SEDENTARY DETECTION CHAIR WITH WEARABLE IMUS

By

Chris Huang (zexih2)

Melissa Wang (wang569)

Jack Gaw (jgaw3)

Design Document for ECE 445, Senior Design, Spring 2026

TA: Weijie Liang

SP February 2026

Project No.95

Abstract

Prolonged sedentary behavior is common among students and office workers and is associated with negative health effects. This project designs a chair-based system that detects extended sitting and encourages verified physical activity before resuming sitting. Chair-mounted load cell sensors determine whether a user is seated, while two wearable IMU nodes attached to the ankle and waist/thigh measure body motion and transmit synchronized motion packets to a chair-mounted ESP32 via Bluetooth Low Energy (BLE). The ESP32 fuses chair and wearable data to trigger an alarm after a configurable sitting interval and clears the alert only after sufficient movement is detected.

Contents

1. Introduction 1
1.1 Problem 1
1.2 Solution 2
1.3 Visual Aid 4
1.4 High-level Requirements 4
2. Design 6
2.1 Block Diagram 6
2.2 Subsystem Overview 6
2.2.1 Seat & Chair Sensing Subsystem 6
2.2.2 Control + Processing 7
2.2.3 Wearable IMU Nodes 9
2.2.4 Alarm + UI 11
2.2.5 Power 12
2.3 Software Design 12
2.3.1 Pressure / Load Sensing Module 13
2.3.2 UI Button Control Module 14
2.3.3 Wearable Device Module 15
2.3.4 Bluetooth Connection Module 16
2.4 Tolerance Analysis 17
3. Design Verification 21
3.1 Chair Sensor Verification 21
3.1.1 Occupancy Detection Timing and Accuracy 21
3.2 BLE Communication Verification 22
3.3 Movement Verification Using Two IMUs 24
4. Ethics, Safety and Societal impact 25
5. Costs 27
5.1 Parts 27
5.2 Schedule 30

6. Conclusion 34
6.1 Accomplishments 34
6.2 Uncertainties 34
6.3 Ethical considerations 35
6.4 Future work 35
References 37

1. Introduction

1.1 Problem

Prolonged sedentary behavior is common in study and office settings, where users can remain seated for long, uninterrupted intervals while focusing on work. Although many people understand that frequent movement is beneficial, it is easy to lose track of time when seated, and the cost of “just sitting a bit longer” is not immediately obvious. Over days and weeks, this leads to excessive sedentary time and fewer meaningful activity breaks, especially during high-focus tasks such as programming, studying, gaming, or desk work.

Most existing solutions are reminder-based (phone apps, desktop pop-ups, smartwatch nudges). These approaches have two core limitations. First, they depend on voluntary compliance: the reminder can be ignored, snoozed, or dismissed with minimal friction, so the system does not reliably change behavior when the user is busy or highly engaged. Second, reminders do not verify behavior. A user can acknowledge the notification without actually moving, or can perform a very low-effort action (briefly standing, shifting posture, shaking a wearable) that does not represent a meaningful activity break. As a result, the intervention fails to guarantee that the user actually reduced sedentary time in a measurable way. This motivates a system that can both detect continuous sitting intervals accurately and require objective evidence of real movement before returning to a normal “seated” state.

1.2 Solution

We propose a smart anti-sedentary chair system that combines chair-based seat-occupancy detection with wearable inertial sensing to enforce verified movement breaks. The chair provides a reliable reference for whether the user is seated, while the wearables provide objective confirmation that the user performed a genuine activity break. The central design goal is to transform "reminders" into an enforceable feedback loop: when the system detects excessive continuous sitting, it triggers an alert that cannot be cleared until the user completes sufficient movement.

At the chair level, a seat-occupancy sensor (e.g., a force/pressure sensor such as an FSR or a load-cell-based measurement chain) continuously monitors whether the user is sitting. A microcontroller (ESP32) timestamps seated intervals and implements a configurable timer threshold (for example, a maximum continuous sitting duration). When the seated timer exceeds the threshold, the system enters an "alert" state and triggers user feedback (e.g., buzzer and/or LED). This makes the user immediately aware that a break is required and removes the need to self-monitor time.

To ensure that the break is meaningful rather than superficial, the system uses two wearable IMU nodes placed at distinct body locations (e.g., ankle and waist/thigh). Each node measures acceleration and angular velocity and transmits motion data to the chair-side ESP32 over Bluetooth Low Energy (BLE). The ESP32 extracts simple, computationally efficient motion features over a short activity window (e.g., step-like periodicity, acceleration magnitude statistics, or movement counts) and applies a threshold-based decision rule to determine whether the user performed sufficient activity. The two-sensor placement is intentional: motion at a single location can be spoofed or may miss important activity patterns, but combining lower-body and torso/upper-leg motion improves discriminability between real walking-like movement and small local motions (e.g., shaking one device).

System behavior is governed by a state machine. In the normal state, the system tracks seated duration when occupancy is detected. When the duration exceeds the limit, the system raises an alert and transitions into a "break required" state. The alert remains active until two conditions are met: (1) the seat-occupancy sensor indicates the user is no longer seated, and (2) the wearable IMU data indicates that the user completed a minimum amount of movement consistent with a real break. Only then does the system clear the alert and reset the seated timer. By requiring both "not seated" and "movement verified," the design directly addresses the two weaknesses of reminder-only systems: it reduces dismiss-ability and adds objective validation that the break occurred.

1.3 Visual Aid

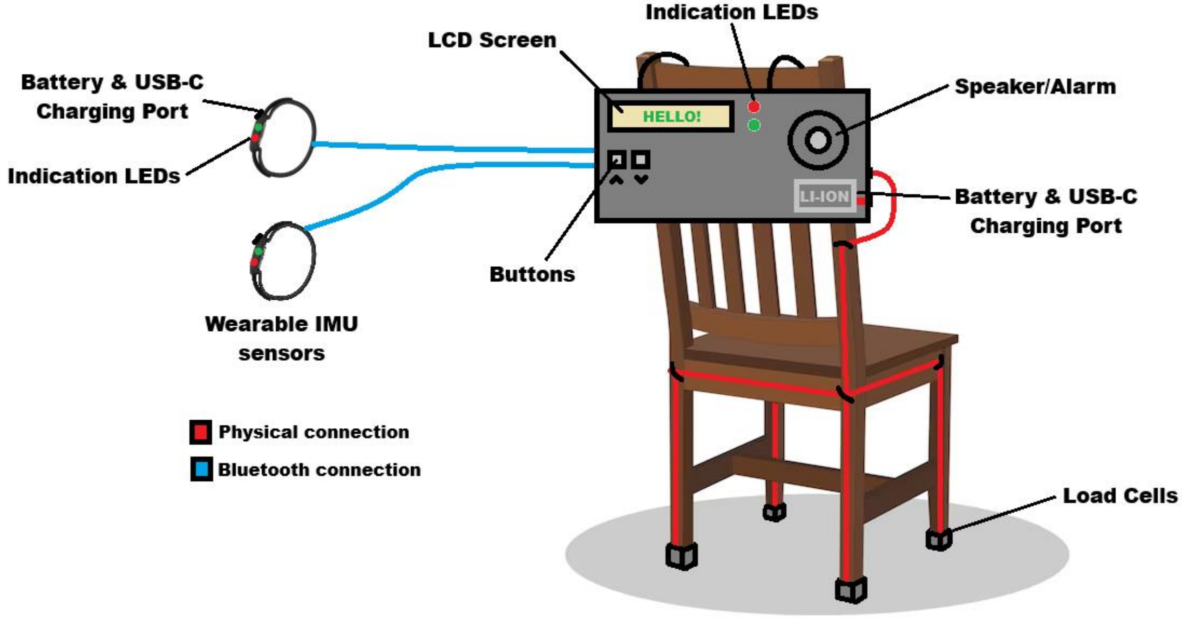

Figure 1. Visual model for Chair-mount system and wearable device

1.4 High-level Requirements

1. Reliable Seat Occupancy Detection (Chair-Leg Pressure Sensing)

   The system shall determine whether the user is seated using pressure/load sensors located under the chair legs. It shall detect true sit→stand and stand→sit transitions within ≤ 1.0 s, and shall not change state due to brief fluctuations shorter than 1–2 s (debounce/hysteresis). Across 50 sit/stand trials, the system shall achieve ≥ 90% correct transition detections within the timing bound.

2. Verified Activity Using Dual IMUs (Anti-Spoof Movement Confirmation)

   The system shall verify that the user performed meaningful movement using two wearable IMUs (e.g., ankle + waist/thigh) and shall require both IMUs to satisfy the activity criteria before a break is accepted. For continuous walking-like motion, estimated activity time shall have ≤ 10% error relative to a stopwatch ground truth. For "spoof" attempts (e.g., shaking a device while not walking), the system shall incorrectly accept the activity in ≤ 1 out of 20 trials (≤ 5% false-clear rate).

3. Alarm Enforcement Based on Verified Break Completion

   When continuous sitting time exceeds a configurable threshold, the system shall enter an alarm state (buzzer and/or LEDs). The alarm shall be cleared only if both conditions are met: (a) chair-leg sensors indicate the user is not seated, and (b) dual-IMU verification indicates sufficient activity. In 20 bypass attempts (e.g., pressing buttons to silence the alert without moving), the byp ass success rate shall be 0%.

2. Design

2.1 Block Diagram

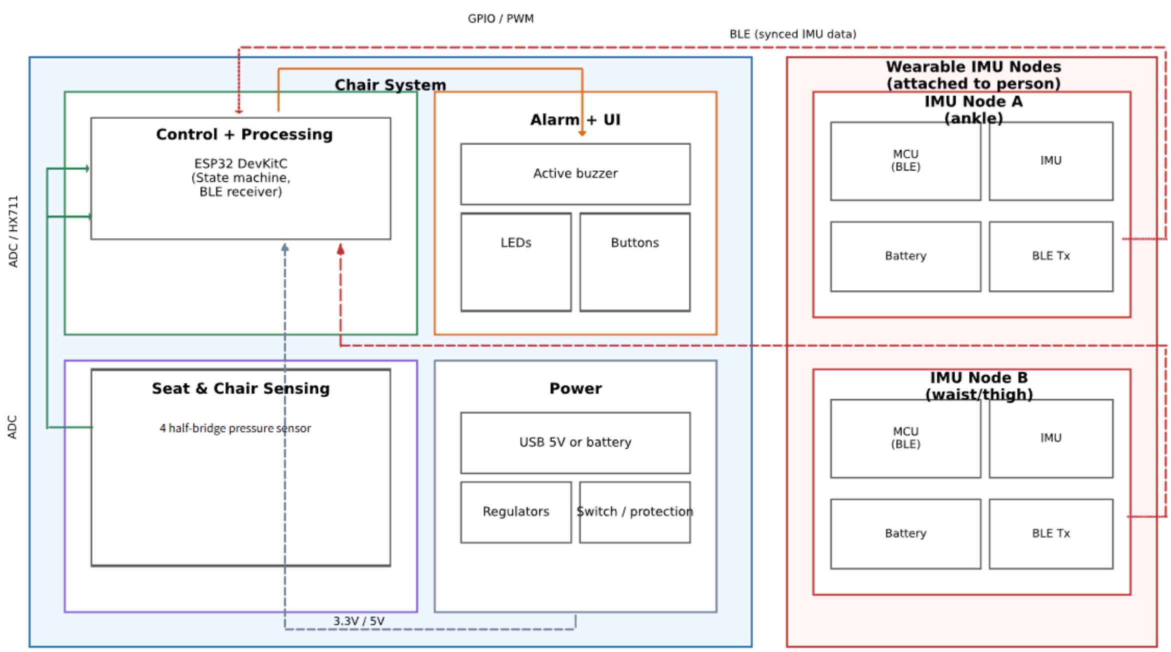

Figure 2. Block diagram

2.2 Subsystem Overview

2.2.1 Seat & Chair Sensing Subsystem

The chair-mounted sensing subsystem is responsible for determining whether a user is seated and for reducing false detections caused by small chair movements. This subsystem is divided into four primary subsystems: (1) the seat & chair sensing subsystem, which detects whether the user is seated using load/pressure sensing at the chair legs; (2) the control and processing subsystem, implemented on a chair-mounted ESP32 that runs the state machine and handles sensor inputs; (3) two wearable IMU nodes that provide objective movement verification over Bluetooth Low Energy (BLE); and (4) the alarm and user interface subsystem that delivers audible/visual feedback and supports limited user interaction. A separate power subsystem provides regulated rails for the chair unit and battery-powered operation for the wearables. Together, these subsystems implement a closed-loop anti-sedentary workflow: detect continuous sitting, trigger an alert after a threshold, and only clear the alert when the user stands and completes verified activity.

2.2.2 Control + Processing

The chair-mounted sensing subsystem is responsible for determining whether a user is seated and for minimizing false detections caused by sensor noise or small shifts in posture. This subsystem provides the primary indicator of sedentary behavior and serves as the trigger for timing prolonged sitting periods.

Seat occupancy is detected using pressure/load sensors located under the chair legs (e.g., load cells mounted under chair supports, read through an HX711 amplifier). When a user sits, the load distribution across the legs increases in a measurable way; when the user stands, the measured load drops toward an unoccupied baseline. The ESP32 samples the digitized load data and computes an occupancy decision using thresholding with hysteresis. At initialization (or via a calibration routine), the system records an "unoccupied" baseline and sets adaptive thresholds to account for different chair weights and floor conditions. Hysteresis and time-based debounce are applied so that brief transients (e.g., the user shifting weight, lightly touching the chair, or partially perching) do not cause rapid toggling between occupied and unoccupied states. The output of this subsystem is a stable binary occupancy signal and optional confidence metrics (e.g., averaged load and variance) that the controller uses for state transitions.

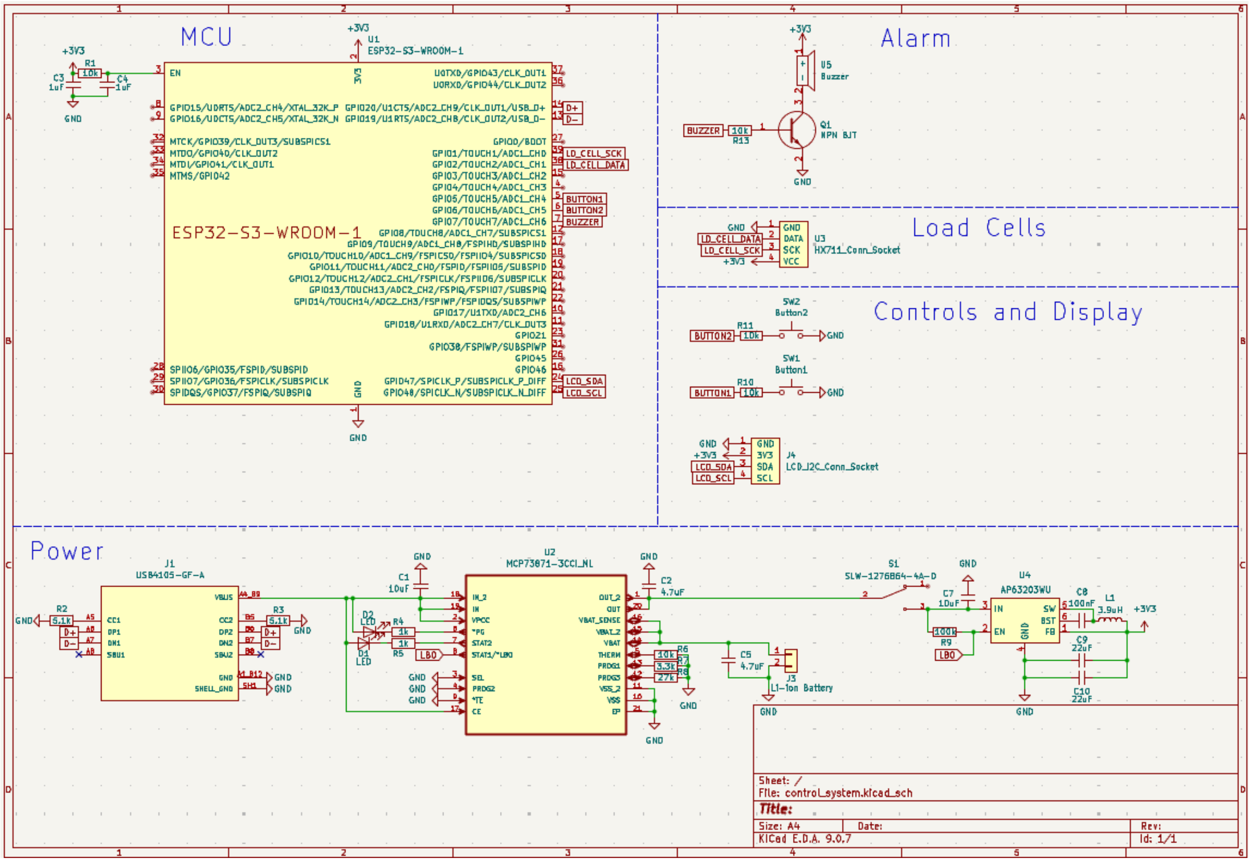

Figure 3. Schematics for Chair-mount control system

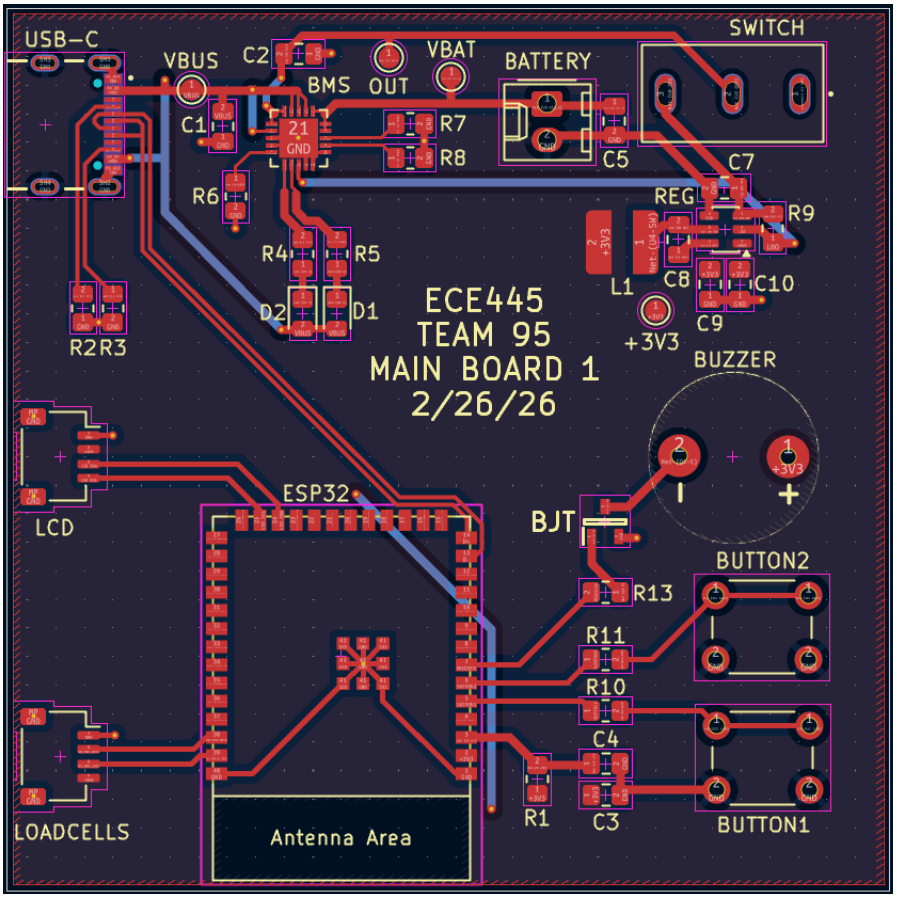

Figure 4. PCB design for Chair-mount control system

2.2.3 Wearable IMU Nodes

The control and processing subsystem is implemented using a chair-mounted ESP32 microcontroller. The ESP32 runs the main system state machine, tracks the duration of continuous sitting, and receives motion data from the wearable IMU nodes over Bluetooth Low Energy (BLE). Chair sensor data (occupancy state and load trends) and wearable motion features are combined to decide when an alarm should be triggered and when it can be safely cleared.

At a high level, the ESP32 maintains system states such as IDLE/UNOCCUPIED, SEATED/TIMING, ALARM_ACTIVE, and BREAK_VERIFICATION. While seated, the ESP32 increments a timer; once the timer exceeds a configurable limit, the ESP32 asserts an alarm condition and activates the user feedback subsystem. To clear the alarm, the ESP32 requires two conditions: (1) the chair-leg sensors indicate the user is no longer seated, and (2) the wearable IMUs report sufficient activity consistent with a real movement break. The ESP32 also handles BLE connection management, packet reception, timestamping, and timeout logic to ensure the system remains safe and deterministic under transient wireless loss.

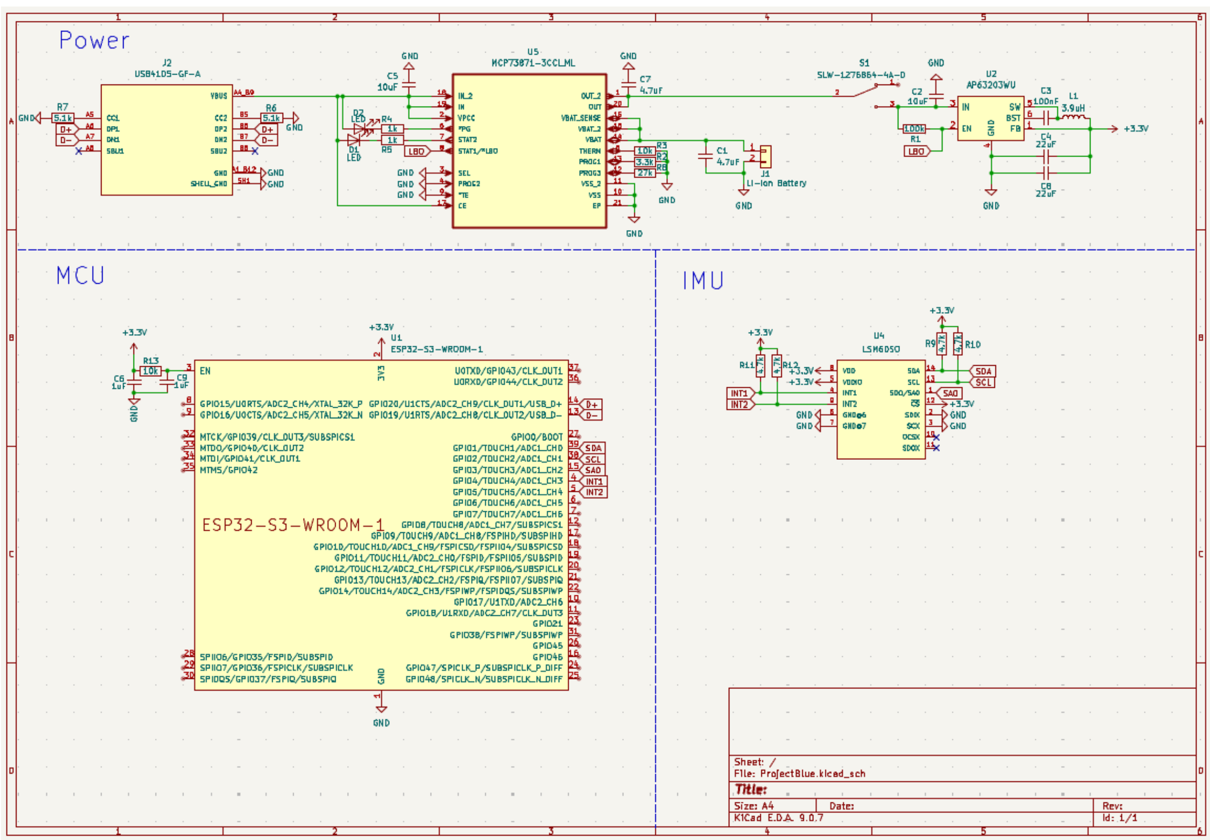

Figure 5. Schematics for Wearable devices

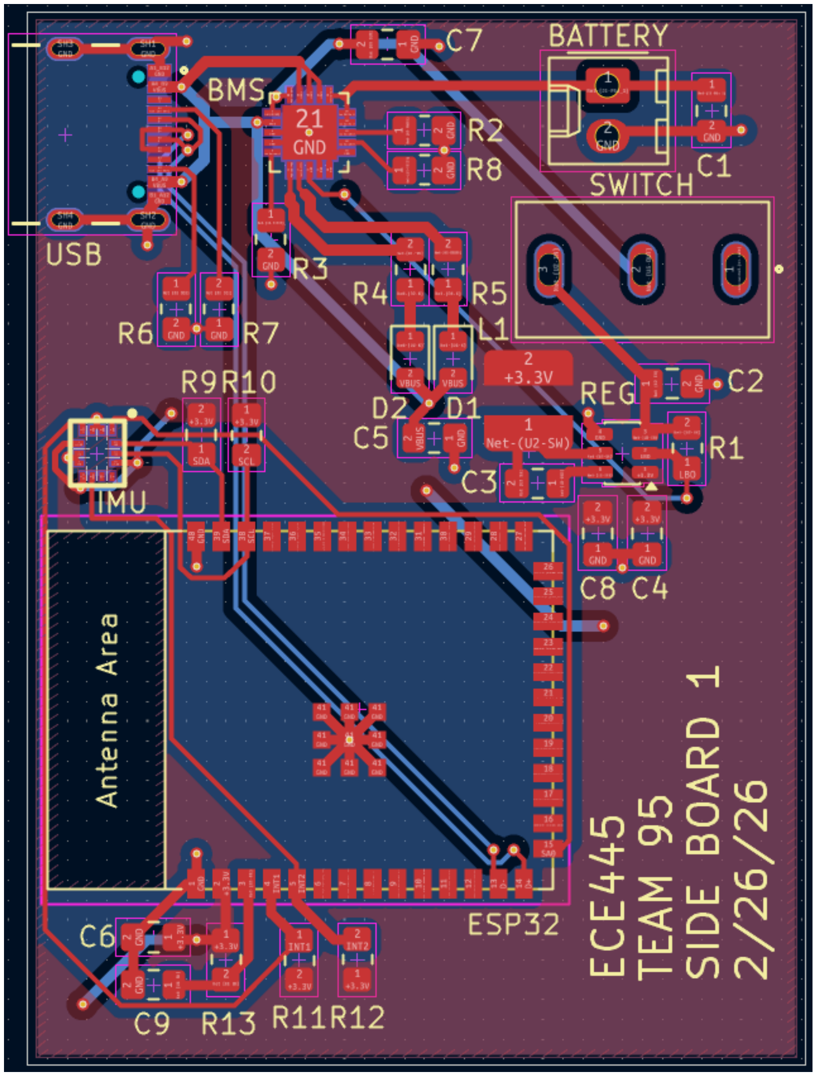

Figure 5. PCB design for Wearable devices

2.2.4 Alarm + UI

User feedback is provided through an audible alarm (active buzzer) and visual indicators (status LEDs) mounted on the chair. When prolonged sitting is detected, the buzzer produces a clear audible alert and LEDs indicate the system state (e.g., normal/seated, alarm active, break verified). Optional push buttons provide limited user interaction, such as acknowledging an alarm, initiating calibration, or resetting the system after a completed break (exact button behavior is defined by the state machine to prevent bypassing verification).

These components are controlled directly by the chair-mounted ESP32 using GPIO (for LEDs and buttons) and PWM or digital drive (for the buzzer depending on the selected buzzer type). UI behavior is designed to be simple and unambiguous: the user can always tell whether the system believes they are seated, whether an alarm is active, and whether a verified break has been achieved.

2.2.5 Power

The chair-mounted electronics can be continuously powered by a 5 V USB supply (e.g., wall adapter or power bank) or can be disconnected and remain powered by the onboard rechargeable Li-ion battery. On-board regulation provides stable 3.3 V for the ESP32 and any associated digital/analog interfaces (e.g., HX711). Decoupling and grounding practices are used to reduce susceptibility to noise from the buzzer and to improve sensor stability.

Each wearable IMU node is powered by its own rechargeable Li-ion battery to enable untethered operation during movement verification. The wearable power chain includes charging and protection circuitry (overcharge/overdischarge protection), and regulation to the IMU/MCU operating voltage. Battery capacity is selected to support the intended use duration (e.g., multiple hours of intermittent activity monitoring), and the firmware can incorporate low-power modes to extend runtime when continuous high-rate sampling is not required.

2.3 Software Design

The software is organized into modular components on the chair-mounted ESP32 (central controller) and two wearable IMU devices (peripherals). Each module has a clear input/output contract so the overall system remains robust under sensor noise, user behavior variability, and intermittent wireless conditions. The primary software modules are: (1) pressure/load sensing under chair legs, (2) UI button control for alarm timing configuration, (3) wearable device firmware (IMU sampling + packetization), and (4) Bluetooth Low Energy (BLE) connection and data handling. In addition, two small supporting modules—alarm/LED output control and a system state machine/timer—are required to integrate the above components into a deterministic, non-bypassable workflow.

2.3.1 Pressure / Load Sensing Module

This module determines the seat occupancy state using pressure/load sensors mounted under the chair legs (e.g., load cells read through HX711). The ESP32 periodically samples sensor readings and applies signal conditioning to produce a stable OCCUPIED/UNOCCUPIED output. Software filtering (moving average or low-pass filtering) reduces measurement noise and short spikes. The occupancy decision is implemented with thresholding + hysteresis to prevent rapid toggling when readings hover near the boundary. A debounce timer is applied so that a transition is only accepted if the new condition persists for a minimum dwell time (e.g., 1–2 seconds).

A calibration routine records an unoccupied baseline at startup or upon user request (e.g., chair empty for several seconds). Thresholds are set relative to this baseline to account for differences in chair weight, floor surface, and sensor mounting. If multiple leg sensors are used, readings are aggregated (sum or weighted sum) to reduce sensitivity to posture-dependent load distribution. The output of this module is (1) the binary occupancy state and (2) optional diagnostics (filtered load value, variance, and stability flags) that can be logged for verification.

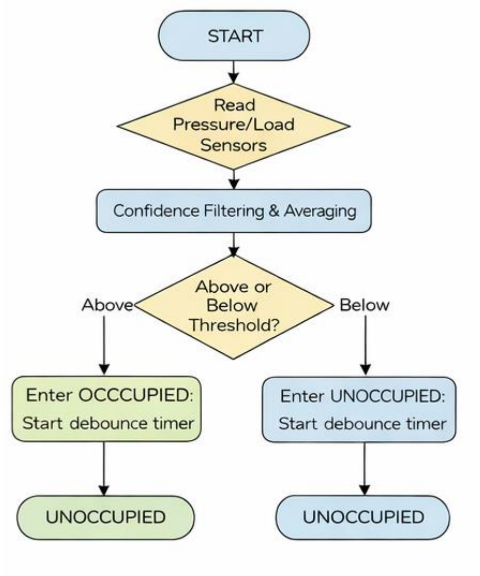

Figure 6. Seat Occupancy Flowchart

2.3.2 UI Button Control Module

Two physical buttons allow the user to configure how long continuous sitting is allowed before an alarm triggers. The button module is responsible for debouncing, interpreting presses into actions, and safely updating configuration parameters without breaking the enforcement logic.

Each button input is sampled with a debounce method (either time-based debounce in software or interrupt + debounce window). Button events are classified as short press and optionally long press (if implemented) to expand the number of available controls without adding hardware. A typical mapping is:

- Button A: decrease sitting-time threshold (e.g., −5 minutes per press)
- Button B: increase sitting-time threshold (e.g., +5 minutes per press)
- (Optional) Long press on both: reset to default threshold or trigger calibration (only when chair is unoccupied)

To prevent bypass behavior, button actions do not directly silence or clear the alarm. Instead, button events update only the configuration variable (the sitting threshold) and are routed through the system state machine. If the system is already in ALARM state, threshold changes may be temporarily locked out or applied only after the next reset, depending on the chosen policy. The module also updates LED indicators to reflect the configured threshold (e.g., discrete levels) if desired.

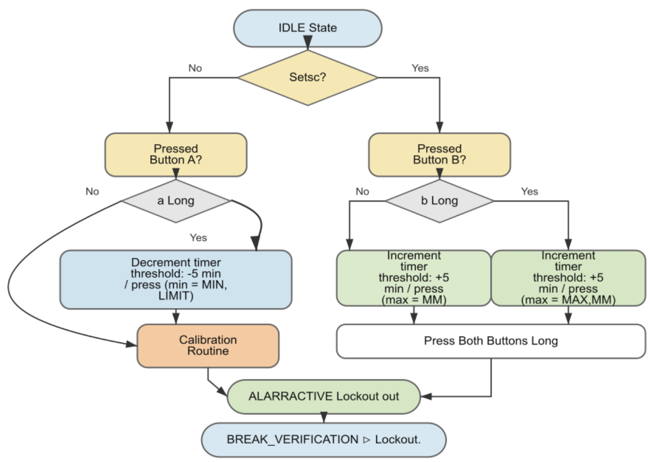

Figure 6. UI control Flowchart

2.3.3 Wearable Device Module

Each wearable device runs firmware that performs consistent IMU sampling and sends motion data to the chair ESP32. The wearable sampling loop reads accelerometer and gyroscope data at a fixed rate (e.g., 50–100 Hz) and either:

- transmits raw 3-axis accel/gyro values, or
- computes lightweight derived values such as acceleration magnitude before transmission (to reduce bandwidth and improve orientation robustness).

Samples are packaged into BLE notification payloads including a sequence number (for packet-loss detection) and a timestamp or sample counter (for timing reconstruction). The wearable firmware is designed to be deterministic: stable sampling intervals, consistent packet sizes, and predictable transmission rate. If battery operation is used, the wearable can support a low-power mode when not connected and resume full-rate sampling when a BLE link is established.

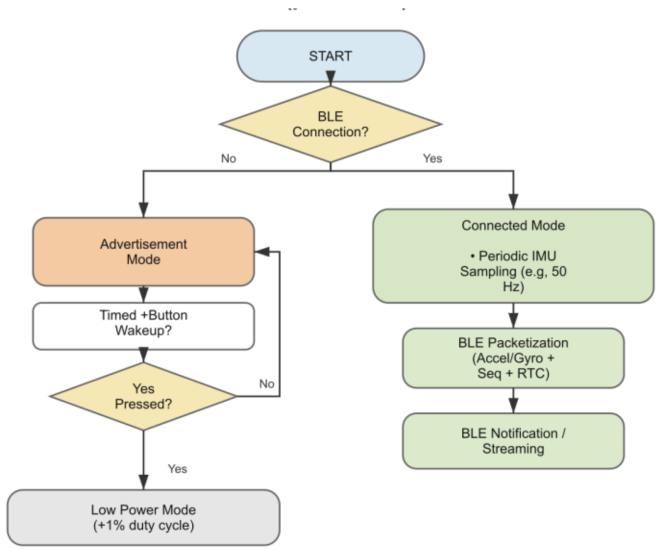

Figure 7. Wearable Device Flowchart

2.3.4 Bluetooth Connection Module

The chair-mounted ESP32 acts as the BLE central and manages connections to both wearable devices simultaneously. The BLE module is responsible for: scanning, connecting, service/characteristic discovery, subscribing to notifications, and maintaining link health. Data integrity is handled through sequence numbers, allowing the chair to compute packet reception rate and detect gaps.

Incoming BLE packets are queued and processed in a controlled manner so that brief bursts do not block time-critical tasks like occupancy detection or alarm driving. The system applies a staleness timeout: if a wearable stops transmitting or disconnects, motion verification is considered invalid and the alarm cannot be cleared based on missing data. The module also handles reconnect behavior (automatic rescan and reconnect) and provides status to the UI layer (e.g., LED pattern indicating one or both wearables disconnected).

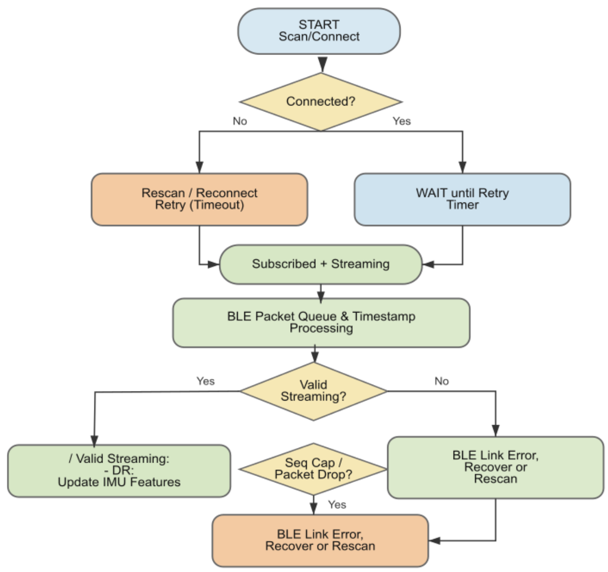

Figure 8. Bluetooth Connection Flowchart

2.4 Tolerance Analysis

A key technical risk for successful completion is ensuring that real-world sensor and system non-idealities do not cause frequent false triggers (false “seated” / false “not seated”) or allow false alarm clearing. Since our design no longer uses piezoelectric vibration sensors, the dominant tolerance concerns shift to (1) chair-leg load sensing variability, (2) wearable IMU measurement and placement variability, and (3) power, ADC/digital-interface margins that can degrade measurement quality and robustness.

Chair-leg load sensing tolerances.

Using load/pressure sensing under chair legs introduces variability due to user weight differences, posture and weight distribution changes, and mechanical mounting differences. Even for the same user, the measured distribution across legs can drift as the user leans, crosses legs, or scoots on the seat. Additionally, floor compliance (carpet vs. hard floor), chair leg geometry, and sensor pad compression can change the effective transfer function from applied force to sensor reading. These effects create two risks: (1) an occupied chair could be misclassified as unoccupied if the thresholds are too aggressive or if load is unevenly distributed; and (2) an unoccupied chair could be misclassified as occupied due to static offsets, sensor drift, or baseline miscalibration.

To mitigate these issues, the system treats occupancy detection as a calibrated, hysteretic decision rather than a single fixed threshold. At startup (or on a user-initiated calibration), the controller records an unoccupied baseline over a short averaging window and sets thresholds relative to this baseline. During operation, readings are filtered using a moving average (or low-pass filter) to reduce noise, and hysteresis + time-based debounce is applied so the occupancy state only changes if the signal remains beyond the threshold for a minimum duration. In addition, using multiple sensors (one per leg, or a subset of legs) allows aggregation (e.g., sum or weighted sum) to reduce sensitivity to shifting weight distribution. The tolerance goal is to preserve a clear separation between the unoccupied baseline and the occupied load envelope across different users and sitting styles.

Sensor drift, offset, and quantization.

Load sensors and their readout chains (e.g., load cell + HX711) exhibit offset and gain variation, temperature drift, and long-term creep under constant load. Quantization and sampling noise can also create jitter near decision thresholds. These effects are addressed by (1) operating on averaged values rather than instantaneous samples, (2) defining a deadband around thresholds via hysteresis, and (3) optionally supporting periodic re-baselining when the chair is confidently unoccupied. If the design uses a digital amplifier/ADC (HX711), the primary interface tolerance becomes susceptibility to EMI and timing issues on the digital lines; this is mitigated with proper grounding, short wiring, and stable sampling schedules.

Wearable IMU tolerances (placement and motion variability).

IMU-based activity verification depends on consistent detection of movement patterns across two sensor locations (e.g., ankle + waist/thigh). In practice, strap tightness, orientation, and placement can vary between users and between sessions. Furthermore, motion styles differ (walking cadence, gait asymmetry, short steps vs. long steps), which can shift feature magnitudes. These tolerances create the risk of false negatives (the user walks but the system does not recognize sufficient activity) or false positives (localized shaking that accidentally matches a simple threshold).

We address this by using orientation-robust features (e.g., acceleration magnitude rather than a single axis), requiring agreement across two IMUs, and evaluating activity over a window (e.g., multiple seconds) rather than a single burst. Thresholds are selected based on empirical testing across multiple users and repeated trials, and the design can incorporate conservative margins (e.g., require a minimum activity duration rather than a single peak). If needed, a short calibration step can record "standing still" noise floors to set per-device motion thresholds.

Power and system-level tolerances.

Reliable sensing also requires stable power rails. The chair unit typically runs from 5 V USB, regulated to 3.3 V for the ESP32 and sensor interfaces. Voltage droop (e.g., from buzzer activation or poor USB sources) can introduce measurement noise, BLE instability, or brownouts. The design mitigates this through local decoupling near the ESP32 and sensor front ends, separation of high-current alarm paths from sensitive measurement grounds, and conservative regulator headroom. For the wearable nodes, battery voltage variation over discharge must remain within regulator operating margins to avoid IMU read errors or BLE dropouts. Firmware timeouts and reconnection logic ensure that brief wireless loss does not place the system into an unsafe or undefined state.

Overall, the tolerance strategy is to (1) avoid single-point, tight thresholds, (2) filter and debounce all state transitions, (3) use calibration and hysteresis to handle offsets/drift, and (4) design the activity verification logic to be robust to realistic placement and motion variation. This ensures that the system behavior remains stable and reliable under typical real-world conditions rather than only under ideal lab setups.

3. Design Verification

Design verification focuses on validating correct seat occupancy detection, reliable wireless communication, and accurate verification of user movement. All verification procedures are designed to be repeatable, measurable, and tied directly to system requirements. Testing is performed with the chair electronics mounted in the intended configuration (pressure/load sensors under chair legs, chair-mounted ESP32 control unit, and two wearable IMU nodes communicating via BLE). For each test, we record raw sensor data, system state transitions, and timestamps so that pass/fail decisions are based on quantitative evidence rather than subjective judgment.

3.1 Chair Sensor Verification

Chair sensor verification ensures that the chair-leg pressure/load sensing subsystem reliably detects whether the user is seated and correctly identifies sit→stand and stand→sit transitions under typical usage conditions. Since the vibration sensor has been removed from the design, chair verification focuses on occupancy accuracy, timing, stability against noise, and resistance to false transitions caused by posture shifts.

3.1.1 Occupancy Detection Timing and Accuracy

Requirement: The system shall detect transitions between occupied and unoccupied states within 1 second.

Verification Procedure:

1. Place the chair on a consistent surface (hard floor or carpet; record which surface is used). Power on the chair-mounted unit and run the baseline calibration procedure to establish the unoccupied reference.

2. Perform 50 transition trials total, consisting of alternating sit and stand actions. Each trial includes:
    - Start in a stable state (fully seated or fully standing) for at least 3 seconds.
    - Perform a single, clear transition (sit down fully or stand up fully).
    - Remain in the new stable state for at least 3 seconds before the next trial.
3. For each trial, record:
    - Ground-truth transition time (measured by stopwatch and/or video timestamp, or a manual marker button press).
    - System-detected transition time (timestamp when the ESP32 occupancy state changes).
    - Detection latency = (system timestamp − ground-truth timestamp).
4. The requirement is met if ≥ 45 out of 50 transitions are detected with latency ≤ 1.0 s.

3.2 BLE Communication Verification

BLE communication verification ensures that motion packets from each wearable IMU node are received by the chair-mounted ESP32 reliably and with acceptable latency under typical indoor operation. This test validates connection stability, packet reception rate, and system behavior under real movement.

Verification Metrics:

- Packet reception rate (PRR) per node

- End-to-end latency (packet timestamp at wearable to receipt timestamp at chair)
- Connection stability (disconnect count)

Verification Procedure:

1. Pair both wearable IMU nodes to the chair-mounted ESP32 and start continuous streaming at the intended operating sample rate.
2. Run a 10-minute test where the user moves within a typical room-scale range (e.g., walking, turning, short distance changes).
3. Log:
    - Total packets transmitted (from wearable counters) and total packets received (at chair) for each IMU.
    - Latency statistics (min/mean/max) using timestamps.
    - Any disconnect/reconnect events and their durations.
4. Communication is considered acceptable if:
    - Packet reception rate is ≥ 95% for each IMU over the test duration, and
    - There are 0 unintended disconnections (or, if a disconnect occurs, the system recovers automatically within a defined timeout and does not enter an undefined state).

3.3 Movement Verification Using Two IMUs

Movement verification testing evaluates whether the system correctly distinguishes meaningful physical activity from minor or spoofed motion. The goal is to confirm that alarms are cleared when the user performs a genuine break (e.g., walking) and are not cleared by insufficient activity (e.g., small shaking or localized movement). This section directly validates the "two-IMU verified movement" concept.

3.3.1 True Activity Acceptance Test (Walking/Exercise)

Requirement: The system shall clear the alarm only after detecting sufficient activity on both IMUs.

Verification Procedure:

1. Force the system into the alarm state by exceeding the sitting-time threshold (or by using a test mode that triggers alarm immediately).
2. The user stands up (chair sensors should indicate unoccupied) and then performs a defined activity routine, such as:
    - Walk continuously for X seconds (e.g., 30–60 s), or
    - Perform a light exercise sequence (e.g., marching in place).
3. Repeat for 10 trials, recording whether the alarm clears, and the measured activity duration reported by the system.
4. The requirement is met if the alarm clears in ≥ 9 out of 10 trials when the user performs the routine correctly.

3.3.2 Spoof / Insufficient Activity Rejection Test

Requirement: The system shall not clear the alarm due to minor movements that do not represent meaningful activity.

Verification Procedure:

1. Trigger alarm state and ensure the chair indicates unoccupied (user stands up but does not walk).
2. Perform spoof attempts such as:
   - Shaking only the ankle IMU while keeping the torso IMU relatively still,
   - Shaking only the torso/waist IMU,
   - Brief movements shorter than the required activity duration (e.g., 3–5 seconds),
   - Small in-place motions that do not resemble walking cadence.
3. Perform 20 spoof trials and record whether the alarm clears.
4. The requirement is met if the alarm clears in ≤ 1 out of 20 trials (≤ 5% false-clear rate).

4. Ethics, Safety and Societal impact

This project aims to improve public health and welfare by reducing prolonged sedentary time. By combining chair occupancy sensing with wearable IMU motion verification, the system avoids clearing alerts from minor fidgeting and encourages users to take real activity breaks. The design is low-cost and low-power, making it practical for personal use without requiring cloud services, which also reduces environmental and infrastructure burden.

Relevant engineering standards and frameworks include the IEEE Code of Ethics and ACM Code of Ethics (prioritizing public welfare, honesty about limitations, and privacy), Bluetooth Low Energy specifications for wireless behavior, and common battery safety expectations for lithium-based rechargeable cells (e.g., industry practice consistent with IEC/UL-style safety requirements for portable batteries and charging). For a commercial version, FCC Part 15 emissions compliance would also apply.

Ethically, the main concern is privacy because motion and occupancy data can be considered sensitive. To address this, the system processes data locally and does not identify users or transmit personal information to external servers. Another concern is accessibility: users with mobility limitations may not be able to satisfy movement requirements. We will clearly state intended use (able-bodied users) and avoid marketing the device as a universal health solution. A third concern is misuse in coercive settings (e.g., imposed by a manager). The device is intended for voluntary personal use, with user-configurable settings and the ability to pause/disable.

Safety concerns include lithium battery risks (short circuit, overheating, charging hazards), electrical faults from wiring/soldering, and mechanical hazards from wearable mounting and chair integration (sharp edges, strain, pinch points). We mitigate these risks by using protected power paths where possible, adding current limiting/fusing, fully enclosing electronics, providing strain relief and secure mounting, performing current-limited bring-up and inspection checks, and including user guidance that activity should be performed safely and at the user's discretion (and to consult a physician if unsure). Overall, these design choices protect both users and developers and align with course safety guidelines and professional ethics principles.

5. Costs

Table 1 summarizes the estimated cost of all major hardware components used in this project. All costs are based on typical retail pricing unless otherwise noted. The total estimated cost remains within the $150 project budget.

5.1 Parts

Estimated costs are based on common retail pricing for development boards and sensor modules.

Part	Manufacturer	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
ESP32-S3-WROOM-1 * 3 https://www.mouser.com/ProductDetail/Espressif-Systems/ESP32-S3-WROOM-1-N4R8?qs=sGAEpiMZZMu3sxpa5v1qrkR%2F6t0IkXq8HDNCqbcSH4Q%3D	Espressif	5.71	17.13	17.13
Lithium Ion Polymer Battery - 3.7V 350mAh * 3 https://www.adafruit.com/search?q=lithium+ion+polymer+battery	PKCELL	5.95	17.85	17.85
MCP73871T-4CAI/ML *3 MCP73871T-4CAI/ML Microchip Technology | Mouser	Microchip	2.41	7.23	7.23
Power Regulator: AP63203WU-7 *3 AP63203WU-7 Diodes Incorporated | Mouser	Diodes Incorporated	0.71	2.13	2.13

Accelerometer: LSM6DSOTR *2 https://www.mouser.com/ProductDetail/STMicroelectronics/LSM6DSOTR?qs=lc2O%252BfHJPVbD197C5c7oNw%3D%3D	STMicroelectronics	3.51	7.02	7.02
Switch * 3 https://www.mouser.com/ProductDetail/Same-Sky/SLW-1276864-4A-D?qs=1Kr7Jg1SGW90vOt02EYiag%3D%3D&mgh=1&utm_id=17222215321&utm_source=google&utm_medium=cpc&utm_marketing_tactic=amercorp&gad_source=1&gad_campaignid=17219204619&gbraid=0AAAAADn_wf1d7vMLkJEH7XdSMqSjPH5v0&gclid=Cj0KCQiA49XMBhDRARIsAOOKJHZqmVJ6trpqWmvjrblf-ANRokYi009oVHaglYaCTH5ePDS75anYwewaAkvKEALw_wcB	Same Sky	0.85	2.55	2.55
USB_C_Port *3 USB4105-GF-A GCT | Mouser	GCT	0.78	2.34	2.34
Inductor * 3 https://www.mouser.com/ProductDetail/652-SDR0403-3R9ML	Bourns	0.45	1.35	1.35
Load cells + HX711 ADC module https://www.amazon.com/gp/aw/d/B097T3SX6W/?_encoding=UTF8&pd_rd_plhdr=t&aaxitk=e7a838f3ad87601be620b01f1a854a18&hsa_cr_id=1628395870701&qid=1771487968&sr=1-1-9e67e56a-6f64-441f-a281-df67fc737124&ref_=sbx_s_sparkle_sbtcd_asin_0_title&pd_rd_w=MVXKW&content-id=amzn1.sym.9f2b2b9e-47e9-4764-a4dc-2be2f6fca36d%3Aamzn1.sym.9f2b2b9e-47e9-4764-a4dc-2be2f6fca36d&pf_rd_p=9f2b2b9e-47e9-4764-a4dc-2be2f6fca36d&pf_rd_r=8XB0JZZZGJ1SK25JWP4N&pd_rd_wg=Ll481&pd_rd_r=f589f86f-85ed-42c7-976f-009ed423bbc7	SazkJere	9.99	9.99	9.99
Buzzer https://www.mouser.com/ProductDetail/VCC/MI-TH12-0323-RKT?qs=PBDs2xElII%2F%2FvBeXnF6EpQ%3D%3D&srsltid=AfmBOor4I-iKhqQ61qZcVHODjE4JdKo5qzT1QCQgVaVGyCCNVzZmaXpr	VCC	1.65	1.65	1.65
LCD character display https://www.adafruit.com/product/292	Adafruit Industries	9.95	9.95	9.95

NPN BJT MMBTA42-7-F Diodes Incorporated | Mouser	Diodes Incorporated	0.17	0.17	0.17
Connectors (10 pack) https://www.adafruit.com/product/4392	Adafruit Industries	3.50	3.50	3.50
Pigtail *2 https://www.adafruit.com/product/3955	Adafruit Industries	1.50	3.00	3.00
Button *2 https://www.mouser.com/ProductDetail/179-TS026643BK100SCR	Same Sky	0.10	0.20	0.20
Diode *6 https://www.mouser.com/ProductDetail/LITEON/LTST-C171GKT?qs=DXsB9OwW3bmgO%252BU8es%2F20A%3D%3D	w	0.12	0.72	0.72
Resistors 100k *3 https://www.mouser.com/ProductDetail/YAGEO/RC0805JR-10100KL?qs=fRxXYNj8ftGPwsA%2BEUhHIw%3D%3D	YAGEO	0.10	0.30	0.30
Resistors 27k *3 https://www.mouser.com/ProductDetail/YAGEO/AC0805JR-0727KL?qs=TI15CVqcli2qQSyvr%2BVsEQ%3D%3D	YAGEO	0.10	0.30	0.30
Resistors 10k *9 https://www.mouser.com/ProductDetail/YAGEO/RV0805JR-0710KL?qs=qpJ%2B%2B%2Bdg6p3bGiIDT3p%2B0w%3D%3D	YAGEO	0.10	0.90	0.90
Resistors 5.1k *6 https://www.mouser.com/ProductDetail/YAGEO/AC0805JR-075K1L?qs=cGU37fGUGy%2F7fJWuoZ1Jnw%3D%3D	YAGEO	0.10	0.60	0.60
Resistors 4.7k *8 https://www.mouser.com/ProductDetail/YAGEO/AC0805JR-134K7L?qs=sGAEpiMZZMtlubZbdhIBIHdTGjmUHUjxlbEsBo6xKeQ%3D	YAGEO	0.27	2.16	2.16
Resistors 3.3k *3 https://www.mouser.com/ProductDetail/YAGEO/RC0805JR-073K3L?qs=KyGMWp10Uktjhcr85NC30g%3D%3D	YAGEO	0.10	0.30	0.30
Resistors 1k *6 https://www.mouser.com/ProductDetail/YAGEO/RC0805JR-071KL?qs=KyGMWp10UkuwbpWNcSKeOA%3D%3D	YAGEO	0.10	0.60	0.60

Capacitor 22uF *6 https://www.mouser.com/ProductDetail/Samsung-Electro-Mechanics/CL21A226KQQNNNE?qs=yOVawPpwOwm2L4FqG%252BUr4g%3D%3D	Samsung Electro-Mechanics	0.20	1.20	1.20
Capacitor 10uF *6 https://www.mouser.com/ProductDetail/KYOCERA-AVX/08056D106KAT2A?qs=i9jpE441ig%2FecNv9tOoT%2FA%3D%3D	Samsung Electro-Mechanics	0.10	0.60	0.60
Capacitor 4.7uF *6 https://www.mouser.com/ProductDetail/Samsung-Electro-Mechanics/CL21A475KPFNNNF?qs=yOVawPpwOwl%252B8PBenlc6lg%3D%3D	Samsung Electro-Mechanics	0.10	0.60	0.60
Capacitor 1uF *6 https://www.mouser.com/ProductDetail/Samsung-Electro-Mechanics/CL21B105KAFNNNG?qs=yOVawPpwOwmXkU%252BG0TK%2F5g%3D%3D	Samsung Electro-Mechanics	0.10	0.60	0.60
Capacitor 0.1uF *3 https://www.mouser.com/ProductDetail/KEMET/C0805C104M5RACTU?qs=VOOUd%252Bza08qHu13WgNByHQ%3D%3D	Samsung Electro-Mechanics	0.10	0.30	0.30
Total				**95.24**

5.2 Schedule

Week	Person	Task
Feb 23 – Mar 1	**Jack**	**PCB v1 design completed by 2/27**; run DRC/ERC; export Gerber/drill files; finalize v1 BOM for ordering

	Chris	Contribute to and review control-system schematics (power tree, I/O map, sensor/alarm interfaces); define bring-up debug points/ports
	Melissa	Set up software repository and project structure; define module split (BLE central / wearable / UI); draft logging + packet field conventions
Mar 2 – Mar 8	**Jack**	Submit/order PCB v1; prepare bring-up checklist (power-first, current-limited power-up, interface test order)
	Chris	Prototype seat-occupancy detection logic (filtering, debounce, thresholds, hysteresis); draft sitting timer + alarm FSM
	Melissa	Wearable prototype: IMU sampling at target rate; BLE advertise/connect; packetization (seq + timestamp)
Mar 9 – Mar 15	**Jack**	Initial mechanical/mounting plan (sensor placement, cable routing, strain relief); prepare assembly materials checklist
	Chris	BLE central prototype: subscribe/stream/queue pipeline; define "valid streaming" checks (timeouts, seq gaps)
	Melissa	UI/button control prototype (interval adjust, long-press/combos); basic alarm outputs (LED/buzzer)
Mar 16 – Mar 22	**Jack**	Pre-arrival PCB checks: footprint/orientation review; soldering/assembly plan and tools readiness
	Chris	Calibration plan: baseline + threshold defaults; define verification test cases and metrics

	Melissa	Robustness work: disconnect/reconnect handling; data logging scripts (serial/CSV) for tuning
Mar 23 – Mar 29	**Jack**	PCB v1 assembly + bring-up (rails, UART, I2C/SPI) using current-limited supply; verify core power integrity
	Chris	Validate on-board sensor chain (seat sensing front-end); stabilize occupancy state output
	Melissa	End-to-end BLE link: wearable → central streaming; verify timestamp/sequence handling
Mar 30 – Apr 5	**Jack**	Hardware fixes/rework (solder, wiring, noise/grounding); draft PCB v2 change list if required
	Chris	Integration: sitting timer → alarm trigger; movement-verification gating before clearing alarm
	Melissa	Complete UI path: interval changes take effect; lockout behavior during alarm; consistent debug logging
Apr 6 – Apr 12	**Jack**	Full assembly reliability (wire management, strain relief, enclosure fit); battery power sanity + safety checks
	Chris	Reduce false clears/false alarms via tuning; add fail-safes (pause/disable/timeout)
	Melissa	BLE stability testing (loss, range, reconnect time); validate queue + timestamp processing

Apr 13 – Apr 19	**Jack**	Finalize mechanical integration (wearability comfort, under-chair mounting stability); prepare demo backup hardware plan
	Chris	Small user testing (2–3 subjects); record metrics (false alarm/clear rates) and retune thresholds
	Melissa	Freeze demo script and test checklist; simplify on-screen/log outputs for live explanation
Apr 20 – Apr 26	**Jack**	Hardware regression testing (long run, temperature spot checks, power stability); enclosure/wiring final polish
	Chris	Integration bug triage; freeze final state machines; write concise test summary for the final report
	Melissa	Software cleanup (README, parameters, build/flash steps); prepare demo video/use-cases
Apr 27 – May 3	**Jack**	Final demo hardware readiness (quick troubleshooting checklist, spare cables/battery)
	Chris	Final end-to-end verification (trigger → verify → clear); prepare technical talking points (robustness/logic)
	Melissa	Final presentation support materials (FSMs, results plots); live logging/visualization for demo
May 4 – May 10	**Jack**	Archive PCB/schematics/manufacturing files; finalize BOM + cost summary; support lab checkout

	Chris	Archive software (tagged release, test notes); refine technical sections of the final paper
	Melissa	Finalize report figures/results; complete handoff package (demo instructions, reproducibility steps)

6. Conclusion

6.1 Accomplishments

This project proposes and designs a smart anti-sedentary chair that combines chair-mounted sensing with wearable IMU nodes to improve reliability in sedentary behavior detection. The architecture fuses seat occupancy detection (pressure/load sensing with filtering and debouncing) and BLE-based motion verification from one or more IMU wearables. Instead of clearing alerts based solely on brief posture shifts, the system requires evidence of meaningful movement before disabling the prolonged sitting alarm. This multi-modal design reduces false positives (e.g., leaning, minor fidgeting) and false negatives (e.g., sensor drift or seat-only ambiguity) compared to chair-only approaches. The design also defines a clear data-handling pipeline including BLE connection management, packet timestamping/sequence tracking, and a state machine that coordinates sitting time accumulation and alarm behavior.

6.2 Uncertainties

Key uncertainties include inter-user variability and environmental variability. Different users may have distinct movement patterns (e.g., leg bouncing vs. standing up), body weights, and seating habits that affect both seat sensor readings and IMU signatures. As a result, calibration parameters (baseline offsets, thresholds, hysteresis/debounce constants) may require per-user tuning or robust default values. Additional uncertainty comes from hardware and communication factors such as sensor noise, mechanical mounting differences (sensor placement under cushions), battery voltage effects, and BLE packet loss or connection instability. These factors can impact the reliability of "valid streaming" and may require conservative timeout/reconnect logic and buffer-based decision making.

6.3 Ethical considerations

The system is designed to minimize privacy and data ethics risks. It processes occupancy and motion data locally on embedded hardware and does not attempt to identify individuals. No biometric data, audio/video, or personal identifiers are collected, and no information is transmitted to external servers by design. Only the minimum sensing needed for sedentary detection and movement verification is used, and motion features can be computed in real time without long-term storage. This reduces the risk of surveillance-like monitoring and limits exposure in the event of device compromise. Any optional logging functionality (if added later) should be strictly opt-in and clearly communicated to users.

6.4 Future work

Future work could improve usability, accuracy, and generalization. One direction is adaptive calibration: automatically learning user-specific baselines and movement thresholds over time to reduce manual tuning. Another direction is richer activity classification using IMU features (e.g., standing, walking, leg movement intensity) to distinguish true breaks from minor motion. System-level improvements could include better power optimization for wearable nodes, multi-device synchronization for improved motion verification, and more robust BLE error recovery with metrics-based streaming validation. If long-term analytics are desired, an optional logging mode could store only high-level summary statistics (e.g., sitting duration and break counts) rather than raw motion data to maintain privacy while enabling behavioral insights.

References

[1] World Health Organization, *WHO Guidelines on Physical Activity and Sedentary Behaviour*. Geneva, Switzerland: WHO, 2020. Available online.

[2] Espressif Systems, *ESP32-S3 Technical Reference Manual*. Espressif Systems. Available online.

[3] Espressif Systems, *ESP32-S3-WROOM-1 / ESP32-S3-WROOM-1U Datasheet*. Espressif Systems. Available online.

[4] InvenSense (TDK), *MPU-6000 and MPU-6050 Product Specification*, Doc. No. PS-MPU-6000A-00, Rev. 3.4, Aug. 19, 2013. Available online.

[5] InvenSense (TDK), *MPU-6000 and MPU-6050 Register Map and Descriptions*, Doc. No. RM-MPU-6000A-00, Rev. 4.2, Aug. 19, 2013. Available online.

[6] Avia Semiconductor, *HX711: 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales* (Datasheet). Available online.

[7] Interlink Electronics, *FSR 402 Force Sensing Resistor Datasheet*. Available online.

[8] SparkFun Electronics, *FSR 400 Series Datasheet* (FSR-402 family). Available online.

[9] SparkFun Electronics, *TAL220B Miniature Load Cell Datasheet (2–50 kg)*. Available online.

[10] TE Connectivity / Measurement Specialties, *LDT Series Piezo Film Vibration Sensor/Switch Datasheet* (e.g., LDT0-028K/LDTM-028K). Available online.

[11] JST Mfg. Co., Ltd., *PH Series: 2.0 mm Pitch Wire-to-Board Connector Datasheet*. Available online.

[12] Adafruit Industries, *Lithium Ion Polymer Battery – 3.7V 500mAh* (Product ID: 1578). Available online.

[13] Adafruit Industries, *I2C / SPI Character LCD Backpack – STEMMA QT / Qwiic* (Product ID: 292). Available online.

[14] Adafruit Industries, *JST PH 2mm 4-Pin to Male Header Cable – I2C STEMMA Cable – 200mm* (Product ID: 3955). Available online.

[15] Adafruit Industries, *2mm 4-pin Horizontal Connector – JST PH Compatible – 10 Pack – STEMMA* (Product ID: 4392). Available online.