INTERACTIVE DESKTOP COMPANION ROBOT FOR STRESS RELIEF

By

Jiajun Gao (jiajung3)

Yuchen Shih (ycshih2)

Zichao Wang (zichao3)

Project proposal for ECE 445, Senior Design, Spring 2026

TA: Haocheng Yang

February 12th 2026

Team No. 4

Contents

1. Introduction

1.1 Problem

Prolonged desk-based work has become the dominant mode of productivity in modern academic and professional environments. According to the American Institute of Stress, over 80% of U.S. workers report experiencing work-related stress, which is strongly associated with reduced cognitive performance, burnout, and long-term physical and psychological health risks. The widespread adoption of remote and hybrid work models has further intensified social isolation and reduced opportunities for spontaneous interpersonal interaction.

While numerous digital wellness applications attempt to mitigate stress through mindfulness prompts or guided exercises, these solutions are inherently screen-dependent and often increase overall screen exposure. Prolonged screen engagement has been linked to eye strain, reduced physical activity, and decreased attention span. In contrast, passive desk objects such as toys or decorative items lack interactivity and fail to adapt to user state or context.

From an engineering perspective, there is a gap between passive physical artifacts and fully immersive robotic systems. Existing consumer robots often target entertainment or domestic assistance, but few systems are specifically designed as lightweight, non-intrusive, stress-relief companions optimized for desktop environments. Therefore, there is a need for an embedded, autonomous system that integrates real-time sensing, multimodal feedback, and conversational interaction into a compact platform capable of delivering low-effort, intermittent engagement without disrupting productivity.

1.2 Solution

To address this problem, we propose an interactive desktop companion robot that provides short and lightweight interactions to help users relax during desk work. The goal of this system is not to replace productivity tools, but to introduce brief moments of engagement that reduce stress without causing major interruptions.

The robot is built around an ESP32-S3 microcontroller, which handles audio input, wireless communication, visual display updates, and motor control. An onboard I2S microphone captures user speech, and a lightweight wake-word detection algorithm runs locally to avoid continuously sending audio to the cloud. Once activated, the recorded audio is transmitted over Wi-Fi to a cloud-based Large Language Model (LLM) server for speech recognition and response generation.

To improve responsiveness, the system uses a streaming Text-to-Speech (TTS) approach. Instead of waiting for the entire response to be generated, the robot begins playing audio as soon as the first portion of the synthesized speech is received. This reduces perceived delay and makes the interaction feel more natural.

The robot also includes a small SPI LCD screen that displays animated facial expressions to reflect different states, such as listening, processing, or speaking. A differential-drive motor system allows limited autonomous motion, while Time-of-Flight (ToF) sensors detect desk edges and nearby obstacles to ensure safe operation. By combining audio interaction, visual feedback, and simple motion behaviors, the robot provides a physical and engaging alternative to purely screen-based stress-relief applications.

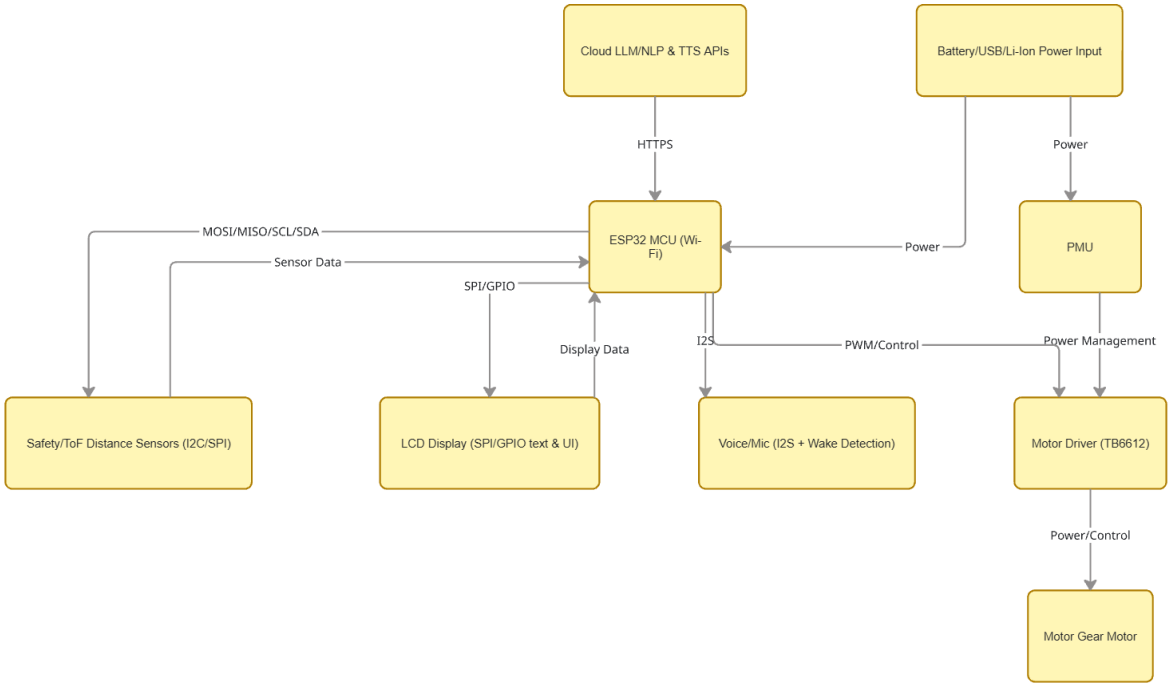
1.3 Visual aid

1.4 High-level Requirements List

- Interaction Latency: The average end-to-end voice interaction latency (from end of user speech to start of robot audio playback) must be less than 5 s under standard network conditions (RSSI > −60 dBm).
- Safety and Reliability: During a 10-minute continuous motion test on a standard desk, the robot must achieve a 100 percent success rate in cliff detection, meaning it must never fall off the desk.
- Autonomous Performance: In obstacle trials, the robot must stop or turn before contacting a stationary object in at least 18 out of 20 tests (90 percent accuracy).

2 Design

2.1 Block Diagram



2.2 Subsystem Overview

- Voice Interaction and Audio Processing: This subsystem manages bidirectional audio communication between the user and the robot. An I2S digital microphone continuously samples ambient sound and feeds raw audio data to the ESP32-S3. A lightweight on-device wake-word engine runs locally to minimize unnecessary cloud traffic and preserve user privacy. Once activated, audio frames are encoded and transmitted over Wi-Fi to the cloud-based LLM server for speech recognition and response generation. Returned Text-to-Speech (TTS) audio is streamed back in small packets and decoded in real time, allowing playback to begin before the full response is received. This streaming approach reduces perceived latency and enables more natural conversational timing.
- Visual Expression and User Feedback: This subsystem provides visual communication through an ST7789 SPI LCD panel. The display renders 16-bit RGB animations that represent internal robot states, including listening, processing, speaking, and idle modes. Simple animated facial expressions and icons are used to convey feedback such as

successful wake-word detection, network activity, or error conditions. Display updates are handled asynchronously to avoid blocking audio or motion tasks. By presenting clear visual cues, this subsystem improves usability and helps users understand system behavior without relying solely on audio output.

- Motion and Actuation: The motion subsystem converts logic-level commands from the ESP32-S3 into mechanical movement using a TB6612FNG dual H-bridge motor driver. PWM signals control motor speed, while GPIO direction pins determine forward or reverse motion for each wheel, enabling differential steering. This configuration supports basic navigation behaviors such as forward motion, in-place turning, and obstacle avoidance maneuvers. Motor commands are continuously monitored by the safety sensing subsystem, which can override drive signals when hazards are detected. This design allows responsive movement while maintaining safety constraints.
- Power Management and Safety: This subsystem distributes energy from a 3.7 V lithium-ion battery to all components while maintaining electrical stability. A step-up/step-down regulation stage provides 5 V for motors and the audio amplifier, while a low-dropout regulator supplies a clean 3.3 V rail for the ESP32-S3 and sensors. Over-current and under-voltage protection circuits prevent battery damage and reduce risk during motor stall or communication bursts. The system also supports brownout detection, allowing the microcontroller to safely shut down motion outputs if supply voltage drops below operational thresholds.
- Safety Sensing: This high-priority subsystem uses multiple Time-of-Flight sensors connected over I2C to detect desk edges and nearby obstacles. Sensors are sampled at high frequency to provide real-time distance measurements in front of and beneath the robot. When a sudden increase in downward distance indicates a desk edge, or when a forward obstacle is detected within a predefined threshold, the subsystem immediately disables motor outputs and commands a stop or turn maneuver. This logic operates independently of higher-level navigation to ensure rapid response. By combining cliff detection with obstacle awareness, the robot maintains stable operation on desk surfaces and avoids unintended falls or collisions.

2.3 Subsystem Requirements

2.3.1 Safety sensing (Desk-Edge Detection)

To ensure safe operation and prevent falls from elevated surfaces, the Time-of-Flight (ToF) sensors used for desk-edge detection must be sampled at a minimum frequency of 50 Hz. This sampling rate establishes the required safety threshold for real-time hazard detection. Given a maximum robot speed of 10 cm/s, a 50 Hz sampling frequency corresponds to a sensing interval of 20 ms. Under this condition, the robot travels no more than 2 mm between consecutive measurements. This constraint ensures that a sudden increase in downward distance is detected with sufficient temporal margin, allowing the control system to disable motor outputs and engage braking before the chassis reaches the desk edge.

2.3.2 Audio Processing

For the audio processing subsystem, the I2S microphone array must maintain a Signal-to-Noise Ratio (SNR) greater than 60 dB. This requirement ensures reliable wake-word detection in typical office environments, where ambient noise levels generally range between 40 and 50 dB. Maintaining this SNR threshold improves the system's ability to distinguish intentional user input from background noise, thereby reducing false-positive activations caused by surrounding conversations, HVAC systems, or other environmental disturbances.

2.4 Tolerance Analysis

One of the primary engineering challenges in this robotic system is managing the inherent delay introduced by relying on cloud-based natural language processing. To ensure a natural user experience, the total interaction latency, denoted as L, is mathematically modeled as the sum of its sequential processing components:

$$L = t_{\{record\}} + t_{\{upload\}} + t_{\{LLM\}} + t_{\{TTS\}} + t_{\{download\}}$$

In this baseline model, the individual timing components are estimated as follows:

- **Recording and Silence Detection ($\mathbf{t_{record}}$)**Capturing the user's spoken audio and accurately detecting the end of speech (silence detection) consumes approximately **0.5 seconds**.

- **Large Language Model Inference (t_{LLM}):** The core cloud-based processing time required to generate an intelligent text response takes roughly **1.5 seconds**.

- **Network Transmission (t_{upload} and $t_{download}$):** The combined network round-trip time for uploading the compressed audio payload to the server and downloading the resulting data adds approximately **1.0 second**.

- **Text-to-Speech Synthesis (t_{TTS}):** The time required for the server to convert the LLM's text output into a natural-sounding audio file.

To ensure a fluid user experience and strictly adhere to the system requirement of keeping total latency under 5 seconds significant optimizations in audio buffering and data pipeline handling are mandatory.

Specifically, the system must transition from a sequential processing model to a streamed Text-to-Speech (TTS) architecture. By configuring the server to stream playback audio chunks as soon as they are synthesized—rather than waiting for the entire text response to be generated and converted—we can effectively execute the TTS processing and the download transmission in parallel. This strategic overlapping of operations eliminates sequential bottlenecks and reduces the total perceived interaction latency by approximately **1.2 seconds** in practical application.

3 Ethics, Safety, and Societal Impact

3.1 Ethical Considerations

In strict alignment with the IEEE Code of Ethics, safeguarding user privacy and ensuring robust data security are foundational principles of this robot's design. This is particularly critical given the continuous operation of the system's onboard microphone array. To effectively mitigate privacy risks, the audio processing architecture enforces a rigid local-first policy: the system guarantees that absolutely no audio data is recorded, buffered in long-term memory, or transmitted to external servers until the integrated, edge-based wake-word detection algorithm is explicitly triggered.

Once a user intentionally initiates an interaction, all subsequent data pipelines bridging the device and the cloud—including audio payload uploads and generated text downloads—are secured end-to-end using industry-standard TLS/HTTPS encryption protocols to prevent unauthorized interception. Furthermore, the system architecture mandates a stateless local environment regarding user data; no personally identifiable information (PII), voiceprints, or conversational histories are ever saved to the robot's persistent onboard storage.

3.2 Safety and Regulatory Standards

Deploying a mobile robotic system in a human-centric environment necessitates rigorous adherence to established safety standards, particularly concerning power management and autonomous motion control. To address battery safety, the device's power distribution network strictly complies with UL 1642 guidelines for the safe operation of lithium-ion cells. This compliance is physically realized through the integration of a dedicated hardware Protection Circuit Module (PCM).

The PCM continuously monitors cell voltage and current, actively preventing hazardous conditions such as overcharging, deep discharging, external short circuits, and thermal runaway.

In addition to electrical safety, the motion control subsystem is engineered with a deterministic "fail-safe" architecture to protect both the device and its surroundings. The main microcontroller continuously polls the health of critical subsystems. If the safety sensors (such as the ToF edge detectors) report an error state, or if the robot detects a complete loss of its Wi-Fi control link, the

firmware immediately forces all motor Pulse Width Modulation (PWM) control signals to zero. This instantaneous, system-level override securely halts the chassis, effectively eliminating the risk of erratic, autonomous, or unintended physical motion during a fault condition.

References

[1] Aalund, R., & Pecht, M. (2019). The Use of UL 1642 Impact Testing for Li-ion Pouch Cells. *IEEE Access*, *7*, 176706–176711. https://doi.org/10.1109/access.2019.2957814

[2] Gülırmak, E. A., & Bingül, Z. (2023). Determining Robot Trajectory Planning Using Image Processing for Wood Painting. *2023 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*. https://doi.org/10.1109/ismsit58785.2023.10304909

[3] Lai, X., Yao, J., Jin, C., et al. (2022). A Review of Lithium-Ion Battery Failure Hazards: Test Standards, Accident Analysis, and Safety Suggestions. *Batteries*, *8*, 248. https://doi.org/10.3390/batteries8110248

[4] Lakhnati, Y., Pascher, M., & Gerken, J. (2024). Exploring a GPT-based large language model for variable autonomy in a VR-based human-robot teaming simulation. *Frontiers in Robotics and AI*, *11*. https://doi.org/10.3389/frobt.2024.1347538

[5] Nakamura, K., Nakadai, K., & Okuno, H. G. (2013). A real-time super-resolution robot audition system that improves the robustness of simultaneous speech recognition. *Advanced Robotics*, *27*, 933–945. https://doi.org/10.1080/01691864.2013.797139

[6] Pollini, D., Guterres, B. V., Guerra, R. S., & Grando, R. B. (2025). Reducing Latency in LLM-Based Natural Language Commands Processing for Robot Navigation. *arXiv*. https://doi.org/10.48550/arxiv.2506.00075

[7] MCP chatbot: https://github.com/78/xiaozhi-esp32.git

[8] Example: https://living.ai/emo/