
# Race of Doom

DESIGN DOCUMENT

sdmay25-08

Client: Dr Bigelow

Adviser: Dr Bigelow

Team Members:

Alex Crandell – Component Designer

Wesley Jansen – Team Representative

Elizabeth Schmitt – Test Specialist

Ben Towle – Client Interaction Specialist

Lalith Vattyam – Team Manager

Email: sdmay25-08@iastate.edu

Website: sdmay25-08.sd.ece.iastate.edu

Date: 12/7/2024

Version: 10

# Executive Summary

Our project “Race of Doom” is a fun way to bring a strong level of understanding in the autonomous driving space. With how technology increases at a rapid rate, it is important to have a wholistic understanding of software and hardware in an area where not only the efficiency is a concern but also the safety or the public and the major ethical concerns that are consistently at play. Race of Doom is meant to mimic a real-world scenario of driving where, we will be using an RC car go around a track and show to everyone how it’s able to navigate around real-world obstacles that drivers see daily, in a safe and efficient manner.

The track that we will be using will be two lanes wide, and we will have obstacles such as people crossing, railway roads, stop signs, and speed bumps. These are common obstacles that we see every day, and developing efficient, reliable, safe, and protected software allows us to utilize our knowledge from our curriculum to come up with a robust solution. We plan on using HC-SRO4 Ultrasonic Sensors, RP-Lidar sensor, and a RealSense camera to get the most optimal data for precise autonomous driving. So far, we have managed to get all the components of our car connected to our Raspberry Pi 5 microcomputer and we were able to get the car to make basic movements. Our design and requirements have been tailored towards our user archetypes who focus on comfortability, safety, reliability, and smart decision making.

During our advisor meetings and presentations, we were able to get feedback as to what design changes need to be made. This allowed us to tailor our hardware design and software plan to be modular. Currently we finished our goal to get our RC car to start basic movements and have our hardware design laid out on the car. Next semester we will be splitting into teams to compete on the different software practices to see which one is better. We will also be putting in some time to make our track and obstacles to start testing our software next semester.

# Learning Summary

## Development Standards & Practices Used

- IEEE Standard Glossary of Computer Hardware Terminology (610.10)
- ISO/IE/IEEE International Standard – Software and Systems Engineering
- IEEE 2846-2022 – IEEE Standard for Assumptions in Safety-Related models for Automated Driving Systems

## Summary of Requirements

We have 5 main requirements for this first semester of Senior Design. The first requirement is to revise the RC car from previous years to improve usability. We must gather components and elements needed to fit our design. Then, we must SSH and connect to Raspberry Pi 5. We must test all of the components. We also need to start on software for basic movements of the RC car using the Raspberry Pi 5.

## Applicable Courses from Iowa State University Curriculum

CPR E 281

CPR E 288

COM S 311

COM S 252

CPR E 310

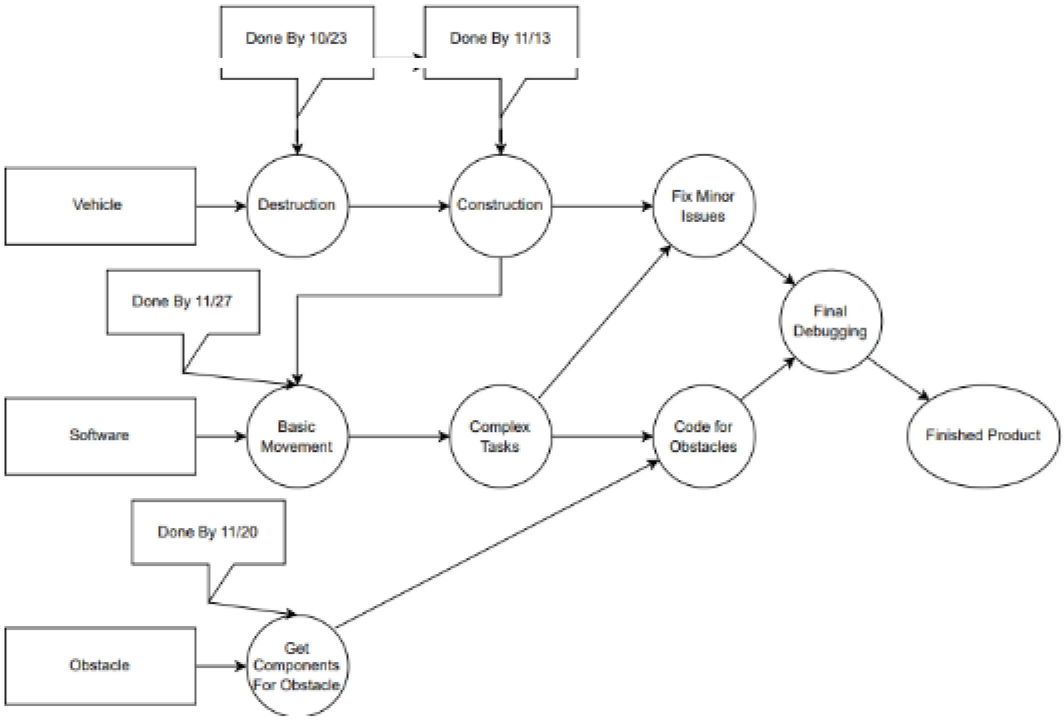
MATH 265 and 267

## New Skills/Knowledge acquired that was not taught in courses

We have been able to practice a lot of new techniques and skills while working on this project. First off, we have had much more practice designing hardware for an embedded system. We have never had so much freedom when it comes to design. We have learned a lot from all this freedom. The team has also learned how to use a Raspberry Pi. None of us have used a Raspberry Pi in the way we must for this project. We have learned to power and SSH on this microcomputer. We have also learned about how to make an electric hat on a Raspberry Pi. This knowledge will be very helpful in the second semester.

# Table of Contents

# List of figures/tables/symbols/definitions



# 1. Introduction

## 1.1. Problem Statement

For our project, “Race of Doom”, we are creating a small-scale model of an autonomous vehicle. The idea is for this model to be used for actual full-sized vehicles. We will be optimizing software and hardware components for the vehicle to be able to move completely autonomously throughout a track with real-world obstacles. This product will help to prevent distracted and unsafe driving. Distracted, tired, or drunk driving is very unsafe. Human error when driving causes many crashes and deaths. The vehicle will move in a way that will be very predictable and safe on the roads with many safety features. Leading to less crashes and human fatality.

## 1.2. Intended Users

We have a large, intended audience. Anyone who wants an autonomous vehicle would be able to use the final full-scale product. Users may want it because they know they get distracted while driving, they get nervous while driving, or they might want to sleep on a long car ride. Everyone on the road will be affected by our design if it is put into full-sized vehicles. Even those who do not drive these cars will have to drive on the same road as those who do.

One potential user is the “Tech Guru”. This user wants the newest, most cutting-edge thing. They also prioritize efficiency and processing power of the design. This person needs the design to be very modern and up to date. They need the best sensors, with many software updates being pushed up to keep everything modern. This user can benefit from an autonomous vehicle because it is the most intelligent, cutting-edge thing in cars. This connects to our overarching goal of getting these vehicles that are less dangerous on the road.

A second potential user is the “Worried Driver”. This user has safety and reliability as their number one concern. We need to make sure that our design has very little room for error. This person is a very important user for us. We need to consider the safety needs above all else, making sure our design can adapt to unexpected behavior on the road. The worriers can benefit from our car design because they may be nervous to drive on their own. A nervous driver can act unpredictably on the road. This driver could use the autonomous vehicle to help them drive more safely.

A third potential user is the “Incompetent Driver”. This is someone who cannot make precise decisions under pressure while driving. This could be a user who simply does not enjoy driving, someone who is tired, a new driver, or someone who cannot drive well in their current state. The common thread between these users is that they need to get someone but cannot do it safely on their own. They need a car that can drive for them. They will benefit from an autonomous vehicle. This car will get them to their destination without help from the user. The incompetent driver connects to our overall goal of getting people around in a timely and safe manner.

# 2. Requirements, Constraints, And Standards

## 2.1. Requirements & Constraints

We have several vehicle requirements and constraints that fit into various categories. These requirements and constraints are listed below in different categories.

**Functional Requirements:**

- The vehicle can move in every direction.
- The vehicle moves completely autonomously. (**constraint**)
- The vehicle has a nerf gun that can shoot at an enemy.

**Team Requirements:**

- The 2 teams use the same vehicle. (**constraint**)
- The 2 teams use the same basic code for movement. (**constraint**)
- Each team writes their own logic off of the basic movement program.

**User Experience Requirements:**

- The vehicle slows to a stop instead of stopping very suddenly.
- The vehicle stays within the track.
- The vehicle moves about the track without hitting any walls or "pedestrians".
- The vehicle stops when it sees a stop sign.
- The vehicle slows down at a yield.

**Monetary Constraint:**

- The budget is $500 for all sensors and hardware used.

## 2.2. ENGINEERING STANDARDS

It might be a big surprise to some people, but engineering standards are an important part of everyday living. Technology is such a huge part of every person's day to day life. The IEEE standards make sure that people have positive experiences with the technology that is so engrained in their everyday lives. Without these standards many items would not be quality enough for users to want to interact with them. An example is the quality of speakers. People are constantly listening to music, whether through tiny or big speakers, within the IEEE standards these speakers could be very staticky, creating a negative experience for many people.

These engineering standards are also very important for our race of doom projects. Below are three IEEE standards that are important to keep in mind as we continue the development of our project.

**IEEE Standard Glossary of Computer Hardware Terminology (610.10)**

This IEEE standard provides us constraints on what we refer to different signals as. These naming conventions are mostly constrained to their respective purpose. Therefore, this IEEE Standard helps us to keep terminology consistent and easy to understand by a tea, of individuals.

This standard holds a great deal of relevance to our project. It will mostly impact the way we document our design. It will be most helpful for previous groups to look at our naming conventions and that will result in less confusion overall.

**ISO/IE/IEEE International Standard – Software and systems engineering – Software testing – Part 2: Test processed**

This IEEE Standard helps developers understand the test process specifications of a software system. These test processes include unit, integration, and regression testing. Overall, this standard gives software developers an idea for what the lifecycle of the software testing process is. This will be especially helpful for us, because our design will demand rigorous testing and trials to ensure that our vehicles autonomous design is reliable and efficient.

This standard has major relevance to our team’s testing process. While we have yet to see these impacts yet, it helps us to plan out the testing phase of our design from a project management point of view.

**IEEE 2846-2022 - IEEE Standard for Assumptions in Safety-Related Models for Automated Driving Systems**

The purpose of this IEEE standard is to specify the minimum assumptions needed to ensure the safety and reliability of autonomous vehicles. Such include what it means for an autonomous car to “Drive Safely”. It is important for us to be on the same page as a team by making sure we follow this standard.

This standard is more applied to actual autonomous cars used on real roads. However, it has some relevance to us since our R.C car simulates a safe driving environment. This is because a lot of our objectives are to stop at stop signs, yield to pedestrians, and avoid hazardous objects.

As a team, we also considered a few other IEEE standards. One of these was IEEE P3412: Standard for Autonomous Driving Architecture. This standard covers the full lifecycle of an autonomous vehicle. It gives important standards for the architecture of autonomous driving. This can be important for us to look at in order to make architectural decisions in the future. It is not one of our main standards since it is talking more about the whole life of a full-sized autonomous vehicle, while we are working on a small prototype.

Another standard that was found was IEEE 7001-2021: IEEE Standard for Transparency of Autonomous Systems. This standard talks about being transparent and testing autonomous systems in an objective way. This will be important for us to do. We need to have objective tests in order to grade each of the team's vehicles fairly, since it is a race. This is a good standard to consider, but we did not think it was as important as those listed above.

In our project we will be implementing the IEEE standards by following a standardized naming convention for the different signals which will allow our software to be acceptable, readable, and understandable by teams.

We also will adhere to the second and third IEEE standard listed above by having a well thought out testing process that will be written on our documents that will not only give our peers but possible future users the scale, scope, and risk of the project. By having this standardized testing process, we get to show the reliability of our software and how the testing that is done gives credibility to the safety of autonomous vehicles as that's a primary concern many people have.

# 3 Project Plan

## 3.1 Project Management/Tracking Procedures

For Race of Doom our team is adopting the agile project management style. Agile is an adaptive workflow style. This works well for our team because we do not know everything upfront. For example, we are not sure if the processing power of a Raspberry Pi is enough. We may need to use Bluetooth to connect the Raspberry Pi to a computer to do data processing. This is something that we will not know until the hardware is all together and we will be able to test and analyze the speed. This is just one example of something that we will not know at the moment, making an agile workflow advantageous. Our goals for the project are not completely clear. We are unsure of what the obstacles will be at this point, depending on recourses and certain limitations it may change. Agile is great for an ever-changing project like Race of Doom.

Our team is using Discord to communicate with one another. This is an easy communication tool to keep us connected. Git will be used to keep track of all our code this and next semester. We are keeping our documentation in a CyBox. CyBox makes collaborating and sharing documents easy.

## 3.2 Task Decomposition

For the Race of Doom project, we have a timeline to complete large tasks that can be split up into smaller/more manageable tasks. For example, one large task we must complete is assembling the hardware design for the R.C. car. This task can be divided into several smaller tasks that can be completed in the span of a few days/weeks. For example, setting up the HC-SRO4 Ultrasonic sensors with the Raspberry Pi processor board is a smaller task that adds to the milestone of completing the task of R.C car assembly. The following tasks can be provided as follows…

I. Disassembly of previous R.C car design - Completed on 10/23

II. Assemble the new HW Design of R.C car - To be Completed on 11/13

   a. Test the connection and data visualization of RealSense D455 Webcam to determine the best place to mount it onto the Car

   b. Repeat task a for the RPLidar Sensor

   c. Repeat task a for the HC-SRO4 Ultrasonic sensors

   d. Mount the needed sensors onto the car after determining the most optimal placements of each sensor

   e. Connect each sensor to the appropriate Raspberry Pi 5 pins using a breadboard and jumper wires

III. Retrieve the needed components to implement our Race course obstacles - To be completed on 11/20

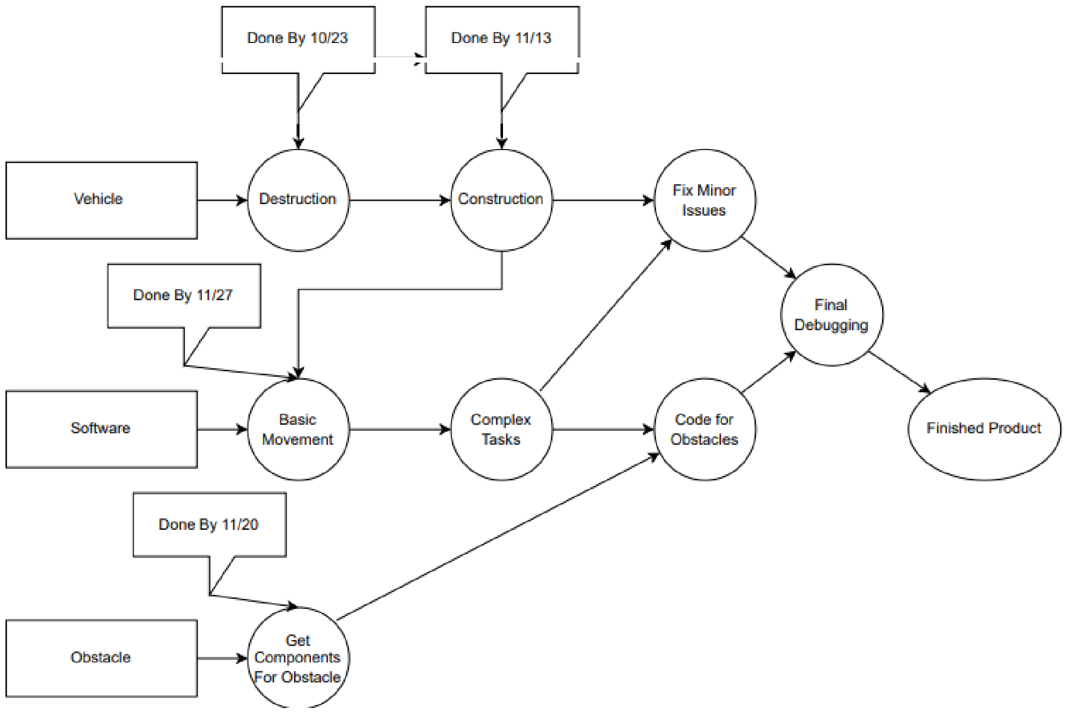
   a. Design our obstacle course

   b. Figure out what components are needed for each obstacle

IV. Implement basic autonomous movement of R.C car - To be completed on 11/27

   a. Install the necessary libraries and SDKs for Raspberry Pi, Camera, and RPLidar Sensor

   b. Construct a rough software design plan to facilitate communication and connections between our central processing unit, and external components/sensors.




## 3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

The key milestones for this semester for the Race of Doom project are deconstruction, construction, retrieving components for obstacles, and basic movements. These can change because of the Agile development process.

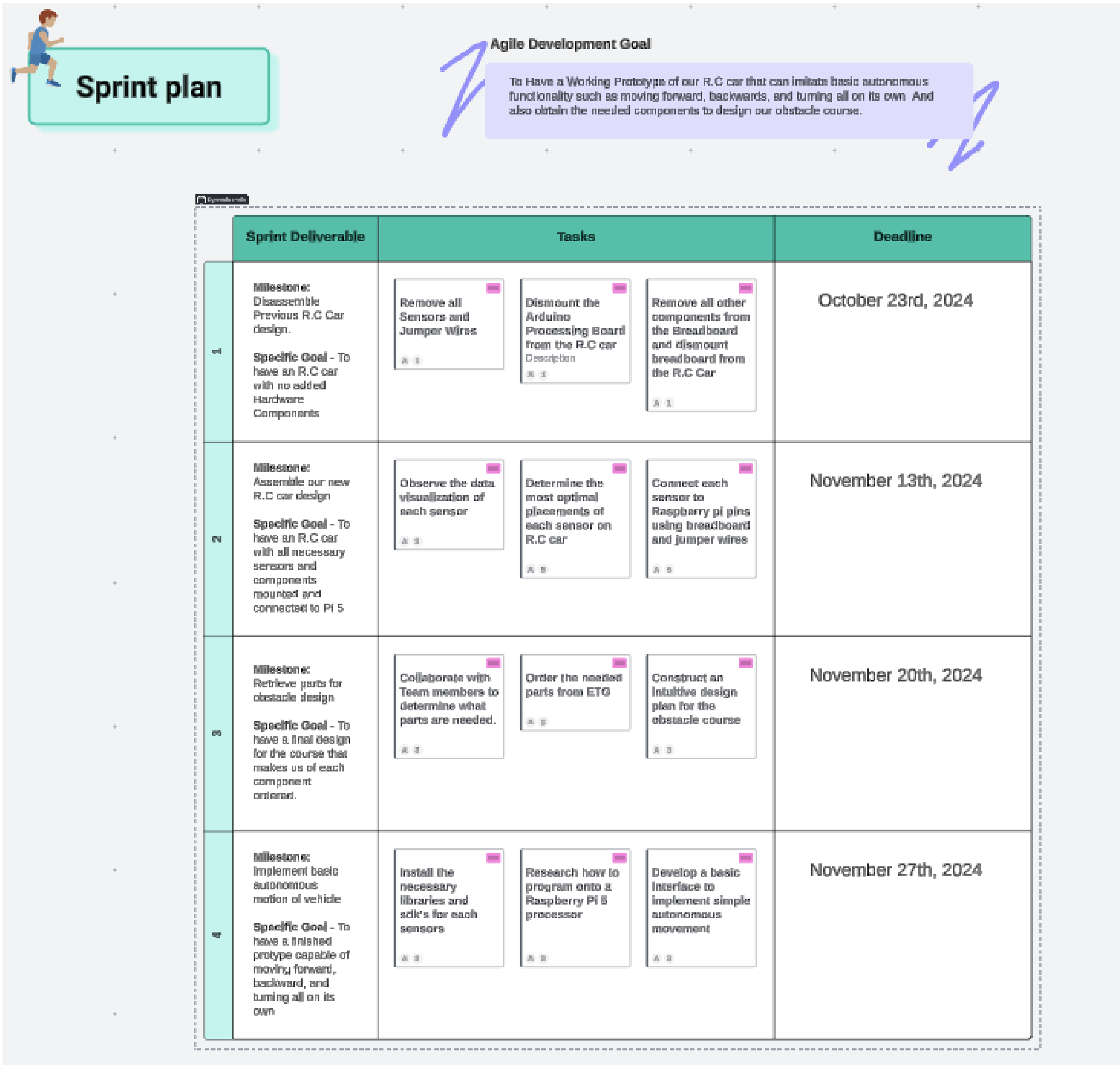
The first milestone is deconstruction, this milestone involves deconstructing the car from last year so that it can be built back up again. The first task of deconstruction involves testing the old team's hardware to see if the car can move using it. The metric to know if we got it done is, if we turned it on and got it to move at all. This metric basically tells us whether the hardware still works to move the car in anyway. We can base some of our design off of their hardware based off of this. The second task involved in this milestone is taking the car apart. The metric for this is that only the main shell of the car remains. There are no wires, circuit boards, or batteries that did not come with the car. Taking the car apart is important to get to the bones of the car and see how everything works.

The second milestone is construction, which involves a lot of planning and assembly. The first task for this milestone is research. We need to research and compare the different processors and sensors to use for our design. The measurable output for this task is a list of at least 2 potential processors and 3 potential sensors. The next task is to create a hardware design diagram based on the research that was done. The diagram must include the sensors and processor we chose and how they are all connected. The diagram is the measurable progress for this task. The next task is to mount the processor. The Raspberry Pi must be mounted onto the car for this step. The metric to determine this is done is having the Raspberry Pi mounted to the middle of the car. Then we must test the connection and data visualization of RealSense D455 Webcam to determine the best place to mount it onto the Car. The output for this task is the RealSense D455 to be mounted onto the car in the optimal location. This task is repeated for the RPLidar Sensor and the HC-SRO4 sensor. The last task is to connect each sensor to the appropriate Raspberry Pi 5 input.

The Third key milestone for our project is to retrieve components for obstacles. The first task for this is to decide what obstacles would be used for the course. Once we have made a final decision on all our obstacles, we will create a document to write down all of our obstacles with visuals to help understanding. This will be how we ensure that we are finished with the first task. The second task is to design our obstacle course. The deliverable for this task will be a diagram of the course. After this design is created the last task is to find the components needed to create the obstacle. The components needed must be found and ordered if needed. All the components must be either obtained or ordered for this milestone to be considered finished.

The last key milestone for Race of Doom this semester is to implement basic autonomous movement for the vehicle. The first task for this milestone is to install the necessary libraries and SDKs for Raspberry Pi, Camera, and RPLidar Sensor. All the libraries and SDKs are needed to be able to code on the Raspberry Pi5. These must be installed on all our computers for the task to be considered done. The final task is to construct a rough software design plan to facilitate communication and connections between our central processing unit, and external components/sensors. The basic code that is created must make the car move autonomously. If it cannot move three feet forward, backward, right, and left this milestone will not be considered done.

### 3.4 Project Timeline/Schedule




| | Sprint Deliverable | Tasks | | | Deadline |
|---|---|---|---|---|---|
| 1 | **Milestone:** Disassemble Previous R.C Car design. **Specific Goal** - To have an R.C car with no added Hardware Components | Remove all Sensors and Jumper Wires | Dismount the Arduino Processing Board from the R.C car Description | Remove all other components from the Breadboard and dismount breadboard from the R.C Car | October 23rd, 2024 |
| 2 | **Milestone:** Assemble our new R.C car design **Specific Goal** - To have an R.C car with all necessary sensors and components mounted and connected to Pi 5 | Observe the data visualization of each sensor | Determine the most optimal placements of each sensor on R.C car | Connect each sensor to Raspberry pi pins using breadboard and jumper wires | November 13th, 2024 |
| 3 | **Milestone:** Retrieve parts for obstacle design **Specific Goal** - To have a final design for the course that makes us of each component ordered. | Collaborate with Team members to determine what parts are needed. | Order the needed parts from ETG | Construct an intuitive design plan for the obstacle course | November 20th, 2024 |
| 4 | **Milestone:** Implement basic autonomous motion of vehicle **Specific Goal** - To have a finished protype capable of moving forward, backward, and turning all on its own | Install the necessary libraries and sdk's for each sensors | Research how to program onto a Raspberry Pi 5 processor | Develop a basic interface to implement simple autonomous movement | November 27th, 2024 |

### 3.5 Risks and Risk Management/Mitigation

All the tasks in the first milestone of Race of Doom, disassembly of the car, have very similar risk involved in each of them. This milestone involves handling a lot of the hardware components. There is a risk of breaking components for these tasks. Some of the pieces, such as the wires, are fragile. Because of this the risk of breaking components while disassembling the car are high. The probably of breaking something is probably around 15%. The good news is the parts that are more fragile are also easier to replace, like the wires. These components are cheap, and we have a lot of them to use if needed. The most important thing for these tasks is to be very gentile when taking everything apart. It is important not to yank things and to think before just pulling. Being mindful of our actions will reduce the risk of broken components by a lot.

For the second milestone while constructing the car there are many risks involved. The most prominent is the risk of short circuits. We must make sure before turning anything on that we know what it is doing and that it will work. If we do have a short circuit, it could just simply ruin a part or fry our battery forcing us into a large setback. Also, while constructing the car there are connectivity risks. We are not sure exactly how different components will work together so trying to force things could be a waste of time or a huge headache. Finally, the physical building of the car and the layout of the sensors comes with the risk of things not being secure. Once the car starts rolling, we do not want things to fly off the car, so this is a definite risk. We also want to have sensors facing every way that we want. The overall risk for this milestone is high and I would assume running into some sort of setback is almost guaranteed at about 90%. This is because no project goes perfectly, and you learn most from just trying things out. It is very important to try to mitigate this risk. Firstly, it is most important to try to prevent short circuiting. Carefully designing the hardware using diagrams and checking these designs thoroughly is vital. When assembling these designs, we must follow the diagrams carefully, making sure everything it turned off before connecting any components. Triply checking the hardware matches the diagram exactly before turning anything on is the last step. Once everything is working properly, they must be attached to the vehicle very securely. Testing movements need to be slow at first, to make sure none of the components fall off or break.

Retrieving parts of the obstacle design is milestone 3. The main risk for the first task of this milestone is being overly optimistic or pessimistic about what our vehicles will be able to do. Since we are not exactly sure how the hardware and software will work yet for our design, the risk of not having a perfect understanding of things that the car does is high. The risk of designing obstacles that are too simple or too difficult is high, probably about 83% chance that this will happen. Knowing this, we need to design obstacles that are adaptable. The components that we order should be able to be used for more than one thing. This way the challenges can either increase or decrease in difficulty depending on how things are going. The second task involves ordering the determined parts from ETG. The risk here is possibly ordering the wrong parts. This can be prevented by double checking the order before placing it. With this check the probability of this happening is low at about 18% chance. If this happens and we need some different components, we can try to look at local stores to prevent more miscommunications. When we have the parts the last task of the course design can happen. Creating the whole course without having the whole picture of what the vehicles will be like will be tricky. Like for the first task, the probability of us estimating our own abilities wrong is high. The chance is at about 76% that this will happen. The design we make for the truck will be subject to change. It would also be good for us to design multiple courses, some easier than others. This will make it easier for us later.

The fourth milestone of implementing basic autonomous movements on the vehicle is done in software. This makes it less risky in general. Mistakes on software are typically much easier to fix than on hardware. The first task of installing different libraries and SDKs may be frustrating. It is often more difficult to install software than it feels like it should be. The risk of installing things incorrectly is high at about 58% chance. Installing something wrong is easy to fix. This part of the project may take some trial and error, but the risk of things going wrong in a way that is unfixable is very low. The second task is to research. The only risk involved in research is spending too much time doing it. The risk of spending too much time researching how to program onto a Raspberry Pi 5 is at about 60%. I think it is important to set an amount of time to investigate this and make sure not to go over that time. It is okay to learn by doing it in this instant because the probability of software being so messed up that it cannot be undone is very small. That ties in well to the last task of developing a basic interface for movement of the car. The risk of messing up software too bad is very low, probably about a 14% chance. This part of the project will not be too risky.

## 3.6 Personnel Effort Requirements

| Task | Alex | Lizzy | Wesley | Lalith | Ben |
|---|---|---|---|---|---|
| Disassembly | 6 hours | 1 hour | 1 hour | 6 hours | 1 hour |
| Assembly | 15 hours | 15 hours | 13 hours | 12 hours | 14 hours |
| Gather Parts for Obstacles | 4 hours | 4 hours | 6 hours | 3 hours | 5 hours |
| Basic Autonomous Motion | 10 hours | 15 hours | 15 hours | 14 hours | 15 hours |

This table shows the number of hours each team member has worked (or plans on working) leading up to the end of the semester. The future tasks are an estimation of how long we plan on working on each milestone and are subject to change once we get into completing these tasks.

## 3.7 Other Resource Requirements

**Hardware Resources**

1. RPLidar Sensor
2. RealSense D455 Depth Camera
3. Raspberry Pi 5
4. HC-SRO4 Ultrasonic Sensors
5. Breadboard
6. Electronic Speed Controller (ESC)
7. Servo
8. Motor
9. 8.4 V Battery

**Software Resources**

1. PyQT – For processing and running scripts compatible with Raspberry Pi
2. RPLidar SDK – For processing data gathered from RPLidar Sensor
3. PyRealSense– For gathering and interpreting Realsense Camera data in Python

# 4 Design

## 4.1 Design Context

### 4.1.1 Broader Context

It is important that we think of the broader context when designing this autonomous vehicle. Autonomous vehicles can drastically change the way people all over the world are able to live their lives. We are designing for communities that have the infrastructure for autonomous vehicles. Larger cities are a good start, but these cars affect everyone. From workers that will manufacture them to those that will own them. Below is a chart that addresses some important concerns and considerations for a large-scale version of the car that we are designing.

| **Area** | **Description** | **Examples** |
|---|---|---|
| Public health, safety, and welfare | Drivers and passengers will benefit from advanced safety features of the autonomous vehicle. The cars ability to yield at yield signs, stop at stops, and avoid obstacles. | Reducing safety risks<br><br>Preventing distracted and unsafe driving<br><br>Promoting safe driving practices |
| Global, cultural, and social | Communities value their members safety and our project will reduce risk of accidents. Adaptability is important in cultures and this project adapts to an ever-advancing society. | Community safety<br><br>Ethical development<br><br>Encourage technological advancements |
| Environmental | We must use energy efficiently with the Pi, which is a low-power component. The vehicle will reduce emissions by being programmed to drive efficiently. | Sustainable materials<br><br>Low-power component: Raspberry Pi<br><br>Energy consumption |
| Economic | The final product must be affordable with affordable components. The product will create more jobs. | Reduction in accident cost<br><br>Job opportunities<br><br>Cost-effective solutions |

### 4.1.2 Prior Work/Solutions

Many groups and companies have been working on autonomous vehicles for some time now. From lane assist to taxis that full on drive themselves. There are varying levels of autonomy. *Supeed Pasricha* has done research on optimal design for autonomous vehicles. This research shows that it is important to have sensors all around the car for things like parking assist, blind spot detection, and street sign recognition [5]. This is like our design in the fact that we will have sensors in the front, back, and on the sides of the car. We think that it is very important to collect all the data that we can to get a accurate read of the cars surroundings.

A research paper was written on the different levels of autonomy and what sensors are used for these levels. This paper talks about a combination of LiDAR, Radar, and camera sensors on the vehicle. The camera is needed in the front, overlapping with other sensors that can process things faster. Other sensors that are quicker are used in other areas around the car as well [3]. This research makes an excellent point about using many different types of sensors that serve different purposes. Our design includes camera and lidar sensors. We also include one other type, the HC-SRO4 Ultrasonic Sensors. This ping sensor is able to sense and process objects very efficiently. The values from it can also be cross referenced with other sensors to lower risk. We feel that this sensor is a great addition to just make our design stand out even more.

This is not the first year that a senior design group has done the race of doom project. In 2024 there was another group that had the same goal in mind. We were given their hardware and software to be able to rework their design. They used the RC car with an Arduino for processing. They used ping sensors to race through obstacles [3]. We chose to keep their RC car as an amazing base. The Ultrasonic Sensors are also being used for our design as quick processing information. One way we chose to differ from them is by having our computers process the information. This way we can store much more data. That way we can have the car do more things without being restricted by the processing power of the Arduino. We found that a Raspberry Pi has more Bluetooth connectivity than the Arduino, so we chose to replace the processing unit. Since we have more storage to work with now, we are also able to use more sensors.

### 4.1.3 Technical Complexity

Our Race of Doom project consists of many complex components. The Raspberry Pi 5 is the brain of everything that our vehicle does. We are required to program this Pi through SSH, while writing all our own code to it. This is a little but powerful machine with many libraries that we must learn to understand. We also must connect this processor to the motor and sensors. We must connect the Pi in a way that allows the car to move in the desired direction given these sensors outputs. The code on it will need to understand all the sensor information, process it, and adjust the car's positioning accordingly. The other components are sensors. The HC-SRO4 Ultrasonic and RPLidar Sensors both have a different set of outputs. We will need to find equations for each of these outputs that indicate the data in a readable format. The camera will require photo processing. This includes being able to detect different colors and shapes. A very complex undertaking that will be new to all of us.

The problem scope includes many different challenges on the racetrack. The industry standard for autonomous vehicles right now includes blind spot detection and backup cameras. Our track will have obstacles including street sign detection. This is photo processing, finding and avoiding walls, yielding, and changing lanes. This far exceeds what the average vehicle can do autonomously right now.

## 4.2 Design Exploration

### 4.2.1 Design Decisions

1. Placement of the sensors

    a. We made a key design decision this week by determining where each sensor should be placed. We decided to place 2 ultrasonic sensors on the peripheral of the car, the RealSense Webcam will be placed at the front for image processing, and the RPLidar sensor will be placed at the very top of the vehicle.

2. GPIO pin functionality

    a. When creating our hardware design, a lot of important decisions to make involve which GPIO pins on the Raspberry Pi should connect to which sensor. The decisions made were based on where each sensor needs power distributed to it, as well as which pins are easiest to read/write from.

3. Communication between P.C and Raspberry Pi

    a. There are different ways to establish communication between where we write our code (PC), and the Raspberry Pi. Some options include communication through Bluetooth, WiFi, and SSH. We decided to use SSH to communicate between the Pi and the PC because that is the most secure and efficient way to communicate.

.

### 4.2.2 Ideation

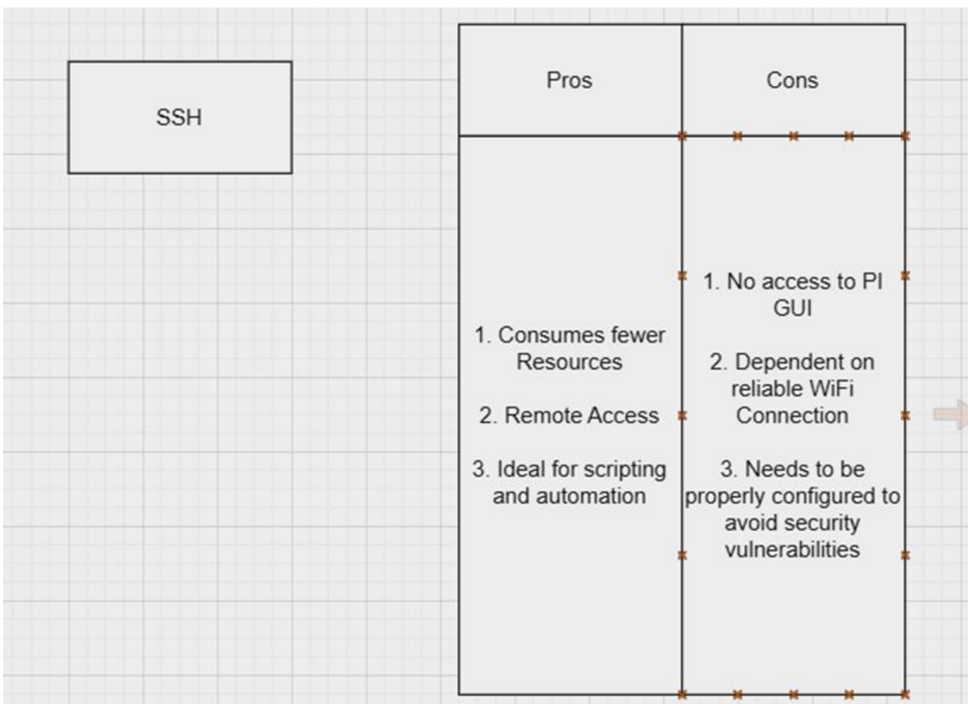
For the decision about communication between the P.C and Raspberry Pi, we had a lot of options to consider for connecting our program onto the Raspberry Pi. The thought process that went into this design decision involves how secure the connection is, how performance and speed will be affected, and how convenient or easy it is for us developers to implement the connection. Here are 5 options that we considered...

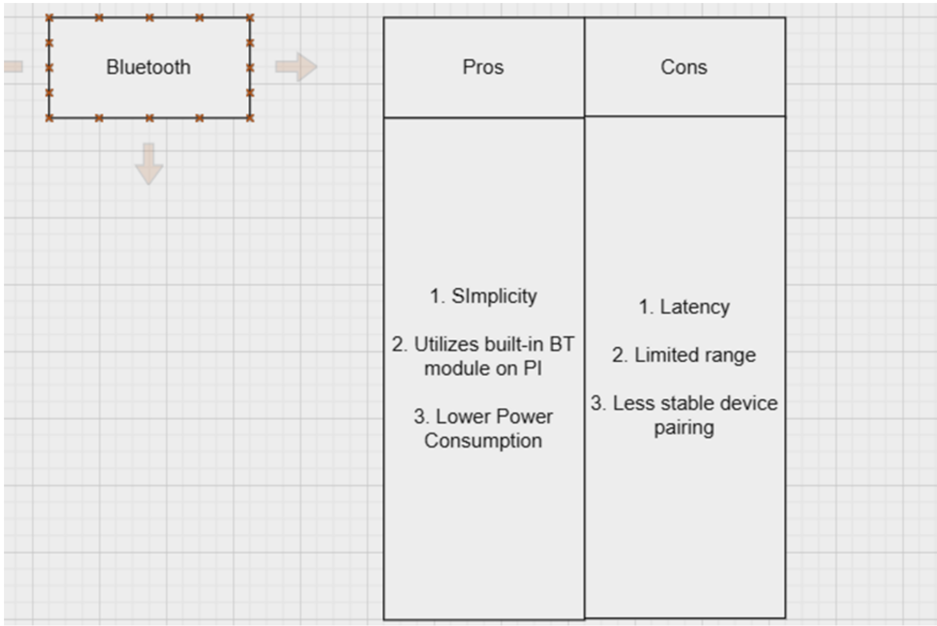
1. SSH - (Secure Shell)
2. Wireless Connection over bluetooth
3. File Transfer using SCP (Secure Copy Protocol)
4. Wired connection with USB or HDMI cable

5. Remote desktop connection (VNC)

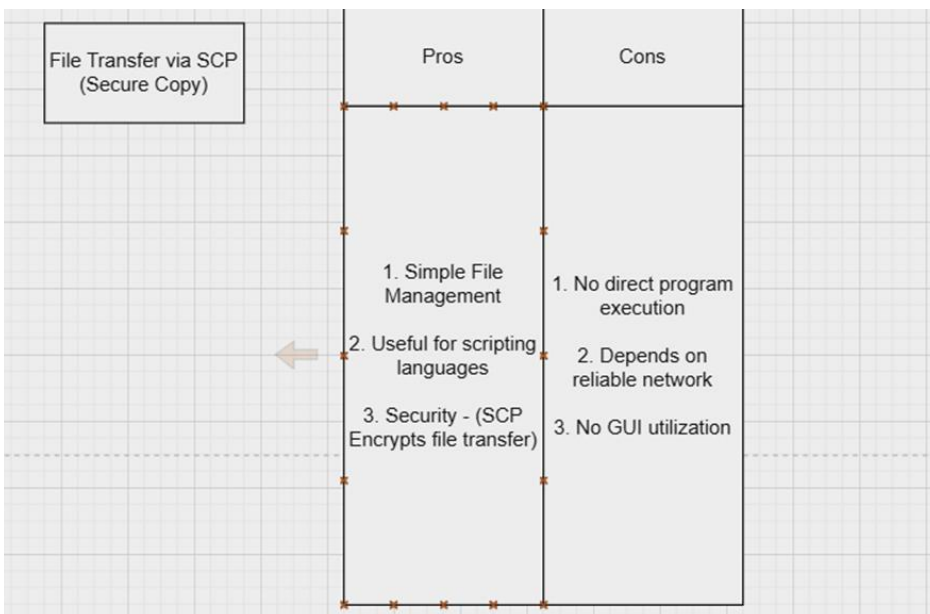
### 4.2.3 DECISION-MAKING AND TRADE-OFF

We created Pros and Cons lists for each of the ideas for the connection between the P.C. and the Raspberry Pi 5. Below you can see the charts we made in order to make this decision. We decided that the scripting and automation abilities as well as remote access were 2 aspects of the connection that were high priority for our project. We also have reliable WiFi here at Iowa State and having a PI GUI was not that important to us. These considerations made SSH an obvious choice in the end. We sacrificed some speed and user-friendly aspects for the sake of remote access. We can access the Pi information on our induvial laptops. This is a huge plus. SSH became an obvious choice after these charts were made.





Bluetooth
Pros
Cons
1. SImplicity
2. Utilizes built-in BT module on PI
3. Lower Power Consumption
1. Latency
2. Limited range
3. Less stable device pairing

File Transfer via SCP (Secure Copy)
Pros
Cons
1. Simple File Management
2. Useful for scripting languages
3. Security - (SCP Encrypts file transfer)
1. No direct program execution
2. Depends on reliable network
3. No GUI utilization

**Wired Connection**

| Pros | Cons |
| --- | --- |
| 1. Immediate/Direct Access<br>2. Less Latency<br>3. Works offline | 1. Not convenient in practice<br>2. No remote access<br>3. Requires a HW setup |

**VNC**

| Pros | Cons |
| --- | --- |
| 1. Full access to Pi GUI<br>2. Allows for remote acess<br>3. Easier configuration | 1. Consumes more resources (CPU/Memory)<br>2. Dependent on reliable network<br>3. Higher latency |

## 4.3 Proposed Design

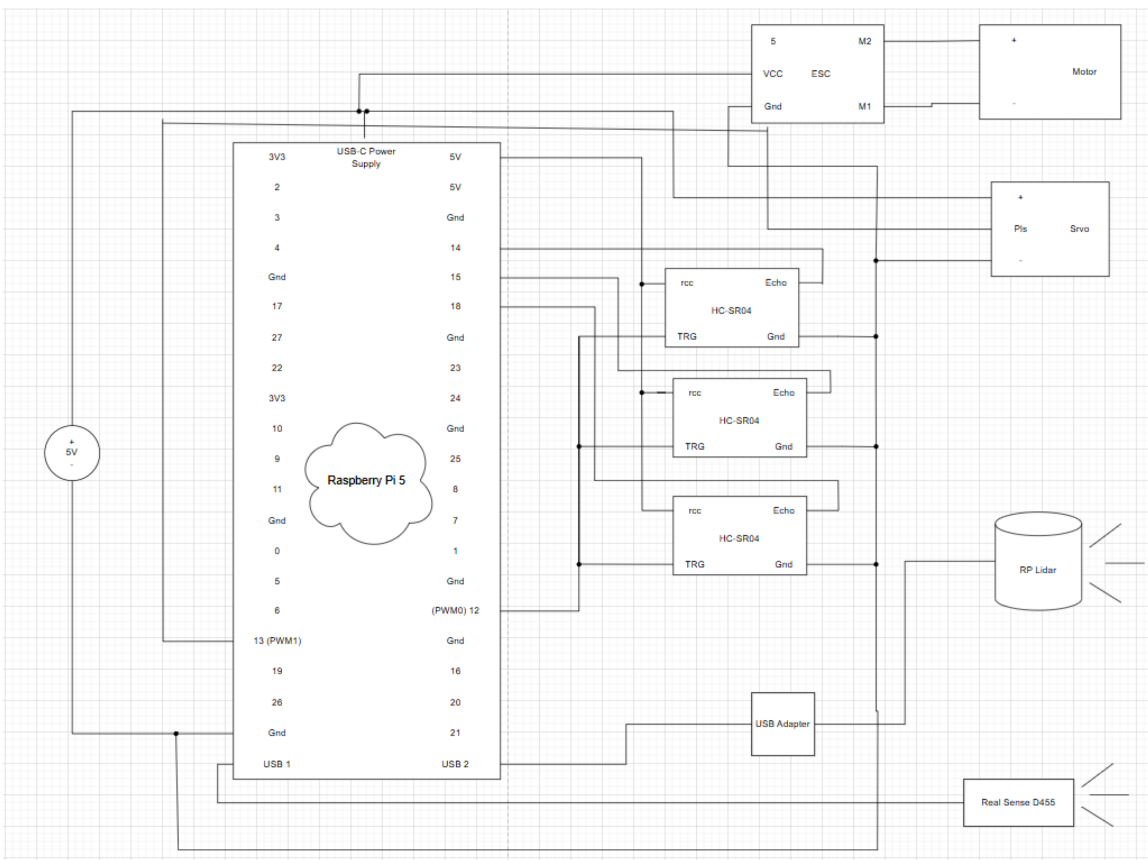
### 4.3.1 Overview

The design of our R.C car begins with our Hardware design layer and mechanical components of the Traxxas Remote Controlled Car. Since for this project, we will have no need for the remote, we will instead use a series of small, programmable computers integrated onto one single board called a “Raspberry Pi” This will allow our software layer (i.e. the code responsible for autonomous movement) to control the car and steer it away from hazardous obstacles.

In addition to the Raspberry Pi, our system design includes the utilization of sensors and cameras to help the car know where it is, and what objects stand in the way. These sensors are connected to the Raspberry Pi so that our software knows what driving decision to make based on the input of the sensors.

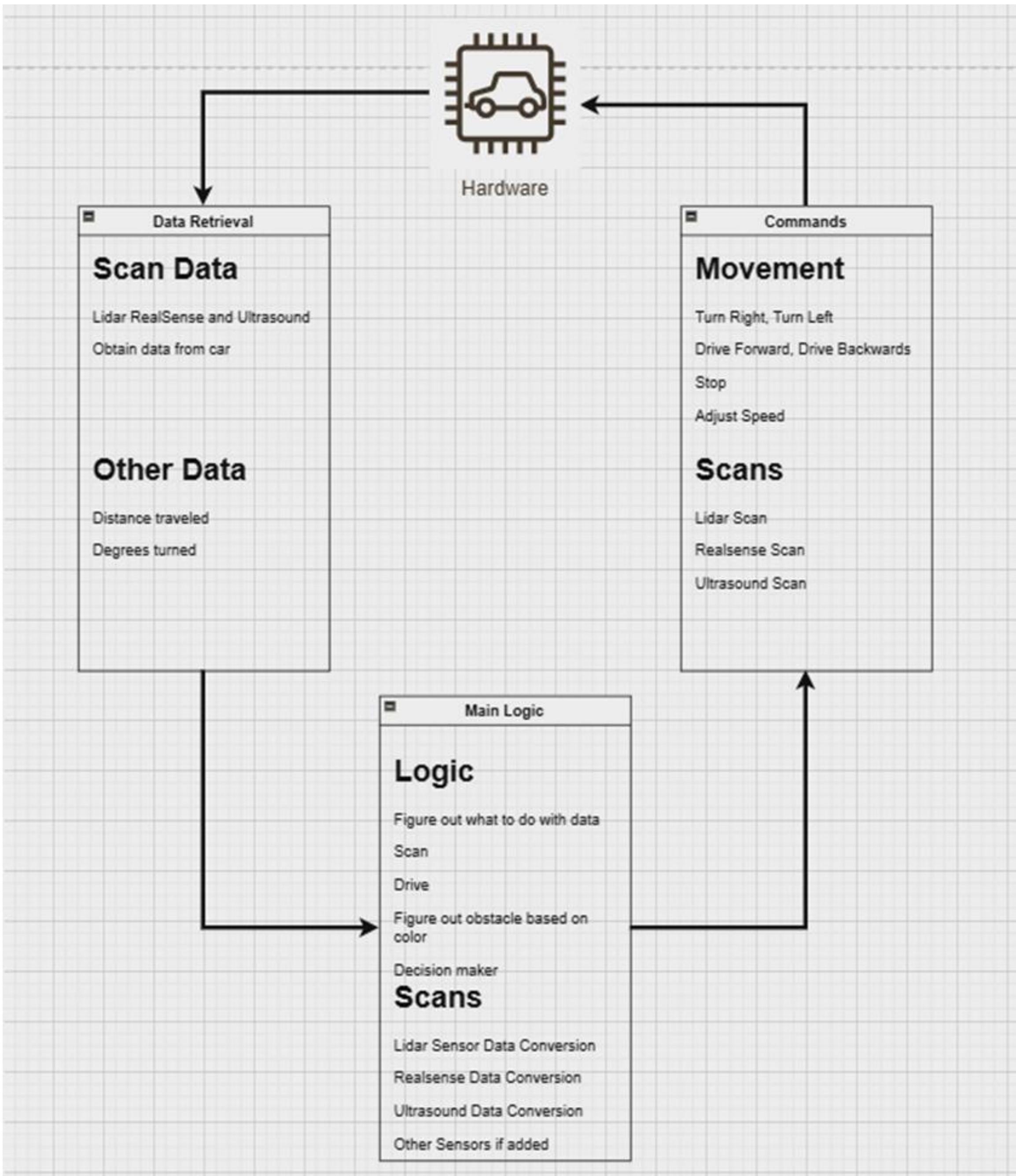
### 4.3.2 Detailed Design and Visuals

**Hardware**




*This schematic Diagram Highlights our hardware implementation for the system. The central operation center of our system is in the Rasberry Pi 5 Processing board. This is meant to direct incoming and outgoing signals from our sensors and distribute power to devices like our servo, motor, and webcam. The Rasberry Pi also comes with USB input and output ports to help us gather sensor information from the camera, RPLidar, and Ultrasound sensors. Some concerns that we may run into with the design include how much power we have supplied to each device, since the webcam and RPLidar sensors consume a significant amount of power supplied by the processor.

**Software**



Here is a very basic and high-level software design. The basic idea is that the car will send data to our computer or Raspberry Pi, which will retrieve the data from the car and the sensors mounted on the car. It will then direct the data to where it will be processed in the main logic. The sensors will convert the raw data they produce and turn it into readable and understandable numbers. All the data we retrieve will go to the main logic where the decisions are made. The decisions will be through logic statements trying to decide on the next best course of action. The decision will be sent to the commands, ultimately telling the car what to do next. This will tell the car to scan, drive or whatever the main logic decides needs to happen.

### 4.3.3 Functionality

Since the actual scope of our project does not have a lot of use in the real-world (unless people race R.C cars for a living), the applications of autonomous vehicle software is revolutionizing modern-day technology. If we look into the scope of real-world autonomous cars, it is clear to see that these applications are intended to aid a user's driving experience. As we know, not all people in the real world are great drivers, so designs like these are intended to reduce the risk of human error on the road. The sensors and components on an autonomous vehicle can help prevent collisions either with other cars, or objects/pedestrians on the road. That being said, our Race of Doom project aims to simulate the real-world driving experience on a smaller scale so that our design can be applicable to users who feel less comfortable driving.

### 4.3.4 Areas of Concern and Development

As of right now, it is unclear whether or not our design will meet the current user needs. Which is why we are constantly modifying and improving our design as we actually develop and implement our ideas. For a lot of us, this is our first experience working on a project like this, so looking that far out into the future is why the user satisfaction is uncertain at this time. We have a set plan to implement our design moving forward. Currently, both sub-teams are working on building the R.C hardware which is standardized for both groups. Our goal for the end of this semester is to have a finished hardware design with basic autonomous movement controlled by the software. In the second semester is where our teams will diverge and develop our own software applications that will be programmed onto the hardware.

This timeline does lead to some concerns because if we encounter a roadblock, or a software bug that cannot be fixed, our hardware may need to be modified to satisfy our user needs. This is why we must design our system carefully and thoughtfully.

## 4.4 Technology Considerations

**Hardware**

- HC-SRO4 Ultrasonic sensor
    - Strengths – High Accuracy, Cost Efficient, Lots of Documentation

- Weaknesses – Lower Response time, Affected by environmental factors

- Raspberry Pi
  - Strengths – Very Popular, Lots of Documentation, High Storage
  - Weaknesses – Limited Processing Power, Unideal for resource intensive software
- RealSense D455 Depth Camera
  - Strengths – Capable of detecting color, can identify certain images
  - Weaknesses – High power consumption, requires lots of processing power

### 4.5 Design Analysis

So far in Race of Doom we have disassembled the original RC car that we were given and started to play around with the different hardware components. We have been testing how to get the car to move without a controller. Trying to get this to move has been a difficult challenge that we are trying to figure out. We have connected the HC-SR04 sensors to a breadboard to be tested with the Raspberry Pi 5. We have started learning how the Real Sense D455 video sensor works. It has 2 different views; we are trying to deliberate which one to use for the depth assessment with our vehicle. Our proposed design above has been working so far. We will work out the kinks as we test and add more components to the car. We have plans to learn more of the hardware and assemble it within the next 2 weeks. We will also be seeing how the different signals are sent to the software.

# 5 Testing

We have put a strong emphasis on testing early and often for our project. It is important that we test each component of our design separately before putting things together. Making sure everything works individually is important and will help us to debug when things are put together. Testing the entire project together will go very smoothly because of our early testing. Below we have laid out our plan at every level.

### 5.1 Unit Testing

The units tested are going to be movement controls, actuators, sensors, and nerf gun mechanism. We will have to write scripts that test command movements and verify responses to moving left, right, backward, forward, and stop. For the RPLidar Sensor, HC-SR04 Ultrasonic Sensors, and Depth camera we also need to simulate and test the different obstacles orientations and distances. By sending commands to the Servo, Motor, and ESC we can verify position and physical response. We can test the nerf gun by activating the mechanism to fire it. The tools we can use for this testing are unit testing framework, RPLidar SDK, PyRealSense, and PyQT.

## 5.2 Interface Testing

Some of the interfaces for our design are sensor to processor, processor to actuators, and processor to nerf gun mechanism. We can using mocking tools, data loggers, integration scripts, and PyQT to do the interface tests. To test the sensor to processor connections we will inject some sample data from sensors then verify that the data is processed correctly by the Raspberry Pi. For the processor to actuator connection we will send commands for movement through the Raspberry Pi and then check the actuator responses. For the nerf gun we can send firing signals, checking the response after.

## 5.3 Integration Testing

Some critical integration paths for Race of Doom are movement command integration, sensor data integration, and safety features integration. All these paths ensure navigation that is autonomous and safe. Sensor data integration consists of combining data from all the different sensors in a coherent way. We will have to test our data fusion algorithms, ensuring accurate representations of the environment. Movement commands must be coordinated and processed correctly through the Raspberry Pi. We can simulate this functionaly through a track with obstacles while we monitor behavior. For safety we must ensure that the vehicle slows in yield areas, stops at stop signs, and avoids all obstacles. We can set up a track to simulate these things. The tools include a data fusion algorithm, physical track setup, and simulation environments.

## 5.4 System Testing

Our strategy for system level testing is to combine unit, interface, and integration tests to validate functionality of the system overall. The tests will include autonomous mode, full track navigation, and emergency stop. We must ensure that the vehicle can navigate autonomously. We must run the car on the entire track to test the movements overall, obstacle avoidance, and safety features. We must also test the vehicles' response to an emergency stop command. We will need simulation tools, test tracks, and data loggers.

## 5.5 Regression Testing

Our strategies for regression testing include re-running tests that were previously passed when new features or code are added. We will also maintain our regression tests that cover all critical features. The critical features are safety features, sensor data accuracy, and autonomous movement. The tools for these tests are automated test scripts, version control systems, and continuous integration tools.

## 5.6 Acceptance Testing

One strategy for acceptance testing is to create test scenarios for all non-functional and functional requirements. We will conduct demonstrations live on the test track. We will gather feedback from our stakeholder and adjust as needed. We will include our client using live demos, acceptance criteria, and feedback sessions.

### 5.7 Security Testing

We will not need security testing since there is no sensitive data involved on our project.

### 5.7 Results

The most numerical testing that has been done at this stage in our design process was the unit test for the ultrasonic sensors. With these tests, we got to see how accurate the sensors distance measurements actually are. When I put an object in front of the sensor and measured the distance out with a meter stick, I found that the sensor read distances accurately with a tolerance of 0.5-1 centimeter. For our purposes, this will work very well considering that these are not the only sensors used. However, some readings can count as outliers which may cause some confusion for our autonomous machine later down the road. For objects that were very close to the sensor, occasionally the sensor will read that the distance is more than 2 meters away (outside of its intended range). For these cases, we will have to be very careful in our software design to ignore outliers like these and identify when a ping may be inaccurate.

# 6 Implementation

Thus far we have implemented basic movements on the vehicle. The car can move forward, backward, left, and right autonomously. We did this using the Raspberry Pi and pythons gpiozero framework. We have also received and tested all of the sensor components individually. Our HCSOR4 Ultrasonic sensors provide us with accurate information about distances of an object within a 2 meter range. Each sensor has power supplied to it directly from the Raspberry Pi's 5V output pin on its GPIO board. Additionally, we have two basic GPIO pins 23 and 24 connected to the trigger and echo pins of the sensors so that this data is sent and received directly through a wired connection.

# 7 Ethics and Professional Responsibility

### 7.1 Areas of Professional Responsibility/Codes of Ethics

| Area of responsibility | Definition | IEEE Code of Ethics |
|---|---|---|
| Work competence | Completing tasks efficiently and with quality | "7. To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others." This relates to us providing proper criticism for |

| | | |
|---|---|---|
| | | our work throughout the project. |
| Financial Responsibility | Providing a solution to our listed problem while maintaining a reasonable budget | “4. To reject bribery in all forms.” Staying honest to our budget constraints as a group and remaining responsible. |
| Communication Honesty | The ability to communicate with the team while maintaining integrity | “3. To be honest or realistic in stating claims or estimates based on available data.” This relates to our group providing valid sample data and not misrepresenting the values. |
| Health, Safety, Well-Being | Focusing on the mental state of others and preventing the overworking of one group member | “8. To treat fairly all persons regardless of such factors as race, religion, gender, disability, age, or national origin.” This ethical statement aligns with the well-being of our team and its members. |
| Property Ownership | Respecting the materials we have been provided, along with honoring the components | “8. To treat fairly all persons and to not engage in acts of discrimination based on race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression.” This ethic also applies here to allow for little judgment with property and ideas. |
| Sustainability | Reduce the wear and tear of the project and respect the usage of materials | “5. To improve the understanding of technology, its appropriate application, and potential consequences.” Respecting material usage reflects understanding how to apply technology responsibly while considering its long-term consequences. |

| | | |
|---|---|---|
| Social Responsibility | Produce easy-to-use products that are user-friendly and provide guidelines | "10. to support colleagues and coworkers in their professional development and to support them in following this code of ethics." Being communicative and speaking with members of the team ensures social responsibility. |

Our group excels in communication, as we are closely connected and effectively coordinate meeting times, task assignments, and weekly responsibilities. However, one area for improvement is ensuring alignment on the project's scope. At times, there is some uncertainty about the target audience for our product. Establishing a shared understanding of this focus could enhance our teamwork and overall effectiveness.

## 7.2 Four Principles

| | Beneficence (Promoting Good) | Nonmaleficence (Avoiding Harm) | Respect for Autonomy | Justice |
|---|---|---|---|---|
| Public Health & Welfare | Improves safety guidelines in following traffic laws | Reduces operator risk by enhancing accurate safety protocols in risk diversion | Enables users to override autonomous driving and have full control of the car | Ensure equitable access to AV technology<br>Addressing potential job displacement in the transportation sector |
| Global/Culture | Enhance global mobility and connectivity.<br>Promote international collaboration on AVs | Addressing cultural differences in driving norms to ensure AVs can adapt to these changes. | Provide local communities with the opportunity to determine how to integrate AVs | Promoting fair distribution of AV benefits and addressing potential negative social impacts. |
| Environmental | Adapts to different environmental conditions for safe driving | Minimizes environmental impact with lower green-house gas emissions | Empower customers to choose AVs that align with environmental values.<br>Provide transparency about the environmental impact | Ensure environmental benefits are shared equitably including underserved communities. |
| Economic | Reduces overall energy consumption which reduces run cost of the car | Standardizes economic value of energy consumption for transportation | Allows customers to not have to worry about significant price changes in run cost of their vehicles. | Ensures economic standardization allowing for customers and the economy to not have significant marginal changes in operational costs for their vehicle |

The nonmaleficence and public health & welfare pair is important to our team. Our design is focused on the safety of the users. One thing we will be doing is creating functionality that will stop and accelerate gently. This is just one example of how to keep the users of this car from getting injured. Our team will improve in this area by testing often and making sure that the car moves safely every time. The margin of error must be very small, especially in situations that may be unexpected corner cases.

### 7.3 Virtues

We have determined three virtues that are most important to our team. These virtues are patience, honesty, timeliness, and ambition. We have listed these virtues below along with our definitions and how we will support them.

- Patience: The ability to tolerate difficulty in design, programming, and collaborating with other teammates.
    - We will support this virtue by allowing a lot of time for design and encouraging taking a break when things feel frustrating. We will also be able to understand when problems may arise between team members.
- Honesty: The ability to bring a problem up to the team when any one of us is having a problem while working on the project or disagrees with another team member in a matter related to the project.
    - We will support this by encouraging everyone's opinions before making decisions. We will also strive to create an environment that is not judgmental.
- Timeliness: Striving to go to every team meeting and to get to them on time.
    - We will support this by attempting to set up meetings times that work for everyone.
- Ambition: The desire to achieve success in our project.
    - We will support this by encouraging everyone to share their ideas and what they personally want to project to look like.

Below, we all have reflected on our own personal strengths and weaknesses when it comes to these virtues.

Alex Crandell:

A virtue that I feel I have done well with this semester is ambition. Most of our core work involved hardware and connecting sensors to the R car. While I am new to the electrical engineering field, the motivation to collaborate with teammates and complete the components remained throughout the semester. This is important to me since I usually would be nervous about completing tasks and letting the team down but newly found confidence and helpful collaboration has allowed me to succeed.

I would say my timeliness was my hardest virtue for the semester since we spent a decent amount of time gathering components and starting our core model of the RC car. Being more efficient and collaborating could help me become more focused and timelier. This is important to me for efficiency reasons as I do not like to procrastinate tasks and dwell on them. I plan on being present as much as possible in the next chapter of the course to maximize time efficiency.

Wesley Jansen:

Throughout this project, I have demonstrated both patience and ambition well. For me, this project has been a major challenge. It is challenging to have something so open-ended and no instructions on how to use anything; figure it out. With this being said, however, I am very interested in our project and bringing our ideas to fruition. My interest fuels my ambition and desire to produce a high-level product I am happy with. I have also run into a lot of issues with how the different components work together and at times was a struggle to not get upset and quit but I stuck with it and kept patient knowing I would have to get it done at some point. These two virtues have been important to me because, without them, I would have simply quit. I could ride the coattails of my teammates and get what I get, but since I have been patient with my abilities and have this ambition for our project, I have been able to stick with it and give helpful additions.

Something I have struggled with more this semester has been my timeliness. Since it has been a lot of new concepts or hardware, I have been working with things that have taken me much longer than I have expected. I would tell my teammates I was going to get a certain part done during a week and set time aside to do it. A certain part I was going to do however would take a lot more time to complete than anticipated leading to poor timeliness. For next semester I will make sure to be more reasonable with my abilities and give myself plenty of time for senior design to get my parts done on time and high quality.

Elizabeth Schmitt:

In this project I feel that I have demonstrated the virtues of patience and honesty. I do not have much experience with hardware implementation and design, so it has been quite a challenge for me. I have been very honest with my teammates about these feelings of confusion and my own shortcomings. With that, I have also exhibited patience. I have been patient with myself since I know I am still learning, and I will not be perfect right away. I have grown a lot in patience this semester through this class. I have learned not to be so hard on myself. I know that I can do this, it might just take a little extra time.

One virtue that I have struggled with while working on this project is ambition. Most of what we have been working on this semester is hardware. I do not have a lot of experience with hardware so all the stuff we have done has been very difficult for me so far. This has made it difficult to strive for success instead of just the bare minimum. I think I can work on this virtue more by learning the hardware better. This will make me more excited about the project. That excitement will ultimately lead to ambition.

Ben Towle:

One virtue that I feel I have demonstrated so far during senior design is the virtue of ambition. I find that this project is fun and interesting to work on. Therefore, I have noticed that I’ve taken a proactive role in our project implementation. I think it is important to find passion and drive towards projects like these so that during moments of struggle and hardship, you remember why you are going through it.

One virtue that I have been falling short on is patience with myself. A lot of times, I get frustrated with myself for not understanding things very quickly and that can lead to a lot more turmoil during my time working on a project. My goal for the next semester is to not overthink and panic when errors and moments of being stuck occur. It is important for me to remember that those moments are a part of the process and can help prepare you for the real world ahead.

Lalith Vattyam:

One virtue which I have demonstrated this semester is organization and participation. This project is a new experience for me as here was a lot of ambiguity at the start. Our group had to work together to come up with ways to complete all tasks and have our final project as a base for future projects to take further. During this phase I enjoyed planning and coordinating a lot as it would keep all of us on task to reach deadlines. We split our work into 2-week sprints which allowed us to reach the stage where we are now.

The project was also a new experience due to the fact that I didn't have much experience in raspberry pi or python and I wasn't very good at circuits. However, this project helped me a lot because it allowed me to learn from my peers and I'm now confident in understanding our solution and have a general idea for participating in discussions. Although it is very important to me, one virtue I find that is important to me but haven't demonstrated is patience. This virtue is very important to me as I hope to climb the corporate ladder in the near future. For this reason it is important that I always practice patience. During this next semester I plan on taking extra time to spend on the project and use my resources better to get my questions answered.

# 8 Closing Material

## 8.1 Conclusion

So far, we have successfully completed our goal of creating standardized hardware capable of basic autonomous movement. We still have further goals regarding our hardware design and development, but the most important pieces are in place. We have our HCSRO4 sensors attached to the Raspberry Pi and we have established direct communication between the sensors and the pi using the GPIO interface.

For the next semester, our goal is to have two separate software applications between the competing teams. Our plan of action towards this goal is to separate two sets of two partners to formulate a team and Alex (EE major) will be working alongside both teams to help with any hardware issues and bugs as well as designing the racetrack and obstacles.

The biggest constraint to us achieving our goals this semester was the uncertainty and open-endedness of our project. At the start of the semester, it took us a while to figure out what we wanted to accomplish, but now that we have that set-in place, we are able to move forward with our design the way we want to.

### 8.2 References


[1] "14. API - Input Devices — gpiozero 2.0.1 Documentation," *Readthedocs.io*, 2015. https://gpiozero.readthedocs.io/en/stable/api_input.html#distancesensor-hc-sr04

[2] A. Nguyen, A. Gienger, B. Dubin, B. Carlson, and C. Tow, "sdmay24-06 • Race of Doom," Iastate.edu, 2024. https://sdmay24-06.sd.ece.iastate.edu/#weeklyreports (accessed Dec. 07, 2024).

[3] D. Coffin, S. Oliver, and J. VerWey, "Building Vehicle Autonomy: Sensors, Semiconductors, Software and U.S. Competitiveness," SSRN Electronic Journal, 2019, doi: https://doi.org/10.2139/ssrn.3492778.

[4] "Raspberry Pi GPIO Pinout," *pinout.xyz*. https://pinout.xyz/

[5] S. Pasricha, "Autonomous Vehicle Design – Sudeep Pasricha," *Colostate.edu*, 2023. https://www.engr.colostate.edu/~sudeep/autonomous-vehicle-design/

**[6]** Traxxas, "XL-5 Electronic Speed Control," Accessed: Dec. 7, 2024. [Online].


# 9 Team

Complete each section as completely and concisely as possible. We strongly recommend using tables or bulleted lists when applicable.

### 9.1 Team Members

Alex Crandell, Wesley Jansen, Elizabeth Schmitt, Ben Towle, Lalith Vattyam

### 9.2 Required Skill Sets for Your Project

1. PCB Design - Alex

2. Python – Wesley, Lizzy, Ben, Lalith

3. Knowledge of Servo Motors, ESC, and Ultrasonic Sensors – Alex/Lalith

4. Raspberry Pi Applications - All

5. Version Control Software (Git) - All

6. Software Design, Development, and Testing – Wesley, Lizzy, Ben, Lalith

### 9.3 Project Management Style Adopted by the team

Our Team will be adopting the Agile Project Management Styles for this project.

## 9.4 Initial Project Management Roles

Alex – Hardware Designer/Tester

Wesley – SCRUM Master

Lizzy – Software Tester

Ben – Client Interaction

Lalith – Project Manager

## 9.5 Team Contract

**Team Members:**

1) Alex Crandell

2) Wesley Jansen

3) Elizabeth Schmitt

4) Ben Towle

5) Lalitha Vattyam

**Team Procedures**

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

   Team meetings are every Monday at 5:00 in Coover.

2. Preferred method of communication updates, reminders, issues, and scheduling

   Primarily face-to-face meetings, primary form of communication will be snapchat and emails

3. Decision-making policy (e.g., consensus, majority vote):

   Majority Vote after a discussion

4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):

   Ben will be the official timekeeper for the group. We will share the total and weekly minutes in our weekly reports.

**Participation Expectations**

1. Expected individual attendance, punctuality, and participation at all team meetings:

   All members are expected to be at every meeting with full preparation and participation. If a member is not able to make it, he/she is expected to communicate at least 1 day prior to the meeting.

2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:

   Each member is expected to complete their assignment in their given timelines, if there are any concerns that are disallowing the member to complete the given task, he/she must make sure to communicate as soon as possible.

3. Expected level of communication with other team members:

   We expect the highest level of communication from our team members. As we are seniors it is expected that we make sure to always keep everyone in the loop.

4. Expected level of commitment to team decisions and tasks:

   Everyone’s voice matters, all team decisions will be taken by a majority vote after a group discussion. After the vote, if a person is still not too happy with the decision we will open up for one more discussion after the vote which will lead to a second vote which will be the final deciding vote.

**Leadership**

1. Leadership roles for each team member:
   a. Ben – Client Interaction
   b. Lizzy – Testing
   c. Alex – Component Design
   d. Lalith – Team Organization
   e. Wesley - Team Representative
2. Strategies for supporting and guiding the work of all team members:
   a. Consistent Communication

b. Openness to asking questions and listening to others

c. Being open to go above and beyond to not only do your task but learn or help another person with theirs

3. Strategies for recognizing the contributions of all team members:

a. A pat on the back and a "job well done"

b. Team outing

c. Specific recognition in reports and public speaking events

**Collaboration and Inclusion**

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
    a. Lalith – Overall team organization and leadership skills along with understanding complex logic, and a professional background with cloud computing techniques and Azure solutions.
    b. Alex – EE expertise, with CPR E background knowledge giving a wholistic understanding of the project
    c. Ben – Complex logic expertise with high work ethic to finish task on hand. Has background knowledge of UI design and development as well as database administration.
    d. Wesley – Very high understanding of CPR E concepts with a commendable internship background in software packaging
    e. Lizzy – Solid CPR E understanding with significant knowledge in operating systems and kernel threads. Also has professional knowledge of C++ which is what most of the reused code is programmed in.
2. Strategies for encouraging and supporting contributions and ideas from all team members:
    a. Whiteboarding
    b. Always reemphasize that there are no wrong answers
3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

a. Our team is solid already in terms of getting along and there is that comfort level among us to let each other know when the team environment would be obstructing their opportunity

b. We are fully in agreement that communication should be easy and flowing amongst us and that the lack of such communication that someone may feel which would obstruct their opportunity would be something to be brought up to the professors.

**Goal-Setting, Planning, and Execution**

1. Team goals for this semester: Come up with a functional prototype (i.e. and R.C car with some type of autonomous movement)

2. Strategies for planning and assigning individual and team work:

    a. Our group will be split up into two sub-teams of 2 working on their own individual cars. Lalith and Ben will be on one team while Lizzy and Wesley will be on the other. Alex will be working alongside both teams by helping with hardware solutions for their respective R.C cars

3. Strategies for keeping on task:

    a. We will be using an excel sheet with a time and date log. This will have all our deadlines along with what parts and segments need to be done by what date to make sure that we finish our project on time. We will also be using 2-week sprints to keep a consistent flow of tasks getting done.

**Consequences for Not Adhering to Team Contract**

1. How will you handle infractions of any of the obligations of this team contract?

    a. We will have a meeting with the team member as a group and explain that further infractions will have to be reported to the professor and advisors

2. What will your team do if the infractions continue?

    a. We will contact the professors and advisors and ask about next steps

*************************************************************************

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*

b) *I understand that I am obligated to abide by these terms and conditions.*

c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) _______Lalith Vattyam______________________________ DATE ________9/19/2024__________

2) _______Benjamin Towle_____________________________DATE ________9/19/2024__________

3) _______Alex Crandall________________________________DATE ________9/19/2024__________

4) _______Wesley Jansen________________________________ DATE ________9/19/2024__________

5) _______Elizabeth Schmitt____________________________DATE ________9/19/2024__________