# Sprint 1 - Agility Design Document

# November 30th, 2022

# Table of Contents

# 1. Executive Summary

## *1.1 Project Overview*

This project is focused on how well the Sphero can move around HH208. Endurance is to see whether or not the robot is capable of enduring moving in long distances between the four corners of the room. The accuracy portion is to see how well it can mimic figure eight on the guiding tape. Lastly, the agility portion is to depict how well the Sphero can be programmed to find ways around obstacles and how well it can move. Sphero is a robot that can be controlled remotely and can be given an algorithm to move in certain directions, speak, and even change colors.

## *1.2 Purpose and Scope of this Specification*

**In Scope**

- The Main point of the accuracy portion is to see if Sphero is capable of during 5 figure 8 motions on the tape provided on the floor of HH208
- Staying within the lines and not running off or being sloppy is within the scope as well.

**Out of Scope**

- Two parts of this 3 part project that are out of scope are the agility and endurance portions.
- Additional code will be added for agility which involves knocking some pins over.

# 2. Product/Service Description

Some factor that may affect the outcome of the product is how the floor is not completely smooth, causing the Sphero to somewhat go off track but as a whole, the product will come out to the end result that is desired.

## *2.1 Product Context*

Sphero is related to other products which involve robotics. It is a programmable robot in a ball that also can be remotely controlled. It is self-contained, having all of the processors and boards inside the ball itself while also having a motor to allow the ball to move around.

## *2.2 User Characteristics*

Professor Eckert who works at Monmouth University, has a bachelor of science at Stockton State College and a Master of Science from Kean University. He has plenty of years of experience when it comes to technology and is technologically savvy when it comes to programming and using the Sphero robot.

## *2.3 Assumptions*

Even though as a group we have all used and programmed a Sphero once does not make us experts. There is still some practice that needs to be had before we are fully capable of using all of its distinct features to effectively go through the course(s).

### *2.4 Constraints*

There are not really any design options when it comes to a fully-built robot. Not really any hardware limitations, but one constraint that exists would be the use of block code.

### *2.5 Dependencies*

The dependent required for the robot to work is the block code that will be programmed with the appropriate website. The block code and algorithm must be completed before the robot is capable of completing the endurance task.

# 3. Requirements

### *3.1 Functional Requirements*

| Req# | Requirement | Comments | Priority | Date Rvwd | SME Reviewed / Approved |
|---|---|---|---|---|---|
| AGG_01 | Robot must begin course on square | Points will be taking off if we don't meet this requirement | 1 | 11/29/2022 | Approved |
| AGG_02 | Robot must stay within the path provided | Points will be taking off if we don't meet this requirement | 2 | 11/29/2022 | Approved |
| AGG_03 | Robot must avoid 3 obstacles in its path completely | Points will be taking off if we don't meet this requirement | 1 | 11/29/2022 | Approved |
| AGG_04 | Robot must jump over the ramp | Points will be taking off if we don't meet this requirement | 1 | 11/29/2022 | Approved |
| AGG_05 | Robot must knock over as many pins as possible, not all are required but is suggested | Points will be taking off if we don't meet this requirement | 3 | 11/29/2022 | Approved |

### *3.2 Security*

#### 3.2.1 Protection

Specify the factors that will protect the system from malicious or accidental access, modification, disclosure, destruction, or misuse.

1. Activity logging, historical data sets
2. Data integrity checks
3. Keep the robot away from others to decrease the chance of destroying the robot
4. Do not tell others your login information to Sphero EDU or github
5. Keep the robot away from dangerous environments
6. If your going to use the robot let the other group members know

#### 3.2.2 Authorization and Authentication

Specify the Authorization and Authentication factors. Consider using standard tools such as PubCookie.

Include documentation of the approval or confirmation of the requirements here. For example:

| Meeting Date | Attendees (name and role) | Comments |
|---|---|---|
| 11/15/2022 | Aidan M(Documenter), Raul(Programmer), Olivia(Data Collector) | |

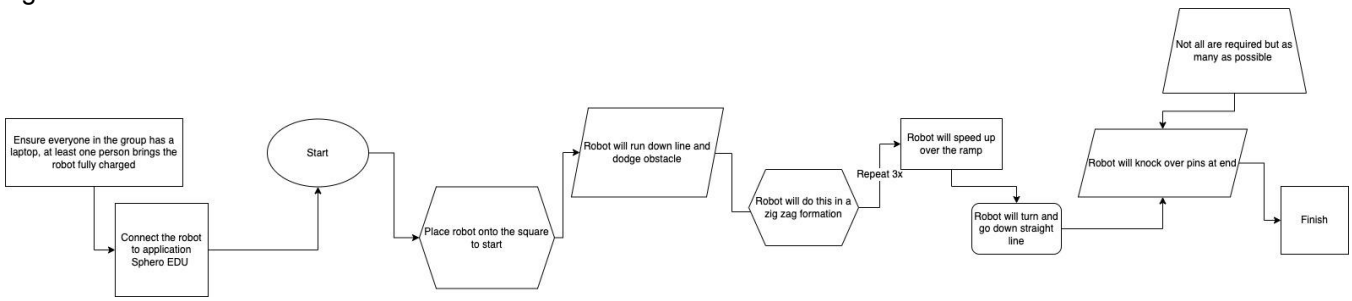
# 4. System Design

## *4.1 Algorithm*

1. Ensure everyone in the group has a laptop, at least one person brings the robot fully charged
2. Connect the robot to application Sphero EDU
3. Start
4. Place robot onto square to start
5. Robot will run down line and dodge obstacle
6. Robot will do this in a zig zag formation repeating 3x
7. Robot will speed up over the ramp
8. Robot will turn and go down straight line
9. Robot will knock over pins at end, not all are required but as many as possible
10. Finish

## *4.2 System Flow*

Develop a flowchart (and show here) that accurately depicts how your software application will act to fulfill the algorithm




## *4.3 Software*

Describe software languages/platforms/api's used to develop and deploy this application

- Sphero EDU application to connect and program our robot
- Google Docs used to organize plans on SDD
- Microsoft Excel to create a chart for testing.

## *4.4 Hardware*

Describe hardware platforms that were used to develop, test and demonstrate this application

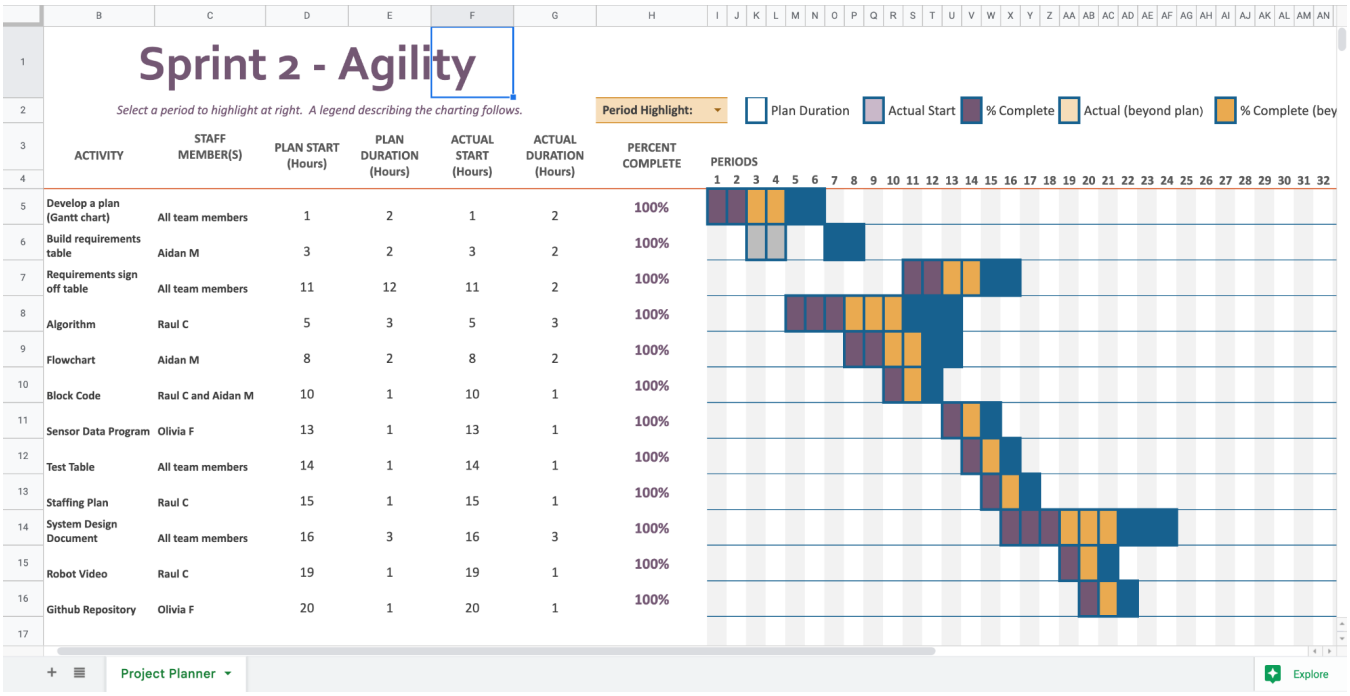
- Laptop
- Robot

## *4.5 Test Plan*

Include a test plan showing all unit tests performed for this application, Include test rational, test date, staff member, pass/fail status

| Reason for Test Case | Test Date | Expected Output | Observed Output | Staff Name | Pass/Fail |
|---|---|---|---|---|---|
| See whether the robot will dodge first obstacle | 11/29/22 | Will not dodge first obstacle | Dodged first obstacle | Aidan, Olivia, Raul | Pass |
| see whether the robot will make the first turn | 11/29/22 | will turn | made the turn | Aidan, Olivia, Raul | Pass |
| See whether the robot will dodge second obstacle | 11/29/22 | will not dodge second obstacle | Hit obstacle | Aidan, Olivia, Raul | Fail |
| see whether the robot will make the second turn | 11/29/22 | will make the turn | made the turn | Aidan, Olivia, Raul | Pass |
| See whether the robot will dodge second obstacle | 11/29/22 | will not dodge second obstacle | dodged second obstacle | Aidan, Olivia, Raul | Pass |
| See whether the robot will dodge third obstacle | 11/29/22 | will not dodge third obstacle | Hit obstacle | Aidan, Olivia, Raul | Fail |
| see whether the robot will make the third turn | 11/29/22 | will make the turn | made the turn | Aidan, Olivia, Raul | Pass |
| See whether the robot will dodge third obstacle | 11/29/22 | will not dodge third obstacle | dodged third obstacle | Aidan, Olivia, Raul | Pass |
| See whether the robot will clear the ramp | 11/29/22 | will not clear ramp | cleared ramp | Aidan, Olivia, Raul | Pass |
| See whether the robot will stop at corner | 11/29/22 | will not stop at corner | did not stop at corner | Aidan, Olivia, Raul | Fail |
| See whether the robot will stop at corner | 11/29/22 | will not stop at corner | stopped at corner | Aidan, Olivia, Raul | Pass |
| see whether the robot will make the fourth turn | 11/29/22 | will not make turn | did not make turn | Aidan, Olivia, Raul | Fail |
| see whether the robot will make the fourth turn | 11/29/22 | will not make turn | made the turn | Aidan, Olivia, Raul | Pass |
| see whether the robot will knock over the pins | 11/29/22 | will not knock any pins | knocked over 8 pins | Aidan, Olivia, Raul | Pass |

## *4.6 Task List/Gantt Chart*



# Sprint 2 - Agility

*Select a period to highlight at right. A legend describing the charting follows.*

Period Highlight: | Plan Duration | Actual Start | % Complete | Actual (beyond plan) | % Complete (bey

| ACTIVITY | STAFF MEMBER(S) | PLAN START (Hours) | PLAN DURATION (Hours) | ACTUAL START (Hours) | ACTUAL DURATION (Hours) | PERCENT COMPLETE |
|---|---|---|---|---|---|---|
| Develop a plan (Gantt chart) | All team members | 1 | 2 | 1 | 2 | 100% |
| Build requirements table | Aidan M | 3 | 2 | 3 | 2 | 100% |
| Requirements sign off table | All team members | 11 | 12 | 11 | 2 | 100% |
| Algorithm | Raul C | 5 | 3 | 5 | 3 | 100% |
| Flowchart | Aidan M | 8 | 2 | 8 | 2 | 100% |
| Block Code | Raul C and Aidan M | 10 | 1 | 10 | 1 | 100% |
| Sensor Data Program | Olivia F | 13 | 1 | 13 | 1 | 100% |
| Test Table | All team members | 14 | 1 | 14 | 1 | 100% |
| Staffing Plan | Raul C | 15 | 1 | 15 | 1 | 100% |
| System Design Document | All team members | 16 | 3 | 16 | 3 | 100% |
| Robot Video | Raul C | 19 | 1 | 19 | 1 | 100% |
| Github Repository | Olivia F | 20 | 1 | 20 | 1 | 100% |

PERIODS 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

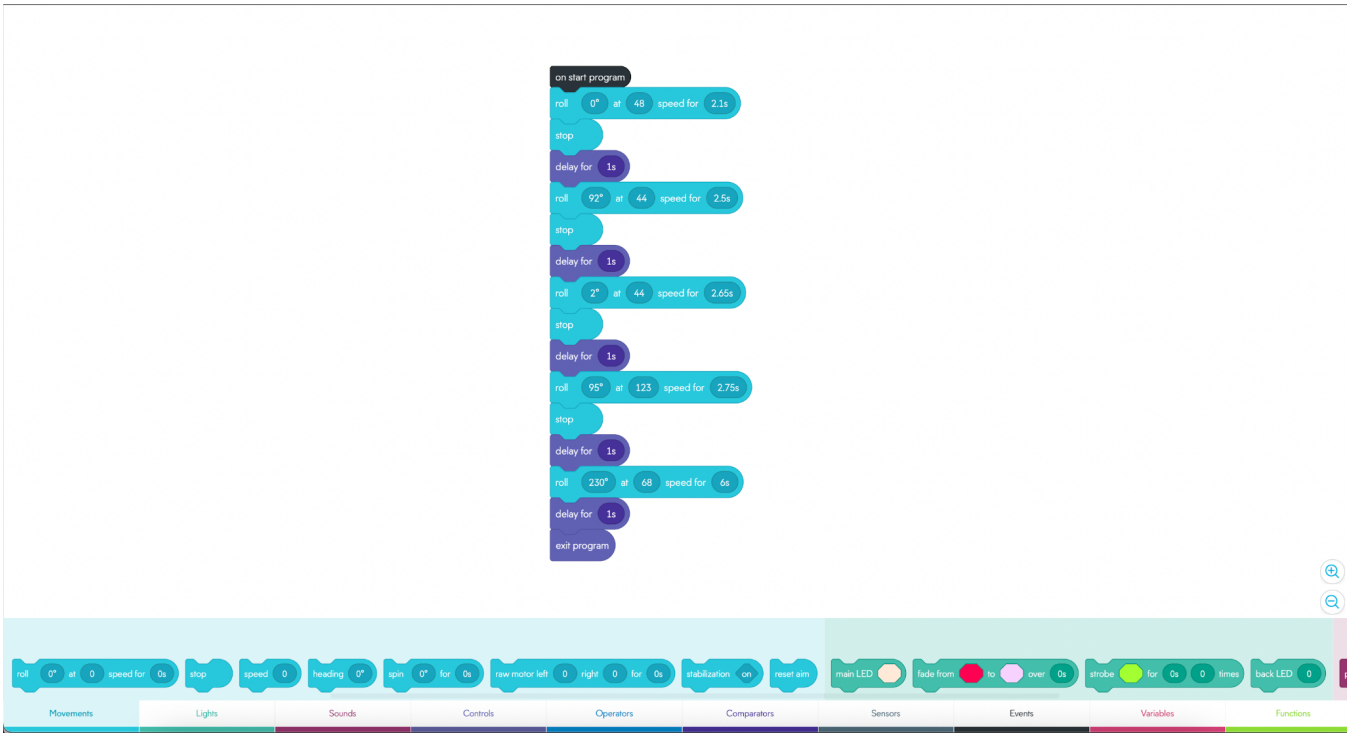
Project Planner

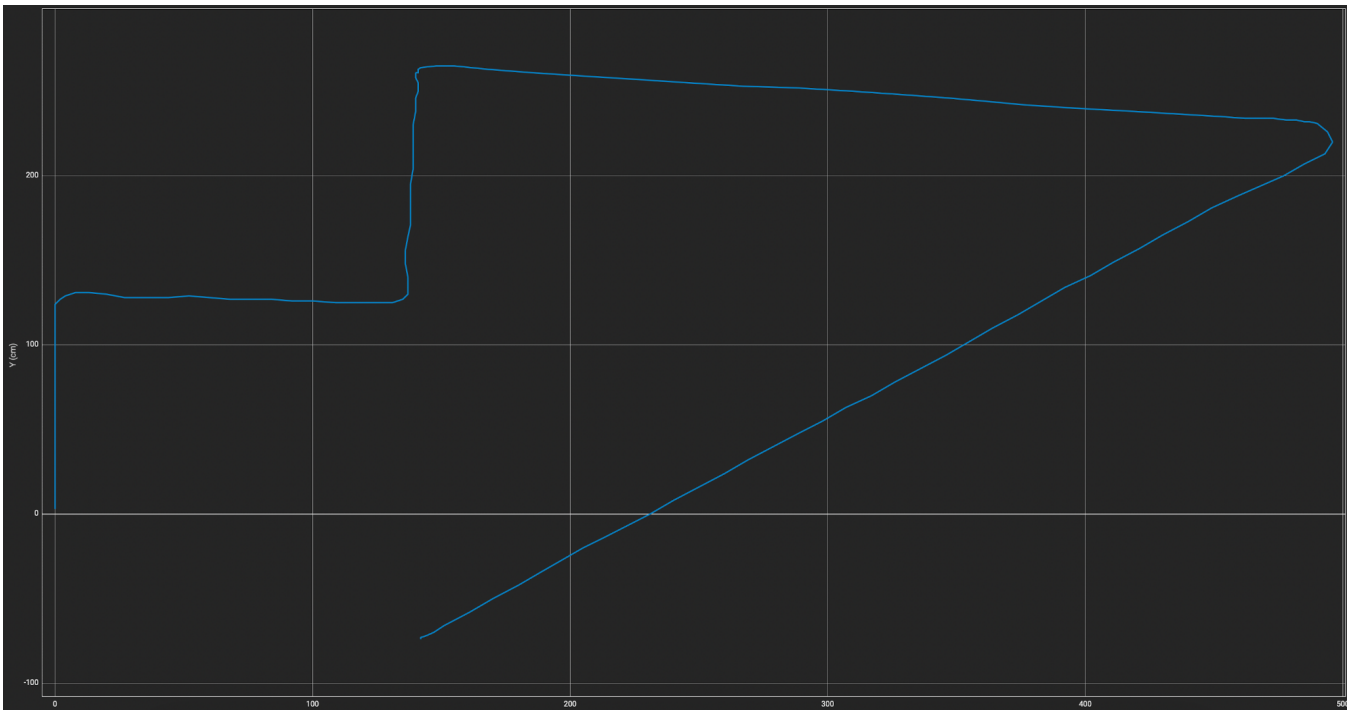
Explore

## *4.7 Staffing Plan*

Insert a chart/table that depicts the roles and responsibilities of each team member that worked on this project

| Name | Role | Responsibility | Reports To |
|---|---|---|---|
| Raul C | Programmer / Management | Executive Summary page / Algorithm / Block Code / Robot Video | |
| Aidan M | Documenter / Programmer | Requirements table / Flowchart / Block Code | |
| Olivia F | Documenter / Data collector | Github Repositories / Sensor Data Program | |


on start program
roll 0° at 48 speed for 2.1s
stop
delay for 1s
roll 92° at 44 speed for 2.5s
stop
delay for 1s
roll 2° at 44 speed for 2.65s
stop
delay for 1s
roll 95° at 123 speed for 2.75s
stop
delay for 1s
roll 230° at 68 speed for 6s
delay for 1s
exit program
Movements
Lights
Sounds
Controls
Operators
Comparators
Sensors
Events
Variables
Functions


Y (cm)
200
100
0
-100
0
100
200
300
400