AI Cart

Team 6: Cam Osborn, Abdullah Almarzouq, Eric Lauber, Daniel Sayenko

Scrum: Bogdan Filipchuk

Table of Contents

Executive Summary 2

Motivation 3

Project Requirements 4

Project planning 8

3rd Sprint Review and Retrospective 10

Prototype Description 16

Prototype Testing 20

Technical Summary 24

Teamwork Discussion 24

Lessons learned 26

References 27

Appendix 28

I. Executive Summary

With this project we planned to automate the most mundane part of the shopping process, pushing a cart around. Using nothing but a simple Raspberry Pi, a camera, and some software we were able to create a controller that follows a designated user. Attaching this to an electric motor system our team propelled the cart around. Adding on an ultrasonic sensor suite, our team taught the AI Cart to avoid objects surrounding itself. With an integrated touchscreen, it gives a visual representation of remaining battery charge. Our Team also added a VOIP calling system with a preset number, allowing customers using these carts to quickly get in contact with customer service. This is extremely helpful to get their questions answered or concerns addressed.

II. Motivation

Our team has identified the grocery cart as the next piece of equipment lacking technological advancements. There have not been many advancements in the grocery store outside of the way one pays their grocery bill. Our team decided it was time to change this struggle. We chose one way we could help grocery shoppers, more specifically, the family, is to create the AI Cart. While the parents go to the grocery store, there can be many challenges. Some of these challenges include carrying their children, walking with their children, and keeping an eye on their children, all while pushing the grocery cart. While these are typical parenting responsibilities, the grocery cart can get in the way of this. We decided to pursue this challenge because implementing something that can follow a user is something that is deemed to be very challenging. In addition to wanting a project that would be challenging, our team wanted to do something that we might see in the future. Artificial intelligence has made its way into the automotive industry, Travel Industry, and others. Our team saw this trend and wanted to challenge ourselves to see if we could solve a problem in the shopping industry. Artificial intelligence will eventually have a place in each sector, but the question is: where will the advancements be made? As stated before, the family at the grocery store can have issues pushing the grocery cart at the same time they are tending to their families. The demographics the AI Cart will directly affect are the families with young children, single parents, and young adults. Our Team does not see the AI Cart directly affecting the older generations because some of them lack the technology to use some of the features implemented within the AI Cart. Some of the older generations will use motorized shopping carts. Finally, some of the older generations may rely on leaning on the original grocery cart to help them walk throughout the store.

III. Project Requirements

Our proposed solution is the AI Cart, which follows the user throughout the shopping experience. This cart allows the user hands-free shopping experience. The demographics will be satisfied by the AI Cart because the parents will be able to take care of their kid's needs without having to worry about pushing the shopping cart. The younger demographic will have the ability to walk around the store casually with their phone in hand. The following functions were implemented on the project prototype. These functions are all things our group found to be imperative to get the AI cart to a position where it could be implemented in the real world.

Functions:

- The AI Cart shall not run into objects or persons. - This is defined as common obstacles found in and around a grocery store (i.e., People, Display Cases, Doors, Shelving, Clothing Racks, Cars, Sidewalks, Pylons).
- The AI Cart shall not harm others.
- The AI Cart shall have a neutral zone.

After defining the function our team wanted to see in the AI Cart, we set out to create the functional decompositions. These decompositions can be seen below.

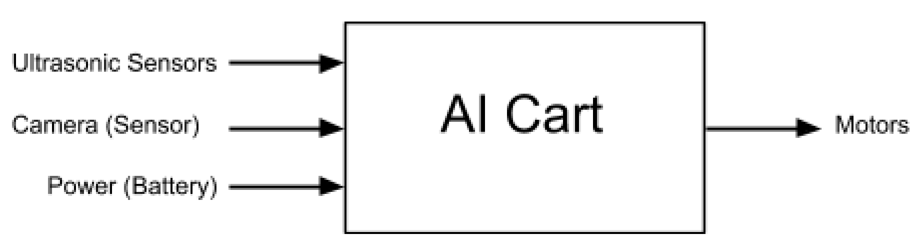


Figure 1: L0 Decomposition

The three inputs are seen in figure 1 that drive the AI Cart. All of these inputs go through the brain of the AI Cart and control the motors. This is the most basic schematic of the AI Cart. While this is neat and easy to understand, the L1 Decomposition shows more of what goes into making the AI Cart Function. This L1 Decomposition can be seen in figure 2, below.

Performance:

- The AI Cart shall avoid obstacles within a designated test course. - common barriers found in and around a grocery store (i.e., People, Display Cases, Doors, Shelving, Clothing Racks, Cars, Sidewalks, Pylons).
- The AI Cart shall follow the user through a designated test course. - in and around a grocery store under three mph.
- The AI Cart should have a day's long battery (8-12 hours). - For now, we are working on a smaller scale; therefore, the battery life can be as long as three hours.

As our team didn't have access to the standard grocery store environment, we decided to use a hallway to mimic the grocery store environment. This testing process is further described below, in short, our team met the requirement to avoid obstacles within a designated testing course. Furthermore, our AI Cart follows the user through the hallway. This, again, was used to mimic the grocery store environment. The cart was able to achieve the desired result, under 3 mph, at 1.5 mph. Finally, our last performance requirement was an all days long battery. However, due to the size of the cart, we used a smaller size battery, with a goal of 3 hours. We used Lipo batteries, which hold longer under load. This has also been discussed in the testing section.

Constraints

- The AI Cart should not flip over. The cart doesn't maintain enough speed to flip over. Furthermore, we have placed the heavy electronic components in the bottom basket to promote this stability.
- The AI Cart should be inexpensive enough for stores to buy it. Our AI cart prototype cost \$216.00.
- The AI Cart should have a weight limit of 25 lbs. While designing the cart, we used a smaller grocery cart, so it does not hold the same weight as a standard large grocery cart, but it does hold twenty five pounds.
- The AI Cart should be designed with a family-friendly design. Our team copied the design of a standard grocery cart.
- The AI Cart should have a speed limit to promote safety. This has been implemented using a motor driver (L293D).
- The AI Cart should have smooth/rounded edges in order not to puncture nor harm any person.

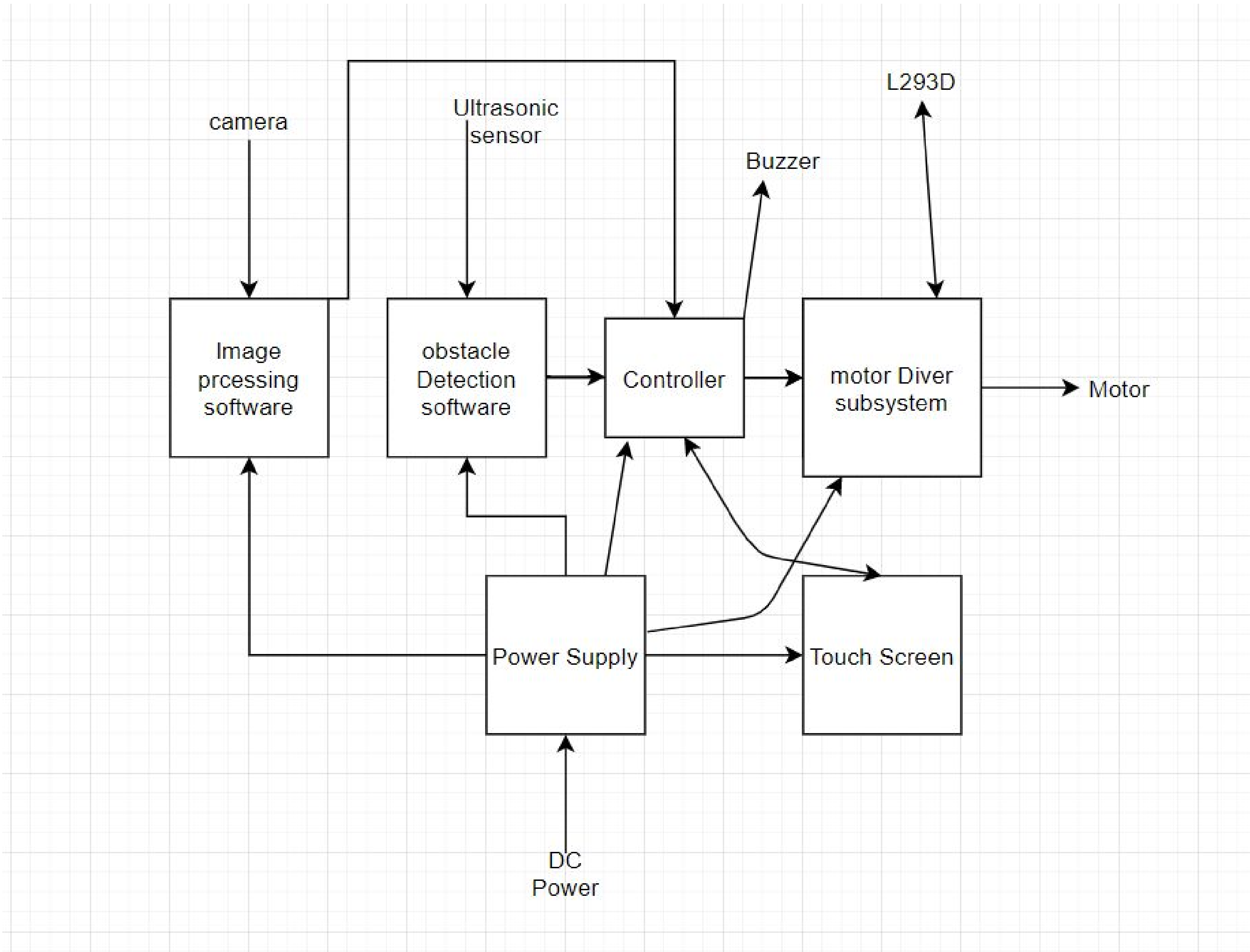
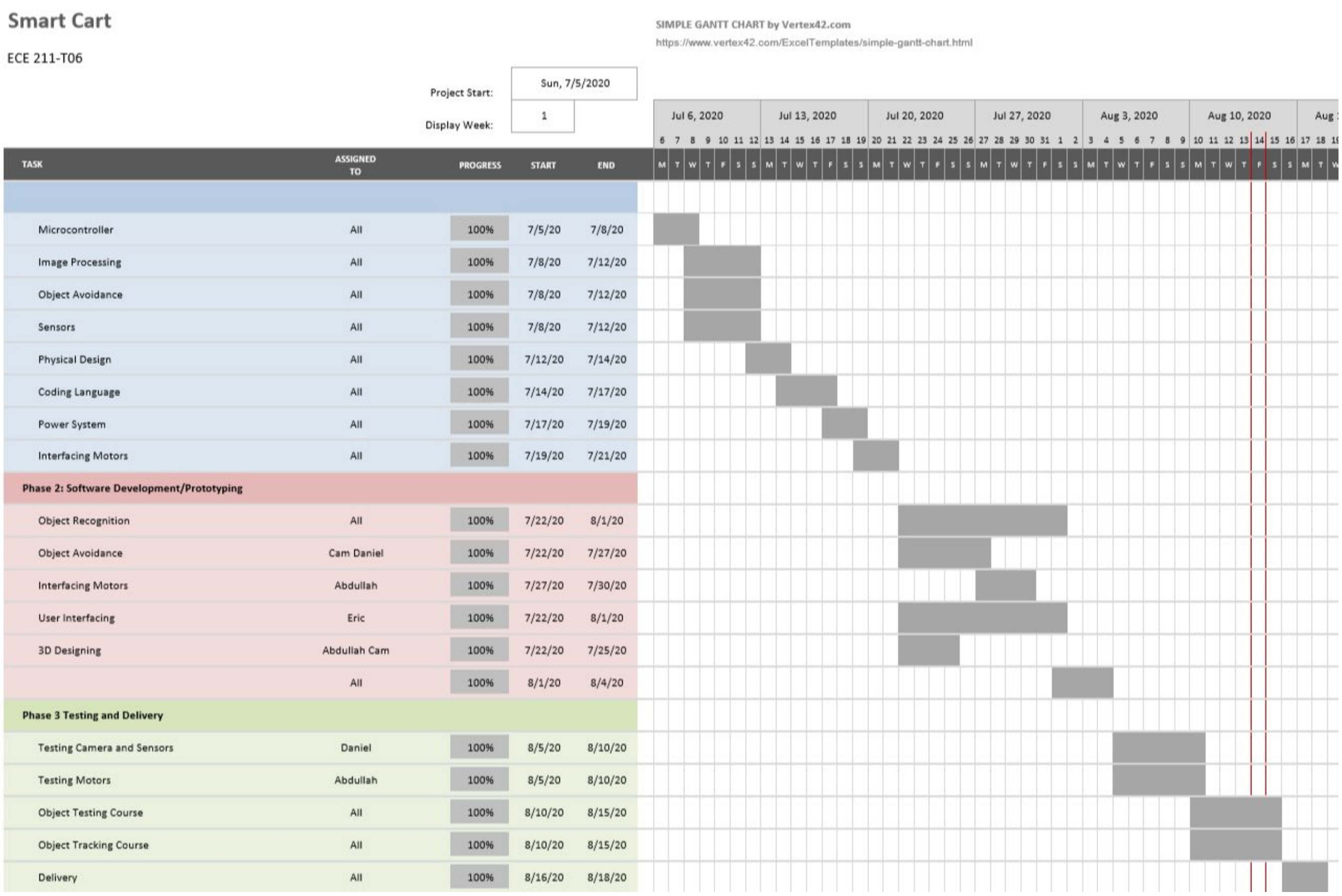


Figure 2: L1 Decomposition

IV. Project planning

As planned at the beginning of the course, we followed the standard Scrum way of doing the project. We had daily meetings (Monday-Friday) and used Discord to communicate with each other when we did not have the scheduled class (Zoom) meetings. Besides using Discord for the meetings, we had organized chats where we talked about the project's progress, problems, and informing each other if we will miss a meeting.



https://drive.google.com/file/d/1NC53nfKGFc5LVV7zYaelowkZzOlSegu3/view?usp=sharing

Figure 3: Gantt Chart

The Gantt chart did not change that much from the first week of the class. When creating the Gantt chart, we knew that we would have setbacks so we made sure we had enough time for each part of the project. Also, we put a lot of time at the start of the course to understand all the things we want to accomplish with the project. The Gantt chart was not too specific, but it had the main details to complete the project. The Trello board was used as a more detailed, refined project plan.

Specializations:

- Abdullah’s specialization for the project was interfacing with the motors and 3D design for the cart.
- Cam’s specialization for the project was Object Avoidance and 3D design.
- Eric’s specialization for the project was the user interface.
- Dan’s specialization for the project was object Avoidance.

Overall, the project planning worked very well; everyone in the group used superb communicating skills, which made it easy for everyone to know at what stage the project was at. The Gantt chart was helpful to visualize the long term goal of the project as well as the timeline. The Trello board worked very well for sprints, as it was designed to do; it made it easy for everyone to know what the tasks are. Trello was an excellent tool for short term timeline tasks.

A few things we could have improved on the project planning was to utilize the Trello board more by adding more details and photos of the project. Comparing the three sprints, the detail on the Trello board was increased. Throughout the project, some tasks became unneeded, and some tasks became more complicated than we have planned for. For example, for Abdullah's responsibility, we didn't account for how large the 'interfacing motors' job was. This task was much more than interfacing motors, and it was the brain of the project where all the sensors and camera communicated with the motors and outputted data to the user.

V. 3rd Sprint Review and Retrospective

ECE 212: Sprint Review, Retrospective & Planning report for team _6___

Cam Osborn, Abdullah Almarzouq, Daniel Sayenko, Eric Lauber

1 Sprint Review

1.1 Previous Sprint Goal

Our Highest priority is having a finished product to share with the class. This product needs to have functioning items such as: a GUI, 3D Printed Body, recognize it's user, and the Sensors and Camera need to give the motors inputs.

1.2 Sprint Backlog Status

SPRINT ITEMS DONE	1. 3D Design Touch Screen Mount 2. Finished Prototype 3. Testing Sensors 4. Testing Motors 5. User Interface 6. Testing GUI 7. Delivery

Sprint Backlog Items Not Done	Reasons for NOT done	Keep in Backlog (Y/N)?
NA	NA	NA

We have completed the final project, there are no candidates for the next sprint.

There are no candidates for the next product backlog.

Our remaining product backlog is as follows. These are all items that we did not have time to integrate.

- Integrated app
- Able to scan barcodes
- audible alert when found user

1.4 Notable Technical Accomplishments

- The cart can keep a constant recognition of the user if they are walking slowly.
- The battery's level and cart's POV can be seen from the LCD screen.
- The cart can last up to 2-3 hours while everything's connected (LCD screen, camera,...,etc.).
- The fully functional finished cart's prototype.

1.5 Technical and Other difficulties

One of the technical difficulties we faced was the wiring of the cart. Unfortunately, we didn't count the space and holes for the wires when we were designing the prototype which made it difficult to wire the circuit & components while also hiding the wires/cables at the same time. However, we managed to drill a couple of holes through the basket to pass the wires and cables. Moreover, some of the other issues that we faced were the shape of the base which forced the motors to be mounted at an angle and the wheels appeared to have too much grip which prevented the cart from turning left or right. However, we fixed those issues by redesigning the base to have squared shape and upgraded to all-directional (Omni-directional) wheels which were the best option for our method of turning. One of the non-technical difficulties was the camera's angle. We realized that the camera needs to be mounted at a specific angle even before assembling the prototype. The camera's angle was adjusted to the most accurate angle during the testing phase.

1.6 New Skills

Cam learned how to animate items using Fusion 360 for the final video. This was very difficult, as it wasn't well integrated into the 3D Modeling software, so it took upwards of 8 hours and over 10 versions.

Eric learned the use of the gtk library in creation of user interfaces and signal handling functions. Refined his soldering skills in the integration of a PCB into our cart design. Learned about the use of I^2C protocol in communicating with a controller and the timing signals going into the implementation.

Abdullah learned the basics of GTK 3.0 C library as he has used Eric's UI program to show the battery level on the progress bar. Abdullah also learned how to program the LCD screen to show the desktop of the Raspberry Pi, the battery level and the cart's point of view. Abdullah also learned how to use the LTC2943 circuit to read the battery's level (aided by Eric's and Bogdan's help and notes).

Daniel Learned how to edit videos for the final presentation, it took over 12 hours to collect audio, video clips, edit, export, and upload to drive.

Sprint Retrospective

2.1

Over all the sprint planning very well, we met everyday and discussed the progress and challenges we had. We went over the Trello board and added tasks and made sure all the tasks were completed.

•2.2 Teamwork and planning - things that went well

- Team members managed to hand over components multiple times which made it easier to build and assemble the cart's prototype.
- The division and distribution of tasks were done responsibly and fairly which led to an efficient use of time and effort as every member of the team was accomplishing some task(s) at all times, almost.
- All of the team members kept an open mind to adjustments and were willing to spend the amount of effort to help other team members accomplish their task(s).
- All of the team members were trying their best to be around and present at all times in case any one of us needed something or had some question(s) about something related to the project.
- None of the team members were hesitant when it came to acquiring/buying the needed components and such. Even if that meant more than one member had to acquire/buy the same component(s) (since we are working remotely).

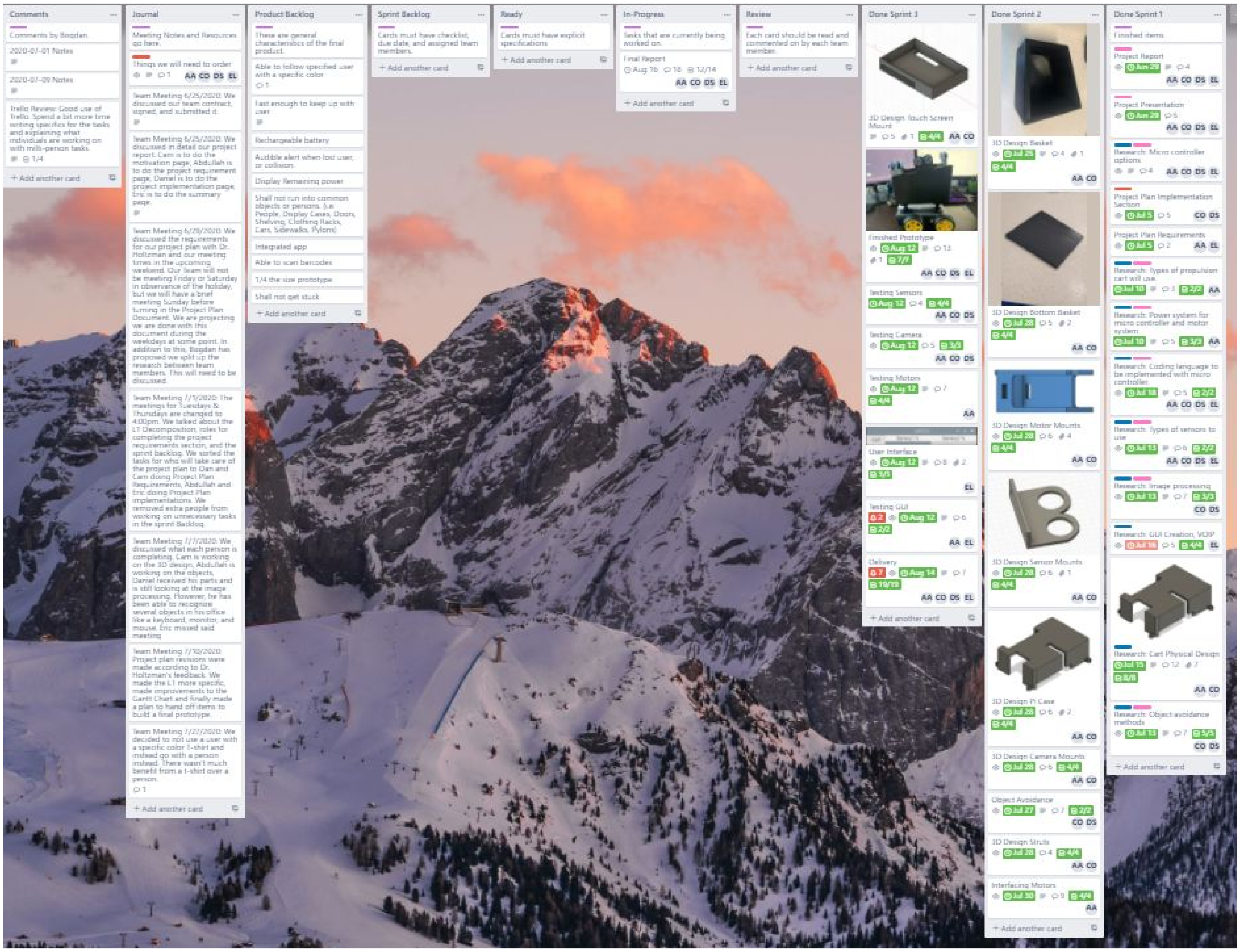
Teamwork and planning - things that could be improved and suggested improvements

- Even though we've managed to deliver a finished prototype, working remotely as a group can be difficult because in some cases some of the team members might lack the needed components and/or skills. Whereas, if we were working together in person we wouldn't have faced those kinds of problems. Yet, the solution to this issue is out of our reach at this moment due to the COVID-19 pandemic.

2.3 Trello

We believe that our team used trello to the best of our ability during the final sprint. Each sprint we improved upon our Trello Board. Each team member during the final sprint increased their usage of the board, adding more checklists, comments, etc.

2.4 Trello Screenshot



https://trello.com/b/mzTsOm5V/ece212su20t06

Overall, we would give ourselves a score of ___5____ (out of 5) for how well this sprint went.

Our team ___6__ *(give name, e.g., T01)* met with our Scrum Master, *Bodan* on Fridays. We discussed Sprint Planning. All team members have read this report and agree that it accurately describes our discussion.

VI. Prototype Description

Our team implemented our prototype of the AI Cart using the 3D printed parts found in the Bill of Materials, figure 7. There have been several iterations of the final AI Cart. The first iteration was a cardboard model, to test the code, sensors, and camera. This was created as a base model and was not meant to carry anything outside of the wiring harness, battery packs, and the sensors. This initial prototype can be seen in figure 4. Next, the 3D printed model was created. This was used as the final prototype. The design of it was modeled after a small grocery cart, at ¼ the size. There were a couple of changes made to the model of a small grocery cart to fit the extra harness and battery packs. The final prototype can be seen in figure 5.

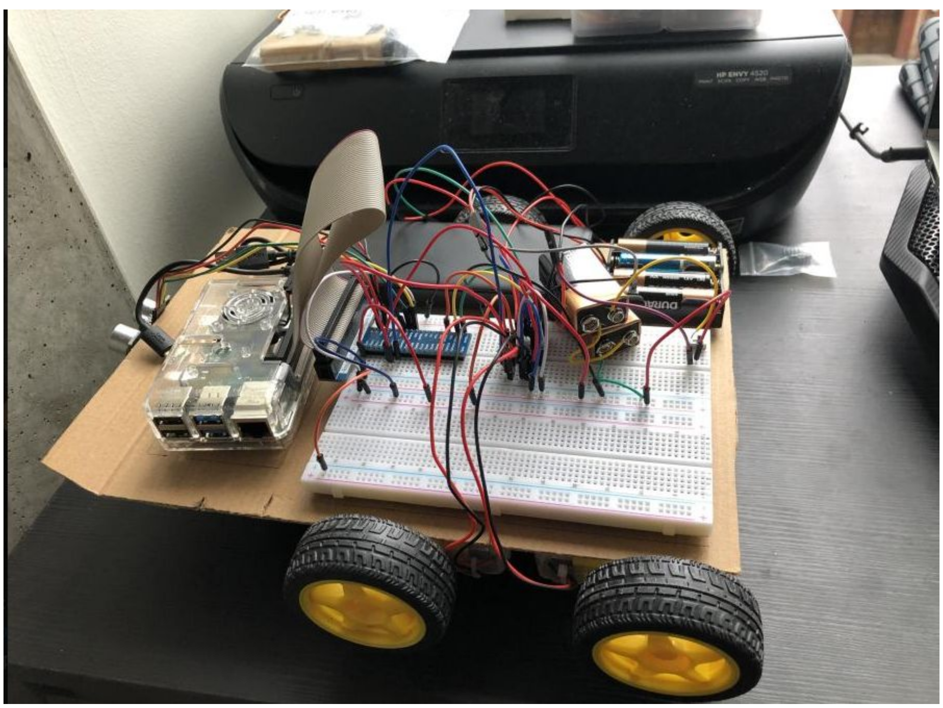


Figure 4: Initial Prototype

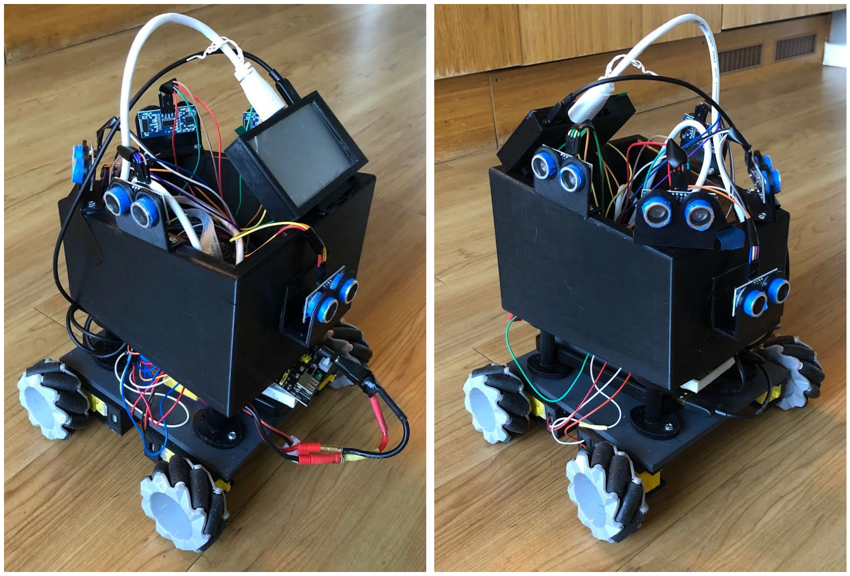


Figure 5: Final Prototype

The implementation of the final prototype was completed using the Raspberry Pi 4B with 4GB of ram. This Raspberry Pi was the brains of the AI Cart. Using TensorFlow lite, google’s image processing software (written in Python), and the Raspberry Pi Camera, our team has been able to recognize people, and the AI Cart will follow based on inputs from the area of the person the camera has identified. While the cart follows people based on data from the Raspberry Pi Camera, the cart avoids objects using the six ultrasonic sensors mounted around the cart. These ultrasonic sensors are installed in the front right and left corner of the cart, left and right side, as well as the front and back of the cart. Our team decided this covered the most area of the cart and would provide protection from standard objects found within a grocery store. The distance of impeding objects supplied from the ultrasonic sensors gave instructions to the Raspberry Pi to decide whether the cart needed to turn. The Raspberry Pi camera’s main function is to find the person and determine the person's area; through the software of TensorFlow Lite. Once the area was identified, the pi would also determine if the person was centered, left, or right. This would then give direction to the motors. Furthermore, the camera would decide if the user was in a neutral zone. This neutral zone would allow the AI cart to stop moving and avoiding its user so the user could add items into the cart.

The user interface has only 1 interactive item, the call button. When pressing the touchscreen the Raspberry Pi will open the softphone program and start a call to a coded number. Once the person on the other side answers a mic has been implemented to allow for voice capture. Once complete ending the call will cause the program to close and the user interface to be seen again. This was accomplished by using an open source UI creation tool called GTK. GTK by itself is only a programming language and to more easily create a UI we employed a program called Glade that allows for drag and drop UI creation while handling most of the back end language creation. Finally we had to link the UI file to the c program that we were creating for the motor system. In the c program we had to initialize the UI by drawing it and then linking the signal of clicking the call button to a function. From there it was simple to define what we wanted the function to perform.

The other component of the UI, is a battery charge reading and will be automatically updated and is meant for informative use only. The charge will be displayed in percent remaining to mimic the system used by cellular phones for easy familiarity. To do this we used I^2C protocol to “talk” to the coulomb counter IC. Since this is a protocol that has been in use since 1982 it was simple to find a library that already had functions built for communicating in this manner. Most items that needed to be defined were simple such as the IC’s address, register address or how long of a data stream to read or write. With the chip manufacturer we chose they already had PCB’s built for use in prototyping and by purchasing one we were able to quickly implement the device into our circuit and all we had to add was a high ohm resistor (10k).

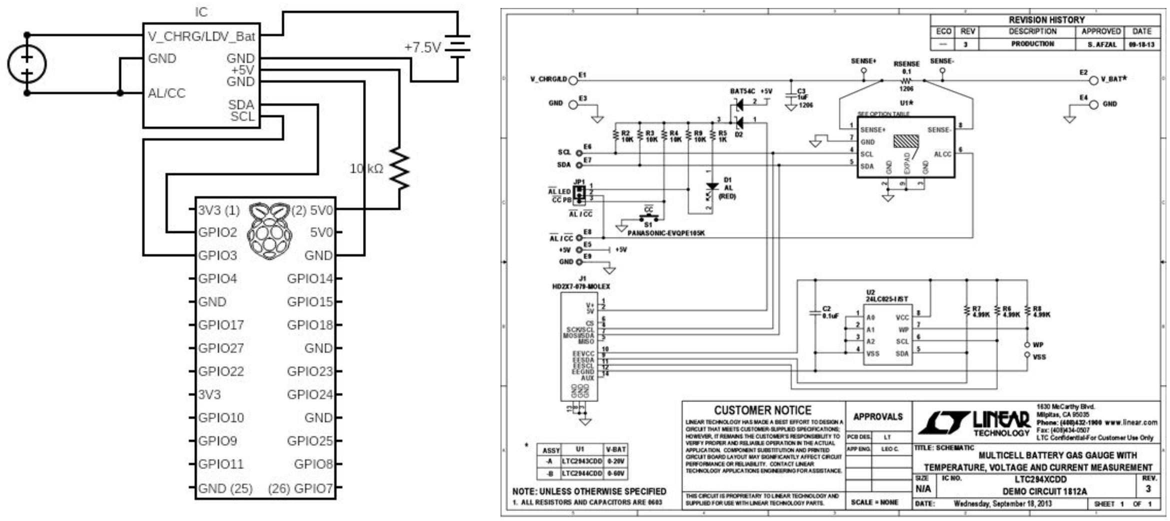


Figure 6 : Left: Coulomb Counter PCB to RPi 4 connections. **Right:** Coulomb counter PCB circuitry.

Our team's biggest challenge was talking between the two different coding languages, python and c. This issue of communication was solved by changing the Tensorflow Lite python code to write area information to an area text file. Then, the C code would output distance information found from the ultrasonic sensors to a distance text file.

<u>User Guide:</u>

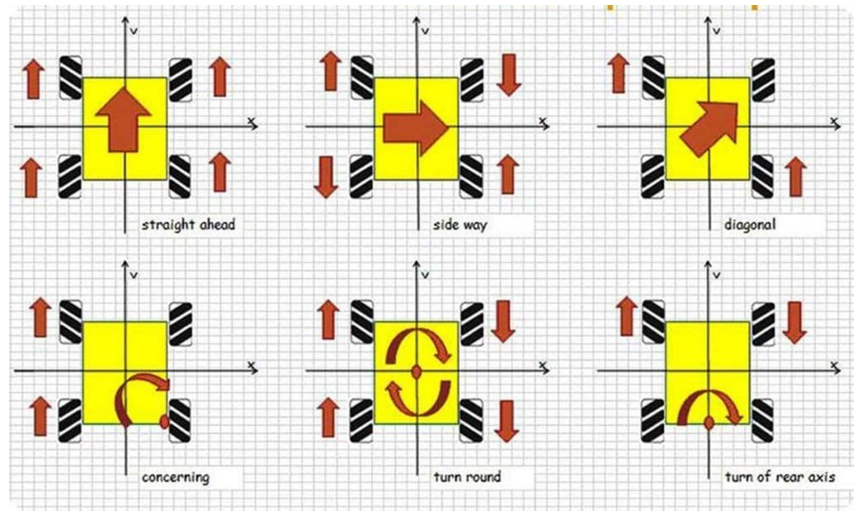
The Ai Cart is simple to operate so there are only three steps. First, to turn on the Ai Cart, the power bank needs to be plugged into the raspberry pi. This is done by connecting the usb-c cable into the Pi, which is located inside the cart. Next, the motors need to be activated by powering on the power module which is located on the back side of the cart,under the basket, where the lcd screen is located. Finally, go in front of the cart that it can follow you and if assistance is needed, click the call button on the screen located at the back of the cart.

VII. Prototype Testing

To test the final (finished) cart prototype, we decided to either build a test course (if applicable) or test the cart through a hallway to see if the cart can actually keep up with the user without crashing into any object or walls. Eventually, we decided to test the cart by walking through a hallway and turning left/right at the end of that hallway to check if the cart can still follow the user without losing them. During the test phase, these are the few issues that we first encountered:

- The cart leans to the left, due to the wheels/motors not being mounted at a straight angle.
- The base of the cart was of a trapezoidal shape, which forced the motors to be mounted at an angle.
- The cart couldn't turn left nor right due to the wheels being mounted at angle and due to grip of the wheels which wouldn't let the cart turn.
- There wasn't enough space between the base and the basket to fit both the power bank and the breadboard.

To fix these issues, first we had to fix the design of the cart's base by making it a square instead of a trapezoid. Then we had to mount the motors and wheels with careful precision to make sure they are mounted at a straight angle; to avoid leaning to the left/right. After that we decided to test the cart with all-directional (Omni-directional) wheels because the regular wheels have too much grip which won't let the cart turn left nor right. Besides, the all-directional wheels are the best option when it comes to our turning/steering method. The figure below shows the shape and functionality of the all-directional wheels:



Last but not least, we decided to increase the space between the base and the basket by printing taller struts. The newer struts are half an inch taller than the original struts which provided us with enough space to fit the power bank, breadboard and the Li-Po battery between the basket and the base.

After fixing all of the previously mentioned issues, the cart appeared to have all of its components functioning properly: The motors/wheels were running without any issues, the sensors were measuring the distances of all objects surrounding the cart with a reading that is only off by 2-3 cm which is accurate enough for the purposes of testing the cart. Also, the camera was working fine, with Raspberry Pi 4 overclocked, the camera reads 4-5 frames per second. The 4-5 FPS might seem slow or not as adequate, however, with this FPS rate the cart recognizes and follows the user when the user is walking slowly, also, keep in mind that the cart isn't at full size yet. And last but not least, the GUI and LCD screen were also functioning correctly. The screen shows the battery level, cart's POV and can make a call by clicking the call button.

After confirming that everything functions properly, we decided now is the time to perform an actual test run. The test run was performed by walking through an empty living room then into a hallway back and forth a couple times. We can confirm that the cart functions correctly if it follows the user from the living room to the hallway and through the hallway back and forth without hitting any obstacle/wall while also keeping an eye on the user. One of the first issues that we've noticed during the first few tests was the camera's angle. The camera needed to be angled to look more upward because the cart wasn't tall enough to see the user's upper body. Therefore, we kept adjusting the camera's angle until we found the perfect spot where the cart can see the user's upper body. Another issue we faced was the radius/distance at which the cart should avoid objects. At first the cart kept on getting way too close to objects before avoiding them. To fix that issue we had to edit/fix the motors.c program. All we had to do is measure the largest radius of the cart when it's turning left or right, then, program that radius/distance to be the minimum distance at which the cart should avoid an object. We found the adequate radius/ distance to be 20 cm. However, this doesn't mean a literal 20 cm radius all around the cart, because the sensors might sense an object at an angle (other than 90 degrees perpendicular) and return a distance that is more than 15cm away when the object is in fact closer. Moreover, after fixing those issues and calibrating the response distance, we tested the cart again through the same path. Yet, this time, the cart avoids walls and almost all objects between it and the cart. Therefore, we came to the conclusion that the cart hits objects that are thinner than the sensors or shorter than the cart. Otherwise the cart functions well and avoids almost all objects. Also, it can definitely avoid walls and big objects (walking through a hallway, the cart avoids hitting the wall almost every time. However, walking through the hallway 10 times, the cart has hit the wall more than once on a single run and on some other run the cart wouldn't hit the walls or any object at all)

Furthermore, the first few tests confirmed that sensors work adequately. However, the camera seemed to have a slightly narrower field of view than what we desired. This means that the cart might lose the user if they were to move far to the left/right really quickly. We tried increasing the resolution of the camera to have a wider field of view but that would also lead to smaller FPS rate which also lead to slower response in terms of recognition. Therefore we decided to keep the “not so wide” field of view since the cart isn’t at full size.

After that, we moved to testing the motors’ speed and capabilities. The first thing we noticed about the motors is that they are slower than before due to the weight of the 3D designed prototype, batteries and actual circuit & components. The current measured speed of the cart is about 0.7 meters per second, which is not that bad since, again, the cart is not at full size yet. As during the test runs, we’ve noticed that the cart recognizes and follows the user with an adequate speed if the user is walking slowly. Also, the calculated weight limit of the cart is about 300 grams since a single DC motor can carry up to 100 grams. Unfortunately, we couldn’t perform a weight limit test because the basket of the cart wasn’t big enough to fit any load other than the Raspberry Pi and all of the wires/cables. We’ve tried our best to wire the cart and hide the wires but, sadly, we didn’t count the space and holes for the wires when we designed the 3D prototype. However, at this point of the testing phase, we came to the conclusion that the cart’s prototype is now finished and is indeed ready for delivery and demonstration. Even though the cart isn’t at full size yet (meaning it’s not fast enough and/or can’t carry an actual grocery load), the cart does indeed functions properly and can indeed recognize and follow the user while also being able to avoid objects/walls and show the battery level & cart’s POV while also being able to make a call through the LCD screen.

VIII. Technical Summary

Many old industries have been updated with AI technologies and our team agreed that the shopping industry needs to have an advancement and the shopping cart are on top of that list. Shopping carts were invented around 1937 and the design has barely changed. The shopping cart is used everyday all around the world and an update is needed. The Ai cart allows users to have a hands-free shopping experience. This will be especially helpful for parents that have kids that love running around the store, the parent will not need to juggle between pushing the cart and looking for their child. The old shopping cart is frequently in the way, making it harder for parents. Finally, the Ai cart is the future of shopping, having a shopping cart that can follow shoppers around the store to pick up groceries will make the job easier for parents and more enjoyable.

IX. Teamwork Discussion

Abdullah's contribution to the project was immense; his knowledge in coding and working with the Raspberry Pi has made him a key member. His desire to get things done quickly and proficiently has made the group have its first (unofficial) prototype done in the first two weeks of class. This made it clear for the group what we want and need for the Ai cart project and what we did not need. His skills in coding helped every team member on tasks that they were stuck on. Abdullah coded in all the sensors of the cart, motors, and made the cart follow the user and avoid obstacles.

Cam's contribution to the project was tremendous; his 3D design skills, coding skills, communication skills, and his strong organizational skills made the group stay on task and made the

Trello board and Discord group very clean and organized. He has done the majority of the 3D design and all the 3D printing, which has made the project look professional. Cam also helped with image recognition so the cart would follow the user.

Eric's contribution to the project was superb; his skills in coding and research made the user interface possible. He did a lot of research to display the amount of battery left in the Ai cart and made it possible for the user to interact with the cart.

Dan's contribution to the project was excellent; his research and coding skills made the Ai cart have image recognition, which helped the cart follow the user. He also has made his prototype to test a different method on turning the cart, by using fewer motors.

Our team worked very well remotely, using tools such as the Gantt chart and Trello, everyone knew what needed to be done, and the work has been divided at the beginning of the course. As a group, when creating the Gantt chart, we have all listed the tasks that needed to be completed, and we broke them down into categories. Next, everyone chose sections that they were interested in and or didn't mind working on. We worked together via Discord by screen sharing or voice calls. As stated above, our team had outstanding communication skills, which made it easy to work together and choose tasks well.

The team has worked very well throughout the entire course; everyone was understanding and was always willing to help, teach, and learn. When someone was stuck, someone was always ready to help. During the start of the course, everyone did a fantastic job sharing their ideas and thoughts about the project.

X. Lessons learned

The first two sprints went very smoothly for our team. We felt our team was far and away ahead of where we wanted to be at that point. However, some challenges arose during the third sprint. These challenges included a “fried” microprocessor, motors that couldn’t carry the weight of the cart, and a redesign of the 3D printed cart. To fix the fried microprocessor, our team was very flexible, with each person buying their own Raspberry Pi, we were able to hand off the project to another team member. After the handoff, that team member was able to get to work and wire the final prototype. After the prototype was assembled, we realized that the motors initially purchased for the prototype could not carry the weight of the cart at a reasonable speed. The motors were an integral part of the cart, as it was our propulsion system. To turn, we relied on slippage, but this heavier weight didn’t allow the wheels to slip. This problem was easily solved because our team had the foresight to brainstorm a solution if this were to happen. We purchased omnidirectional wheels and had our problem solved. Our final issue in the 3rd sprint was the 3D print. While we tried to design the AI cart based on a small grocery cart, we didn’t realize the trapezoidal shape of the grocery cart would raise an issue with our motor mounts. It was pivotal our AI Cart drove straight, so we took the “bottom basket” and switched it from a trapezoid to a rectangle to ensure the motors weren’t mounted at an angle.

If our team were to do anything differently, it would be designing our cart. This is the AI Cart, not the old school grocery cart. Redesigning the AI Cart would allow for a cleaner look. Redesigning the AI Cart would allow our team to create channels for the intrusive wires, integrate the sensors into the body of the cart, and create spaces for the two on-board batteries. In addition to the redesigning of the look of the cart, our team would also use larger motors to carry the cart to desired locations. Larger motors would allow the cart the ability to keep up with the user. Finally, our team would also use a more diverse microcontroller. While the Raspberry Pi was able to fit our functionality on a small scale, it could not meet the functionality on a large scale. Our team was averaging 2-4 frames per second with all of the code running. To achieve better results with the cart, a better CPU and GPU should be integrated to create higher frames per second.

Our team would recommend another group completing this project. If another group were to complete this project, they should start with our lessons learned, so they do not need to make the same mistakes we made. This was a very challenging project but a rewarding experience.

XI. References

Electronics, E., 2019. *How To Run Tensorflow Lite On Raspberry Pi For Object Detection*. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=aimSGOAUI8Y> [Accessed 10 July 2020].

The GTK Team. “The GTK Project - A Free and Open-Source Cross-Platform Widget Toolkit.” *The GTK Team*, 1997, www.gtk.org/.

Henderson, Gordon. “I2C Library.” *Wiring Pi*, 2020, wiringpi.com/reference/i2c-library/.

Analog Devices. “LTC2943.” *LTC2943 Datasheet and Product Info | Analog Devices*, 2020, www.analog.com/en/products/ltc2943.html#product-overview.

Analog Devices. “DC1812A-A.” *DC1812A-A Evaluation Board | Analog Devices*, 2020, www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/dc1812a-a.html#eb-overview.

XII. Appendix

Item	Quantity
3D Printed Items	
HC-SR04 Mount	6
Pi Camera Mount	1
Cart Base	1
Struts	4
Motor Mount Front Left, Back Right	2
Motor Mount Front Right, Back Left	2
Pi Case	1
Cart Basket	1
Touchscreen Case	1
Electronic Items	
Uctronics 3.5 Inch Touch Screen	1
7.4V Lipo Battery	1
Dual Shaft Gear Motor	4
HC-SR04 Ultrasonic Sensor	6
Raspberry Pi Camera	1
Raspberry Pi Quad Core 4GB Ram	1
LTC2943 Multicell Voltage and Current Measurement	1
PowerADD Portable Charger	1

Figure 7: Bill of Materials