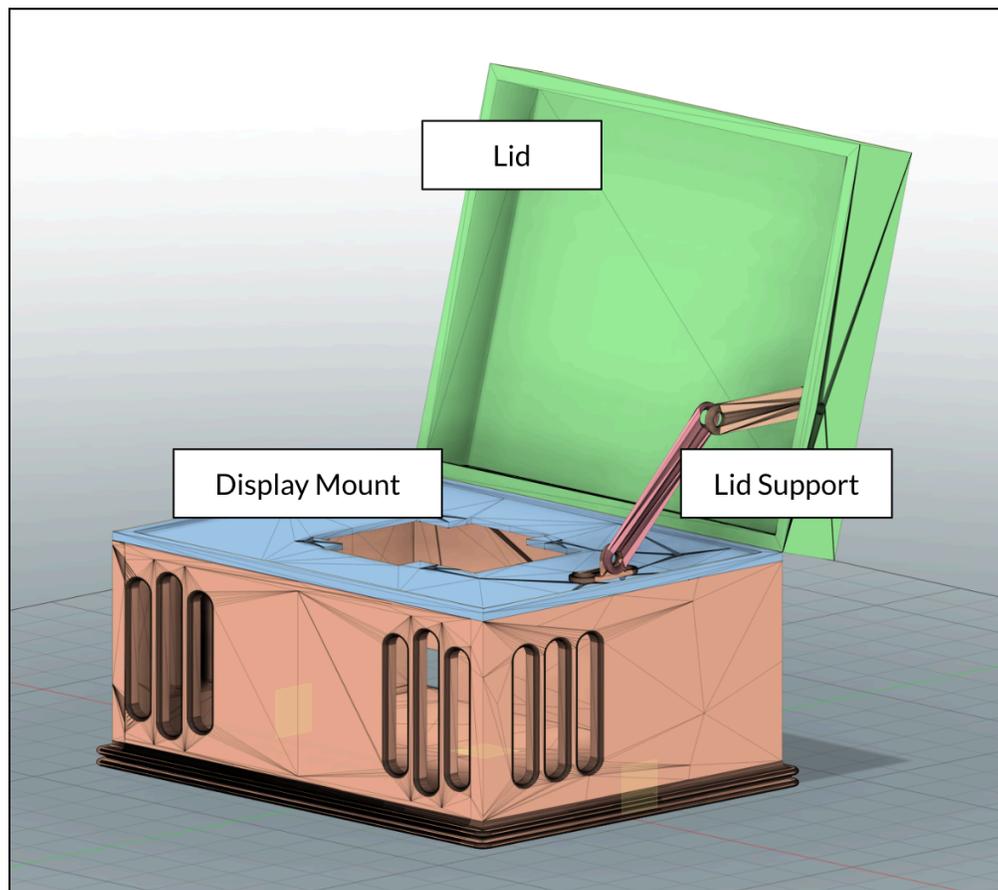


Figure 2: System Block Diagram

Users interact with *Roomify* through the Display System (top right), which has a touchscreen display for selecting room device presets, adding remotes, and controlling music. The Display System is controlled by the Box System, which is the central unit of *Roomify*. The Box System contains the main MCU, which controls infrared signal transceiving and Wi-Fi operations. The Repeater System redirects IR signals from the Box System to increase coverage within the room. Repeaters will be separate, battery-powered units placed throughout the room to relay IR signals to devices not in direct line of sight of the Box System. Finally, the Power System converts offline AC power to 3.3V for the Box and Display Systems.

## 2.2 Physical Design

*Roomify* will appear like a vinyl player with a hinged “wooden” box and square touchscreen display. The box will be 3D printed and painted with a wooden finish. The [square touchscreen display](#) will be mounted to the top panel of the box with the ribbon connector passed through a slit to the inside of the box. The box will house the PCBs for the power supply and MCU, as well as infrared LEDs and receivers.



*Figure 3: Front View of Physical Box Design*

The box will have two hinges connected to the lid that control the opening and closing of the box. For signal transmission, the infrared LEDs will be wired to the PCB and oriented in all directions (each IR LED has ~20-degree conical coverage). The box will be printed with white PLA plastic, which is transparent to infrared signals with slight (-0.5dB) attenuation [2]. The power supply PCB will be enclosed in an aluminium [case](#) to provide isolation from the rest of the system and for safe handling. The aluminum case will have two holes drilled into either side

for the outlet cord (input) and USB-C (output). To prevent shorting, the board will be suspended within the aluminum case with four M3 screws and 3D printed spacers.

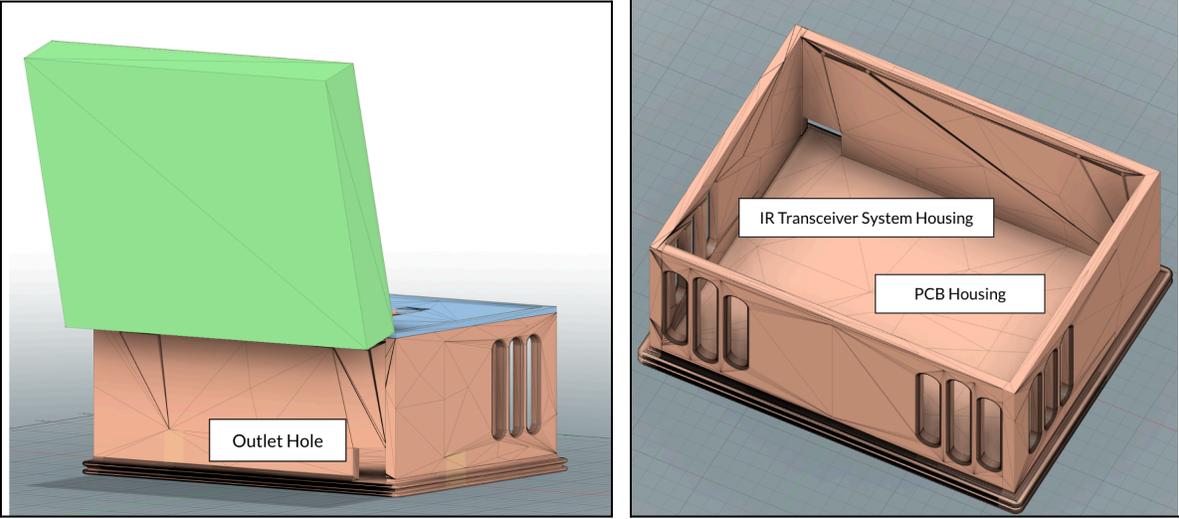


Figure 4: Back and Internal View of Physical Box Design

The Roomify box physical design was modified from an online project and build manual by Concept Bytes [3].



Requirements	Verification
<ul style="list-style-type: none"> <li>Receive and decode 38kHz NEC infrared signals with irradiance <math>&gt; 0.15 \frac{mW}{m^2}</math>.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an infrared receiver to an Arduino Nano (or other dev board with IR decoding capability).</li> <li>Point and click a button on a device remote (LED light remote) from ~20ft. away at the Arduino receiver and then the <i>Roomify</i> ESP32-S3.</li> <li>Compare decoded signal from Arduino to decoded signal from <i>Roomify</i> Transceiver System and confirm that data bits match exactly.</li> <li>Connect output of RMT peripheral on ESP32-S3 to oscilloscope and verify that there are no bit flips.</li> </ul>
<ul style="list-style-type: none"> <li>Transmit 38kHz NEC infrared signals at <math>&gt; 130 \frac{mW}{sr}</math>.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an infrared receiver to oscilloscope and test <i>Roomify</i> transmission at ~20ft from the receiver.</li> <li>Observe noise levels, signal integrity, and check for bit flips.</li> <li>Test transmission of a valid remote code signal with a device receiver and verify correct, predictable operation (e.g. turn LED lights red, then green, and then blue).</li> </ul>
<ul style="list-style-type: none"> <li>(Near) Omni-directional transmission of infrared signals, <math>&gt; 300^\circ</math> coverage along any axis.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an infrared receiver to oscilloscope.</li> <li>Place the <i>Roomify</i> box 10 ft. away from the receiver at the same height.</li> <li>Continuously rotate the box by 30 degrees, testing correct reception of the signal at the oscilloscope (no bit flips, relatively unnoisy signal).</li> <li>Repeat the test after elevating the receiver by 5 ft and confirm correct transmission.</li> </ul>

### 2.3.2 Repeater System

The Repeater System extends the infrared coverage of *Roomify* by receiving IR commands from the main Transceiver System and retransmitting them in a different direction (user can decide which direction to point the repeater in). Essentially, the repeaters will act as “mirrors” that can relay infrared signals to devices that are not in direct line of sight of the main transmission unit (e.g., behind a wall). Each repeater will have simplified receiver, transmitter, and microcontroller units. The receiver unit will have an IR sensor that passes the raw signal to the MCU. To avoid amplifying noise, the MCU will decode the signal and verify that it is a valid NEC remote code. The decoded signal is then re-encoded and re-transmitted. The repeater operates independently

using battery power and is designed to be lightweight and simple so users can place several of them throughout their room.

Requirements	Verification
<ul style="list-style-type: none"> <li>Receive and decode 38kHz NEC infrared signals with irradiance <math>&gt; 0.15 \frac{mW}{m^2}</math>.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an infrared receiver to an Arduino Nano (or other dev board with IR decoding capability).</li> <li>Point and click a button on a device remote (LED light remote) from ~20ft. away at the Arduino receiver and then the <i>Roomify</i> ESP32-S3.</li> <li>Compare decoded signal from Arduino to decoded signal from <i>Roomify</i> Transceiver System and confirm that data bits match exactly.</li> <li>Connect output of RMT peripheral on ESP32-S3 to oscilloscope and verify that there are no bit flips.</li> </ul>
<ul style="list-style-type: none"> <li>Transmit 38kHz NEC infrared signals in a single direction at <math>&gt; 130 \frac{mW}{sr}</math>.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an infrared receiver to oscilloscope.</li> <li>Place the <i>Roomify</i> box outside of direct line of sight from the receiver (e.g. around the corner of the wall).</li> <li>Place a repeater at a midpoint (~15ft.) between the <i>Roomify</i> box and the infrared receiver so that it is in direct line of sight of both.</li> <li>Configure the repeater's retransmission direction to point at the oscilloscope receiver.</li> <li>Activate <i>Roomify</i> signal transmission and confirm correct reception of the signal at the oscilloscope.</li> </ul>

### 2.3.3 *Roomify* Display Interface System

The Display System is the primary visual user interface for *Roomify*. The Display System consists of [capacitive touch TTL TFT](#) screen and a special [ESP32-S3 MCU](#) that can drive the RGB-666 display. The display MCU communicates with the TFT screen using a SPI interface (purple wired data path), enabling high-speed graphics updates such as spinning vinyl animations or preset visuals. The Display System communicates bidirectionally with the main *Roomify* MCU via UART (wired data path), allowing for full control over the display and for touchscreen triggers to be sent to the central MCU. In response, the *Roomify* MCU can execute IR transmission presets. All execution is still handled by the *Roomify* MCU, and the Display System simply acts as the user interface.

Requirements	Verification
<ul style="list-style-type: none"> <li>• Provide user touchscreen data to the main MCU via UART with latency &lt; 50ms</li> </ul>	<ul style="list-style-type: none"> <li>• Use an logic analyzer to measure timestamp when interrupt is generated at the Display MCU.</li> <li>• Use an logic analyzer to measure timestamp when UART packet is received by the <i>Roomify</i> MCU.</li> <li>• Measure average end-to-end latency across &gt;10 trials</li> <li>• Requirement passes if worst-case latency &lt; 50 ms.</li> </ul>
<ul style="list-style-type: none"> <li>• Display animations and images at &gt; 15 FPS.</li> </ul>	<ul style="list-style-type: none"> <li>• Use Qualia ESP32-S3 to display an animation (GIF) with 15 FPS.</li> <li>• Confirm that the animation does not lag, glitch, or otherwise display incorrectly.</li> </ul>
<ul style="list-style-type: none"> <li>• Display stored user presets and settings.</li> </ul>	<ul style="list-style-type: none"> <li>• Preload multiple presets involving &gt;3 room devices into memory on the <i>Roomify</i> MCU.</li> <li>• Verify correct preset names, correct icons, and correct associated IR mapping.</li> <li>• Use <i>Roomify</i> touchscreen display to select a preset and confirm matching operation of room devices.</li> </ul>
<ul style="list-style-type: none"> <li>• Provide an interface that allows users to use all functions of <i>Roomify</i>, such as selecting a preset and pairing new remotes.</li> </ul>	<ul style="list-style-type: none"> <li>• Confirm ability to navigate between select preset screen and add remote screen without errors.</li> <li>• Ensure all features accessible within <math>\leq 3</math> screen interactions.</li> </ul>

### 2.3.4 Power Systems (Box and Repeater)

There are two independent power architectures shown in the diagram. The power system for the *Roomify* box uses an AC outlet input, followed by power factor correction (PFC) and a flyback converter to generate 24V DC (see Figure 7) [6]. The box power system converts 120VAC from the outlet into regulated DC rails [7]. A buck converter then steps this down to 3.3V for logic circuits like MCU, IR subsystems, and Display System (see Figure 6).

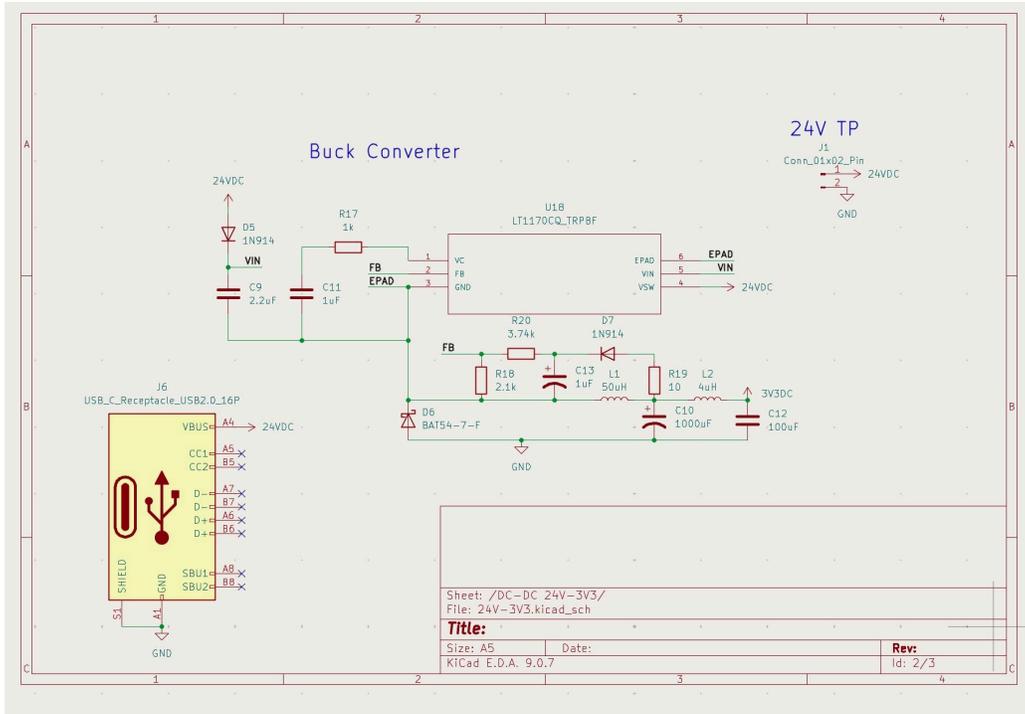


Figure 6: Schematic Diagram for 24V-3.3V Buck Converter

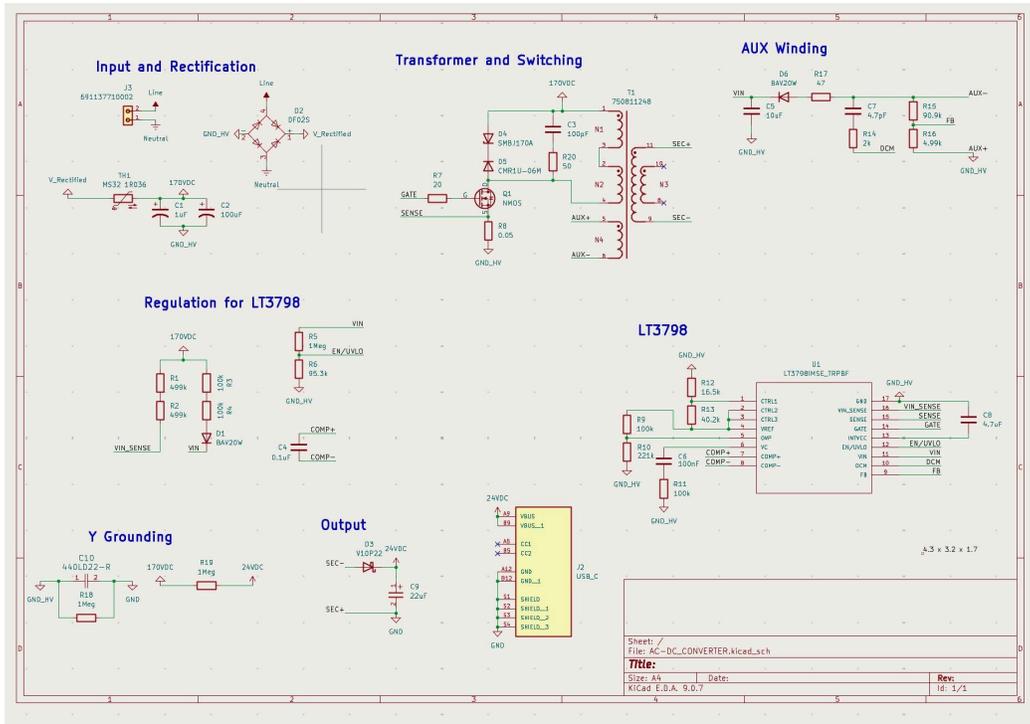


Figure 7: Schematic Diagram for AC-DC Converter

The power system for the repeaters uses a battery source regulated down to 3.3V for its MCU and IR circuits.

Both systems maintain electrical isolation between high-power (24V) and logic-level (3.3V) domains to ensure stable operation and safe interfacing between subsystems.

Requirements	Verification
<ul style="list-style-type: none"> <li>Convert 120V AC from outlets to 24VDC and 3.3V.</li> </ul>	<ul style="list-style-type: none"> <li>Connect the Roomify box to a 120V AC outlet and power on the system.</li> <li>Using PCB test points, place the multimeter negative probe on GND.</li> <li>Measure the 24V rail with the positive probe and confirm voltage is <math>24V \pm 0.5V</math>.</li> <li>Measure the 3.3V rail and confirm voltage is <math>3.3V \pm 0.3V</math>.</li> <li>Use an oscilloscope to probe 24V and 3.3V rails.</li> <li>Use the oscilloscope to measure ripple voltage and confirm that it is within <math>\pm 0.5V</math></li> </ul>
<ul style="list-style-type: none"> <li>Impose current limits for safe operation and outlet usage.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an adjustable electronic load to the 3.3V rail.</li> <li>Gradually increase current draw beyond rated limits.</li> <li>Observe system response (current limiting, shutdown, or protection trigger).</li> <li>Confirm no overheating or component damage occurs.</li> </ul>
<ul style="list-style-type: none"> <li>Power supply should provide peak 3A for the Roomify system at 3.3V +/- 0.3V.</li> </ul>	<ul style="list-style-type: none"> <li>Connect an electronic load to the 3.3V rail.</li> <li>Increase load current gradually up to 3A.</li> <li>Monitor voltage using a multimeter.</li> <li>Confirm voltage remains within <math>3.3V \pm 0.3V</math> under full load.</li> </ul>
<ul style="list-style-type: none"> <li>The battery system should provide peak 500mA at 3.3V +/- 0.3 V for <i>Roomify</i> repeaters.</li> </ul>	<ul style="list-style-type: none"> <li>Power repeater using a fully charged battery.</li> <li>Connect electronic load to 3.3V output.</li> <li>Increase current draw up to 500mA.</li> <li>Confirm voltage remains within <math>3.3V \pm 0.3V</math>.</li> </ul>

### 2.3.5 Box + Logic System

The Box System is the central hardware unit of Roomify and contains power conversion, logic control, and IR communication. The *Roomify* box will be constructed from wood and painted to look like a vinyl player with the square display acting as the spinning record (see Figure 1, Visual Aid).

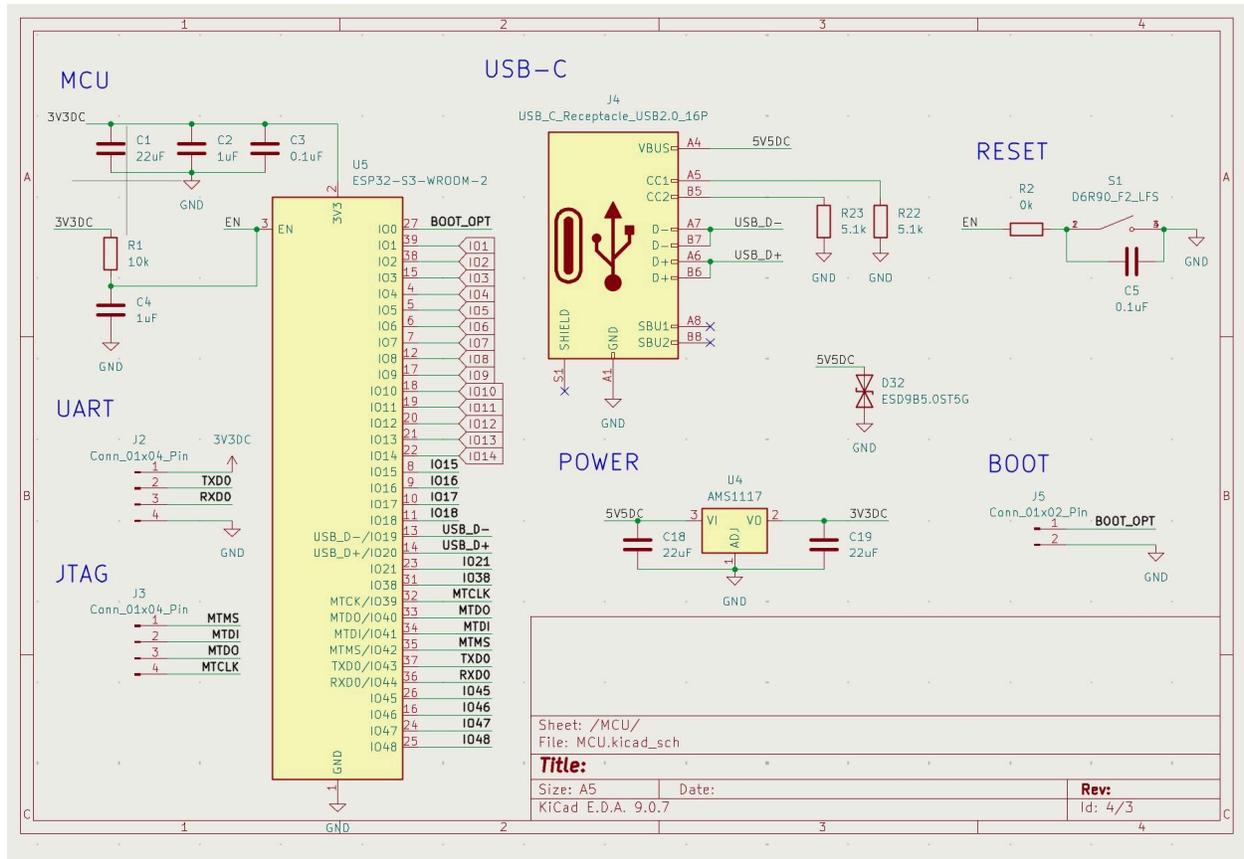


Figure 8: Schematic Diagram for ESP32-S3

The central controller for the Logic System is an ESP32-S3 MCU, coordinating IR transmission/reception, display communication, and Wi-Fi (for making API calls to Spotify, mobile web application control, etc.). The Transceiver System allows the box to learn remote codes and send omnidirectional NEC-encoded commands wirelessly to room devices and repeaters using the IRremote Arduino library [8]. The Box System also connects to the Display System via wired 3.3V power and UART communication, acting as the master controller for the

entire Roomify device. Finally, the MCU will be able to receive user instructions from mobile devices via Wi-Fi, allowing users to use *Roomify* through a web application on their phones.

Requirements	Verification
<ul style="list-style-type: none"> <li>Control the Display System</li> </ul>	<ul style="list-style-type: none"> <li>Send known control commands from the main MCU to the Display System via UART.</li> <li>Visually verify correct screen response (screen updates, animations, UI changes)</li> </ul>
<ul style="list-style-type: none"> <li>Control the Transceiver System</li> </ul>	<ul style="list-style-type: none"> <li>Send IR transmission commands to the Transceiver System.</li> <li>Verify correct operation of room devices.</li> <li>Use an infrared receiver connected to an oscilloscope to validate signal transmission (check that bits match).</li> </ul>
<ul style="list-style-type: none"> <li>Receive and process user requests over Wi-Fi within 500ms</li> </ul>	<ul style="list-style-type: none"> <li>Connect system to Wi-Fi network.</li> <li>Trigger a request from the mobile/web application.</li> <li>Log timestamp when request is sent and when action is executed on the MCU.</li> <li>Confirm that average action execution latency is &lt; 500ms.</li> </ul>
<ul style="list-style-type: none"> <li>Store user information (presets or remote signal codes) to a database over Wi-Fi reliably</li> </ul>	<ul style="list-style-type: none"> <li>Create or modify presets through the application.</li> <li>Confirm preset data is written to database.</li> <li>Restart the <i>Roomify</i> system and reload the application.</li> <li>Verify stored data persists and matches expected values.</li> </ul>
<ul style="list-style-type: none"> <li>Make valid HTTP requests over Wi-Fi to Spotify API</li> </ul>	<ul style="list-style-type: none"> <li>Use ESP32-S3 to send a HTTP request to Spotify to play a song.</li> <li>Verify HTTP status code is 200.</li> <li>Confirm correct behavior on the Spotify account (song plays).</li> </ul>

### 2.3.6 Mobile Device Control System

In addition to the *Roomify* display interface, users will also be able to control *Roomify* with their phones via a web application. User presets and virtual remotes will be stored in a database linked to user accounts. After signing in to the web application, users can access the same interface as the *Roomify* display to select presets and control room devices. This system will use Wi-Fi to

allow for wireless control of the *Roomify* box. When users select an option on their phone, a request will be sent to the *Roomify* ESP32 MCU, triggering appropriate actions.

---

Requirements	Verification
<ul style="list-style-type: none"><li>Communicate with the <i>Roomify</i> Box System over Wi-Fi with &lt; 200ms latency</li></ul>	<ul style="list-style-type: none"><li>Connect the mobile device and <i>Roomify</i> Box to the same Wi-Fi network.</li><li>Instrument the web application to log the timestamp when a user presses a control button.</li><li>Log the timestamp on the <i>Roomify</i> ESP32 when the command is received.</li><li>Calculate the time difference across at least 20 trials and verify that average latency is &lt; 200ms.</li></ul>
<ul style="list-style-type: none"><li>Display stored user presets and settings.</li></ul>	<ul style="list-style-type: none"><li>Create a user account with predefined presets stored in the database.</li><li>Log into the web application using the account.</li><li>Verify that all presets, names, icons, and settings are correctly displayed.</li></ul>
<ul style="list-style-type: none"><li>Provide a clean, intuitive interface for users to add new remotes, create presets, and control devices through Roomify (time to teach someone how to use the application should be &lt; 3 minutes).</li></ul>	<ul style="list-style-type: none"><li>Select at least 5 users unfamiliar with the system.</li><li>Provide a short 1 minute explanation of the application's purpose (no walkthrough).</li><li>Assign tasks:<ul style="list-style-type: none"><li>Add a new remote</li><li>Create a preset</li><li>Trigger a device</li></ul></li><li>Measure the time required to understand and complete tasks and verify that &lt; 3 minutes of instruction were necessary.</li></ul>

---

## 2.4 Tolerance Analysis

### 2.4.1 Infrared Signal Transmission

Infrared signal transmission is a core part of *Roomify*. The infrared signals from the Roomify box must be strong enough to reach the device receivers, either directly or through the *Roomify* repeaters.

From the [datasheet](#) for the [infrared LED](#) we plan to use, the IR LED has a radiant intensity of 130 mW/sr when using 200mA pulses [9]. To determine how much power reaches the receiver, we multiply this value by the solid angle subtended by the receiver.

Assuming a worst case distance of 10m (for a 15 ft. x 15 ft. x 10ft. room, the distance from opposite corners of the room is 7 meters), we can calculate the irradiance of the signal at the receiver as follows:

$$\begin{aligned} E &= \frac{I_e}{r^2} \\ E &= \frac{130 \frac{mW}{sr}}{100m^2} \\ E &= 1.3 \frac{mW}{m^2} \end{aligned}$$

[Standard infrared receivers](#) have  $E_{\min} > 0.15 \frac{mW}{m^2}$  [10]. Calculating the link margin as  $10 \log_{10}(E/E_{\min})$ , we get a link margin of 9.37 dB.

To account for insertion loss from the PLA box, we assume signal attenuation of approximately -0.5 dB, resulting in a substantial margin of 8.87 dB for our device. In addition, the maximum rating for the infrared LED is 1A pulses (for 750 mW/sr), meaning we can increase signal strength significantly compared to using 200mA pulses if necessary.

With 1A pulses, the irradiance at a receiver 10 meters away is about  $7.5 \frac{mW}{m^2}$ .

### 2.4.2 Box Power System

Another critical component of *Roomify* is the power subsystem and its ability to deliver enough power and current. The voltage regulator IC we chose for our microcontroller unit is the LT1170 [7]. This IC allows us to have a wide input voltage and an output of 3v3. Because *Roomify* will have two MCUs (one “main” MCU and one special MCU for controlling the touchscreen display, we will need around 2.4A to power both (assuming a peak current draw of 1.2A for an ESP32-S3). *Roomify* will drive at most three infrared LEDs at a time at 200mA pulses, requiring peak draw of 600mA.

Combining the current requirements for the MCUs and the IR LEDs gives a total budget of 3A. The rating for the LT1170 is 5A at 3v3. For our 24V system that powers our 3v3 system, it is required to output around 10 Watts. Our flyback converter can deliver up to 40 Watts of power, so we will have an ample amount of power for our system. Below (Figure 9) is a simulation of our LT1170 component that delivers enough voltage and current.

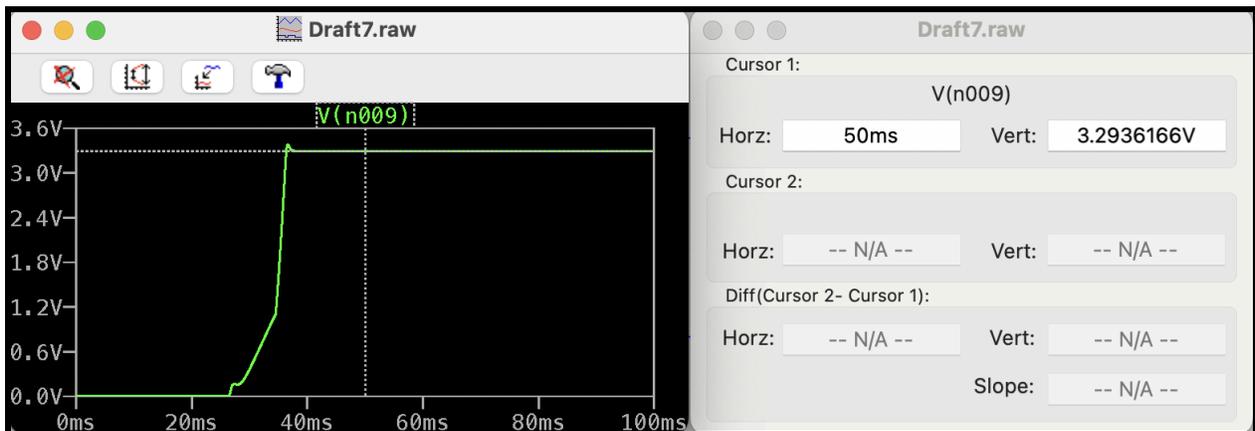


Figure 9: Simulation of AC-DC Power Converter Providing 3.3V

### 3. Cost and Schedule

The total cost for parts before shipping is \$186.23. Adding 10% for sales tax and shipping gives a total of \$204.85. Assuming a \$40/hour salary and 50 hours to complete the project for each team member, the total labor cost is \$40/hr \* 50 hours \* 3 team members = \$6000. Summing up materials and labor, the total cost to develop and manufacture *Roomify* is \$6204.85.

#### 3.1 Cost Analysis

Description	Manufacturer	Quantity	Price	Link
4" SQUARE RGB TTL TFT DISPLAY	Adafruit	1	\$44.95	<a href="#">Link</a>
QUALIA ESP32-S3	Adafruit	1	\$19.95	<a href="#">Link</a>
AL-71P DIECAST ALUMINUM ENCLOSURE	Polycase	1	\$11.54	<a href="#">Link</a>
ESP32-S3-WROOM-2	Espressif Systems	1	\$11.44	<a href="#">Link</a>
IR333-A	Adafruit	25	\$17.00	<a href="#">Link</a>
IC OFFLINE SWITCH FLYBACK 16MSOP	Analog Devices Inc.	1	\$8.30	<a href="#">Link</a>
XFMR LED DR AC/DC CONV 300UH TH	Würth Elektronik	1	\$7.48	<a href="#">Link</a>
IC REG BUCK BOOST ADJ 5A DDPK-5	Analog Devices Inc.	1	\$19.96	<a href="#">Link</a>
CORD 18AWG NEMA5-15P - CBL 1.67'	Orion Fans	1	\$6.11	<a href="#">Link</a>
IC OSC SNGL TIMER 100KHZ 8-SOIC	Texas Instruments	5	\$1.40	<a href="#">Link</a>
IC GATE DRVR LOW-SIDE SOT26	Diodes Incorporated	20	\$6.40	<a href="#">Link</a>
IC REG LINEAR 3.3V 1A SOT-223-3L	UMW	1	\$0.30	<a href="#">Link</a>
MOSFET N-CH 20V 800MA SSM	Toshiba Semiconductor and Storage	20	\$6.40	<a href="#">Link</a>
Box + Repeater 3D Print	CUC Fab Lab	1	\$25	<a href="#">Link</a>

### 3.2 Schedule

Week	Task	Person
<b>March 1 - March 7</b>	Make BOM and order parts	Everyone
	Complete Box Design	Benjamin
	Complete Repeater PCB Design	Owen
	Complete Power Supply & MCU PCB Design	Warren
<b>March 8 - March 14</b>	Solder Power Supply & MCU PCBs	Benjamin and Owen
	Revise Power Supply & MCU PCB Designs	Warren
	Revise Repeater PCB Design	Owen
<b>March 15 - March 21</b>	Complete physical design for Repeater (3D Print)	Benjamin
	Set up user accounts, server, and data storage.	Owen
	Revise Power Supply & MCU PCB Designs	Warren
<b>March 22 - March 28</b>	Complete add remote flow (receive and store infrared remote codes)	Benjamin
	Begin working on <i>Roomify</i> web-app interface	Owen
	Working offline power supply	Warren
<b>March 29 - April 4</b>	Add transmission capability (operate LED string lights with <i>Roomify</i> )	Benjamin
	Finish V1 of <i>Roomify</i> web-app interface and link with server and user accounts.	Owen
	Modify aluminum enclosure to encase power supply and connect with MCU board	Warren
<b>April 5 - April 11</b>	Working UART communication between <i>Roomify</i> MCU board and display screen board	Benjamin
	Adapt <i>Roomify</i> web-app interface to square display screen	Owen
	Assemble <i>Roomify</i> box (3D print + assembly), working repeaters	Warren
<b>April 12 - April 18</b>	Working touchscreen control of <i>Roomify</i> through the square display screen	Benjamin
	Display Spotify/music controls on <i>Roomify</i> square display screen	Owen
	Test and verify subsystems	Warren
<b>April 19 - April 25</b>	Finalize assembly + integration tests, prepare for demo	Everyone
<b>April 26 - May 2</b>	Final Presentation + Demo	Everyone

## 4. Ethics

### 4.1 Impact and Engineering Considerations

As engineers, we have an ethical and professional responsibility to design safe and transparent systems. The IEEE Code of Ethics states that engineers should “hold paramount the safety, health, and welfare of the public” (IEEE Code of Ethics, Principle 1) [11]. Given this context, the main considerations in designing *Roomify* are electronic safety and data privacy. Because *Roomify* will be constantly powered on in users’ rooms, it is extremely important to mitigate any electrical risks through electrical isolation, safe operating voltages, fuses, and controlled signal transmission.

The most critical component in *Roomify* is the power subsystem. Because we are designing our own power supply to work with standard 120V AC household outlets, we have adequate safeguards to guarantee safe operation. The power subsystem will use an isolated AC-DC converter to separate high and low voltage components. To reduce shock risk, the *Roomify* PCB will be housed in a wooden box to prevent user access. For development, the box will be easily disassembled (using sliding panels and walls) so that we can extract the PCB if necessary. The only electrical component that users can interact with is the capacitive touchscreen (purchased component), which is designed to be safe to touch. In addition, we will use fuses to enforce current limits and prevent overheating and component failure. Mechanical components, such as the hinges, will be enclosed within the wooden box to prevent pinch hazards.

By identifying potential hazards, classifying risk, and implementing necessary safeguards, *Roomify* will adhere to the UL 62368-1 standard for audio, video, and ICT equipment. When working with AC mains power and high-voltage circuits during development, the team will follow UIUC lab safety policies such as wearing appropriate PPE, ensuring circuits are de-energized before interacting with them, and using one hand instead of two. Due to the high-risk nature of working with AC mains power, we have also included a [lab safety document](#) with specific protocol for dealing with high-voltage/current.

Furthermore, the ACM Code of Ethics requires computing professionals to “avoid harm” (ACM Code 1.2) and to “respect privacy” (ACM Code 1.6). To align with these principles, *Roomify* will

completely avoid invasive data collection. The system does not include cameras or microphones, and it only stores user-provided IR remote codes and preset configurations. Any network communication, such as Spotify API usage, is limited to authenticated user requests and follows the terms of service of the external platform. Though we are not storing private data, we will implement user accounts to protect user information. For the web application, we will implement proper authentication protocols, input validation, and protection against common web vulnerabilities such as SQL injection and cross-site request forgery. To prevent misuse, *Roomify* will only transmit IR commands that have been explicitly learned and labeled by the user, reducing the risk of unintended control of nearby devices.

Additionally, IEEE emphasizes that engineers must “be honest and realistic in stating claims or estimates” (IEEE Code of Ethics, Principle 3). Throughout this course and design process, we will ensure that all documentation and demonstrations accurately represent the capabilities and limits of our system. For example, instead of implying universal compatibility, we will clearly specify which IR devices and API platforms *Roomify* supports.

From a societal and economic perspective, *Roomify* provides a safe, low-cost alternative to existing smart home ecosystems, improving accessibility for students and renters who cannot modify their living spaces or invest in smart devices. Environmentally, *Roomify* will not increase waste or have a significant carbon footprint since it is a reusable, low-power system. Above all, *Roomify* is a product for improving users’ lifestyles, and all design choices will prioritize safety, regulatory standards, and user privacy.

## **4.2 Lab Safety Manual**

The main risks in the project due to the 120VAC source, soldering, and energy-storing components [12]. Potential hazards include electrical shock, fire from overcurrent conditions, battery failure, and equipment damage during assembly or testing. These risks are reduced through proper component selection and secure design, but they cannot be completely removed. Therefore, we must define formal operating procedures that ensure consistent and safe practices during construction.

### **1. Workspace Safety**

- Maintain a clean, dry, uncluttered bench space
- Keep liquids away from powered circuits

### **2. Soldering Safety**

- Always return iron to a stable soldering stand
- Use fume extraction or work in a ventilated area
- Wash hands after handling solder
- Use heat-resistant silicone mat
- Never leave soldering iron unattended while powered

### **3. AC Power Safety**

- Always unplug before servicing and ensure circuits are de-energized
- Apply Lockout-Tagout (LOTO) during testing
- Maintain 3 feet clearance from breaker panel
- Use one hand when interacting with high-voltage circuits and place the other hand in pocket
- Wear appropriate PPE
  - Rubber-insulated protective gloves and approved protective (leather) gloves
  - Safety glasses (ANSI Z87.1 certified)
- Test GFCI before use

### **4. Battery Safety**

- Store in a fire-resistant area
- Follow proper disposal methods

### **5. Capacitor Discharge Procedure**

- Power off system
- Wait 10 seconds minimum
- Verify discharge with multimeter
- Use discharge resistor if necessary
- Never short capacitors directly without resistance

### **6. Personal Equipment Requirements**

- Closed-toe shoes
- Safety glasses
- Remove conductive jewelry
- One-hand technique when probing live circuits

In addition to the procedures described above, UIUC electrical safety precautions should also be followed according to the following manual:

<https://drs.illinois.edu/Page/SafetyLibrary/ElectricalSafetyInTheResearchLaboratory> [13].

## References

- [1] Chan, Manila. “Tiktok Made Me Buy It! Tiktok Home Decor Trends.” *The Daily Well*, 18 June 2025, <https://thedailywell.life/gadgets-gear-for-families/tiktok-home-decor-trends/>. Accessed 27 Feb. 2026.
- [2] Amendola, Caterina, et al. “Optical Characterization of 3D Printed PLA and ABS Filaments for Diffuse Optics Applications.” *PloS One*, U.S. National Library of Medicine, 16 June 2021, <https://pmc.ncbi.nlm.nih.gov/articles/PMC8208549/>. Accessed 27 Feb. 2026.
- [3] *Spotify Record Player STLS + Build Manual* | Concept Bytes, <https://www.patreon.com/posts/spotify-record-127390794>. Accessed 27 Feb. 2026.
- [4] “Data Formats.” *Vishay Semiconductors*. <https://www.vishay.com/docs/80071/dataform.pdf>. Accessed 27 Feb. 2026.
- [5] “Remote Control Transceiver (RMT).” *Espressif Systems*, [docs.espressif.com/projects/esp-idf/en/stable/esp32s3/api-reference/peripherals/rmt.html](https://docs.espressif.com/projects/esp-idf/en/stable/esp32s3/api-reference/peripherals/rmt.html). Accessed 27 Feb. 2026.
- [6] “LT3798 Isolated No Opto-Coupler Flyback Controller with Active PFC” *Linear Technology*, [www.analog.com/media/en/technical-documentation/data-sheets/3798fa.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/3798fa.pdf). Accessed 27 Feb. 2026.
- [7] “LT1170/LT1171/LT1172 High Efficiency Switching Regulators”. *Linear Technology*, [www.analog.com/media/en/technical-documentation/data-sheets/117012fi.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/117012fi.pdf). Accessed 27 Feb. 2026.
- [8] “IRremoteESP8266.” *Arduino Docs*, <https://docs.arduino.cc/libraries/irremotesp8266/>. Accessed 27 Feb. 2026.
- [9] “5mm Infrared LED Technical Data Sheet.” *Adafruit*, [cdn-shop.adafruit.com/datasheets/IR333\\_A\\_datasheet.pdf](http://cdn-shop.adafruit.com/datasheets/IR333_A_datasheet.pdf). Accessed 27 Feb. 2026.

- [10] “IR Receiver Modules for Remote Control Systems.” *Vishay Semiconductors*.  
<https://www.vishay.com/docs/82491/tsop382.pdf>. Accessed 27 Feb. 2026.
- [11] “IEEE Code of Ethics.” *IEEE*, [www.ieee.org/about/corporate/governance/p7-8](http://www.ieee.org/about/corporate/governance/p7-8). Accessed 27 Feb. 2026.
- [12] Hall, Brian. “Debunking Low Voltage Myths: Why 120 Volts Isn’t as Safe as You Think.” *Guidant Power*, 14 May 2025,  
[www.guidantpower.com/post/enhancing-electrical-safety-ppe-and-insulation-best-practices-for-120-volt-control-panels](http://www.guidantpower.com/post/enhancing-electrical-safety-ppe-and-insulation-best-practices-for-120-volt-control-panels). Accessed 27 Feb. 2026.
- [13] “Division of Research Safety.” *University of Illinois Urbana-Champaign Division of Research Safety*.  
<https://drs.illinois.edu/Page/SafetyLibrary/ElectricalSafetyInTheResearchLaboratory#ElectricalHazards>. Accessed 27 Feb. 2026.