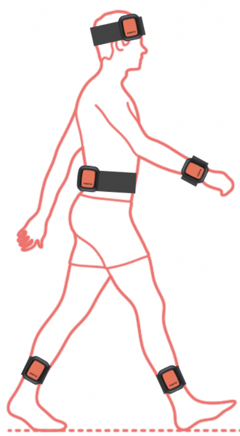
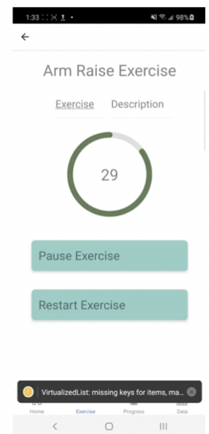
COLLEGE OF ENGINEERING
MECHANICAL ENGINEERING
UNIVERSITY OF MICHIGAN





Smartphone-Based Personalized Balance Training Platform

Final Report

ME450 Section 003 Team 8 - Winter 2021

Haiqal Azham, Aliah Sallehuddin, Xiyu Tian, Jia Chen Wee, Jia Le Yu

Sponsors: Christopher DiCesare, Xun Huan, Jamie Ferris, Safa Jabri

Section Instructor: Kathleen Sienko

Executive Summary

Older adults are at an increased risk for falls and they must undergo balance training exercises. Conventionally, they are done with a physical therapist (PT) in a clinic-based setting but this method of balance training has limitations in terms of cost, travel, insurance reimbursement policies, and PT availability - all of which can be partially or fully removed by having a home-based solution that can replicate as much of the clinical experience as possible. Current home-based physical rehabilitation solutions are limited in terms of their ability to provide data-driven, personalized exercise recommendations and performance assessments. They also cannot simultaneously measure kinematic data from more than one body part. Therefore, we aimed to develop a solution that can record movements of multiple body parts synchronously and to support features like a novel machine learning (ML) algorithm that recommends balance exercises tailored to an individual; auto-uploading balance performance data to a secure cloud account; and self-performance ratings.

Having talked to our sponsors - who work in the Sienko Research Group researching balance training technologies - and read literature on balance training and medical devices throughout the semester, we have iterated and refined our project requirements and specifications. There are 15 requirements that our solution must meet which cover aspects such as IMU & smartphone specifications, safety, data processing, data security, and adjustability.

Ideas of how the solution would look like have been explored using various concept exploration methods. The selected concept includes a smartphone that can communicate wirelessly with: (1) multiple inertial measurement units (IMUs) via Bluetooth to extract kinematic data from multiple body parts; and (2) a secure cloud database via Wi-Fi to store the data. It also includes a smartphone application that can provide balance training exercise instructions, track exercise progress, and visualize the data collected by the IMUs.

Engineering analyses have been carried out to develop the chosen concept as part of the solution development process. These analyses include benchmarking different smartphones and IMUs, Bluetooth signal strength analysis, database accessibility testing, and formative usability testing for the app. The results we have obtained after having done the analyses proved that our selected concept would be able to satisfy our requirements and specifications - our final solution is therefore largely similar to the selected concept with some minor modifications. The solution verification process was done next to ensure that our solution meets all specifications; most specifications were met and for some, we have developed a verification plan as we did not have enough time to verify if they were met. We have also included a critique of our design solution in this report where we discussed the pros and cons of our design and offered suggestions for improvements in future iterations of the project.

For ME450, our goal was to build a proof-of-concept balance training platform for our sponsors which would be handed over to them at the end of the course for further testing and development with hopes that the solution will be made into a commercially available product.

Table of Contents

Problem Definition

Problem Description and Background

Older adults have an increased risk of falling compared to younger adults [1]. Having fallen before and/or fearing falls could negatively impact the quality of life in the elderly. Falls are also the main cause of physical injuries, injury-related disabilities, and even fatalities in the elderly [2]. Why are falls more common among older adults and what can be done to mitigate fall risks?

Stability in humans, or lack thereof, can be attributed to balance control. Balance is defined as a body's ability to control its center of mass [3] and it is an important health consideration that is often being taken for granted. Falls happen in older adults due to their lack of balance control that can be attributed to the physiological deteriorations, among other factors, that human bodies experience as they age. Examples of such deteriorations include reduced visual perception and cognitive functioning [4]; having cardiovascular conditions and syncope [2]; and reduced muscle strength [5].

According to Winter, our balance control is being challenged the most during walking, and "about 50% of the falls (in older adults) occur during some form of locomotion" [6]. As humans, it is important for us to have mobility (i.e., the ability to move) to be able to perform most tasks in our daily lives. One's mobility can be determined by whether or not they can walk in a manner such that they can control their balance and not fall. A person or an object is considered stable if their center of gravity (COG) is located within their base of support (BOS) which refers to the region around the parts of the person/object that are in contact with a supporting surface - see Figure 1. When humans, who are bipeds start to walk, we lift one foot off the ground and move it forward while the other foot is still on the ground at the same position. This initial transition from standing to walking decreases the size of the BOS (from the one in Figure 1(A) to the size of the foot that is still on the ground) and shifts the COG forward - these two changes cause our COG to be outside the new BOS making us temporarily unstable. The situation is made worse due to the fact our body acts like an inverted pendulum because two-thirds of our body weight are located above our waists. Our balance control systems will prevent us from falling during this brief period of instability by "telling" our central nervous system (CNS) to provide feedback that ensures that we will become stable again. To achieve stability, the raised foot will be put back on the ground, increasing the size of our BOS to make the COG be located within it - Figure 1(B) visually represents this stance.

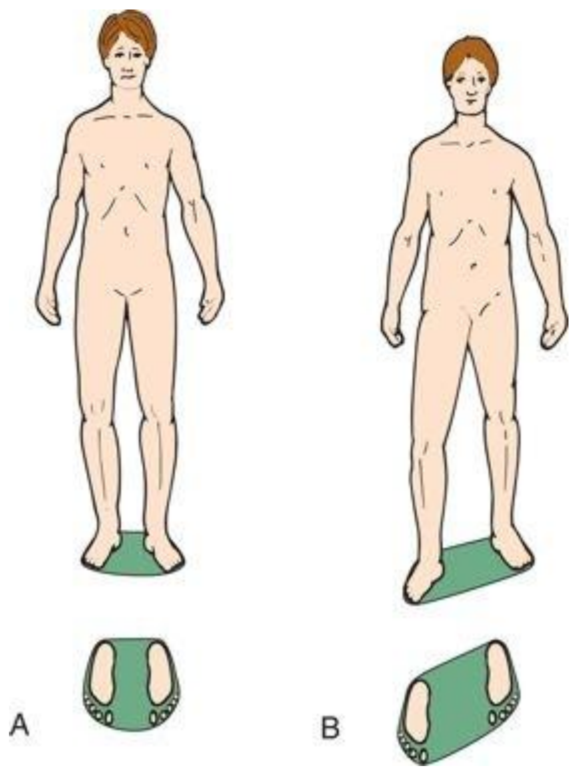


Figure 1. Illustration of a human standing upright (A) and walking (B). The bases of support during these two stances are marked in green. (Image credit: Sheik Abdul Kadir)

Balance Training as Means of Rehabilitation

Balance and resistance training improves muscle strength and balance [7]. A structured exercise framework has the potential to maximize postural control, decrease dizziness and provide rules for exercise progressions [8]. Therefore, there are six categories of balance training, which correspond to the six different balance control systems that are included in the Balance Evaluation Systems Test (BESTest) [8]. They are static standing, compliant surface standing, weight shifting, modified center of gravity, and gaze stabilization or VOR training [8].

Each exercise can be modified that will affect the level of difficulty by changing the foot stance, surface, head movements, variations in gait, visual input, and dual-task. The difficulty increases when the base of the support area decreases by doing the exercise with feet apart, feet together, semi-tandem Romberg, tandem Romberg and single leg stance [8–10]. The exercise will also be more difficult in the following order of surface progression: firm, firm with incline, firm with incline, and foam [8]. Older adults have higher visual dependence for balance control which makes the exercise more challenging when it is done with eyes closed compared to eyes opened [8, 9, 11]. Balance exercises that incorporate head movements in the yaw direction are more challenging than those in the pitch direction. Cognitive tasks require higher attention demand so adding tasks that demand voluntary movement, autonomic postural response, anticipatory postural adjustment, or a combination of all three conditions challenges the patients for optimal recovery [8, 11]. In addition to that, adding variations to gait exercises can also increase the difficulty for the patient. For example, by changing gait speeds, incorporating quick stops/starts, stepping over objects of different sizes, or walking on toes [8, 9].

The Need for a Portable Balance Training Platform

In a conventional physical therapy session, a patient visits their physical therapist (PT) who is typically located in a hospital or a clinic, and performs a series of exercises on-site as prescribed by the PT. The PT will then review and/or rate the patient's performance before prescribing additional exercises that they can perform on their own at home before meeting the PT again for another session.

Balance training exercises are one of the activities offered in physical therapy. It is important for older adults who need to regain balance to perform these exercises. However, balance training in the conventional method comes with a number of limitations which include the availability and accessibility of physical therapists; cost of therapy; and decreased exercise compliance in prescribed home-based balance training [12]. This decrease in compliance is understood to be due to the absence of expert feedback on the exercises that patients perform at home and the loss of motivation which eventually makes them lose interest in continuing their training regimes [13] - this is especially true in situations like the current COVID-19 pandemic where older adults are more cautious about going out of their homes due to health concerns [14]. This is where a portable, personalized balance training platform can help increase patients’ accessibility to balance training exercises.

Home-Based Rehabilitation

Home-based rehabilitation exercises are effective on balance function improvement and are suitable for those who are unable to receive supervised exercise programs or those who need to perform it as a post-treatment [13, 15]. However, lack of supervision reduces the patient compliance to the exercise protocol that causes ineffective exercise execution [7, 15]. Lack of expert feedback also leads to loss of motivation, reduced improvement, and eventual discontinuation of balance exercises [5, 13]. Supervision affects executive function; a small amount of supervised training within mainly unsupervised training gives a positive impact to balance training compared to fully unsupervised training [7].

Problem Statement

From research and discussions, our team has developed this need statement for the project:

> *“There is a need to develop a personalized, smartphone-based balance training platform to support balance training in the homes of older adults.”*

This need statement has the goal to create a smartphone application that incorporates multiple IMUs from different body segments, provides data-driven and personalized exercise recommendations, supports machine learning algorithms, and has a secure cloud account.

Summary of Information Sources

Stakeholders

The stakeholders of this project include the Sienko Research Group researchers, as well as older adults. At this stage, we will be prioritizing the researchers as the main stakeholder as the project goal is to deliver a framework of a smartphone-based balance trainer that the researchers can use to conduct tests with subjects and build on top of the framework. We have identified 4 explicit requirements, which are (1) supports multiple IMUs; (2) supports a machine learning framework to recommend user exercises; (3) automatically uploads user data to the cloud securely; and (4) captures user's self-performance rating. During the interviews with stakeholders, we have discussed their training process in the lab, which can range from 2 to 3 hours including setup, exercise, breaks, and post-exercise interview. We have also identified the importance of making sure that the application software is extensible and modifiable so that the researchers can continue working on improving the application after the ME450 project is completed. We engage with stakeholders by meeting once every two weeks and sending out weekly updates through email.

Literature Review

Literature reviews include academic literature and standards. Academic literature, systematic reviews, and research papers provide an understanding of balance training exercises, physiological impacts to falls in older adults, rehabilitation programs, and smartphone-based platforms. Journal articles were also considered when deciding on the placements of the sensors on body segments: head, trunk, and ankles. Higher levels of strength in the lower muscle groups have shown to reduce body sway and muscle strength at the ankle joints correlate positively with the range of limit of stability and balance performance [12]. In addition, acceleration of the head, trunk, ankles, and arms have great impacts on body balance as they correlate with the acceleration of the body gravity centers [16]. Articles also provide techniques to approach limitations when designing smartphone applications for older adults and the user interfaces. Commercially available IMU sensors, as well as fitness and balance training applications, were benchmarked in the concept selection phase.

Benchmarking Existing Technologies

The existing balance training platform includes exergames using Wii Fit, insole wedges with pressure sensors, wearable IMUs, and smartphones as sensors. They are proven to show positive impacts to balance training and postural stability but they have some limitations.

Interactive exergaming using force platforms (Wii Fit and Wii Balance Board) or video systems (Microsoft Kinect) improves balance and postural stability with benefits including a concordance of visual and proprioceptive information, enhanced information about joint movements, and incorporation of gaming features [17, 18]. However, force platforms restrict the base of support during exercising, which makes it harder for older adults to balance and may result in falls during training [18]. Camera-based exergame systems require a continuous unobstructed sightline, which limits the placement of desk or chair as support that ensures the safety of the older adults [18].

Wearable pressure sensors are used to cue weight symmetry distribution between feet while standing or weight variation during the stance phase while walking [10]. Insole wedges embedded with pressure sensors restores standing and walking symmetry, balance, and gait. However, it does not emphasize speed of movement, which is a critical feature in dynamic exercise [19]. Smart shoes, with pressure sensors on the shoe pad, used for gait training also significantly reduced the number of falls and enhanced mobility, endurance, strength, and flexibility. Although the shoes come in three sizes: small, medium, and large, they were not a perfect fit for each subject [20].

IMUs placed at various body segments are most frequently used as sensors for balance training platforms. The IMUs placed on the lower back provide feedback about the sway of the center of mass while IMUs placed on the shanks, thighs, and lower back measure lower extremity motion [10]. However, this option is still limited to measuring movement of only one body segment at a time or of only the lower extremity when measurements of upper and lower extremities are important for body balance assessment.

Furthermore, we explored existing technologies beyond balance training that are home-based or give personalized recommendations. One such technology is the I-TRAVLE, a robot-assisted upper limb rehabilitation [21]. It is used for patients suffering from Multiple Sclerosis. It consists of a haptic robot that acts as an input and output device, as well as a large TV screen to display the exercises in the form of games. In terms of personalized recommendations, the system uses parameters such as task completion time and error rate to determine the patient's performance and automatically adjust the difficulty level of the game. One benefit of the system is that the automatic adjustments are based on the patient's condition on the day of the exercise, instead of it being based on the previous session with a PT.

We have summarized the comparison of features between the different existing technologies in Table 1. Images of said technologies are also presented in Figure 2. It should be noted that none of the existing technologies fulfill all the features listed below (which is what our project aims to achieve).

Table 1: Comparison of features between different existing technologies

	Wii Balance Board	**Microsoft Kinect**	**Smartphone Balance Trainer v1**	**I-TRAVLE**
Can be used at home	Yes	Yes	Yes	Yes
Measures data from multiple body segments	No	No	No	No
Provides balance training	Yes	Yes	Yes	No
Gives personalized recommendations	No	No	Yes*	Yes

* Requires recommendations from PT through the internet

(i) Wii Balance Board (ii) Microsoft Kinect (iii) Smartphone Balance Trainer v1 (iv) i-TRAVLE

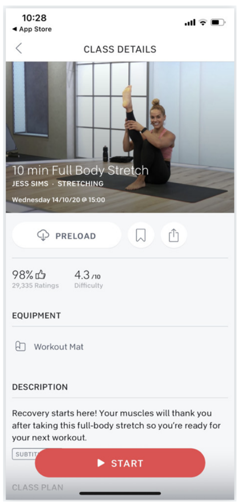
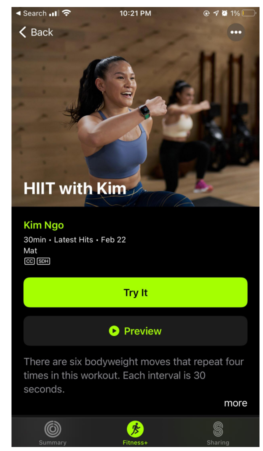
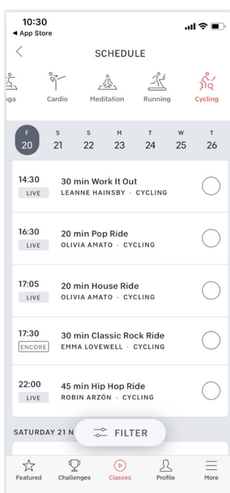
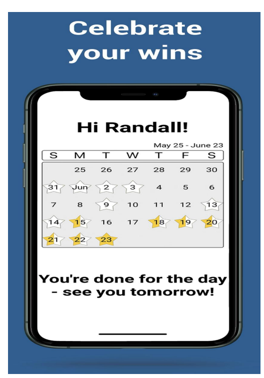
Figure 2: Images of the existing technologies. **(i)** Wii Balance Board, **(ii)** Microsoft Kinect, **(iii)** Smartphone Balance Trainer v1, **(iv)** I-TRAVLE

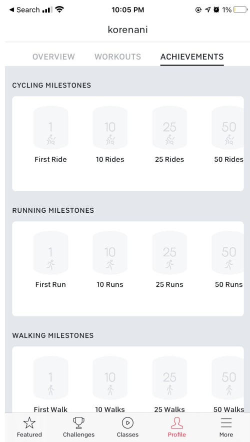
The four features listed on the leftmost column in Table 1 are the key features that our solution needs to have based on the project description. As seen in the same table, none of the existing physical rehabilitation technologies benchmarked are able to be used at home, measure kinematic data from multiple body segments, provide balance training, and give personalized exercise recommendations at the same time - each only has some, not all of these features. Additionally, none of the existing solutions can be used to measure kinematic data from multiple body segments which led us to believe that there is not a commercially available balance training platform that can perform this task. We aim to develop a smartphone-based balance training platform that has all of these features and for each of the features, we plan to look into the existing technologies that have it as a guide to inform our design decisions.

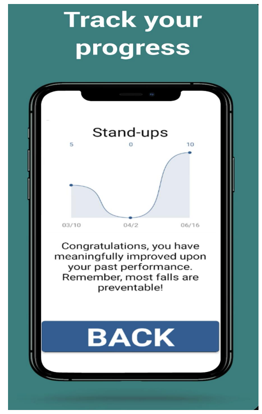
Benchmarking Existing User Interface Designs

We also benchmarked balance training and fitness-related applications to gain perspectives on the processes and performance of the applications already being used by consumers. Even though our software would not be a user-facing product at this stage, we gained valuable insights on how different components and the underlying structure are designed. We looked mainly into how these applications present their content, provide incentives to encourage users' activities, and how personal progress is being tracked in the applications. The information is summarized in Table 2.

Table 2: Benchmark of popular fitness applications

	Peloton	**Nymbl Balance Training**	**Fitness+**
User interface	Videos with text description 	Animation with large font size 	Videos and the flexibility to choose background music 
User incentives	Exercise schedules 	Daily progress calendar 	Training with friends, achievement badge 
Data visualization	Personal achievements	Personal progress plot	Personalized data visualization





Information Gaps

Information gaps, especially related to elder users, are anticipated throughout the course of software development. Designing an informative and interactive UI for older adults is a challenging task since most mobile applications are designed for younger users that are exposed to generic gestures and UI interactions. Studies have found that most seniors only perform basic tasks on their smartphones, such as texting and making phone calls, and only a small portion of seniors exploit the full potential of smart devices and run applications on their smartphones [22]. The lack of senior users in the first place creates a vicious cycle in the smartphone application market; those profit-driven application development companies are less likely to take senior’s cognitive abilities into consideration, which consequently limits existing UI frameworks we can reference. Another challenging aspect is that our design team may not have direct user feedback from seniors, since the early version of our application will mostly be used by researchers in a laboratory environment. Thus, our interface will focus on the researcher's user experience that emphasizes data visualization and information processing, and senior users may experience a hard time traversing menus and completing their tasks.

Requirements and Specifications

The user requirements were mainly determined by our sponsors through interviews with them. We’ve also gathered information from benchmarking, literature review, and the CDC anthropometric data to inform our requirements and specifications. Furthermore, the corresponding specifications to each requirement are quantifiable and testable. We have divided the requirements and specifications into two parts, hardware, and software, for clarity. Table 3 below shows the requirements and specifications of the sensor system (hardware) and Table 4 shows the requirements and specifications of the application (software).

Table 3: Requirements and Specifications of the sensor system (hardware)

Priority	Requirements	Specifications	Source

1	Includes a smartphone	- Supports both Bluetooth (at least v4.0) and Wi-Fi connections - No service plan and contract (unlocked) - Android or iOS - Has accelerometer & gyroscope sensors - Screen size > 5”	Stakeholders
1	Must pair multiple IMUs directly to a smartphone	- ≥ 4 IMUs - Sampling rate ≥ 60 Hz - Accelerometer range of 4-12 g - Gyroscope range of 400-1000 °/s - Accelerometer & gyroscope resolution ≥ 16 bit - Signal strength ≥ -78 dBm - Synchronized timestamps between each sensors	Stakeholders, Lueken et al.
1	Safe	- Operating temperature < 43°C	Stakeholders, IEC
2	Is adjustable for different body shapes	- Adjustable circumference in cm (5th to 95th percentile): Head: 51.9 - 59.7 Waist: 82.8 - 136.3 Mid-upper arm: 23.0 - 43.3 Mid-thigh: 83.6 - 129.0 Mid-calf: 28.4 - 46.1 Ankle: 19.3 - 24.0	Rollins JD, CDC
2	Has sufficient battery life	- Battery life > 3 hours**	Stakeholders, Lesinski et al.
2	Does not constrain body motion	- Lightweight, < 2.5kg	Stakeholders, Abass et al.

** More than 6 hours is recommended in future development phase

Includes a smartphone

The stakeholders have requested smartphone specifications that are compatible with our system. The smartphone at a minimum should support both Bluetooth (at least version 4.0) for data collection from IMUs, and Wi-Fi connection for user credentials and cloud upload. The Bluetooth specification is such that the smartphone is compatible with Bluetooth Low Energy (BLE) so that the smartphone has a reduced energy consumption when connected to the IMUs. Furthermore, the phone should be unlocked and free from any carrier service plan. Connection to the internet is expected through Wi-Fi connections, so a carrier service plan is an unnecessary expense for the project. The smartphone should also support Android or iOS, although priority will be given to the Android system due to development constraints from Apple's platform. Additionally, the phone may also be used as a sensor and consequently should also have both accelerometer and gyroscope embedded. Screen size greater than 5 inches is also required so that the text and video won't be too small for effective instructions when used by older adults.

Must pair multiple IMUs directly to a smartphone

The finalized platform should support simultaneous kinematic readings from multiple IMUs in order to get a better estimation of the center of mass (COM) for evaluation of postural control. This requires an anthropometric model, which consists of several body segments and a full kinematic description of each marker attached to specific proximal and distal body landmarks [22]. Each IMU unit needs to have both an accelerometer and a gyroscope embedded for linear and angular accelerations measurement. Four IMUs are required for mapping the body motion, two on legs, one on the head, and one on the waist, and the sponsor also mentioned that 5 IMUs may be optimal for more accurate motion capture. A detailed set of IMU specifications were acquired from research papers and stakeholder meetings, including sample rate, sensor range, resolution, and signal strength. The sensor range and resolution are specifically tuned for walking and low-intensity body motion. Additionally, the sensor system needs to achieve synchronized reading to ensure that the latency between two sensor's readings is less than the sampling rate.

Safe

Safety is also our highest priority. A fully wireless system is preferable, as excess wire could potentially trip or strangle the users during the exercise. However, if wires need to be included in our system due to unstable wireless connection or other design issues, the wires should not swing and must be firmly attached to the body during the exercise such that the sensor system will not lead to any safety concerns. The operation temperature of each IMU should also be less than 43℃ following the IEC standards.

Is adjustable for different body shapes

Most IMUs use a strap to secure body attachment, and some others are directly taped on bodies. Regardless of the attachment method, the sensor system should fit the 5^{th} to 95^{th} percentile of the elder population's body shape based on anthropometric reference data [23] [24]. The attachment method should also ensure that the IMUs remain stationary during the exercise for accurate data collection.

Has sufficient battery life

While most balance training ranges from 15 to 90 minutes [25], the sponsors also mentioned that their entire training session may take up to 3 hours. Even though data collection might not happen all the time within the 3 hours, the IMUs will still be left on. In the future, it is recommended to have more than 6 hours of battery life for a 2x safety factor.

Does not constrain body motion

A good design should consider how the sensors are connected and attached to the user's body. The sensors should not interfere with the user's body motion or lead to discomfort while remaining relatively flat and stationary throughout the training exercises. Therefore, the sensor system should only introduce negligible extra weight and each sensor unit must be small. The upper limit of the total weight was set to 2.5kg, roughly 5% of body weight that may affect balance during gait exercise [26]. The sensor system and mobile device will likely utilize Bluetooth or a wireless network to establish a connection as a wired connection would limit how far the user can be from the mobile device and potentially trip the users.

Table 4: Requirements and specifications of the application (software)

Priority	Requirements	Specifications	Source
1	Well documented	- ≥ 1 comment on each function implemented in the code - A readme file explaining the structure of the software	Stakeholders
1	Captures user rating	- Use a 5 point scale	Stakeholders
1	Has file export capabilities	- Export raw balance training data in a .csv file - Choose between ≥ 4 data metrics to export - Must include raw acceleration, raw angular	Stakeholders

		velocity, quaternion and euler angle	
1	Tracks exercise progress	- Shows time left on timed exercises - A start and stop button - 3 second buffer before timer starts	Stakeholders

2	Identifies each user separately	- A login page with a 10-20 character identifier for each user which indicate
		    - Project ID
		    - Test subject ID
			Stakeholders

2	Includes user friendly design	- Have an icon accompany the text for exercise instructions	Stakeholders
2	Recommends user exercises	- Recommended exercises in each category based on user's previous session - Gives an option to do other exercises as well	Stakeholders
2	Visualizes balance training data	- Comparing user's data to normative data - PT rating simulating ML response	Stakeholders
3	Automatically uploads balance data to cloud	- Encrypted with 256-bit AES encryption - Uptime of ≥ 99.9% - Response time ≤ 1.4 s - Bandwidth ≥ 1.44 MB/s	Stakeholders, Zhou et al., Tsiachri Renta et al.

Well documented

As we expect the project to go further than what we will be doing within our scope in ME450, a well-documented code would be helpful for the researchers when needing to update or modify the software. It should have at least 1 comment per function describing why the function was implemented and how the function works. Additionally, there will be a readme file explaining the structure of the software, which should give a broad mental framework of how the program is organized and how it works.

Captures user rating

The user rating is an important aspect for the user to improve in balance training. It will employ a 5 point scale. The user rating will be used by the researchers to quantify the user's balance performance and recommend exercises. An optimum exercise for the user would be that of the scale of 3, which provides a moderate level of challenge [5].

Has file export capabilities

In order to further analyze the balance performance data elsewhere, having the ability to export the files associated with the data is crucial. The data must export in a .csv format, which is a file format that has high compatibility with various software applications. The data export should support at least 4 data metrics, which are raw acceleration, raw angular velocity, quaternion, and Euler angle. These metrics come directly from the IMUs' measurement.

Tracks exercise progress

To ensure better data collection from the IMU, the software should be designed such that it will only capture data during the exercise progress. In order to achieve this, a clear start and stop button should be present to the user so that they can start and stop the exercise when they're ready and done, respectively. After pressing the start button, a 3-second buffer will be added so that the user will have time to get ready to perform the exercises. Additionally, a timer must be present for timed exercises, which will let the user know how much time is left on the exercise.

Identifies each user separately

To correlate the recorded data to the user, the application should be able to identify each user separately. The researchers categorize each user with a project ID and test subject ID. To accommodate this, the application will include a login page with a 10-20 character input. Once the correct ID is inserted, all the data recorded during that session will be associated with the respective ID.

Capable of data visualization

For the users and researchers to be able to monitor and diagnose their balance performance, some form of data visualization is needed. As such, the data visualization will be in the form of comparing user's data with normative data and physical therapists' ratings simulating the machine learning response. Since this is a proof-of-concept project, the normative data used for data visualization is for one of the exercise categories, which is static standing obtained from a research paper by Roman Liu, Danuta. The data for other categories will be randomly generated. This also applies to the physical therapists' ratings as the machine learning algorithm is out of our project scope.

Recommend user exercises

As the older adults are performing the exercises regularly, one of the next steps to balance training is showing improvements in performance. Normally, this is done with a PT physically available with them. However, in the absence of a PT in their homes, a machine learning algorithm will be used to recommend balance exercises based on their performance. It will make decisions based on at least 2 inputs, the kinematic data from the IMU (which measures how "correct" the user has done the exercises) and user rating (which measures how comfortable they are with the current exercise regimen).

Automatically upload balance data to the cloud

Cloud infrastructure allows the user's balance performance data to be analyzed by a PT without being physically available. To maintain best practices of uploading user data, the cloud infrastructure will be secured with 256-bit AES encryption, an industry standard. The encryption key will have 2^{256} possible combinations, which will take millions of years to crack. In addition, the cloud should remain available for at least 99.9% of the time to avoid hampering the user experience. This means that it should at most be down for about 9 hours in a year. This is based on benchmarking different cloud providers hosted in North America including but not limited to Google, Amazon, Alibaba, and DigitalOcean [27]. Furthermore, the response time to the cloud should not be more than 1.4 s, based on good healthcare data management practices on the cloud [28]. The bandwidth of the cloud should also be at least 1.44 MB/s, based on the IMU file size estimate using a sampling rate of 60 Hz [29].

Concept Exploration

The balance training platform system that we are developing for this project can be divided into three smaller subsystems: (1) hardware; (2) software; and (3) hardware-software integration. Since they are independent of each other (i.e., the design of one subsystem does not impact that of another subsystem), we explored three different solution spaces - one for each subsystem. Fundamentally, regardless of which subsystem we were exploring the solution space for, we would like to be sure to apply divergent thinking techniques to come up with as many solution ideas as possible before gradually narrowing down to the one solution that we think is the best one - though it may not be the final one - to develop and test in the Solution Development stage of the design process.

Hardware: Concept Generation and Development

By the time we started going through the concept exploration stage, we already had a rough idea of the kind of sensors we will be using for the balance trainer after communicating with our stakeholders. The sensors are inertial measurement units (IMUs) that are off-the-shelf/available commercially and they have dimensions of a typical car key fob. In addition, we also knew that the use of a smartphone is a requirement for the system and it can be used as an IMU and/or a media player for exercise videos.

For the sensor system, we benchmarked several commercially available IMUs to inform our design process. The information is summarized in Table 5 below.

Table 5: Benchmarks of commercially-available IMUs

	Xsens DOT	**Mbientlab MMR**	**LPMS-B2**
Supported platform(s)	Android iOS Windows macOS	Android iOS Windows Linux	Android Windows
Simultaneous connections (Bluetooth)	Up to 7	Up to 3	Up to 7
Battery life	Up to 6 hours	Up to 8 hours	Up to 6 hours
Cost (single unit)	$105	$80.99 - $87.99	$199 - $249

From the benchmarking results, we found that Bluetooth can be severely limiting with multiple concurrent connections. This is because the central device (the smartphone) only has one radio and one antenna, and when it is shared between the different peripheral devices (the IMUs), the throughput of the Bluetooth receiver becomes limited, reducing the speed at which the data is transferred. However, Bluetooth is the only way we have identified to connect directly to smartphones as there were no other wireless protocols that didn't require a hub (an intermediary between the IMUs and smartphone).

Having known how the hardware components should look like for the system, we started brainstorming ideas for how to attach the IMUs to the body parts for which kinematic data are measured - both legs near the mid-calf, the trunk, and the head, to name a few - as well as how to use the smartphone during an exercise. We had a small brainstorming session (~10 minutes) where every team member generated as many ideas as possible on a Google Jamboard without any regard for their feasibility. The rationale behind using brainstorming as an ideation tool is that we wanted to create a platform for everyone to think creatively without feeling like they were being judged as the best solutions can sometimes be inspired by many different ideas. Figure 3 below shows some of the ideas that were generated during the session.

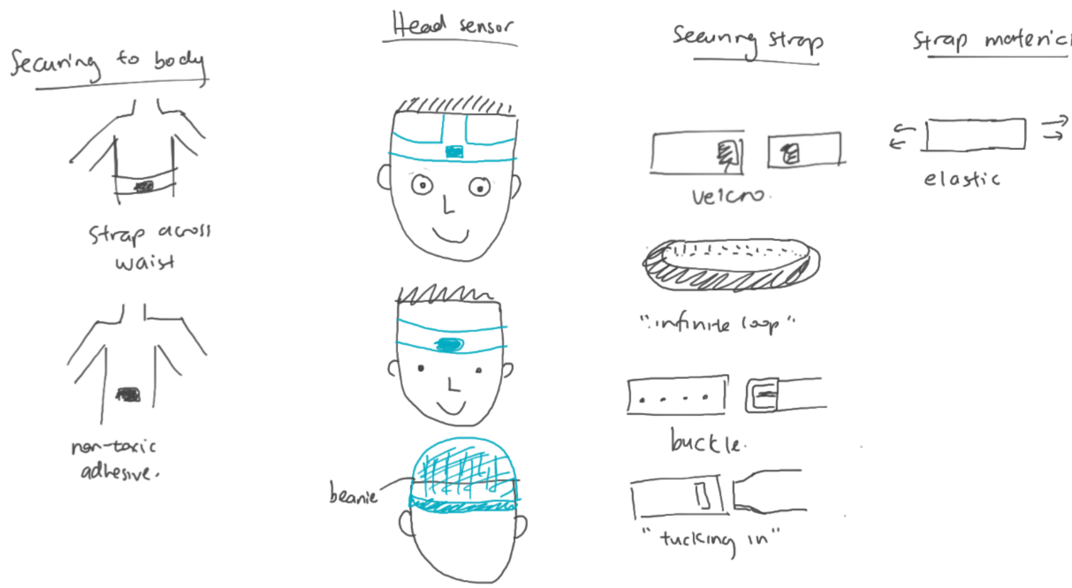


Figure 3: Several sensor attachment ideas generated from the brainstorming session.

Ideas generated during the brainstorming were then further developed through morphological analysis. When performing this analysis, the first step being done was identifying the four sub-functions of the system's hardware: (1) IMU attachment to the trunk; (2) IMU attachment to head; (3) IMU attachment to limbs (arms & legs); and (4) smartphone placement during exercise. Then, a morphological chart was made such that each row on the chart represents the ideas (one in each column) that correspond to one sub-function. Generated ideas were organized into what sub-functions they belong to using this chart. We looked at each row on the chart and thought about additional solutions that could fulfill the sub-function that was being looked at. The use of a morphological chart as a concept development tool for the system's hardware is beneficial to our team as we were able to structurally generate/develop ideas according to the requirements our system must fulfill. We thought that other tools like design heuristics and TRIZ were less suitable for our purposes as these tools support novelty and variety in finding a solution which we prioritize less when compared to functionality. Figure 4 presents the morphological chart that we have created and a full description of each of the ideas put in this chart can be found in Appendix B.

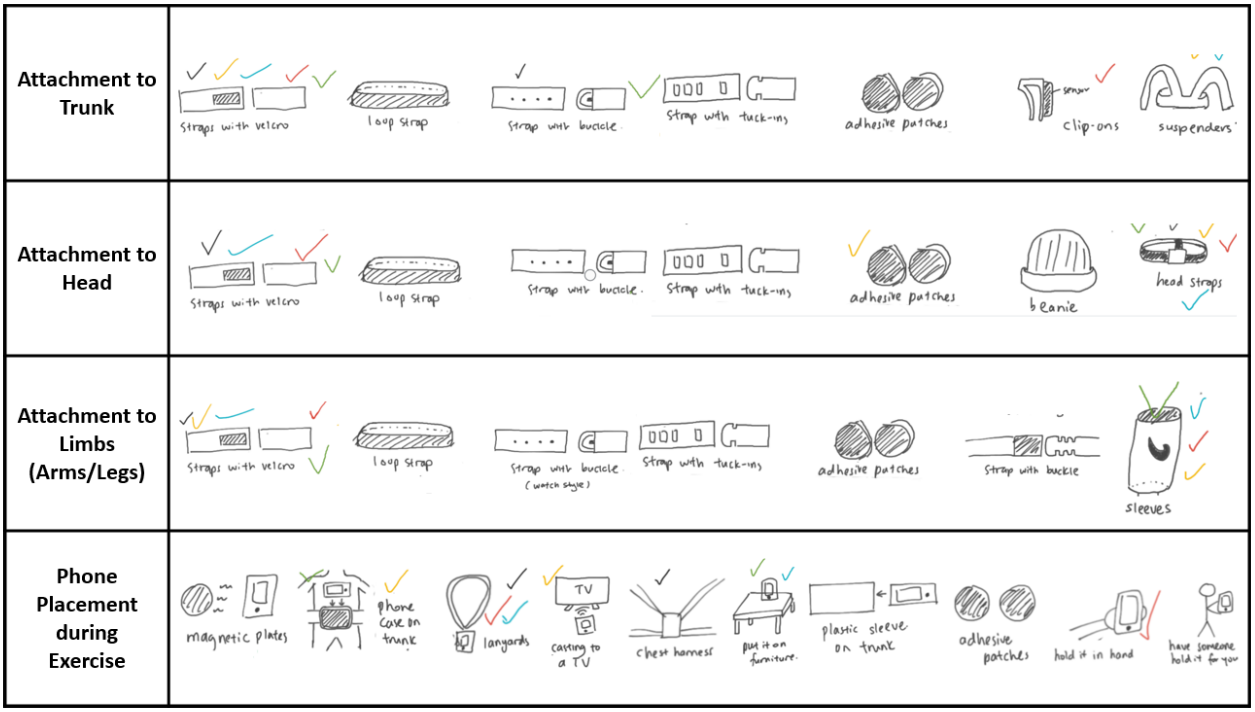


Figure 4: Morphological chart for the hardware. The check marks in multiple colors represent the team members' votes for the best ideas for each sub-function (explanation below).

With at least 7 components (concepts) in each row, we have identified at least 7^4 = 2401 possible combinations from the morphological chart, one of which would ultimately become the selected concept. It would be very time-consuming and counterproductive to evaluate all of the possible combinations before selecting a solution to develop, so we have decided for each team member to take a vote on the two best ideas from each row leaving only 2^4 = 16 possible combinations that would move on to the concept selection process. The top two ideas for each sub-function as decided by a majority vote by all team members are shown in Figure 5. Any one combination is formed by putting together four ideas, one from each sub-function.

The Concept Evaluation/Selection section below explains how these 16 combinations would be evaluated using a structured process which helps us to finally converge to one solution that we will develop in the next design stage.

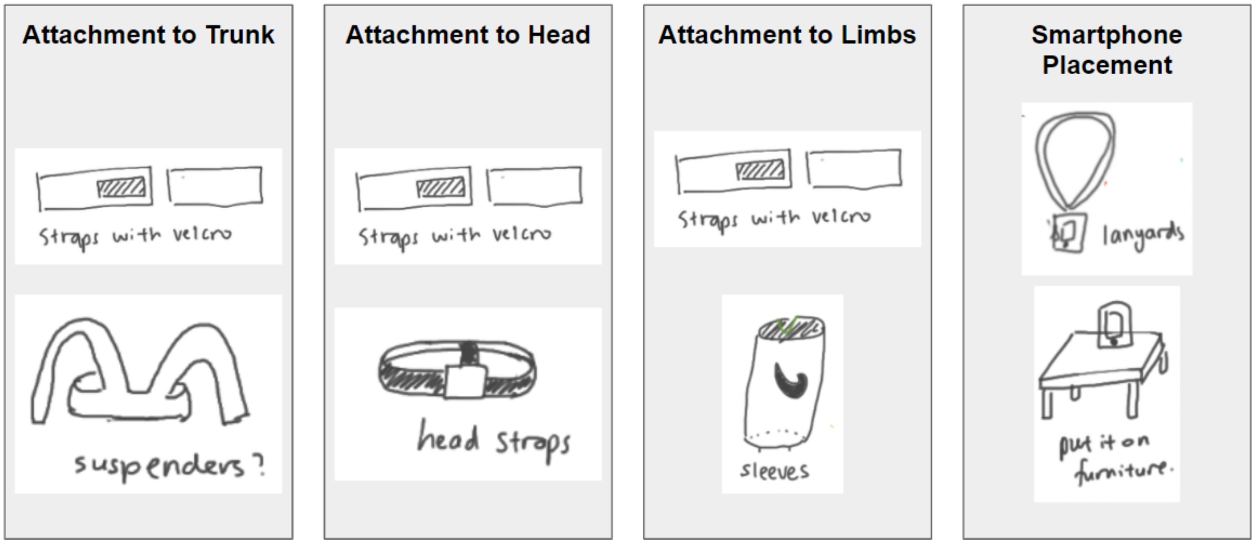


Figure 5: Top two ideas for each sub-function as voted by the team members.

Hardware: Concept Evaluation/Selection

Concept Screening

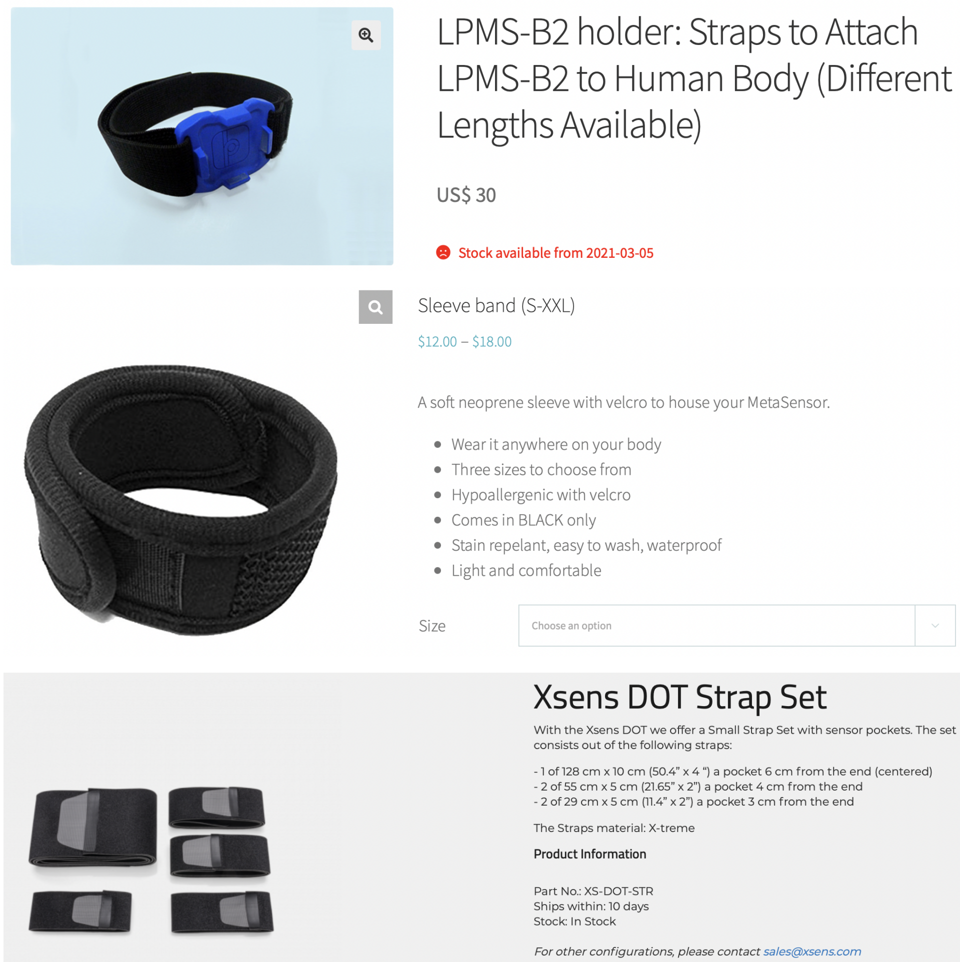
The first step we took in selecting the best solution to move forward with was screening each of the 16 combinations from the concept development stage by comparing the ideas in Figure 5 against the list of requirements and specifications in Tables 3 and 4. The idea(s) that we thought would not be able to fulfill any of the requirements if it was/they were realized.

The only idea that failed the concept screening process was the sleeve that would be used to attach IMUs to limbs as we thought that it would not be adjustable enough for different body shapes - it cannot be a one-size-fits-all solution. If we would like to accommodate the 95th percentile limb circumferences using sleeves, it would only be achievable if the sleeves come in multiple sizes like how most T-shirts and pants do. Since the final product will be first used by our sponsors who are also researchers in balance training studies, they should have a product that can accommodate as many users (research subjects) as possible without the need to change some of the parts for different people.

Eliminating the sleeve idea reduced the number of possible combinations to eight (8).

Concept Evaluation

After screening the concepts and getting the number of possible combinations to eight (8), these combinations go through an evaluation process that begins with judgment of their feasibility. Since this product will be eventually used by patients once this proof-of-concept is successfully developed by our stakeholders, we do not aim to custom-make the hardware and we would like to get as many parts off the shelf as possible. Hence, we have decided to conduct a feasibility judgment on the eight combinations by researching whether or not the parts in these combinations are commercially available (i.e., ready-made for purchase). While browsing for different off-the-shelf sensor systems, we discovered that each of the sensor (IMU) manufacturers sells straps with pockets designed specifically for their sensors as one of the ways to attach them to the human body. We also came across head strap mounts for action cameras like GoPro cameras on Amazon that we thought could be retrofitted for our IMUs by using adhesives to mount them to the strap at the position where an action camera would be placed. Lanyards with phone pouches can also be found easily from most retailers. However, we could not find any kind of suspender-style devices that can be used as sensor attachment devices to the body. Figure 6 below shows some examples of commercially available sensor attachment devices.



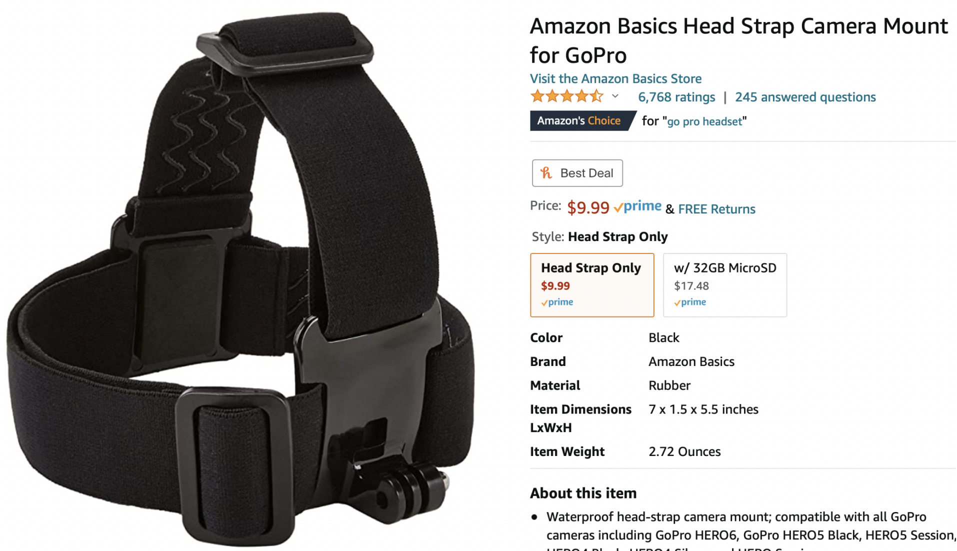
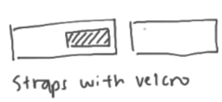
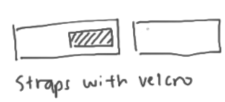
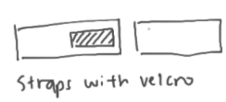
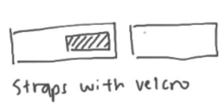
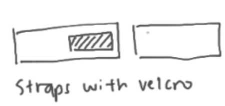
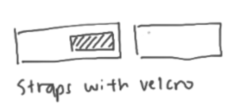
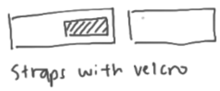
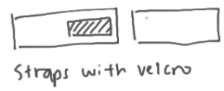
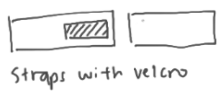
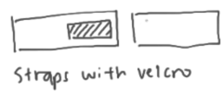


Figure 6: Some examples of commercially available sensor attachment devices. From the top: strap for LPMS-B2 sensors, strap for MbientLab MMR sensors, straps for Xsens DOT, and head strap mount for GoPro cameras available on Amazon.com.

After the possible solutions went through a feasibility judgment process, we were able to further reduce the number of possible combinations to four (4). These combinations are shown in detail in Table 6. They were then compared against each other using a list of pros and cons, and the best solution based on the list would be selected for development.

Table 6: Four possible combinations which would be further evaluated to select one solution.

Options	Attachment to Trunk	Attachment to Head	Attachment to Limbs	Smartphone Placement
1				
2				

3				
4				

Concept Selection

We discussed the advantages and disadvantages of each of the options in Table 6 above and wrote them in a table of pros and cons as shown in Table 7 below.

Table 7: Pros and cons for each of the options.

Options	1	2	3	4
Pros	Can use phone as IMU All straps are made for a specific sensor	Lightweight configuration Can view content on phone easily	Can use phone as IMU	Lightweight configuration Can view content on phone easily
Cons	Phone weight may exert pressure on neck No variety	Phone is far from user No variety	Need adhesives for sensor on head - may be messy Phone weight may exert pressure on neck	Need adhesives for sensor on head - may be messy Phone is far from user

Having listed down the pros and cons, we were able to easily decide the solution that we think will most likely succeed. Since we prioritize the balance trainer system being lightweight, we would choose either option 2 or 4. While options 1 and 3 have the advantage of the phone being able to be used as an IMU in addition to the dedicated sensor system that we will purchase, we do not think that this is a need for our project at this stage of the design process. To decide between options 2 and 4, we looked at their list of cons as they have the same list of pros and we ultimately decided that the need to use adhesives to mount an IMU to the head strap might be inconvenient to the users. Hence, option 2 was chosen as the solution that we would like to develop moving forward. However, it is important to note that this solution is not final and since the design process is an iterative one, we may come back to this stage of the design process to choose another solution to develop should the development of option 2 fail.

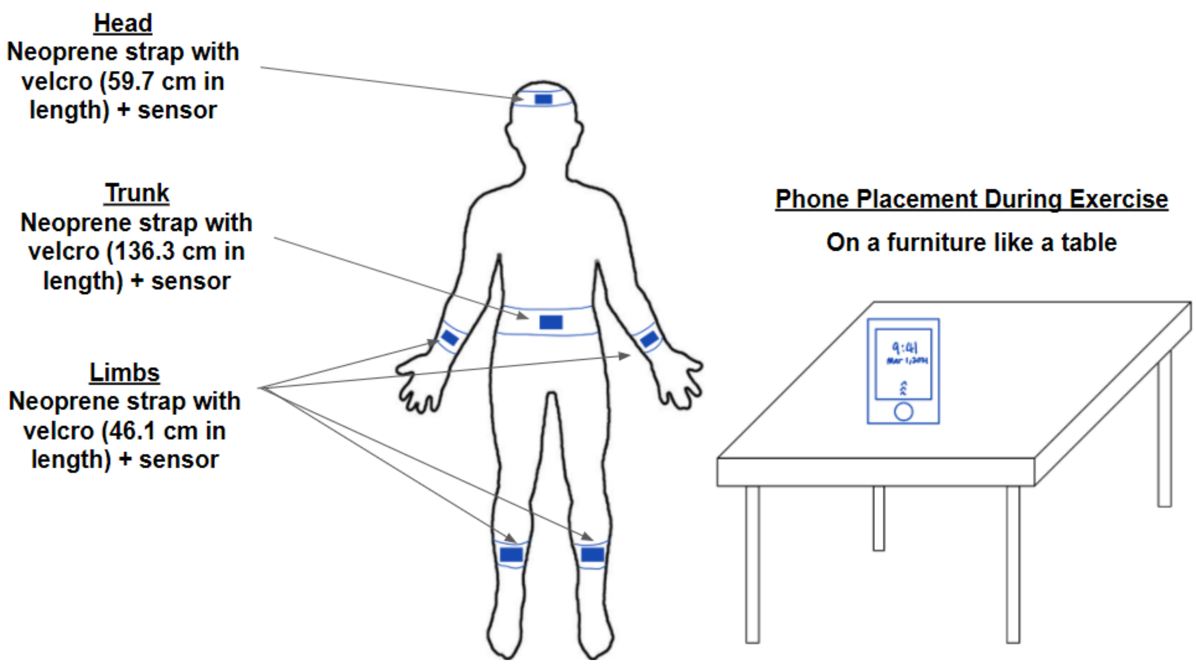


Figure 7: The solution that will be developed (option 2) for hardware. The straps should be adjustable up to their maximum lengths which are stated above. These lengths are determined using the 5th to 95th percentile body part circumferences as listed in the requirements and specifications table as a guide. The placement of the smartphone away from the user during exercise allows them to use it as a media player to watch exercise videos while performing balance training exercises. This solution can fulfill the hardware requirements such that the use of a smartphone is incorporated into the system, the straps are adjustable up to the 95th percentile circumferences, and the hardware itself is lightweight in general which should not constrain body motion during exercise.

Software: Concept Generation and Development

The concepts for the user interface design were generated using four different concept generation methods to come up with possible solutions for our mobile application. The methods are (1) brainstorming, (2) benchmarking, (3) functional decomposition, and (4) Usability Heuristics for User Interface Design. The concept considers older adults' limitations as they are one of our stakeholders and their usability needs are different from other adults, especially for the user interface and user experience design. The benchmark of mobile applications can be seen in Table 2 on page 11.

Brainstorming

The concept generation of the user interface design began with brainstorming the software flow: user journey and back-end flow as well as the functional decomposition of the design. Brainstorming was used to generate ideas for software because we are all comfortable with mobile applications individually and brainstorming will be able to bring together all the diverse experiences and suggestions from each member to find solutions to our problems. We conducted the brainstorming session through Figma, a web-based design and prototyping tool for UI and UX design applications, where all the ideas were written or developed into multiple frames so they can be visualized instantly.

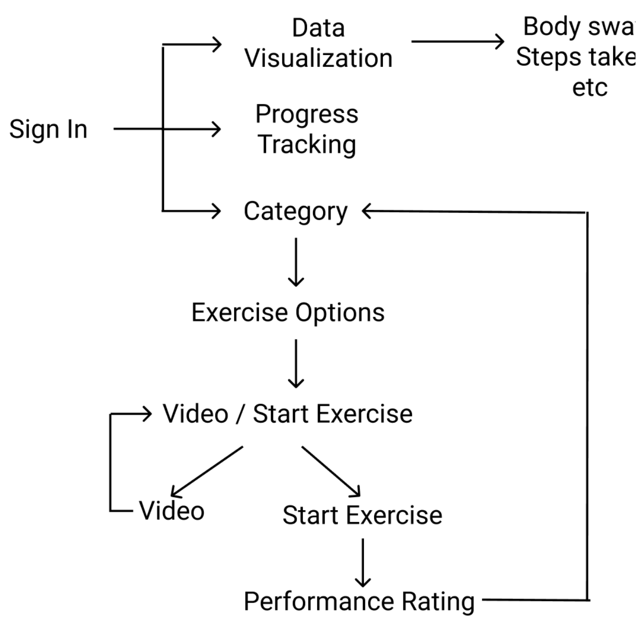


Figure 8: User Journey

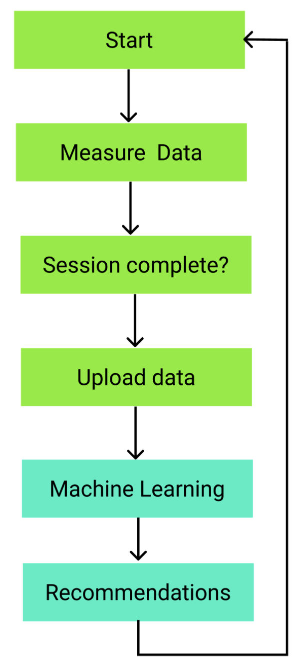


Figure 9: Back-end flow

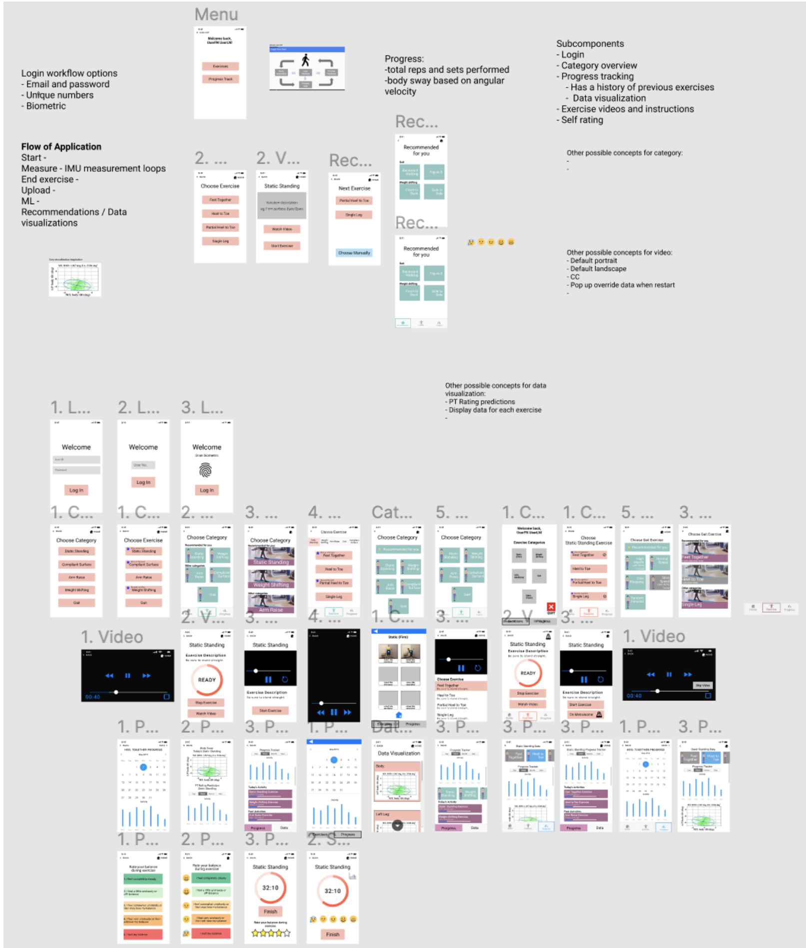


Figure 10: Screenshot of Figma prototyping interface used during brainstorming session.

Functional Decomposition

We performed functional decomposition for our interface designs during brainstorming sessions to break down the overall function of the application to smaller, fewer dominant components so that we can focus on the usability of these main branches before branching out and adapting them to the application interface. The five main components we identified initially include a login component, exercise overview, exercise instructions, progress tracking, and performance rating component. The functions were then narrowed down from five to three after getting feedback from stakeholders that the concept for both the login function and the performance rating function has been decided by the stakeholders. Thus, the updated functional components now consist of exercise overview, exercise instructions, and progress tracking with data visualization.

Usability Heuristics for User Interface Design

Usability Heuristics for User Interface Design by Jakob Nielsen was used as usability guidelines during concept development of the interactive design of the mobile application. These guidelines are used to identify the ways to address important issues with usability to prevent any major usability flaws in the early stage of the design. A more specific explanation of the usability heuristics that were used is shown in Table 8 below [30].

Table 8: Application of Usability Heuristics to Interface Concept Development [30]

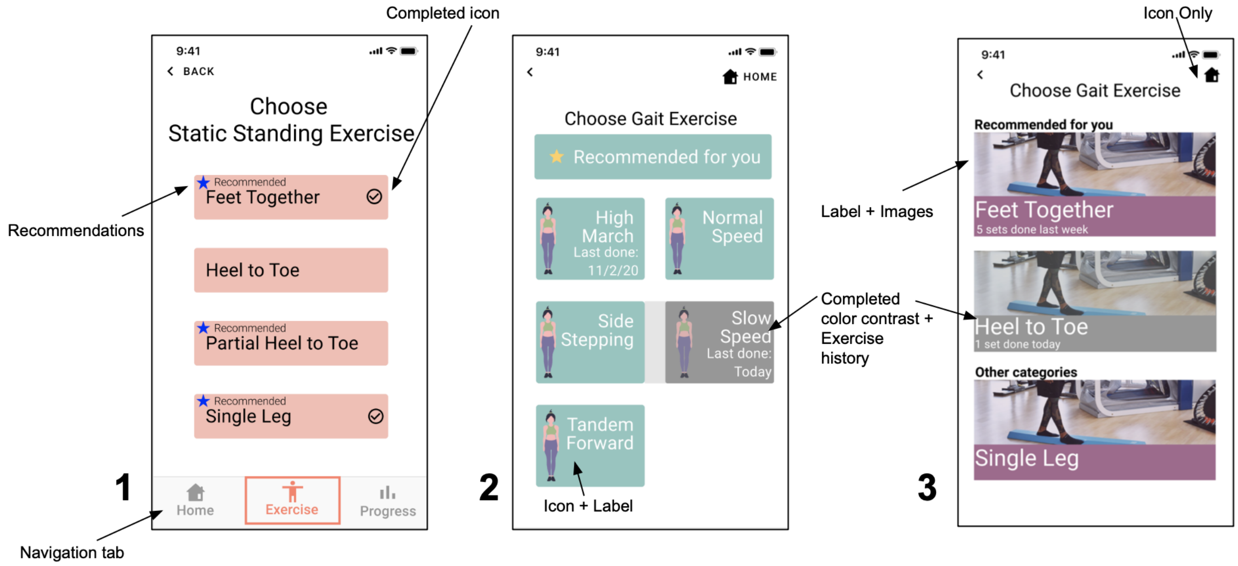
Usability Heuristics	Explanation	Application to interface design
Visibility of system status	How well is the state of the system is conveyed to the users	- Display of remaining time left for timed exercise - Grayed out exercise that has been done
User control and freedom	Allow users freedom to be in control of interaction, even if they made a mistake and will need an exit way	- Large stop and cancel exercise button - Large back button
Consistency and Standards	Follow platform or industry conventions	- Use of universal icons - Consistency in exercise icons
Aesthetics and minimalist design	Interfaces should not include irrelevant information	- Visuals highlights exercise focus

Software: Concept Evaluation

These concepts were then evaluated using a pros and cons list because these concepts were a mix of different features. By using a pros and cons list, we can evaluate the features of the concepts to pick their good or preferable features and avoid features that are not suitable for our users' needs and limitations. Based on the feedback we receive from stakeholders, we prototype the application, which combines the good features from the pros and cons list, and tested it to finalize our user interface design.

Top Concepts

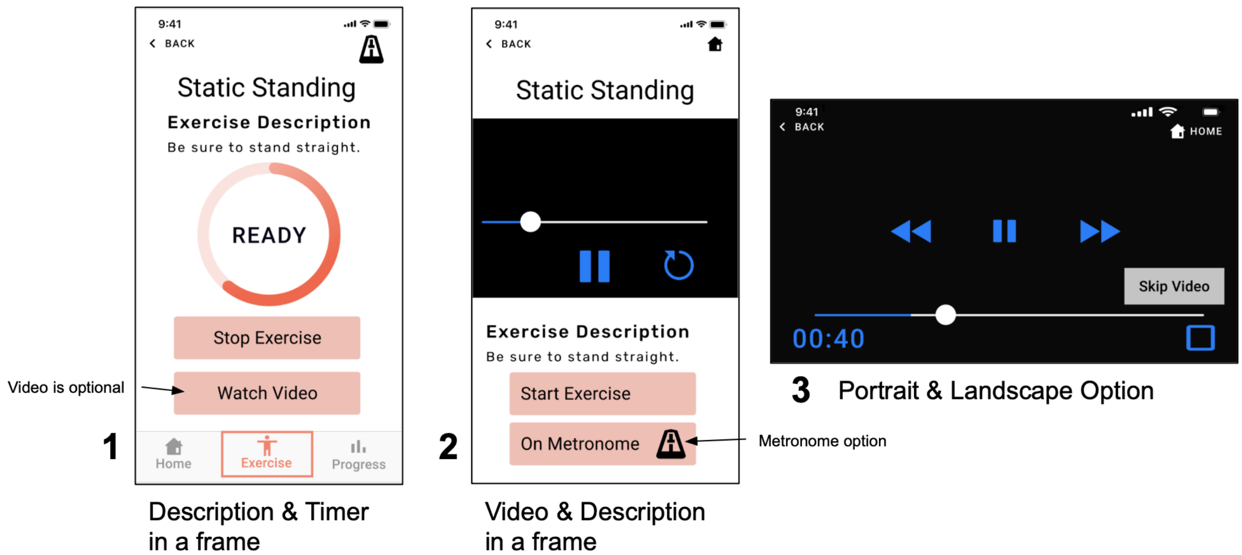
<u>(1) Exercise Overview</u>
The concepts for exercise overview are combinations of multiple features, which are the use of icons or images as a visual aid, different representations of exercise recommendations, exercise history, and navigation buttons. All three concepts include recommended exercises to meet the software requirements as shown in Table 4 on page 15.



Concepts	1	2	3
Pros	- Tab bar for easy navigation - Minimal design - All options present in a frame	- Icon as visual aid - Only recommended options present - Has exercise history - Color contrast to indicate completed exercise	- Image as visual aid - Has exercise history - Color contrast to indicate completed exercise
Cons	- No visual aid - No exercise history	- Small navigation button	- Small navigation button - Need to scroll

<u>(2) Exercise Instructions</u>

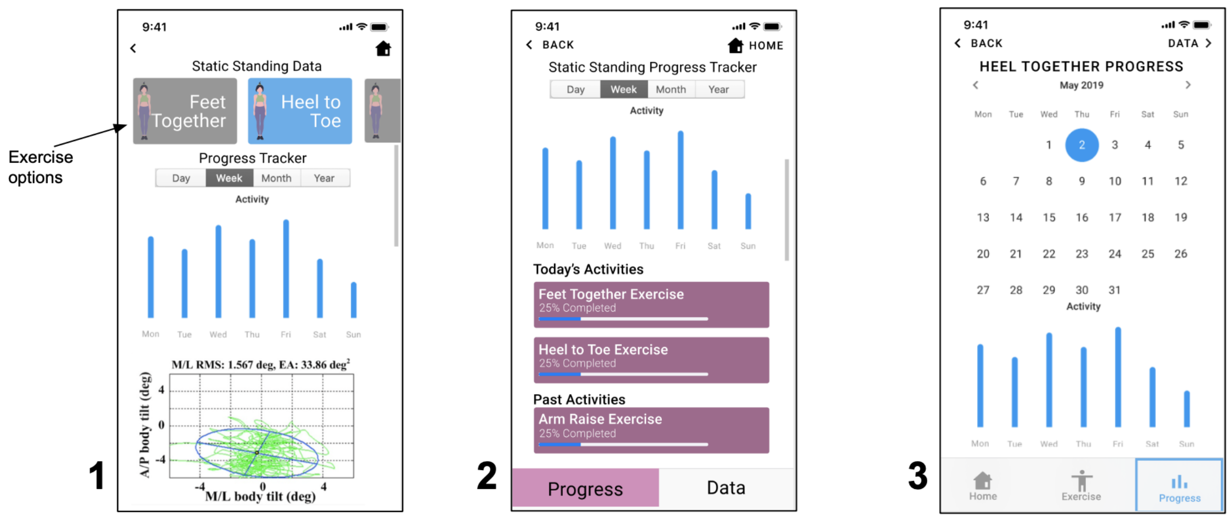
Based on the requirements of the project, the application has to provide exercise instructions for the users. As shown below, the concepts generated for the exercise instructions function incorporate this requirement in two forms, which are video and text format. The video could be an option and can be skipped if it is no longer needed.



Concepts	1	2	3
Pros	- Text-based instructions are easier to follow. - Ready button with countdown timer is attention-grabbing.	- Hybrid with video as a source of reference to text instructions. - Maximize content on screen.	- Video-based instructions showcase motions in great details. - Users have more control over video playback.
Cons	- New users might misinterpret instructions. - Does not work well with complicated instructions.	- Influx of instructions might cause cognitive overload.	- Other functionalities are temporarily disabled during video playback. - Bigger screen real estate on video means less room for other components such as notifications.

<u>(3) Data visualization and Progress Tracking</u>

The mobile application is required to visualize the balance training data of the user as per the requirement in Table 4 on page 16. The data comes in two forms which are user progress tracking and balance data visualization for each exercise.



Concepts	1	2	3
Pros	- Data viz and progress in a frame - Data and progress for each exercise	- Overview of today's and past activities - Tab bar of progress tracking and data viz - Data and progress for each exercise	- Calendar view of daily check-in provides user incentives - Data and progress for each exercise
Cons	- Information overload - Data viz might be irrelevant for older adults	- Information overload	- Small data viz button

Prototyping

From the pros and cons list, we combine the good features into the prototype which is in the order of the user journey in Figure 8 page 27. Older adults being one of our stakeholders and having different usability needs for user interface design compared to other adults, we have taken their limitations into consideration when prototyping our user interface design. Soft colors were used for the buttons in the application as older adults might have vision problems when dealing with bright colors [31]. As for the text, black Sans-serif fonts, with at least 14 point size were used to help older adults with poor visibility to read as it improves contrast with the background and clarity of the text [31–33]. We chose the concept with an icon next to the label for buttons as simple illustrations can help with understanding the options in the application and improve older adults' experience with the interface [31, 34, 35]. We also include icons in the exercise instructions to ensure a more user-friendly interface as per the requirement. As older adults experienced difficulty with successfully choosing targets due to reduced motor control, coordination of fingers, and dexterity issues; large graphics, buttons, and icons are used to ensure they are pushing the right buttons [31, 32, 35].

Our prototype can be seen in Figure 11. When prototyping, we also made sure that the prototype meets the software requirements and specifications of the project in Table 4 on pages 15 and 16. Before starting the exercise, users will enter a 10-20 character identifier which indicates the project ID and test subject ID so the researcher can identify each user separately. Users can then choose from the 5 recommended exercises for each category based on their previous sessions, a feature that meets the requirements and specifications. The exercise that has been done during the session is grayed out to prevent users from repeating the same exercise and to indicate that it is no longer appropriate for them [32]. The interface has a tab bar at the bottom for easy navigation and to make the options more accessible. To meet the requirements and specifications of tracking exercise progress, the timer interface has a large start and stop exercise button, a 3 seconds buffer before the timer starts to give users time to be ready and it will show the time left on timed exercises. For exercises that require users to match a tempo, a metronome option is available in the application. Users are able to rate their perceived stability after each exercise using a visual analog scale of 1-5 scale which aligns with the requirements and specifications of capturing user rating [5]. The application also includes exercise instructions, which come in two formats: text and optional video to give flexibility in use to accommodate their preferences [32]. The exercise instructions will also include simple icons which describe the exercise activities to help them understand the focus of the activity and make the interface more user-friendly [31, 34, 35]. The application will also visualize data for each exercise in at least two plots: bar chart for progress tracking and body sway graph for data visualization in different tabs to accommodate our different users, which are either researchers or older adults.

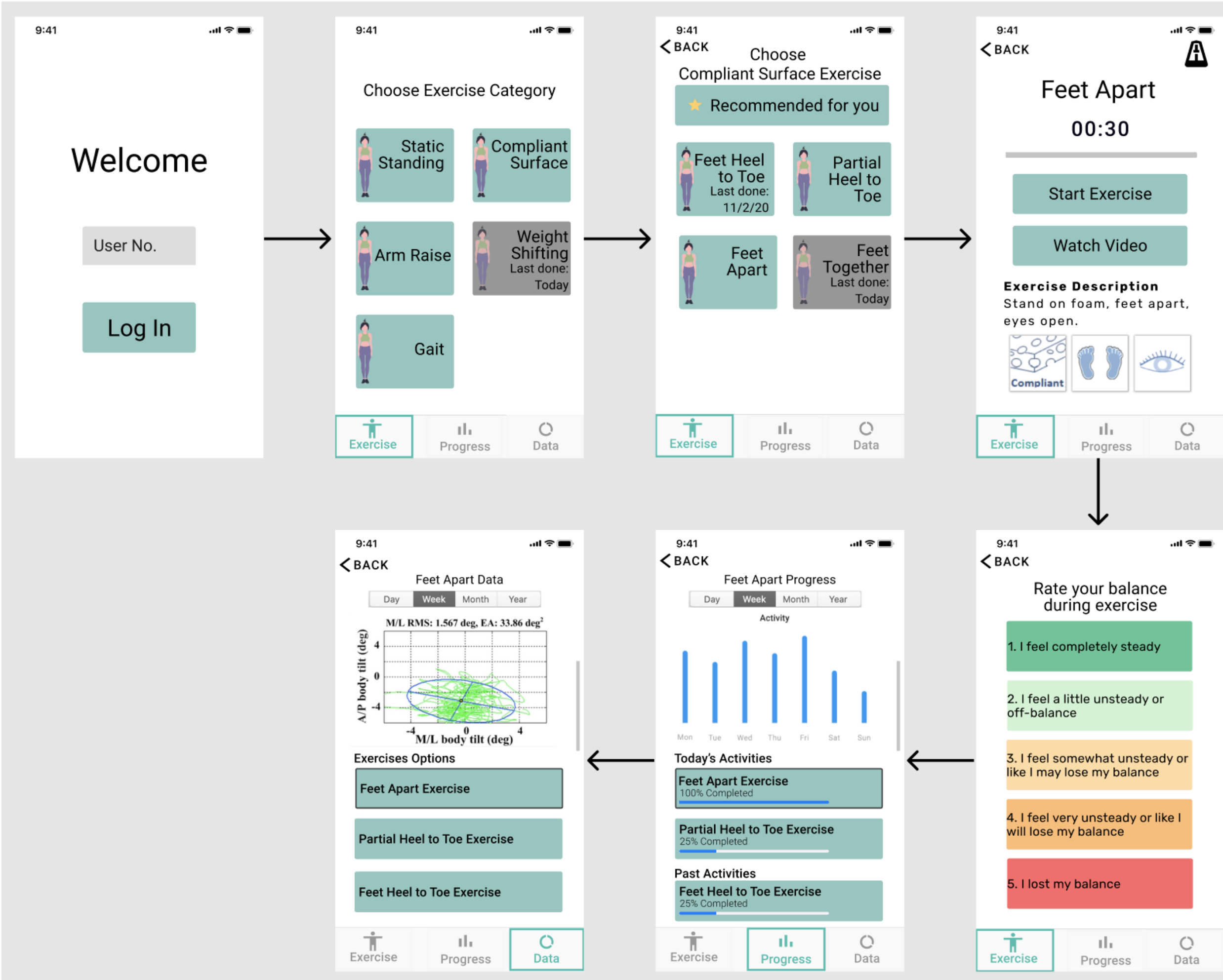
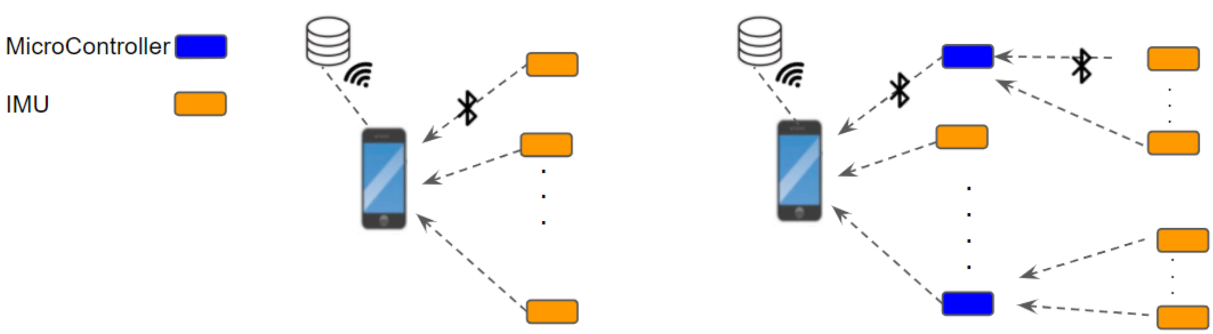


Figure 11: User Interface Prototype

Hardware-Software Integration (Sensor Connection Topology)



1.Direct Connections between Phone and IMUs

2.Indirect Connection with intermediate micro-controllers

Figure 12: Sensor Connection Topologies.

Several wireless body area network options have been explored for our system. Most proposed sensor network architectures are specifically built for laboratory environments, and thus, our options are constrained by market supply and only a few practical options can be chosen based on commercial availability.

The first method is connecting the IMU directly with the smartphone in a Bluetooth piconet [36], shown as the first diagram in Figure 12, and the smartphone would handle synchronization and data logging tasks. Several IMUs we have benchmarked using this connection architecture also provide SDK support for mobile platforms (XSENS DOT and Mbientlab MMR, etc). The software implementation would be relatively simple, as the phone exchanges information directly with the IMUs and sends a uniform request to start and stop the recording. One obvious downside is that the classic Bluetooth connection only allows at most 7 slave devices in one Bluetooth piconet, and as a result, the sensor number is not scalable.

One alternative method is instead of connecting the phone and IMUs directly, the system can include a microcontroller as an intermediate master node and form a Bluetooth scatter net [37], shown as the second diagram in Figure 12. The micro-controller would either combine multiple IMU readings into a single data stream or serialize the IMU data and send the data one at a time. By connecting the sensors, microcontroller, and smartphone in a tree structure, more sensors can be included in the system. However, the phone needs to ensure synchronization between the controller and IMUs and supports data streaming from the controllers; therefore, the app development for this topology will be more complicated. Additionally, most IMU suppliers using the second topology don't have SDK support for mobile platforms and we would have to build everything from scratch. Most importantly, the micro-controllers are usually external hubs or external access points, and the stakeholders explicitly mentioned that the system shouldn't include any external hub in our system. Therefore, our application development will revolve around the first connection topology.

Solution Development and Verification

Engineering Analysis

In order to evaluate and optimize our design, we conducted analyses that address the design drivers and design questions as shown in Table 9. The phone and IMU selection design driver is vital for our project as the smartphone has to be suitable for older adults' use and the IMUs need to meet the specifications to collect kinematic data of balance exercise. Hence, this design driver is of priority 1. The second design driver, which is the connection between IMU and smartphone, is also of priority 1 because we need to ensure that the IMUs and the phone are able to communicate with each other to collect and process kinematic data through the mobile application. To ensure that the communication is seamless, we also come up with SDK integration as the third design driver, with the same priority level. The last design driver with priority level 1 is the back end of the project. This back end infrastructure is crucial for our project as the back end stores the processed data and is the source for data visualization of our project. All the design drivers with priority level 1 are critical in ensuring that the system operates as needed.

The design drivers with priority level 2 are the user interface/experience of the mobile application and the risk analysis. The user interface driver is important to ensure that the application is user-friendly and easy to be used while preventing any bugs. The risk analysis is also crucial to assess the risk of the system and ensure the safety of the users. However, if the analysis for these design drivers fails, it would not destroy the whole concept solution.

Table 9: Design Drivers and Analysis

Priority	Design Driver	Design Questions	Analysis
1	Phone & IMU selection	• Which smartphone works best for this application? • Which IMUs will work based on specifications?	• Benchmarking Analysis
1	Connection between IMU & Phone	• How reliable is the bluetooth connection? • Will there be any latency issues?	• Bluetooth Analysis
1	SDK Integration	• What is the workflow between SDK and IMUs? • How compatible are the SDK functions with the rest of code?	• Unit testing for each sub-function. • Stress test and stability test over long period without crashing

			• Runtime analysis on response delay between phone and IMUs
2	UI/UX	• How intuitive and easy to understand is the interface? • Will there be any unresponsive gestures or app crashes?	• Formative Usability Testing
1	Back end	• How do we make sure our back end will collect all the data we need? • How do we organize the data for access later?	• Create an entity relationship diagram • Run tests on Firebase database
2	Risk Analysis	• How safe is the product for end users to use? • How do I minimize these concerns?	• Risk Assessment • Failure Modes and Effect Analysis (FMEA)

IMU Benchmarking

To make an informed decision on our IMU selection, a benchmark of commercially available IMUs was performed. A preliminary benchmark was done during the concept generation phase, which revealed 3 IMUs suitable for this project: Xsens Dot, Mbientlab MMR, and LPMS-B2. Since then, we populated the benchmark with more details such as accelerometer and gyroscope ranges and resolution. Additionally, existing IMUs previously used by the stakeholders such as the APDM Opal and Xsens MTw were added for comparison. The benchmark table can be seen in Appendix C.1.

Benchmarking is a great way to get an overview of the specifications of existing products on the market and comparing them to our requirements and specifications. It allows us to see the strengths and weaknesses of each product and make informed tradeoffs with regards to the price and functionality of the IMUs. Once an IMU is selected, it will determine the development platform and narrow down the smartphone choices based on the respective IMU's SDK requirements.

After a discussion with our stakeholders about the benchmark with regards to price, ability to simultaneously connect to a smartphone, accelerometer and gyroscope specifications, sampling rate, documentation, and lead time, the IMU selected was Xsens Dot. Even though our selection is based on specifications, there are other factors that cannot be determined purely from the specification sheet. One such factor is Bluetooth connection reliability. There are no research papers with regards to this specific IMU on connection performance, and while customer support has indicated that the IMU can connect up to 5 at once, we will be performing a Bluetooth analysis to quantitatively measure the Bluetooth performance.

Smartphone Benchmarking

Similar to IMU benchmarking, the same process was applied for smartphones. It gives us an overview of the specifications of existing products and allows us to compare them to our requirements and specifications to choose the best smartphone for the project. However, there are infinitely many smartphones to choose from based on our specifications, so we focused on two main operating systems: Android and iOS, and from that choosing a higher-end model and a budget model. Our final smartphone list includes: iPhone XR, iPhone 11, Samsung Galaxy A71 5G, Samsung Galaxy A51, and OnePlus Nord N100. The benchmark table can be seen in Appendix C.2.

One of the things we tried to find was the Bluetooth chip and accelerometer/gyroscope chip. This is to evaluate the performance of Bluetooth connections based on the smartphone and the performance of the smartphone's accelerometer/gyroscope respectively. However, phone manufacturers normally don't disclose this information. In the table, we've highlighted both specifications in red to signify that not enough information can be found. As such, we're not able to determine the Bluetooth performance or accelerometer/gyroscope performance of the smartphones.

From the benchmark, iPhones were eliminated earlier because they run on iOS, which requires a Mac to do application development. This limits the platform for future application development, especially when it is handed over to the researchers. From the rest of the Android smartphones, OnePlus was also eliminated because of its relatively unknown brand recognition in research, even though it is a budget smartphone. On the other hand, Samsung is a common sight in research. Of the remaining Samsung phones, we decided to choose the Samsung Galaxy A71 5G in the end because of better futureproofing. Even though it's more expensive, it has a better processor and RAM, which will be important when handling simultaneous Bluetooth connections. Additionally, the better processor and RAM can help maintain smooth operations of additional features in further iterations of the application.

Bluetooth Analysis

A Bluetooth analysis was performed to answer questions regarding the reliability of the Bluetooth connections between the smartphone and IMUs, as well as finding any latency issues. An empirical analysis was conducted with a total of two experiments, the results of which will tell us the optimum number of IMU connected to a smartphone and maximum distance between IMU and smartphone without losing data. Empirical testing is suitable for quantitatively measuring the Bluetooth performance because real-life testing allows us to take various factors that affect connectivity into account such as Wi-Fi (operating at 2.4GHz) and Bluetooth interference from other electronic devices.

<u>Experimental Setup</u>

This experiment requires three things: the smartphone, the IMUs, and a room (at least 9 m in length), as shown in Figure 13. Additionally, it requires a Wi-Fi/Bluetooth analyzer application to detect the presence of additional signals that may cause interference and the Xsens DOT application to collect balance data, shown in Figure 14.

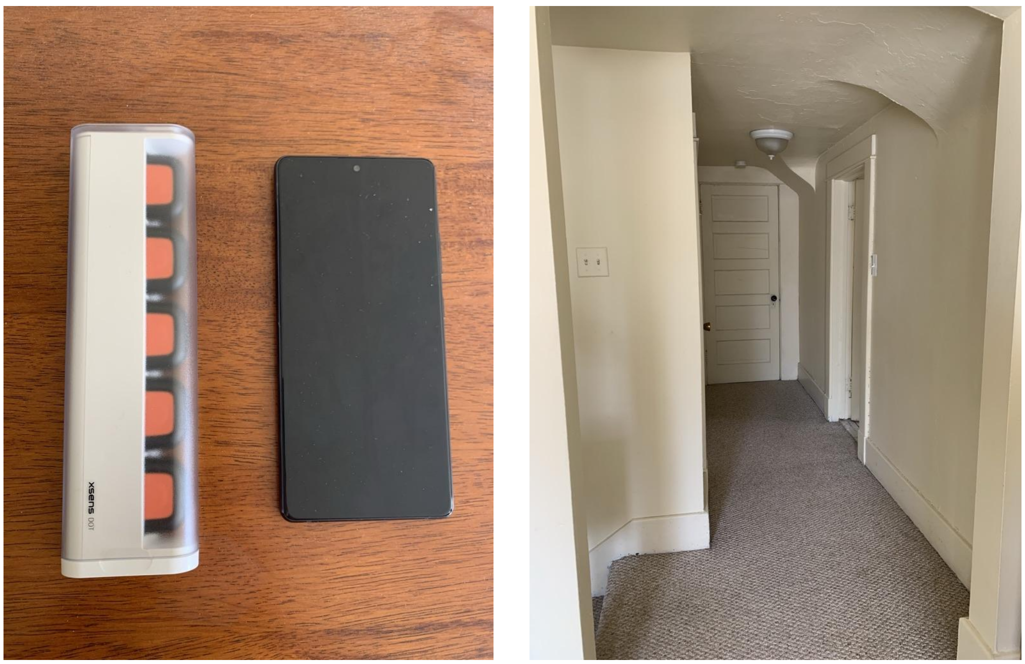


Figure 13: Experimental setup (from left to right) -- IMU, smartphone, and a room

Figure 14: Apps used for Bluetooth Analysis

Methods

Table 10: Detailed Bluetooth Analysis Procedure

Parameter	Varying Distance	Varying Number of IMU
Steps	1. Connect one IMU to the phone and attach it to the trunk 2. Stand 1 meters away from the phone 3. Perform one leg raise exercise for 30 seconds 4. Record data and bluetooth signal strength during the exercise 5. Repeat step 1 to 4, but with 2m, 3m, 4m, 5m, and 9m away from the phone	1. Connect one IMU to the phone and attach it to the trunk 2. Stand 1 meters away from the phone 3. Perform one leg raise exercise for 30 seconds 4. Record data and bluetooth signal strength during the exercise 5. Repeat step 1 to 4, but with 2 (add right leg), 3 (add left leg), 4 (add head), and 5 IMUs (add right wrist) connected respectively

Results

Using the Wi-Fi scanner, there were a total of 7 Wi-Fi networks operating at 2.4Ghz present in the room. Additionally, the Bluetooth scanner tells us that there were 23 Bluetooth signals present in the room (excluding the IMUs). The number of Bluetooth devices is more than double that of an average US household in 2020, which had around 10 devices [38]. As such, the experiment was performed at the extreme end of what the system will realistically be used at.

In terms of varying distance, there is a general trend of increased signal strength as the distance between phone and IMU is increased (Table 11). From analyzing the file size, the file sizes are within 9% of each other, with an exception at 4m, where the file size is 24.9% smaller than the average. However, seeing that no data loss occurs at longer distances, we can conclude that for 1-9m, the connection is reliable enough to transmit data without significant data loss.

In terms of varying the number of IMUs, the signal strength remained relatively constant with an average of -77.4 dBm (Table 12). There is no clear trend of signal strength as the number of IMUs increases. From analyzing the file size, they remain relatively constant with each number of IMU, with the largest difference being 6.3% with 2 IMUs. We can conclude that the number of IMUs does not affect the signal strength much, and with 5 IMUs connected, the connection is reliable enough to transmit data without significant data loss.

Looking back at our requirements and specifications for signal strength, a minimum of -78 dBm is required, based on research studying Bluetooth data loss [39]. For varying distances, none of the IMUs achieved this value, but it still manages to transmit data without significant data loss. On the other hand, for varying numbers of IMUs, the average signal strength meets this specification, and it does transmit data without significant data loss.

Table 11: Results of Varying Distance in Bluetooth Analysis

Distance (m)	1	2	3	4	5	9
Signal strength (dBm)	-80	-87	-82	-89	-92	-92
File size (KB)	288	285	278	219	273	298

Table 12: Results of Varying Number of IMUs in Bluetooth Analysis

Number of IMU(s)	1	2	3	4	5
Signal strength (dBm)	-80	-85, -72	-82, -71, -70	-82, -69, -73, -73	-85, -79, -83, -80, -77
File size (KB)	288	301, 320	322, 322, 320	303, 302, 300, 301	344, 344, 342, 339, 341

SDK Integration

Unit tests would be performed on all provided SDK subfunctions, including synchronization, recording, streaming, and calibration. For example, during a certain instance of streaming, data from an IMU may be missing due to signal interference or the IMU's internal failure, and our program may attempt to read the nonexistent data and append the invalid data to the end of the .csv file, leading to program crashing and file corruption. The same testing procedures from the Bluetooth analysis for each subcomponent and record the failure rate under different setups. Stress test and stability test would be applied to ensure that there would not be any compatibility issue or crashing until the sensors need to be recharged. Lastly, the team would also do runtime analysis on the maximum delay between phone and IMUs to derive an optimal timeout interval between subsequent operations and improve the overall program performance.

Database Accessibility Testing

Figure 15 below shows the entity-relationship diagram for the database. Each user represents an instance in the database. Each user instance also includes all the relevant attributes. The trial results are stored as a nested instance within the user instance and keep track of information on exercise name, category, timestamp, and relevant IMU data and measurement. The data from trial results will come mainly from IMU sensors and can be organized based on category and exercise for a fast query.

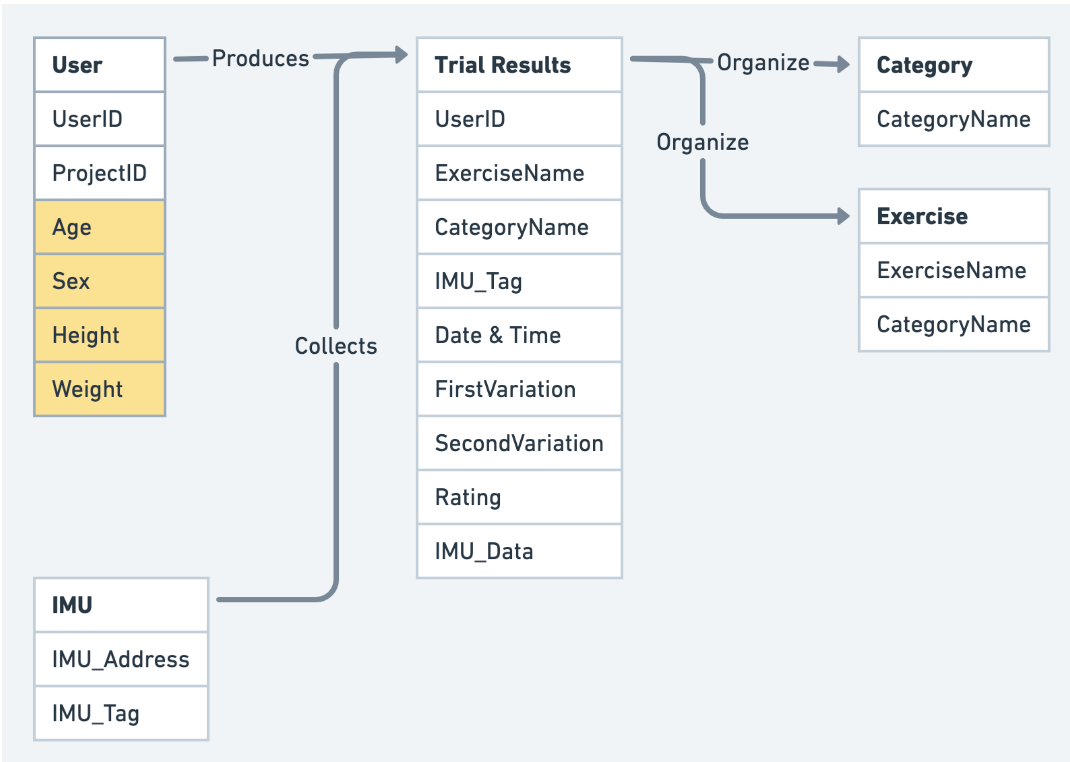


Figure 15: Entity-relationship diagram of the balance data. Attributes in yellow are not stored in the cloud.

Relating this back to the requirements and specifications, since the back end system will spin up a new instance each time a new user is introduced, this structure allows us to identify each user separately, the timestamp attribute in the trial results instance also gives us a way to map out users' activities. There have been concerns raised regarding sensitive information being stored in the database that can act as an "identifier" for a particular user. The team has proposed an alternative that works around storing sensitive information while collecting required information. Some of the attributes in the database entity diagram are highlighted in yellow to indicate this shift.

The accessibility testing for the database aims to provide functionality and performance reassurance on the established link between the user interface and the array of systems involved in back end development. In the case of our application, since the database is hosted in the cloud, the testing also reflects key performance metrics of the cloud infrastructure, specifically response time, availability, and reliability. We used Firebase as our main database, which is part of the Google Cloud ecosystem of products. Early testing can be conducted by checking the communication channel between the application and the server. We want to access if the application can successfully request data from the server, which is a read request, and subsequently modify existing data on the server, which is a write request while monitoring latency, throughput and capacity. More advanced test cases need to be fed into the system throughout the development process via a command line interface. Figures 16 and 17 below show the dashboard of Firebase and the sample instance created in the database.

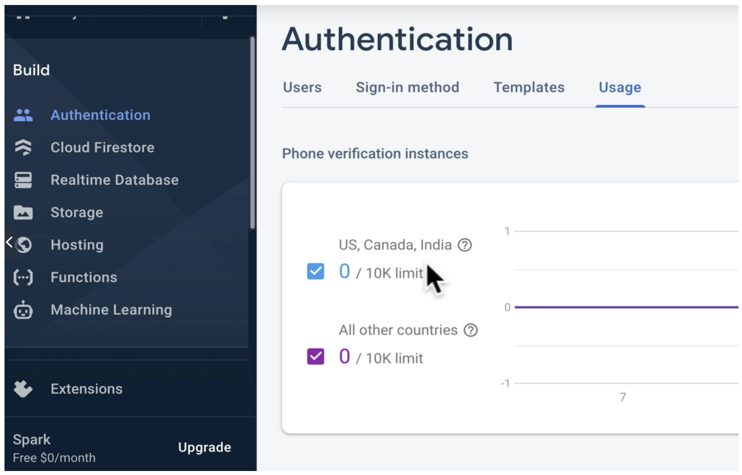


Figure 16: Dashboard of Firebase showing that the connection between the app and cloud has been established, but with 0 usage.

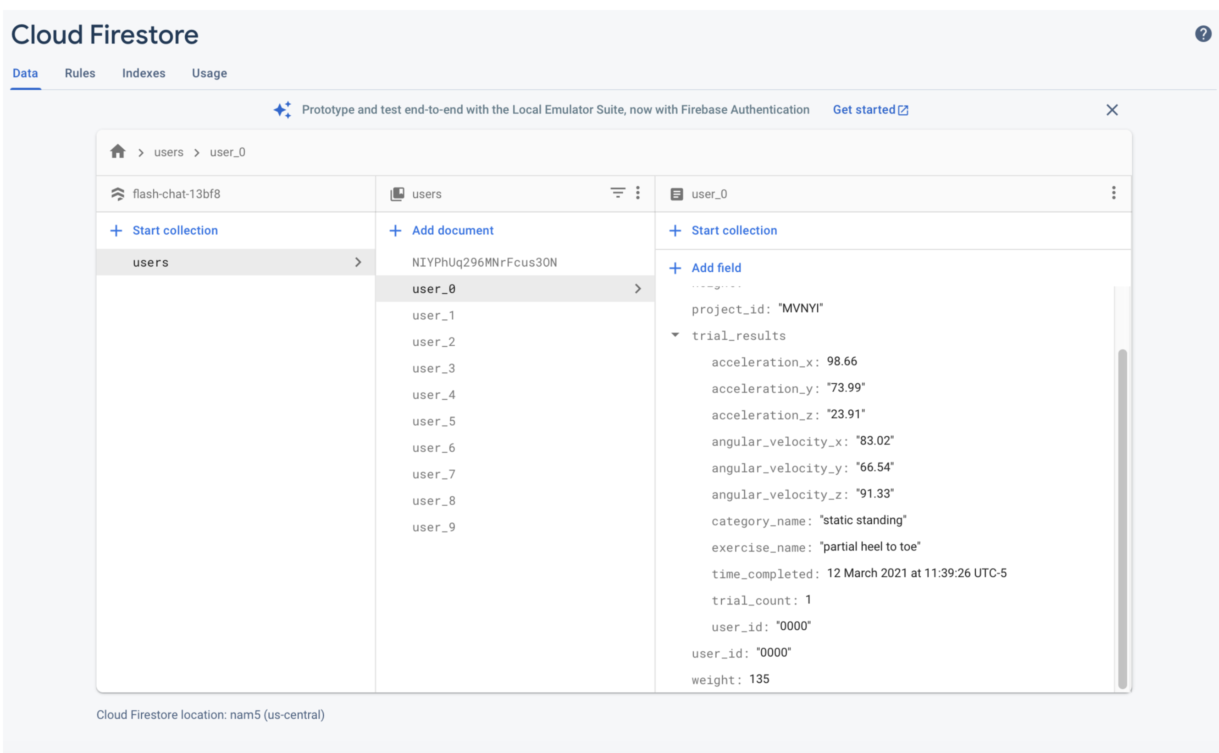


Figure 17: Implementation of the entity-relationship diagram in a Firebase instance.

Formative Usability Testing

The formative usability testing was conducted to evaluate the user interface design of the mobile application by identifying improvement opportunities and confirming that it is evolving in the right direction. This was conducted during the development phase of the application to enhance the user experience and success rate of the application. The testing plan for both in-person and remote testing is shown in Table 13.

Table 13: Test plan for formative usability testing

Background	The testing will be conducted after the main functionalities are ready to be reviewed. This qualitative testing is to analyze how intuitive and easy to understand the interface design is.
Purpose	The usability testing for the user interface design of the application is conducted to identify the interface strengths and weaknesses as well as study the user's interaction with the design. The information from this test will be used to improve the usability and experience of the design. This will also be a way to identify any hazards that might have not been detected.
Test Apparatus	For in-person testing, the items needed for the testing include the IMUs, a microphone, a laptop, and a smartphone. Our focus for the testing is on the interface and the IMU can be included to perform an additional task. For remote testing, the item needed is a laptop with Zoom and Android Studio installed.
Participants	The participants for this testing will be the members of the team and/or their close friends. Although the end users are older adults, for this project, the main user will be the researchers.
Test Environment	● In-person testing ○ Setup the laptop to cast the phone screen, screen, and audio record. ○ The phone will be connected to the laptop and the laptop screen will be dimmed to make it a more realistic environment ○ The participants will be provided with a list of tasks to complete and they are instructed to think aloud throughout the test. ○ The whole test will be recorded for review. ● Remote testing ○ Setup the laptop so that it will screen share the emulator during Zoom call. ○ The participants will be given remote control access to the screen to test the interface. ○ The participants will be provided with a list of tasks to

	◦ complete and they are instructed to think aloud throughout the test. ◦ The Zoom call will be recorded for review.
Tasks	• Think aloud and give comments on the text, icons, and any visuals in the design • Log in with provided user id and a random user ID • Choose exercise category and exercise options • Play the exercise videos and read the instructions • Start and stop an exercise; use the metronome (if applicable) • Rate yourself after each exercise • View your exercise data and progress • Connect and synchronize the IMUs at different settings (in-person testing)
Data collection	From the recording, the participant's subjective feedback, uncertainties, difficulties in carrying out the tasks, suggestions, and the task failures will be collected.
Data analysis	The data will then be analyzed by identifying similar dislikes, suggestions, and preferences from different participants. The dislikes will be improved and preferences will continue to be applied in the design. The suggestions will be reviewed with all the members to discuss the UI's benefits and disadvantages.

The formative usability testing session was performed remotely through Zoom. It was conducted with three members of the sponsor team and a physical therapist. All the comments and responses were evaluated and the root causes of the use error were identified. To improve the usability of the mobile application, modification steps to address the errors are planned as shown in Table 14 below.

Table 14: Summary of the responses from formative usability testing

Use Error	Root causes	Modification / Next steps
Pressed the wrong button	• IMU name not specific and unclear • Metronome is confusing • IMU battery is always there	• Change IMU • Remove metronome icon
Misread parameter on display	• No descriptions on data visualization • Data is hard to interpret and understand	• Add descriptions on normative data obtained • Add standard deviation on figures (nice to have) • Reconsider data metrics and

		work on data processing
Selected wrong option	● Tab names are confusing ● Toggles have no label	● Change tab names ○ Description to Video (timer/exercise screen) ○ Remove toggle

Additional Comments	Next Steps / Modification
Exercise options in order of difficulty	● Ask for balance training exercise difficulty table ● Update list accordingly
Differentiate options and exercise ● Sign Out button ● Exercise in progress tab	● Change colors of buttons ● Progress screen ○ Add exercise trial details
Include variation options in app and multiple trials right after another	● Add another screen after selecting exercise for variations
Users should not be able to change frequency range	● Remove the frequency range option and set a value as default
Yearly progress	Progress over time (nice to have)

Risk Assessment

In order to reduce the risks of the smartphone-based balance training platform, we conducted two different risk analyses. The first analysis is risk assessment where potential hazardous situations are identified, as shown in Table 15. The likelihoods of the situations to happen were predicted and ways to reduce the impact or eliminate the hazard is developed. The second analysis is the Failure Mode and Effects Analysis (FMEA) where the failure modes, effects, and impacts of the risks of each component are assessed, as shown in Table 16. The Risk Priority Number (RPN) is also assigned by calculating the product of Severity (S), Probability (P), and Detection Rate (D).

The main components of the solution are mainly commercial products that meet necessary compliance. Hence, the ways to mitigate the risk and reduce the impacts are mostly from the information received from the manufacturer. We also acknowledge that the combinations of the different components in our solution might cause additional risk. From the analysis, the overall risk of the solution is low given that it is used according to the manual and appropriate testing is completed thoroughly.

Table 15: Risk Assessment

Hazard	Hazardous situations	Likelihood	Impact	Steps
Overheat	When the user is charging the IMU, the battery is overheated	Low	Serious	Charging is stopped to protect battery and IMU [40]
Irritation	The friction between the IMU and skin during exercise might cause irritation	Low	Serious	Place the IMU in the pocket and straps properly
Fire / Explosion	When the IMU is charged or stored at high-temperature environment, the IMU could explode because it has lithium battery	Low	Serious	Store in low humidity and temperature. Avoid charging in high-temperature environments [40]
Damage to IMU	When the straps are being put incorrectly, the IMU can slip out of the pocket or fly off during exercise and breaks the IMU	Low	Serious	Ensure the orientation of straps and position of IMU is correct

Table 16: Failure Modes and Effect Analysis (FMEA)

Comp-onents	**Function**	**Failure Modes**	**Effects of failure**	**S**	**P**	**D**	**RPN**	**Cause of failure**	**Prevention steps**
Body straps	Attach IMU to the body parts for data collection	Loose straps	IMU will not be in place	3	3	2	18	Fatigue / Wear	Make sure the end of the strap is pointed away from surface with friction, such as between the legs, or the inside of the wrist [40]
		Too tight	Circulatory	4	3	1	12	Grip of	Ensure snug fit

			issues / Discomfort					straps	but avoid over-tightening
IMU	Collect data	Overheating	Burn	7	1	1	7	Overheated battery	Charging is stopped to protect battery and IMU [40]
		Calibration interference	Inaccurate calibration	6	2	2	24	Strong magnetic field	Do not expose sensor to strong magnetic field [40]
		Fire/Explosion	Damage to IMU and user	10	1	1	10	Lithium battery	Store in low humidity and temperature. Avoid charging in high temperature environment [40]
		Error in measurement	Inaccurate measurement	6	2	2	24	Incorrect placement	Place the sensor where there is either less muscle or soft tissue and the sensor has a flat surface to adhere onto. Make sure it is put horizontally. [40]
		No IMU connection	No data collected	8	2	2	32	Connection limit/Loss of connection	Reduce sensor number connection or payload [40]
		IMU not synchronized	Data collected is not synchronized	7	2	2	28	Clock domain is not time synced	Synchronize the IMU prior to putting them on the user. The closer the IMUs to each other, the easier for them to synchronize. Alternatively, reduce the number of IMUs

									used.
Applic-ation	Interacts with user	Freezing	App stops responding and the user will not be able to perform any operations.	8	2	2	32	Low memory capacity, high CPU usage	Manage memory and app testing; design app with lightweight framework [41]
		Crashing	Improper output, problems with data and software, loss of data	8	2	2	32	Software failure	Thorough app testing, cloud data sync & backup [41]

Detailed Design Solution

Our design solution consists of hardware and software solutions (Figure 18). It is currently a proof of concept of a smartphone-based balance training platform that will be developed further in the future. It is intended for research purposes instead of commercial use. The main users at this stage will be the sponsors rather than older adults. The solution only implements a few critical features such as logging IMU data and provides the building block for future features. Most notably, the machine learning algorithm is absent in this design solution, but variables and inputs relevant to the algorithm have been set up.

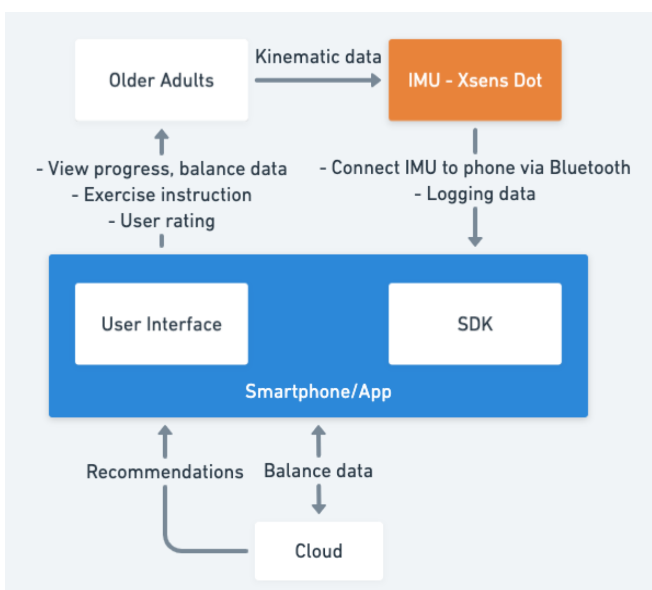


Figure 18: Integration between hardware and software solutions.

Hardware

The hardware solution is built up of three key components: the smartphone, the IMUs, and the straps (Figure 19 and 20). For our smartphone, we went with the Samsung Galaxy A71 5G. It provides us with a fairly modern CPU (Exynos 680 @ 2.2 GHz) and a lot of RAM (6 GB) to ensure smooth performance when handling simultaneous Bluetooth connections and writing/reading data to/from the cloud. For our IMU, we went with the Xsens Dot. This IMU fulfills our requirements and specifications as it provides us with the capability to connect more than four IMUs at once and has sufficient battery life for our purpose. It also has good SDK documentation for us to use when developing the software solution. For our straps, we went with the straps provided by Xsens that are compatible with our IMU. The straps feature three sizes: long (128 cm), medium (55 cm), and short (29 cm). These lengths were verified and were found to fulfill the median and lower range of anthropometric data of relevant body parts.

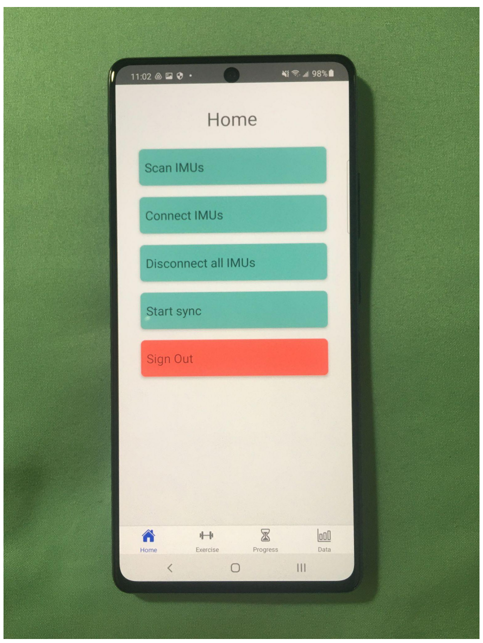


Figure 19: Samsung A71 5G and the balance training application.

Figure 20: Xsens Dot IMUs in dock and the provided straps.

Figure 21 shows the real-life view of the physical design solution. A conceptual view of the location of the IMUs is provided for clarity (Figure 22). The IMU locations are as follows: head, right wrist, trunk, right leg, and left leg. Additionally, the IMU tags have been named with their respective location i.e. the tag for the IMU attached to the head is called “Head” and the tag for the IMU attached to the left leg is called “Left_Leg”.

Figure 21: Real life view of the physical design solution.

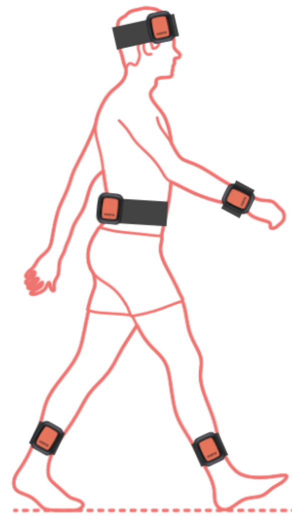


Figure 22: Conceptual view of the location of the IMUs. They are located at the head, trunk, right wrist, right leg and left leg.

To ensure accurate measurements, the IMU locations have been specified based on discussions with the sponsors and the Xsens Dot documentation [40]. When wrapping the straps, make sure that it is a snug fit, but not too tight. This prevents the IMU from moving around when doing the exercises while also preventing circulatory issues or discomfort for the users. Additionally, when inserting the IMUs in the pocket of the straps, ensure that it is inserted horizontally instead of vertically for a snug fit (Figure 23). For the head, the IMU is placed at the forehead level and on the right side using the medium-length strap (Figure 24(i)). For the trunk, the IMU is placed on the lower back at the center using the longest strap (Figure 24(ii)). For the right wrist, the IMU is placed a few centimeters above the wrist to allow for full hand motion without disrupting the IMU (Figure 24(iii)). The IMU should be facing outwards. The shortest strap was used for the right wrist. For the right and left leg, the IMU is placed a few centimeters above the ankle to allow for full foot motion without disrupting the IMU. The IMU should be facing inwards. The short- and medium-length straps were used. Ensure that the end of the strap is not facing inwards to prevent the straps from unraveling during the exercise (Figure 24(iv)).

Figure 23: The IMU should be placed horizontally instead of vertical in the pocket of the strap to ensure a snug fit.

(i)

(ii)

(iii)

(iv)

Figure 24: The exact location of IMU with regards to the (i) head, (ii) trunk, and (iii) right wrist. Additionally, (iv) ensure the ends of the straps are not facing inwards to prevent it from unravelling during exercise.

Software

The software solution consists of several components, shown in Figure 25. For our front end, we used React Native. React Native allows for a much faster pace of development for the user interface and has extensive third-party libraries for us to utilize, such as that for YouTube videos and data visualization. For the back end, we used Xsens Dot Android SDK to handle communication with IMU and Firebase for communication with the cloud. Here, we will present the high-level functionalities of the application and its relation to the requirements and specifications. Technical documentation is provided on GitHub and Appendix D.

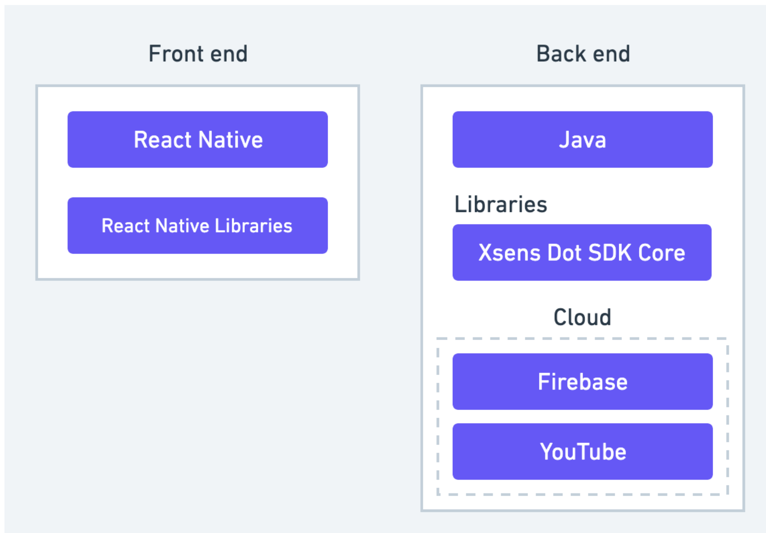


Figure 25: Software stack of the application.

The software flowchart (Figure 26) shows the processes the application goes through. As per our requirements and specifications, the software identifies each user separately, tracks exercise progress, captures user rating, has file export capabilities, shows data visualization, and uploads the data to the cloud. A requirement that was not achieved was providing machine learning-based exercise recommendations as mentioned in the introduction. This is currently out of scope for our project, but implementation in the future is recommended as more data is gathered and patterns begin to emerge with regards to what constitutes a good performance of an exercise.

Front End

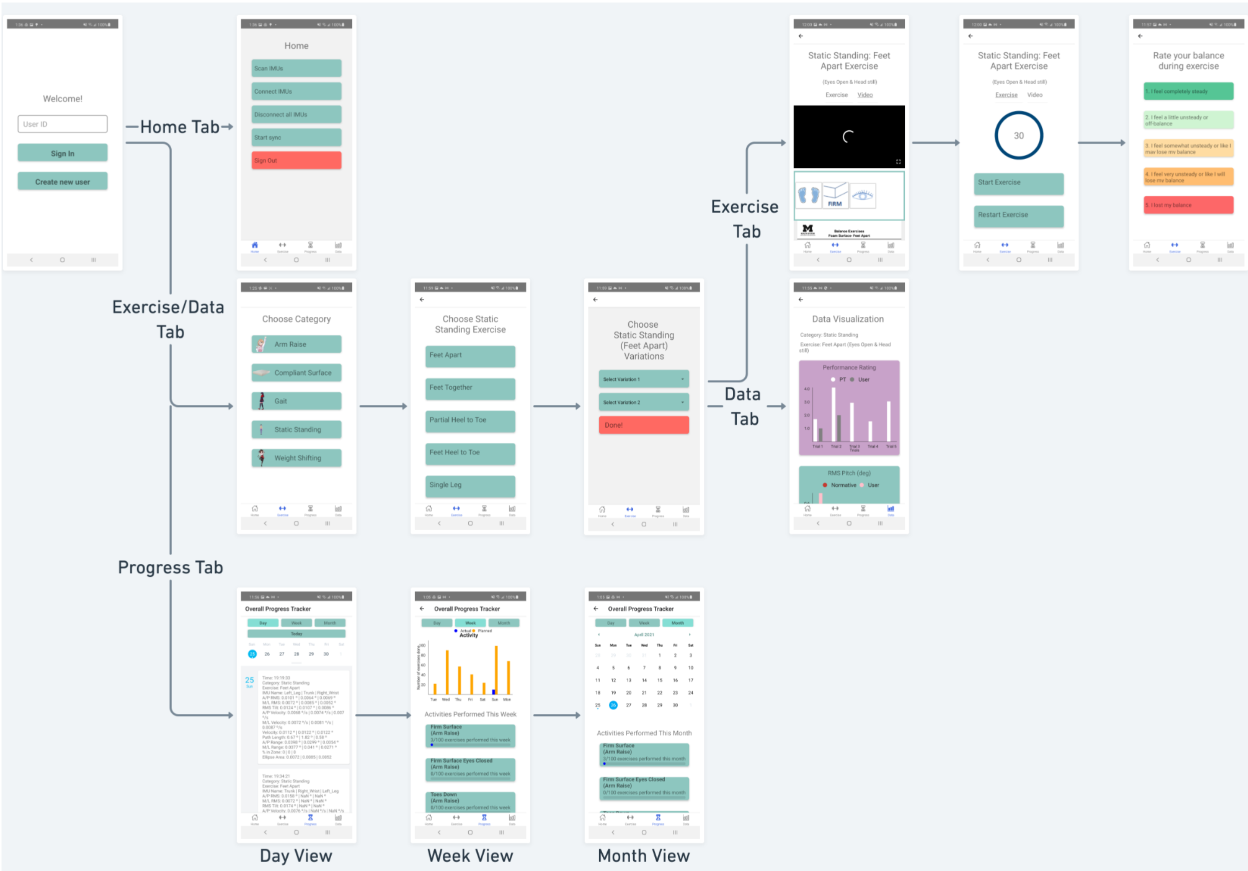


Figure 26: Software and interface flowchart

At the beginning of the app, a login screen appears (Figure 26). To identify each user separately, we store the list of users in our Firebase database. The user ID is just a string in Firebase, so it can be set to any alphanumeric combinations desired, as stated in our specifications. As of now, we are using a 4 digit number format (example: 0001) for testing purposes. When the user inputs a user ID, the application checks with Firebase whether the user exists or not. If the user exists, the user will enter the application. If the user does not exist, then a dialog box will appear letting the user know that the user ID is invalid. The user can then proceed to create a new user.

When the user wants to do an exercise, they are presented with the category of exercises at the top level in the “Exercise” tab. By choosing a category, the user can then refine by choosing a particular exercise, then selecting the variations for that exercise (if applicable). The user can then track the current exercise progress. A timer is available to let the user know of the remaining time left of their exercise. Start and restart buttons are also present for the user to initiate and repeat the exercise. When the user presses the start button, it will change to a stop button, a 3-second buffer is included in the form of “Ready, Set, Go!” before starting the timer and data collection process. Upon pressing the restart button, the IMU will stop recording the data and the timer will reset back to the original time.

At the end of the exercise, the user will be prompted to rate their perceived performance. The rating screen uses the analog rating scale of 1-5 [5]. Once a rating is selected, the processed balance data, rating, category name, exercise name, and variation are sent to the cloud. Raw IMU data is also saved locally on the smartphone. An alert will then appear where the user can choose whether to repeat the same exercise or proceed to the next exercise category.

For data visualization, we have implemented two different types of visualizations. One is for progress tracking, that is, to see how someone has been doing their exercises with respect to time in the “Progress” tab. The other visualization is comparing the user’s processed balance training data with normative data from existing literature under the “Data” tab. As the normative data is only available for certain IMU locations and certain exercises from the Static Standing category, the visualization has only been implemented for those. However, with these implementations in place, it will be easy to replicate the visualization for other IMU locations and exercises.

As mentioned previously, user data and processed balance data, as well as relevant metadata such as category and exercise name, are stored on the cloud.

With regards to file export capabilities, the application already saves the raw balance performance data in a .csv file in the smartphone’s internal storage. One .csv file contains Euler angle, free acceleration, and angular velocity in the X-Y-Z plane for one IMU. As such, a single exercise performed will produce five .csv files if five IMUs were used. One omission in our application is the quaternion (as specified in the specifications). This is due to a hardware limitation, as the Xsens DOT cannot output Euler, quaternion, acceleration, and angular velocity at the same time. At most, it can provide three of these at once. However, because quaternions can be converted from Euler angles, its omission can be dealt with by introducing additional calculations for quaternions in the code. If for whatever reason the currently selected combination of metrics is unsatisfactory (Euler angle, free acceleration, and angular velocity in the X-Y-Z plane), a change of two lines in the code can provide different metrics to measure from the IMU. More information about this can be found in the documentation.

Back End

For communicating with the cloud, the back end uses Firebase. We mainly use the Firestore database to store the processed IMU data (Firestore is one of Firebase core services among others such as Authentication, Hosting, etc.) shown in Figure 27.

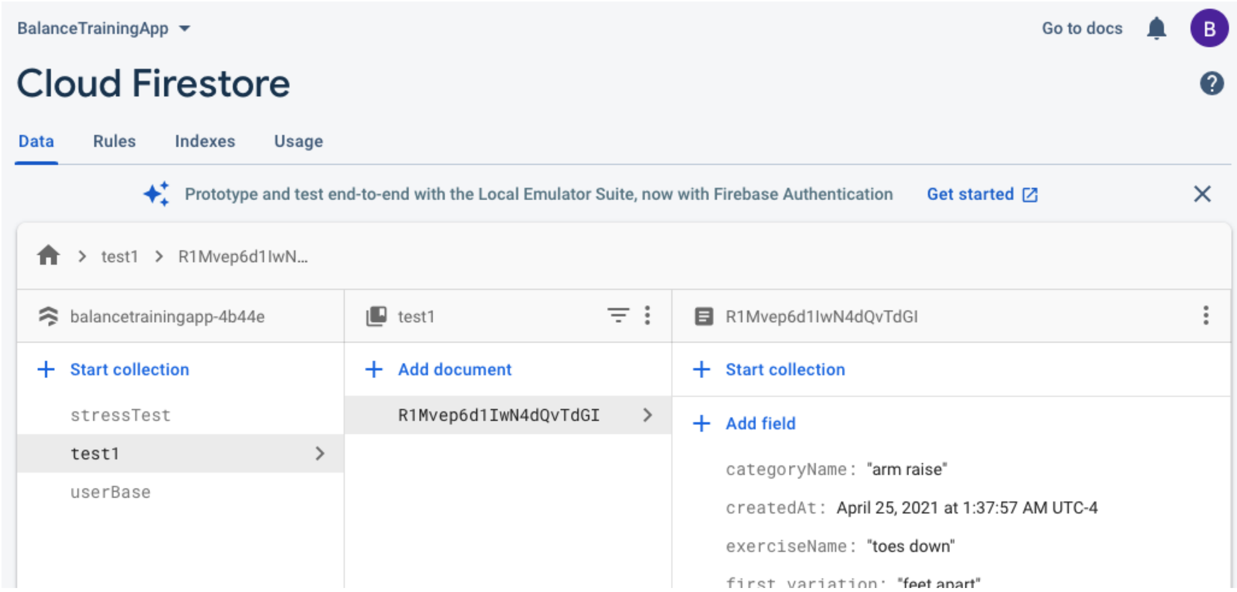
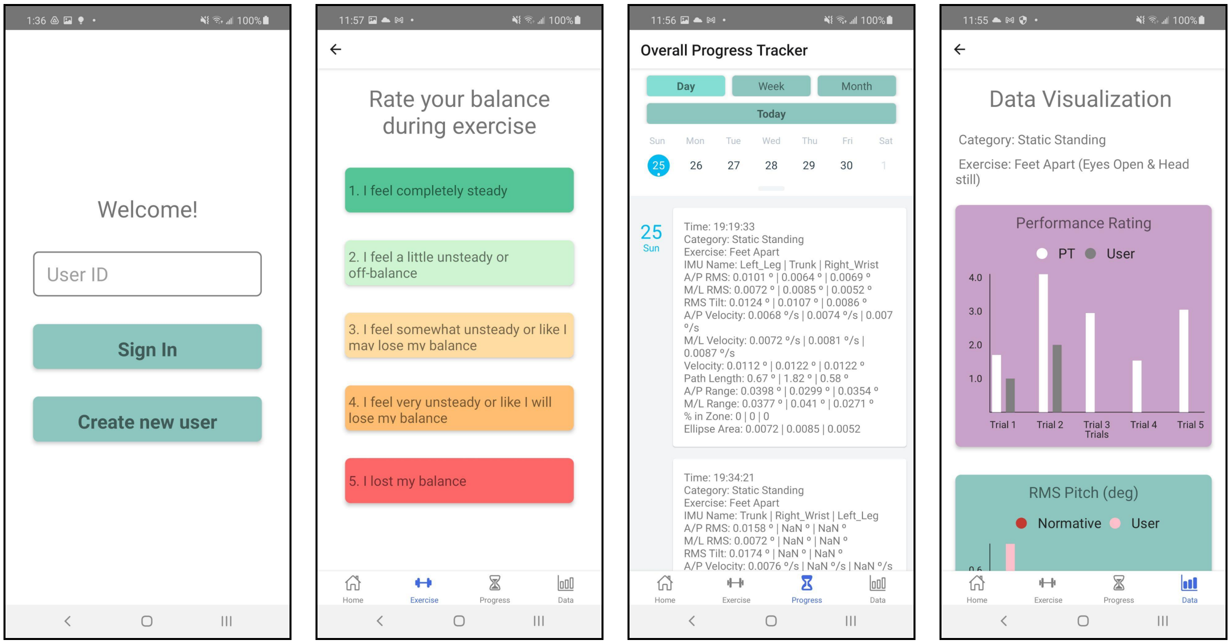


Figure 27: Screenshot of the Firestore database, storing our processed IMU data.

Data storage in Firestore starts with "collections" (on the left). Going one layer deeper, we have "documents" (in the middle). Going another layer deeper, we have "fields" (on the right). In other words, we can have a collection or multiple collections at the root level, where each of the collections hosts documents, which then hosts fields.

In the application, there are four main points where we need access to the data in the cloud (Figure 28):

i. The authentication screen, to check whether the inputted user ID (a) exists in Firebase before letting the user "Sign in" or (b) doesn't exist in Firebase before allowing "Create new user"
ii. The rating screen, where the IMU data needs to be uploaded to the cloud after the user has performed a particular exercise.
iii. The progress screen, where we need to calculate parameters such as medial/lateral velocity (mlVelocity), anterior.posterior RMS (apRMS) etc. to give a historical overview of users' performance to date on a monthly, weekly, and daily basis.
iv. The data visualization screen, where we again need IMU data to calculate parameters to be plotted against normative results.



i) Authentication screen ii) Rating screen iii) Progress screen iv) Data visualization screen

Figure 28: Screens where data is accessed from cloud

With that, it seems like a reasonable way to go about this is to have the database structured as shown in Figure 29 below, which is to have each user as a separate new collection. Under each collection (a.k.a. each user), we will have each document representing one exercise, and then logically in that document (a.k.a. exercise), we will have different fields such as “categoryName”, “timestamp”, etc. to store the values associated with that exercise. The processed IMU data is stored in “imuData”.

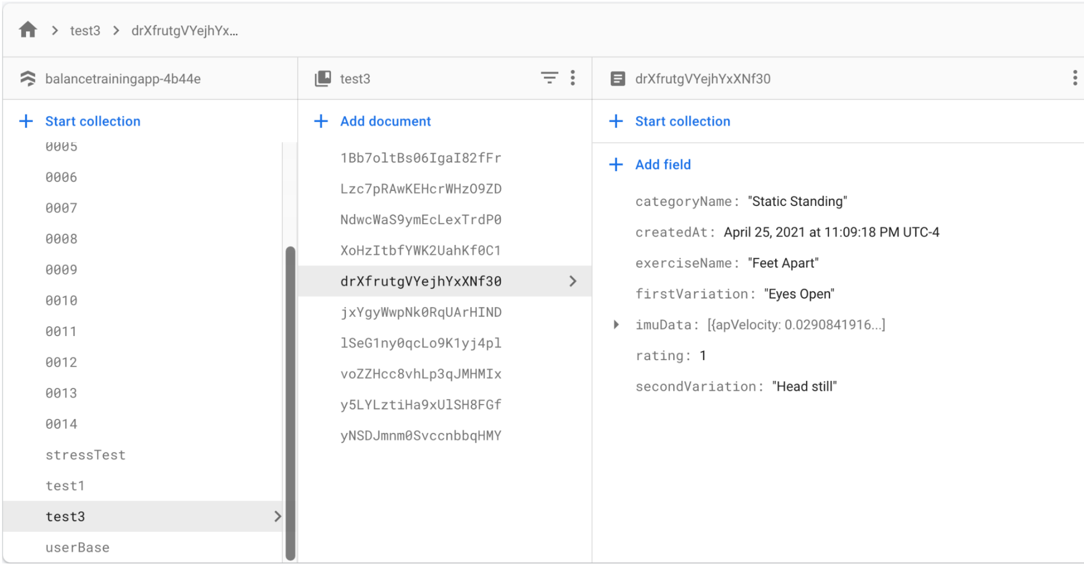


Figure 29: Screenshot of the database structure for the application.

Verification

Verification tests were performed to verify that our design solution meets the requirements and specifications set in the beginning, summarized in Table 17. In general, we are confident that our solution meets the requirements and specifications. Most of the verifications have been performed except for the summative usability testing. We have written a high-level plan to perform the summative usability testing in the future.

Table 17: Verification tests performed with respect to the requirement.

Requirement	Verification
Pairs multiple IMUs to smartphone	Data verification
Has sufficient battery life	Battery stress test
Is adjustable to different body shapes	Strap measurement
Is well documented	Code inspection
Captures user rating and recommend exercise	Summative usability testing
Tracks exercise progress	
Includes instructions of exercise	
Visualizes balance training data	
Automatically uploads balance data to cloud	Firebase usage test

Data Verification

To demonstrate that the IMUs are functioning properly and are connected to the smartphone successfully, a data verification was performed by comparing body-based kinematic data to the normative data from existing literature. For this verification, the exercise choice is for the Feet Apart exercise, which is in the category of Static Standing, with the variation of Eyes Open and Head Still. The kinematic data were collected and processed before comparing it to the normative data. Since the IMU placement in the normative data paper is at the trunk, the kinematic data compared to it is also the data collected from the IMU positioned at the trunk. A more detailed explanation of the normative data used is in the discussion section on pages 71-73.

The metrics are the mean values of anterior/posterior RMS value (apRMS), medial/lateral RMS value (mlRMS), anterior/posterior acceleration value (apAcceleration) and medial/lateral acceleration value (mlAcceleration). As shown in Table 18 below, the apRMS, mlRMS, apAcceleration and mlAcceleration values of the kinematic data are within the range of the normative data values. From this comparison, we can conclude that the data from IMUs are able to connect successfully and are functioning as expected.

Table 18: Mean (SD) of normative data and mean of kinematic data from two trials

Data	Normative	Trial 1	Trial 2
apRMS (°)	0.39 (0.18)	0.74	0.55
mlRMS (°)	0.13 (0.08)	0.13	0.19
apAcceleration (m/s^2)	0.07 (0.03)	0.03	0.03
mlAcceleration (m/s^2)	0.02 (0.01)	0.03	0.03

Battery Stress Test

The specifications require the battery life of the IMUs to be at least 3 hours in order to be usable inside the lab for a balance training session. A battery stress test was conducted to determine the maximum battery life of each of the five IMUs. The test allows us to simulate the conditions under which our design will be used by choosing the appropriate parameters. The test was done by streaming data from the IMU using the official Xsens Dot application until the battery dies. The payload type chosen was the same as the one used in our application, which is "Custom Mode 1", providing Euler angle, free acceleration, and angular velocity. Note that this payload type is the most intensive payload type out of the other options as it provides the most information from the IMUs. The sampling rate was set to 60 Hz, which is also the same as our application. The battery of the IMUs was checked every 30 minutes. The result of the test is shown in Table 19 below.

Table 19: The battery life of each of the five IMUs across time.

Time (h)	Battery life (%)				
	Head	Right Wrist	Trunk	Left Leg	Right Leg
0.5	100	100	100	100	100
1	92	90	90	91	92
1.5	85	83	82	84	85
2	77	74	72	75	77
2.5	67	64	62	66	68
3	59	57	54	58	60
3.5	53	51	47	51	54
4	47	44	38	45	48
4.5	39	35	27	37	42
5	28	24	16	26	32
5.5	21	17	10	20	25
6	13	11	5	13	17
6.5	8	6	0	8	11

The battery stress test reveals that the IMUs have plenty of battery life for our purpose. At the 3 hour mark, the battery life was still above 50%. It only starts dipping below 50% at around 3.5 hours. Furthermore, the shortest battery life of the IMUs was just above 6 hours. We can conclude that the specification for IMU battery life is verified, and a safety factor can be incorporated into our design solution. Note that with regards to IMU usage time, we are only considering streaming the IMU data. If we were to record the IMU data on the IMU itself, the storage capacity of the IMU would become the limiting factor, cutting our usage time to 88 minutes (based on the Xsens Dot user manual [40]). However, because our design solution doesn't actually record IMU data locally, this limitation can be ignored.

Strap Measurement

To ensure that the straps are adjustable to different body shapes, we measured all five of the straps that we purchased alongside the Xsens Dot IMUs. The lengths of the straps were already provided by the manufacturer, but measuring them again allowed us to verify that the lengths were true and within our specifications based on anthropometric data. The straps were measured using a home measuring tape, so the accuracy may be less than ideal. The lengths measured were the maximum length of the straps, which could then be adjusted to a smaller value by using velcro to tighten the fit. The measurements are provided in Table 20 below.

Table 20: Strap measurements of different lengths, compared to the provided measurements and relevant anthropometric data.

Strap length	Actual measurement (cm)	Provided measurement (cm)	Relevant anthropometric data (cm)
Longest (1x)	128	128	Waist: 82.8 - 136.3
Medium (2x)	55	55	Head: 51.9 - 59.7 Ankle: 19.3 - 24.0
Shortest (2x)	29	29	Ankle: 19.3 - 24.0

When compared to the relevant anthropometric data, the strap lengths fall within the range (except for the shortest strap for the ankle). It does not fulfill the upper end of the spectrum, which may reduce the number of people who are able to use this design. Additionally, the anthropometric data does not include wrist circumferences, which would be relevant for the shortest strap. However, it is likely that the wrist would be smaller than the ankle, which we know the shortest strap can accommodate based on the measurements.

Code Inspection

Code inspection is conducted to ensure that our code meets the requirements and specifications. Particularly, there should be at least 1 comment per function and a readme file to explain the structure of the project. After reviewing the code, there is indeed at least one comment per relevant function explaining the purpose of it (Irrelevant functions are empty functions that are just required by some packages but don't provide any purpose for us). Additional comments are placed inside the body of the function when necessary. Figure 30 shows an example of a well-commented function.

```java
/**
* Close the data output stream.
*/
private void closeFiles() {
    mIsLogging = false;
    for (HashMap<String, Object> map : mLoggerList) {
        // Call stop() function to flush and close the output stream.
        // Data is kept in the stream buffer and write to file when the buffer is full.
        // Call this function to write data to file whether the buffer is full or not.
        XsensDotLogger logger = (XsensDotLogger) map.get(KEY_LOGGER);
        if (logger != null) logger.stop();
    }
}
```

Figure 30: A well documented function, which provides the purpose of the function as well as additional comments to understand how the function works.

A readme guide has also been provided along with a full documentation. The project structure, a setup guide, as well as relevant libraries and deeper exploration of the functions are included in the documentation. The documentation can be found in Appendix D.

Additionally, we have performed a knowledge transfer with the sponsors, giving them an overview of our code. Even though the code inspection has not been completed at that point, the comments and documentation that has been written were positively received by the sponsors.

A limitation of our code inspection is that it has only been done by the members of the team, which means that the quality of the comments cannot be measured. There could be assumed knowledge that may have not been realized in text due to not having a third party code inspection. We have provided a thorough documentation to the best of our ability, but the caveat still stands.

Summative Usability Testing Plan

The analyses performed assess functionality of specific components of the final solution. A summative usability testing is to be conducted in order to evaluate the functionality and usability of the solution as a whole. However, due to time limitation, the usability testing could not be conducted by our team members. Hence, a summative usability testing plan is provided as a guide for the research team as shown in Table 21 below. The plan includes both an ideal case and small case plan. It also includes the possibility that either the participants could be either from the research team or other user groups. There are additional tasks for the participants from the research team as we assumed that the solution will be used in a research setting.

Table 21: Summative usability test plan

Background	The summative usability testing should be conducted prior to applying for regulatory clearance to test any unusual malfunctions. This is to obtain objective evidence and validate that the solution is safe to be used.
Purpose	This testing is conducted to evaluate the usability of the smartphone-based balance training platform as a whole. This is to confirm that the users can interact with the platform in a safe, effective manner and to identify any overlooked errors.
Test Apparatus	● 5 Xsens Dot IMUs ● A smartphone (Samsung Galaxy A71 5G) ● Xsens Straps ● Video camera ● Timer
Participants	The participants for this testing will be the members of the research team or/and physical therapists. This is a small scale summative usability test plan. The ideal testing should include at least 15 participants from each distinct user group or at least 25 participants from one user group.
Test Environment	● Ensure that the mobile application has been installed in the smartphone prior to the test. ● Brief the user on device overview. ● The participants will be provided with a list of tasks to complete throughout the test together with body straps, IMUs and the smartphone. ● Record participants' responses and task times throughout the test
Tasks	● Tasks for physical therapists or other groups as participants ○ Create a new user using the provided userID and log into the application ○ Choose exercise category, exercise and variation options ○ Play the exercise videos and read the instructions ○ Start your exercise and rate yourself. Repeat the trial thrice ○ View your daily, weekly and monthly exercise progress ○ View your performance data in the data tab ○ Fill out necessary forms and answer any questions after you complete all the tasks ● Additional tasks for researchers as participants ○ Connect and sync the IMUs with the smartphone. Make sure that all the IMUs and the smartphone are close to each other while synching ○ Once synching succeeds, put the IMUs into the straps in

	the correct position and put the straps onto the participant's body as labelled on the IMU.
Additional Attachment	● Small scale test ○ Device overview ○ Rating form ○ Risk identification ○ Interview questions ● Additional attachment for ideal test ○ Informed consent form ○ Background interview questions
Data collection	From the recording, the participant's subjective feedback, uncertainties and difficulties in carrying out the tasks will be collected. The time participants take to finish the tasks will also be recorded as well as the rating scale evaluation.
Data analysis	The data will then be analysed by: ● determining the root cause of any interaction problem or difficulty or use errors ● evaluating whether the time for each screen to interact with the user is less than 3 seconds. ● identifying delays while performing each task ● evaluating the rating given by the participants Any comments or post-test feedback will be reviewed to improve the usability of the whole system.

Firebase Usage Test

For this project, we are using the free version of Firebase, which is called Spark version. This version has daily usage quotas for write and read operations. For preliminary use of the solution, we conducted the Firebase usage test to evaluate the usage capacity that can be done for the cloud.The daily quota for write and read operations are 20K and 50K operations respectively.

Figure 31 below shows some of the parameters in Firebase quotas tracking. The parameter of interest here is memory usage, which sits at 0.0068 out of 1 GB with 100 exercises. This sets the memory capacity limit at around 14K exercises for 1 GB of free storage. Figure 32 shows the read and write requests sit well within the daily quota limit. The read request hits 4K as we conduct stress testing which runs a lot of load operations throughout the day. Figure 33 shows the historical peak read and write requests in the past week. We also acknowledge that Firebase might not be an ideal cloud choice for future implementations due to reasons explained in the discussion section. However, this cloud choice is verified to have sufficient capacity for preliminary application of the solution.

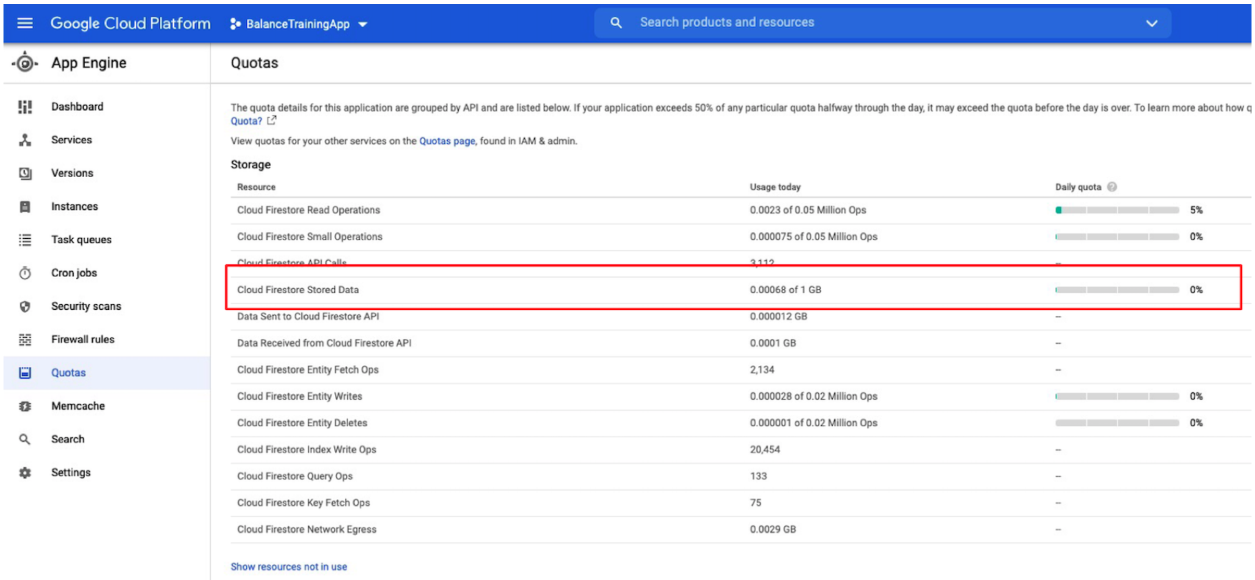


Figure 31: Screenshot of Firebase quotas. Highlighted is the storage usage for 100 exercises, which sits at 0.0068 out of 1 GB.

Figure 32: Screenshot of the daily read and write requests against the daily quota.

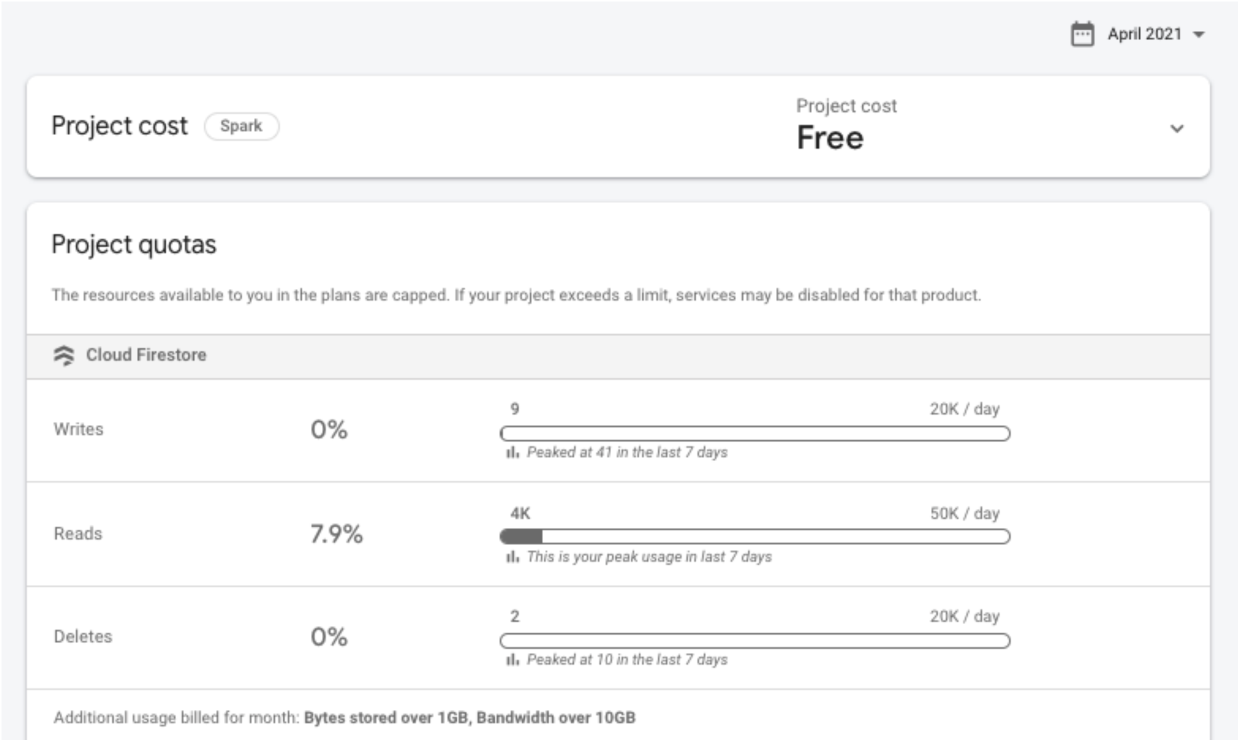


Figure 33: Screenshot of historical peak read and write requests in the past week.

Discussion and Recommendations

This section has a more in depth discussion on components of the final solution and detailed-level recommendations, together with identified strengths and weaknesses of the system. At the system-level, we encourage the individuals in charge of the future development of the system to look into the robustness of using multiple distinct programming languages in the system while familiarizing themselves with these languages prior to the implementation process.

Design Critique

We have identified several strengths and weaknesses of the final solution as shown in Table 22 below. These strengths and weaknesses are further explained with suggestions in the following subsections.

Table 22: Strengths and weakness of final solution

Strength	Weakness
Able to collect data from multiple body segments	Requires multiple attempts to sync IMU
Provides balance training instructions and progress with visual aid	Needs assistance to hold the phone while exercising
Is dynamic, customizable and extensible	Exercise instructions font is small
Can be used at home	Limited normative data implemented
Provides exercise selections in order of difficulty	

Software stack

As we developed the application, the integration between the different programming languages proved to be difficult. The IMU SDK is written in Java, so we had to use Java to communicate with the IMUs. However, for the front end, we chose to write it in React Native due to its faster pace of development, as mentioned in the detailed design solution. Because of this, we had to use Native Modules to use Java functions in React Native and pass data back and forth between the two programming languages. However, there are limitations with Native Modules, such as the added complexity to pass data, as well as the passing of data working one way. This is detailed further in the documentation in Appendix D. As such, there were some user interface elements that we had to omit due to the complexity, such as having a list of IMUs in the home screen. There's not really an easy way to pass the list of IMU from Java to React Native. If we had to start over, we would recommend exploring Kotlin for front end development. It is a programming language that is tightly coupled with Java (both developed by Google) and has a fast pace development model, similar to React Native.

User Interface

In this project, one of the specifications is to provide the user with exercise instructions in the form of video and text as a guide. We have successfully implemented this functionality in the final solution by using Youtube videos and displaying the pdf document of the exercise instructions. This causes the font of the pdf to be fixed and appear small. As of now, the pdf document can be zoomed in for a bigger view of the instructions. To accommodate the limitations of older adults, these changes can be made by extracting the instructions in text form and rendering them in the application.

User Interface Dynamicity

For user interface, the team is trying to make the software dynamic to address some of the concerns raised in previous stakeholder meetings. When a code file is written, what happens under the hood is that the code file is compiled and converted into machine language, which is then translated into the content on the phone screen. This establishes a one-to-one relationship between the code file and a particular page on the phone. The rules and the format that decide how a page should look are all specified in the code file. Since the application is built mainly for researchers, some of the contents are still subject to changes, especially the number of categories and exercises based on our discussion with the sponsor. This could be a potential problem. On the current model where everything is hardcoded, each code file will have to be modified correspondingly when a change is introduced. The solution we implemented is to introduce an additional layer, which consists of a template generator and list structure on top of the current model. When we run the template generator, it will fetch the required information from the list structure, and generate these code files based on the predefined template. Since this process is automated, new changes can be introduced directly from the list structure without having to modify the underlying code files. The team has performed test cases for both normal and corner cases scenarios to test the dynamic infrastructure for categories and exercise pages (Figure 34 and 35). More test cases and unit testing that covers basic functionalities will be conducted as more components are added, such as login authentication, video playback etc.

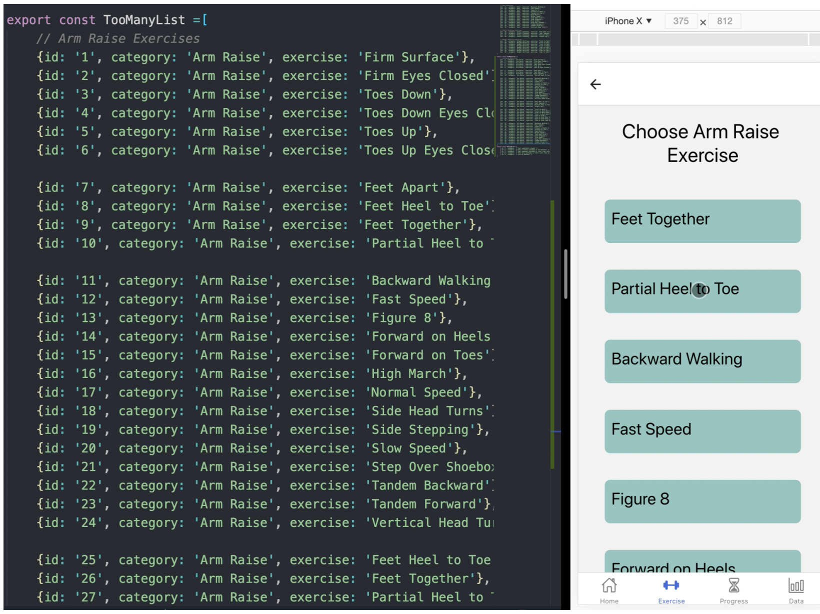


Figure 34: The application renders the long list with no errors

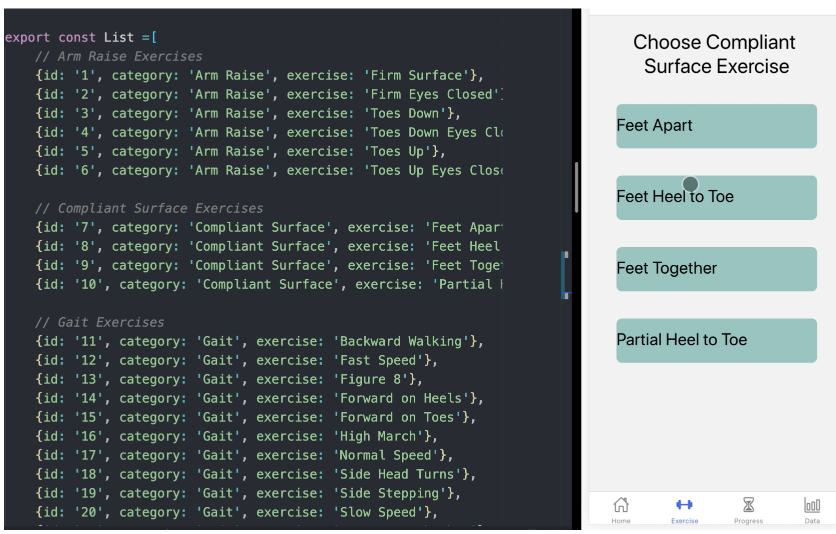


Figure 35: The application renders only four exercises which aligns with the list in code

Data Processing

There was a need to display metrics that are meaningful to the users in the app, hence raw data collected by the sensors must be analyzed and processed first behind the scenes before these metrics are displayed. Specifically, the metrics of interest to be displayed are the root-mean-square (RMS) values of the anterior/posterior (A/P) and medial/lateral (M/L) tilts; average values of A/P and M/L tilt velocities; path length; A/P and M/L ranges; “in-zone” percentage which is related to how well a user performs an exercise; and area of ellipse which is related to the spread of tilt data. These are all just numbers which make more sense to a user, compared to arrays with thousands of elements in each for the raw data. For now, we are setting A/P and M/L tilt values to be the Euler angles in x- and y-direction, respectively and A/P and M/L tilt velocities to be the angular velocities in x- and y-direction, respectively as we do not have enough information on how the A/P and M/L positions are measured for body parts other than the trunk and our sponsors have told us to simply represent them as any quantity output by the IMUs for now to make sure that the app would work.

We were provided with MATLAB scripts that our sponsors have been using to process data from IMU in their research work to be implemented in our app; they are used to convert raw IMU data into the aforementioned metrics. Translating the scripts written in MATLAB's proprietary ".m" file format to the Javascript ".js" file format was a simple enough task to be done manually as it was just a matter of looking at Javascript's documentation to determine what are the equivalent MATLAB commands in Javascript. The more challenging part, however, came in when we needed to find the equivalent basic 1-D array and statistical operations used in MATLAB in Javascript - there are parts of the MATLAB scripts where operations like finding the dot product of two arrays, finding the average value of elements in an array, and finding the covariance of two arrays are needed. While there is an existing third-party library for these operations and more called *math.js*, we ran into issues installing this library in our app development environment and we decided that it was easier to rewrite the functions that perform the operations we needed to do ourselves despite it being a tedious and repetitive task. Unlike their equivalent functions in MATLAB that are more flexible with the data structures allowed in their function inputs, these functions are limited in a way that they only allow specific data structure to be input to them - for example, 1-D arrays must be 1-by-n in size instead of n-by-1 and the covariance function can only take vectors or 1-D arrays of size 1-by-n.

While this approach may make our app run slightly more smoothly than having to install a dedicated math library that has unnecessary functions for our purposes, it may limit the work that may be done in the future if there is a need to display additional metrics that are calculated using operations beyond what we have implemented in this project. We would suggest future developers of this app to get the *math.js* library installed as part of the project files or move data processing and analysis operations to another platform that uses programming languages like Python which are more friendly with these tasks as there are math libraries that come with them.

Data Visualization

As mentioned in the design solution section, one of the visualizations is comparing the user's processed balance training data with normative data from literature. Most of the research papers found provide normative data for a certain variation combination of exercises in the Static Standing category. We decided to visualize user data by displaying the anterior/posterior RMS value (apRMS), medial/lateral RMS value (mlRMS), anterior/posterior velocity value (apVelocity) and medial/lateral velocity value (mlVelocity). This is because those metrics complements both the data processing variables described above and those found in research papers. After filtering the research papers that meet the processed data units, exercise variation combination and position of the sensors, we end up using two sources. The normative values for a certain variation combination of exercises in the Static Standing category are shown in Table 23 below. The value for the rest of the exercises and the physical therapists' rating are randomly generated.

Table 23: Normative data of certain Static Standing exercises

Exercise	Variation 1	Variation 2	Normative Data			
			apRMS (°)	mlRMS (°)	apVelocity (°/s)	mlVelocity (°/s)
Feet Apart [42]	Eyes Open	Head still	0.39	0.13	0.45	0.17
		Pitch	0.56	0.14	2.03	0.37
		Yaw	0.4	0.16	0.71	0.74
	Eyes Closed	Head still	0.41	0.13	0.52	0.19
		Pitch	0.66	0.16	2.5	0.45
		Yaw	0.44	0.21	0.9	1.05
Partial Heel to Toe [42]	Eyes Open	Head still	0.51	0.34	0.7	0.46
		Pitch	0.68	0.45	2.16	0.87
		Yaw	0.57	0.45	1.14	1.09
	Eyes Closed	Head still	0.57	0.39	0.86	0.58
		Pitch	0.79	0.53	2.61	1.09
		Yaw	0.72	0.56	1.47	1.41
Feet Together [43]	Eyes Open	Head still	2.02	2.02	-	-
	Eyes Closed		3.18	3.18	-	-

The first source of normative data provides the mean data from participants of all age ranges (young, middle-aged, old and very old) across four trials. One sensor is mounted on each subject's posterior lower back at the level of the iliac crest [42]. The second source of normative data provides the hip sway, which is the product of medial-lateral and anterior-posterior sway of the hip. The participants in this paper are of individuals 65 years or older recruited from primary, secondary and tertiary health care settings, community providers, assisted living facilities, retirement homes and aging service organizations [43]. In this source, five sensors are attached to the shins above ankles, thighs above knees and lower back close to the sacrum.

For uniformity, the user data displayed in the visualization screen will be the data from the sensor attached to the trunk. This can be revised to accommodate a more complex data processing that takes into account data from all five sensors. This implementation also can further be built on by expanding the normative data list to accommodate other exercises and integrating the machine learning algorithms to provide the physical therapists' rating into the platform.

The second data visualization is progress tracking, that is, to see how someone has been doing with their exercises with respect to time. As mentioned in the detailed design solution section, the user can view their daily, weekly and monthly progress. As the application develops, the yearly overview could be beneficial for the user to view their progress and improvement over the years. Hence, this can be implemented in future development using the current React Native library.

SDK Integration

The Java SDK functions can be generalized into two categories: synchronous functions whose executions are sequential and can be directly controlled by JavaScript functions, such as the "connectIMU()" function; asynchronous events that can occur at any moment and thus are less predictable, such as the "onXsensDotBatteryChanged()" event. Communication between the native Java functions and JavaScript UI is achieved through ReactMethod functions and event emitter, which are directly linked to the two types of SDK functions mentioned above. For example, when the "start exercise" button is clicked by a user, the JavaScript UI will invoke the native synchronous function "setMeasurementMode" to start IMU streaming in given mode, and the streamed data from IMU will arrive in form of asynchronous events that trigger the event emitter to update the data array.

Our implementation can be further improved by incorporating more asynchronous events that may contribute to better UI interaction. Right now, our SDK integration mainly utilizes the synchronous functions and performs a single operation to achieve intended functionality. For example, connection to IMUs is fulfilled by the "Scan and Connect" button, which performs a single scan and stores the connected IMU information in a HashMap; if one sensor dies during an exercise trial, because the HashMap is only updated by the button, the user won't be notified with this information and has to re-click the button on the home page to reinitialize the HashMap and sensor connection state. A solution to this could be periodically checking for sensor connections and providing a popup if disconnect events occur, at the cost of more complex logic design and battery life consumption.

Another improvement is to reduce the potential Bluetooth congestion and data stream load through serialization. During our stress test with all five IMUs connected and synchronized, we noticed that after a few trials, the local file and uploaded data from some of the sensors became empty. We conjectured that the most likely reason is that all 5 IMUs are recording at 60Hz at the same time and the program is overwhelmed by the data stream. Thus, instead of streaming the data directly, the recording can be done locally on each IMU, and after the trial completes, the data can be extracted from each IMU one by one to reduce load and congestion; the downsides of this alternative implementation are that the logic design will be more complicated, that more edge case bugs may be introduced and that the user needs to wait longer for the serialized data extraction.

Back End Infrastructure

As explained in the detailed software solution, there are four main points of cloud access in the application. The initial database structure is shown in Figure 36. It was built based on the IMU data shown in Figure 37. A zoomed in view of the IMU data field freeAccX in Firebase is shown in Figure 38. This is the system that we came up with at the beginning:

i. Authentication screen will check whether a user exists in the database by iterating through the collections and comparing values.
ii. Every save in rating screen will add one additional document into the corresponding collection, depending on which user is currently signed in.
iii. Progress screen will load all the required IMU data to perform calculations and render results.
iv. Data visualization screen will also do a load and calculate the parameters we need to plot the graph.

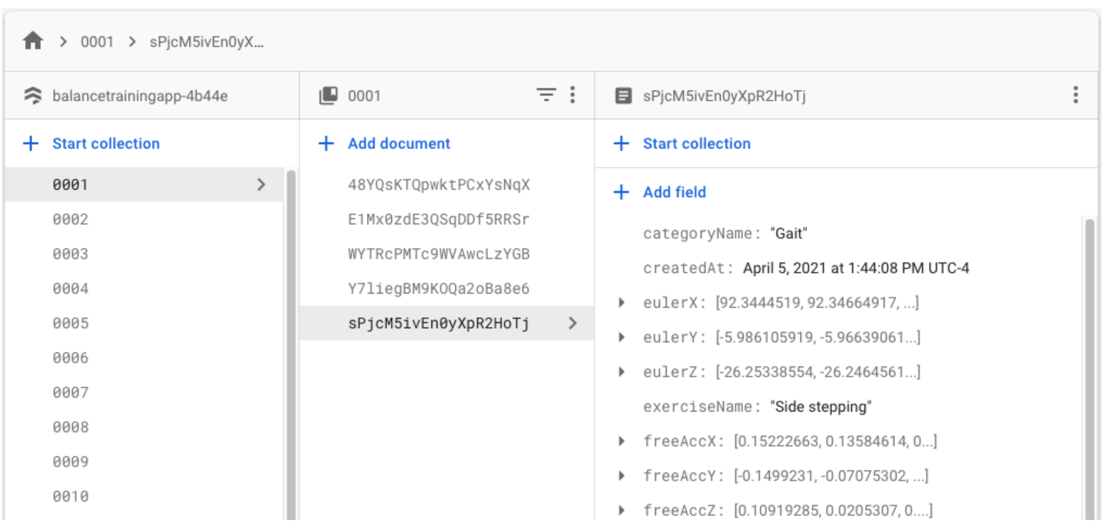


Figure 36: Screenshot of the initial database structure for the application.

D422CD0005FA_20210317_210630_405.csv   Open with

	A	B	C	D	E	F	G	H	I	J	K
1	sep=										
2	DeviceTag:	Xsens DOT									
3	FirmwareVersion:	1.6.0									
4	AppVersion:	2020.4									
5	SyncStatus:	Synced									
6	OutputRate:	60Hz									
7	FilterProfile:	General									
8	Measurement Mode:	Custom Modes - Custom Mode 1									
9	StartTime:	2021-03-17 21:06:30.405 -0400									
10	© Xsens Technologies B. V. 2005-2021										
11											
12	PacketCounter	SampleTimeFine	Euler_X	Euler_Y	Euler_Z	FreeAcc_X	FreeAcc_Y	FreeAcc_Z	Gyr_X	Gyr_Y	Gyr_Z
13	1	386318989	92.3444519	-5.986105919	-26.25338554	0.15222663	-0.1499231	0.10919285	-0.3890953958	-0.6418159008	-0.466650635
14	2	386335656	92.34664917	-5.96639061	-26.24645615	0.13584614	-0.07075302	0.0205307	0.6741326451	0.5664696693	-0.9389102459
15	3	386352323	92.34750366	-5.950919151	-26.23518944	0.055924505	7.42E-04	0.007399559	0.6256989837	0.8986021876	-0.6970111728
16	4	386368990	92.34526825	-5.944288254	-26.23130608	0.01636506	0.03456315	0.010164261	0.3941538334	0.4698521197	-0.1487201303
17	5	386385657	92.3405838	-5.944776058	-26.23031235	0.04622136	0.022090232	0.060756683	0.2222349793	0.2584129274	0.287099123
18	6	386402324	92.33516693	-5.950044632	-26.23083687	0.07679525	-0.03209065	0.102555275	0.1651058644	0.1356256306	0.5788149238
19	7	386418991	92.33525848	-5.954366684	-26.2290802	0.11333615	-0.07110215	0.11251354	0.5236452818	0.4087978899	0.5121451616
20	8	386435658	92.34037018	-5.950512886	-26.21525002	0.093945175	-0.08116873	0.06025982	0.8839583397	0.9540730715	-0.002164273057
21	9	386452325	92.34619904	-5.942861557	-26.19716454	0.08630584	-0.03869814	0.013155937	0.9551807046	1.216148615	-0.2410923392
22	10	386468992	92.34323883	-5.940423965	-26.18380737	0.08192465	0.012465318	-0.035082817	0.3948901892	0.9134193659	0.08413131535
23	11	386485659	92.33406067	-5.943852425	-26.17861748	0.11363846	0.01295794	-0.005378723	-0.01895339601	0.5314186811	0.4526468217
24	12	386502326	92.33235168	-5.946899891	-26.17712784	0.12151125	-0.029999461	0.08092499	0.4052505195	0.3003665209	0.4388543963
25	13	386518993	92.34635925	-5.945000648	-26.17090225	0.09078292	-0.032221626	0.13823509	1.380082607	0.5985879898	0.1302469671
26	14	386535660	92.37276459	-5.934217453	-26.15818405	0.0586346	0.01111217	0.091311455	2.167225122	0.9893482327	-0.4196385145
27	15	386552327	92.39815521	-5.921905041	-26.15037727	0.031349257	0.024286024	0.0060892105	2.073290348	0.6644894481	-0.4985639453
28	16	386568994	92.41136169	-5.91247654	-26.1566925	0.030246444	0.030642286	-0.0534544	1.253095031	-0.1755494475	-0.2901918888
29	17	386585661	92.41321564	-5.904263973	-26.17574692	0.069252245	-0.057016365	-0.07796097	0.4936079085	-0.9268595576	-0.1853463054
30	18	386602328	92.43727112	-5.889029026	-26.19916153	0.14788377	-0.066040926	-0.04311657	1.806636691	-1.136973381	-0.5973393917
31	19	386618995	92.47000885	-5.87157917	-26.21125603	0.07314473	0.01575364	-0.042675972	2.378997087	-0.6611662507	-0.7520273328
32	20	386635662	92.50144196	-5.853800774	-26.2256279	0.0035750896	0.09615974	-0.0028676987	2.296261549	-0.6898779869	-0.7686242461
33	21	386652329	92.52432251	-5.841487408	-26.25156021	-0.0576146	0.124137305	0.068323135	1.705293655	-1.441841483	-0.4079206586
34	22	386668996	92.53447723	-5.83283186	-26.30224991	-0.07138289	0.07413656	0.08464241	0.7975347042	-2.8441751	-0.1246441901
35	23	386685663	92.53533936	-5.828541279	-26.36735916	0.08176305	-0.032012865	0.12266827	0.1519025117	-3.700228214	0.1756070405

Figure 37: The screenshot shows the IMU data for 30s, with only data points 1-23 visible. The list is still running. A single column has roughly 1600-1700 numbers each. Complete data available here.

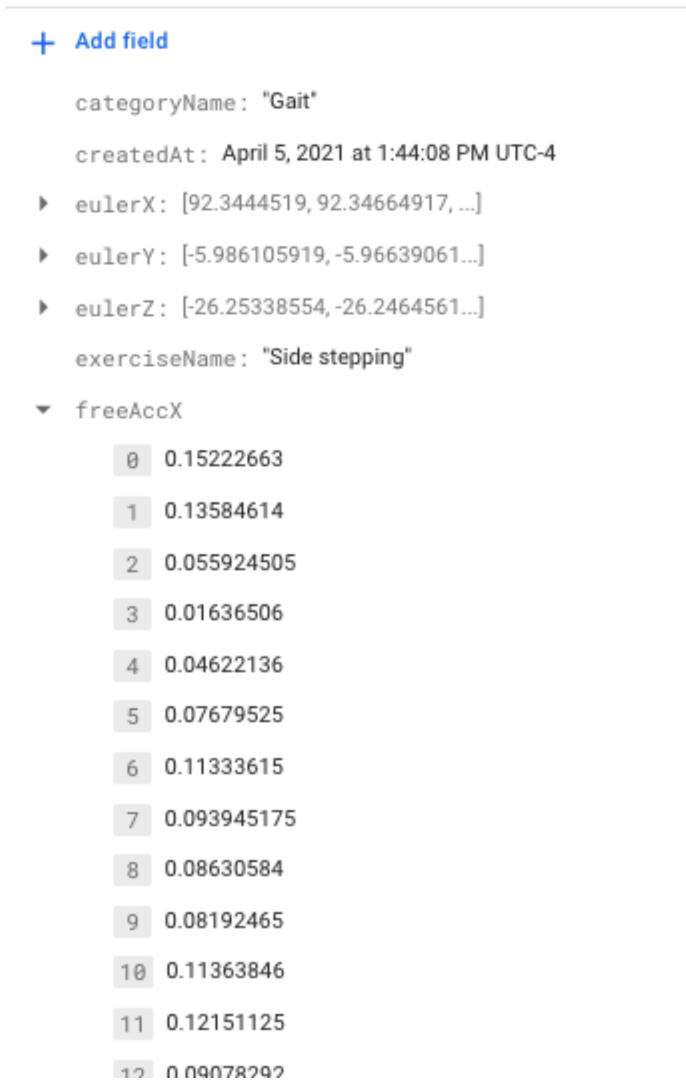


Figure 38: A zoomed in field showing freeAccX in Firebase. freeAccX contains 1600 values, similar to eulerX, Y, etc.

The system above works well and the application renders the expected output. However, as more and more exercise data get uploaded to the cloud, we started seeing a growing latency in response time. In an extreme case, for a user with 15 exercises, the progress page takes as long as 13s to (1) perform the load, (2) calculate the results from the loaded data and (3) render the results on screen. For the progress page specifically, the load operation stalls the entire process, delaying both the operations for calculations and rendering further down the pipeline. With this level of performance for 15 exercises, the current system will not survive the stress test. This is far from ideal.

A closer inspection reveals the reasons behind slow loads. When the front end pulls exercise data from the base, we get all the fields associated with the document (aka variables associated with a particular exercise). These include the categoryName, exerciseName, timestamp and most importantly the lists that store the entries for Euler_X, FreeAcc_X, etc.

Each of these lists have a minimum of 1600 entries for a 30s exercise and for one IMU, we'll have around six such lists (Euler_X,Y, Z and FreeAcc_X, Y, Z). Therefore, a load with these six lists alone will be a load that reads 9600 numbers from the database at the minimum. When we load 15 exercises, we're loading at least 144000 entries from the database.

The load seems inevitable, as we need the data from all the lists in order to perform the calculations to have something to display on the progress screen. However, we realized that when we load all the lists into the progress screen for our calculations, this is not the first time the app has "seen" these lists. The first time these lists showed up was in the rating screen, where users save their IMU data into the cloud, which is how we have these lists in Firestore in the first place. For each exercise, instead of uploading the whole raw data from the IMUs, we can just (1) perform all the calculations required for progress/data visualization screen at the rating screen using the IMU data, (2) pack the calculated result into a new mini object to upload to the cloud (3) and when we load from the progress/data visualization page, only load the mini object that contains all the variables needed to render the page properly, which is much smaller compared to 144000 entries. By shifting the math over and performing calculations to assemble the mini object, the amount of data transferred in read and write requests to Firestore is cut down significantly. The progress screen can now communicate with the back end on a mini package, distilled down to only the few variables needed to render the page properly and plot the chart.

To further optimize this, we can remove the need to load from Firestore in the progress screen and data visualization screen so that there's no delay in rendering content. When users are logging in at the authentication page, we can simultaneously load the mini package directly into a global variable (variable that can be passed between screens, small in size), which the progress page and data visualization page will then read from. The problem with this is that since loading from the cloud only happens upon logging in, the mini package (which is now the global variable), will become stale when the logged-in users perform a new exercise and navigate to the data visualization page right after. To solve this, we should do a reload on the global variable after each write as well.

Under this new structure, there are only two points where we need access to the data in cloud:

i. Authentication screen to check if the user ID exists and simultaneously load mini packages from the Firestore into a global variable.
ii. Rating screen to save IMU data and mini package to cloud and also do a reload on global variables. Notice that we now need to load after each write but we do not need to load every time the users navigate to the progress and data visualization page.

These are the two main optimizations implemented. With these changes, the extreme case in progress screen mentioned above loaded almost instantaneously. A stress test where we flood a particular user with 100 exercises shows that the load request comes back in <1s. Please refer to the documentation in Appendix D for more technical details.

Additionally, there were concerns about using Firebase as the database for the user collection and IMU data as the app collects medical data, which is sensitive information. There are rules regarding sensitive information that we have to oblige upheld by the Health Insurance Portability and Accountability Act (HIPAA) and the University of Michigan. However, because the only identifier for users in our application is an alphanumeric combination, our implementation is not personally identifiable. Additionally, we're only storing the processed IMU data, not the raw IMU data, in Firebase. The raw IMU data is stored locally in the smartphone, with a string label in the mini package to help researchers identify which raw data the calculated results are associated with. If there is a need to move this data to a private server in the future, an implementation worth exploring is creating an SSH connection to this server while using the app, so that the raw IMU data saves to the private server instead of to the smartphone's internal storage (assuming that the private server is a Linux server).

Machine Learning Integration

The exercise recommender system remains unfinished and a black box that needs to be implemented by the researchers in the future. After a user selects a self-rating from the rating screen, the next screen in the navigation stack is the category screen that includes every available exercise. The rating screen has all required input variables for the machine learning model: raw n*1 data array stored in "deviceArray" such as 3-dimensional free accelerations, processed data stored in "mock" such as apRMS, and the self-rating scaling from 1 to 5. Assuming the machine learning framework will be implemented on cloud, the input variables can be uploaded to the cloud server after the user selects a rating, and the server will process the trial data and return a list of recommended category/exercise to the phone as JS promise, which can be passed as props to determine what category/exercises to render in the next screen.

Conclusion

After having gone through the ME450 design process over the past 14 weeks, we were able to successfully build a smartphone-based personalized balance training platform that is able to collect kinematic data from multiple body parts, provide balance training for older adults, collect self-performance ratings, and support machine learning algorithms (in the future) for providing data-driven exercise recommendations and performance ratings tailored to an individual. This will help us deal with the high risk of falls experienced by older adults by providing them with a home-based balance training platform. This platform consists of five Xsens DOT sensors, an Android smartphone (Samsung Galaxy A71 5G), and an app that users can interact with to do balance training exercises and that can communicate with the sensors and the back end infrastructure (Firebase) to transfer data collected from the sensors to a secure cloud database.

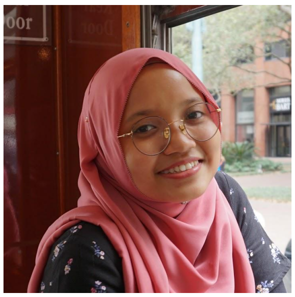
Our solution meets all of our requirements and specifications virtually; there are some that we were not able to verify due to time constraints for which we have developed a plan to do so (summative usability testing) and based on our stakeholders' feedback when we presented them our final solution, we are confident that it will pass this verification process in the future.

The Xsens DOT sensors that we have acquired for this project were working as expected in most cases. They can collect the different types of kinematic data needed for the balance training app and ultimately, balance training research that our sponsors are doing. Our tests have also proved that they have a very promising battery life - at least 1.5 times longer than what we have listed in our battery specification. The Bluetooth connection between the sensors and the smartphone we are working with is also reliable; there is no loss in data transmission between the two devices even though we reported that the signal strengths were lower than specified most of the time. However, there is currently an issue with the sensors where the synchronization process between IMUs requires multiple trials before it can be successfully performed due to hardware limitation with the IMUs. Doing the synchronization process with the official Xsens Dot application will also sometimes fail. We recommend that the IMUs and smartphone be placed together closely during the synchronization process. Additionally, using less IMUs can also result in a higher rate of success for synchronization.

The app we have developed for this project has the necessary features as requested by our sponsors which include a user identification system, exercise instructions, progress tracking, and data visualization of exercise performance. While there are still random values in normative data for certain exercises implemented in the data visualization page since we did not have enough information on that, we have already set up a framework for our sponsors to easily do so when they take over this project. Having run the app countless number of times, we have not had any issues with reading and writing data to Firestore where we store the user data. Despite Firestore working well for our project, there are concerns raised as it is not HIPAA-compliant for data privacy. However, for the foreseeable future, there are no issues when the product is used for internal research purposes by them as the data being stored in Firestore is encrypted and anonymized; the data for different users can only be identified by alphanumeric user IDs and only our sponsors know to whom they belong to.

Authors

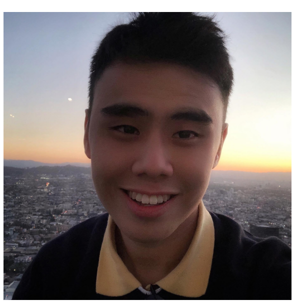
Haiqal Azham is from Selangor, Malaysia. His choice to major in mechanical engineering came about when he took part in robotics in elementary school. His interests include programming, photography and reading. Outside of school, he's the electrical subteam lead for Woven Wind, a project team that builds wind turbines and educates elementary school students on broad STEM topics that include the design process and clean energy.



Aliah Sallehuddin is from Johor, Malaysia. Within the field of mechanical engineering, she has interests in energy, climate, and controls. On-campus, she is currently a member of the M-HEAL: Project Alivio team. Her interests include theatre and learning foreign languages.

Xiyu Tian is from Zhanjiang, China. He was fond of physics and robotics in high school, and decided to pursue his interest in mechanical engineering. He was a member of Mars Rover's chassis team. He also loves electronics and excels at circuit layout design and computer architecture. He loves playing guitar and watching NBA during spare time.

Jia Chen 'JC' Wee is from Selangor, Malaysia. Growing up, he has always been interested in how things like toys and household appliances work and enjoys working on DIY projects as well as simple repairs, making him decide to pursue an education in mechanical engineering. His interests include traveling, consumer tech, and watching sitcoms. He is actively involved in Pi Tau Sigma (PTS) and the Michigan Malaysian Students' Association (MiMSA).



Jia Le 'Alex' Yu is from Johor, Malaysia. His interests include programming, playing board games & basketball. He has past internship experiences in web development, computer vision and natural language processing. He is also involved in Michigan Magic and Michigan Aviators in pre-COVID semesters.

Acknowledgements

First and foremost, we would like to express our deepest gratitude to our sponsors (Ms. Jamie Ferris, Ms. Safa Jabri, Dr. Christopher DiCesare & Prof. Xun Huan) for consistently giving us useful feedback, critiques, and suggestions for our work in the project. Subsequently, we would also like to thank our section instructor, Prof. Kathleen Sienko for her support, guidance, and trust in our team as well as for the resources she has provided our team with in order for us to successfully complete this project. We also appreciate the input that we have received during the formative usability testing from Dr. Wendy Carender, a physical therapist working in Michigan Medicine who is also working with our sponsors in their research on portable balance training platforms. The completion of this project was also not possible without the inspiration and constructive criticisms that we have received from our classmates in Section 003 (Teams 6-11) during our project presentations since the start of this course until the CoE Design Expo. Additionally, we would like to thank Team 7 (Stu Lee, Kai Myran, Kristina Nunez, Brady Oosse & David Wells) for the consultation session that we had primarily discussing the issues each of us were facing with our Xsens DOT sensors since both teams were using the same sensors for our projects. Finally, we are grateful to have talked to Prof. K. Alex Shorter about our project during the section swap session in early March as it was helpful to gather opinions about our work from someone who was not directly involved in the project and with whom we did not constantly communicate.

References and Information Sources

1. Tinetti, M. E., Speechley, M., & Ginter, S. F. (1988). Risk Factors for Falls among Elderly Persons Living in the Community. *New England Journal of Medicine*, *319*(26), 1701–1707. https://doi.org/10.1056/NEJM198812293192604
2. Vieira, E. R., Palmer, R. C., & Chaves, P. H. M. (2016). Prevention of falls in older people living in the community. *BMJ*, i1419. https://doi.org/10.1136/bmj.i1419
3. Shumway-Cook, A., Woollacott, M. H., Shumway-Cook, A., & Woollacott, M. H. (2001). *Motor control : theory and practical applications* (2nd ed.). Philadelphia: Lippincott Williams & Wilkins.
4. Lee, H., Park, B., & Yang, Y. (2016). Comparison of older adults' visual perceptual skills, cognitive function, and fall efficacy according to fall risk in the elderly. *Journal of Physical Therapy Science*, *28*(11), 3153–3157. https://doi.org/10.1589/jpts.28.3153
5. Bao, T., Carender, W. J., Kinnaird, C., Barone, V. J., Peethambaran, G., Whitney, S. L., … Sienko, K. H. (2018). Effects of long-term balance training with vibrotactile sensory augmentation among community-dwelling healthy older adults: a randomized preliminary study. *Journal of NeuroEngineering and Rehabilitation*, *15*(1). https://doi.org/10.1186/s12984-017-0339-6
6. Winter, D. A. (1995). *A.B.C. (Anatomy, Biomechanics and Control) of Balance During Standing and Walking*. Ontario, Canada: Waterloo Biomechanics.
7. Lacroix, A., Hortobágyi, T., Beurskens, R., & Granacher, U. (2017). Effects of Supervised vs. Unsupervised Training Programs on Balance and Muscle Strength in Older Adults: A Systematic Review and Meta-Analysis. *Sports Medicine (Auckland, N.Z.)*, *47*(11), 2341–2361. https://doi.org/10.1007/s40279-017-0747-6

8. Klatt, B., Carender, W., Lin, C., Alsubaie, S., Kinnaird, C., Sienko, K., & Whitney, S. (2015). A Conceptual Framework for the Progression of Balance Exercises in Persons with Balance and Vestibular Disorders. *Physical medicine and rehabilitation international*, *2*(4). Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4968039/

9. Ma, C. Z.-H., Wong, D. W.-C., Lam, W. K., Wan, A. H.-P., & Lee, W. C.-C. (2016). Balance Improvement Effects of Biofeedback Systems with State-of-the-Art Wearable Sensors: A Systematic Review. *Sensors*, *16*(4), 434. http://dx.doi.org.proxy.lib.umich.edu/10.3390/s16040434

10. Gordt, K., Gerhardy, T., Najafi, B., & Schwenk, M. (2018). Effects of Wearable Sensor-Based Balance and Gait Training on Balance, Gait, and Functional Performance in Healthy and Patient Populations: A Systematic Review and Meta-Analysis of Randomized Controlled Trials. *Gerontology*, *64*(1), 74–89. https://doi.org/10.1159/000481454

11. Woollacott, M. H. (2000). Editorial: Systems Contributing to Balance Disorders in Older Adults. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, *55*(8), M424–M428. https://doi.org/10.1093/gerona/55.8.M424

12. Fung, A., Lai, E. C., & Lee, B.-C. (2018). Usability and Validation of the Smarter Balance System: An Unsupervised Dynamic Balance Exercises System for Individuals With Parkinson's Disease. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *26*(4), 798–806. https://doi.org/10.1109/TNSRE.2018.2808139

13. Lee, B.-C., Kim, J., Chen, S., & Sienko, K. H. (2012). Cell phone based balance trainer. *Journal of NeuroEngineering and Rehabilitation*, *9*(1), 10. https://doi.org/10.1186/1743-0003-9-10

14. McGarrigle, L., Boulton, E., & Todd, C. (2020). Map the apps: a rapid review of digital approaches to support the engagement of older adults in strength and balance exercises.

BMC Geriatrics, *20*(1), 483. https://doi.org/10.1186/s12877-020-01880-6

15. Kao, C.-L., Chen, L.-K., Chern, C.-M., Hsu, L.-C., Chen, C.-C., & Hwang, S.-J. (2010). Rehabilitation outcome in home-based versus supervised exercise programs for chronically dizzy patients. *Archives of Gerontology and Geriatrics*, *51*(3), 264–267. https://doi.org/10.1016/j.archger.2009.11.014
16. Xue, J., Wu, Q., & Mu, W. (2019). Real-time Training Model for Elderly People Balance Ability. *Journal of Physics: Conference Series*, *1325*, 012025. https://doi.org/10.1088/1742-6596/1325/1/012025
17. Alhasan, H., Hood, V., & Mainwaring, F. (2017). The effect of visual biofeedback on balance in elderly population: A systematic review. *Clinical Interventions in Aging*, *12*, 487–497. https://doi.org/10.2147/CIA.S127023
18. Schwenk, M., Grewal, G. S., Holloway, D., Muchna, A., Garland, L., & Najafi, B. (2016). Interactive Sensor-Based Balance Training in Older Cancer Patients with Chemotherapy-Induced Peripheral Neuropathy: A Randomized Controlled Trial. *Gerontology*, *62*(5), 553–563. https://doi.org/10.1159/000442253
19. Sungkarat, S., Fisher, B. E., & Kovindha, A. (2011). Efficacy of an insole shoe wedge and augmented pressure sensor for gait training in individuals with stroke: a randomized controlled trial. *Clinical Rehabilitation*, *25*(4), 360–369. https://doi.org/10.1177/0269215510386125
20. Byl, N., Zhang, W., Coo, S., & Tomizuka, M. (2015). Clinical impact of gait training enhanced with visual kinematic biofeedback: Patients with Parkinson's disease and patients stable post stroke. *Neuropsychologia*, *79*, 332–343. https://doi.org/10.1016/j.neuropsychologia.2015.04.020
21. Octavia, J. R., & Coninx, K. (2014, May 28). Adaptive Personalized Training Games for

Individual and Collaborative Rehabilitation of People with Multiple Sclerosis. *BioMed Research International*. Clinical Study, Hindawi. https://doi.org/10.1155/2014/345728

22. Lafond, D., Duarte, M., & Prince, F. (2004). Comparison of three methods to estimate the center of mass during balance assessment. *Journal of Biomechanics*, *37*(9), 1421–1426. https://doi.org/10.1016/S0021-9290(03)00251-3
23. Rollins, J. D., Collins, J. S., & Holden, K. R. (2010). United States Head Circumference Growth Reference Charts: Birth to 21 Years. *The Journal of Pediatrics*, *156*(6), 907-913.e2. https://doi.org/10.1016/j.jpeds.2010.01.009
24. Comparative Anthropometry of the Foot. (n.d.). United States Army Natick Research & Development Laboratories. Retrieved from https://apps.dtic.mil/dtic/tr/fulltext/u2/a126189.pdf
25. Lesinski, M., Hortobágyi, T., Muehlbauer, T., Gollhofer, A., & Granacher, U. (2015). Effects of Balance Training on Balance Performance in Healthy Older Adults: A Systematic Review and Meta-analysis. *Sports Medicine (Auckland, N.Z.)*, *45*(12), 1721–1738. https://doi.org/10.1007/s40279-015-0375-y
26. Abass, S., Ghazi, M., & Alalsaidissa, J. (2017). The Effects of Body Mass Index BMI on Human Gait Analysis. https://doi.org/10.9790/1813-0610034654
27. Service Status | CloudHarmony. (n.d.). Retrieved February 8, 2021, from https://cloudharmony.com/status-in-america_north-group-by-regions
28. Tsiachri Renta, P., Sotiriadis, S., & Petrakis, E. G. M. (2017). Healthcare Sensor Data Management on the Cloud. In *Proceedings of the 2017 Workshop on Adaptive Resource Management and Scheduling for Cloud Computing* (pp. 25–30). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3110355.3110359
29. Zhou, L., Fischer, E., Tunca, C., Brahms, C. M., Ersoy, C., Granacher, U., & Arnrich, B.

(2020). How We Found Our IMU: Guidelines to IMU Selection and a Comparison of Seven IMUs for Pervasive Healthcare Applications. *Sensors (Basel, Switzerland)*, *20*(15). https://doi.org/10.3390/s20154090

30. Experience, W. L. in R.-B. U. (n.d.). 10 Usability Heuristics for User Interface Design. *Nielsen Norman Group*. Retrieved February 28, 2021, from https://www.nngroup.com/articles/ten-usability-heuristics/

31. Abdullah, N., & Hamid, N. F. binti A. (2019). Interface design features of mobile application for senior citizens. *Indonesian Journal of Electrical Engineering and Computer Science*, *14*(1), 436–442. https://doi.org/10.11591/ijeecs.v14.i1.pp436-442

32. Kascak, L. R., Rébola, C. B., & Sanford, J. A. (2014). Integrating Universal Design (UD) Principles and Mobile Design Guidelines to Improve Design of Mobile Health Applications for Older Adults. In *2014 IEEE International Conference on Healthcare Informatics* (pp. 343–348). Presented at the 2014 IEEE International Conference on Healthcare Informatics. https://doi.org/10.1109/ICHI.2014.54

33. dos Santos, M. M. T., Antonelli, H. L., Rodrigues, S. S., de O. Silva, C. L., Fortes, R. P. M., & Castro, P. C. (2016). Personalizing health-related ICT interface and application: older adults and elderly caregivers preferences. In *Proceedings of the 7th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion* (pp. 331–338). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3019943.3019991

34. Correia, C., Oliveira, E., & Nunes, F. (2020). Using illustration to create more inclusive user interfaces for older adults. *Interactions*, *27*(2), 79–81. https://doi.org/10.1145/3378556

35. Lin, C. J., & Ho, S.-H. (2020). The development of a mobile user interface ability evaluation system for the elderly. *Applied Ergonomics*, *89*, 103215.

https://doi.org/10.1016/j.apergo.2020.103215

36. Mukherjee, P., & Mukherjee, A. (2019). Chapter 1 - Advanced processing techniques and secure architecture for sensor networks in ubiquitous healthcare systems. In N. Dey, J. Chaki, & R. Kumar (Eds.), *Sensors for Health Monitoring* (Vol. 5, pp. 3–29). Academic Press. https://doi.org/10.1016/B978-0-12-819361-7.00001-4

37. Aileni, R. M., Suciu, G., Valderrama Sukuyama, C. A., Pasca, S., & Maheswar, R. (2020). 14 - Internet of wearable low-power wide-area network devices for health self-monitoring. In B. S. Chaudhari & M. Zennaro (Eds.), *LPWAN Technologies for IoT and M2M Applications* (pp. 307–325). Academic Press. https://doi.org/10.1016/B978-0-12-818880-4.00015-6

38. Average number of connected devices in U.S. 2020. (n.d.). *Statista*. Retrieved March 22, 2021, from https://www.statista.com/statistics/1107206/average-number-of-connected-devices-us-house/

39. Lueken, M., Mueller, L., Decker, M. G., Bollheimer, C., Leonhardt, S., & Ngo, C. (2020). Evaluation and Application of a Customizable Wireless Platform: A Body Sensor Network for Unobtrusive Gait Analysis in Everyday Life. *Sensors (Basel, Switzerland)*, *20*(24). https://doi.org/10.3390/s20247325

40. Xsens DOT User Manual. (2020, December 15).

41. Sherman, E. (n.d.). Top 6 reasons mobile apps crash. *TechBeacon*. Retrieved March 15, 2021, from https://techbeacon.com/app-dev-testing/top-6-reasons-mobile-apps-crash-how-best-avoid-murphy

42. Alsubaie, S. F., Whitney, S. L., Furman, J. M., Marchetti, G. F., Sienko, K. H., & Sparto, P.

J. (n.d.). Reliability of Postural Sway Measures of Standing Balance Tasks. *Journal of Applied Biomechanics*, *35*(1), 11–18. https://doi.org/10.1123/jab.2017-0322

43. Mohler, M. J., Wendel, C. S., Taylor-Piliae, R. E., Toosizadeh, N., & Najafi, B. (2016). Motor Performance and Physical Activity as Predictors of Prospective Falls in Community-Dwelling Older Adults by Frailty Level: Application of Wearable Technology. *Gerontology*, *62*(6), 654–664. https://doi.org/10.1159/000445889
44. FIPS 197, Advanced Encryption Standard (AES). (2001). *Federal Information Processing Standards*, 51.
45. Server-side encryption | Firestore. (n.d.). *Google Cloud*. Retrieved April 26, 2021, from https://cloud.google.com/firestore/docs/server-side-encryption
46. ASME Code of Ethics of Engineers. (2006). American Society of Mechanical Engineers. Retrieved from https://www.asme.org/wwwasmeorg/media/resourcefiles/aboutasme/get%20involved/advocacy/policy-publications/p-15-7-ethics.pdf

Appendix A - Required Supplemental Appendices

(A.1) Engineering Standards

The product that we are developing in this project is a medical device. As important as it is for our product to be able to satisfy our potential users' healthcare needs - in this case, to perform balance training exercises at home in a way that resembles a clinic-based setting -, it is equally important that it can be safely used by the users. To ensure that this product can be safely used, we have incorporated the International Electrotechnical Commission (IEC) 60601-1 standard which dictates the allowed maximum temperature of medical electrical equipment since the design solution contains the Xsens DOT IMUs which are electronic devices. Specifically, we stated that the maximum operating temperature of our design solution must be less than 43 °C in our requirements and specifications list, which is also the temperature listed in the standard for an applied part having contact with a patient for a duration of over 10 minutes. This timeframe is chosen since users are very likely to perform balance training exercises for at least 10 minutes per session and we assume that they do not take the sensors off their bodies in the middle of a session. Even though the product will not be made commercially available at this stage of the project, our sponsors will conduct human trials when the project is taken over by them and it is crucial that they do not cause any harm to the trial subjects during the process. Hence, adopting this standard throughout our design process was our top priority.

On top of the safety considerations of our product, we must also prioritize the privacy of our users given that the product collects users' health data which are considered sensitive information. To protect the privacy of these data over the Internet, they must be encrypted and the standard that we are incorporating in our design solution for data encryption purposes is the Advanced Encryption Standard (AES). This standard, established by the United States National Institute of Standards and Technology (NIST) is also the one adopted by the United States government as their data encryption standard since 2002 [44]. Seeing how this product will be used primarily in the United States when it is made commercially available, we believe that adopting the AES in our design solution for data privacy is the right choice. Currently, we are using Google's Firebase platform to host the data collected by the IMUs as well as other relevant exercise data; the data and metadata stored on this platform are encrypted with the 256-bit AES according to Google [45].

(A.2) Engineering Inclusivity

At the current stage of this project, we identified our only immediate stakeholders as our sponsors who are doing research on the topic of smartphone balance training platforms and will continue developing our design solution after the end of this senior design course, and Prof. Sienko who are directly supervising our sponsors in their research. That being said, we must also keep our future potential stakeholders in mind when defining our design problem since this proof of concept smartphone balance trainer product may become commercially available in a few years when further research and development have been done to the product. These potential stakeholders that we have identified so far are older adults and their caregivers as well as physical therapists who supervise their balance training routines.

At the beginning of this project, we recognized that there are big differences between our team's and our current and future stakeholders' identities: (1) we are students majoring in mechanical engineering and had no prior experience and knowledge of balance training routines and designing a medical device like the one we are developing; (2) our sponsors and Prof. Sienko are more experienced in biomechanical engineering than we are, and they have worked with actual balance training patients in trials for their research projects in the past; and (3) our future stakeholders are the ones directly involved in balance training routines in their daily lives. It was important for us to communicate effectively with our sponsors from the very start in order to verify and correct any prior assumptions that we have had about what balance training is about so that we could understand better what was the problem that we had to solve, and subsequently we would be able to fully define the problem as soon as possible before moving on to generating concepts for our solution and developing the solution itself. Some of the requirements for our project (e.g.: adjustable for different body shapes, user friendly design) were listed based on the consideration that the product will not just be used by our sponsors, but also the PTs and older adults. During our formative usability testing for the smartphone application, we have invited Wendy Carender, a PT working with Michigan Medicine who also works with our sponsors in their research, to evaluate our app's user interface design and to offer her thoughts about the design. Receiving an input from her helped us to keep ourselves on the right track when developing our design solution to accommodate as many stakeholders as possible.

Additionally, because a big part of our project involves writing code for the software component and the hardware-software integration for our app and our sponsors may not be as experienced as some of our team members are in mobile app development, we found that it was necessary for us to communicate with them constantly about what we were doing so that they would always be informed of the design decisions we made and be able to share their thoughts and/or concerns with us about our decisions. We would do this to ensure that any important changes to our decisions could be made as early as possible.

We think that we could have made our design process more inclusive by having interactions with these older adults who are potential users of our product. However, we understood that due to the current COVID-19 pandemic, interacting directly with older adults who undergo balance training to gain insights from them about our design process was not a feasible thing to do. We hope that when this project is further developed in the future, there will be an emphasis on including the potential users in any design decisions made.

(A.3) Environmental Context Assessment

The main components of our design solution are the Xsens DOT IMUs and the straps that they come with, a smartphone, and a mobile application. When this solution is widely adopted in the future, we believe that there will be negligible environmental impacts associated with the development of the mobile application mainly because the only impact that it has is due to the use of electricity by the developers to write code for the application. For the use of smartphones as part of the balance training platform system, there certainly are environmental impacts associated with depending on how the materials used to make the smartphones are sourced as well as the facilities in which they are manufactured - this may vary between smartphone manufacturers. While these environmental impacts may not be negligible when they are looked at from a general point of view, they are not something inherent to just our design solution because almost everyone owns at least one smartphone nowadays. The IMUs we are using and the smartphone application we developed in our design solution are intended to work with smartphones with at least Android 10 - an operating system that was released in 2019 - installed; they can also work with iOS smartphones/iPhones should our sponsors choose to make the balance training platform available in both major smartphone operating systems.

There are, however, larger concerns about the environmental impact associated with the XSens DOT IMUs and the straps for the sensors which are also made by the same manufacturer. The sensors, although equipped with a common battery type (Li-ion rechargeable 2032 button cells) that can be simply bought by a consumer, are not designed to be taken apart by users to make any repairs such as installing a new battery (in fact, they warned users against opening the sensor "for safety reasons"). While the rechargeable battery inside the sensors should last at least a few years with normal use before it is considered "consumed" - which is when a battery cannot hold charge long enough for normal operations - users will not be able to continue using these sensors at the end of the battery's life. Xsens provides two-year warranties for these IMUs from the date of delivery and it is possible for users to get their sensors repaired by Xsens during this timeframe. Once the two-year mark has passed, it does not appear that Xsens offers any sort of out-of-warranty repair services - at least they are not promoted on their official website. Additionally, for in-warranty repairs, the sensors must be shipped to the Netherlands where their headquarters are located and the shipping costs are to be borne by users which may discourage them from getting the sensors repaired. The fact that the IMUs are designed to be non-user repairable and the repair service, or the lack thereof, offered by Xsens can potentially contribute to a significant amount of e-waste when their batteries cannot support normal operations anymore since users will be likely to discard them in landfills.

We are not quite certain about whether the straps can significantly impact the environment as we do not know what materials exactly they are made of (Xsens calls the material used “X-treme” on their website) and how they are manufactured. The environmental impact of the straps may need to be analyzed further if better information about them can be found, but our assumption is that they do not have as big of an impact as the sensors themselves do.

In general, we think that our design solution, if adopted commercially, can reduce the carbon footprint associated with older adults having to frequently travel to their PTs in a clinic or a hospital to perform balance training exercises since the balance training platform replicates the experience that the older adults would have when they are doing these exercises with their PTs at the comfort of their own homes. We foresee that this environmental benefit outweighs the environmental costs associated with the use of the IMUs. Based on the discussion of the environmental context assessment in this section, our design solution will be able to make significant progress towards an unmet and important social challenge, namely the lack of a more accessible way to perform balance training exercises for older adults. Our design solution will also most likely not have potential to lead to undesirable consequences in its lifecycle that overshadow the social benefits given how reducing carbon footprint from traveling to a PT can outweigh the possible environmental impacts from the Xsens DOT sensors.

It is important to note that this assessment will only apply if our design solution is made available commercially once our sponsors have made further developments and conducted trials and tests with it. As of this stage of the project, since we are building a proof of concept smartphone balance trainer platform for our sponsors, we do not believe that there are significant environmental impacts associated with our project.

(A.4) Social Context Assessment

Based on the limitations of the conventional way of performing balance training exercises with a PT in a clinic-based setting that we have discussed in our Problem Definition section, we can see that our product can address these limitations - especially those related to cost and the need to travel -, making it likely to be adopted and self-sustaining in the market if it is made commercially available in the future. While we do not foresee that the COVID-19 pandemic will still be happening when the product is available to the public, having a medical device simulating as much as possible an experience visiting a PT that can be used at the comfort of the older adults’ homes for their balance training routines will probably be very useful for them as they can still receive the medical care that they need while not compromising their own safety - this also then makes our product resilient to disruptions in business as usual as the product is relevant for use by older adults at any time or in any situation as it is a portable device that can be used at home.

As discussed in the Environmental Context Assessment section above, we are concerned about the e-waste potential of the Xsens DOT IMUs and whether the straps are manufactured without any significant environmental impacts, although we did highlight the possible benefit of reducing carbon emissions from the reduced travel due to the use of the product. We are not very certain at this stage if planetary or social systems will be worse off if this product is economically successful in the future and is widely adopted as we do not have enough details to inform ourselves of the potential impacts of this product to the environment and society.

Our immediate goal throughout the duration of this course was to develop a framework for a smartphone-based balance training platform that our sponsors can use for their research in this topic and we did not discuss with our sponsors thoroughly about the long-term goals of the project. We hope that this project will gain more traction in the physical rehabilitation and biomedical engineering communities, for example, so that there will be a better understanding on how much of a social impact it will have if the product is made commercial.

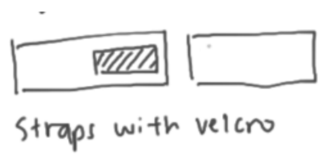
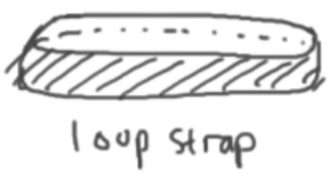
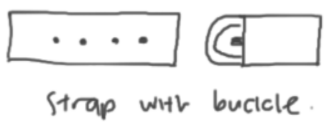
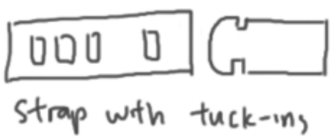
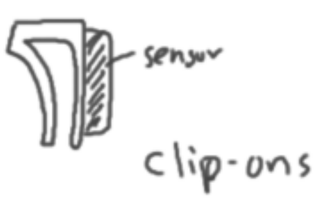
(A.5) Ethical Decision Making

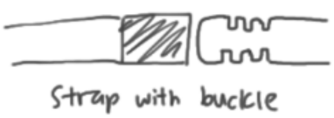
Throughout this course, we have never been in any situations where ethical dilemmas were present as we did not have a lot of standards to comply with in our design solution and that we were mainly developing the solution according to our sponsors' needs since they will be using the product after the end of ME450. We made sure to always adhere to the following fundamental principles in ASME's Code of Ethics for Engineers to prepare ourselves for any ethical dilemmas we would be faced with [46]:

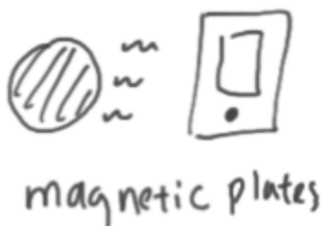
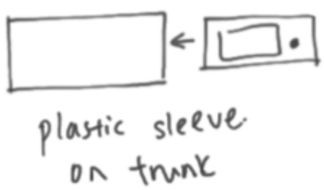
> *"Engineers uphold and advance the integrity, honor and dignity of the engineering profession by:*
>
> > *I. using their knowledge and skill for the enhancement of human welfare;*
> >
> > *II. being honest and impartial, and serving with fidelity their clients (including their*
> >
> > *employers) and the public; and*
> >
> > *III. striving to increase the competence and prestige of the engineering profession."*

The first principle is related to the application of the skills that we have learned throughout our mechanical engineering undergraduate career at the University of Michigan in our design process as well as our skills in programming, specifically, in developing a smartphone application for the balance training platform. Next, the second one is related to how we constantly communicated with our sponsors and Prof. Sienko throughout the project duration to inform them of our progress and any problems that we were facing - regardless of whether a problem was big or small, we always remained honest with them and did not hold back any important information from them. Finally, the last principle has to do with our learning process throughout the course. In the beginning of ME450, we had almost zero knowledge on how balance training works and the development of medical devices. Some of us also did not have the programming background needed for mobile app development. With the constant desire to learn and to improve ourselves as engineers in the making, we brought ourselves up to speed with the skill sets and knowledge needed for this project guided by the design process that we learned for this course. While none of us are likely to work in the field of biomedical engineering in the future, the experience we had from working on our project will certainly inspire us to be more adaptable to new environments and to be open to learning new topics in the process to continue our professional development, as stipulated in one of the fundamental canons in ASME's Code of Ethics for Engineers.

Appendix B - Concepts Generated for Hardware

Concept	Explanation/Description
	A strap made out of either nylon or a stretchable material that has velcro throughout the strap for infinite adjustability.
	A strap without any buckles or clasps that is made out of silicone-based material that can be stretched a lot of times while maintaining its elasticity like the Apple Watch Solo Loop band.
	Something that looks like a belt (or a watch if the strap is used for limbs) and can be adjustable for a finite number of lengths.
	A strap that has multiple slits throughout its length to secure the “head” of the strap in. It can support adjustability for a finite number of lengths.
	Non-toxic adhesive that sticks to an IMU on one side and a body part on the other. There is a lot of flexibility in terms of where it can be placed on the body.
	Sensors can be put into a pouch or a holder that is mounted to a clip that can be used with clothing.

	Sensor is placed on the part of the suspender that goes around the trunk.
	A pocket can be made on the open end of the beanie to put a sensor inside.
	Inspired by head strap mounts for action cameras like GoPro. A sensor is mounted at the location where cameras would be mounted.
	A release buckle is used at the ends of the strap and there is excess length on the strap which can be adjusted to suit different body part sizes.
	A cylindrical-shaped fabric with a pouch for sensors.

	Like the adhesive patches, but using magnets instead of adhesive. One plate is mounted to a strap and the other is mounted to a smartphone. Example: https://www.scosche.com/magicmount-replace-kit.
	Phone case is mounted to a strap.
	Like the phone case idea except a plastic pouch is used to secure the smartphone on the user's body parts.
	A lanyard with a phone pouch to be hung around the neck.
	Casting smartphone screen to a TV using Apple TV, Google Chromecast, etc. to view content while performing exercises.

	Similar to the one used in action cameras like GoPro.
	Smartphone will be put away during exercise.
	This can be done in settings where an assistant to the user is present like in a research study setting.
	The user would hold onto their phone during exercise.

Appendix C - Solution Development Supplemental Details

(C.1) IMU Benchmark

Table C.1.1: Benchmark of different sensor systems for our project. The Xsens DOT sensor would be chosen to be part of our final design solution.

	Xsens DOT	**Mbientlab MMR**	**LPMS-B2**
Supported platform (SDK support)	Android/iOS/Windows/macOS	Android/iOS/Windows/Linux	Android/Windows
Connects directly to smartphone	Yes	Yes	Yes
Simultaneous connections (Bluetooth)	Up to 7	Up to 3	Up to 7
Battery life	Up to 6 hours	Up to 8 hours	Up to 6 hours
Cost (single unit)	$105	$87.99	$249
Storage capacity (MB)	16	8	32
Weight (g)	10.8	5.7	12
Dimension (mm)	36.3*30.35*10.8	26*17*2.5	39*39*8
Sampling rate	Up to 120Hz for recording, Up to 60Hz for streaming 800Hz raw sampling rate	Up to 800Hz for recording, Up to 100Hz for streaming	Up to 400Hz
Accuracy			
Static	0.5º RMS	1º RMS	0.5º RMS
Dynamic	1.0º RMS		2º RMS
Accelerometer	Yes	Yes	Yes
Range (g)	± 16	± 16	± 16
Resolution (bit)	16	16	16
Output noise density (µg/√Hz)	120	-	90
Gyroscope	Yes	Yes	Yes
Range (º/s)	± 2000	± 2000	± 2000
Resolution (bit)	16	16	16
Output noise density (º/s/√Hz)	0.007	-	0.007
Magnetometer	Yes	Yes	Yes
Range (G)	± 8	-	± 16

Resolution (bit)	16	-	16
Output	Quaternion/Acceleration/Angular Velocity/Magnetic Field/Euler angle/Earth acceleration	Quaternion, Rotation Matrix, Euler Angles, Linear Acceleration, Earth Acceleration	Raw data / Euler angle / Quaternion
Documentation	Xsens DOT landing Page - Offers dedicated customer support - Also has a community forum - Includes app creation documentation	MetaMotionR – MbientLab - Often referred to community forums instead of having dedicated customer support - Includes API usage documentation	LPMS Documentation - LP-RESEARCH Knowledge Base - Confluence (atlassian.net) - Not much documentation on mobile app specifically
Ordering	Ships within 10 days	Ships within 5 days	Available on March 5th
Available Straps	- 1 of 128 cm x 10 cm (50.4” x 4 “) a pocket 6 cm from the end (centered) - 2 of 55 cm x 5 cm (21.65” x 2”) a pocket 4 cm from the end - 2 of 29 cm x 5 cm (11.4” x 2”) a pocket 3 cm from the end - Adhesive patches	- Wristband - Adhesive pads - Velcro (multiple lengths between 10 and 48 inches)	- Holder + strap

Table C.1.2: Comparing the different types of sensor systems currently used by our sensor in the Sienko Research Group for their research. This table was used in conjunction with Table C.1.1 above to compare them to the potential sensor systems that we would use for our project.

	APDM Opal	Xsens MTw
Supported platform (SDK support)		
Connects directly to smartphone	No	No
Simultaneous connections (Bluetooth)	Up to 24	Up to 20
Battery life	Up to 16 hours	Up to 6 hours
Cost (single unit)		
Storage capacity (MB)	1000	

Weight (g)	25		16
Dimension (mm)	43.7*39.7*13.7		47*30*13
Sampling rate	Up to 128 Hz		120 Hz, 1000Hz raw sampling rate
Accuracy			
Static	1.15º RMS		0.5º RMS
Dynamic	2.8º RMS		0.75º RMS
Accelerometer	Yes		Yes
Range (g)	± 16	± 200	± 160
Resolution (bit)	14	17.5	-
Output noise density (µg/√Hz)	120	5000	200
Gyroscope	Yes		Yes
Range (º/s)	± 2000		± 2000
Resolution (bit)	16		-
Output noise density (º/s/√Hz)	0.026		0.01
Magnetometer	Yes		Yes
Range (G)	± 8		± 1.9
Resolution (bit)	12		-

(C.2) Smartphone Benchmark

Table C.2.1: Benchmark of iOS-based smartphones.

Particulars	Description	iPhone XR	iPhone 11
Launched		2018	2019
Resolution	Width x Height	828 x 1792 pixels	828 x 1792 pixels
Screen Size	Measured diagonally	6.1"	6.1"
Phone width	Chassis dimensions	2.98"	2.98"
Phone height	Chassis dimensions	5.94"	5.94"
Chipset/CPU	Good CPU = better processing of data	Apple A12 Bionic	Apple A13 Bionic
Latest OS available	Latest: iOS 14, Android 11	iOS 14	iOS 14
Turn off updates?		Yes	Yes
Storage/Memory	Enough RAM space is needed to stream/record data from multiple IMUs	64GB/3GB RAM	64GB/4GB RAM
Bluetooth Version	At least v5.0 is recommended by sensor manufacturers	v5.0	v5.0
Bluetooth Chip	Brand & Model	Apple/USI 339S00580	Murata 339S00647
Accelerometer/Gyro	Brand & Model	?	Bosch Sensortec unknown model
Retail Price	For the cheapest variant	$499	$599
Lead Time	Delivery time	2 days (same-day available)	2 days (same-day available)
Return Policy	From when device is delivered	14 days	14 days
Reviews		**iPhone XR**	**iPhone 11**
nanoreview.net	Overall chip score (out of 100)	72	**85**
androidbenchmark.net	CPU Mark Rating (higher = better)	N/A	N/A
iphonebenchmark.net	CPU Mark Rating (higher = better)	4471	**4747**
PCMag.com	Review Score (out of 5)	3.5	**4**
PhoneArena.com	Review Score (out of 10)	**9**	8.5
AndroidCentral.com	Review Score (out of 5)	N/A	N/A
AndroidAuthority.com	Review Score (out of 5)	N/A	N/A

Table C.2.2: Benchmark of Android-based smartphones.

Particulars	Description	Samsung Galaxy A71 5G	Samsung Galaxy A51	OnePlus Nord N100
Launched		2020	2020	2021
Resolution	Width x Height	1080 x 2400 pixels	1080 x 2400 pixels	720 x 1600 pixels
Screen Size	Measured diagonally	6.7"	6.5"	6.5"
Phone width	Chassis dimensions	2.97"	2.9"	2.96"
Phone height	Chassis dimensions	6.4"	6.24"	6.49"
Chipset/CPU	Good CPU = better processing of data	Exynos 980	Exynos 9611	Qualcomm Snapdragon 460
Latest OS available	Latest: iOS 14, Android 11	Android 11	Android 11	Android 10
Turn off updates?		Yes	Yes	Yes
Storage/Memory	Enough RAM space is needed to stream/record data from multiple IMUs	128/6GB RAM	128GB/4GB RAM	64GB/4GB RAM
Bluetooth Version	At least v5.0 is recommended by sensor manufacturers	v5.0	v5.0	v5.1
Bluetooth Chip	Brand & Model	?	?	Qualcomm WCN3998
Accelerometer/Gyro	Brand & Model	?	?	?
Retail Price	For the cheapest variant	$599 ($499 on Amazon)	$399	$179
Lead Time	Delivery time	2 days	2 days	3 days
Return Policy	From when device is delivered	15 days (30 days on Amazon)	15 days	15 days
Reviews		**Samsung Galaxy A71**	**Samsung Galaxy A51**	**OnePlus Nord N100**
nanoreview.net	Overall chip score (out of 100)	**51**	36	32
androidbenchmark.net	CPU Mark Rating (higher = better)	**3195**	2001	2393
iphonebenchmark.net	CPU Mark Rating (higher = better)	N/A	N/A	N/A
PCMag.com	Review Score (out of 5)	**4**	3.5	**4**

PhoneArena.com	Review Score (out of 10)	**8.2**	7	N/A
AndroidCentral.com	Review Score (out of 5)	**4.5**	3.5	N/A
AndroidAuthority.com	Review Score (out of 5)	**3.5**	3	3