

Pressure Sensor Device

DESIGN DOCUMENT

Team 12

Clients:

Adaptive Adventures & BAE Systems

Advisors:

Santosh Pandey, Rachel Shannon, Nick Fila,
John Wagner, Yunsoo Park

Team Members:

Aina Azman
Sabrina Francis
Bilal Hodzic
Zane Lenz
Osaid Samman
Ivan Alvarado-Santoy
Nathan Turnis

Email: sdmay25-12@iastate.edu

Website: <https://sdmay25-12.sd.ece.iastate.edu>

Executive Summary

Pressure sores, also known as pressure ulcers, are a serious health risk for individuals with lower extremity disabilities or those who have suffered significant damage to their lower limbs. These individuals often lack the sensation necessary to detect when they are sitting with poor posture, which can lead to prolonged pressure on vulnerable areas such as the sit bones. Over time, this can result in the development of pressure sores, which are not only painful and life-altering but can also be fatal in severe cases. To tackle this problem, our project set out to design a pressure-sensing device integrated with a mobile application. The primary design requirements were: comfort during prolonged use, minimal friction to avoid further skin irritation, sufficient battery life for daily use, reliable detection of pressure imbalances, and timely notifications to the user when a risky posture is detected.

Our solution employs load cells, commonly used in digital scales, to measure the distribution of weight across a seating surface. The hardware prototype consists of a firm cutting board with four strategically placed cut-out squares, each housing a load cell. These sensors gather real-time pressure data, which is amplified and transmitted to a Raspberry Pico W microcontroller. The microcontroller then sends the processed data via Bluetooth to an Android application. The app visualizes the pressure distribution as a heat map, making it easy for users to understand their current posture. An algorithm continuously analyzes the data and triggers notifications if it detects sustained imbalances that could lead to pressure sores.

Initial testing with the prototype indicates that the system meets the core requirements: it is comfortable to sit on, does not introduce additional friction, and provides accurate, real-time feedback. The app's heat map and notification system have been well-received in trials, suggesting the design effectively addresses the needs of the target audience.

The potential implications of our design are significant, as it lays the groundwork for a practical and accessible solution to help prevent pressure sores in individuals with lower extremity disabilities. Looking ahead, we see a lot of potential for the next team to

build on our work. Our prototype functions well, but it is ready to be refined into a device that is easier to manufacture and more polished for everyday use. Next steps would include expanding app compatibility to iOS through cross-platform frameworks like Flutter or React Native would broaden accessibility. On the software side, there is room to improve the app's visuals and add useful features such as offline support or customizable pressure alerts, making the device even more versatile and adaptable to a wider range of user needs. Additionally, transitioning from prototype materials to professional-grade manufacturing would enhance durability and commercial appeal, while addressing regulatory requirements would ensure compliance with safety and quality standards for market distribution. Together, these steps would enable the transition from a functional prototype to a commercially available, user-friendly device that could have a meaningful impact on the lives of those at risk for pressure sores.

Learning Summary

Development Standards & Practices Used

- ISO/IEC/IEEE International Standard - Software and systems engineering - Software testing – Part 2: Test processes
- ISO/IEC/IEEE International Standard - Systems and Software Engineering--Life Cycle Management--Part 5: Software Development Planning
- IEEE 802 Nendica Report: Flexible Factory IoT: Use Cases and Communication Requirements for Wired and Wireless Bridged Networks

Summary of Requirements

- The device does not cause additional friction
- Comfortable to sit on
- Sufficient battery life for average use
- Detects an imbalance in pressure
- Timely notifications of imbalance in pressure
- Wireless connection between app and device
- Functional for different types of adaptive sports equipment
- Easy for the user to set up the system by themselves
- Water-resistant

Applicable Courses from Iowa State University Curriculum

- | | |
|--------------|--------------|
| • CPRE 2880 | • COM S 3110 |
| • COM S 3090 | • COM S 3190 |
| • COM S 2270 | • EE 2010 |
| • COM S 2280 | • EE 2300 |
| • SE 3390 | • CPRE 4890 |
| • CPRE 4190 | • CPRE 3880 |
| • CPRE 4140 | • CPRE 18 |

New Skills/Knowledge acquired that was not taught in courses

- Kotlin programming language and modern Android application development
- 3D Modeling, 3D Scanning, 3D Printing
- Soldering

Table of Contents

1. Introduction	8
1.1. PROBLEM STATEMENT	8
1.2. INTENDED USERS	8
2. Requirements, Constraints, And Standards	10
Functional Requirements	10
Resource Requirements	10
Physical Requirements	11
Aesthetic Requirements	11
User Experiential Requirements	11
UI Requirements	12
3 Project Plan	13
3.1 Project Management/Tracking Procedures	13
3.2 Task Decomposition	14
3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria	15
3.4 Project Timeline/Schedule	17
3.5 Risks and Risk Management/Mitigation	17
3.6 Personnel Effort Requirements	19
3.7 Other Resource Requirements	21
4 Design	21
4.1 Design Context	21
4.1.1 Broader Context	21
4.1.2 Prior Work/Solutions	23
1. TekScan CONFORMat System [5]	23
2. TekScan Body Pressure Measurement System (BPMS) [6]	23
1. TekScan F-Socket System [7]	23
4.1.3 Technical Complexity	25
4.2 Design Exploration	28
4.2.1 Design Decisions	28
4.2.2 Ideation	29
4.2.3 Decision-Making and Trade-Off	31
4.3 Final Design	33
4.3.1 Overview	33
4.3.2 Detailed Design and Visual(s)	36
4.3.3 Functionality	39
4.3.4 Areas of Challenge	41
4.4 Technology Considerations	41
5 Testing	41
5.1 Unit Testing	42
5.2 Interface Testing	42
5.3 Integration Testing	43

5.4 System Testing	43
5.5 Regression Testing	43
5.6 Acceptance Testing	43
5.7 User Testing	44
5.9 Results	44
6 Implementation	44
6.1 Design Analysis	46
7 Ethics and Professional Responsibility	47
7.1 Areas of Professional Responsibility/Codes of Ethics	47
7.2 Four Principles	49
7.3 Virtues	50
8 Conclusions	55
8.1 Summary of Progress	55
8.2 Value Provided	55
8.3 Next Steps	56
9 References	56
10 Appendices	56
Appendix 1 – Operation Manual	57
Appendix 2 – Alternative/Initial Version of Design	68
Appendix 3 – Code	69
Appendix 4 – Team Contract	69
Team Procedures	70
Participation Expectations	70
Leadership	71
Collaboration and Inclusion	71
Goal-Setting, Planning, and Execution	72
Consequences for Not Adhering to Team Contract	72

List of Figures

Figure Number	Figure Title	Page Number
3.2.1	Progression with Agile Sprints	15
3.4.1	Project Timeline	17
3.6.1	Project Effort Estimates	20
3.6.2	Actual Effort	20
4.1.1	Project Design Considerations	22
4.1.2	Comparison of Solutions	24
4.2.2.1	Load Cells	29
4.2.2.2	ShuntMode MatrixArray	30
4.2.2.3	FlexiForce A201 Sensor	30
4.2.2.4	Thin Film Pressure Sensor	30
4.2.2.5	Thin Film Force Sensor	31
4.2.3.1	Weighted Decision Matrix	32
4.3.2.1	Hardware Design Schematic	36
4.3.2.2	Application Wireframe	36
4.3.2.3	Application Housing	36
4.3.2.4	Strain Gauge	37
4.3.2.5	Strain Gauge Circuit	37
6.1	Final Prototype of Pressure Sensor Device	45
6.2	Bottom View of Sensor Device	45
6.3	Final Prototype of Android Application	47
7.1.1	Professional Responsibility Table	48
7.2.1	Guiding Principles Table	50

1. Introduction

1.1. PROBLEM STATEMENT

People with disabilities often need special equipment to do things that most people can do easily. With modern technology, doing things like playing sports, skiing, and rock climbing are no longer out of reach for people with disabilities. Even with advancements in this area, there are still some issues that can make activities like this problematic for people with disabilities. One of these potential problems is pressure sores. Pressure sores occur when there is too much pressure or an imbalance of pressure on some body part. Pressure sores can develop within minutes, and for people who are partially paralyzed or have missing limbs, it can be very hard to know when they may be at risk of developing one. Pressure sores can potentially become so severe that they are life-altering or life-threatening. For someone who is disabled and does not have an effective way of preventing pressure sores, this can become a constant worry when going out and doing things that are more physically demanding. For people like this, a device that can monitor pressure in areas where they cannot feel it is almost necessary; however, such a device does not yet exist.

1.2. INTENDED USERS

a. Individuals with Adaptive Equipment

Disabled individuals who use adaptive equipment often need to abandon their standard tools to participate in physical activities like sports. These users frequently experience challenges due to a lack of sensation or nerve damage in certain parts of their body, making it difficult to detect excess pressure. Additionally, they may have missing limbs or be paralyzed, further complicating their ability to monitor physical strain. To address these challenges, these individuals require a way to recognize excessive pressure on specific parts of the body, enabling them to play adaptive sports safely without a heightened risk of injury. They also need a reliable method to detect and act on preventable injuries before they escalate.

The benefits of a pressure-monitoring device are significant for this group. It enables users to accurately identify excess pressure on body areas with reduced sensation, fostering a safer experience during physical activities. Such a device also provides peace of mind, empowering users to participate in adaptive sports freely without constant fear of injury. Furthermore, it helps prevent pressure-related injuries that could lead to severe complications, enhancing overall well-being and quality of life.

b. Caregivers/Coaches for Adaptive Sports Athletes

Caregivers and coaches for adaptive sports athletes work with individuals with disabilities, helping them engage in sports safely. These professionals are trained to accommodate various needs and understand the risks and potential complications associated with adaptive sports. They require tools to assist individuals who may be unable to communicate or detect physical problems, ensuring that their players are taking adequate precautions to avoid injury. Additionally, caregivers and coaches need methods to encourage participation among those hesitant to engage in sports due to concerns about pressure-related injuries.

A pressure-monitoring device offers several advantages for caregivers and coaches. It allows them to assist athletes in detecting injuries related to pressure that they might not feel, ensuring the safety of their players during activities. By providing this level of support, caregivers can instill confidence in their athletes, enabling them to enjoy adaptive sports without undue worry. Moreover, such a device can help caregivers build trust with skeptical participants, encouraging broader involvement in adaptive sports.

c. Healthcare Providers

Healthcare providers specializing in disabilities play a critical role in ensuring patient safety and well-being. These professionals have the expertise to recognize when preventive measures are necessary and identify individuals at high risk of complications from pressure-related injuries. Providers need tools to help patients remain safe when outside their direct care, as well as new options to recommend to those interested in adaptive sports but hesitant due to the potential for injury. Additionally, they require methods to gather accurate data on patient activity to suggest appropriate preventive measures.

With a pressure-monitoring device, healthcare providers gain the ability to recommend practical solutions for individuals at risk of pressure sores that they cannot feel. This device enhances patient safety during high-risk activities and provides valuable data for healthcare professionals to analyze and use in creating customized treatment plans or preventive strategies. Ultimately, this technology empowers providers to better support their patients' active lifestyles while minimizing the risk of injury.

2. Requirements, Constraints, And Standards

2.1. REQUIREMENTS & CONSTRAINTS

Functional Requirements

- **Pressure Sensing:** The mat must detect and measure pressure levels across multiple zones of the lower extremities, providing real-time feedback.
- **Threshold-Based Alerts:** The system must notify the user when pressure exceeds a set threshold in any zone, helping prevent pressure sores.
 - Constraint: Alert within 1 minute of threshold breach.
- **Wireless Connectivity:** The mat must wirelessly transmit data to a mobile app for remote monitoring.
 - Constraint: Bluetooth range of up to 10 meters.
- **Data Storage:** The app must store pressure data logs for post-use analysis, with at least 7 days of data retrievable at any time.
- **Multi-Zone Sensitivity:** The mat must distinguish between different pressure zones, allowing for precise monitoring of specific areas
 - Constraint: Minimum of 4 sensor zones across the mat.

Resource Requirements

- **Sensor Technology:** The device must integrate pressure-sensitive sensor technology (e.g., Sensitronics Matrixarray, Load Cell) capable of detecting small changes in pressure.

- Constraint: Detects pressure up to 100 lbs.
- **Power Source:** The system must be powered by a rechargeable battery that allows for continuous use.
 - Constraint: Minimum battery life of 8 hours.
- **Microcontroller:** The system must include a microcontroller that can process data from multiple sensors and handle wireless transmission (e.g., Arduino or ESP32).

Physical Requirements

- **Mat Dimensions:** The mat must be large enough to cover the typical area of use for the lower extremities.
 - Constraint: Minimum size 15 inch x 15 inch.
- **Portability:** The mat should be lightweight and foldable, allowing for easy transport and storage
 - Constraint: Total weight must not exceed 2 lbs.
- **Durability:** The mat must be durable and withstand regular use without sensor degradation
 - Constraint: Must function properly after at least 1000 hours of use.

Aesthetic Requirements

- **Low Profile Design:** The mat must have a thin, flexible design so it can be comfortably placed under the user's legs without causing discomfort
 - Constraint: thickness must not exceed 4 inches.
- **Neutral Color Scheme:** The mat should be available in neutral colors to blend with different environments.

User Experiential Requirements

- **Ease of Setup:** The mat must be simple to set up and use, requiring minimal user interaction beyond placing it under the lower extremities.
- **Comfort:** The mat cannot cause extra pressure or friction on the skin.

- **Skin-Friendly Materials:** The surface of the mat must be made from breathable, hypoallergenic materials to avoid skin irritation during prolonged use.
- **Mobile App Interface:** The mobile app must be intuitive and provide clear, easy-to-read pressure data and alerts.
 - **Constraint:** The average user should be able to understand the app within 5 minutes of use.

UI Requirements

3. **Real-Time Visualization:** The app should display real-time visual feedback of pressure distribution across the mat, allowing users to quickly identify problem areas.
4. **Customizable Alerts:** Users must be able to set custom pressure thresholds and receive notifications tailored to their specific needs.
5. **Pressure History Review:** The app must allow users to review historical data and trends to track how pressure distribution changes over time.

5.1. ENGINEERING STANDARDS

Engineering standards play a vital role in ensuring the safety, reliability, and compatibility of modern technologies, including home power systems, phone chargers, and electric vehicles. Among the IEEE standards reviewed, the "ISO/IEC/IEEE International Standard - Software and Systems Engineering - Software Testing -- Part 2: Test Processes" [1] provides comprehensive guidelines for software testing, making it directly relevant to projects involving both app and device development. Similarly, the "ISO/IEC/IEEE International Standard - Systems and Software Engineering—Life Cycle Management—Part 5: Software Development Planning"[2] outlines critical requirements and practices for effective software planning and management, aligning well with structured development workflows. In contrast, the "IEEE 802 Nendica Report: Flexible Factory IoT" [3] focuses on communication requirements in factory environments and is

not applicable to non-industrial projects. Additionally, another team member highlighted the "ISO/IEC/IEEE International Standard - Systems and Software Integration,"[4] which emphasizes integration processes for systems and software. To integrate these standards, the project will prioritize software testing as a key component of the development process while leveraging an already established framework for software development planning and task management, ensuring compliance without requiring significant design modifications.

3 Project Plan

3.1 PROJECT MANAGEMENT/TRACKING PROCEDURES

We adopted both the waterfall and agile methodologies. We continuously developed the hardware alongside the software and tested each. The software relied on the data from the hardware, so it required some waterfall thinking.

We used GitLab to store all code related to the project. We used the *Issues* feature in GitLab to break down things we wanted to do into small tasks. These issues can have branches created from them, keeping our code separate and helping to avoid conflicts. These issues were contained within Issue Boards, which kept track of the status. We used Discord for communication and providing updates.

3.2 TASK DECOMPOSITION

To effectively achieve the project's objectives, we broke the work into specific tasks and subtasks, taking into account current limitations and planning for future enhancements. This decomposition ensured that progress was measurable and dependencies were clearly managed across hardware, software, and testing phases.

On the hardware side, the first major task involved the selection and testing of load cell sensors. Various sensor types were evaluated to ensure accuracy and sensitivity, followed by rigorous performance testing under varying weight conditions. The second key

task focused on integrating the Raspberry Pi Pico W as the microcontroller for the project. The Pico W was programmed to process sensor inputs and configured to establish a Bluetooth-based client-server model for transmitting data. In this setup, the Pico W served as the server, broadcasting real-time pressure data, while the Android app acted as the client, retrieving and displaying the data. Additionally, we validated the reliability and speed of data transmission in this initial prototype using the Bluetooth configuration. The third hardware task was the design and development of the physical mat, which involved creating sensor placements using CAD software and developing 3D-printed prototypes to house the sensors securely while ensuring durability.

For the software tasks, the first priority was to develop an intuitive user interface for the Android app using tools like Figma or Android Studio. The app includes features such as graphing to visualize pressure distribution in real time. The next step was to focus on ensuring seamless Bluetooth data transmission between the Pico W server and the app client. This involved testing real-time data updates and measuring latency to ensure the system functions effectively. The last step was to receive the data from user testing and create & integrate algorithms to analyze incoming sensor data, detecting imbalances and generating threshold-based alert notifications to prompt corrective action. Finally, the software team's testing and validation are critical components of the project. Unit testing ensures the functionality of individual components, such as sensors, Bluetooth modules, and app UI features. Integration testing combines the hardware and software systems to validate end-to-end functionality, simulating multiple scenarios like varying user weight and pressure distributions. System testing involves user simulations, with individuals sitting on the prototype mat to validate real-time data flow from sensors to the app, triggering notifications for detected imbalances.

To structure progress effectively, the project was divided into four Agile sprints. The first sprint focused on the initial hardware setup, including sensor integration with the Pico W and the establishment of basic server functionality. The second sprint emphasized app interface development and establishing seamless wireless communication. The third sprint integrated real-time data visualization, alerting features,

and comprehensive testing. The final sprint was dedicated to user testing, refining the prototype, and gathering feedback to guide future iterations.

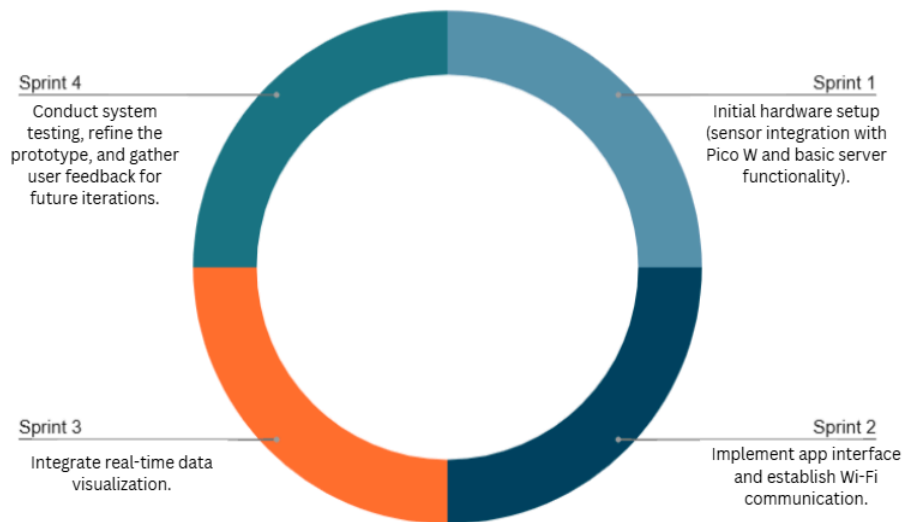


Figure 3.2.1 Progression with Agile Sprints

3.3 PROJECT PROPOSED MILESTONES, METRICS, AND EVALUATION CRITERIA

Our project milestones were designed to align with the key tasks outlined in the decomposition, with measurable metrics to evaluate progress effectively. For hardware development, a primary milestone involves finalizing the selection and integration of load cell sensors. These sensors must detect pressure variations with at least 95% accuracy and respond to pressure changes within 1 second. Additionally, programming the Raspberry Pi Pico W as a server to process and transmit sensor data via Bluetooth is another critical milestone. This functionality will be evaluated based on its ability to maintain a stable data transmission range of 10 meters with a latency of under 200 milliseconds and an uptime of 99% during continuous testing. The pressure sensor device design is also a priority, with targets to achieve a maximum thickness of 4 inches for user comfort and a durability benchmark of 1000 hours of simulated use while supporting up to 100 lbs per load cell.

On the software side, key milestones included the development of a real-time pressure data visualization interface within the Android app. This will be measured by user testing to ensure that users can navigate the app and interpret visualizations within 5

minutes. The app is expected to update pressure readings every second or less. Another significant milestone was the implementation of the pressure imbalance detection algorithm, which must correctly identify imbalances in 95% of test cases and deliver notifications within 5 seconds of detection. Ensuring seamless Wi-Fi data transmission between the hardware and the app is also critical, with metrics focusing on connection stability for at least 8 continuous hours and minimal data loss in at least 95% of transmission tests.

Testing and validation milestones encompass unit, integration, and system-level testing. All individual system components underwent rigorous unit testing, with an expectation of 100% coverage for critical functions. Integration testing validated seamless data flow from load cells to the app through the microcontroller, with a performance benchmark of handling at least 10 simultaneous readings without delays. Finally, system testing involved user feedback to ensure the prototype met real-world usability and accuracy requirements. Feedback from at least five test users indicated a satisfaction rate of 90% or higher, and all critical issues identified during testing were resolved before final deployment.

As we followed an agile development process, these milestones and metrics were refined at the end of each sprint based on feedback and iterative testing. This approach ensured adaptability and continuous improvement throughout the development cycle.

3.4 PROJECT TIMELINE/SCHEDULE



Figure 3.4.1 Project Timeline

3.5 RISKS AND RISK MANAGEMENT/MITIGATION

1. Sensor Performance and Compatibility

- Risk: Sensor accuracy may fall below the required resolution, or selected sensors may not be compatible with the microcontroller.
- Probability: 0.4
- Severity: Moderate

- Mitigation Plan:
 - Alternative Task: Source higher sensitivity sensors or develop signal-processing code to filter noise.
 - Off-the-Shelf Solution: Purchase a more robust, commercially available sensor with known compatibility.
 - Alternative Technology: Test multiple microcontrollers to assess compatibility.

The sensor did not end up falling below the required resolution.

2. App Functionality and User Experience

- Risk: App may not maintain alerts when closed, impacting user responsiveness.
- Probability: 0.5
- Severity: High
- Mitigation Plan:
 - Alternative Task: Investigate background service management for persistent notifications on Android.
 - Off-the-Shelf Solution: Leverage frameworks or libraries specifically designed for background task persistence.
 - Alternative Technology: Explore Power BI's real-time alert capabilities to assist in app functions.

This risk was mitigated by implementing what's known as a Foreground Service in Android. The user is notified that the foreground service is running, which keeps the app running in the background, like a music player.

3. Scheduling and Agile Sprints

- Risk: Delays in sensor testing may impact agile sprint progress and affect overall schedule.
- Probability: 0.4
- Severity: Moderate
- Mitigation Plan:

- Alternative Tasks: Extend sprints by a week to accommodate testing and identify alternatives that might bypass delay factors if necessary.

We did have a slight delay in testing, due to our initial board coming out too small, which broke easily. This required us to draw up a larger board, which took longer than expected to complete.

3.6 PERSONNEL EFFORT REQUIREMENTS

Task	Number of Hours Required	Details
Hardware Integration	120 hours	<ul style="list-style-type: none"> • Need to find the right hardware. • Need to find how to integrate the chosen hardware.
Mobile App Development	160 hours	<ul style="list-style-type: none"> • Need to develop User Interface (UI). • Need to connect wirelessly. • Need to develop the notification's logic.
Wireless Connectivity	40 hours	<ul style="list-style-type: none"> • Need to figure out how to use Bluetooth.
Data Processing	130 hours	<ul style="list-style-type: none"> • Need to figure out when to send a notification and what data will need to be sent.
Testing & Validation	80 hours	<ul style="list-style-type: none"> • Need to constantly test the implementation of

		hardware and software.
--	--	------------------------

Figure 3.6.1 Project Effort Estimates

Task	Number of Hours	Details
Hardware Integration	~120 hours	<ul style="list-style-type: none"> • Needed to find the right hardware. • Needed to find how to integrate the chosen hardware.
Mobile App Development	~ 160 hours	<ul style="list-style-type: none"> • Needed to develop User Interface (UI). • Needed to connect wirelessly. • Needed to develop the notification's logic.
Wireless Connectivity	~ 60 hours	<ul style="list-style-type: none"> • Needed to figure out how to use Bluetooth.
Data Processing	~ 130 hours	<ul style="list-style-type: none"> • Needed to figure out when to send a notification and what data will need to be sent.
Testing & Validation	~ 80 hours	<ul style="list-style-type: none"> • Needed to constantly test the implementation of hardware and software.

Figure 3.6.2 Actual Effort

Our time frames ended up being similar to the actual effort requirements. Creating a Bluetooth connection is longer than expected, as implementing Bluetooth Low Energy in

Android is a relatively complex process, with no easy solution. In the most recent weeks leading up to the conclusion of the project, we have spent most of our time testing.

3.7 OTHER RESOURCE REQUIREMENTS

Skill Acquisition

- Kotlin Programming: Team members needed to learn Kotlin to develop and maintain the Android app efficiently. Resources may include online courses, documentation, and tutorials.

2. Development Tools and Software

- Android Studio: Primary IDE for developing and testing the Android app.
- Database Systems: Any databases or cloud storage services needed for storing and retrieving user data and sensor readings.

3. Testing Equipment

- Debugging and Testing Devices: Access to Android devices (phones or tablets) for testing the app's functionality and real-time sensor data alerts. We used a HMD as our primary Android device.
- Simulators: Android emulators or simulators for testing different device environments and Android versions.

4 Design

4.1 DESIGN CONTEXT

4.1.1 Broader Context

The design of our pressure-sensing device was situated within the broader context of improving accessibility and safety for individuals with disabilities, particularly in the realm of adaptive sports and physical activities. The primary communities we were

designing for included individuals with disabilities who use adaptive equipment, such as wheelchair users, athletes with nerve damage, or those with limited mobility. These individuals often face challenges in detecting excessive pressure on their bodies, especially in areas with reduced sensation, which can lead to serious health complications like pressure sores. The project also directly impacted caregivers, coaches, and healthcare providers who are involved in supporting these individuals. By addressing the need for a tool to monitor and prevent pressure-related injuries, our design provides an invaluable resource to enhance the safety, independence, and quality of life for this community.

Area	Description	Examples
Public health, safety, and welfare	This project enhanced the well-being of individuals by addressing risks related to pressure sores and promoting safer participation in physical activities for individuals with disabilities.	Reducing the risk of pressure sores for wheelchair users, preventing avoidable injuries, improving the safety of adaptive sports participants.
Global, cultural, and social	The design supported values of inclusion and equity by providing tools that empower individuals with disabilities to participate in adaptive sports safely and confidently.	Promoting accessibility in adaptive sports, reflecting global values of disability rights, enhancing inclusivity and social participation.
Environmental	The project aimed to minimize its environmental impact by using sustainable materials and energy-efficient design practices, ensuring the product is eco-friendly.	Using hypoallergenic and eco-friendly materials, reducing waste by using rechargeable batteries, minimizing environmental footprint through sustainable production practices.
Economic	The project was designed to be affordable for the target users, while also providing long-term economic benefits by preventing costly health complications and improving the quality of life.	Making the device affordable for individuals with disabilities and healthcare providers, reducing medical expenses from pressure sores, creating economic opportunities in the adaptive sports sector.

Figure 4.1.1 Project Design Considerations

4.1.2 Prior Work/Solutions

The development of our pressure-sensing device built on a growing body of research and existing products in the field of pressure monitoring systems for individuals with disabilities. These products and studies have laid a solid foundation for understanding the challenges and opportunities in this domain. Below, we detail three relevant prior works, evaluate their advantages and shortcomings, and compare them to our proposed solution.

1. TekScan CONFORMat System [5]

The TekScan CONFORMat system is a state-of-the-art pressure mapping solution designed for seating and positioning analysis. It provides real-time pressure distribution data and tracks the Center of Force (CoF) trajectory, making it a valuable tool for optimizing wheelchair seating systems. The system boasts over 2,000 sensing elements, delivering high-resolution data that can inform treatment decisions and improve cushion selection. Additionally, it uses vivid visual displays to engage patients and identify asymmetries or pelvic obliquities.

2. TekScan Body Pressure Measurement System (BPMS) [6]

The BPMS is an advanced pressure measurement tool designed for optimizing seating, bedding, cushioning, and positioning solutions. It is widely used in clinical and research settings to reduce the incidence of pressure sores, ulcers, and tissue deterioration. The ultra-thin, flexible pressure mapping matrix provides high-resolution data, enabling precise identification of anatomical structures causing concentrated pressure.

1. TekScan F-Socket System [7]

The F-Socket system is a high-resolution pressure mapping tool tailored for orthotists and prosthetists to enhance the design, fit, and functionality of prosthetic limbs. It uses ultra-thin sensors to measure pressure within the socket interface, providing

valuable insights for optimizing socket-stump interfaces and improving user comfort and compliance.

Comparison and Differentiation

Solution	Pros	Cons
TekScan CONFORMat	High-resolution mapping for precise identification of pressure points.	Expensive, limiting accessibility for individual users.
	Portable and lightweight, allowing easy mobility between users and facilities.	
	Useful for research and clinical justification for treatment decisions.	Does not integrate with mobile applications, reducing usability in dynamic, real-world contexts.

TekScan BPMS	Captures detailed measurements of location and magnitude of peak pressures.	The BPMS is prohibitively expensive, making it inaccessible for many users.
	Helps reduce the risk of pressure sores and accelerates healing.	Designed primarily for bed-bound patients, limiting its applicability for individuals using wheelchairs or participating in adaptive sports.
	Provides biofeedback, enabling tangible and visible insights for clients and clinicians.	Lacks integration with mobile applications for real-time feedback in dynamic environments.
TekScan F-Socket	Paper-thin sensors can be	Expensive, limiting its

	tailored to fit various prosthetic designs.	accessibility to clinical or research settings.
	Provides detailed graphical displays and quantitative analysis of pressure distribution.	Focused solely on prosthetics and not applicable for dynamic seating environments or adaptive sports.
	Improves socket longevity and enhances user comfort.	Requires advanced expertise for setup and operation, making it less practical for everyday use by individual users.
Our Solution	Affordable	Comfortable, Bluetooth connectivity, up to 8 hours of consistent battery life, easy to understand.
	Real-time alerts	
	Mobile app integration	
	Designed for adaptive sports and dynamic use	

Figure 4.1.2 Comparison of Solutions

Our solution bridged the gap between existing systems by offering a **cost-effective, portable, and mobile-integrated pressure monitoring solution** tailored for wheelchair users and adaptive sports participants. Unlike the BPMS and F-Socket systems, which were either too specific or inaccessible for everyday use, our device prioritizes dynamic, real-world usability with a focus on affordability and inclusivity.

4.1.3 Technical Complexity

Our project demonstrated sufficient technical complexity through its multi-component design and the challenging requirements it addressed. The system incorporated distinct hardware and software subsystems, each requiring the application of various scientific, mathematical, and engineering principles. Furthermore, the problem scope involved multiple challenging requirements that aimed to match or exceed current

industry solutions in accessibility, usability, and effectiveness. Below, we provided a detailed breakdown of the technical complexity.

1. Multi-Component Design

Hardware Subsystem

a. Load Cell Sensors:

- Utilizes strain gauge principles to measure changes in electrical resistance caused by pressure. This requires precise calibration to ensure accurate and reliable readings across a wide range of user weights and seating conditions.
- Challenges include accounting for sensor drift and ensuring uniform data across multiple sensors integrated into the seating surface.

b. Raspberry Pi Pico W:

- Acts as the central processor for collecting and transmitting data via a Wi-Fi-based client-server model. This involves integrating hardware communication protocols and ensuring low-latency, reliable data transmission to the software subsystem.
- Requires programming in MicroPython to handle real-time data processing and transmission while maintaining system stability.

c. Physical Pressure Sensor Device:

- Designed using principles of mechanical engineering and material science to ensure durability, comfort, and optimal placement of sensors. The thin, flexible device must withstand prolonged use while maintaining accuracy in pressure detection.

Software Subsystem

a. Mobile Application:

- Developed using Android Studio, the app uses user interface (UI) design principles to present real-time pressure distribution as a heatmap or graph.

- Challenges include ensuring smooth data visualization with minimal latency and developing a user-friendly interface accessible to individuals with varying technical proficiency.
- b. Threshold Detection Algorithm:**
- Applies mathematical models to analyze real-time pressure data, detecting when pressure levels exceed safe thresholds. This involves setting dynamic thresholds based on user profiles and optimizing the algorithm to reduce false positives.
 - Challenges include defining safe thresholds for pressure sores without direct access to clinical expertise, which adds complexity to the algorithm design.
-

2. Problem Scope and Challenging Requirements

Our project addressed several challenging requirements that match or exceed the current industry standards:

a. Real-Time Monitoring:

The system provided real-time updates with minimal latency, a key requirement for adaptive sports participants. Achieving this required optimizing both hardware (sensor responsiveness) and software (efficient data processing).

b. Portability and Comfort:

Unlike existing solutions that are bulky or limited to clinical environments, our system was designed for dynamic, real-world use. This involved engineering a lightweight and unobtrusive device while maintaining accuracy and durability.

c. Affordability

One of the most challenging requirements was ensuring the system remained cost-effective for individuals with disabilities. This necessitated careful selection of components and efficient system design to minimize production costs without compromising functionality.

d. Bluetooth Integration:

Implementing a Bluetooth-based client-server model for seamless data transmission between the hardware and app involved challenges such as

maintaining connection stability, optimizing power consumption, and ensuring scalability for future Bluetooth integration.

3. Scientific, Mathematical, and Engineering Principles Applied

- a. **Scientific Principles:** Strain gauge mechanics (load cells), material science (device design), and wireless communication protocols (Wi-Fi).
- b. **Mathematical Models:** Signal processing for pressure data, real-time algorithm optimization, and statistical analysis for calibrating thresholds.
- c. **Engineering Applications:** Embedded system programming (MicroPython for Pico W), mobile app development (Java/Kotlin for Android Studio), and mechanical design for sensor housing and device durability.

4.2 DESIGN EXPLORATION

4.2.1 Design Decisions

a. Sensor Type and Configuration

Choosing the type and configuration of sensors was critical, as it directly impacted the accuracy and responsiveness of the pressure readings. In this project, load cell arrays have been selected for its ability to detect small changes in pressure across different contact points and have a relevant range weight up to 110 pounds for every load cell. This decision ensured that users receive timely alerts for any potentially harmful pressure levels, making it central to the effectiveness and efficiency of the device.

b. Data Transmission Method

The decision to use Bluetooth for data transmission (instead of WiFi) was based on considerations of energy efficiency, ease of use, and compatibility with mobile devices. Bluetooth's lower energy consumption made it ideal for wearable devices, extending battery life and reducing maintenance costs for users. Additionally, Bluetooth connectivity offered a simpler user experience for pairing the patch with a mobile app, ensuring the device remained convenient and user-friendly. This is


attributed to a very successful project, as communicating over Bluetooth allowed the device to not need to be connected to the Internet. It made testing easier, allowing us to connect to a server to transmit data, instead of being forced on a Wi-Fi signal with no connection to the Internet.

c. Mobile App Functionality and Interface

Deciding on an intuitive and accessible interface design for the mobile app was essential. The app's dashboard displayed real-time pressure data and sent alerts, ensuring that users can quickly interpret data and take corrective action if needed. This focus on user-friendly design helped ensure that the end product met the needs of adaptive sports athletes and others at risk of pressure sores, increasing the likelihood of adoption and successful use. We made this decision because it allowed users to see the data as it comes in, so we were not just relying on an alert to identify the need for a posture change.

4.2.2 Ideation

For the decision on **Sensor Type and Configuration**, we used **experimental prototyping** or hands-on comparative analysis techniques to explore potential sensor options. Below are the five options considered:

Potential Sensor Options	Explanation
<p>1. Load Cells</p>  <p><i>Figure 4.2.2.1 Load cells</i></p>	<p>Selected for their high load capacity of up to 110 pounds, which supports the range of body weights encountered in the application. Each load cell is small (80 g) and compact, making it ideal for a wearable device. Often used in digital bathroom scales, this sensor is a half-bridge strain gauge, providing the</p>

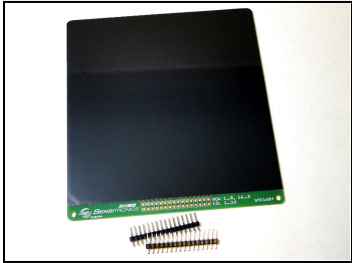
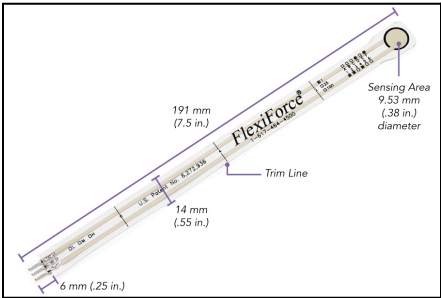
	precision required for monitoring body pressure effectively.
<p>2. ShuntMode MatrixArray</p>  <p><i>Figure 4.2.2.2 ShuntMode MatrixArray</i></p>	<p>A rigid sensor array with 256 force-sensing cells arranged in a 16x16 grid, covering a 4x4 active area. Each cell can measure up to 50 lbs, making it suitable for multi-touch input and pressure mapping. However, its lack of flexibility and limited force range presented challenges for continuous body pressure sensing. This array requires additional soldering to connect to PCBs.</p>
<p>3. FlexiForce A201 Sensor</p>  <p><i>Figure 4.2.2.3 FlexiForce A201 Sensor</i></p>	<p>This sensor offers various lengths and a thin profile (0.203 mm) but has a small sensing area (9.7 mm diameter) that restricts its ability to cover larger surfaces. It has a standard force range of up to 445 N and operates in a broad temperature range (-40°C to 60°C), which is advantageous in various environmental conditions. However, its limited surface area was less practical for widespread skin contact monitoring, moreover, for the adaptive sports athletes.</p>
<p>4. Thin Film Pressure Sensor</p>	<p>An affordable option with a maximum force capacity of 10 kg. This ultra-thin and</p>



Figure 4.2.2.4 Thin Film Pressure Sensor

flexible sensor responds rapidly to pressure changes and is both waterproof and sensitive. Although it offers high sensitivity and is well-suited for applications like wearables, its low maximum force limit makes it less suitable for the project's needs.

5. Thin Film Force Sensor (SF15-600)



Figure 4.2.2.5 Thin Film Force Sensor

This flexible sensor emulates the perception capabilities of human skin, offering high sensitivity, quick response, and durability. It is resistant to bending and temperature fluctuations (-20°C to 60°C) and provides reliable performance. Despite its low price, its 10 kg limit was insufficient to support the required pressure ranges in this project.

4.2.3 Decision-Making and Trade-Off

To evaluate the different sensor options, we used a **weighted decision matrix** that considered key factors relevant to the project, including **load capacity**, **size and flexibility**, **power efficiency**, **cost**, and **durability**. Each criterion was weighted according to its importance to the project's success. Here is a breakdown of the decision-making process, followed by an explanation of why the load cell was ultimately chosen.

Weighted Decision Matrix

Sensor Option	Load Range (30%)	Size & Flexibility (25%)	Power Efficiency (15%)	Cost (15%)	Durability (15%)	Total Score
Load Cells	9	8	8	8	9	8.45
ShutMode Matrix Array	6	4	7	5	7	5.85
FlexiForce A201 Sensor	7	5	9	5	8	6.65
Thin Film Force Sensor	4	9	10	9	6	6.80
Thin Film Pressure Sensor	5	9	10	9	6	7.05

Figure 4.2.3.1 Weighted Decision Matrix

Decision Criteria and Weights

1. Load Capacity (30%)

This is a critical criterion, as the sensor must be able to accommodate the pressure exerted by various body parts. Load cells scored highest here due to their wide range of up to 110 pounds, meeting the project's requirements.

2. Size & Flexibility (25%)

Flexibility and compactness are essential for user comfort and integration into a wearable device. Thin film sensors performed well in this category due to their ultra-thin design, while load cells also scored reasonably well for their compact dimensions.

3. Power Efficiency (15%)

Sensors must consume minimal power to extend the device's battery life. Thin film and FlexiForce sensors scored highly for low power consumption, but load cells also proved efficient enough for this application.

4. **Cost (15%)**

Staying within budget is essential, especially as we tested multiple sensors. Thin film sensors had an advantage here with low prices, while load cells were slightly more expensive but still reasonable.

5. **Durability (15%)**

Sensors need to withstand long-term use under various pressures. Load cells excelled in this category, as they are often used in scales and other applications requiring resilience.

Analysis and Choice

After scoring each option, Load Cells emerged as the highest-scoring sensor due to their broad load capacity, compact size, adequate power efficiency, and high durability. Although they were slightly more expensive than thin film options, their reliability in accurately measuring the necessary pressure range made them the best fit. The load cells provide stability and consistency, essential for ensuring that the device delivers accurate alerts to users, ultimately supporting the project's goal of effective pressure monitoring.

4.3 FINAL DESIGN

4.3.1 Overview

Our pressure sensor patch was specifically designed to support adaptive sports athletes with nerve damage who may not feel pressure buildup and are unaware when pressure sores begin to develop, particularly on the lower extremities. For these athletes, monitoring pressure points is crucial since they may remain unaware of damaging pressure

on certain areas due to impaired nerve function. The patch continuously tracks pressure levels and alerts the user if any area experiences excessive pressure for too long, allowing timely position adjustments to prevent sores.

This device is especially valuable for athletes in wheelchairs or those with limited mobility who frequently sit or place weight on specific parts of their lower body. By proactively alerting users when pressure thresholds are exceeded, the patch helps them prevent pressure sores before they become problematic, supporting a safer, more active lifestyle.

Key Components of the Pressure Sensor Patch

1. Load Cell Sensors

These small sensors detect the amount of pressure applied to different areas where the patch is placed. They act like the pressure-measuring technology in digital bathroom scales, allowing us to monitor the force exerted on each sensor. By using multiple load cells, we can cover a larger area and get detailed pressure readings from different body parts.

2. Microcontroller

The microcontroller is a small computer chip that processes the information from the load cell sensors. It takes in the pressure readings and checks if any of them exceed a safe threshold. If the pressure becomes too high, the microcontroller triggers a warning to alert the user.

3. Bluetooth Connectivity

This feature allows the device to send pressure data wirelessly to a mobile app. With Bluetooth, users can view real-time pressure data on their smartphone, which also alerts them when adjustments are needed.

4. Mobile App

The app provides a user-friendly interface that displays pressure readings from the sensors in real time. When excessive pressure is detected, the app sends an alert to the user, prompting them to adjust their position to relieve pressure. The app also logs pressure data over time, helping users and caregivers track patterns and identify areas that may need extra care.

How these Components Works Together

Our pressure sensor patch was specifically designed to support adaptive sports athletes with nerve damage who are unable to feel pressure and may not realize when pressure sores are developing, especially on the lower extremities. The device continuously monitors pressure levels through load cell sensors and sends this data to a microcontroller, which processes the readings and checks for any areas where pressure exceeds a safe threshold.

When excessive pressure is detected, the microcontroller initiates a response: it communicates with the app via Bluetooth to send an alert to the user's phone, and the mat itself produces a vibration or sound, giving an immediate and localized alert. This dual alert system—visual feedback from the app and sensory feedback from the mat—ensures that the user can quickly identify and relieve pressure on vulnerable areas, even without constant app monitoring.

This integrated design promotes user comfort and safety by offering real-time alerts and logging pressure data to track trends over time, helping prevent pressure sores before they develop. Lightweight, portable, and easy to use, the device is a practical tool that fits into an active lifestyle, providing peace of mind for athletes and their caregivers.

4.3.2 Detailed Design and Visual(s)

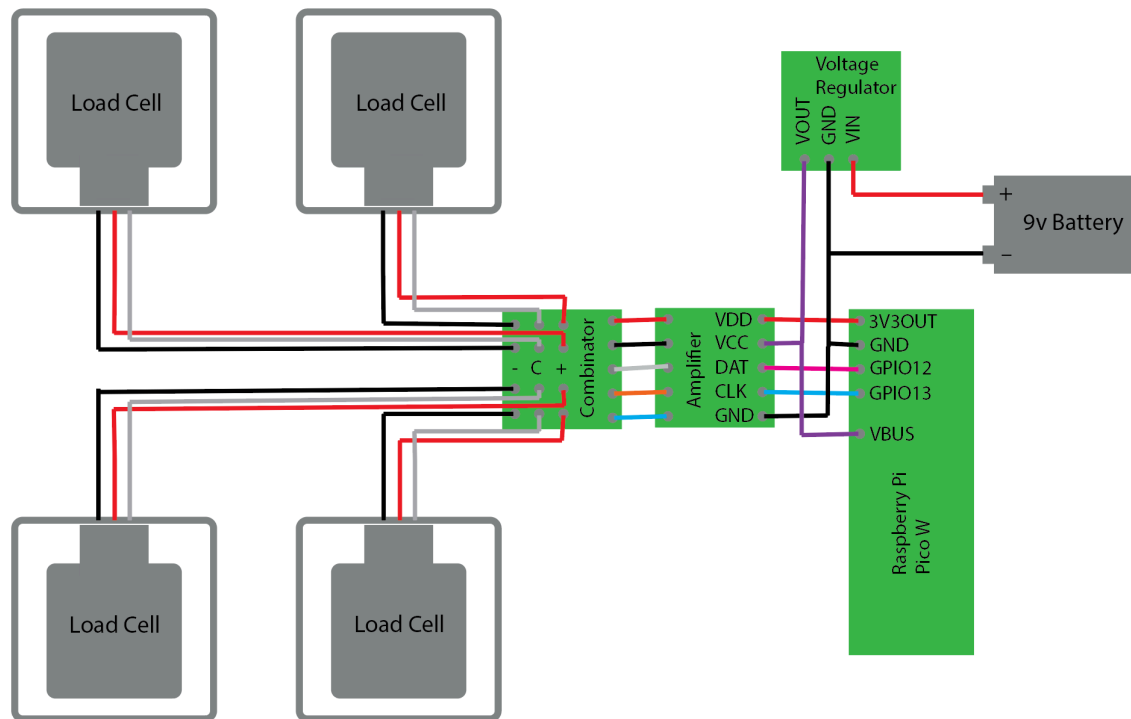


Figure 4.3.2.1 Hardware Design Schematic

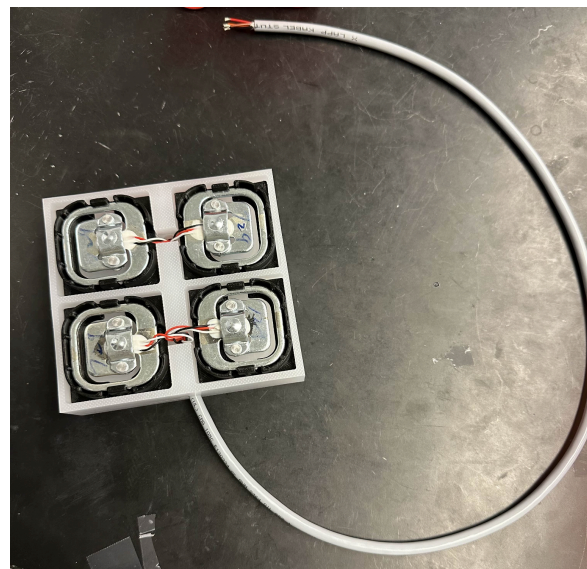
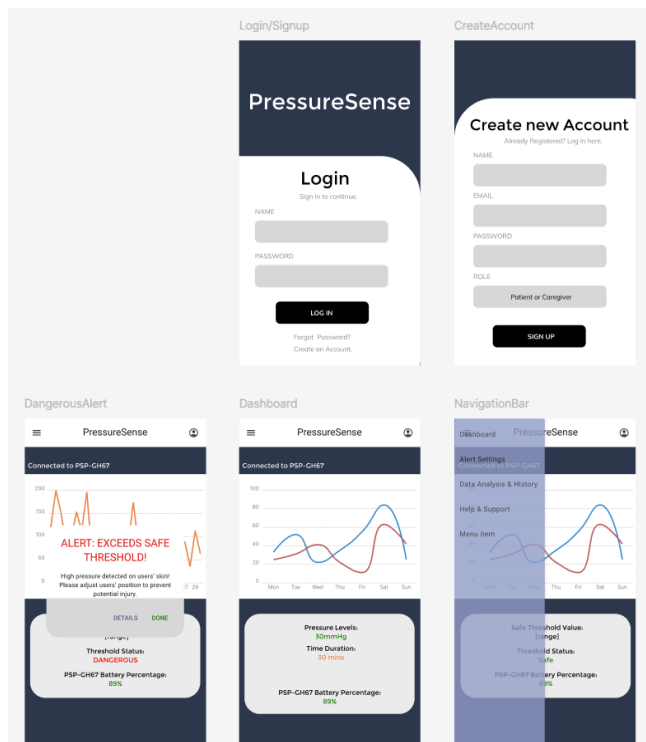


Figure 4.3.2.2 (left) and 4.3.2.3 (above)
Application Wireframe and Housing

Our design was split into two main components, the Android application and the pressure-sensing hardware. The hardware consists of several components that allow relevant data to be collected and transmitted. In Figure 4.3.2.1, a detailed diagram of the hardware components is shown. The silver objects in the corners are the load cells and they each use a mechanism known as a Strain Gauge. They are resistors in metallic z-shape strip form, shown in Figure 4.3.2.4 below.

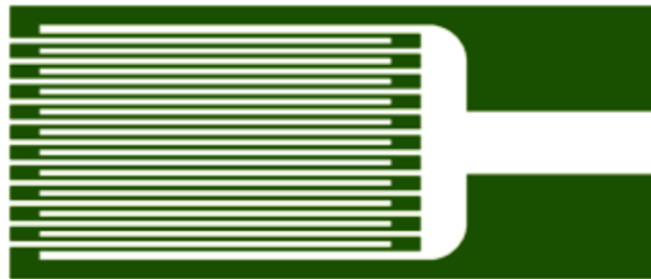


Figure 4.3.2.4 Strain Gauge

They are capable of detecting small variable changes in resistance as deformation known as strain. A single load cell does not provide accurate measurement of applied force but four strain gauges in a wheatstone bridge circuit configuration can provide accurate and reliable force measurement as shown in Figure 4.3.2.5 below.

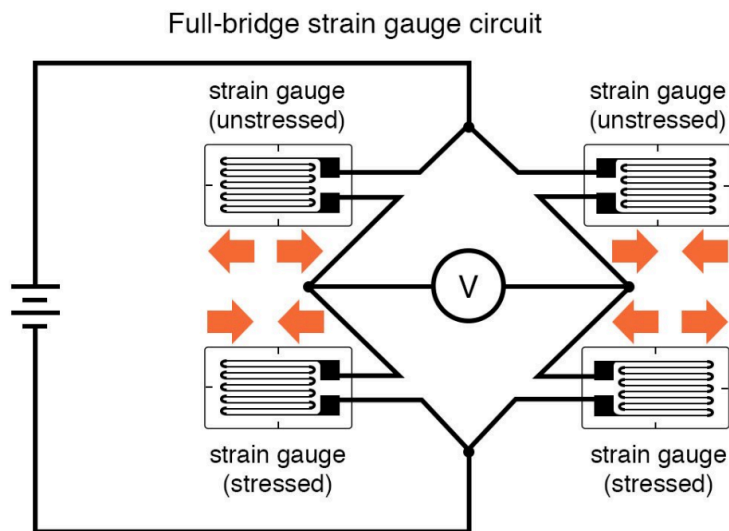


Figure 4.3.2.5 Strain Gauge Circuit

In this configuration, there is a voltage applied to the 4 strain gauges creating a balance of resistance that is then unbalanced when a force is applied introducing a change in resistance to the gauges that can then be read from the circuit to measure the force applied accurately. Since each load cell has only one strain gauge, a separate hardware component known as a combinator allows for four load cells to be directly connected as shown in figure 4.3.2.1 to achieve the desired wheatstone bridge circuit that is needed. The output signal coming out of the combinator from the load cells is too weak to be directly given to the microcontroller for usable input, so instead the combinator is wired to a load cell amplifier. The amplifier first uses its internal Analog to Digital converter to process the analog signals coming from the combinator and transform them into digital signals. Then the amplifier strengthens the digital signal to be adequate for the microcontroller to process. The amplifier is then wired to a microcontroller with wireless communication capabilities. The microcontroller will then wirelessly transmit the received signal to an Android application.

The Android application receives the signal wirelessly from the microcontroller in the hardware system. The Android application will be written in the Kotlin programming language. Kotlin is the google standard programming language for the development of native Android apps. For the UI we will be using Jetpack Compose. Jetpack Compose is a declarative UI design tool for Android applications written in Kotlin. Jetpack Compose is the modern Android UI framework and allows for faster development of complicated user interfaces. By building with native android we will be able to easily access hardware components of the device, such as the bluetooth adapter and the notification manager. The Android application includes a user interface similar to the wireframe shown in figure 4.3.2.2, which is designed to make the application easy and pleasant to use. Standard user features will be available such as a login and signup allowing the user to view previous data or incidents on different devices. The application also includes a real time heatmap, that when connected will show the weights of all currently active pressure sensors. This heatmap will allow users to see a potential imbalance in real time and adjust themselves accordingly. If the user does not notice the imbalance, a notification system will be implemented. The notification system uses streaming algorithms to watch for a potential

imbalance and ping the user if one is detected. This system runs in the background of the device, allowing the user to receive notifications even when they are not using their mobile device.

Another sub component of the system will be the housing. The load cells, combinator, and amplifiers will be placed into a secure housing as shown in figure 4.3.2.3, which will protect the circuits from damage as well as provide a flat surface for accurate readings from the load cells. These housings are meant to be placed under a seat where the user can still remain comfortable while gathering much needed information.

4.3.3 Functionality

Our pressure-sensing device was designed to assist individuals with disabilities in monitoring and managing pressure distribution while using adaptive equipment, such as wheelchairs or custom seating. The device operated seamlessly in real-world contexts, such as during prolonged sitting or participation in adaptive sports, and provided users with timely feedback to help prevent pressure-related injuries.

User Interaction and Workflow

1. Setup:

- The user begins by placing the pressure sensing device on their seating surface (e.g., wheelchair or sports chair). The device is lightweight and portable, designed for easy placement and removal.
- Once the device is positioned, the user turns on the system, which automatically connects to the mobile application via Bluetooth.

2. Monitoring:

- The device's embedded load cell sensors continuously measure pressure distribution across different areas. This data is processed by the Raspberry Pi Pico W, which acts as the system's central processor and server.

- The mobile app displays real-time pressure data as a heatmap or graph. This visual representation helps users understand which areas are experiencing higher levels of pressure.

3. Feedback:

- The system compares the measured pressure levels to predefined safe thresholds. When the pressure in a particular area exceeds these thresholds, the app triggers a notification to alert the user.
- The alert includes a message, vibration or sound, prompting the user to adjust their position or redistribute their weight to alleviate excessive pressure.

4. Data Tracking:

- The app stores pressure data over time, allowing users or caregivers to review historical trends. This feature helps in identifying recurring patterns or risk areas that may require further adjustments or interventions.

5. Adjustment and Actions:

- Based on the feedback, the user can take immediate action, such as shifting their weight, changing their seating position, or consulting with a caregiver for further assistance.

Real-World Situation Example

Imagine a wheelchair user seated during a prolonged sports activity. As the user moves and shifts their weight, the pressure sensing device continuously monitors changes in pressure distribution. At a specific point, the system detects excessive pressure on a vulnerable area, such as the lower back. The mobile app sends a notification, prompting the user to adjust their position. The user follows the prompt, redistributes their weight, and continues participating in the activity safely, reducing the risk of developing a pressure sore.

4.3.4 Areas of Challenge

In our design we faced challenges related to powering the device. We needed to provide adequate power to each of the load cells or they would not read correctly and the microcontroller we chose could not provide enough power for four sets of load cells. To address this we introduced a new disconnected power circuit allowing us to provide power to the load cells without relying on the microcontroller to power them. Another challenge we faced was getting the sensors to output data quickly. The library that we initially used did not provide enough reading per second to meet our requirements of real time data readings. To address this we created our own software package, allowing us to push the device to hardware maximum and get the fastest possible readings.

4.4 TECHNOLOGY CONSIDERATIONS

We opted to use strain gages as our weight sensing hardware device. Strain gauges require a flat surface to operate effectively and the values must be converted and calibrated to be relevant to our application. However strain gages have a very high maximum load, making them ideal for our application.

On the software side we opted to develop in native Android as opposed to a multiplatform framework. Native android has better support for interacting with the phones hardware, making it ideal for our bluetooth and notification reliant application. Native android however is difficult to develop in and has a steep learning curve. It also prevents us from creating an IOS application without rewriting the entire codebase.

5 Testing

Testing is a vital aspect of our project, ensuring that the pressure-sensing device met its functional, safety, and user experience requirements. Our testing strategy was designed to validate both hardware and software components individually and as an integrated system. Early and iterative testing played a central role in identifying and resolving issues throughout the development process. This proactive approach minimized risks and ensured that the final product is reliable and user-friendly.

Testing Philosophy and Challenges

Our testing philosophy emphasized a user-centered approach, focusing on ensuring the device's accuracy, responsiveness, and ease of use for individuals with disabilities. The iterative nature of our testing aligned with the Agile development process, enabling continuous refinement of the prototype. A unique challenge in testing our system was simulating realistic use cases, such as prolonged periods of sitting with varying pressure distributions. Additionally, ensuring real-time communication between the hardware and the mobile app adds complexity, particularly in maintaining low latency and reliable Bluetooth connections.

5.1 UNIT TESTING

The units being tested were divided between the hardware components and the software components. On the software side, we tested all individual methods within the frontend Android UI code and all individual methods within the backend server code. These methods were tested using modern unit testing frameworks such as Espresso Mockito and JUnit.

On the hardware side, we have 2 key components that were tested, the load cell weight sensors and the Bluetooth communication module. Both of these components were tested separately using an external device. Outputs were written to a console and verified manually to ensure the components were functioning as expected. A program like Putty was used to test network communication and verify that the hardware can communicate with external devices.

5.2 INTERFACE TESTING

Software Interface:

Our software interface included an Android application that connected to our pressure sensor device. To test our app, we utilized unit tests, along with simulation

testing through Android Studio. We also tested our wireless connection with a physical Android phone.

Hardware Interface:

Our hardware interface included our pressure sensor device. To test our device, we used weights to ensure that our load cells are reading weight accurately. We also tested that our device is sending data correctly and timely by printing the output to a console and analyzing the data.

5.3 INTEGRATION TESTING

The critical integration path included reading data from load cells, obtaining a Bluetooth connection to send the data from the load cells, and analyzing and displaying the data in an app. To test the load cells reading data, we wrote a program to send the data output to a console. To test the Bluetooth connection from the sensor to the app, we started out with sending the Picos internal temperature as a baseline. We then ported this code to work with our pressure sensors once the testing was completed.

5.4 SYSTEM TESTING

To test the entire system, we had multiple individuals sit on the device and go through various seating positions to collect generous amounts of data. We stored this data in Apache Kafka, and sent it into a Machine Learning Regression Model to help identify a threshold to send an alert.

5.5 REGRESSION TESTING

To ensure that any new additions did not break the old functionality, we utilized our current unit tests, along with creating new unit tests to cover the changes. We also tested individual sections of the system at a time so that if there was an issue, we would know where it was coming from.

5.6 ACCEPTANCE TESTING

To demonstrate that the design requirements, both functional and non-functional, were being met, we tested our application and pressure sensor device with real users and

collected and analyzed data to ensure that the system worked as expected. Through this testing, we found that despite having the device on a soft chair with a foam pad on top, the pressure sensors were still able to read data accurately.

5.7 USER TESTING

We conducted our testing with various users, including ourselves. This gave us a wide range of characteristics such as age, weight, and height. One major component of our user testing revealed that comfortability is a major factor. Users want to be able to sit on the device without noticing it is there. We had to alter our design to make our sensors be placed further away from each other, increasing the surface area of the device, which helped to improve comfortability.

5.9 RESULTS

In testing we found that our device accurately read values from a user sitting on it. The device was capable of sending accurate data in real time to the Android application where it can be viewed by the user in an easy and intuitive to read heatmap. The application also functioned smoothly even when sending data in real time to a web server. The coloring of the heatmap accurately lined up with the pressure applied onto the device, and accurately accounted for the total weight put on the device by adjusting its scaling factors based on the average weight applied. Through our observed testing we found that the device can be used as an effective tool to correct sitting posture, and prevent pressure sores as a result of improper posture.

6 Implementation

We have designed and constructed a fully functional pressure sensor device that accurately detects weight through four separate tiles. These values were then amplified and transmitted via a Bluetooth Low Energy signal to the Android Application. The android app uses this data and displays it on a Heat-Map style graph, using a range of colors between red and green to indicate a certain tile that may have too much pressure on it, and the user needs to shift their position.

The device was built in a sturdy plastic frame, with four tiles that hold the sensors in each corner. The center of the device houses the amplifier and the Raspberry Pico W. The device can fit comfortably on a majority of chairs. A soft foam pad is recommended to be placed on top of the device for increased comfort.



Figure 6.1 Final Prototype of Pressure Sensor Device

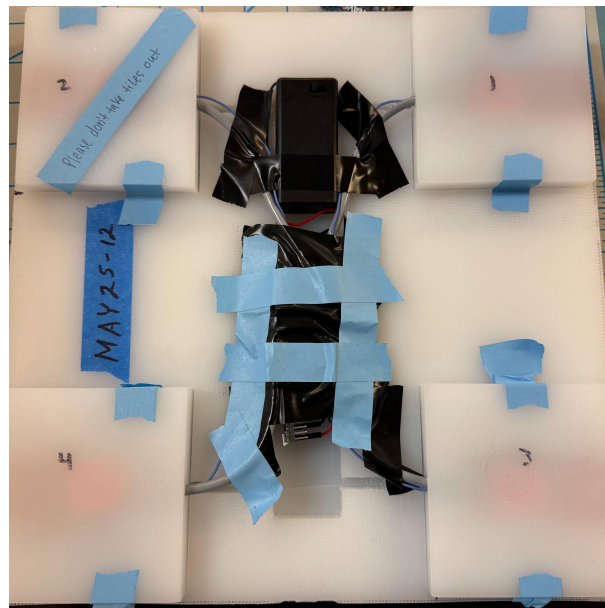


Figure 6.2 Bottom View of Sensor Device

When the user opens the app, they are presented with an empty heat map and an indication that the Bluetooth device is disconnected. The user can select the Pico from the device list, and a live connection will be made. The heat map then updates live as the values flow in from the sensors.

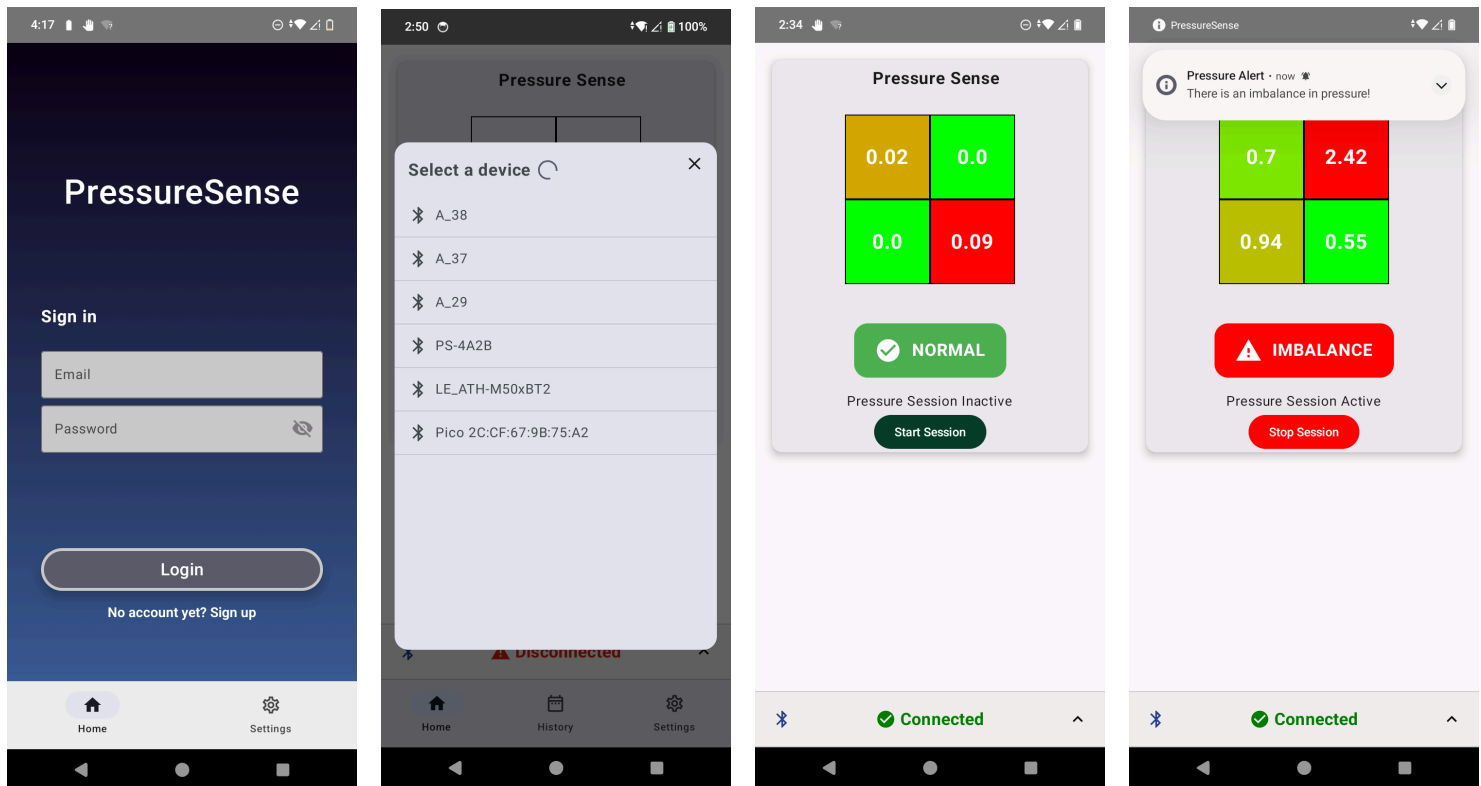


Figure 6.3 Final Prototype of Android Application

6.1 DESIGN ANALYSIS

Our design worked as expected. As we did not care about the specific weight values that were being sensed, but rather the relative difference between them, placing weight sensors on a softer surface still worked very well. Our communication was also very fast, we received data over Bluetooth in a very quick manner which allows us to notify the user efficiently if a potential pressure sore is developing. We also had very quick data calls over the Wi-Fi network, which has improved the speed of our testing. One design aspect that felt short was the thickness of the device. Our device was thicker than initially planned, but still thin enough to remain comfortable. We could have potentially spent more time researching thinner solutions, if that is possible.

7 Ethics and Professional Responsibility

7.1 AREAS OF PROFESSIONAL RESPONSIBILITY/CODES OF ETHICS

Areas of Professional Responsibility

Area of Responsibility [8]	Definition	Relevant Item from Code of Ethics [9]	Team Interaction and Adherence
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence	IEEE Code of Ethics #6: "Improve the understanding of technology and its application."	The team has consistently pursued skill development, including learning Kotlin and sensor integration techniques, to ensure technical competence.
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	IEEE Code of Ethics #9: "Avoid injuring others, their property, reputation, or employment by false or malicious action."	The team has adhered to budget constraints by choosing cost-effective components, balancing affordability with functionality.
Communication Honesty	Reports work truthfully, without deception, and are understandable to stakeholders.	IEEE Code of Ethics #3: "Be honest and realistic in stating claims or estimates based on available data."	The team has provided regular updates to advisors and clients through progress reports and meetings, ensuring transparency.
Health, Safety, and Welfare	Minimize risks to safety, health, and well-being of stakeholders.	IEEE Code of Ethics #1: "Hold paramount the safety, health, and welfare of the public."	The team has designed the device to prioritize the prevention of pressure sores, reducing health risks for users.
Property Ownership	Respecting the property, ideas, and information of clients and other stakeholders.	IEEE Code of Ethics #7: "Treat all persons fairly and with respect, and to not engage in discrimination based on characteristics such as race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression"	The team ensures the protection of client-provided data, design requirements, and intellectual property by using secure project repositories and agreements.
Sustainability	Protect environment	IEEE Code of Ethics #1	The team has incorporated

	and natural resources locally and globally	and #2: "Improve the understanding of technology's potential impact on society and the environment."	eco-friendly materials and rechargeable batteries to reduce waste and environmental harm.
Social Responsibility	Produce products and services that benefit society and communities.	IEEE Code of Ethics #4: "Avoid unlawful conduct in professional activities, and to reject bribery in all its forms."	The project is designed to enhance inclusivity, ensuring individuals with disabilities can participate safely in adaptive sports.

Figure 7.1.1 Professional Responsibility Table

Team Performance Analysis

Area of Strong Performance: Health, Safety, and Welfare

The team excelled in prioritizing the safety and well-being of the end users. By designing the device to monitor and prevent pressure sores, the project directly addresses a critical health concern for individuals with disabilities. Additionally, the team implemented thorough testing protocols to ensure the reliability and safety of the device under various conditions. This focus on user safety demonstrates a commitment to adhering to the principle of holding paramount the health and welfare of the public, as outlined in the IEEE Code of Ethics.

Area for Improvement: Communication Honesty

While the team maintained transparency in progress reports and regular updates, there have been occasional challenges in clearly articulating technical limitations and timelines to stakeholders. For example, delays in hardware testing due to unexpected integration challenges were not communicated immediately, leading to misaligned expectations with the advisors. To improve, the team planned to enhance communication practices by implementing a standardized reporting template that included a detailed status update, identified challenges, and proposed resolutions. Additionally, scheduling

regular check-ins with stakeholders to align on progress and expectations ensured more effective and honest communication moving forward.

7.2 FOUR PRINCIPLES

Broader Context Area [10]	Beneficence	Nonmaleficence	Respect for Autonomy	Justice
Public Health, Safety, and Welfare	The pressure-sensing system helps reduce the risk of pressure sores, improving overall health and well-being.	The device aims to prevent the harm caused by pressure sores, which can lead to serious health complications.	Provides individuals with disabilities the ability to take proactive control over their health by receiving alerts.	Ensures equitable access to health tools for those who need them, including individuals with limited resources.
Global, Cultural, and Social	Promotes inclusivity by enabling disabled individuals to participate in adaptive sports safely.	By preventing injuries, the product ensures that users can engage in activities without long-term harm.	Empowers users to decide when and how to take action regarding their health based on the real-time data.	Ensures that adaptive sports can be accessed by individuals with disabilities, supporting equal opportunities.
Environmental	The use of eco-friendly materials ensures the design benefits the environment by reducing waste.	The design minimizes harmful environmental impacts by opting for sustainable materials and energy-efficient design.	Allows users to make environmentally conscious decisions, such as reducing waste by using rechargeable batteries.	Strives to use sustainable practices, ensuring the product is affordable without compromising environmental values.
Economic	Offers a low-cost, accessible solution for those at risk of pressure sores, promoting financial well-being.	Minimizes long-term healthcare costs by reducing the need for expensive treatments for pressure-related injuries.	The product enables users to make decisions about their participation in sports based on their health needs.	Makes the technology available to diverse socioeconomic groups, addressing disparities in healthcare access.

Figure 7.2.1 Guiding Principles Table

7.3 VIRTUES

Virtues our team upheld include:

Honesty:

Honesty as a virtue requires that all individuals are honest in what they say and do. Individuals should not mislead others, or be intentionally deceitful. Our team was honest with our clients in what was realistically achievable within the given budget and current state of technology. Our team was truthful in what we have accomplished, and what we expected to accomplish.

Commitment to Quality:

Commitment to quality dictates that an individual will strive to achieve a product that is at the highest quality they can achieve. Our group supported this virtue by striving to use state of the art technologies and frameworks to ensure our system was as modern as possible. We also ensured that our system functioned exactly as expected and eliminated any deficits that may have existed in the system when possible to do so.

Clear and Thorough Documentation:

Clear and thorough documentation dictates that an individual must document their process in a way that others can clearly understand what was done and how any system is meant to work. Our group supported this virtue by creating documentation surrounding our system that will allow others to repeat our steps to create a similar system.

Osaid Samman:

- One virtue I had demonstrated in my senior design work was cooperativeness.
 - Cooperativeness is crucial to me because it helps foster collaboration, ensures tasks are completed efficiently, and maintains harmony within the team. As the Scrum Master, it aligns perfectly with my responsibility to keep communication open and productive among team members, helping everyone work together effectively toward our goals.

- I demonstrated cooperativeness by facilitating clear communication between team members, organizing regular Scrum meetings, and ensuring that everyone was aligned on our project goals and deadlines. By actively listening to feedback and addressing concerns, I created an environment where everyone felt heard and motivated to contribute their best.
- One virtue that is important to me that I have not demonstrated in my senior design work thus far is civic-mindedness.
 - Civic-mindedness is important to me because it involves understanding and contributing to the broader context of our work, including technical details and their potential societal impact. By being more aware of the technicalities my team members worked on, I could have better supported their efforts and ensured our project aligned with both technical excellence and societal needs.
 - To demonstrate civic-mindedness, I can take a more proactive approach to understanding the technical aspects of our project.

Ivan Alvarado-Santoy:

- A virtue that I have demonstrated thus far through my senior design project is kindness.
 - This is important to me because without kindness, it is difficult to build a strong team relationship that is needed for a successful project and worthwhile experience.
 - I have demonstrated kindness by encouraging my teammates to share their ideas and concerns.
- One virtue that I hold as important but have not demonstrated in this project is reliability.
 - It is important to me because without being a reliable teammate, it is difficult to work effectively as a team and achieve the goals that we have set. I would not want my teammates to have doubt on me making our team meetings when I say I am going to be there but am not or do not communicate the progress of my work
 - I will demonstrate this virtue by prioritizing my team meeting times in my future schedule and planning accordingly with the team on when I can not attend meetings but make sure they are aware of the status of my work so as to not leave them in the dark of my progress.

Sabrina Francis:

- One virtue that I have demonstrated through my senior design work thus far is honesty.
 - This is very important to me because being dishonest can lead to many problems down the line.
 - I have demonstrated honesty by always being honest with our client and my team members and being transparent about project status, budget, and all other project related subjects.
- One virtue that I have not yet demonstrated through this project is clear and thorough documentation.
 - This virtue is important to me because having detailed documentation makes it much easier for both yourself and others to continue working on a project later on. It is also useful for when revisions or changes are being made to the design, so you can look back at previous documentation.
 - To demonstrate this virtue in the future, I could make an organized system for adding documentation on design details and changes to ensure all useful information about our project is being documented.

Zane Lenz:

- A virtue that I believe I have demonstrated is reliability.
 - Reliability is important to me because it brings me peace of mind to know that other people will get their tasks done, and I would be a hypocrite if I didn't try to hold myself to the same standard.
 - I believe I have shown it throughout the semester by attending nearly all meetings and lectures. In addition to that I think I have done a good job contributing to group assignments and getting my own project tasks done in a timely manner.
- A virtue I have been lacking in this semester has been communication.
 - It is vital to any project that everyone is on the same page and communication is the key to making that happen. With good communication problems can be avoided and better ideas could be implemented.
 - I think I could speak up a bit more in group meetings and contribute more to conversations with clients and advisors.

Bilal Hodzic:

- One virtue I have demonstrated thus far is commitment to quality.
 - This virtue is important to me as I believe that everyone should strive to produce the best product possible and should not become complacent when developing systems.
 - I have demonstrated this virtue by striving to ensure our codebase will be warning and error free, while also ensuring that we follow industry standard conventions and principles when working with software.
- One virtue I have not demonstrated is clear and thorough documentation.
 - This virtue is important to me as I believe that all systems should be able to be worked on by anyone with a similar skill set at any point in time.
 - To demonstrate this virtue I will strive to write accurate documentation within the code base, such as API documentation and comments throughout the codebase.

Nathan Turnis:

- One virtue I have demonstrated thus far is commitment to design.
 - This virtue is important to me because proper design helps to make a successful project. Designing an application where it is easy-to-use by the user is important.
 - To demonstrate this virtue I will create designs that are user-friendly and easy to understand. I need to design code that has good readability and reusability. Using common design principles and patterns instead of throwing things into one file.
- One virtue I have not demonstrated is clear communication.
 - This virtue is important to me because when there is effective communication, group members are knowledgeable on what I have been working on. I have not always been clear with what I have done or will do.
 - To demonstrate this virtue, I plan on updating the team more through our communication channels on what I have done so far and what I plan to do in the future. I plan on asking for constructive feedback which I can apply to my code.

Aina Azman:

- One virtue I have demonstrated in my senior design work thus far is clear and thorough documentation.
 - Clear and thorough documentation is important to me because it ensures that the process is transparent and easily understood by others. It helps maintain continuity in the project, and it ensures that the design, system and decisions are well-documented for future reference.
 - I have demonstrated this virtue by ensuring that all components of our project are well-documented, including the design decisions, code, and system requirements. I have also helped establish initial documents for documentation that allows all team members to access and contribute to the project details.
- One virtue I have not demonstrated in my senior design work thus far is clear communication.
 - Communication is critical because it ensures that the team is aligned, progress is tracked, and all team members are aware of any challenges or changes. It helps build a collaborative environment and reduces misunderstandings. Without clear communication, it's difficult to stay informed about the overall project status and can lead to missed opportunities for improvement or collaboration.
 - To improve communication, I plan to make a conscious effort to engage more frequently with the team, ask for updates, and share my progress openly. I will work on being more proactive in discussions, making sure I stay informed about what's happening with the project, and contributing regularly in meetings.

8 Conclusions

8.1 SUMMARY OF PROGRESS

As a team we have come close to a completed product that effectively addresses the problem. On the software side we have developed a user friendly mobile application that effectively connects to our hardware and shows live data to the user. Our application was designed to show the user visually where they have an imbalance in pressure, allowing them to intuitively adjust with visual feedback. The app was also set up to collect data,

which was then used to further our efforts to automatically detect at-risk scenarios and alert the user. On the hardware side, we successfully made a functional prototype in the form of a pressure pad. The pressure pad reads weight values in real time accurately and transmits data to the application via a Bluetooth connection. Overall, with our efforts towards software and hardware, we have effectively addressed all of our project's goals. We have successfully monitored pressure on at-risk areas for a user. We have communicated this data visually to the user. We have also developed a system that will alert the user when they are at risk. Overall, as a team, we have successfully accomplished our goals and have developed a prototype that meets the client's needs effectively.

8.2 VALUE PROVIDED

Our design met our users' needs well. Our users are relying on external devices to feel imbalances for them and our device shows the user where pressure is applied in an easily digestible manner. This addressed the problem that disabled individuals have of not being able to identify when they are sitting with incorrect posture. Our project addresses a larger problem within the disabled community as during our research we found that no such device currently exists. Pressure sores are a large problem and can be a roadblock for disabled individuals looking to pursue adaptive sports. Our project design gave them an easy-to-use way to monitor pressure on sit bones and actively prevent pressure sores. Our device was easy to set up, simply requiring a flick of an on switch to immediately start reading and sending data. Our app design was intuitive to use, making it simple for users to connect to the pressure sensing device and showing the data in an easy to understand way. Our usage of a heat map gave the user a real visual of where they are applying pressure even when they cannot feel it, which can then give them an intuitive way to adjust and see if their adjustment was adequate. This real time feedback for the user is what gave our design merit as a tool for preventing pressure sores and provided real value to disabled users, in the form of an easy to understand replacement for what many able bodied people are able to feel naturally.

8.3 NEXT STEPS

We would like to see future teams polish our design and make it into a real product that can be easily set up and used. We believe that the prototype is complete and is ready for the next stage which would be working to create an easily manufacturable design. This step is important as without ease of production, the product will not be able to help the masses, and will be limited to strictly testing environments. There are also significant improvements that can be made on the software application. The app is currently designed to display data effectively but it can also be made to appear more visually appealing. There are many quality-of-life features that could be beneficial to implement within the software application, such as offline support or customizable thresholds. These features would give the device more use cases and would make it more applicable to a wide range of adaptive scenarios.

9 References

[1] ISO/IEC/IEEE International Standard - Software and Systems Engineering - Software Testing -- Part 2: Test Processes, ISO/IEC/IEEE 29119-2:2013, 2013.

[2] ISO/IEC/IEEE International Standard - Systems and Software Engineering-Life Cycle Management-Part 5: Software Development Planning, ISO/IEC/IEEE 24748-5:2017, 2017.

[3] IEEE 802 Nendica Report: Flexible Factory IoT, IEEE 802 Nendica, 2020.

[4] ISO/IEC/IEEE International Standard - Systems and Software Integration, ISO/IEC/IEEE 24765:2017, 2017.

10 Appendices

APPENDIX 1 – OPERATION MANUAL

Iowa State University

2520 Osborn Dr,

Ames, IA 50011

sdmay25-12@iastate.edu

Pressure Sensor System

User Manual

OVERVIEW

People who have experienced significant damage to the lower extremities or suffer from disabilities affecting the lower extremities often are unable to feel when they are sitting with incorrect posture. Sitting with incorrect posture for long periods can develop pressure sores on the sit bones, which can be life-changing or even fatal in extreme cases. We have created a pressure-sensing device alongside an app that can alert users when they may be at risk of developing a pressure sore.

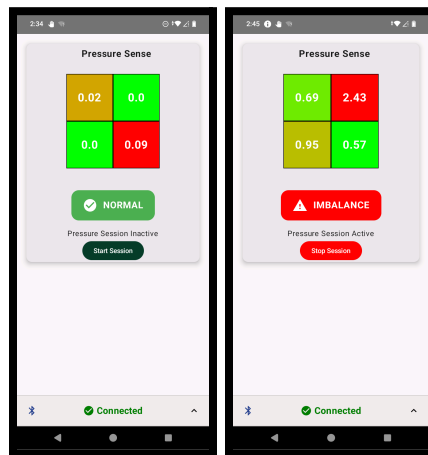
GOALS

1. Detect and alert the user when they may be at risk of forming a pressure sore based on device pressure readings and user information
2. Display a real-time visual representation of pressure distribution across the user's seat

SYSTEM COMPONENTS

Android Application

This system includes an Android application called **PressureSense** that connects to the pressure sensor device over Bluetooth and shows real-time pressure distribution across the seat. The application also detects and alerts the user through a notification if they are at risk of forming a pressure sore.



Pressure Sensor Device

This system also includes the pressure sensor device that the user places under their seat cushion, which sends constant pressure readings to the application.



Step 1 – Power On Device:

- Locate the power switch (**Figure 1**)

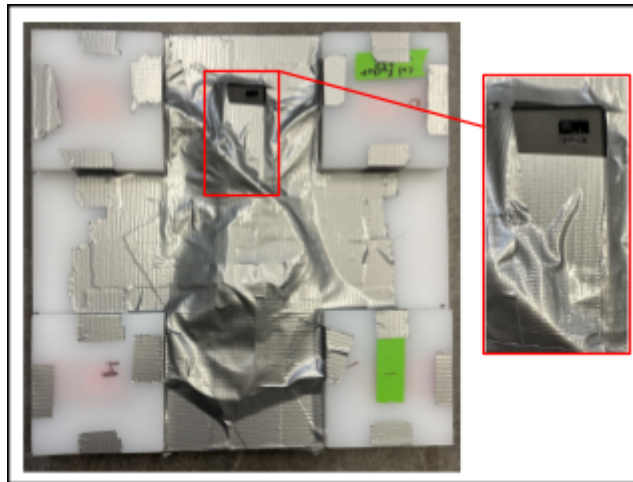


Figure 1: Power Switch

- Place the device down on the chair with the switch side facing the seat (**Figure 2**)
 - This is to ensure that the device calibrated correctly when it is turned on



Figure 2: Device Switch Side Down on Seat

- Flip the switch from **OFF** to **ON**

Step 2 – Position the Device:

- Locate “**FRONT**” label on the top of the device surface (**Figure 3**)

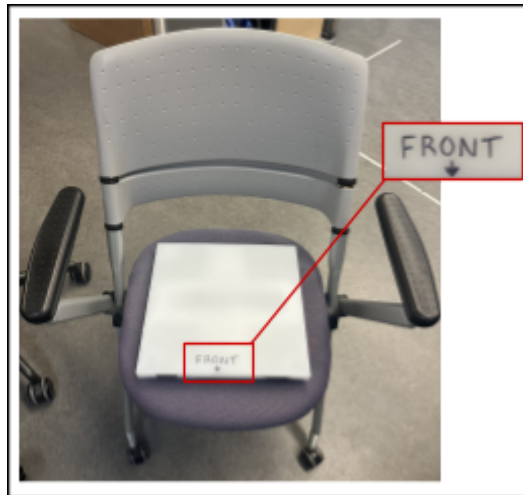


Figure 3: Device “FRONT” Label

- Place the “**FRONT**” side forward-facing in the chair (**Figure 4**)
 - Ensure that the device is centered in the seat and lying flat



Figure 4: Correctly Facing Device

- Set personal cushion on top of the device and ensure that it is centered
(Figure 5)



Figure 5: Centered Device with Cushion on Top

Step 3 – Launch Mobile Application and Sign Up:

*Before following the steps below, ensure that the provided phone is connected to WiFi

- Locate and open the PressureSense application on the provided Android phone (**Figure 6**)



***Figure 6:** PressureSense Application Icon*

- Once the app is launched, the **Sign Up** page should be present (**Figure 7**)

PressureSense
Create New Account

First Name

Last Name

Email

Password

Confirm Password

Role

Height (cm)

Weight (kg)

Home Settings

Figure 7: User Sign Up Page

- To sign up, fill in the necessary user information fields (**Figure 8**)

First Name
John

Last Name
Doe

Email
johndoe@gmail.com

Password

Confirm Password

Main User

Height (in)
71

Weight (lb)
205

Sign Up!

Home Settings

Figure 8: Necessary User Information

- Once sign-up is successful, the app is now ready to be used

Step 4 – Connecting to the Device:

- Locate and click on the Bluetooth tab (**Figure 9**)
 - This will pull up the **select a device** tab

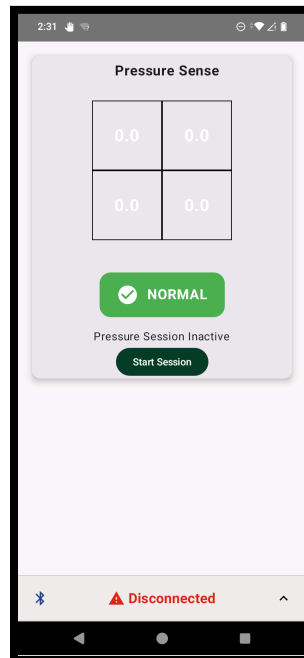


Figure 9: Bluetooth Tab

- Find the device with “**Pico**” at the start of the name and click on it to connect (**Figure 10**)

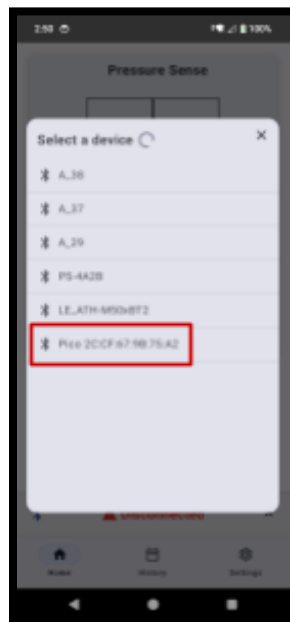


Figure 10: Pressure Device Bluetooth Name

- If the connection was successful, the Bluetooth tab should now say **“Connected”** (Figure 11)

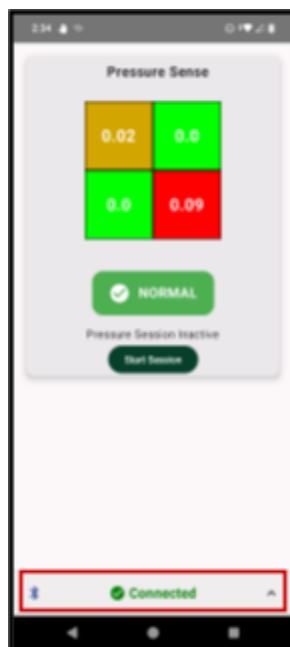


Figure 11: Successful Device Connection

Step 5 – How to Use the App:

***Before following the steps below, ensure that the user is sitting on the device**

- To begin a pressure sensing session, locate and click on the “**Start Session**” button (**Figure 12**)
 - This will begin data analysis of the pressure readings from the device to determine if a pressure sore risk is present

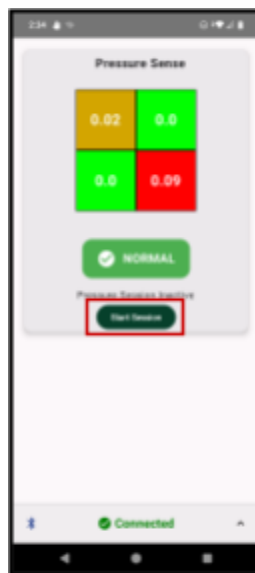


Figure 12: Start Session Button

- If a pressure sore risk is detected, the phone should send a notification with a **ping** sound (**Figure 13**)

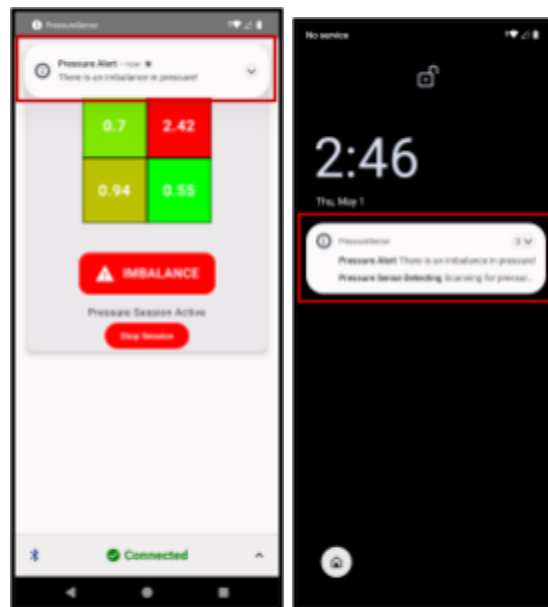


Figure 13: Pressure Sore Risk Notification Pop-Up

- The PressureSense app will also display a real-time visual representation of the pressure distribution with a heat map (**Figure 14**)
 - This will display the amount of pressure in each quadrant of the device, where **RED** represents a large amount of pressure, and **GREEN** represents a lower amount of pressure

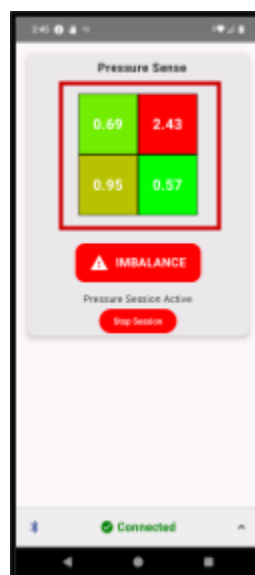


Figure 14: Pressure Heat Map

Notes

- Ensure that your smartphone remains within Bluetooth range of the Pressure Sensor Patch for uninterrupted connectivity.

APPENDIX 2 – ALTERNATIVE/INITIAL VERSION OF DESIGN

The initial idea specified a patch to be placed on the skin, similar to a band-aid. After more research was done, we came to the conclusion that the technology needed to accomplish this does not fully exist yet. Research studies are still being conducted for sensing pressure in this environment, however, this only pertains up to about 15 pounds. We pivoted to the next best solution, which is a device in which the user sits on. Our solution accomplishes the original goals outlined by Adaptive Adventures.

When designing our initial version, we had hoped to create a thinner solution to more match the idea of the patch. However, as we realized in order to meet the requirements of accuracy and comfortability, the decision was made to use more reliable sensors, which required a thicker device.

APPENDIX 3 – CODE

GitLab repository: <https://git.ece.iastate.edu/sd/sdmay25-12>

APPENDIX 4 – TEAM CONTRACT

Team Members

1. Aina Qistina Binti Azman
2. Bilal Hodzic
3. Nathan Turnis
4. Osaid Samman
5. Sabrina Francis
6. Zane Lenz
7. Ivan Alvarado-Santoy

Required Skill Sets for Your Project

- Knowledge of Android Development
- Soldering
- 3D Modeling, 3D Scanning, 3D Printing
- Networking

Skill Sets covered by the Team

- Knowledge of Android Development
 - Aina Qistina Binti Azman
 - Bilal Hodzic
 - Nathan Turnis
- Soldering
 - Ivan Alvarado-Santoy
- 3D Modeling, 3D Scanning, 3D Printing
 - Sabrina Francis
 - Zane Lenz
- Networking
 - Osaid Samman

Project Management Style Adopted by the team

A mix of waterfall and agile.

Individual Project Management Roles

Hardware Manager - Ivan Alvarado-Santoy

Software Manager - Bilal Hodzic

Team Contract

Team Members:

1. Aina Qistina Binti Azman
2. Bilal Hodzic
3. Nathan Turnis
4. Osaid Samman
5. Sabrina Francis
6. Zane Lenz
7. Ivan Alvarado-Santoy

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:

Weekly Team Meetings will be on Sundays in the TLA at 2:00 PM
 Weekly Software Meeting: Wednesdays 1-2 pm. Virtual or In Person
 Weekly Hardware Meeting: Thursdays 7-8 pm. In-person (TLA)
 Client Team Meetings: Twice a month. Virtual meetings.
 Faculty Advisor Meetings: Friday 11-12pm

2. **Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):**
 Discord for communication outside of meetings.
 Weekly in-person team meetings.
3. **Decision-making policy (e.g., consensus, majority vote):**
 Consensus, if possible; if not, then vote within 10-15 minutes.
4. **Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):**
 Discuss agendas, tasks, and what we've done during the Scrum Meetings, and it will be kept track of. Later, when tasks are more defined, we can have separate meetings between different groups.

Participation Expectations

1. **Expected individual attendance, punctuality, and participation at all team meetings:**
 Presence at weekly meetings is expected unless communicated ahead of time for being absent.
2. **Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:**
 Each team member is fully responsible for completing their assigned individual work by the designated deadline, and teamwork assignments are to be broken up as needed. Any concerns for not meeting deadlines or need for help should be communicated ahead of time.
3. **Expected level of communication with other team members:**
 Respond to Discord messages within the same day, otherwise communicate as needed.
4. **Expected level of commitment to team decisions and tasks:**
 All work must be completed by an agreed-upon deadline. If challenges/issues arise they must be communicated ahead of time to be resolved within a timely manner.

Leadership

1. **Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):**
 - 1) **Aina Qistina Binti Azman** - Software Developer
 - 2) **Bilal Hodzic** - Software Lead
 - 3) **Nathan Turnis** - Software Developer
 - 4) **Osaid Samman** - Scrum Master/Manager/Team Organization
 - 5) **Sabrina Francis** - Hardware Developer
 - 6) **Zane Lenz** - Hardware Developer
 - 7) **Ivan Alvarado-Santoy** - Hardware Lead
2. **Strategies for supporting and guiding the work of all team members:**
 - Proactively communicating any challenges that might occur so the team is aware and can lend support as needed.
 - Starting work early.
 - Do your best not to procrastinate.
3. **Strategies for recognizing the contributions of all team members:**

Task tracking to document those who completed the work.

Collaboration and Inclusion

1. **Describe the skills, expertise, and unique perspectives each team member brings to the team.**

Aina: Prior experience with web/application development, unit testing. Experience in Java, C, C++, HTML/CSS, JS.

Bilal: Prior experience with software web development. Experience in Java, C#, C, Python, HTML/CSS, JS.

Nathan: Prior experience in software web app development. Experience in Java, C#, Python, HTML/CSS, JS. Experience in Android frontend development.

Osaid: Prior experience with IT, Java, C, C#, HTML/CSS/JS, Embedded, Networking, Circuits.

Sabrina: Prior experience with embedded programming and security. Experience in debugging, unit testing, and simulation testing. Experience in C, C++, Java, Python.

Zane: Prior experience with web development, automation, and embedded systems programming. Java, C/C++, Python, HTML/CSS, JavaScript

Ivan: Prior experience with writing and debugging code for hardware, Testing, Integrating hardware, soldering. Good with Linux/Unix, C/C++, Containers, and 3D printing

2. **Strategies for encouraging and supporting contributions and ideas from all team members:**
 - Listen and don't shoot down ideas right away.
 - Scheduled weekly meetings discussing future and current work, plus extra meetings if anyone needs help.
3. **Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)**
 - Make sure everyone has work to do in weekly meetings.
 - Reach out if any collaboration issues are noticed.
 - Make sure everyone enjoys the work they are doing.

Goal-Setting, Planning, and Execution

- **Team goals for this semester:**
 - App wireframes.
 - Select hardware components.
- **Strategies for planning and assigning individual and team work:**
 - Plan and assign work during weekly scrum meetings.
 - Estimate deadlines and try to have work done by then, no hard deadlines.
- **Strategies for keeping on task:**
 - Breaking tasks down to manageable components.

Consequences for Not Adhering to Team Contract

1. **How will you handle infractions of any of the obligations of this team contract?**
Figure out what went wrong and try to help and fix it. Communication is key.
2. **What will your team do if the infractions continue?**
Communicate the situation to the instructor.

- a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*
- b) *I understand that I am obligated to abide by these terms and conditions.*
- c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) Osaid Samman

DATE 9/8/2024

- 2) Nathan Turnis
- 3) Aina Qistina Binti Azman
- 4) Sabrina Francis
- 5) Bilal Hodzic
- 6) Zane Lenz
- 7) Ivan Alvarado-Santoy

DATE 9/8/2024
DATE 9/8/2024
DATE 9/8/2024
DATE 9/8/2024
DATE 9/8/2024
DATE 9/8/2024