Mocopy

Motion Capture and Real-Time Feedback for Physical Therapy

Anthony Russo, Diya Saha, Davin Muramoto, Kenny Kim, Najel Alarcon, Ruben Movsesyan, Peter Han, Ryan Hoang

Executive Summary:

The Mocopy motion capture and real-time feedback system for physical therapy achieves low cost and reliable motion capture functionality through the use of multiple wireless, wearable smart devices worn using a system of adjustable bands. Furthermore, Mocopy provides intuitive real-time feedback to the user through a system of vibration motors attached to wearable devices. The goal of this system is to assist physical therapy patients, athletes, and others to learn new movement patterns and enable physical therapists to recognize and diagnose improper movement patterns to assist in physical rehabilitation, sports training, and more. The companion smartphone app facilitates the setup of the system and allows users to directly access exercise reports and load new exercises to the system as prescribed by a healthcare professional. The entire system requires no additional devices beyond the system of bands, a smartphone, and the charging accessories and can therefore be used from the comfort of one's home, with appropriate approval from a healthcare professional.

Ethics Statement:

As our product deals with collecting the user's health data, cyber-security and misuse of confidential data are concerns that Mocopy takes seriously. Mocopy aims to use this data for nothing else but to benefit the user in their physical therapy needs.

Table of Contents:

Introduction:

Need Statement

Physical therapy patients find it difficult to commute to therapy and to safely carry out exercise routines remotely with minimal supervision.

Goal Statement

Circumvent the cost and time commitment posed to physical therapy patients by visiting a therapist in person and helping them focus on healing in a safe and convenient environment.

Personas

Canva Page (Used to edit): [Link]

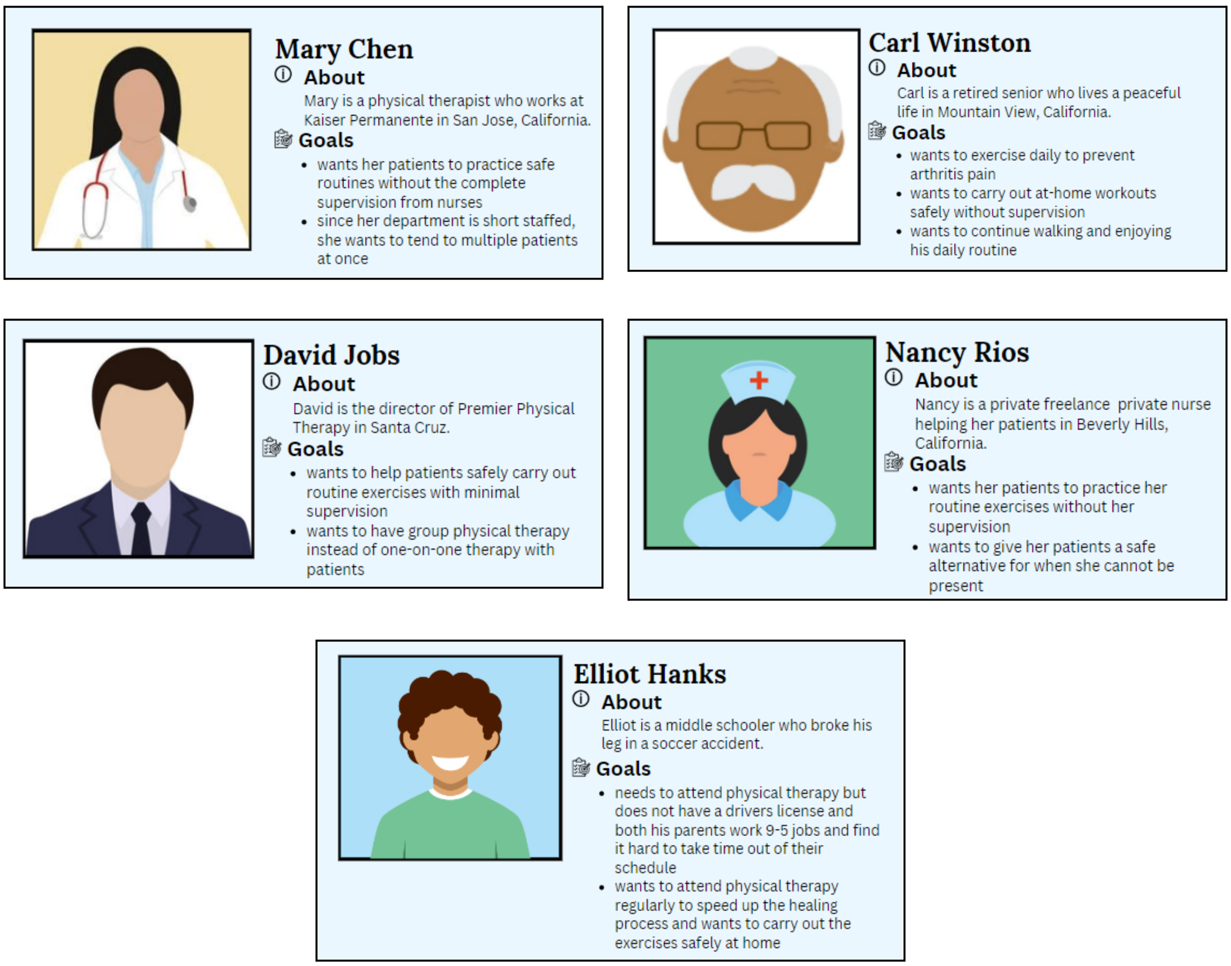


Fig.1. User personas

Existing Designs

The Mocopy system is a unique synthesis of motion capture and telemedicine technologies and real-time “biofeedback therapy”. As such, it indirectly competes with a number of existing solutions implementing any combination of the aforementioned techniques. The following section highlights the differences and similarities between the Mocopy system and existing solutions in addition to the benefits and drawbacks associated with both regarding any particular task.

Sony Mocopi

At a glance the Sony Mocopi motion capture system appears similar to the Mocopy system in a number of ways. Both use an array of wearable inertial measurement unit sensors to capture a user's motion and wirelessly communicate this information to a central device. However, the Sony Mocopi system is primarily designed for entertainment and media purposes such as virtual reality and animation. Moreover, the Sony Mocopi system is intended to interface with a more processor-intensive and costly computer capable of running either the Unity, Unreal, or MotionBuilder engine software whereas the Mocopy system only requires a smartphone on the consumer end. Ultimately, the most notable differences between the two solutions are the intended uses, media versus physical therapy, and the presence/absence of real-time biofeedback to the user. Whereas the Mocopy system is specifically designed to assist in the learning of new movement patterns and presents a convenient interface for health care professionals, the Sony Mocopi system lacks the functionality to reinforce necessary movement patterns and instead focuses on solely capturing motion.

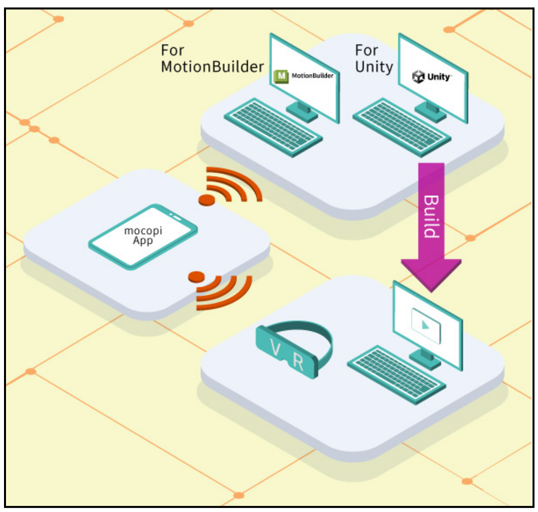


Fig.2. A diagram of the Sony Mocopi system including a smartphone or tablet, professional-grade computer, and virtual reality setup consisting of the motion capture devices. (Source)

Dari Motion

Dari Motion is a motion capture system for physical therapy similar to Mocopy, however, the approach taken by this solution varies considerably in terms of both technology and cost. The Dari Motion system uses an array of camera sensors spread across a dedicated exercise space that contribute to both the system's high cost and infeasibility for at-home use. As the solution requires the installation of many sensors into a dedicated exercise room it can be reasonably assumed that the cost of this system far exceeds that of Mocopy which requires an array of nine wearable sensors and can be used in a shared space without installation. Furthermore, the Dari Motion system relies on the in-person presence of a health care professional to provide feedback to the patient regarding their movement patterns as opposed to Mocopy's real-time vibration feedback techniques which lends itself to telemedicine approaches.

Fig.3. A demonstration of the Dari Motion system using an array of camera sensors installed to the ceiling of a dedicated exercise room. (Source)

Jintronix

Jintronix is a computer vision-based motion capture system made to assist in physical therapy and fitness. Most notably, Jintronix is capable of at-home use due to its relatively low cost and unobtrusive size as the system consists of a screen and a single array of sensors mounted on top of the display. The design closely resembles that of the Xbox Kinect product line. Furthermore, the Jintronix system provides patients with real-time feedback using the installed display for instructions and progress indicators. Jintronix's approach to real-time feedback, however, fails to account for exercises and motions which require the patient to turn their body or head away from the display as they would no longer be able to view instructions and feedback. For comparison, the Mocopy system's vibration motor feedback does not require a patient to face any particular direction after setup and calibration which lends itself to a wider variety of exercises. Last, the Jintronix system, while more portable than a fixed installation such as Dari Motion, cannot be as easily transported as the Mocopy system due to the large metal stand on which the display and sensors are mounted. Without an easy method of transport, the Jintronix system becomes more cumbersome to repair and effectively prevents the use of the system during travel.

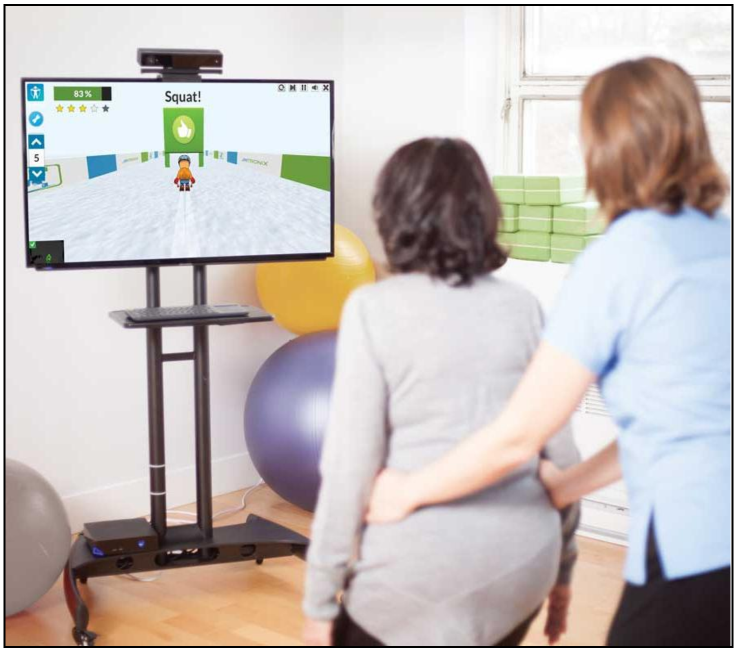


Fig.4.The Jintronix at-home physical therapy system includes a computer display and a top-mounted sensor array to capture patient motion.

Misc. Telemedicine Solutions

The Mocopy system is designed to complement existing telemedicine solutions by providing motion capture and analysis technology spanning a patient's full body and range of motion. However, one can imagine the use of telemedicine solutions, especially video calls, as a simplistic alternative to the Mocopy system by calling a healthcare professional and having the patient stand in view of a device's camera. Specific solutions of this nature include Zoom, WebEx, MDLive, and more. In comparison to the Mocopy system, such a solution is certainly lower cost requiring only a smartphone or laptop device, however, these solutions also lack many of the features necessary to adequately service physical therapy patients and assist healthcare professionals. First, the simple camera arrangement fails to capture the three-dimensional nature of a human's movement. Second, such an approach has far lower precision and higher latency in its real-time feedback as a healthcare professional can only focus and remark on a limited number of patterns at a time while the Mocopy system can provide feedback through multiple bands simultaneously which more quickly and intuitively convey information to a patient during the exercise. Finally, such a solution still requires a healthcare professional to be present at their device for the duration leading to reduced scheduling flexibility and increased cost while Mocopy only requires limited professional presence during the initial tutorial and check-in stages of the process.

Fig.5. A demonstration of an MDLive video call with a healthcare professional. (Source)

At-home Exercise Equipment

On the other end of the technology spectrum lies at-home exercise equipment such as a door frame mounted shoulder pulley. Such equipment is useful for leading a patient's movement in the correct direction, however, these solutions fail to account for any limited mobility due to injury and do not provide any feedback to the patient or data to a healthcare professional. At-home exercise equipment costs vary widely and may be necessary to complement solutions such as those discussed above and Mocopy itself, however, on its own such equipment is thoroughly lacking in its physical therapy capabilities.

Fig.6. A demonstration of a door frame-mounted shoulder pulley exercise device. (Source)

Sustainability Statement

At Mocopy, we are committed to incorporating sustainable practices into every aspect of our project. Over the past six months, we have strived to minimize our environmental impact and promote social responsibility. This sustainability statement outlines our guiding principles and the steps we have taken to achieve our sustainability goals.

Environmental Responsibility

a. Energy Efficiency: Throughout the project, we have prioritized energy-efficient practices by utilizing low-power hardware and optimizing code efficiency.

b. Waste Reduction: We have emphasized waste reduction by adopting a paperless approach and encouraging digital collaboration.

c. Carbon Footprint: We have actively sought to minimize our carbon footprint by promoting remote work and encouraging the use of eco-friendly modes of transportation for commuting.

Social Responsibility

a. Diversity and Inclusion: Mocopy is dedicated to fostering a diverse and inclusive environment. We have embraced diversity in our team composition and respect different perspectives and backgrounds.

b. Ethical Sourcing: We have prioritized ethical sourcing practices by using sustainable materials and minimizing our own environmental impact.

Long-term Impact

a. Scalability and Flexibility: Mocopy has designed the project to be scalable and adaptable to future needs. By considering the project's long-term impact and planning for growth, we aim to minimize the need for major overhauls or replacements, thereby reducing waste and resource consumption.

By incorporating these principles into Mocopy, we aim to create a positive impact on the environment and society as a whole. We understand that sustainability is an ongoing journey, and we remain committed to continuously improving our practices, integrating new technologies, and working towards our vision for a more sustainable world.

Design

Aesthetic Prototype

The goal of the aesthetic prototype was to create a sleek and comfortable device that is as easy on the eyes as it is on the body. By implementing the electronics housing as a distinct feature on each band it can be used as a point of reference to guide the user to orient the device correctly on their limbs. The design also places vibration motors on the interior of the band strap at even and well-spaced intervals to ensure the direction of vibration feedback is clearly identifiable by the patient. The device is designed to be strapped on easily so it is accessible to patients of all needs. These bands need to be comfortable. If the patient is performing physical therapy for multiple hours at a time, they need the bands to be both comfortable and sweat resistant for even some of the more intense exercises.

For the aesthetic prototype, it was ensured that the device has a discernible top so that the patient knows how to orient each band. Additionally, the motors were adequately spaced out to facilitate easy determination of the band's movement direction. The device was designed to be easily strapped on, making it accessible to patients with varying needs. Comfort was prioritized for the bands as they should be comfortable and sweat-resistant, even during more intense exercises, especially when the patient engages in physical therapy for extended periods.

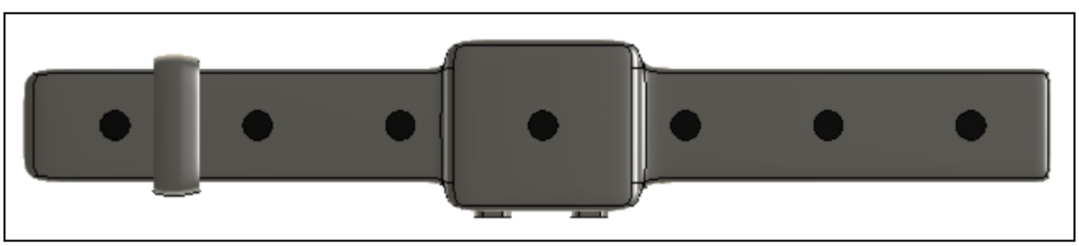


Fig.7. Armband

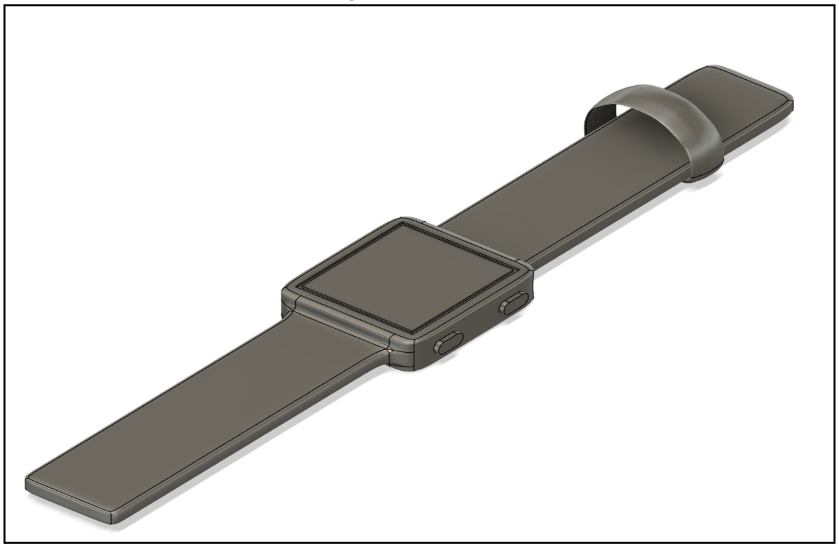


Fig.8. Armband facing up

A model of the band and where the vibration motors will be located

To most accurately indicate the direction of movement the patient should perform, the vibration motors are embedded along the inside center of the band and vibrate according to the direction of the calculated error. The band will be attached to itself with a magnetic strap to allow for the simplest ease of use when taking on and off the band. The band will have two buttons on it. One will be the power button and the other will be a Bluetooth pairing button. The Bluetooth pairing button will be used to link two bands together, whether it be the central band and the middle band or the middle band and the external band. The housing in the center of the band will be there to hold the electronics for each band. The housing will contain the inertial measurement device and the wirelessly connected embedded processor.

Design for Manufacture

The design of Mocopy is inherently suited for a simple and streamlined manufacturing process. Several factors contribute to this efficiency including a minimal number of components, a modular design, and standardized interfaces, ensuring a smooth production workflow.

First, Mocopy's design keeps the number of components to a minimum with a single small form factor PCB with a detachable lithium-ion battery, simplifying the overall structure and functionality of the device. With fewer parts, the manufacturing process becomes more straightforward, reducing the likelihood of errors or complications during assembly. Second, Mocopy incorporates a modular design approach, where individual components can be easily interchanged or replaced. This modularity facilitates efficient production by allowing for independent manufacturing of modules, which can later be integrated into the final product. This empowers future developments by allowing each component’s manufacturing process from being updated independently of the others and enables flexibility in the production of the final product.

Next, the use of standardized interfaces between components, such as the 2-pin JST-PH connector on the internal battery, further simplifies manufacturing. By employing established connectors, fasteners, and attachment mechanisms, compatibility and integration between different parts are ensured. This standardization not only simplifies the sourcing of components but also reduces the time and effort required for assembly. Mocopy is designed with manufacturability in mind throughout. Factors such as component accessibility, ease of manufacturing individual parts, and optimization of assembly processes are considered at each design process step. This proactive approach addresses potential manufacturing challenges early on, resulting in a smoother production process. The selection of readily available and cost-effective materials such as acrylonitrile butadiene styrene (ABS) further contributes to a streamlined manufacturing process. By using materials with established manufacturing processes, the supply chain is simplified, reducing lead times and production costs.

Additionally, quality control processes are easily implemented due to Mocopy's limited component count and, thus, limited points of failure. With fewer components and standardized interfaces, it becomes easier to perform rigorous quality checks at various stages of manufacturing, ensuring that each device meets the desired quality standards.

Last, the manufacturing process for Mocopy is designed to be scalable by using industry-standard materials which can be sourced at any location and by minimizing the number of steps in a single assembly run. It can efficiently accommodate increased production volumes without requiring significant modifications, allowing for rapid scaling up in response to market demand.

Design for Maintenance

The design of Mocopy is optimized for easy repairability. Here are some key considerations incorporated into the design to achieve this:

The device is designed with modular components that can be easily detached and replaced individually such as replaceable bands. This allows for targeted repairs without the need to replace the entire device. Each band has standardized connectors that can be easily disconnected and replaced. The internal components of the device are easily accessible for repair purposes. These components are easily accessible so that a technician can go in and repair them without having to replace the entire device. The device is designed to use standardized parts that are readily available. This simplifies the repair process as it reduces the need for sourcing specialized components. Using widely available components also ensures that replacements can be easily obtained.

The device will have clear labeling and documentation to assist with troubleshooting and repair. This includes labeling components, connectors, and interfaces, as well as providing detailed repair manuals or guides. The documentation will be easily understandable and accessible to the technicians performing the repairs. Mocopy will incorporate a system that provides error reporting and diagnostics to identify faulty components or issues. This is achieved through integrated sensors or software that will detect and report malfunctions. The device will utilize standardized connectors and cables for interconnecting various components. This allows for easy replacement or swapping of components when necessary. Using connectors that are commonly available in the market ensures compatibility and simplifies repairs. The device will be designed with durability in mind, considering the expected wear and tear in a physical therapy environment. Using robust materials, reinforcing critical areas, and employing protective measures can reduce the likelihood of damage and the need for repairs. By incorporating these design principles, Mocopy can be engineered to be easily repaired, reducing downtime and maintenance costs while extending the lifespan of the device.

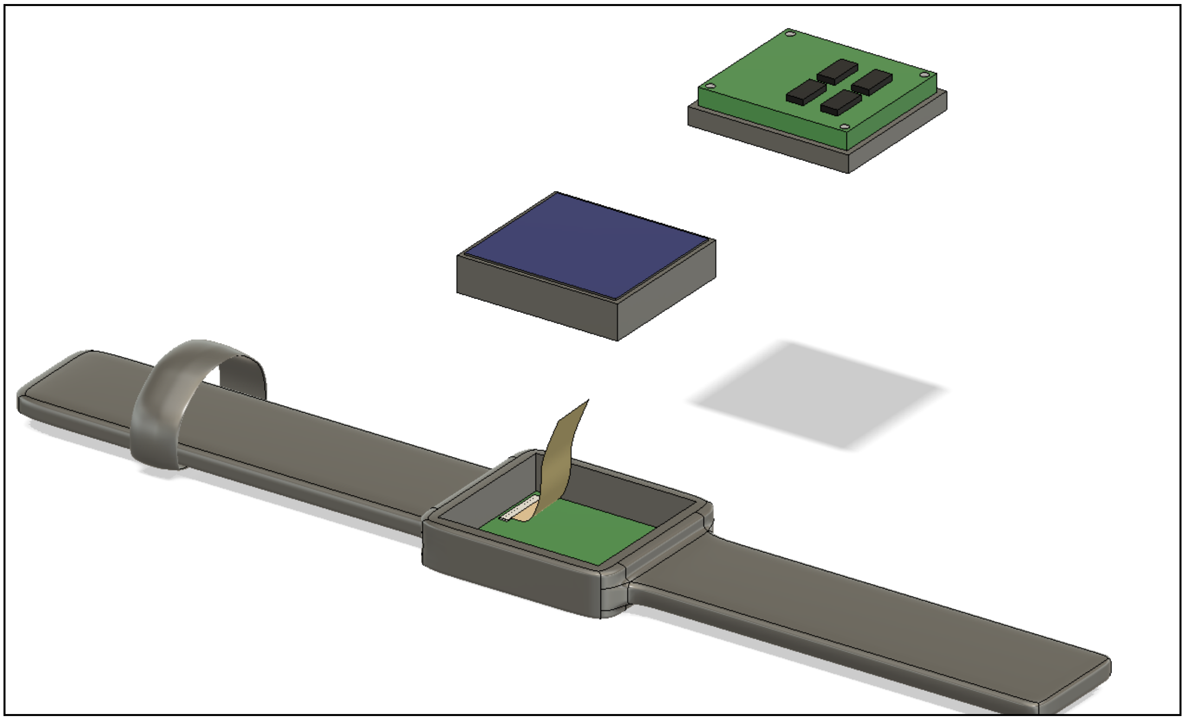
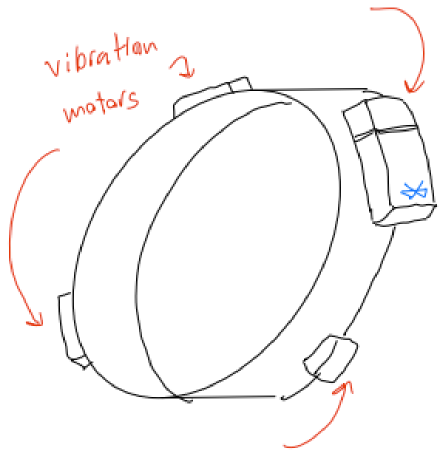


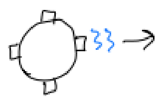
Fig.10. Armband facing up

The final design will have the components all mounted on the flat panel and therefore will be easy to remove and install a new panel, as it isn't connected to the band with a wire or screw. Since it is a flat panel, it will be simple for users to remove as they won't have to fidget or wrestle the band to remove the panel.

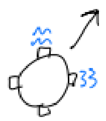
Newly manufactured components include the clips to attach the vibration motors to the insides of the wearable bands and the electronics housing for the wireless smart devices.
The clips can be easily manufactured using an injection molding process and then attached fabric straps by a machine that simply folds the locking clip over the fabric and presses them down in place before forming a loop with the strap. The electronics housing will be designed to use snap-joints to reduce (or even eliminate) the number of screws necessary to securely enclose the device. Such a plastic housing could also be manufactured using an injection molding process similar to the aforementioned clips. The casing for the electronics housing will also be designed around a flat back panel which will then have a deeper casing attached over the panel to allow for easy assembly before the final casing is enclosed.



Microcontrollers, vibration motors, and batteries will be attached to a wristband for Mocopy. The wristband itself will be made of silicone, housings will be created on 3D-printed surfaces.



If the vibration motors all vibrate on one side, then the user should move that body part to the vibrating side.



If the vibration motors are vibrating on two adjacent sides, they should move the body part to the diagonal between the two vibrations.

Block Diagrams

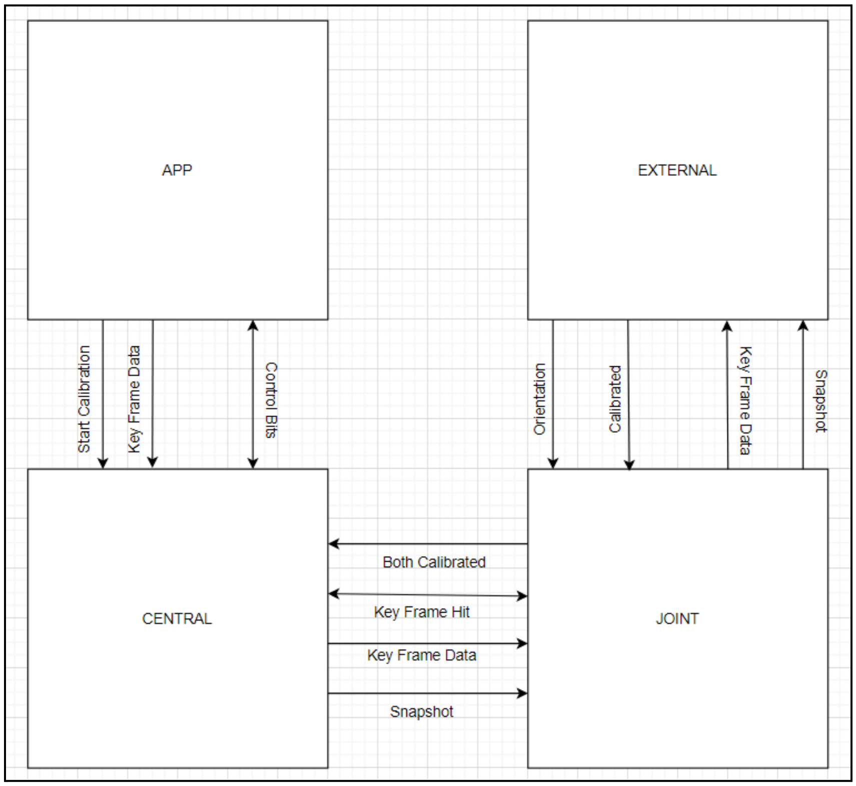


Fig. 9. Top-Level Block Diagram

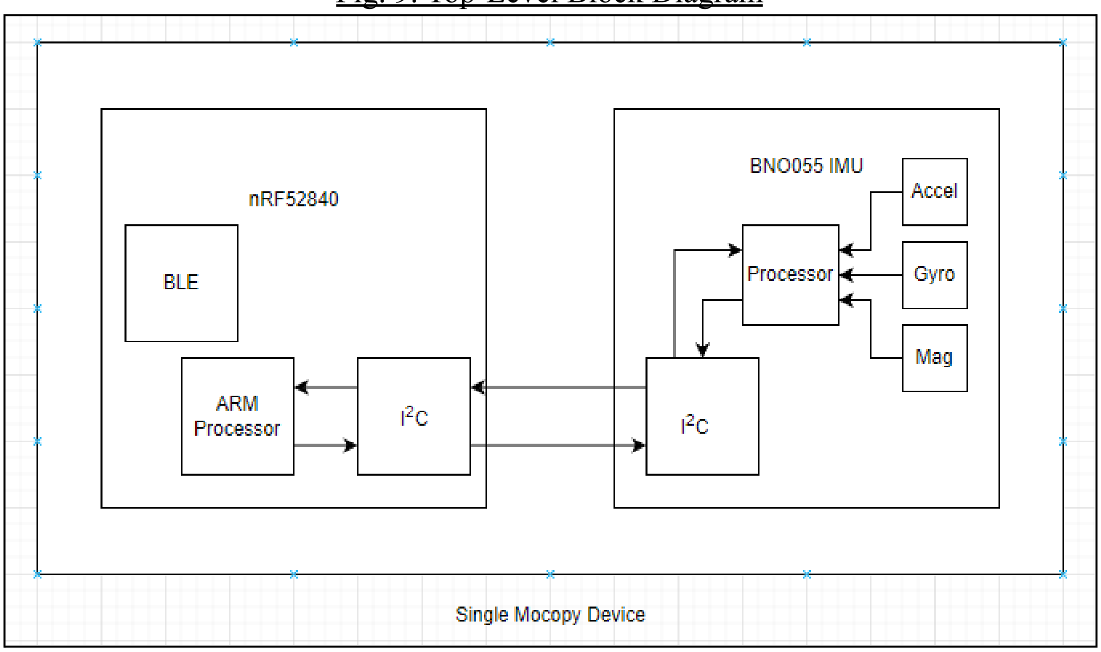


Fig. 10. Low-Level Block Diagram

Wiring Diagrams

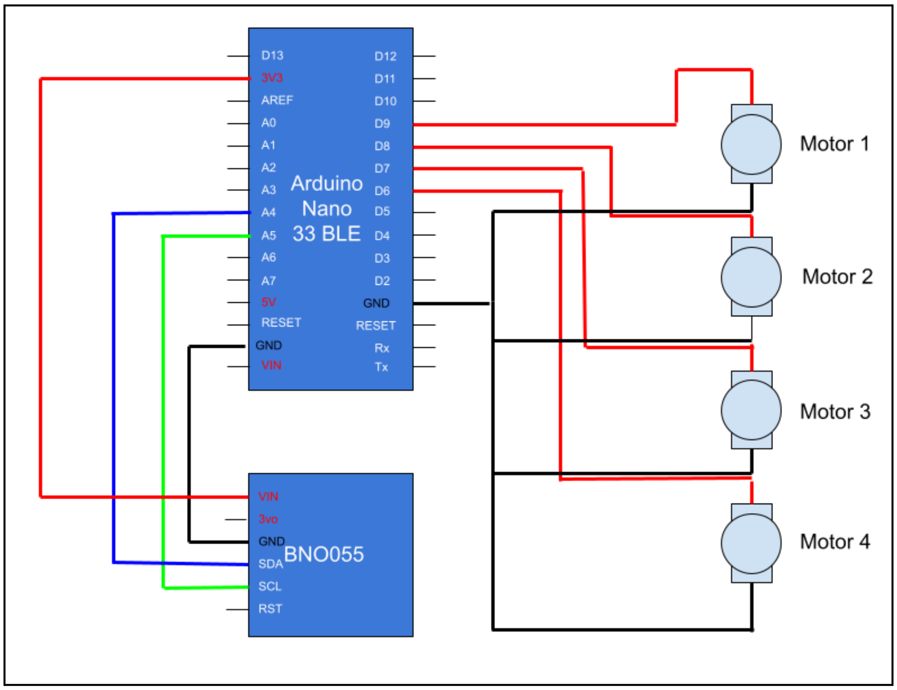


Fig. 11. Wiring diagram of a single Mocopy device

State Transition Diagrams

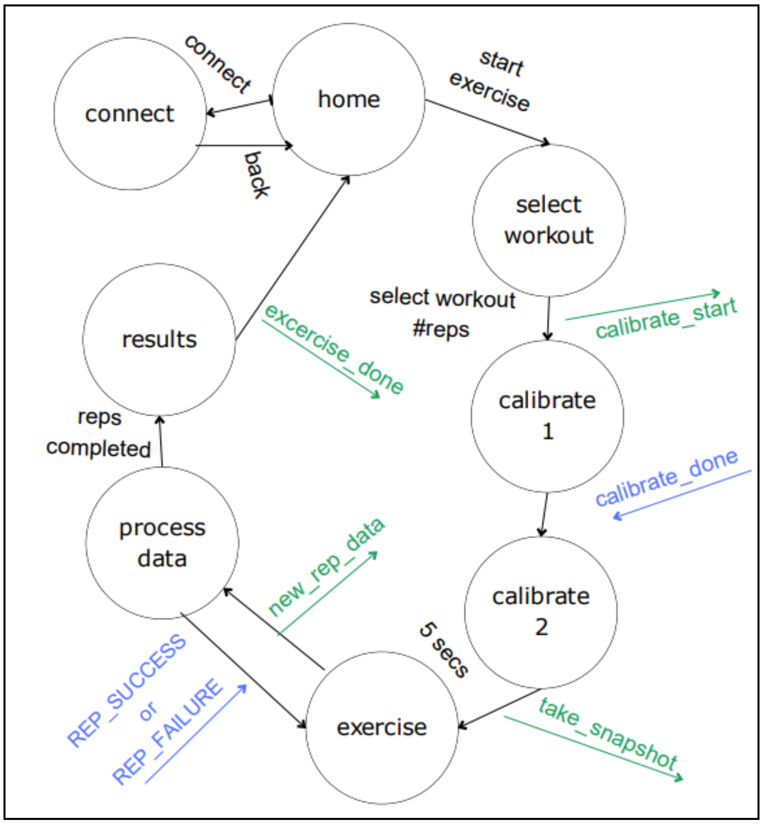


Fig. 12. App State Machine

App:

This state machine is contained within the smartphone and relies on communicating with the hardware state machine in order to function properly. The app's state machine begins in the HOME state and the user must press the connect button in order to transition to the CONNECT state in which the smartphone scans for the Mocopy central device and connects to it. After this, you must click on the back button to go back to the HOME state. After this, you are able to click on the start exercise button to transition to the SELECT WORKOUT state. In this state, the user is provided with a selection of workout icons to which they can choose. After one is chosen, an alert pop-up comes up to select the number of reps you want to do. Choosing the rep number triggers the transition to CALIBRATE 1 state. Also, in this transition, the CALIBRATE_START characteristic is sent to the hardware state machine for processing on their end. In this state, the app waits for the CALIBRATE_DONE characteristic from the hardware state machine on the central device which confirms the IMU is calibrated. After receiving this, the app transitions to the CALIBRATE 2 state where's 5-second timer is started to allow the user to get in position for the exercise to start. After the timer hits zero, the TAKE_SNAPSHOT characteristic is sent to the central device and we transition to the EXERCISE state. Within this state, the app sends the keyframe data for the first keyframe through a characteristic to the central device doing so transitions to the PROCESS_DATA state. In this state, the app waits for a response from the central device if the keyframe failed or succeeded. Based on what happened the app stores the data so that an overall result of the exercise is generated. After this happens the app goes back to the EXERCISE state. This process of state transitions between EXERCISE and PROCESS_DATA repeats for all keyframes of one rep and then over and over again til all reps are completed. Once all reps are completed, the app transitions to the RESULTS state where the results are displayed for the exercise which can be accessed later by the user and the doctor. Then the user can press a button to return back to the HOME state and writes to the characteristic EXERCISE_DONE indicating to the hardware state machine that the current exercise has been completed.

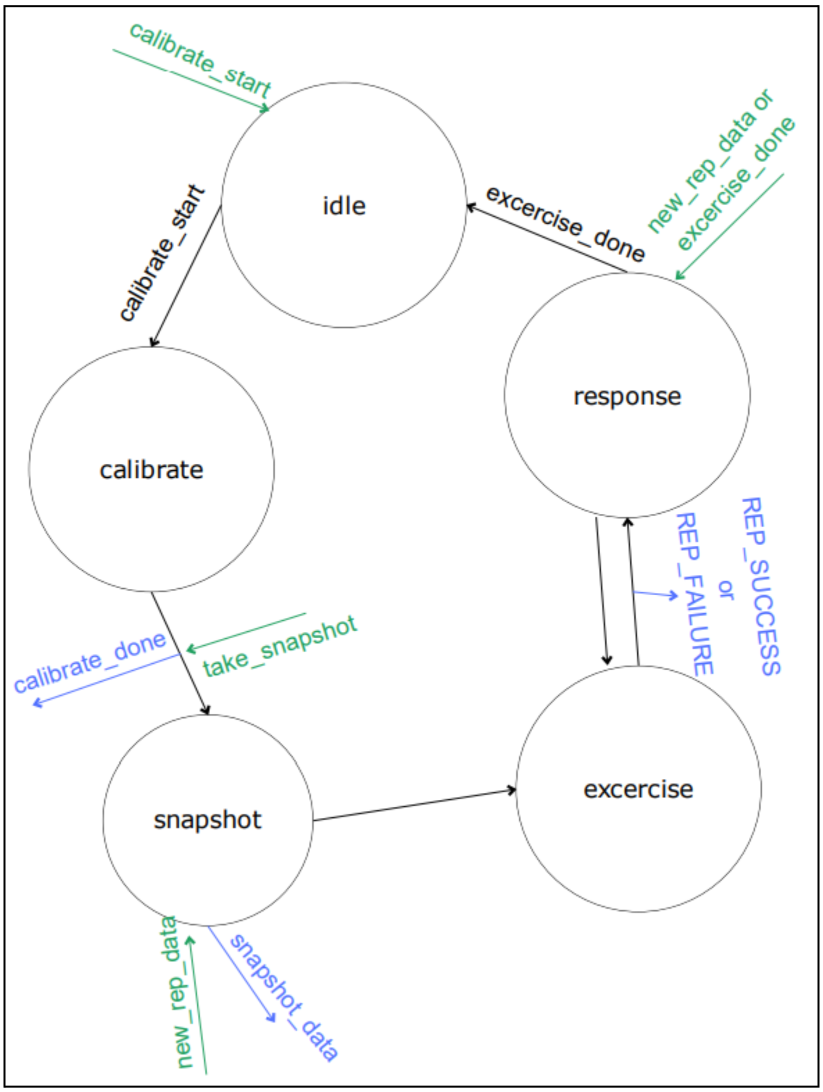


Fig. 13. Hardware State Machine

Hardware:

This state machine is contained within the central band, which handles all the wireless signals being sent to the app for further processing. It begins in the IDLE state and when it receives the signal to begin calibrating from the app, it will transition to the CALIBRATE state, where it will wait until it receives the “calibration successful” signals from both the joint and external devices. Once it receives that both of them are calibrated, it will send a signal to the app that both are calibrated and transition to the SNAPSHOT state. Once in the SNAPSHOT state, it will wait for a snapshot signal from the app. Upon receiving a “take_snapshot” signal from the app, the central device will signal to all devices to record their current orientations for use as a reference throughout the exercise. Then it will transition to the PRE-EXERCISE state. The PRE-EXERCISE state will wait for the app to send a key frame and once received, it will forward it to the other devices and transition to the EXERCISE state. In the EXERCISE state, it waits for a signal from the peripheral devices indicating whether or not the key frame pose / orientation was reached. The central device will then write to the app whether the keyframe pose was successfully hit or not and then transition to the RESPONSE state. Once in the RESPONSE state, it will wait for either the next keyframe data or a signal from the app that the exercise is done. If the exercise is complete, it will return to the IDLE state. If not complete, it will read in the new keyframe data from the app, forward the data to the peripheral devices, and then transition to the EXERCISE state once more.

Technology

The wearable devices in this design needed to fulfill a number of criteria in regards to ease of use, power consumption, form factor, and precision. To this end a number of technologies were considered spanning a range of applications from wireless connectivity, charging capabilities, sensor technologies, and choice of programming language. Due to the relatively lax bandwidth and latency requirements of the Mocopy system Bluetooth Low Energy (BLE) was selected as the wireless communication protocol for this system. The lower power consumption of BLE also makes it well suited for small form factor devices that cannot contain a larger battery such as the Mocopy wearable bands. By connecting each of the bands into a mesh topology data can be sent and received across multiple devices through the central device and finally to the host smartphone. Lithium ion batteries were chosen for power supply as they are available in small form factors (e.g. 11.5mm x 31 x 3.8mm) and are able to be recharged for continued use. Qi Wireless Charging was selected as the method to recharge the included batteries due to its convenience. The BNO055 absolute orientation sensor was selected for its combination of MEMS accelerometers, gyroscopes, and magnetometers which features a minimal footprint on the device form factor and low power operation. Furthermore, a combination of low-level programming languages including Arduino-C and C++ were selected to allow for the fine-tuned control of device memory and further improve power consumption.

Simulations

To simulate the information being sent from the Mocopy device to the companion app, dummy data was utilized. Dummy data is synthetic or fabricated information that mimics the format and structure of real data without containing actual patient or sensor readings. The process involved generating artificial data that resembled the expected data packets transmitted by the Mocopy device during physical therapy sessions. This data was designed to represent various parameters and measurements, such as joint angles, muscle contractions, and exercise performance metrics. A data generation mechanism or algorithm was established that followed the expected data patterns and formats. This algorithm simulated the behavior and characteristics of the actual data that would be transmitted from the Mocopy device to the companion app. The dummy data was then transmitted through the established communication channel between the device and the app (e.g. BLE), replicating the data transmission process in a controlled environment. The app received and processed this synthetic data as if it were real-time information from the Mocopy device. Using dummy data allowed the assessment of the functionality and performance of the app's data reception and processing capabilities without relying on actual patient data or the physical Mocopy device. It enabled the simulation of different scenarios, validate the app's ability to handle various data inputs, and testing the responsiveness and accuracy of the app's data visualization and analysis features. By employing dummy data, seamless integration of the Mocopy device with the companion app was made simpler and ensured the app's compatibility and reliability in receiving, interpreting, and presenting the transmitted data. This approach facilitated thorough testing, troubleshooting, and refinement of the app's functionality before incorporating real patient data, contributing to a more robust and dependable user experience.

Evaluation

Functional Prototype

Bluetooth:

To evaluate the feasibility of the Mocopy design a functional prototype was created using a combination of 3D printer, test boards, and a mobile app framework. This prototype featured an Arduino Nano 33 BLE board to fulfill the embedded processing and wireless connectivity needs of each wearable device in addition to an Adafruit BNO055 breakout board to provide orientation sensing capabilities. These two devices were connected to a shared protoboard using I2C wired communications and then connected to a wearable band on a custom 3D printed mount. Each wearable band featured a single large belt loop-style piece to host the electronics on the custom mount and three other smaller belt loop pieces to host the vibration motors / LEDs.

The devices were then wirelessly connected to and configured by the companion app on a host Android smartphone via BLE. The companion mobile application was created using React-Native and the BLE communication was handled by the react-native-ble-plx library. As far as BLE goes on the app side, the smartphone acts as the central device to the central Arduino in the sense that it initiates a connection by scanning for the Arduino's advertisement and connects to it. The Arduino and smartphone only communicate over one service named the central service. This service has three characteristics: KF_Data, KF_Hit, and Control. KF_Data is written to by the app only and sends exercise keyframe values to the central. KF_Hit is read by the app to determine if a keyframe was hit or not. Finally, the control characteristic is both read and written to by the app and helps us navigate the app and hardware state machine. Some things that it controls are the calibration statuses when to take the snapshot, and when the overall exercise is finished.

The Bluetooth Low Energy interface created to facilitate the exchange of information across the network of devices relied on a mesh network topology with many of the devices behaving as both a central and a peripheral device to "daisy chain" connectivity and disperse information throughout the system of bands. The central band device in the configuration acts as a router sending information to each of the joint devices and relaying information back to the smartphone throughout an exercise. An overview of the BLE services and characteristics required for a single limb is detailed in the table below where the letter "R" in parenthesis indicates a characteristic is enabled for reading and the letter "W" indicates a characteristic is enabled for writing.

Device	Central Band	Joint Band	External Band
Role	Central	Central-Peripheral	Peripheral
Characteristics	KF_Data (W)	KF_Data (W)	KF_Data (W)
	KF_Hit (RW)	KF_Hit (RW)	Ext_Orientation (R)
	Control (RW)	Snapshot (W)	Snapshot (W)
		Both_Calibrated (R)	Calibrated (R)

Fig. 14. BLE Characteristics for each band

The "KF_Data" characteristic relays information regarding the current keyframe data, i.e. the current orientation each device should reach to count as a keyframe hit. This characteristic includes 24 bytes of data to store six floating point numbers. The "KF_Hit" characteristic is a single-byte boolean that originates with the joint device and indicates whether or not the current keyframe has been reached by the user. The "Control" characteristic is a single-byte integer characteristic used to send and receive control instructions between the smartphone, central device, and other devices. The "Snapshot" characteristic is a single-byte boolean characteristic that is sent to each device by the smartphone to indicate that the device's current orientation needs to be captured as a reference position for the remainder of the exercise. The "Ext_Orientation" characteristic is a twelve-byte three-float characteristic containing the orientation of the external device and is read by the joint device to determine the local error vector and whether the current keyframe has been reached. Last, the "Calibrated" and

"Both_Calibrated" characteristics are both one-byte boolean characteristics to indicate whether the external device and both the joint and external device are correctly calibrated respectively. These characteristics are sent to the smartphone to ensure the system is calibrated before beginning an exercise.

Arduino:

The prototyped board is done on an Arduino Nano BLE 33. This board was chosen for prototyping for multiple reasons. First of all the Arduino Nano BLE 33 has a very compact form factor. This makes it suitable for prototyping a wearable device. The Arduino Nano BLE 33 integrates Bluetooth Low Energy (BLE). With the built-in Bluetooth Low Energy capability, we can enable wireless communication between the different parts of the physical device and wireless communication between the physical device and its companion app. The Arduino Nano BLE 33 comes with a variety of onboard sensors, including an accelerometer, gyroscope, and magnetometer. These sensors can be utilized along with the extra BNO055 sensor to generate the most accurate motion data for the Mocopy device. The Arduino Nano BLE 33 is as the name implies an Arduino board, meaning that it is part of the Arduino ecosystem. This makes it great for prototyping because there is an extensive online support community. There is a wide range of libraries, examples, and tutorials available to make prototyping and development of applications using the Arduino platform as developer-friendly as possible. The Arduino Nano BLE 33 can be programmed using the Arduino IDE, which uses a simplified version of the C++ programming language. This ease of programming allows for rapid prototyping and iterative development of the device software. The additional support of the Arduino third-party libraries further simplifies the development process of the Mocopy device. The Arduino Nano BLE 33 is relatively affordable compared to other development boards with similar capabilities. This cost-effectiveness makes it an excellent choice for prototyping the Mocopy device. The Arduino Nano BLE 33 is based on open-source hardware and software, which means that the design and specifications are publicly available. This allows for customization, modification, and integration of the Arduino Nano BLE 33 into the Mocopy device design.

Mobile Application:

The functional prototype of the Mocopy device features a companion mobile application on a smartphone device. A companion app is beneficial for both the prototype and the final product for device controls, data review, and user guidance. The companion app displays real-time or recorded motion capture data in a visual format such as a 3D animated human model. This allows patients and therapists to play back their movements visually, making it easier to understand and analyze their performance during physical therapy sessions. Visual feedback can help identify areas for improvement and track progress over time. The app also tracks and monitors the patient's performance metrics, such as range of motion, joint angles, or exercise completion. This data is logged and used to set goals, measure progress, and provide personalized feedback. Performance monitoring also serves to motivate patients and allow therapists to track their rehabilitation progress more effectively. Furthermore, the mobile app provides step-by-step instructions, demonstrations, or video tutorials for different physical therapy exercises which can greatly reduce the risk of injury during the patient's performance. Such guidance assists patients with the correct exercise techniques, timing, and repetitions, ensuring they perform exercises accurately and safely. Clear instructions can help patients adhere to their prescribed therapy regimen even when unsupervised.

The mobile app was designed to be simple and aesthetically pleasing with the goal of being as unobtrusive to the user's primary focus, physical therapy. An intuitive user interface greatly improves a user's ability to operate the system in an unsupervised manner. Consideration was given to the fact that a significant portion of Mocopy's anticipated end users are senior citizens and may have limited familiarity with smartphone technology. With that in mind, the menus were designed to be clean, minimizing clutter to direct the users' focus on the most relevant information at each stage. By doing so, the aim was to prevent user confusion and ensure they do not navigate to unintended screens. Furthermore, emphasis was placed on the app's user-friendliness and reliability. Given that troubleshooting may pose challenges for the end user if they encounter errors, efforts were made to ensure tasks such as connecting to the bands, calibration, and exercise performance are as simple and error-free as possible.

The companion app also allows the customization of therapy programs based on individual needs and goals. Therapists can create personalized exercise routines and set specific parameters for each patient such as mobility limitations due to personal ability or injury. Additionally, the mobile app can adapt to the patient's progress, adjusting the difficulty or intensity of exercises as they improve. Importantly, this includes setting limits to ensure only safe exercises are attempted by patients, especially those with possible injuries or physical limitations.

The app sends reminders and notifications to patients to encourage adherence to their therapy schedules. It can remind patients of upcoming sessions, provide exercise reminders, or send motivational messages. Such reminders can help patients stay consistent and engaged in their therapy, improving overall outcomes. The companion app facilitates data sharing between patients and their healthcare providers, such as physical therapists or physicians. Using the host smartphone's internet connection, the application allows patients to securely share their progress reports, exercise logs, or recorded sessions with their healthcare team. This promotes better collaboration, remote monitoring, and more informed decision-making. By recording and storing patient data over time, the companion app can generate long-term reports and trends to guide rehabilitation efforts. This enables therapists to analyze the effectiveness of the therapy program, identify patterns, and make informed adjustments to optimize the treatment plan. Long-term tracking can also provide valuable insights for research or clinical studies with the patient's consent.

Overall, a companion app for the Mocopy device can greatly enhance the effectiveness, engagement, and convenience of physical therapy using visual feedback, guidance, and customization which further enables patients to perform exercises correctly, stay motivated, and achieve better rehabilitation outcomes. Additionally, the app allows for seamless communication and collaboration between patients and healthcare providers, promoting continuity of care and informed decision-making.

Hardware:

The Mocopy motion-capture system incorporates motion sensors, such as accelerometers, gyroscopes, and magnetometers to estimate each band's current attitude/orientation in space in real-time. By capturing precise motion data, they enable the device to accurately track and analyze the patient's movements during physical therapy exercises. Mocopy utilizes bands that are worn by the patient with built-in vibration motors for vibration feedback. These small motors provide controlled feedback to emulate a "resistance" to incorrect movements or "assistive force" during exercises. More specifically, the patient is intended to imagine these vibrations as forces preventing incorrect movements by pushing the errant limb from incorrect positions or, alternatively, forces assisting movement by pushing the errant limb into the correct position. The motors can be adjusted to apply the desired level of vibration intensity based on the prescribed therapy program. This feature allows patients to perform exercises with customized feedback, promoting targeted muscle engagement and enhancing therapeutic outcomes. Each device incorporates a microcontroller, such as the nRF52840, as the brain of the device. The microcontroller receives input from the motion sensors and processes the data to determine the patient's movements and positions accurately. It also controls the motorized bands, sending signals to adjust the resistance or assistance provided. The microcontroller acts as the central processing unit, coordinating the various hardware components and facilitating real-time feedback and interaction. A rechargeable lithium-ion battery was chosen as the power source to operate each device's internal hardware components. Such a battery must provide sufficient power to successfully drive the microcontroller, wireless communications, sensors, and motorized bands for the duration of the physical therapy session. The power supply should be designed to be long-lasting, ensuring that the device can be used for extended therapy sessions without interruption. In the case of the functional prototype, we have used a direct connection to power to avoid any complications while prototyping. Mocopy includes wireless connectivity, namely Bluetooth, to enable communication with external devices like smartphones or companion apps and internal communication within the multiple devices that make up the entirety of the hardware of the device. This connectivity allows for data transfer, synchronization, and control of the device through a user-friendly interface, enhancing the user experience and enabling remote monitoring and analysis by healthcare providers. Each wearable device features a user interface, such as tactile buttons, to provide the patient with a mechanism to toggle power to the device and initiate Bluetooth connections. Finally, the hardware components of Mocopy are housed in a lightweight and ergonomic casing. The design aims to ensure comfort and ease of use for the patient, allowing the device to be strapped securely, comfortably, and easily to different body parts. The strapping mechanisms should be adaptable to accommodate patients with varying needs and body sizes.

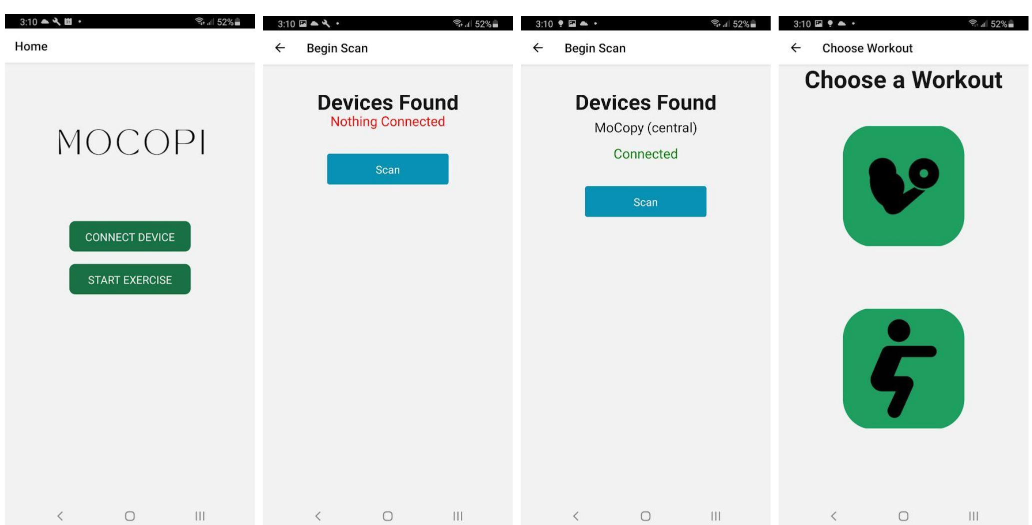
3D Printing:

The initial step in using 3D printing for Mocopy prototyping is creating a digital 3D model of the device. This can be done using computer-aided design (CAD) software. The 3D model allows for precise visualization and modification of the design before manufacturing. Design iterations can be easily made by adjusting the digital model, allowing for rapid prototyping and evaluation of different design concepts. Once the digital model is ready, it can be translated into physical prototypes using 3D printing technology. 3D printers work by depositing layers of material (such as plastic or resin) based on the digital design. This layer-by-layer approach enables the creation of complex geometries and intricate details accurately. With 3D printing, it is possible to create functional prototypes of Mocopy that closely resemble the final product. The prototypes can include housing, strapping mechanisms, and compartments to house the hardware components. By printing these functional prototypes, it becomes possible to assess the ergonomics, fit, and functionality of the device in real-world scenarios. 3D printing allows for easy customization of prototypes. Since the digital design can be modified quickly, specific changes can be made to accommodate different patient needs or design improvements. This flexibility enables the rapid adaptation and refinement of the Mocopy prototype based on user feedback or specific requirements. During the prototyping process, 3D printing can be used to create separate parts that can later be assembled and integrated with the hardware components. This allows for the testing and validation of the mechanical interfaces, ensuring proper alignment and functionality. 3D printing significantly reduces the time and cost associated with traditional prototyping methods. It eliminates the need for expensive tooling or molds, as well as the manual labor required for complex manufacturing processes. 3D printing also enables faster iteration cycles, facilitating a more efficient development timeline. The 3D-printed prototypes can undergo thorough testing and validation to assess their performance, durability, and user experience. This includes evaluating the fit and comfort of the strapping mechanisms, assessing the functionality of the hardware components, and conducting user trials to gather feedback. Based on the feedback and test results from the 3D-printed prototypes, design refinements can be made before moving to the next stage of production. Adjustments to the geometry, materials, or structural elements can be incorporated into the digital model, allowing for improved iterations of the design.

Outstanding Issues

- Electromagnetic Interference caused by vibration motors might interfere with the magnetometer.
- Ensure only RoHS-compliant materials are used, notably lead-free solder.
- How will the wireless devices and mobile application securely and privately deliver the patient’s information to only approved recipients such as their therapist?
- What should we do if IMU polling falls out of sync across multiple devices? Should the data include timestamps? Should the system be designed to tolerate such an error?

- What should we do if there is packet loss? What if packet loss only occurs on some devices? Should we attempt to tolerate the missing data, use interpolation, or some other technique?
- Will vibration motors be strong enough to be felt through clothing? Will the direction of the vibration be discernible?
- Will the straps/bands remain at the same location on the patient even after lots of movement?
- Will the design of the electronic housing ensure the patient's safety from electric shock?
- Could the bands restrict blood flow if they are too tight? If the bands are to be user-adjusted then how can we best instruct them to adjust the bands accordingly?
- The bands and device housing shouldn't cause too much sweat buildup and/or cause the patient to become clammy.
- A patient could injure themselves while attempting to equip the bands on their body, especially seeing as they might have some physical limitation or injury. How can we prevent the patient from accidentally injuring themselves and ensure they only attempt to equip the devices where it is safe to do so?
- How will the device and/or companion app notify the user when the battery is low?
- How will the system function without an active internet connection? Should the exercise data be stored securely and locally until a connection is made or should the system require a connection throughout its use?
- How will pre-recorded exercises account for the limited mobility of different users?
- Will each vibration be strong enough to elicit a response, but not too overpowering that the patient will not be able to feel the direction from which the vibration is coming from?



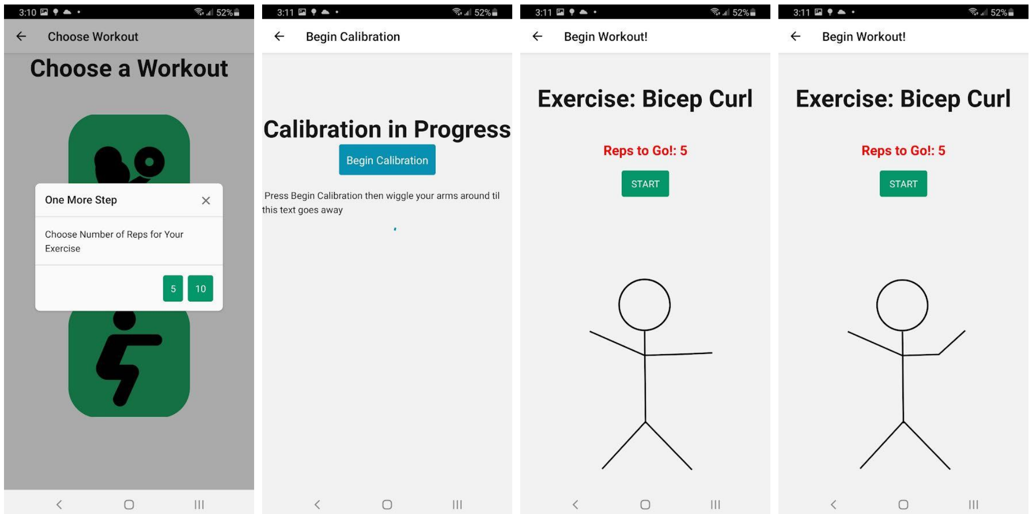


Fig. 15. Screens for the Mocopy app

High-fidelity of the app
https://www.figma.com/proto/GIsAs5Ufcct2EGwlMS4Ksb/123A?node-id=5%3A575&scaling=scale-down&page-id=0%3A1&starting-point-node-id=5%3A575

Motor Specifications:

Current Rating	85mA
Voltage Range	DC1.5-3.7V
Rated Speed	12000 RPM
Body Size	10mm x 3mm / 0.39” x 0.08”

Wristband Specifications:

- I2C communication between Arduino Nano 33 BLE and BNO055
- 3.3-5V power for Arduino Nano 33 BLE
- Modular connectivity
- Powered by 5V battery

Addendum

Supplementary content provided after receiving feedback from Harrison.

Source Code Repositories:

- https://github.com/mocopy/mocopy-planning
- https://github.com/mocopy/mocopy-app
- https://github.com/mocopy/mocopy-devices

Github Projects Page and Gantt Chart:

- https://github.com/users/mocopy/projects/1

Appendix 1 - Problem Formulation

Conceptualization

To aid in the conceptualization of a solution to the need statement a number of methods were employed including a mind map diagram and a round-robin brainstorming session. The mind map technique starts at a central node indicating the general need from their related topics and subtopics were stemmed from the central node to explore all possible approaches to solving the problem. The figure below illustrates the result of this exercise.

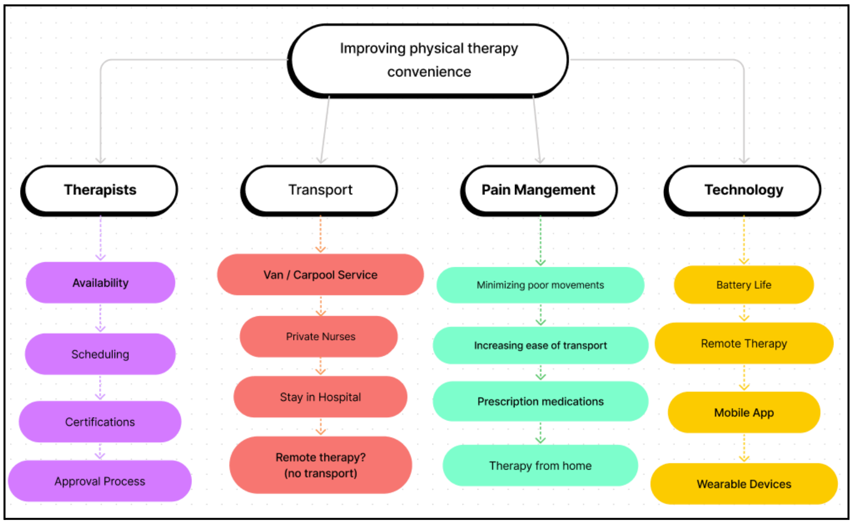


Fig. 16. A diagram of a mind map used to explore related topics and potential solutions to the proposed need statement.

Next, a number of ideas were proposed to solve the need including a scheduling app for physical therapy patients, an automated shuttle service to transport patients to their healthcare professionals and vice versa, a wearable exoskeleton device to guide patient movements, an online repository of exercise definitions and guides, and a motion capture feedback solution.

Brainstorming

The following are the contributions of each team member during a round-robin brainstorming session employed to get a wide variety of ideas to fulfill the proposed need statement. Throughout the brainstorming process each team member was asked to deliver three total ideas within a limited time frame to ensure each member contributed their unique perspective to the problem and encourage creative thinking unimpeded from criticism and feasibility. With these ideas as a starting point further details were explored and developed with a higher degree of scrutiny.

Diya

1. Creating a remote device to help the patient record their session and use AI to correct them and give them feedback
2. Using motion capture bands to record the movements and use an app to check the movements and correct them
3. Create a organized transportation system to mitigate the commute to therapy offices and make it easier for patients

Anthony

1. Create a standalone smartphone app that uses the smartphone's cameras and computer vision to determine the user's motion and provides feedback at the end of the exercise.
2. Create a standalone smartphone app that allows patients to easily schedule visits from physical therapists who travel to their location.
3. Create an at-home smart training gym that uses adjustable "arms" (like those used for computer screens) to adjust pull cables to the correct angle for physical therapy exercises and records the user's applied force and angle.

Kenny

1. Create a smartphone app that is easy to use that will help physical therapy patients plan certain exercises and therapists can approve of them for control.
2. Create exercise bands that will help physical therapy patients move in a more controlled and limited range.
3. Create motion sensors that will collect data from patients' exercises and relay that data to physical therapists.

Davin

1. Hospital partners with nearby hotels/apartment complexes within walking distance so that doctors can commute to the patients
2. Doctors create a library of instructional physical therapy exercises for patients to follow
3. Have doctors travel and tend to patients in a certain location/city (like PT travel nurses).

Najel

1. Creating a simple smartphone application, that tracks patients' physical therapy with a device that sends data to doctors for analysis.
2. Using motion detection, to indicate if the patient is correctly performing the required physical therapy treatment.
3. Setting up virtual physical therapy sessions using armbands, with a physical therapist on the virtual call (Over Zoom, or secured web cam connection)

Ruben

1. Using automated resistance bands to adjust a patient's limbs in the correct direction.
2. Create an automated shuttle service that brings patients to and from the hospital.
3. Create an exoskeleton that can assist and adjust the patient's body in the required range of motion.

Peter

1. Create a dedicated bus to help transfer patients from location to location.
2. Bring the doctor to the patient so the patient does not need to travel.
3. Use telehealth and a certified nurse to assist with patient visits and act as the doctor for simple routine checkups.

Ryan

1. Determine a new system between patient and doctor such as consistent video calls
2. Patients can use specifically developed technology to track and monitor their improvement/progress. Similar idea as HRM's, but with more functionality
3. Can have an app/system like Doordash, but with doctors/physical therapists

Decision Tables

Design 1: Wearable "smart" devices to capture motion data using IMUs, process movements on a central device, and provide vibration feedback. The companion app connects via Bluetooth and saves data and provides detailed feedback.

Design 2: A computer vision system using a smartphone's camera and CV algorithms to record the patient's motions and provides feedback after the exercise is done through the smartphone interface.

Design 3: A smartphone application to facilitate the scheduling and visiting of physical therapists to the patient's location and provide a simple interface for therapists to present information and documents to their clients.

Design Objective	Design1	Design2	Design3
Inexpensive	Mostly	Very	No
Battery life	Low	N/A	N/A
80% accuracy	High	Low	Low
Easy setup	Very	Very	No
Minimal supervision	High	Very	Low
Feedback	Real-time	Post facto	Real-time

Table 1: Basic Ranking

Design Objective	Units	Design1	Design2	Design3
Inexpensive	Dollars	\$50-\$100	\$0	\$300+

Battery life	Sessions	3	20	N/A
80% accuracy	Millimeters	< 30 mm	< 50 mm	N/A
Easy setup	% Trials succeeded	95%	95%	99%
Minimal supervision	% Time supervised	< 10%	< 10%	100%
Feedback	Feedback Delay (ms)	< 10	> 60000	< 30000

Table 2: Quantified Ranking

Design Objective	**Units**	**1**	**2**	**3**
Inexpensive	Dollars	6	10	1
Battery life	Sessions	4	10	10
80% accuracy	Millimeters	8	6	10
Easy setup	% Trials succeeded	9	9	10
Minimal supervision	% Time supervised	9	9	0
Feedback	ms	10	1	2
Total		46	45	33

Table 3: Evaluation Scales

Design Objective	**Units**	**Weights**	**Design1**	**Design2**	**Design3**
Inexpensive	Dollars	35	2.1	3.5	0.35
Battery life	Sessions	5	0.2	0.5	0.5
80% accuracy	Millimeters	5	0.4	0.3	0.5
Easy setup	% Trials succeeded	15	1.35	1.35	1.5
Minimal supervision	% Time supervised	20	1.8	1.8	0.0
Feedback	ms	20	2.0	0.2	0.4
Total		100	7.85	7.65	3.25

Table 4: Weighted Evaluation

Appendix 2 - Planning

Basic Plan / Gantt Chart

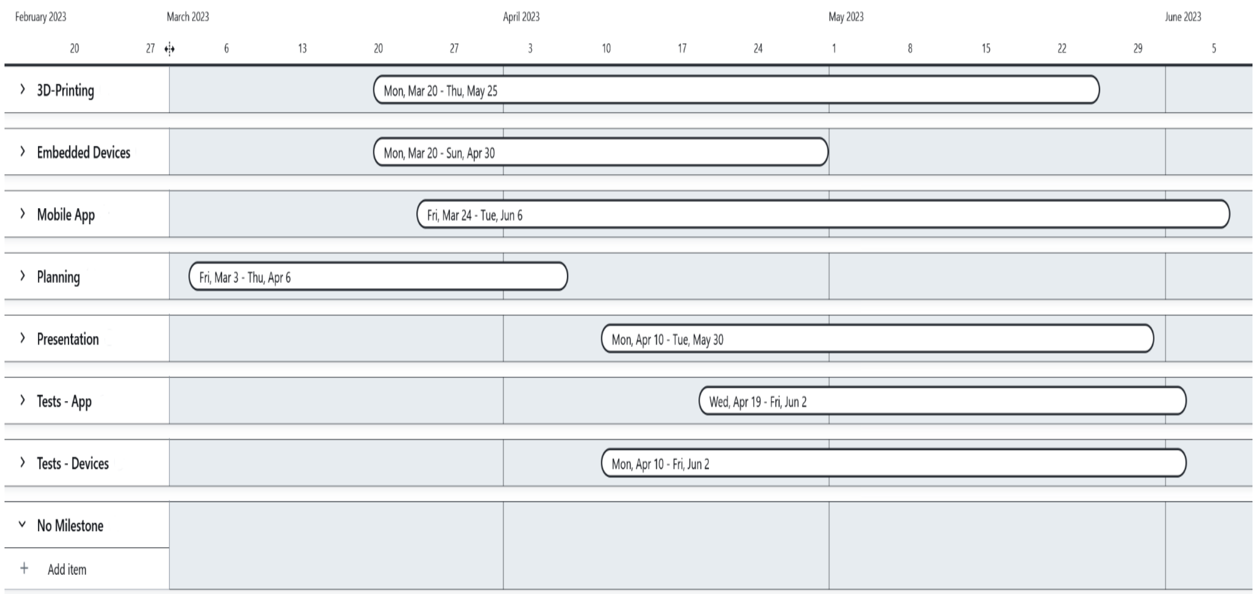


Fig. 17. Gantt chart

Critical Path Method

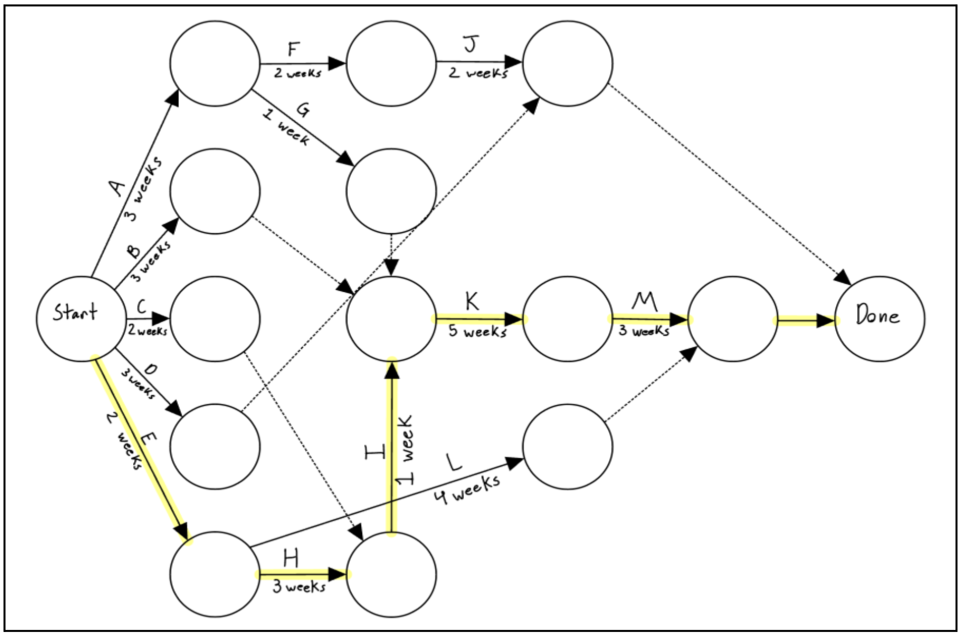


Fig. 18. Critical path method

Division of Labor (Prototyping)

When it comes to the division of labor, we had three sub teams in our main team. Each team had a dedicated portion of the project assigned to them. Our three sub teams were the embedded team, the app team, and the 3D printing team. The embedded team would work on getting all the hardware to work. Tasks included making sure the hardware all connects to each other, making sure the accelerometers work properly, and making sure that everything is soldered properly. The app team's responsibilities was to ensure that the communication between the app and the hardware works properly. Specifically, they were responsible for scanning/establishing a bluetooth connection with the correct device, translating/sending correct data, and reading certain characteristics when supposed to. Other tasks the app team were responsible for include creating an app that is easy to navigate, aesthetic, and could track users' exercise progress. The 3D printing team was responsible for making the aesthetic prototype and printing and assembling all the physical components for the functional prototype. The 3D printing team is also responsible for making design decisions for each component of the prototype. Those design decisions included the type of materials used for the bands and the sizing of each strap. It was the work of the 3D Overall, every team member was responsible for communicating with all the other team members to make sure that everyone is up to date on the latest changes to the Mocopy prototype.

Collaboration Methods

For team collaboration we used GitHub and Discord. We decided to use GitHub for multiple reasons. GitHub provides version control allowing team members to track changes, manage branches, and merge code seamlessly. Each team member can work on different changes and different branches and then merge all changes at the end. GitHub also allows for code reviews. Code reviews allow other team members to see the changes that were made and approve or decline them or make suggestions to the changes. GitHub also allows issue tracking. The issue tracker is great for tracking large bugs, tasks, and features. Team members can create, assign, and prioritize issues, providing visibility into the project process and ensuring tasks are properly managed.

We decided to utilize Discord for team communication. Discord offers a variety of very useful features for team communication. Discord offers real-time communication with voice, video, and text communication channels allowing teammates to collaborate, discuss and share ideas, updates, and issues promptly. Discord also allows for dedicated channels for different topics. This allows our team to have separate channels for each subteam (embedded, mobile app, and 3D printing). This organization allows our team to keep discussions focused and prevents the information from getting lost in a sea of messages. Discord's voice and video meetings allow for our team to conduct meetings and discuss important issues even from remote locations. This allows for team cohesion even when members are not all located in the same place. Discord also allows for file and link sharing. This allows team members to exchange sources, code snippets, and external files that don't need to be uploaded to the GitHub repository. Discord also allows integration with Github using webhooks. These integrations can provide automated notifications, reminders, and updates, keeping the team informed about code changes, issues, or other important events.

Appendix 3 - Test Plan & Results

Sensors Tests:

Idle Drift Test:

Purpose:

Sensor drift is generally viewed as inevitable when dealing with IMUs, however, the magnitude of drift varies according to the quality of the device and a number of software techniques to stabilize the values. While not moving, an IMU's orientation should not drift more than 5 degrees within 10 minutes.

Procedure:

1. Connect the IMU device to the microcontroller over I2C and power rails
2. Calibrate the device accordingly
3. Keep the IMU device still for 10 minutes while recording orientation data to the microcontroller

Results:

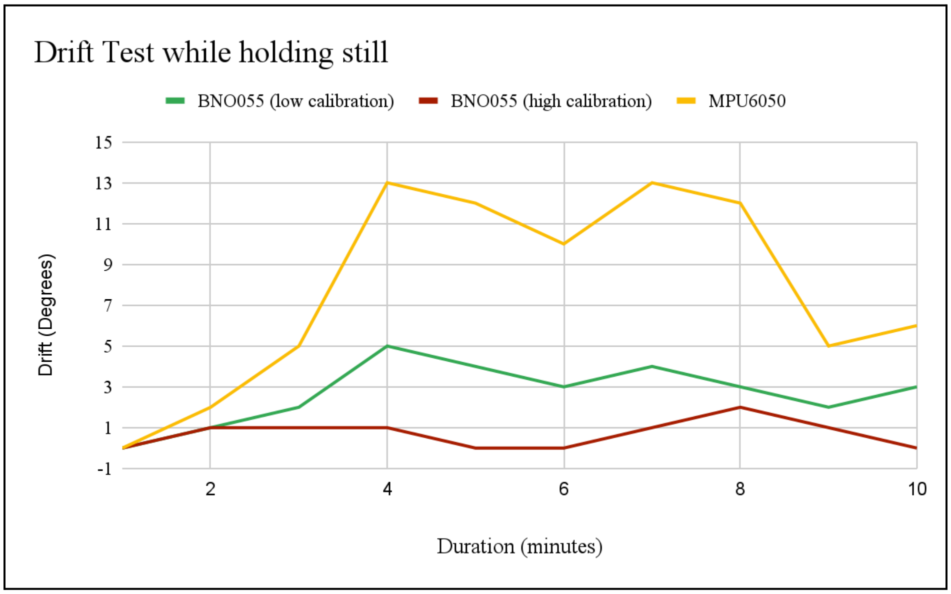


Fig.19. Drift test while holding still on the BNO055 with low and high levels of calibration and the MPU6050 over 5 minutes

Analysis:

Upon recording the drift while holding the sensor still, a significant difference in the drift magnitude between each of the devices was noted. As shown in the figure above, the MPU6050 exhibited a high amount of drift, with the maximum value reaching 13 degrees and the minimum is 2 degrees. On the other hand, the BNO055 had better drift performance compared to the previous IMU. Calibrating it to the highest tolerance made it experience much less drift, however, the BNO055's drift performance was found to be tolerable at both the full and reduced calibration levels.

Exercise Drift Test:

Purpose:

Sensor drift is generally viewed as inevitable when dealing with IMUs, however, the magnitude of drift varies according to the quality of the device and a number of software techniques to stabilize the values. During an exercise, in this case, a simple bicep curl, an IMU's orientation should not drift more than 8 degrees when returned to its starting position within 10 minutes. I.e. changing the sensor's orientation and then returning it back to its original orientation should give the same reading, however much this orientation is off by is the drift.

Procedure:

1. Connect the IMU device to the microcontroller over I2C and power rails
2. Calibrate the device accordingly
3. Perform 10 reps of a simple bicep curl with the IMU device attached to the user's arm
4. Record the total drift of the IMU each time the IMU is returned to its original position for the exercise

Results:

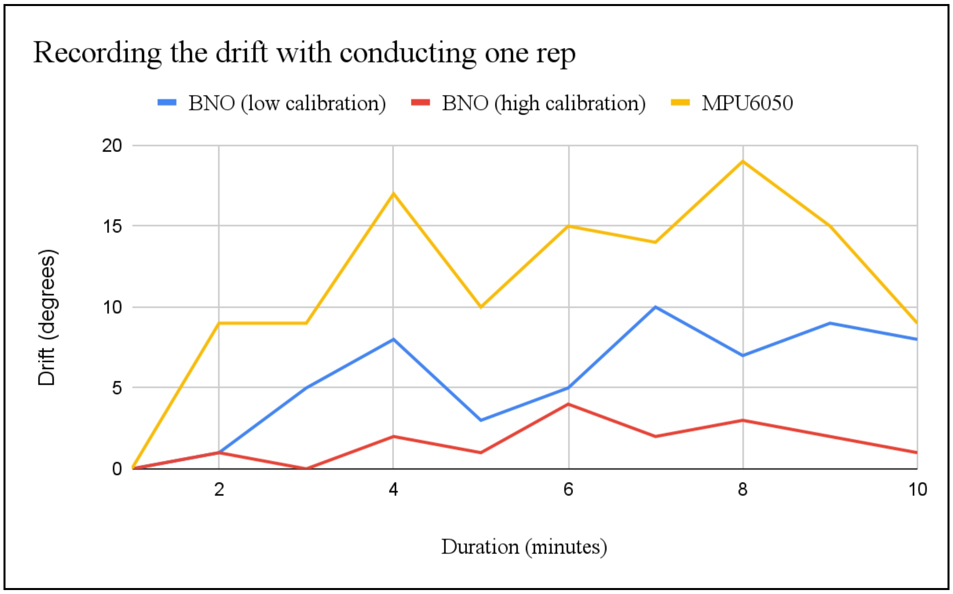


Fig. 20. Drift test while doing one rep on the BNO055 with low and high levels of calibration and the MPU6050 over 10 reps

Analysis:

Upon recording the drift while conducting 1 rep, a significant difference in the drift magnitude between each of the devices was noted. As shown in the figure above, the MPU6050 exhibited a high amount of drift, with the maximum value reaching 19 degrees and the minimum is 0 degrees. On the other hand, the BNO055 had better drift performance compared to the previous IMU. Calibrating it to the highest tolerance made it experience much less drift, however, the BNO055's drift performance was found to be tolerable at both the full and reduced calibration levels. This makes sense because when the devices were sitting still the drift level was pretty much similar but it increased as there was a lot of movement.

Drift vs Sample Rate Test:

Purpose:

According to the BNO055's datasheet new orientation data can be generated by the device at a variety of sample rates according to the highest bandwidth sensor, the accelerometer. In theory, sampling at a higher rate allows for a better estimation of orientation and thus reduced drift while holding the device still. For each setting, the reduced calibration levels were set and the devices were held still for 10 minutes to record total drift.

Procedure:

1. Program the IMU device with the corresponding sample rate
2. Connect each IMU device to the microcontroller over I2C and power rails
3. Calibrate the device accordingly
4. Keep the IMU device still for 10 minutes while recording orientation data to the microcontroller

Results:

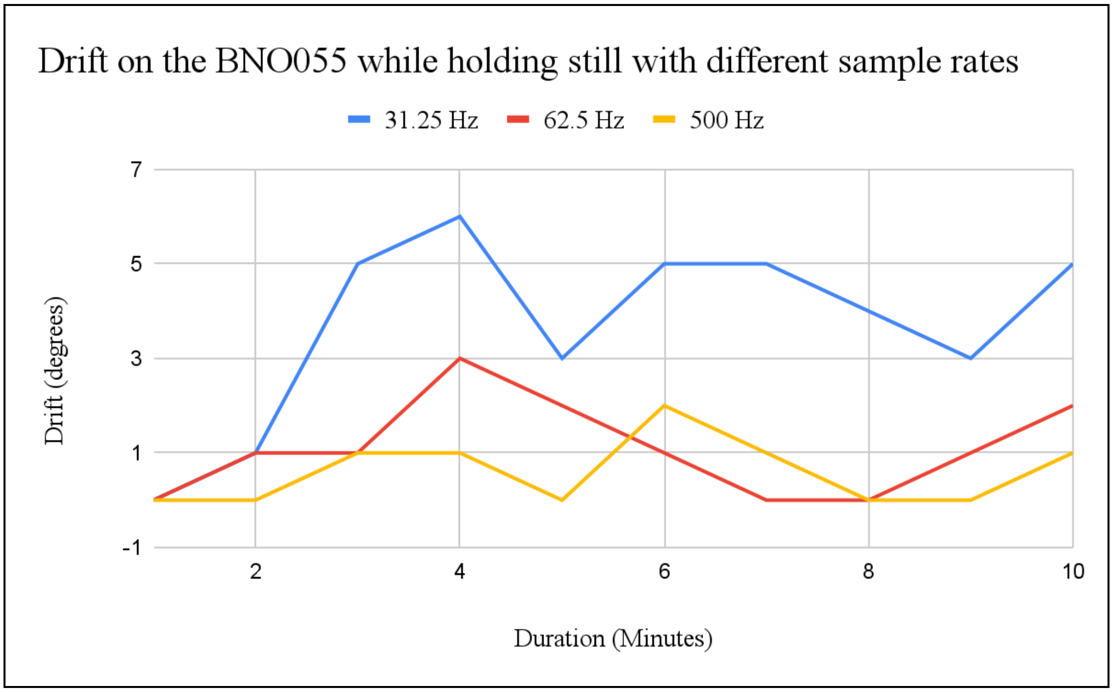


Fig. 21. Drift on the BNO055 while holding still and different sampling rates

Analysis:

As we can see from this test, at lower sampling rates, the drift is much more severe over the span of several minutes. At much higher sampling rates, the drift is a lot more tolerable over time. At some sampling rates, the difference in drift over time isn't very noticeable from a higher sampling rate, as we can see from the results above. At 62.5 Hz and 500 Hz, the drift for both of them isn't too different from each other. We chose to stick with 62.5 Hz, as there are diminishing results by increasing the sampling rate.

Sensor Fusion Modes Test:

Purpose:

The BNO055 includes a number of built-in “fusion modes” which combine raw data from multiple independent sensors to generate processed absolute orientation data. These modes include the IMU, M4G, and NDOF modes. The IMU fusion mode only uses the accelerometer and gyroscope sensors to determine orientation data strictly relative to the power on position. The M4G or the ‘magnetometer for gyroscope’, uses the accelerometer and magnetometer with the magnetometer taking the place of the gyroscope and generates absolute orientation data. Finally, the NDOF mode uses the accelerometer, gyroscope, and magnetometer together to generate absolute orientation data with the magnetometer providing high fidelity data for the heading axis. To evaluate the accuracy of each operation mode the same BNO055 device was programmed into each mode, calibrated with the reduced calibration level, and then held still for 10 minutes to record the drift.

Procedure:

1. Program the IMU device with the corresponding sensor fusion mode
2. Connect each IMU device to the microcontroller over I2C and power rails
3. Calibrate the device accordingly
4. Keep the IMU device still for 10 minutes while recording orientation data to the microcontroller

Results:

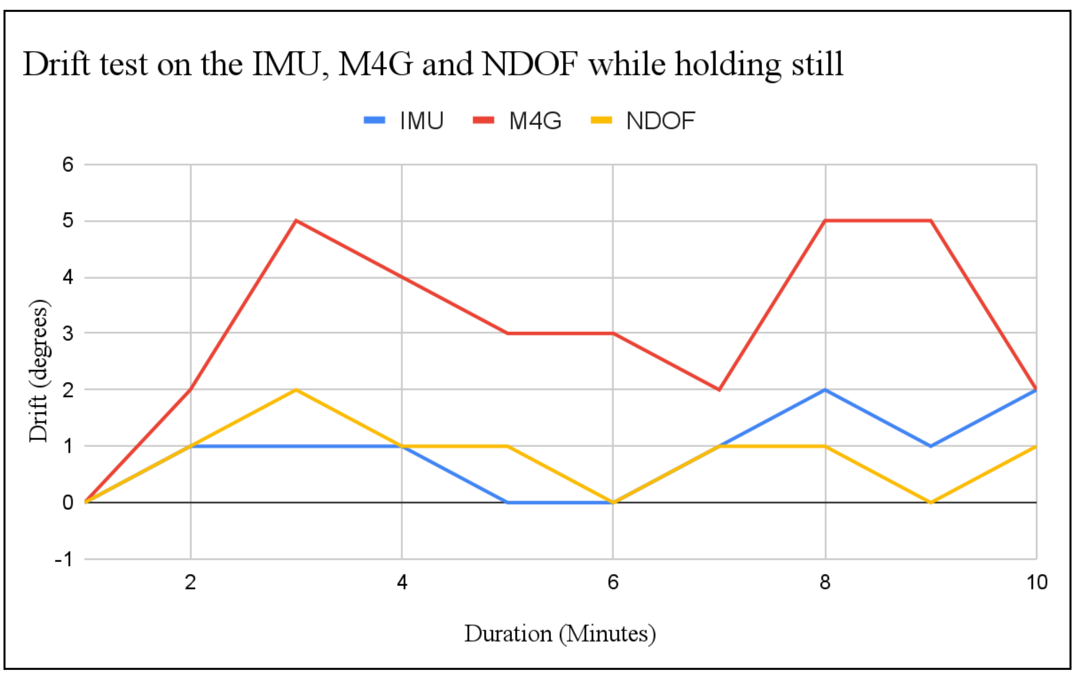


Fig. 22. Drift test while holding still with IMU, M4G and NDOF

Analysis:

As seen in Figure 22 above, the M4G / "magnetometer for gyroscope" experienced considerably more drift in the idle test than the other two operation modes, likely due to the abundance of electrical equipment and metal infrastructure surrounding the test environment. It is suspected that the NDOF mode, which incorporates all three sensors, does not achieve better results than the IMU mode, which only uses two sensors, for the same reason as the magnetometer likely introduces considerable error in such an unreliable environment. It is believed that perhaps changing the operation mode from NDOF to IMU would help increase the reliability of the sensors in a variety of environments going forward. This shift is plausible as the exercise definitions do not require absolute positioning relative to magnetic north which the NDOF mode incorporates but is left unused.

Calibration Time Test:

Purpose:

Each Mocopy device should be able to calibrate within a reasonable time, in this case, no more than 60 seconds.

Procedure:

1. Perform calibration on bands with a higher tolerance on the IMU
2. Record the time taken to fully calibrate joint and external
3. Perform calibration on bands with a lower tolerance on the IMU
4. Record the time taken to fully calibrate joint and external
5. Repeat steps 1-4 15 times total

Results:

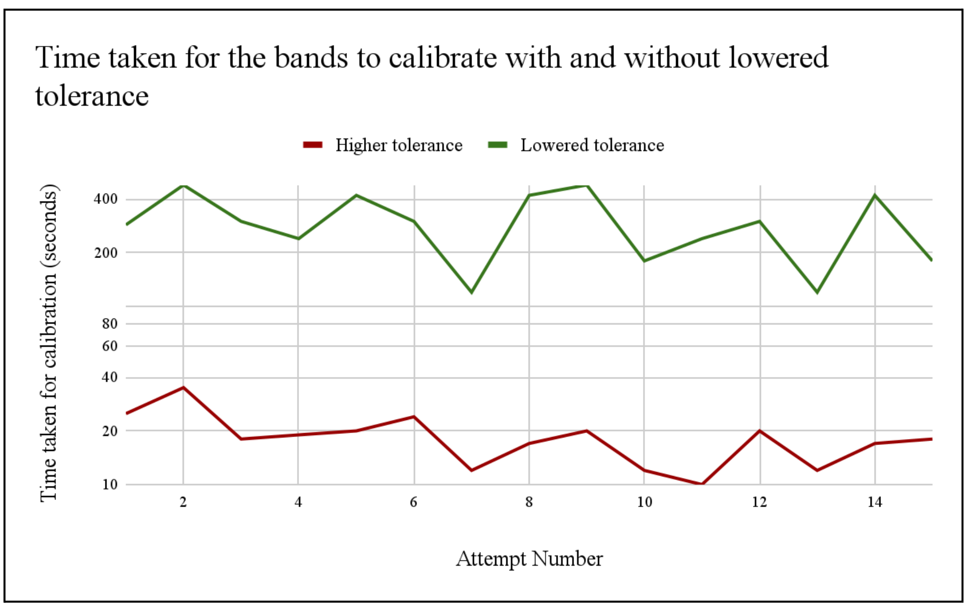


Fig. 23. Time taken for both bands to calibrate before and after lowering the tolerance

Analysis:

Based on the results of the testing, it is evident that lowering the tolerance for the IMU's calibration increases the time taken for calibration. The results show that on average the amount of seconds taken for calibration when there is a lower tolerance is around 250. When the tolerance is increased the seconds of calibration decrease to around 25 seconds. This shows that increasing the tolerance decreases the time for calibration taken by ten times. This is a significant difference making it a clear choice to increase the tolerance in our project as no user would want to spend several minutes calibrating.

Limited Motion Calibration Test:

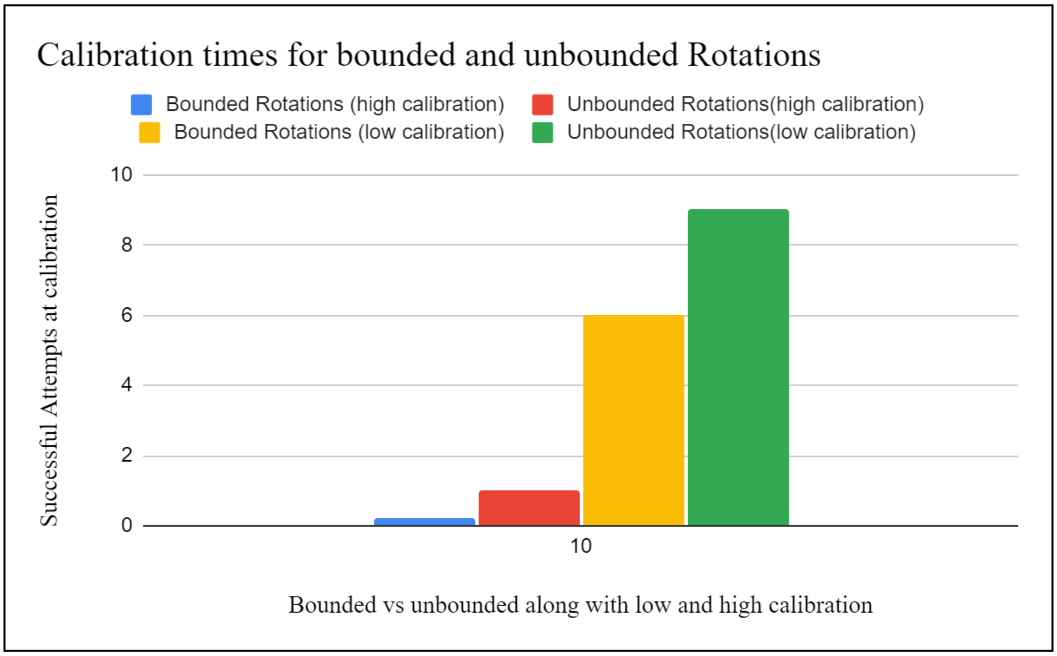
Purpose:

As Mocopy needs to be suitable to a wide variety of patients with unique needs the calibration process should require little to no bodily motion from the patient while worn as they might have a limited range of motion and be unable to move in the necessary patterns. This test is designed to check whether the required calibration level from the devices can be reached while rotating the device less than 30 degrees on each axis and in less than 4 minutes.

Procedure:

1. Connect the IMU device to the microcontroller via I2C and power.
2. Slowly rotate the IMU device at a variety of angles during calibration. Limit the range of rotations according to the test iteration.
3. Record the time and calibration status of the device throughout.
4. Note the total time required to calibrate the device or failure if calibration took longer than 4 minutes.

Results:



<u>Fig. 24. Successful attempts for unbound and bound rotations with low and high calibration for both</u>

Analysis:

With the bound vs unbound rotations, the results show that if we have to calibrate the IMU at maximum calibration values, the user is unable to successfully calibrate as it takes too long. The results also show that the calibration is slightly more successful if the user is unbound by the rotations they make, as the freedom of moving makes it easier for a wider variety of users.

Exercise Tests:

Grace Angle Test:

Purpose:

Each exercise is defined in terms of the angles between joints, however, it is unrealistic to expect every rep to reach the exact angle required to complete the exercise due to measurement jitter and floating point rounding error. Due to this, a “grace angle” is defined as the tolerance between the system’s current angle measurement and the correct angle measurement which will still trigger a “keyframe hit” command. The grace angle should ensure that patients can still reliably reach each keyframe position without being so lenient that wildly incorrect positions register as a success. To this end, a few grace angle values were tested using the simple bicep curl exercise to find a balance between these two extremes.

Procedure:

1. Program the Mocopy device with a 5-degree grace angle.

2. Power on and calibrate the device.
3. Attach the device to the patient.
4. Ask patients to complete a simple bicep curl exercise.
5. Measure the number of successful attempts.
6. Repeat steps 1 through 3 using a 15-degree grace angle and then 25 degrees.

Results:

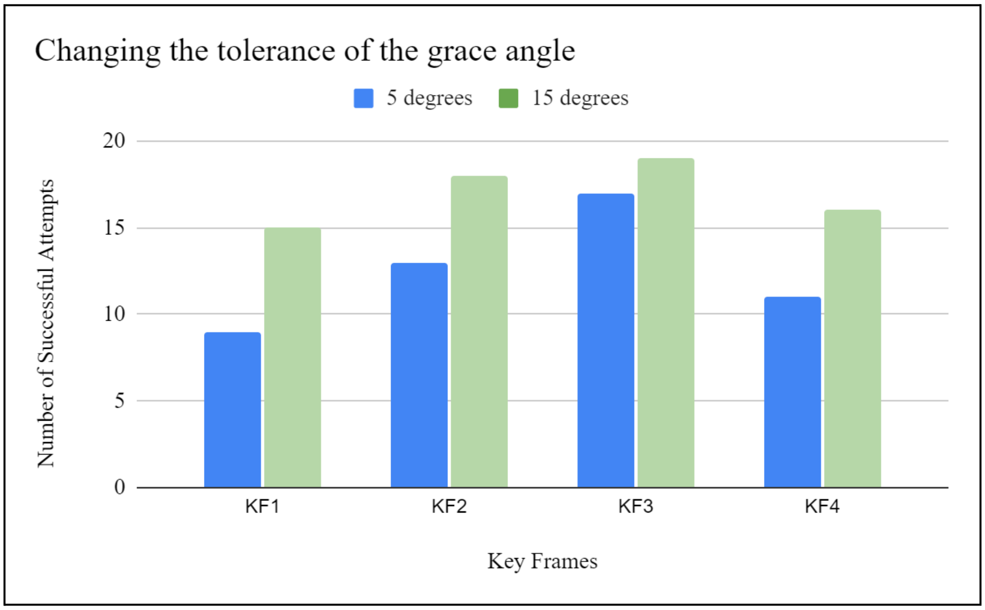


Fig. 25. KeyFrames hit before and after tolerancing the grace angle

Analysis:

From our tests above, with our original value of 5 degrees, it was not as simple to hit the keyframes with such a limited grace angle. By increasing the grace angle to 15 degrees, we were able to hit the keyframes with much more succession. We came to the conclusion that 15 degrees were enough that the user would be able to hit keyframes, but not hit them while not being close to them. This would give us the most accurate results as it wouldn't be too tolerant but it wouldn't make the user unable to hit keyframes.

Keyframe Amount Test:

Purpose:

The exercises have to be simple enough that users are able to successfully complete them. By having a higher number of keyframes, we are able to split up the exercise into more accurate movements. By having more keyframes, we can define an exercise in a more strict and precise manner.

Procedure:

1. Set up the Mocopy device with a set number of keyframes

2. Ask the patient to perform an exercise and hit all the keyframes
3. Measure the total number of successful attempts
4. Repeat steps 1 through 3 with a varying number of keyframes

Results:

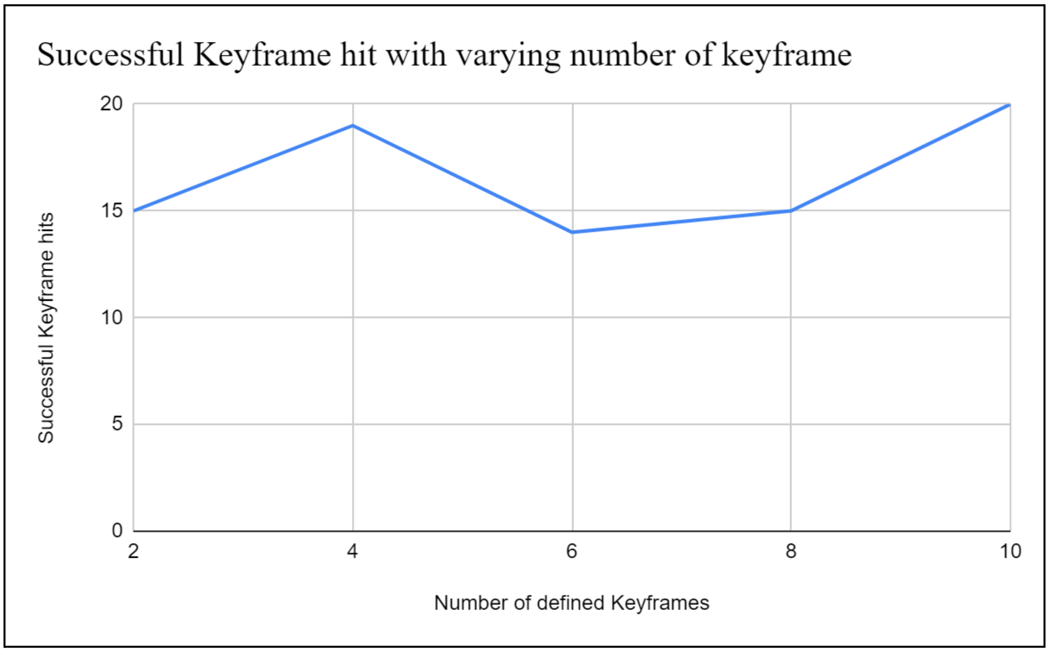


Fig. 26. Successful Keyframe being hit with a varying number of defined keyframes

Analysis:

As the results show, there is a significant difference between having a lower number of keyframes and a higher number. There is also a notable cutoff for having too many keyframes, as was tested. By having too many keyframes, the user is able to hit every single one and doesn't exactly give good feedback on whether or not the exercise was successful or not. By tuning this down to 4 keyframes, it was successful enough that an exercise could be done accurately and precisely.

Connectivity Tests:

BLE Latency Test:

Purpose:

As the Mocopy system strives to guarantee real-time feedback, minimizing latency between the wearable bands, their sensors, and the smartphone device is of high importance. To test the latency of the BLE wireless communication a series of synthetic payloads containing timestamp data were sent from the smartphone device to the central band, then to the joint band, then the external band before being sent back up the hierarchy to measure their total latency and give an expectation for the maximum latency for the complete system.

Procedure:

1. Configure the mobile app to send data to the central device of the Mocopy prototype at a relatively lax interval (50ms).
2. Program each wearable device to receive data from its neighboring device and forward it to the next device going from smartphone to central to joint to external and back.
3. Record the time taken for a single batch of data to return to the smartphone device according to the timestamp value.
4. Run the synthetic workload for 1000 iterations and record the latency for reach.

Results:

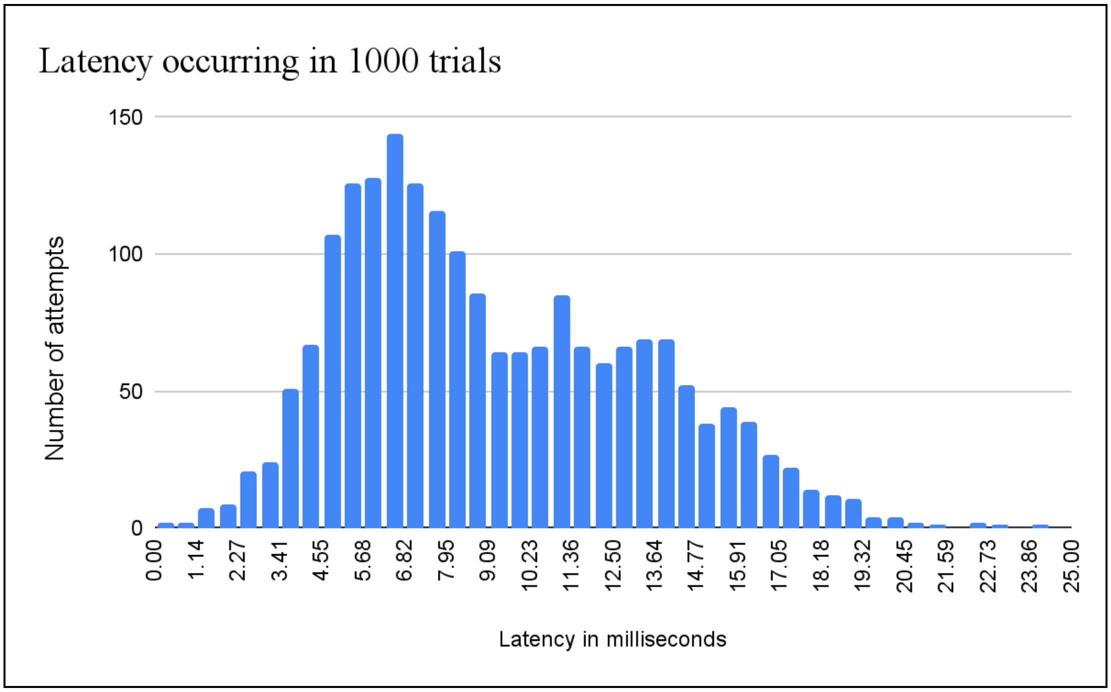


Fig. 28. Latency occurring over 1000 attempts

Analysis:

As shown in the above plot, the latency experienced by each packet generally tended around 6 to 8 milliseconds with considerable spread into ranges closer to 13 milliseconds, however, given the [average human reaction time] (on the order of hundreds of milliseconds) this range is more than sufficient to be considered real-time for our purposes and empower patients to complete even fast-paced exercises with a high degree of accuracy and dependable feedback.

BLE Duration Test:

Purpose:

The Bluetooth connection must be able to last at least an entire therapy session. The device must be able to stay connected to itself and the app. If the device were to lose connection, it would make it difficult for the patient to restart or continue the exercise.

Procedure:

1. Set up the Mocopy device
2. Run for 3 hours
3. Report any errors in connection
4. Repeat steps 1 through 3 10 times

Results:

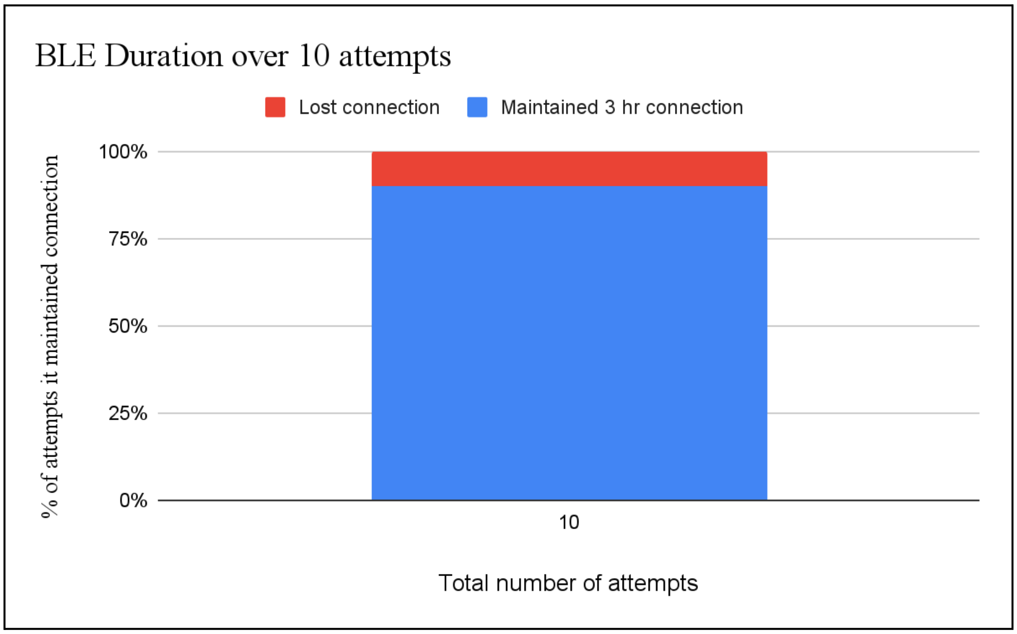


Fig.29. Maintaining BLE connection for 10 attempts

Analysis:

Based on the results, it is evident that the Mocopy central device can maintain a stable BLE connection with the smartphone 90% of the time. Even though the device did lose connection once throughout the tests, for it to work 9 out of 10 times is reliable. Moreover, 3 hours is a large amount of time for a user to exercise so for a device to lose connection just once in that amount of time is acceptable and all the user must do is navigate back to the connection screen.

BLE Bandwidth Test:

Although the Mocopy motion capture system uses relatively little of BLE's total bandwidth potential, testing the capabilities of BLE's data transfer helps gauge the protocol's capabilities for future-proofing the device in case new features are to be added in the future. To test BLE's bandwidth, a number of synthetic payloads were constructed holding timestamp data and transferred between the smartphone device and the series of wearable devices.

Procedure:

1. Configure the mobile app to send data to the central device of the Mocopy prototype at the shortest interval designated by the BLE specification (12.5ms).
2. Program each wearable device to receive data from its neighboring device and forward it to the next device going from smartphone to central to joint to external and back.
3. Record each incoming data packet and determine if any packets were lost or otherwise untransmitted due to congestion.
4. Run the synthetic workload for 1000 iterations and record the results for each.

Results:

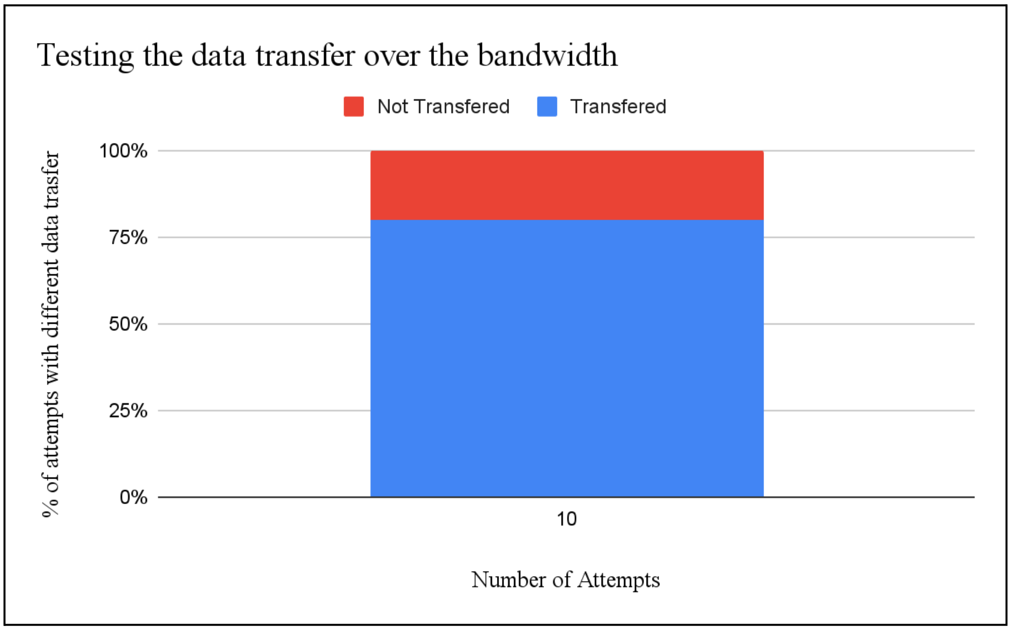


Fig.30. Conducting a bandwidth test

Analysis:

The results from the bandwidth test show that the device can maintain a high transmission rate for 10 minutes. The 92% rate of success is more than enough to prove that the daisy chain infrastructure is a viable option for the way of communicating data throughout the devices. Moreover, even though there are some failures for this test, high data throughput is not something that is very important as there is not much data being transmitted from device to device.

Power Tests:

Power Stress Test:

Purpose:

The power stress test will provide insight to how much power the Mocopy devices draw when under maximum load. This insight will be used to determine the power rating of the battery that would be suitable for our devices. Ideally, the devices should last at least 2-3 hours under intense load to guarantee power for consumers for the entirety of one session.

Procedure:

1. Configure the Mocopy devices to run all motors and wireless communication continuously.
2. Place a digital multimeter in parallel with the incoming USB power rails to each Mocopy device and set the device to measure current.
3. Record the displayed current and perform algebra to undo the current division to yield the expected current through the active components.
4. Repeat steps 1 through 3 nine times. Multiply the calculated current by 3.3V to determine the total power draw.

Results:

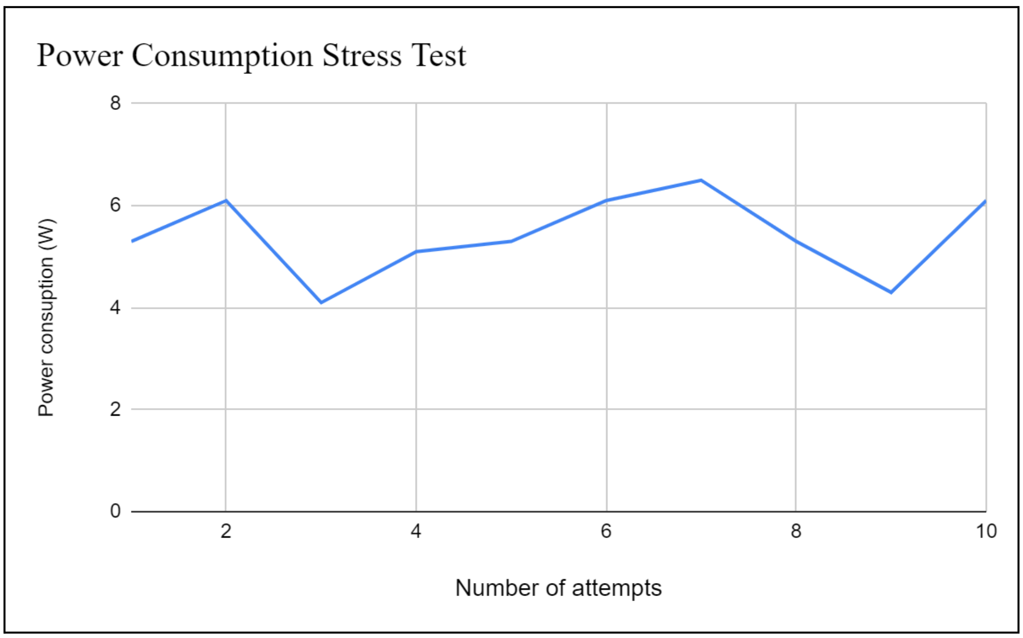


Fig.30. Power consumption stress test

Analysis:

Based on the test above, the average power consumption when the Mocopy device is under stress is around 5 W. The highest power draw we have experienced is 7W. This power is more than sufficient enough for the Mocopy device when under maximum load(exercising). The power draw also is sufficient enough for the Mocopy devices to last for a full 1-2 hour workout.

Idle Power Test:

Purpose:

If the user were to leave their device sitting idly by, the power draw on it should be minimal. If it is too high, it will be difficult for the user as they would have to constantly charge the device at all times. The device should be able to draw as minimal power as possible when not doing anything other than occasionally scanning for or advertising BLE services.

Procedure:

1. Configure the Mocopy devices to remain in idle mode.
2. Place a digital multimeter in parallel with the incoming USB power rails to each Mocopy device and set the device to measure current.
3. Record the displayed current and perform algebra to undo the current division to yield the expected current through the active components.
4. Repeat steps 1 through 3 nine times. Multiply the calculated current by 3.3V to determine the total power draw.

Results:

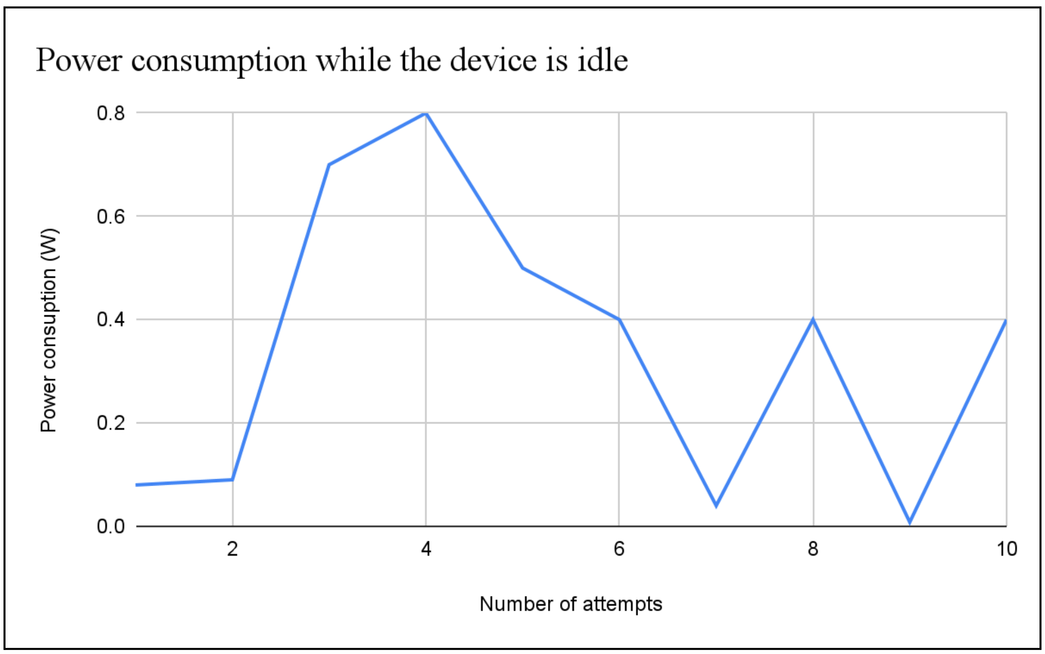


Fig.31. Power consumption stress while the device is idle

Analysis:

From the results above, it shows that when the device is idle, it draws very minimal power, ranging from 0.08-0.5W. This means that it is successful in reducing power consumption, as 0.08-0.5W is negligible and will allow users to let the device sit idly.

Hardware Tests:

Vibration Intensity Test:

Purpose:

In order to ensure the real-time vibration feedback could be successfully interpreted by the end user the prototype devices were connected to a volunteer and powered with a PWM signal of varying duty cycle and their response was recorded as a number from 0 to 3 with 0 representing no vibration felt and 3 being strong vibration felt. The test was performed twice, once with the device directly on the volunteer's skin and again with the device over the volunteer's t-shirt sleeve.

Procedure:

1. Attach a single Mocopy device to the patient's arm with the vibration motors facing inward.
2. Configure the vibration feedback to occur on a single motor at a time and send the PWM signal to drive the motor.
3. Ask the volunteer for their rating as discussed above.
4. Repeat steps 1 through 3 with the volunteer's t-shirt sleeve rolled down.

Results:

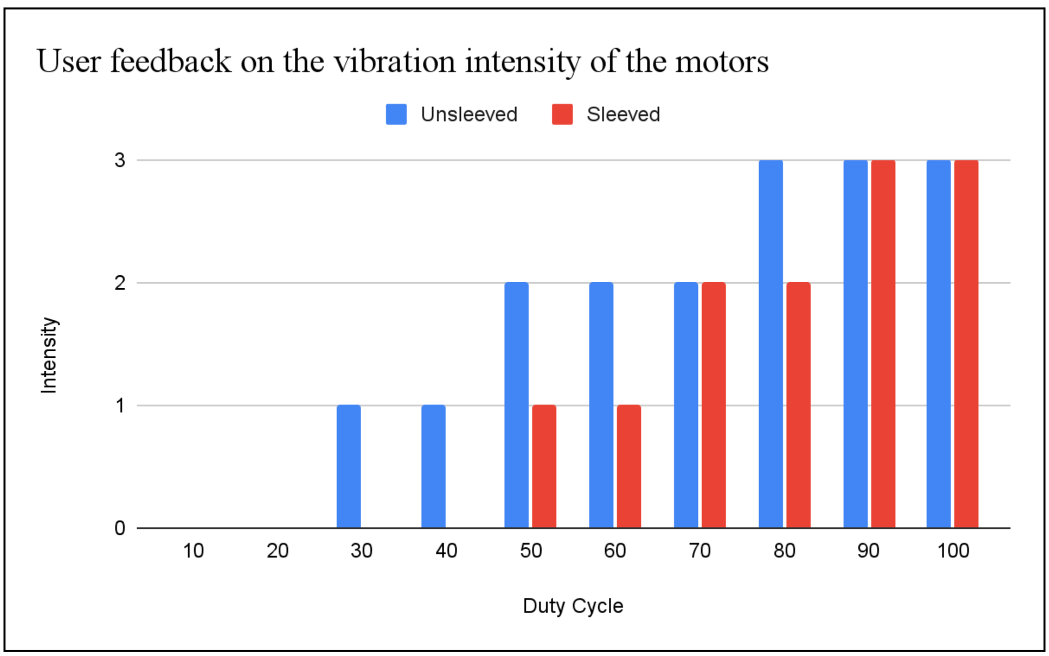


Fig.32. User feedback based on the duty cycle of the vibration motors

Analysis:

As seen in the chart above, the volunteer was unable to detect any vibration for the low duty cycle PWM signals indicating that the motors may not have had sufficient power to overcome inertia. However, as the duty cycle, and thus power delivery, increased the patient was able to discern a greater range of vibration intensities. When the device was placed on top of the volunteer's sleeve, the volunteer found the vibration intensity considerably reduced which may indicate a concern for patients being able to wear heavier clothing while still making use of this system. Overall, however, the volunteer's responses were seen as acceptable as most patients will be expected to wear workout clothing or similar during their exercises so as to not reduce their mobility through constrained attire.

Appendix 5 - Reviews

Anthony A. Russo

At the beginning of this capstone project and working on a motion capture system using embedded systems I found myself anxious at the thought of working with so many devices I had never even heard of before. However, through much effort, me and my teammates were able to learn a number of new concepts related to IMU sensor devices and get a better grasp on managing Bluetooth Low Energy. If we had to do this project again, I think some of the major changes I would make are doing more research into our sensors and wireless connectivity options beforehand as we often found ourselves changing major components part ways through the project. In particular, I wish we had spent more time understanding what we needed from our BNO055 IMU device since the Arduino Nano 33 BLE actually includes an embedded gyroscope and accelerometer which would have been sufficient for our use case as we did not need the magnetometer the BNO055 afforded us and could have saved on time and cost. Perhaps more than anything though, I wish we had managed our scheduling and started sooner as we often found ourselves staggering meeting times with team members coming and going during a single meeting as we fought to accommodate everyone's class schedule. Overall, however, I am more than satisfied with the exemplary work we were able to accomplish and all the experience myself and my teammates were able to gain regarding the design process and the technical skills needed to implement a product such as this.

Ruben Movsesyan

This project represented a significant undertaking, requiring a substantial investment of time and effort to ensure its completion and full functionality. At certain times, I dedicated extended periods of work to address specific features that were impeding progress for the entire team. However, the ultimate satisfaction derived from successfully implementing these features was great. Witnessing the integration of new functionalities within our project was truly gratifying, as it allowed for the complete meshing of our different project components. The collaborative engagement of all team members in this project proved immensely advantageous. It fostered effective communication, ensuring proper execution, efficient progress, and consensus on many project-related decisions. If I were to do this again, I would have focused more energy on the project's planning phase. Substantial time was invested in testing project components that were ultimately discarded. By establishing a clear direction earlier in the process, we could have minimized the exploration of alternative approaches and allocated resources better.

Kenny Kim

This project was very challenging as everyone had different schedules and we had a shaky start, but in the end, we were able to get what we needed done. If we had the chance to do this project again, I would personally prioritize this project over some other assignments at some point in time, as I had multiple instances where I would need to be caught up on everything and where everything was going. I also believe that if I had the chance to do this again, I would just invest in better IMUs from the beginning, as the IMUs had a lot of issues in our first quarter and would cause a lot of difficulty in terms of writing the libraries. I think what really worked was when we would all meet up and just work on it together, as I feel that once we all were in the moment and working together on this, we got more done than when we were working separately and slowly.

Davin Muramoto

For the most part I feel satisfied with the end product for this project. In the end, I feel like the product and the Bluetooth network that includes the Arduino and the smartphone went well and sort of according to plan. Everything did not go as expected in this project and that is to be expected especially when designing a complex project. If we were to do this project again there are definitely some things I would do differently. First, on the app side, I feel like react native was a sufficient platform for creating our app but the standard BLE library for it was shaky. During the project, I would constantly lose Bluetooth connection with the central Arduino and would have to quickly reconnect whenever this occurs. Also, data translation was a bit tricky as data for characteristics are read/written in base64 which was hard to decipher when sending and reading data. On the hardware side, I feel like getting better IMUs could solve most of the problems. We noticed that our IMU drifts a lot thus making hitting key frames a lot harder. As a whole, I feel like for the most part, everything went smoothly except for the end when we ran into a bunch of issues. Next time I feel like we should integrate the app and hardware sooner to have more time to troubleshoot.

Diya Saha

This project from the beginning seemed ambitious, but working together with my team members made the whole project possible. I am really satisfied with the end product and how the whole project turned out. Working on this project made me learn a lot of new topics. For example, I learned how to code BLE and construct an app using React Native. I learned how to map different keyframes in a 3D space and navigate them using different devices. I learned about daisy chaining and how different devices communicate with each other using BLE. If I was to do this project all over again, I would document my progress a lot better. Throughout this project we had to make a lot of changes in our design and test different sensors and their behavior, I would document and conduct extensive tests to help navigate the changes and make better decisions. I mainly contributed by helping make the UI elements for this project and conducting tests for different components of our device. Overall I am satisfied with what we were able to accomplish in the given time frame.

Peter Han

This project had many challenges that took multiple attempts to accomplish. From handling the motors to understanding the rigidity of PLA, I approached these challenges with grit. I knew that the first time I would 3D print something it would not be perfect. Each test print gave me an idea of what to expect from the 3D printer we were using. Some prints could not withstand the vibration of the motors so I needed to account for that in my design. Measuring the straps to the correct millimeter thickness took several attempts to ensure that they would slide through the component housing. Not only did those components need to slide onto the strap but having them stay in the correct spot on our limb also took several attempts to fix.

Najel Alarcon

For this project, I felt like having different schedules was one of our biggest challenges, but in the end, we were able to finish our project. Everything was going alright, but I knew that learning React was going to be a challenge and that our smartphone app could've been a lot more UI appealing, but getting the functionality was the biggest priority at the time. Not only doing the BLE but figuring out the keyframes to complete a singular rep was a huge task as well as that took a lot of trial and error to do.