Catching Z's: Bedside Sleep Assistance System

ECE 445 Design Document – Spring 2026

Project #88

Srikar Palani, Prineet Parhar, Suprathik Vinayakula

Professor: Arne Fliflet

TA: Zhuchen Shao

Contents

1. Introduction 1
1.1 Problem...1
1.2 Solution 1
1.2 Visual Aid 1
1.3 High-level Requirements List 1
2 Design 3
2.1 Physical Design...3
2.2 Block Diagram 3
2.3 Functional Overview & Block Diagram Requirements 4
2.3.1 Audio Sensing Subsystem 4
2.3.2 Processing Subsystem...4
2.3.3 Audio Output Subsystem...4
2.3.4 User Interface & Power Subsystem 4
2.4. Design Verification * Verification at each step needs to be added to each subsystem bullet point .6
2.4.1 Latency Verification 6
2.4.2 Audio Output Verification..6
2.4 Hardware Design...7
2.5 Software Design 8
2.6 Commercial Component Selection * Might Not Need this 9
2.7 Tolerance Analysis 10
3 Cost and Schedule..11
3.1 Cost Analysis 11
3.2 Schedule...12
4. Ethics and Safety..13
References 15
Appendix A * Move this to verification table in 2.3 Subsections 16
Requirement and Verification Table...16

1. Introduction

1.1 Problem

Sleep fragmentation caused by sudden environmental noise is a widespread issue in urban and suburban environments. Common disturbances such as sirens, barking dogs, traffic, HVAC cycling, door slams, or late-night conversation can abruptly wake individuals during light sleep stages. Even brief awakenings disrupt natural sleep cycles, reducing sleep quality and negatively impacting long term health. Conventional white noise machines attempt to mitigate this issue by producing a constant masking sound throughout the night. While continuous white noise can reduce the perceived impact of background sounds, it introduces several limitations. First, it operates at a fixed volume regardless of the environment, resulting in unnecessary sound exposure during already quiet periods. Prolonged exposure to continuous masking noise may itself become disruptive or uncomfortable for users. Second, fixed level white noise may be insufficient against sudden, high volume disturbances such as sirens or sharp transient sounds. In such cases, the masking sound may not respond quickly enough or loud enough to prevent awakening. Furthermore, traditional white noise devices do not actively monitor the acoustic environment. They lack the ability to differentiate between baseline ambient noise and disruptive sound events. As a result, they cannot adapt their output dynamically to changing environmental conditions. This absence of real-time acoustic analysis prevents conventional systems from providing targeted, efficient masking. Smart speakers and mobile applications offer limited automation but often rely on cloud connectivity or pre-set timers. These approaches introduce privacy concerns when microphones are used in bedroom environments, and they still fail to deliver rapid, event based responses to unpredictable disturbances. Therefore, there exists a need for a locally processed, low latency embedded system capable of continuously monitoring ambient sound, establishing a dynamic baseline, and generating adaptive masking noise only when a disturbance exceeds a defined threshold. Such a system must respond quickly enough to prevent full awakening, operate reliably throughout an entire sleep cycle, and avoid unnecessary sound output during quiet periods. Addressing these limitations forms the foundation of the Catching Z's system design.

1.2 Solution

As a solution we propose Catching Z's, a bedside system that monitors ambient audio in and adaptively generates masking noise in response to disruptive sound events. The device uses a high sensitivity electret microphone to constantly sample room audio. An onboard microcontroller (ESP32-S3) processes this data to establish a baseline for the ambient noise. When the system detects a sudden spike in amplitude that exceeds the established baseline by a set threshold it activates the audio output subsystem.

The audio subsystem generates masking audio that smoothly fades in to neutralize the disturbance. Once the environmental noise returns to the baseline level, the masking noise gradually fades out. This adaptive response minimizes unnecessary noise exposure while preventing the masking system itself from waking the user. The user can adjust the sensitivity of the detection and the maximum volume of the masking noise using a rotary encoder interface.

1.2 Visual Aid

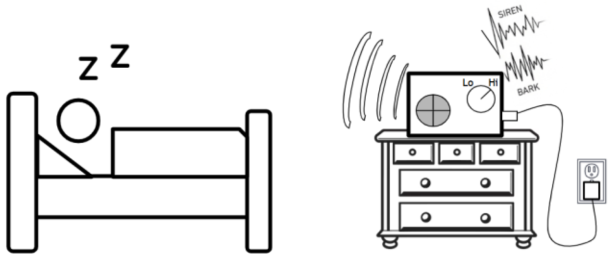

Figure 1: Representation of the Catching Z's system in a bedroom environment. The device (on the nightstand) detects incoming noise and emits a masking sound around the user, powered by a standard Micro USB-B wall connection.

1.3 High-level Requirements List

- **Detection Latency:** The system must trigger the masking noise playback within 100 milliseconds of a sound event that exceeds a baseline by ≥ 10 dB to effectively mask the disturbance.
- **Output Sound Pressure Level:** The audio subsystem must be capable of producing a masking noise over an adjustable range of 40 dB to 75 dB measured at 0.5 meters (standard bedside distance).
- **Continuous Operation Stability:** The system must operate continuously for a minimum duration of 8 hours (a standard sleep cycle) without any software or hardware crashes.

2 Design

2.1 Physical Design

The physical design of Catching Z's consists of a compact enclosure designed for bedside tables. It houses the PCB and speaker. The PCB has a footprint of 2.825 inches by 3.125 inches and will be held by four corner mounting holes.

The enclosure will have cutouts to allow user access to the buttons, the encoder, and windows to the LEDs. These are placed on the front of the device, while the cutout to access the USB port will be on the back of the device. The enclosure will have a grill of holes to allow sound from the internal speaker, and there will be an additional hole near the microphone. The microphone will be separated from the speaker and facing a different direction to avoid sound collision. Further, there will be internal separation between microphone and speaker in the enclosure. Finally, there will be small vents near the regulator to manage thermals for overnight device operation.

2.2 Block Diagram

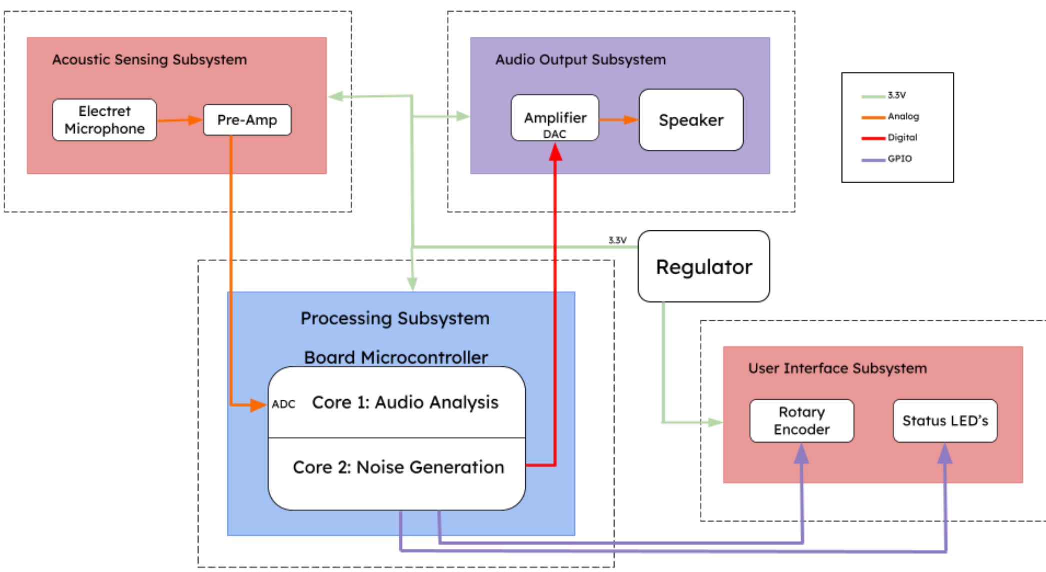

Figure 2: Block Diagram of the Catching Z's Subsystems interfacing with each other

The design of Catching Z's is divided into three primary subsystems: Sensing, Processing/Audio Output, and User Interface/Power. The sensing subsystem produces a digitized stream that the processing subsystem analyzes to maintain a moving baseline of ambient noise. When an event exceeds the baseline by **≥ 10 dB**, the processing subsystem works with the audio output chain to generate masking noise (white/pink/brown) that is faded in and then faded out. The user interface configures sensitivity and maximum masking volume, while the power subsystem ensures low noise sensing and reliable audio output for an entire sleep cycle.

2.3 Functional Overview & Block Diagram Requirements

2.3.1 Audio Sensing Subsystem

The acoustic sensing subsystem uses an electret microphone with a MAX4466 preamplifier. The microphone output is centered around mid-supply and fed into the ESP32-S3 ADC. The ADC samples at approximately 16 kHz. A moving average baseline is maintained. A disturbance is detected when the RMS value exceeds the baseline by 10 dB. This subsystem directly supports the 100-millisecond latency requirement by using short analysis windows and fast sampling. If the gain is too low, disturbances will not be detected. If it clips or is too noisy, false triggers will occur. A design risk is incorrect gain selection. If the microphone gain is too high, loud sounds will saturate the ADC. If too low, the 10 dB increase may not be measurable. This can be evaluated by measuring peak voltages at expected sound levels and ensuring they remain within the ADC range.

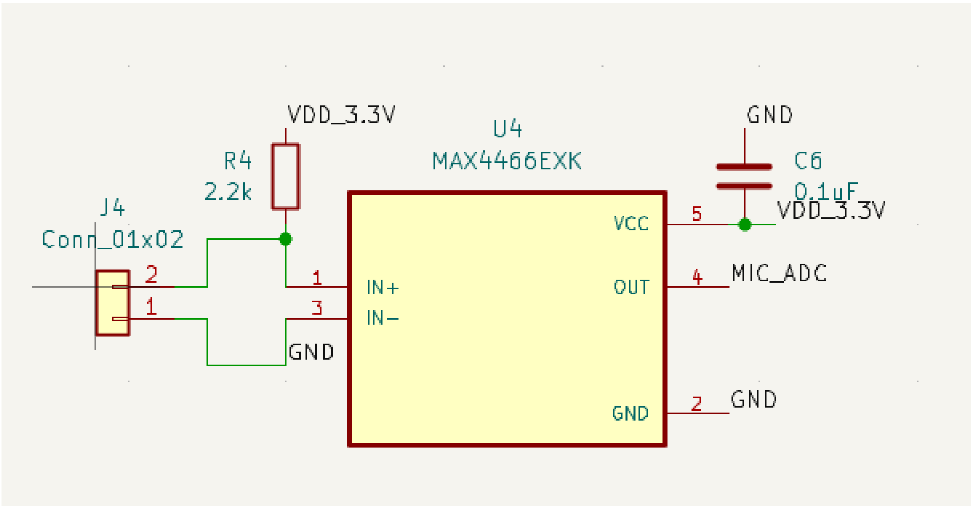

Figure 3: Circuit Schematic of components and connections involved for the Audio Sensing Subsystem

Requirement	Verification
The system shall detect a sound event when the RMS value exceeds the moving average baseline by greater than or equal to 10 dB.	1. Use a calibrated sound source to establish a steady baseline. 2. Increase source volume by 10 dB. 3. Poll the ESP32 to confirm whether there is a disturbance.
ADC should sample microphone output at 16 kHz.	1. Use a timer to measure the interval of ADC interrupts.

2.3.2 Processing Subsystem

The processing subsystem uses the ESP32-S3-WROOM-1 microcontroller. One core handles sampling and detection, while the second core generates masking noise. A simple state machine is used with listening, masking, and release states. When a disturbance is detected, the processor sends audio data to the amplifier through I2S. The total delay includes sampling time, RMS calculation, decision time, and audio startup time, and is kept below 100 ms by limiting frame size and minimizing buffering.

The processor must continuously compute RMS values, update the baseline, and respond quickly to changes. If processing falls behind or buffers overflow, the latency requirement will not be met. The subsystem must also apply a smooth fade-in and fade-out to avoid sudden sounds. The envelope can be implemented as a simple ramp where gain increases gradually over a short period. Without stable processing timing and correct task handling, the system would miss disturbances or respond too slowly.

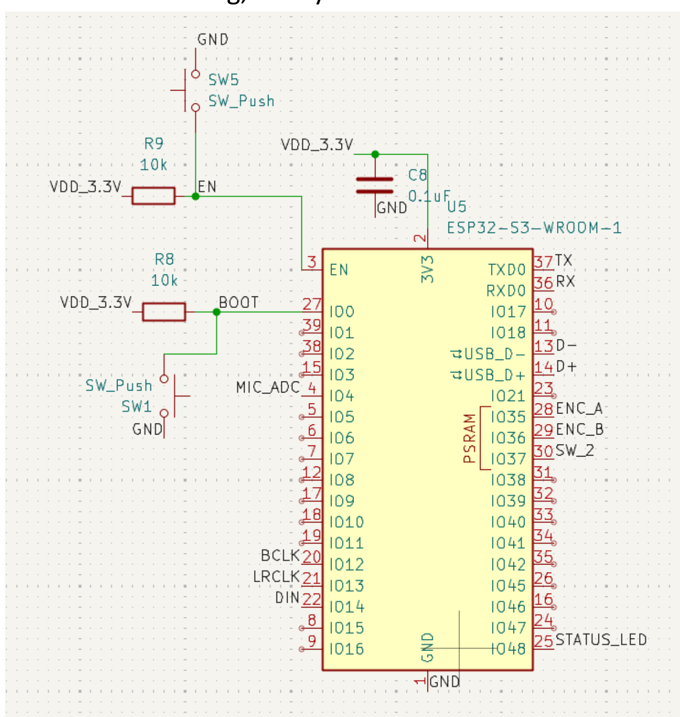

Figure 4: Circuit Schematic of components and connections involved for the Processing Subsystem

Requirement	Verification
Total system response time should be less than 100 ms.	1. Trigger a sudden sound spike. 2. Use a timer or oscilloscope to measure the time it takes.
The system should default to listening to the room and only change to masking when the threshold is met.	1. Run the system in a room for an hour. 2. Make sure that there are no false positives that cause the state of the system to change.

2.3.3 Audio Output Subsystem

The audio output subsystem consists of a MAX98357A I2S Class-D amplifier and a 4 Ω, 3 W speaker. The microcontroller generates white, pink, or brown noise digitally and streams it to the amplifier. The expected sound pressure level can be estimated using $\boldsymbol{SPL = S + 10\log(P) - 20\log(r)}$, where S is speaker sensitivity in dB at 1 W and 1 m, P is power in watts, and r is distance in meters. At 0.5 meters, the distance adds about 6 dB compared to 1 meter. With typical small speaker sensitivity, the required 75 dB SPL is achievable with well under 1 W of power. This subsystem must produce adjustable output from about 40 dB to 75 dB SPL at 0.5 meters. It must respond immediately to processor commands and avoid clicks or distortion. A design risk is speaker sensitivity variation. If the speaker is less efficient than expected, the maximum SPL may not reach 75 dB. This risk will be tested using a sound level meter at 0.5 meters and adjusting digital gain as needed.

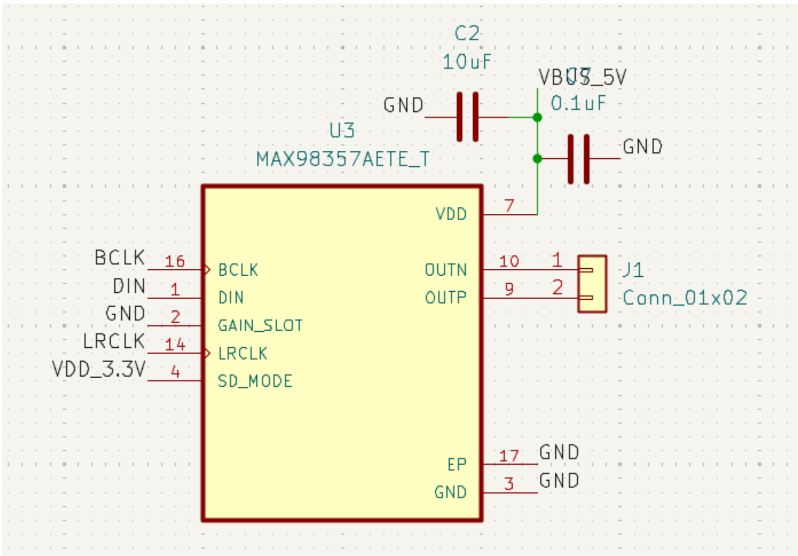

Figure 5: Circuit Schematic of components and connections involved for the Audio Output Subsystem

Requirement	**Verification**
The output should be between 40 and 75 dB at 0.5 meters.	So, set a microphone up and test the bounds to make sure it is between that
The device must be able to operate for the entire 8 hours of night.	**1. Check for power via the USB-B port.** **2. Then, turn the noise on for the whole 8 hours and check if it operates.** **3. Additionally, you can check the temperature of the system to make sure it's not overheating.**

2.3.4 User Interface Subsystem

The user interface consists of a rotary encoder connected to GPIO pins. The encoder adjusts detection sensitivity and maximum masking volume. Status LEDs indicate power and masking activity. **Add more

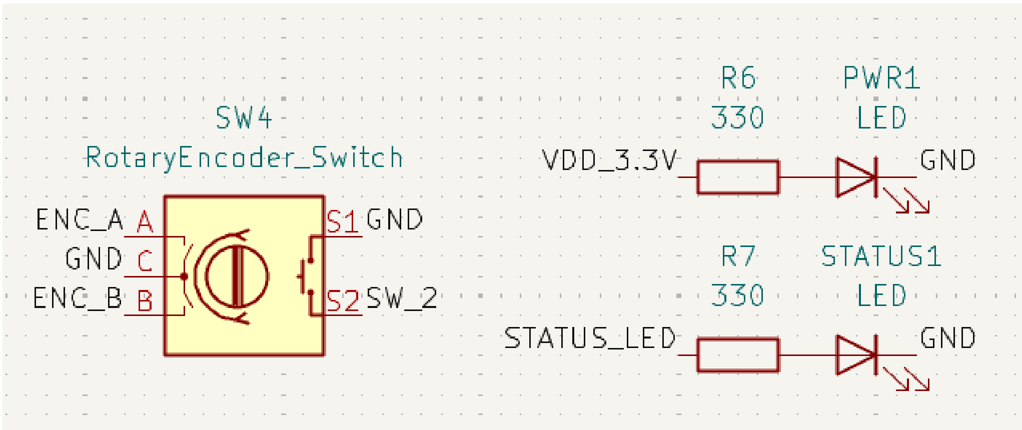

Figure 6: Circuit Schematic of components and connections involved for the User Interface Subsystem

Requirement	Verification
The rotary encoder should adjust the sensitivity and masking volume.	1. Rotate the encoder and measure both when the trigger for an event occurs and how loud the noise is. If it follows the expected behavior, then it should work.
The LEDs should provide accurate feedback.	1. Once the system is powered on, confirm that the power LED reflects that. 2. Confirm that the status LED color reflects the correct mode that the system is in (baseline vs. Masking).

2.4. Design Verification

The verification process focuses on ensuring the system meets the criteria for success.

2.4.1 Latency Verification

To verify the requirement that masking noise triggers within 100 milliseconds of a detected event, we utilize an oscilloscope to monitor both the analog microphone input and the I2S output line. A controlled noise impulse is generated, and the time difference between the input spike and the activation of the audio output is measured.

2.4.2 Audio Output Verification

The audio subsystem is tested using a calibrated decibel meter placed at a standard bedside distance (approx. 0.5 meters). The system will output noise at minimum and maximum settings to verify the controllable range of 40 dB to 75 dB SPL.

2.7 Tolerance Analysis

The most critical performance feature of Catching Z's is reliable event detection**.** The system must trigger masking noise when a sound event exceeds the ambient baseline by greater than or equal to 10 dB. A 10 dB increase in sound pressure level corresponds to an amplitude ratio given by:

$$\frac{V_{event}}{V_{baseline}} = 10^{(10/20)} = 3.162$$

As such, the measured RMS amplitude of a disturbance must be at least 3.162 times greater than the measured baseline RMS amplitude for detection to occur.

The microphone output is digitized by the ESP32 ADC. using a 12 bit ADC with a 3.3 V reference:

$$LSB = \frac{3.3V}{4096} = 0.805 \text{ mV}$$

The maximum quantization error is approximately ±0.5 LSB:

$$\epsilon = 0.402 \text{ mV}$$

To ensure reliable detection, the measurement error must be small compared to the baseline signal. If we limit the amplitude error to less than 6% (≈0.5 dB), then:

$$\frac{\epsilon}{V_{baseline}} < 0.06$$

Solving for the minimum acceptable baseline RMS voltage:

$$V_{baseline} > \frac{0.402mV}{0.06} = 6.7mV$$

Therefore, the analog front end gain must be set so that the typical ambient baseline produces at least 7 mV RMS at the ADC input. This ensures that noise does not significantly affect detection accuracy. Under these constraints we can ensure that component tolerances in microphone sensitivity will not prevent correct detection of greater than 10 dB disturbances.

3 Cost and Schedule

3.1 Cost Analysis

The total cost for parts as seen below in Figure 7 before shipping is $56.63. With a 5% shipping cost and a 10% sales tax makes the total $65.15. We can expect an hourly wage of $45 for a UIUC ECE grad. This gives us a salary of $45/hr×2.5×42 hours = $4725 per team member. Multiplying this amount by the three team members gives us $14175 in total labor cost. This comes out to be a total cost of $14220.

Parts Costs

Part	**Manufacturer**	**Part #**	**Quantity**	**Cost ($)**
ESP32-S3-WROOM-1	Espressif Systems	ESP32-S3-WROOM-1-N16	1	5.92
MAX4466 Mic Amp	Maxim Integrated	1528-1013-ND	1	6.95
MAX98357A Amp	Maxim Integrated	1528-1696-ND	1	5.95
4Ω 3W Speaker	Adafruit	1528-2435-ND	1	1.95
PEC11R Switch Encoder	Alps Alpine	4809-EC11E15244G1-ND	1	4.49
AMS1117-3.3	UMW	5272-AMS1117-3.3DKR-ND	2	0.54
Capacitor - 10µF / 20% / 10V (0603)	Cal-Chip	2571-GMC10X5R106M10NTTR-ND	4	0.40
Capacitor - 0.1µF 10% / 10V (0402)	YAGEO	C0402C104K8PACTU	3	0.30
Resistor - 330Ω (0603)	YAGEO	RT0603FRE07330RL	2	0.20
Resistor - 5.1kΩ 5%(1/8W) (0805)	Stackpole Electronics	RMCF0805JT5K10	2	0.20
Resistor - 10kΩ (0603)	YAGEO	311-10KGRDKR-ND	2	0.20
Resistor - 2.2kΩ 5%(1/8W) (0805)	Stackpole Electronics	RMCF0805JT2K20	1	0.10
LED - RED LTST-C150CKT (1206)	LITE-ON	LTST-C150CKT	2	0.28
Switch - Tactile	Littelfuse	PTS645SL43SMTR92 LFS	2	0.72
Connector - Micro USB-B	Amphenol	10118194-0001LF	1	0.43
PLA FILAMENT - 1.75MM	Hatchbox	N/A	1	28.00
Total				**56.63**

Figure 7: Itemized list of Components and Cost

3.2 Schedule

Week	Task	Person
February 23rd – February 27th	Submit order form for parts needed for prototyping	Suprathik
	Finalize parts for first PCB iteration and finish schematic	Prineet
	Finish adding custom/external footprints and PCB layout	Srikar
	PCB ORDER February 27th	Everyone
March 2nd – March 6th	Configure ADC sampling and test microphone input	Suprathik
	Design enclosure concept and begin CAD modeling	Srikar
	Implement baseline noise tracking algorithm	Prineet
	PCB ORDER 3/5	Everyone
March 9th – March 13th	Develop digital white noise generation	Suprathik
	Integrate I2S communication with amplifier	Prineet
	Finalize 3D Prints	Srikar
	PCB ORDER 3/12	Everyone
March 23rd – March 27th	Perform SPL output testing for 40–75 dB verification	Suprathik
	Stress test regulator and amplifier under load	Prineet
	Implement fade-in and fade-out for noise playing	Srikar
	4-hour continuous runtime stability test	Everyone
	FINAL PCB ORDER 3/26	Everyone
March 30th – April 3rd	Optimize digital gain scaling and noise smoothness	Srikar
	Finalizing integration from breadboard to PCB	Suprathik
	Work on assembly of 3D prints	Prineet
April 6th – April 10th	Final enclosure adjustments and mechanical assembly	Prineet
	Latency verification measurement	Srikar
	Prepare demo testing scenarios	Suprathik
April 13th – April 17th	Final hardware validation	Everyone
	Complete documentation and verification tables	Everyone
	Final Demo Rehearsal	Everyone

Figure 8: Schedule for Project Progression

4. Ethics and Safety

1. Discuss how your project makes a positive contribution to public health, safety and welfare considering economic, environmental, social, cultural, and global factors.

2. Identify the engineering standards (IEEE, ACM, UL, etc.) that apply to your project.

3. Explain how IEEE/ACM codes of ethics and course ethics guidelines relate to your project. Discuss ethical concerns for your project.

4. Discuss electrical and mechanical safety concerns referring to course safety guidelines page. Add any ethical and safety concerns that arose since your proposal.

5. Document procedures to mitigate the safety concerns of your project. For example, include a lab safety document for batteries, human/animal interfaces, aerial devices, high-power, chemicals, etc. Justify that your design decisions sufficiently protect both users and developers from unsafe conditions caused by your project. Projects dealing with flying vehicles, high voltage, or other high-risk factors, will be required to produce a Safety Manual and demonstrate compliance with the safety manual at the time of demo.

A primary risk is the potential for the system to overreact to brief, harmless sounds (false positives). While we hope to mitigate this, environmental variability remains a challenge. We may require more complex machine learning models to distinguish between disruptive noises (sirens) and non-disruptive spikes (coughing).

The design of Catching Z's follows the ethical principles outlined in the IEEE Code of Ethics and the ACM Code of Ethics. Because the system uses a microphone in a private bedroom environment, privacy is a central concern. All audio processing occurs locally on the ESP32-S3, and the device does not record, store, or transmit audio data. No wireless communication is enabled in the baseline design, reducing the risk of misuse or data breaches.

The system operates at low voltage, minimizing electrical hazards. Proper grounding and insulation will reduce shock and overheating risks. The maximum sound output is limited in firmware to 75 dB at 0.5 meters to prevent hearing damage, and masking noise fades in and out gradually to avoid startling the user.

Engineers are responsible for accurate performance reporting and avoiding exaggerated health claims. While the device may help mitigate sleep disruption from environmental noise, it is not a substitute for broader noise control solutions. Safety and privacy are at the core of this project, and we as engineers aim to meet professional engineering and societal standards.

References

[1] B. Berglund, T. Lindvall, and D. H. Schwela, "Guidelines for Community Noise," World Health Organization, Geneva, Switzerland, 1999. [Online]. Available: https://apps.who.int/iris/handle/10665/66217

[2] *ESP32-S3-WROOM-1 / ESP32-S3-WROOM-1U Datasheet*, v1.1, Espressif Systems, Shanghai, China, 2023. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-s3-wroom-1_wroom-1u_datasheet_en.pdf

[3] *MAX4465–MAX4469: Low-Cost, Micropower, SC70/SOT23-8, Microphone Preamplifiers with Complete Shutdown*, Maxim Integrated, San Jose, CA, USA, 2001. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/MAX4465-MAX4469.pdf

[4] *MAX98357A/MAX98357B: Tiny, Low-Cost, PCM Class D Amplifier with Class AB Performance*, Maxim Integrated, San Jose, CA, USA, 2015. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/MAX98357A-MAX98357B.pdf

[5] *PEC11R Series - Contacting Encoder*, Bourns, Inc., Riverside, CA, USA, 2015. [Online]. Available: https://www.bourns.com/docs/Product-Datasheets/PEC11R.pdf

[6] *AMS1117: 1A Low Dropout Voltage Regulator*, Advanced Monolithic Systems, Livermore, CA, USA, 2016.