NORTHERN ARIZONA UNIVERSITY

School of Informatics, Computing, and Cyber Systems

To : Krista Branch
From : AmpEd
Date : 12th December 2019
RE : Group Design Review 2

Dear Ms. Branch,

Attached below is the team's group design review 2. In this assignment, we present a report on the current status of our capstone project, laid out such that any of our classmates could read it and clearly understand our project. We present the project and client, followed by previous related literature, and a top-level system overview of our project. We then discuss our early prototypes and results. Finally, we present our current progress, as well as our plans as the team moves into next semester.

Very respectfully,

Team AmpEd

Augmented Mobility Platform Education (AmpEd)
Daniel Beckett, Khaled Khaled, Lauren May, Taylor Yee

Project:

Augmented Powered Mobility

Group Design Review 2

Emails of Team-members:

Kaa325@nau.edu – **Khaled Khaled**
Dbb72@nau.edu – **Daniel Beckett**
Lm933@nau.edu – **Lauren May**
Tey24@nau.edu – **Taylor Yee**

Client & GTA emails:

Kyle.Winfree@nau.edu – **Dr. Winfree**
si229@nau.edu – **GTA Shafkat**

Faculty Advisors:
Client: Krista Branch, PT, PCS, ATP
Sponsor: Dr. Kyle Winfree
Faculty Mentor: Shafkat Islam

TABLE OF CONTENTS

The Client

Krista Branch is a physical therapist, pediatric certified specialist, and assistive technology professional, who works with disabled and special-needs children in the Flagstaff Unified School District (FUSD). She has previously worked with a team of mechanical engineering students at NAU to develop the Augmented Powered Mobility (APM) platform, but currently needs a more robust, adaptable, and updated platform in order to meet the needs of her patients.

The Project

The team's faculty sponsor, Dr. Kyle Winfree, has devoted much of his research to measuring and improving healthcare through wearable technologies. Go Baby Go! (GBG) is an example of one of his projects, and provides the basis for the team's APM project. GBG is a non-profit organization that operates alongside the Cerebral Palsy Foundation (CPF). The GBG program itself started at the University of Delaware (UD), but now operates nationwide, and even has remote sites across the world. Winfree's GBG project is based out of Flagstaff, AZ, and like many other across the U.S., is aimed at providing modified toy car rides to children with disabilities. These specially modified cars are primarily geared towards young children. However, there are still other, older children who have never experienced independent movement, and are currently too big for the GBG cars. This is where the AmpEd team at Northern Arizona University (NAU) comes in.

The APM project's goal is to design and develop electronics and data collection software to meet the needs of two different parties. First, to help the disabled children that Branch works with to experience independent mobility and practice force feedback training. And second, to provide data collection in an easy-to-read manner for the client, so that she can tailor future therapy sessions with her patients to better meet their specific needs.

Custom electric powered wheelchairs (EPWs) for children with disabilities are expensive, and in most cases, the children who need them have never experienced driving a powered wheelchair before. Assistive devices and platforms have been made to address some of these problems, but they are also expensive, and larger in scope than what the client is seeking. The primary objective of the team is to develop a low-cost, adaptable platform that contains additional inputs to address the needs of disabled children who need powered wheelchairs.

Previous Work

Introduction

As a team, we were first individually tasked with a literature review assignment. This assignment was meant to get the team thinking about the different aspects of the project in front of them. Each member found at least eight sources of literature relating to previous work in APM. The focus of this section falls primarily into two areas: prior arts, which refer to what others have physically done in the past to address the problem at hand, and previous standards, which refer to what technical and ethical standards others have followed as good engineering practices. First, our research discusses some previous work done on powered mobility platforms, as well as good practices and standards that researchers have followed. Next, an overview and analysis of some GBG case studies are presented. These are followed by research done in EPWs. Finally, specific components that the team plans to implement are investigated and analyzed for viability.

Powered Mobility Platforms and Good Practices

The main goal of our team is to create a device that aids in the training of adolescents as they become eligible to begin using a powered mobility vehicle rather than a regular standard wheelchair. The idea is to allow the user to attach components to their existing unpowered wheelchair and have the ability to use it similarly to a regular powered vehicle [1]. The chair must also collect data to aid medical professionals in understanding the effect the chair has on the development of the individual and aid in the training of the user. The vehicle shoulds have different settings to assess different needs each child may have including settings that steer the user away from incoming obstacles and steer them into obstacle to teach them how to operate the vehicle on their own. An existing powered mobility platform is provided and the control, motor driver, sensor, and graphical user interface (GUI) models need to be implemented. Authors Rama Kallam and Harish Sharma explain the definitions of different types of powered mobility vehicles, their functions and different issues that arise during the development of each vehicle. The problem of any powered mobility device is to help any disabled user live more unaccompanied and autonomously by upgrading a typical unpowered wheelchair. They talk about how to upgrade a typical mobility vehicle into what they call a “Smart Chair,” which is a type of mobility platform that has autonomous capabilities that make driving operations easier for the user [2]. The paper breaks down each aspect of the Smart Chair and any requirements or calculations associated with it. Since there are a large range of powered mobility devices, there are no industry standards for “smart” powered mobility platforms, however, the overall consensus for the definition of a “smart” device is that the device should include operations that occur autonomously. Even though it is not our goal to make a smart device, many of the modules used in the chair must also be present in our training device to be fully functional. The final product needs to have the following modules: A motor driver, controller unit, joystick, sensors and a GUI (Graphical User Interface) are all necessary to create a functional powered vehicle as was the result in the paper.

This paper by authors Joshua Usoro, Lisa Kenyon, John Farris and Samhita Rhodes explore how independent mobility affects children's development by using mutual information and EEG scanning. The goal of the authors was to better understand the changes that occur when children with severe mobility issues have the opportunity to independently explore their environment. They found a positive increase in the amount of information transferred in the child's brain when using a powered mobility vehicle compared to times when they did not have access. Although in the time periods of not interaction with a power mobility device there was a positive increase in the frontal and parietal lobes which are thought to be due to an increased anticipation of interaction. Although this paper mostly focuses on the correlations of brain activity in different mobility circumstances, this information can be useful when we will be collecting data to help medical professionals determine the child's progress.

In 2005, the Centers for Medicare and Medicaid (CMS) issued a new national coverage determination for mobility assistive equipment including powered mobility platforms. These standards were developed by clinicians and adopted by many insurance companies by the late 2000's [3]. These requirements specify different necessities mobility devices must include depending on the group of people the device was intended for. Since our mobility device must be able to accommodate children with severe disabilities, the device should comply with all standards required for devices sold on the commercial and residential levels. This means we must include different forms of controls rather than just a joystick. Head switches and buttons are both viable options to complete this requirement. Our device should also accommodate children that do not have the ability to transfer mobility devices independently. To satisfy this, we can make the chair control components easily attachable to several areas on all different types of wheelchairs. If we are not able to meet the requirements listed in the article, the training available to children may be limited due to the fact the training may not be covered by insurance.

Innovative technology changes lives by those it touches, which was the intention when the IBOT 3000 mobility system was created. The authors Heikki Uustal and Jean L. Minkel developed different ways to measure the safety and levels of mobility users were able to achieve while using the device. The device aims to solve many of the current drawbacks of powered mobility platforms that are currently on the market by incorporating gyroscopes, motors, and wheels to achieve dynamic balance reaction in the fore-aft directions [4]. These elements allow users to go on terrain that is typically not supported by most mobility devices. The chair was also able to provide different settings to help train individual users to help them be able to navigate more efficiently. They were able to conclude that all subjects involved in the study were able to improve their driving performance at their own pace. The subjects also reported to medical professionals any instances of falling or inability to perform a necessary task, which was reduced with the introduction of the IBOT 3000 into their daily lives. This study will help give my team andwea more defined insight into the different types of settings that can be incorporated for training purposes. More research needs to be done to fully understand the settings that were used in the study.

The articles discussed above cover different aspects of powered mobility device design. These considerations include the definition and requirements of a “smart” power chair, the effect on a child’s development and the requirements that chairs should abide by in order to be available to different groups of people with varying disabilities, and new innovative technologies in the power mobility realm. Each of these topics add another perspective to the scope of the project and the final product that should be achieved. There is still more research and context needed, however, the information explored above better explains the problem statement our team was given and can be used as a foundation for further research.

The Go Baby Go (GBG) Project

The cars developed by the UD GBG team have had far-reaching impacts that they could not have anticipated. One of the effects of their program has been to help improve the self-esteem of disabled children, as well as making them feel comfortable when they are interacting with other kids. Since its development, this methodology of improving motor skills for kids' development has overall received positive reviews. However, there are constant suggestions on how the project can be improved. These suggestions cater to various motor skill requirements for the kids and various age groups within the pre-high school age bracket. The improvements will ensure that different groups of individuals are catered for, and their mobility is not affected in a way that makes the affected individuals feel “left out”, or otherwise separated from the community. Various researchers and groups have come up with different approaches to provide improvements that can help different demographics, regardless of their situation.

Colombian researchers Restrepo, Velásquez, Múnera, and Quintero Valencia, explored the feasibility of implementing a GBG program in their home country, where the health-care system struggles to provide solutions for young disabled children. To approach this problem, the team adapted the existing GBG design by modifying the seats and the car to improve them and make them fit for different people regardless of age [5]. First, they selected children who had cerebral palsy, and provided their families with one of the modified cars that they could ride. After allowing them to ride the cars, the team gathered feedback from the parents of the children they had chosen for the study in order to analyze the efficacy of the cars [5]. The feedback received showed that the improved mobility of the car increased the comfort of the kids and as such was seen as an improvement to the original program developed at UD. The article therefore concludes that the modifications were beneficial to improving the quality of life for the children, hence being a success.

Mechanical engineering students in their senior year at the University of Portland, Pickering, Fox, Elliot, and Wolwowicz, worked to implement a new improvement to the existing GBG model. The article provides an overview of the currently used GBG and its functions, which is to bring varied modes of movement to the kids with disability [6]. This is known to help them reach an important milestone in their development. For the students’ senior design project, they primarily focused on improving the existing GBG cars by increasing the electrical power supplied to the cars from the original 6V to 12V. By increasing the voltage supplied, the improved car can carry a greater amount of weight, which can allow the GBG cars to reach a greater and more varied audience [6]. The students also explored the use of a head controlled timed switch to activate the 12V car, as well as the possibility of including options for the kids to move in a direction of their choice, either forward, left or right [6]. They were overall successful in implementing multiple head switches, an Arduino microcontroller, and a smoother steering system, among other modifications. These could serve as good examples for our team's initial designs and possible goals.

The team behind "Toy-based technologies for children with disabilities simultaneously supporting self-directed mobility, participation, and function: a tech report" investigated two innovative extensions that could be added to the existing GBG model [7].The first modification was to incorporate a switch that would encourage the mobility of infants below three years. Referred to as the "sit-to-stand" technology, an easy-to-activate, large switch would be added. However, the child would be required to stand up to activate the switch, which would encourage independent mobility [7]. The second modification was to add an arm to the GBG technology, similar to a baseball pitching arm. Known as "”throw-baby-throw" technology, this would allow children with upper extremity issues to experience the feeling of throwing an object, such as a foam ball [7]. The throw-baby-throw would help the kids to easily achieve the motor skill of throwing by providing them with a switch that would make it easier for them to throw things by using the car they are riding on. The incorporation of the switch will help in the mobility of the children as they will be able to move around with no problem.

The GBG project needs to be improved such that it caters to various motor skills for various groups of individuals. Most of the modern research on the improvement of GBG has been directed towards improving the mobility skills of the children [7]. In regards to the children in the age range of the GBG project, they should be provided with a mechanism for developing their motor skills, from simple to more complex movements [8] - [9]. By doing so, the child's overall development will increase, and be able to more closely match the motor skills of other kids of their age who are not disabled. In specific, this could include some playing capability [10]. The same should be applied for the children with disabilities who are past the age of the GBG cars, which could be done through the use of modern communication methodologies.

Augmented Powered Mobility and Electric Powered Wheelchairs (EPWs)

The APM target audience is disabled children who will need an EPW, if they do not already own one. However, disabled children are required to demonstrate their ability to operate an electric powered wheelchair (EPW) before they can be determined liable to receive financial compensation for one [11]. In order to practice viability, our APM platform will utilize various sensors to record performance in several different methods. According to research, some commonly looked for factors are distance and velocity [12] - [13]. Another factor that could be implemented into the APM is slope detection [14], which would be used to match predicted speeds with desired speeds [15]. Additionally, with a target audience of disabled individuals, methods for control need to be determined with accessibility in mind [16] - [18].

Methods for controlling EPWs can vary wildly as it is dependent mainly on the level of disability of their target audience. The team behind "Kinect-based Powered Wheelchair Control System" focused on the elderly who have lost precise motor functions in their hands. Their solution was to utilize a Microsoft Kinect to recognize general hand motions to call the wheelchair to them, and a touchscreen for directional input [16]. While effective, this method would increase the price of the EPW. In another paper, Sreejith T, Vishnu J, and Gopika Vijayan found that a trackball would increase usability while decreasing the cost [18]. While both solutions are workable, neither of these would give the APM's target audience an idea of how to operate a traditional EPW. Therefore, another research paper was examined in order to find alternative solutions.

Microcontrollers are another viable means of controlling an EPW. The research done in "Design and implementation of the electrically powered wheelchair controller based on STM32" shows a method of utilizing two STM32 microcontrollers. The first of the two is used to collect and process sensor data and controller inputs, and the second one is dedicated to controlling the DC motors. This method showed an improved response time between the controller and motor, while allowing for more data acquisition [17]. The research here will certainly be of use to the team, as a microcontroller such as an Arduino, will most likely be used to control our platform and thus the wheelchair as well. Additionally, the concept of using multiple microcontrollers dedicated to separate processes can lead to a more robust design.

Parallel processing can lighten the load on individual microcontrollers. In the 2015 International Conference on Computing, Communication and Automation (ICCCA2015), a "smart" system was presented that utilized an Arduino Uno R3 and Raspberry Pi in conjunction to wirelessly and remotely water plants [19]. Scripts for the Arduino and Raspberry Pi were written in C and Python, respectively. The Raspberry Pi served as the system controller and data hub, while the Arduino served to carry out the sensory I/O, such as powering the water pump, controlling the solenoid valve, and dispensing water according to the level detected in the tank. The control system overall was able to automate and regulate the watering of the plants but was unable to report whether components failed [19]. This source presents a new idea to the team – accomplishing the goal of wireless data transmission could be done with just an Arduino, but the addition of a Raspberry Pi would allow for a microcomputer that could act as the control and data center between the Arduino and the GUI. Furthermore, this combination presents an energy efficient solution, which can be helpful in later stages of the project, as one of the team's secondary goals is to create a low-power platform.

As mentioned previously, utilizing two separate processors simultaneously to increase the functionality of data and input acquisition is important because processing power will determine the number of sensors the APM will be able to use at once. The team lead by Iheb Soussim published a research paper discussing a program to determine if a user can properly operate an EPW [13]. They used sensors to determine distances to surrounding objects and the velocity of the EPW. They also determined that the optimal number of sensors is for distance calculations was five. Though the APM will also have assistive operation modes which will utilize object avoidance/attraction and gravity compensation through a force feedback controller. The two papers labeled “Power-assisted wheelchair with gravity compensation” [14] and “Plugging Brake System as a Hill Descend Control for Electric Powered Wheelchair: Experimental Analysis” [15] utilize slope detection sensors to alter the speed of the wheelchair. The first method is adjusting the amount of current supplied to compensate for the addition push force required for going uphill, allowing the user to stop on a hill without the use of hands. The second method is electronically apply the brakes to a small degree to prevent the EPW from reaching an undesirable speed [15].

Sensors and Graphical User Interfaces

Current EPW testing scenarios can only simulate so many situations while keeping device and user safety in consideration. Therefore, teams are researching the development of simulations for testing EPW viability [12], [20]. Since simulations circumvent the need for a physical EPW to practice on, user accessibility is greatly increased with the decrease of cost. The first paper “Development of a new virtual environment for a power wheelchair simulator: A user-centered approach” focuses on giving users an opportunity to train in a virtual environment [12]. The second paper “Modeling and control techniques for electric powered wheelchairs: An overview” instead used standards defined in American Disability Act (ADA) to create a model to evaluate the difficulty an EPW would have when maneuvering around specific real life locations. This model could then be used to determine where users need to improve and recreate real world scenarios for them to practice with [20].

One of the final major components that the team needs to provide for the PT is a GUI. This is so that the PT can utilize and have control of the configurable inputs for the mobility platform. One of the many studies presented at the 2016 Conference of the IEEE Engineering in Medicine and Biology Society focused on a MATLAB-based GUI [21]. The goal of the GUI was to parse through surface electromyography (sEMG) signal data and view real-time results of adjustments to muscular activity, without having to conduct experiments on physical patients. The sEMG data could be imported as MATLAB (.mat) or text (.txt) files. The GUI that the team developed included user-adjustable sliders for times and thresholds, the choice to import new data, and which limb to focus on [21]. Overall, the GUI was quite successful in providing visualizations for users who wanted to study the effects of sEMG signals on muscular activation and demonstrating the power and value of a MATLAB GUI. Furthermore, this source serves as a strong example of what a clean and “good” GUI could look like, especially when working with professionals who are not as proficient in engineering-specific software. Though the topic was on a niche area in the medical field, valuable information could be obtained on how to proceed with building a GUI using a program that engineers at NAU recognize.

System Organization

To better understand the structure of the system a diagram that details the how the different parts of the system fit together was created. The final product will include many different subsystems, however, to best understand the concepts we listed the most important modules that are needed to have a final product that meets all requirements and constraints.

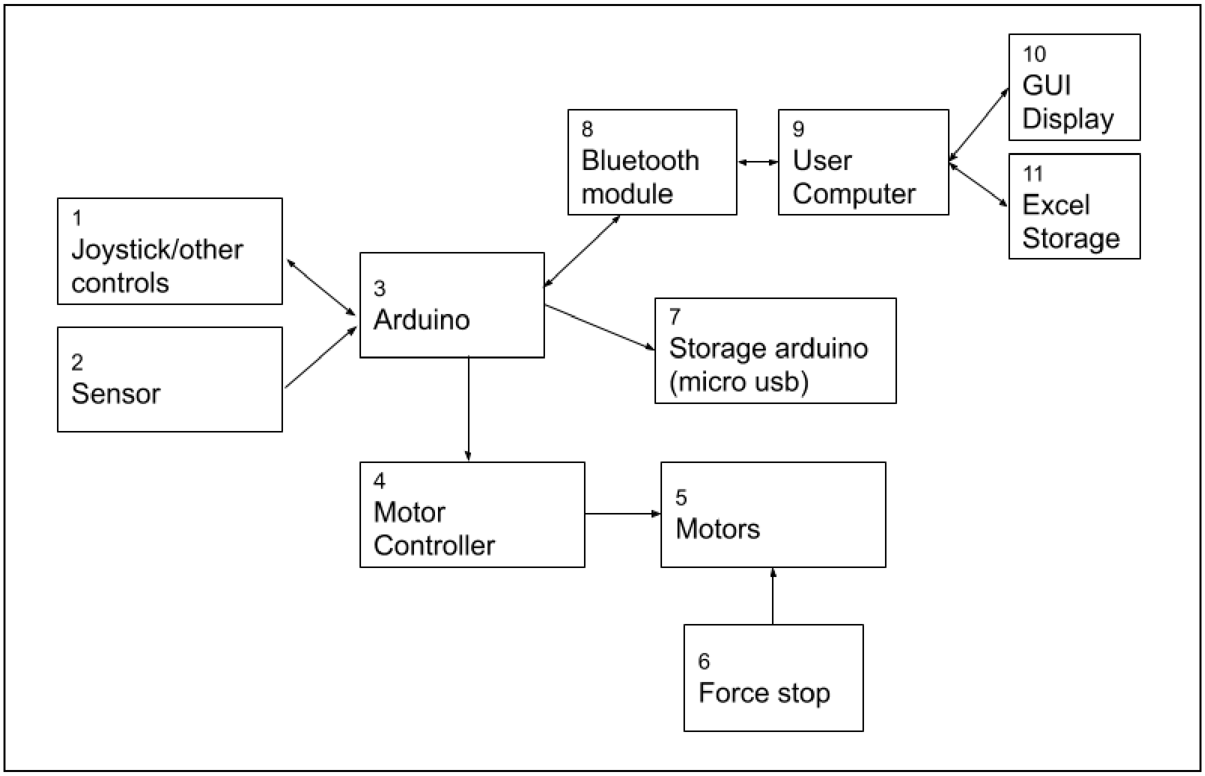


Figure 1: Block Diagram of System

1. Joystick/Other Controls: The controls will be connected to the arduino and will include any device used to move the platform forward/backward and turn. We plan to implement a joystick but head switches are another viable option.
2. Sensors: The sensors will collect data about the platform's orientation in accordance to any obstacles. Using the data we can create graphs that will show the child's driving performance and if there is any improvement.
3. Processor: The processor is the communication between all of the different subsystems in the platform. It will handle the inputs from the driving controls as well as the data collected from the sensors and communicate with the PT's personal computer.

4. Motor Controller: The motor controller communicates between the driving controls and the motors that make the vehicle move. Based on the inputs given by the controls, the motor controller decides what to do, and this is how we have the ability to program different controls to do different operations.
5. Motors: The motors drive the motion of the wheels, so the motors are what ultimately control how the wheels will move.
6. Force Stop: There will be an option to stop the direct flow of electricity to the motors in case of an emergency and can also be used to keep the vehicle parked for long periods of time.
7. Storage: All data collected will be stored on a separate disk in case the data that is sent to the PT's computer is corrupted or lost.
8. Bluetooth Module: All wireless communication will be enabled by BT. The BT module will allow for communication between the GUI on the PT's computer to the processor and will give the PT the ability to wirelessly collect data and change different settings.
9. User Computer: This is the PT's computer that will have the GUI software downloaded onto it.
10. GUI Display: This will be the PT's main form of interaction with the platform. There will be several settings that can be changed from their computer screen as well as visualizations of the driving performance of the child.
11. Excel Storage: This will be the main form of data storage. The data will be laid out in a way that is intuitive to the PT and will allow them to go back and review previous driving sessions.

Early Prototyping and Results

Introduction

The three prototypes that we debuted during the presentation to our client were: two motors controlled by a simple joystick through a motor controller similar to the one we plan to use in the final product, a real-time graph on Processing using a proximity sensor, a paper outline of the plan of the implementation and goals of the GUI. These three prototypes represent three main objectives of the problem statement we were given. The connection of the motor driver circuit will give the child control of the device itself, eventually, the joystick will become force feedback and settings will determine how the force feedback will interact with the child. The sensors will then return different data on how well the child is driving and where there could be an improvement. The sensors will also give feedback to the joystick to help create muscle memory in the children learning to drive powered mobility chairs. The GUI is the main way the physical therapist (PT) will interact with the student so it must be well thought out. Since the target audience for this product are physical therapists, the GUI must be intuitive to the human mind and the display of the data should be easy to comprehend.

Motor Driver Module

The first prototype that we decided on was to demonstrate that we knew how to operate a DC motor with an Arduino and a joystick. The team decided to prototype a smaller version of the final product, in order to learn more about the interaction between joysticks, motor drivers and motors. Since the motor driver we plan on using on the final product would not arrive in time for the prototype, we instead used a L928 dual motor driver to control two 3V direct current (DC) motors. It was expected for this prototype to be tricky, since none of our members have any background in robotics, but paramount due to its direct correlation to the final project. Some aspects that were unknown about this prototype included how to read values from the joystick, and how to connect those to motor speed and direction of rotation. The initial prototype consisted of an Arduino Uno, a single 6V DC motor, and a small analog joystick typically found in Arduino starter kits. The joystick was tested individually with the serial monitor to understand how it worked. Then the single motor and the joystick were brought together and tested, but it was determined that two 12V DC motors would be more suitable to work with, test, and demonstrate. The 3V motors were unreliable as their 16,000 RPM made observations difficult and many burned out as the driver's minimum operation voltage required 12V and no additional measures were taken to protect the 3V motors. The hardest aspect of this project, proved to be the code. There were similar examples online to what we were trying to accomplish, so they were used as a foundation. The logic of the code was simple to code but there were some functions that needed to be used and were more difficult to implement. Once the motors were moving on command, the speed had to be controlled to be more reasonable since the equivalent speed was 35 mph before it was changed. There were a few other small tweeks before the prototype was done that helped with functionality but were not integral to the success of the demonstration. To better show how the module worked, a basic car was built with the motors installed on it using spools of yarn as wheels.

This prototype was deemed a success as it allowed for movement forward and back, turning, and the ability to pivot, though it took a little bit more work and time than the team expected to work out some of the kinks. The biggest hurdle faced with this prototype was programming the motor driver properly to get the motors to rotate in the correct direction and in sync with each other. If this were to be done again, the team would probably create a cleaner physical platform to run the entire system, so that it would be easier to see the parts working. The experience of understanding how a joystick interfaces with DC motors was useful when the new motor driver arrived and we were able to implement the same joystick but with the larger motors installed on the mobility platform. The motor driver that was ordered was actually more simple to use than the prototype we designed and was fully functional by the end of the day. Although it was more work to create to motor driver prototype, the information gained from the experience helped the team understand how the motor driver controls the motors which aided in the quick installation of the full sized version. Though needing to observe 16,000 RPM motors lead to a deeper understanding on how pulse width modulations (PWM) can be mapped to control motor speed. While the prototype allows for movement, the code likely lacks the fine tuning required for a comfortable user experience. Therefore, the next step is to optimize this experience once we can ride the APM ourselves.

Showing Range Finder Data in Real-Time on User PC

The second prototype that we decided on was to demonstrate that we could use Arduino and Processing together to read sensor data and display it in real-time to a user in a closed-loop system. In the project description, the physical therapist specified that she would like user adjustable parameters on things such as object avoidance. In order to detect object avoidance from a computer's perspective, infrared (IR) or ultrasonic sensors are some of the most common. The initial concept was to use an Arduino to read analog inputs from an IR sensor and send them wirelessly over Bluetooth (BT) communications to a Processing.org script. The team decided to prototype an IR sensor that would send data through a serial connection to an Arduino Uno, which would then forward this data to Processing, where it would be graphically displayed to the user. Since the team has never worked with Processing prior to capstone, it was both a challenge and an important goal to at least nail down some basic fundamentals that will be used in the future.

Things went mostly as planned – the closed-loop system created by Lauren was able to detect objects in the 7cm-100cm range of the IR sensor, and output those to the Processing console as green lines, refreshing every second. The prototype was implemented from a salvaged IR sensor from the previous team's project. It was then connected it to an Arduino Uno, and then the data was sent to the serial console in the Arduino software to determine that both the sensor and her initial approach worked. After that step was finalized, the next step was to connect the Arduino and Processing together in order to display graphics that correlated to the sensor's readings. Before the demonstration, it was important (and a major hang-up until this realization was made) to understand that serial transfers, especially those between Arduino and Processing, only occur with strings, and not integers, floats, or doubles. This distinction is especially important because changes had to be made in order for the data to transfer over correctly, otherwise there are readings shown on the output but will not be incorrect. Additionally, a conversion to cm needed to be made so that the numbers and readings had value to them.

This prototype was demonstrated as a success as well. If the prototype were to be redone, a nicer graphic could have been established, with labels and axes to give more weight to the readings displayed. The IR sensor could have also been tested against different surfaces and different colors as well to determine if it behaved the same across the board or was affected by these factors. The experience of understanding how sensors can communicate with Arduinos, as well as the Arduino and Processing IDEs respectively, will be important moving forward as the PT will want to be able to read data with zero knowledge of the electronics and workings. As such, this prototype has made the team reconsider how we will approach the physical display of results, and what we will need to prioritize.

Graphical User Interface

The third prototype that we decided on was to demonstrate that we had a working understanding of what a GUI is, does, and how ours will tentatively look when it comes to the PT. In the project description, the PT specified that she would like a user PC side GUI for setting of parameters and assessment of driving skills. No one in the group has ever really designed user-end interfaces that are meant for people who are not engineers, so we were hoping to learn about how Processing displays and writes to the console, as well as begin to understand what aspects of the GUI are more important for the PT. Things went mostly as planned, though the team over-thought this prototype. Taylor was primarily in charge of this prototype, with inputs and initial ideas from the rest of the team, as well as Dr. Winfree. A rough GUI that allowed a user to communicate with an Arduino Uno, LEDs, and a 3V-6V DC motor with the Arduino IDE, a Processing console, and mouse clicks was first created. Although this did not fit well into the scope of this prototype, especially after talking with Dr. Winfree, having a working understanding of Processing was helpful in what the prototype became. We ended up drawing an idea of what the GUI might look like, and we detailed some of the source and pseudocode that would drive each module in Processing in order to display things that we wanted to see happen. This prototype was demonstrated as a success, and though there was not much that had to be addressed before the prototype was functional, a lot of planning and thought went into breaking down the GUI before anything went on paper. This was probably the trickiest to prototype, simply because it was more theoretical in nature and mostly written only on paper, because this design will probably change overtime when it is better understood what she personally wants to see. However, it is good to start thinking about it now, because we have a feeling this part of the project will become the most labor-intensive part in the later stages of this production.

If we were to do this prototype again, we would want to get in contact with the PT earlier, so that we could start tailoring our thoughts and ideas to meet her needs. Though the prototype itself did not take as long as we thought it would, it had the positive effect of personally opening my eyes to see that we need to think as designers and engineers who are meeting a client's needs. Moving forward we'll continue to remodel our GUI to further improve clarity as well as implementing the features our client expressed. Even though our prototype changed direction halfway through, we think the initial approach was still a learning experience in controlling the Arduino through a GUI. The advice on drawing GUI designs will save us time as making tweaks to an existing GUI is unnecessary. Our client is not an engineer and does not necessarily need all the details that we would obsess over, but instead needs something clean, easy to read, and intuitive to use.

Division of Work

To complete the prototypes we assigned each member a different problem but came together when that person encountered a problem. The work was divided in the following manner: Daniel worked on connecting a motor driver to an external joystick, Taylor worked mostly on GUI research and development, Khaled worked on several necessary aspects of the final project (i.e. the website), while Lauren developed the sensor module. Then when programming errors with the motor driver occured, Lauren and Daniel worked together on programming the DC motor to align with the prototype specifications. This type of approach allowed each person to focus on their prototype and if a problem arose a fresh set of eyes were there to look it over. The approach for each of the prototypes was to find projects that were similar to what each of us was working on and then use the information to create a template to work off of. Then using the well-defined goals we established for the prototypes, we created pseudocode which was then turned into functioning code.

Prototype Learning Outcomes and Demonstration Summary

The main goal of this assignment was to be able to develop three main functions of the final product on a smaller more digestible scale. We expected to learn about how to implement some of the main functions of the project and specifically chose different aspects that our team lacked expertise in. Both the sensor and GUI portions are aspects that no one on the team had previous experience working with, so we wanted to use the time while developing the prototype to learn how to implement these systems. We also chose these prototypes partially since we believed these three aspects of the project would be some of the most difficult.

We learned in-depth about connecting types of controls to a motor controller using an Arduino to program. We also learned about Processing.org script to code a user interface as well as connecting to a range finder sensor to display data. Our team members have developed a deeper understanding in the field of their chosen prototype.

The final prototype presentation was successful and did receive a pass from the client. Before each prototype was fully functional, there were some challenges that prolonged the completion of select prototypes.

Future Implementation of Prototypes

Major challenges that we may face in the future involve the fact that we didn’t have all the parts that will be implemented in the final design. To address this we looked into projects that used the products that were ordered for the final design and kept them in mind while developing the prototypes in hopes that the transition will be relatively seamless.One of the main struggles for our chosen prototypes is that they must be flexible designs so that they may be easily altered to work with the products that were ordered. For example, the motor driver that was used for the prototypes has different specifications than the one that has been ordered and on its way in. The prototype created for the GUI was actually portrayed on paper since the main functions are completely dependent on the client’s needs and preferences which were not able to be clarified until close to the demonstration deadline. Now that the overall goals for the GUI have been established, we are able to move forward with fabricating code for each of the needed functions.

Conclusion

In conclusion, the team was able to successfully create three different prototypes that are integral to our capstone project which included a range finder, GUI, and controls. Unfortunately, the design and creation of each prototype did not go exactly as planned, there were a few hiccups that pushed progress back. For the demonstration, we chose to create a motor driver module that would be able to take input from a joystick to control the direction of the motors. Problems with the motor driver or the sensor module were code based so they were simple fixes but the GUI prototype needed to be altered so that it would be ready in time for the demonstration. We had originally planned to create a simple version of a GUI to control a motor but we decided to change directions since controlling the chair remotely will not be implemented in the final GUI. We then opted for a paper prototype of the GUI that details a visual representation of the user interface as well as the functions that are to be implemented represented by pseudocode. The sensor was the other prototype that needed to be changed so that it was ready to show for a demonstration to the client, however, we did develop a sensor module that was able to take data from a sensor that measures distance and be able to graph the data in real-time. Originally the goal was to develop a closed-loop system between the sensor and the motors so that if something were to get close to the sensor the motors would either stop or slow down. Each of the described functions is important to the success of the final product and should be fairly easy to integrate into future designs.

We also received insight as to how we might be going in the wrong direction. For example, we used an IR sensor to detect range, though IR sensors are dependent on the surface material, therefore we know to start researching other sensors, such as sonar or light. We were also advised to purchase higher end BT adaptors for the Arduino as they are more reliable. And where each prototype fell short of our original goal, we now have a clear direction as to what we need to continue researching.

Thinking about the different aspects that were explained above, we are very excited to continue progress on this project. Although we experienced a few setbacks during the project, we feel we are currently on track to finish the final product on time. Overall, the experience gave our team more confidence in our collective ability to troubleshoot any issues that will arise during the remainder of the project. Anytime we felt we had reached a point where we were having difficulty completing the required task, there was a team member that was willing to help work through it.

Planning and Implementation

As we continue to work and make progress on our project, we also need to plan for the upcoming spring semester. To do so, and in order to communicate clearly with our client, we have broken down our plans for next semester into two types of visuals. The first type is a Gantt chart, which takes the visual breakdown of deliverables for a project, and transfers it to a timeline-oriented version, so that we can estimate how much time each task will take, determine what tasks rely on the completion of others, and budget our time and resources appropriately. The second type is a Work Breakdown Structure (WBS), which breaks our project into smaller components with descriptive deliverables in more of a tree-like fashion.

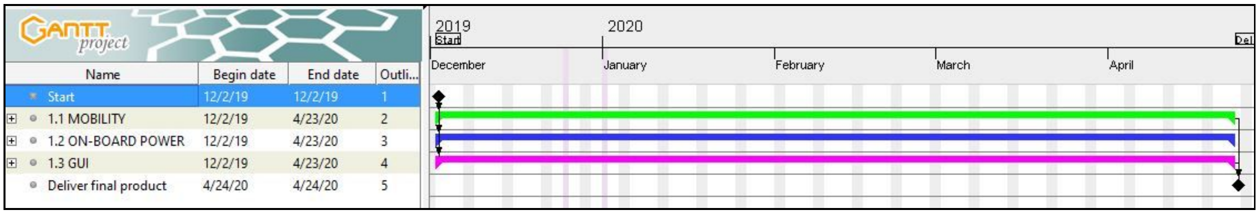
Gantt Charts

The Gantt charts shown in the four figures below provide tentative timelines for our project as a whole, and for each subsystem individually. All of the charts begin with a start date of December 2, 2019, as we based them off of the most recently due deliverable. They all have a common end date of April 24, 2020, as we will have to present our project at NAU's annual Undergraduate Symposium (UGRADS) on that day. Each subsystem has a common deliverable of being ready to test with the other subsystems by March 4, 2020, by which point we hope to have a full system ready for field testing.

A header with each month marked runs across the top of the charts. Arrows represent contingencies and predecessors - for example, a task bar that has an arrow connecting forward in time to another task, denotes that the next task cannot begin until the previous one is completed. Diamonds represent milestones, or important dates and deadlines. Each Gantt chart's content is briefly described below.

System Overview:

Figure 2 presents a basic timeline overview of our project in terms of the three main subsystems described in the WBS charts below: mobility, power, and the GUI. We have already made progress on each subsystem, and anticipate that they will be completed and functional by mid-March 2020, which can be seen in the following three figures. The mobility subsystem is marked in green, the on-board power subsystem is marked in blue, and the GUI subsystem is marked in purple.

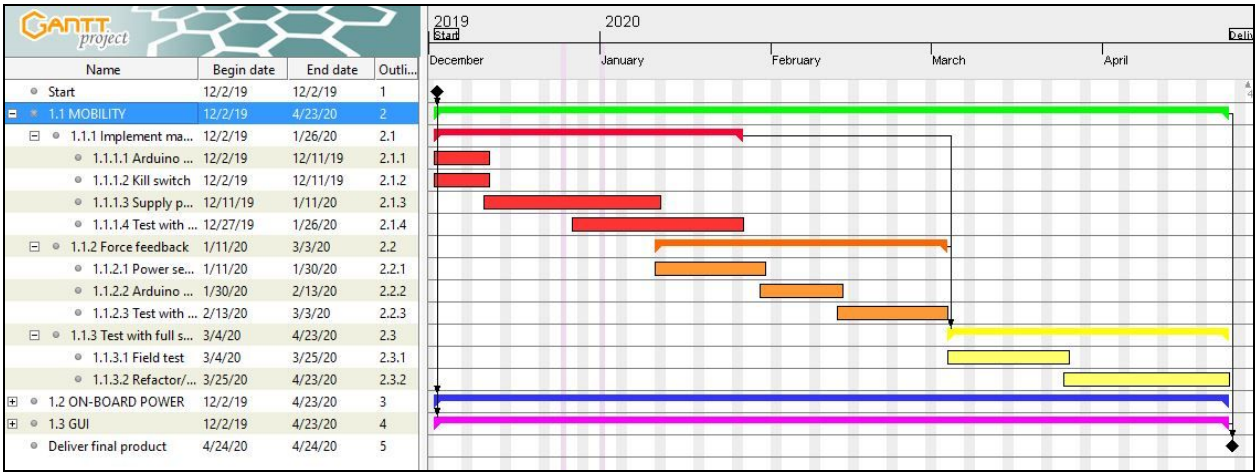


Name	Begin date	End date	Outli...
Start	12/2/19	12/2/19	1
1.1 MOBILITY	12/2/19	4/23/20	2
1.2 ON-BOARD POWER	12/2/19	4/23/20	3
1.3 GUI	12/2/19	4/23/20	4
Deliver final product	4/24/20	4/24/20	5

Figure 2: Gantt chart for System Overview

Mobility Subsystem:

Figure 3 presents a more detailed view of the mobility subsystem that is presented in the WBS in Figure 6. We have already begun work on the main motor driver implementation. Based on the progress we have made in connecting the driver to the pre-existing joystick, we anticipate that we will be able to properly power and drive the platform by the end of January. Once we are able to safely supply the needed amount of power to the joystick and begin testing, we want to concurrently begin working on adding the force feedback joystick element. We anticipate that this will take a bit more time and resources, since we need the majority of the main motor driver deliverable to be completed before we can begin. As such, we have allotted time through the beginning of March to implement the force feedback joystick and adjustable parameters.

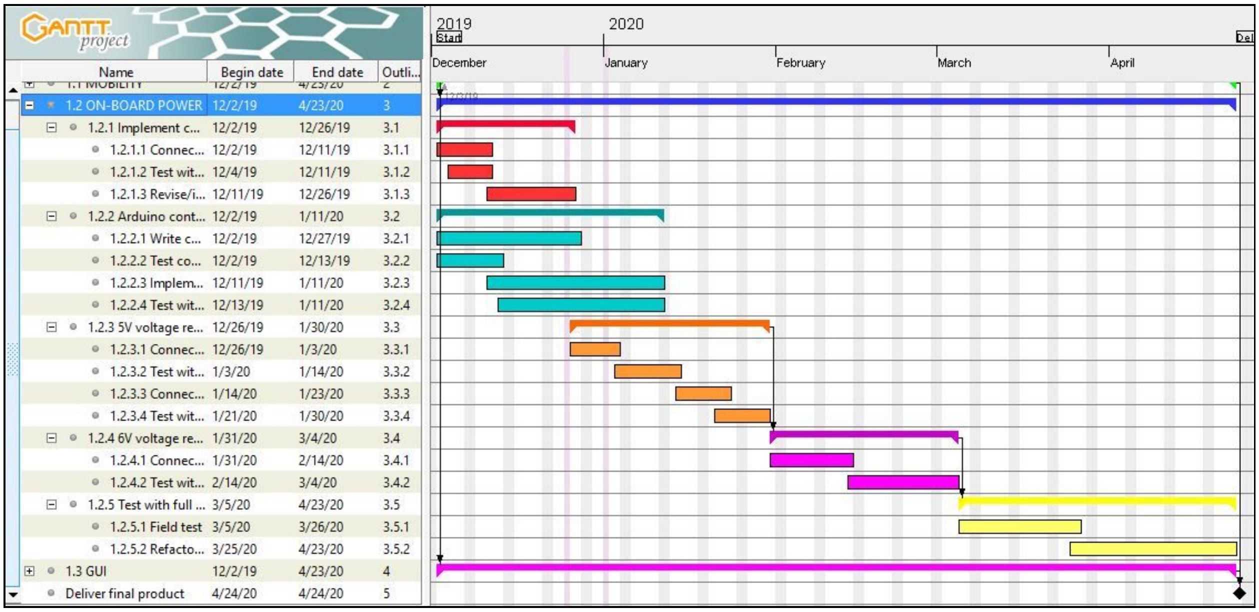


Name	Begin date	End date	Outli...
Start	12/2/19	12/2/19	1
1.1 MOBILITY	12/2/19	4/23/20	2
1.1.1 Implement ma...	12/2/19	1/26/20	2.1
1.1.1.1 Arduino ...	12/2/19	12/11/19	2.1.1
1.1.1.2 Kill switch	12/2/19	12/11/19	2.1.2
1.1.1.3 Supply p...	12/11/19	1/11/20	2.1.3
1.1.1.4 Test with ...	12/27/19	1/26/20	2.1.4
1.1.2 Force feedback	1/11/20	3/3/20	2.2
1.1.2.1 Power se...	1/11/20	1/30/20	2.2.1
1.1.2.2 Arduino ...	1/30/20	2/13/20	2.2.2
1.1.2.3 Test with ...	2/13/20	3/3/20	2.2.3
1.1.3 Test with full s...	3/4/20	4/23/20	2.3
1.1.3.1 Field test	3/4/20	3/25/20	2.3.1
1.1.3.2 Refactor/...	3/25/20	4/23/20	2.3.2
1.2 ON-BOARD POWER	12/2/19	4/23/20	3
1.3 GUI	12/2/19	4/23/20	4
Deliver final product	4/24/20	4/24/20	5

Figure 3: Gantt chart for the mobility subsystem

On-Board Power Subsystem:

Figure 4 presents a more detailed view of the on-board power subsystem that is presented in the WBS in Figure 7. We have already made some progress in implementing the Sabertooth 2x32 motor driver and Arduino control schemes, but we need to revise them for safety purposes and optimal efficiency. As such, we have allotted additional time for these two main deliverables, but we anticipate that they will be functional by early January. In order to stay on track for our project, we decided that once the Sabertooth and Arduino deliverables are well on their way, we could also turn our attention to the 5V voltage regulator. We believe that once we are able to implement the 5V voltage regulator with the Sidewinder joystick, RPLIDAR laser sensor, and other sensors as needed, that the same principles should apply to the 6V voltage regulator. For this reason, we gave ourselves a bit more time for the 5V regulator than we did for the 6V regulator. We anticipate that this subsystem will also be done by the beginning of March so that we can test it alongside our other subsystems.

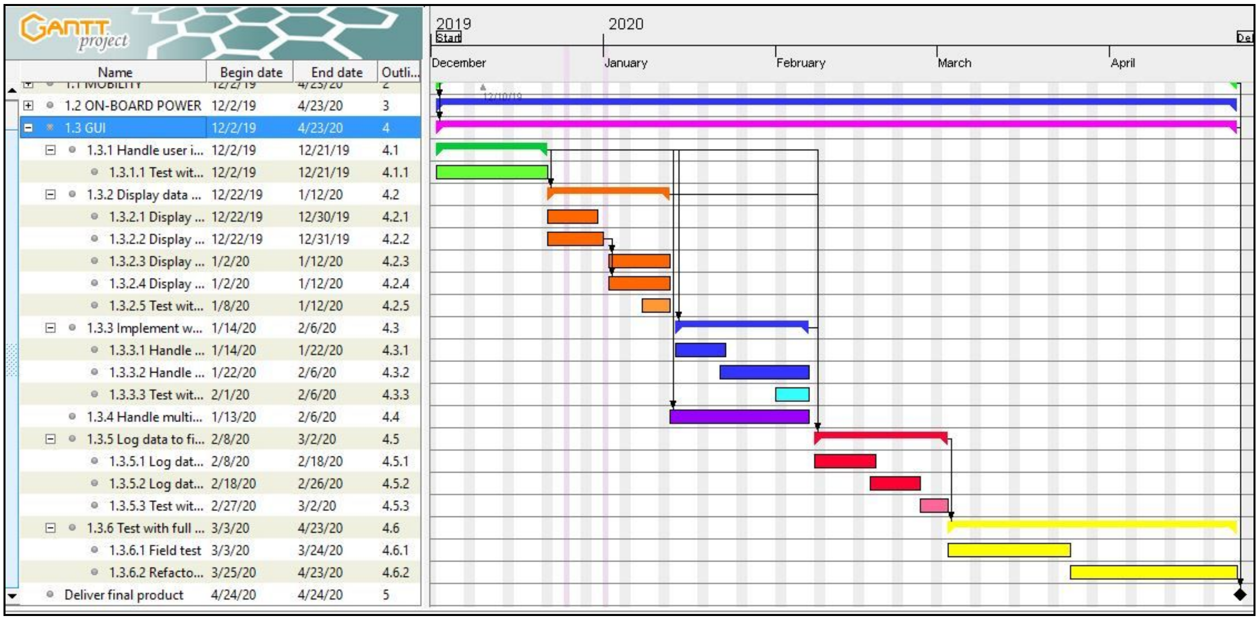


Name	Begin date	End date	Outli...
1.1 MOBILITY	12/2/19	4/23/20	2
1.2 ON-BOARD POWER	12/2/19	4/23/20	3
1.2.1 Implement c...	12/2/19	12/26/19	3.1
1.2.1.1 Connec...	12/2/19	12/11/19	3.1.1
1.2.1.2 Test wit...	12/4/19	12/11/19	3.1.2
1.2.1.3 Revise/i...	12/11/19	12/26/19	3.1.3
1.2.2 Arduino cont...	12/2/19	1/11/20	3.2
1.2.2.1 Write c...	12/2/19	12/27/19	3.2.1
1.2.2.2 Test co...	12/2/19	12/13/19	3.2.2
1.2.2.3 Implem...	12/11/19	1/11/20	3.2.3
1.2.2.4 Test wit...	12/13/19	1/11/20	3.2.4
1.2.3 5V voltage re...	12/26/19	1/30/20	3.3
1.2.3.1 Connec...	12/26/19	1/3/20	3.3.1
1.2.3.2 Test wit...	1/3/20	1/14/20	3.3.2
1.2.3.3 Connec...	1/14/20	1/23/20	3.3.3
1.2.3.4 Test wit...	1/21/20	1/30/20	3.3.4
1.2.4 6V voltage re...	1/31/20	3/4/20	3.4
1.2.4.1 Connec...	1/31/20	2/14/20	3.4.1
1.2.4.2 Test wit...	2/14/20	3/4/20	3.4.2
1.2.5 Test with full ...	3/5/20	4/23/20	3.5
1.2.5.1 Field test	3/5/20	3/26/20	3.5.1
1.2.5.2 Refacto...	3/25/20	4/23/20	3.5.2
1.3 GUI	12/2/19	4/23/20	4
Deliver final product	4/24/20	4/24/20	5

Figure 4: Gantt chart for the on-board power subsystem

GUI Subsystem:

Figure 5 presents a more detailed view of the GUI subsystem that is presented in the WBS in Figure 8. We have already been able to show real-time data for a single sensor, as well as demonstrate that we can receive and handle user input, hence why we believe the handling of user input and displaying data deliverables can be completed by mid-January. Once these two deliverables are completed, we believe that we can concurrently focus on the transferring of metrics and data, both wired and wirelessly, and the handling of multiple sessions of recording. These two deliverables could take more work, hence why we blocked out almost a month for them. After we are able to transfer the data that we want to save, we can turn our attention to writing and storing the data in the Arduino's MicroSD card and from there, to an Excel file and possibly other formats as needed. Our goal for this subsystem is to have a working GUI by early March, so that we can test it alongside the other subsystems in a cohesive manner.



Name	Begin date	End date	Outli...
1.1 MOBILITY	12/2/19	4/23/20	2
1.2 ON-BOARD POWER	12/2/19	4/23/20	3
1.3 GUI	12/2/19	4/23/20	4
1.3.1 Handle user i...	12/2/19	12/21/19	4.1
1.3.1.1 Test wit...	12/2/19	12/21/19	4.1.1
1.3.2 Display data ...	12/22/19	1/12/20	4.2
1.3.2.1 Display ...	12/22/19	12/30/19	4.2.1
1.3.2.2 Display ...	12/22/19	12/31/19	4.2.2
1.3.2.3 Display ...	1/2/20	1/12/20	4.2.3
1.3.2.4 Display ...	1/2/20	1/12/20	4.2.4
1.3.2.5 Test wit...	1/8/20	1/12/20	4.2.5
1.3.3 Implement w...	1/14/20	2/6/20	4.3
1.3.3.1 Handle ...	1/14/20	1/22/20	4.3.1
1.3.3.2 Handle ...	1/22/20	2/6/20	4.3.2
1.3.3.3 Test wit...	2/1/20	2/6/20	4.3.3
1.3.4 Handle multi...	1/13/20	2/6/20	4.4
1.3.5 Log data to fi...	2/8/20	3/2/20	4.5
1.3.5.1 Log dat...	2/8/20	2/18/20	4.5.1
1.3.5.2 Log dat...	2/18/20	2/26/20	4.5.2
1.3.5.3 Test wit...	2/27/20	3/2/20	4.5.3
1.3.6 Test with full ...	3/3/20	4/23/20	4.6
1.3.6.1 Field test	3/3/20	3/24/20	4.6.1
1.3.6.2 Refacto...	3/25/20	4/23/20	4.6.2
Deliver final product	4/24/20	4/24/20	5

Figure 5: Gantt chart for the GUI subsystem

WBS Charts

The WBS charts shown in the three figures below provided more detailed deliverables for each of our three main subsystems, as well as tasks and sub-deliverables for each main deliverable. All of these subsystem charts stem from the Augmented Powered Mobility Platform project, and are critical in order to develop a functional platform for our client. The charts read from top to bottom, starting from the most general blocks and proceeding to more specific tasks at each descending level.

Each number can be read as derived from the block at the level above it. For example, the mobility subsystem is denoted as 1.1, the on-board power system is denoted as 1.2, and the GUI subsystem is denoted as 1.3. The main motor driver is denoted as 1.1.1, which means it is the first deliverable that needs to be completed under the mobility subsystem. The kill switch is denoted as 1.1.1.3, which means it is the third deliverable that needs to be completed under the main motor driver deliverable. Each WBS chart's content is briefly described below.

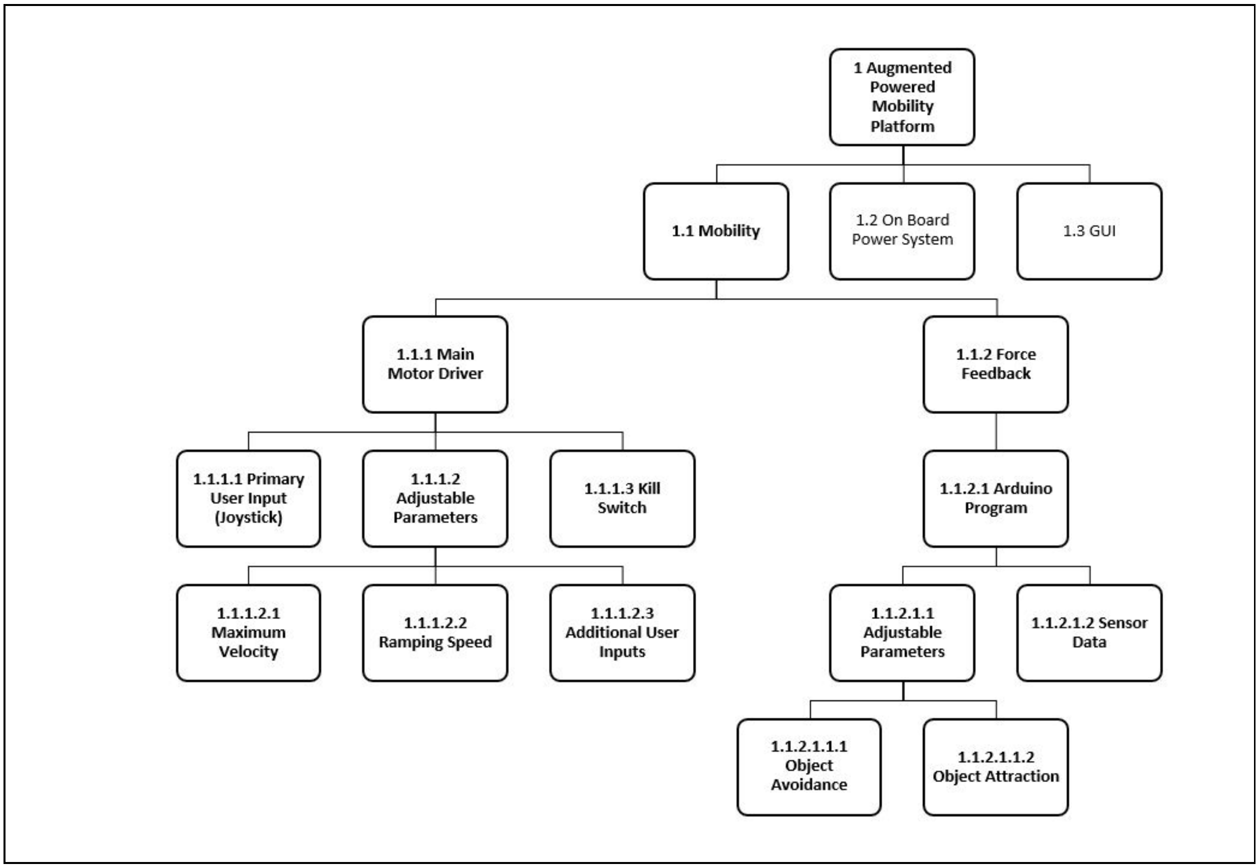


Figure 6: WBS for the mobility subsystem

1.1.1 Main Motor Driver

The Sabertooth 2x32 Motor driver is responsible for powering and operating both primary motors. It receives serial packet commands from a physical link to the Arduino. The proper motor driver dip switch configuration is: 1=OFF, 2=OFF, 3=OFF, 4=ON, 5=ON, 6=ON. Using Library functions for the Sabertooth 2x32, the Arduino has several commands that send 8-bit serial packets to change how the motors operate. This serial exchange has a BAUD rate of 9600 and the input on the motor driver is labeled “S1.Using analog input values from the joystick into the Arduino, write code to specify which command to run according to joystick positioning. Implement options to adjust motor driver parameters, this is to help our client shape the behavior of our device to best suit each individual kid. Reduce the maximum PWM value when mapping the joystick’s analog input values. Lowering the duty cycle will limit maximum achievable velocity. Implement a feature that systematically breaks down changes to PWM and uses delays to reduce the rate at which top speed can be achieved. No such method has been finalized, possible ideas are EEG devices, gyroscopes, and piezoelectric strips. A kill switch controls whether the Sabertooth 2x32 receives power from the two 12V batteries. It does this by using a relay to cut off the positive terminal of the 24V series battery when the kill switch is pressed.

1.1.1.1 Primary User Input (Joystick)

The Microsoft Sidewinder 2 is going to be used as the primary user input to drive the device. It will send analog values to the Arduino, which will then determine how the Sabertooth controls each motor. Alternative user inputs should be implemented in order to broaden usability.

1.1.1.2 Adjustable Parameters

Adjustable parameters will allow the device to be better customized for the personal experience of each user. Changing parameters such as maximum attainable velocity or ramping speed will allow children to operate a safer device in accordance to their skill level and physical well being.

1.1.1.3 Kill Switch

A kill switch has been implemented to ensure that the device can be easily shut down. This was to ensure the safety of the eventual user as well as the team operating the device during developmental stages.

1.1.2 Force Feedback

While force feedback is limited to those who can grasp a joystick, it should still be implemented because of its potential to improve teaching through either added guidance or challenge. Force feedback is achieved by two DC motors that can adjust the X and Y axis position of the joystick. In order to control both of these motors, a smaller secondary motor driver is needed. This secondary motor driver will be located within the base of the controller. Code will need to be written to ensure the force feedback motors act in accordance to the distances of detected objects around the device. Exactly how the force feedback motors respond should be able to be tweaked by our client to ensure the device teaches most effectively. Changing the motors to pull the joystick towards any detected objects within a certain range. This will challenge the user to either react to an unexpected stimulus or increase spatial awareness. The Arduino will be controlling the force feedback motor driver based on analog readings from distance-based sensors. Our team is still debating on which exact method to use for detecting surrounding objects.

1.1.2.1 Arduino Program

An Arduino program will be used to determine how the two motors inside the force feedback controller should respond to a given situation.

1.1.2.1.1 Adjustable Parameters

The Arduino program should be able to switch between different operational modes for the force feedback motors. Two modes current modes being proposed are object avoidance and object attraction. Object avoidance would have the controller pull away from any incoming obstacle, teaching children how to react to collisions. Object attraction would alternatively pull the joystick into obstacles, testing to see if the children can actively avoid obstacles in a more intense scenario.

1.1.2.1.2 Sensor Data

The Sensor data will likely be coming from a RPLIDAR laser sensor. This sensor is a rotates a laser around 360 degrees to scan the room around the device, plotting a 2D map of any obstacles or walls. The distances between the device and plotted obstacles will be used to determine how the force feedback motors operate.

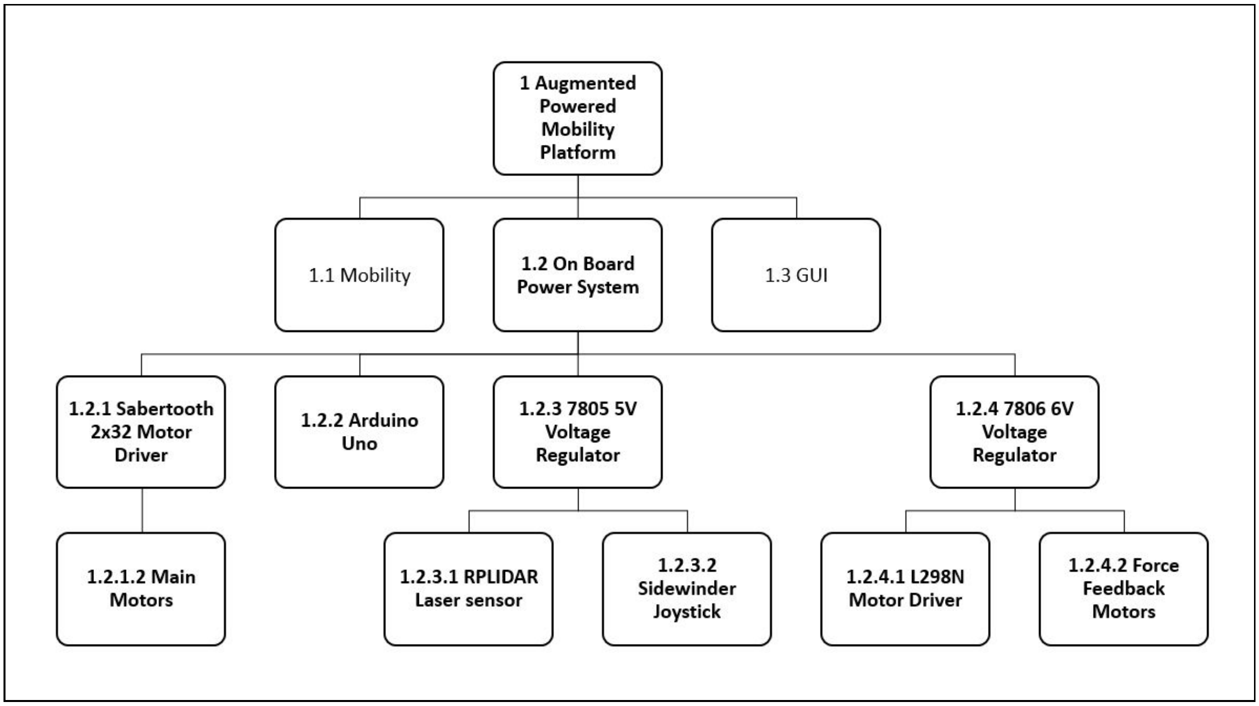


Figure 7: WBS for the On-Board Power System

1.2. On Board Power System

The breakdown of the on board power system outlines the flow of current to each component. The main battery sources are two 12V lead acid batteries. Placing them in parallel will double operational hours while maintaining a voltage more appropriate for the internal components.

1.2.1 Sabertooth 2x32 Motor Driver

The Sabertooth will be directly powered by the two 12V batteries. This device will power the main motors and allow them to draw the necessary amount of current needed for movement.

1.2.2 Arduino Uno

The Arduino Uno has an acceptable voltage input range of 7V-12V. Therefore the Arduino can pull directly from the batteries.

1.2.3 7805 5V Voltage Regulator

The 7805 voltage regulator is capable of stepping the 12V from the battery to 5V and can operate at currents up to 1.5A. Since most logic operations run at 5V, it will be used to power the RPLIDAR laser sensor, as well as the Sidewinder joystick

1.2.3.1 RPLIDAR Laser Sensor

The RPLIDAR requires 5V to power both its laser and motor. The laser scan and motors draw peak currents of 600mA and 100mA respectively.

1.2.3.2 Sidewinder Joystick

The Sidewinder requires 5V for its digital logic and draws negligible current. Therefore it should be powered by the same 7805 as the RPLIDAR.

1.2.4 7806 6V Voltage Regulator

The 7806 works exactly as the 7805, except that it will step the voltage down to 6V instead of 5V.

1.2.4.1 L298N Motor Driver

The L298N operates at 12V and therefore will be powered directly from the main battery. It allows for up to 2A of current to be drawn through it continuously.

1.2.4.2 Force Feedback Motors

Each motor used in the Sidewinder joystick for the force feedback system only draw a peak current of 0.7V. Therefore the L298N motor driver will be used to power both of them.

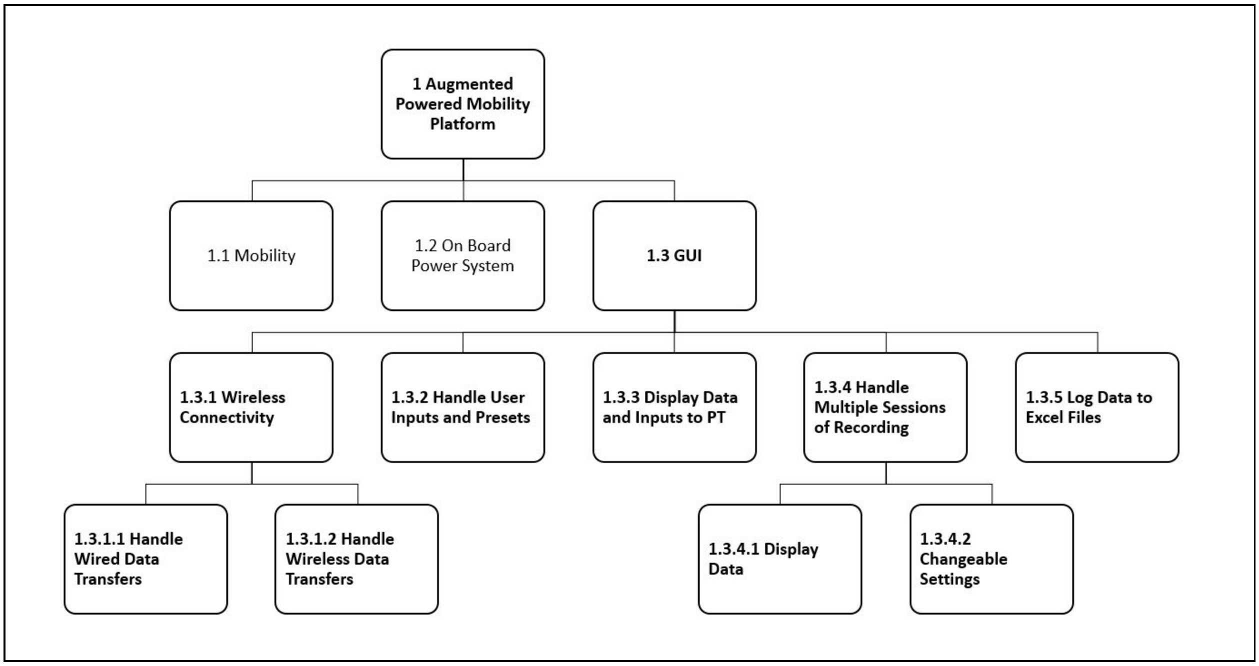


Figure 8: WBS for the GUI

1.3.1 Wireless Connectivity

The Arduino needs to be able to communicate data to the GUI through wireless interaction. For example, if the physical therapist (PT) were to view or edit the real-time data that is being displayed, she should not have to do so off of a laptop perched on the back of the platform. Wires can become desoldered and otherwise disconnected, hence the need for wireless connectivity.

1.3.2 Handling User Inputs and Presets

The GUI will need to be able to take in inputs from the PT, and send command signals accordingly. For example, if the PT decides that the platform should only be able to move at a certain max speed, she could type in the speed, or increment it through the use of a built-in slider. Another example is if the PT decides that the child should only be able to drive forwards, and not backwards, she should be able to change that in the settings.

1.3.3 Displaying data and input to PT

The GUI should be able to display whatever data the PT thinks are important to see. It should be clean, easily read, and able to be compared to other sets of data if needed. Furthermore, the GUI should also be able to display in real time, and possibly be able to switch views, if only certain data needs to be viewed. The data must be presented in an easy-to-understand fashion, so that the PT can draw conclusions from the results without having to run the data through some complex processes. It should also have clear indicators for which settings/presets are in use.

1.3.4 Handling Multiple Sessions of Recording

If the PT wants to be able to record multiple driving sessions, she should not have to change anything, settings-wise. The GUI should be able to record data based on a start-stop button system, without her having to work in the code. Additionally, the GUI should also be able to handle a naming scheme, for cases where she might have multiple patients. These need to be able to be saved separately, and easily understandable/usable for non-engineering users, who might not know how to work on the more technical aspects of the code

1.3.5 Logging data to files

The data collected from the Arduino needs to be logged somehow, in case the PT wants to look back on it later. This could possibly be done solely in the Arduino, but for now it has been placed under the GUI subsystem. One example of an easy-to-access file type is an Excel spreadsheet (.xlsx). From there, the PT could easily create graphs if she so chooses, though she has also stated that the spreadsheet on its own would be invaluable. Furthermore, a comma-separated-values (CSV, .txt) file type was suggested, though this could be a bit trickier, in terms of readability for the PT.

Conclusion

Our team has put in a lot of work in terms of background research. Through our literature review, we learned about how others have designed general powered mobility platforms, and some of the good practices and standards that they followed in order to put the safety and quality of life of the users first. We also found that GBG is not a project unique to Dr. Winfree and the team at NAU, and is in fact a nationwide project that teams have been determined to advance, in order to reach more children and provide new solutions. Furthermore, we investigated different approaches that teams have taken in developing smarter sensing EPWSs, and applications of parallel processing in relation to EPWS. Finally, we discovered some methods of implementing sensors into an APM platform for children, and also gained valuable knowledge in what a GUI can look like in order to be beneficial to users who do not think like engineers do.

We prototyped three components that we believed to be critical in order to make significant full-scale progress on our project. Our motors, motor controller, and joystick prototype demonstrated that we understood how to wire them together and write code for them properly, as well as power the system as a whole. Our real-time graph of proximity sensor data using the Processing UI proved that we could handle a basic sensory input and display it in real time to a user. And our paper outline of the GUI showed that even though we did not have clear direction in terms of what the PT wants to see, data-wise, we can plan out what kinds of designs will be the cleanest and most efficient. Even before this document was generated, these prototypes proved invaluable in helping our team make further progress.

We created WBS and Gantt charts to help not only the client, but us, as team AmpEd, visualize what our plans will look like as we move forward. Now that we have a better understanding of what exactly needs to be done in order to make this project successful, we can begin to assign deliverables and tasks to members of the team so that each member can contribute, while also playing to their personal strengths as engineers. We can also hold each other accountable, and the client can keep us on track as well, time-wise. By using the charts we created, we can modify certain goals and aspects of the project if it appears that we start to either get ahead or fall behind in terms of our projected outcomes.

As team AmpEd, but also as senior-level students at NAU, we want to complete a capstone project that not only allows us to put our theoretical engineering knowledge into practice, but has tangible results that we can see as helping others. The team has a daunting task in front of them: to create an augmented mobility platform for young children to not only practice driving an EPW, but experience independent mobility. We want our project to be not only functional and able to help the current patient that the PT has in mind, but to also be robust and adaptable, such that it can be used in conjunction with disabled children for years to come. We have made a lot of progress thus far, and are extremely excited to continue working on the APM project as we move into the next semester.

APPENDIX I: REFERENCES

[1] R. Kallam and H. Sharma, "Development of Intelligent Powerd Wheelchair", *2017 IEEE 7th International Advance Computing Conference (IACC)*, 2017. Available: https://ieeexplore-ieee-org.libproxy.nau.edu/document/7976855. [Accessed 20 October 2019].

[2] J. Usoro, L. Kenyon, J. Farris and S. Rhodes, "Changes in EEG spectrum and mutual information during power mobility training", *2017 IEEE Great Lakes Biomedical Conference (GLBC)*, 2017. Available: https://ieeexplore-ieee-org.libproxy.nau.edu/document/217385. [Accessed 20 October 2019].

[3] B. Dicianno and E. Tovey, "Power Mobility Device Provision: Understanding Medicare Guidelines and Advocating for Clients", *Archives of Physical Medicine and Rehabilitation*, vol. 88, no. 6, pp. 807-816, 2007. Available: https://www-sciencedirect-com.libproxy.nau.edu/science/article/pii/S000399930700233X

[4] H. Uustal and J. Minkel, "Study of the independence IBOT 3000 mobility system: An innovative power mobility device, during use in community environments", *Archives of Physical Medicine and Rehabilitation*, vol. 85, no. 12, pp. 2002-2010, 2004. Available: https://www.sciencedirect.com/science/article/pii/S0003999304006562

[5] Restrepo, P., Velásquez, J., Múnera, S., and Quintero Valencia, C. A., in *Assistive Technology*, “Adapting ride-on toy cars as a tool to promote leisure: A feasibility study in Colombia,” Jun. 2019. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/?term=Quintero%20Valencia%20CA%20MSPH%5BAuthor%5D&cauthor=true&cauthor_uid=31199193

[6] Pickering, L., Fox, A., Elliott, P., and Wolwowicz, J., “Go Baby Go - Assistive Technology,” University of Portland, Portland, OR, USA, 2019. [Online]. Available: https://pilotscholars.up.edu/cgi/viewcontent.cgi?article=1019&context=egr_project

[7] Logan, S.W. *et al*., “Toy-Based Technologies for Children with Disabilities SImultaneously Supporting Self-Directed Mobility, Participation, and Function: A Tech Report,” Frontiers in Robotics and AI, Ithaca, NY, USA., 2017. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2017.00007/full

[8] Fitzgerald. J. R., “The Impact of Family Income and Primary Caregiver Education Level on the Usage of Modified Ride-On Cars Among Children with Disabilities”, B.S. thesis, Dept. Pub. Health., Oregon State University, OR, USA, 2019. [Online]. Available: https://ir.library.oregonstate.edu/concern/honors_college_theses/37720j77h

[9] Logan, S.W. *et al*., “Standing Tall: Feasibility of a Modified Ride-On Car That Encourages Standing,” Jan. 2019. [Online]. Available:

https://www.ncbi.nlm.nih.gov/pubmed/30557294

[10] Logan, S.W. *et al.*, "Real World Tracking of Modified Ride-On Car Usage in Young Children With Disabilities," in *Journal of Motor Learning and Development*, Dec. 2019. [Online]. Available: https://journals.humankinetics.com/view/journals/jmld/7/3/article-p336.xml

[11] "The Process," *The Process Of Obtaining Power Chairs And Manual Wheelchairs*. [Online]. Available: https://www.freedomhme.com/The-Process. [Accessed: 28-Oct-2019].

[12] P. S. Archambault *et al*., "Development of a new virtual environment for a power wheelchair simulator: A user-centered approach," *2013 International Conference on Virtual Rehabilitation (ICVR)*, Philadelphia, PA, 2013, pp. 216-217. doi: 10.1109/ICVR.2013.6662129

[13] I. Soussi, M. Mrabet, F. Fnaiech and P. Gorce, "Expert system for the decision on the ability to drive power wheelchair based on fuzzy logic," *2013 International Conference on Electrical Engineering and Software Applications*, Hammamet, 2013, pp. 1-6. doi: 10.1109/ICEESA.2013.6578454

[14] S. Hwang, C. Lee and Y. Bang, "Power-assisted wheelchair with gravity compensation," *2012 12th International Conference on Control, Automation and Systems*, JeJu Island, 2012, pp. 1874-1877.

[15] S. M. Asyraf, P. M. Heerwan, I. M. Zulhilmi and I. M. Izhar, "Plugging Brake System as a Hill Descend Control for Electric Powered Wheelchair: Experimental Analysis," *2019 IEEE 4th International Conference on Advanced Robotics and Mechatronics (ICARM)*, Toyonaka, Japan, 2019, pp. 319-324. doi: 10.1109/ICARM.2019.8834047

[16] C. Chang, C. Chen, C. Chen and B. Lin, "Kinect-based Powered Wheelchair Control System," *2013 4th International Conference on Intelligent Systems, Modelling and Simulation*, Bangkok, 2013, pp. 186-189. doi: 10.1109/ISMS.2013.74

[17] X. Shan, M. Li, H. Yan, Q. Wang and Z. Lan, "Design and implementation of the electrically powered wheelchair controller based on STM32," *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*, Beijing, 2015, pp. 1484-1488. doi: 10.1109/ICMA.2015.7237704

[18] S. T, V. J and G. Vijayan, "Trackball Controlled Novel, Cost Effective Electric Wheelchair," *2018 International Conference on Control, Power, Communication and Computing Technologies (ICCPCCT)*, Kannur, 2018, pp. 205-209. doi: 10.1109/ICCPCCT.2018.8574266

[19] N. Agrawal and S. Singhal, "Smart drip irrigation system using raspberry pi and arduino," *International Conference on Computing, Communication & Automation*, Noida, 2015, pp. 928-932.

[20] R. Velázquez and C. A. Gutiérrez, "Modeling and control techniques for electric powered wheelchairs: An overview," *2014 IEEE Central America and Panama*

Convention (CONCAPAN XXXIV), Panama City, 2014, pp. 1-6. doi: 10.1109/CONCAPAN.2014.7000435

[21] N. Agrawal and S. Singhal, "Smart drip irrigation system using raspberry pi and arduino," *International Conference on Computing, Communication & Automation*, Noida, 2015, pp. 928-932.