Sprint 3 - Agility Design Document

November 29, 2022

By Dominick Del Bene, Aaila Arif, and Zaccery Tarver

Table of Contents

[1. Executive Summary 3](#_Toc21616852)

[1.1 Project Overview 3](#_Toc21616853)

[1.2 Purpose and Scope of this Specification 3](#_Toc21616854)

[2. Product/Service Description 3](#_Toc21616855)

[2.1 Product Context 3](#_Toc21616856)

[2.2 User Characteristics 3](#_Toc21616857)

[2.3 Assumptions 3](#_Toc21616858)

[2.4 Constraints 3](#_Toc21616859)

[2.5 Dependencies 4](#_Toc21616860)

[3. Requirements 4](#_Toc21616861)

[3.1 Functional Requirements 5](#_Toc21616862)

[3.2 Security 5](#_Toc21616863)

[3.2.1 Protection 5](#_Toc21616864)

[3.2.2 Authorization and Authentication 6](#_Toc21616865)

[3.3 Portability 6](#_Toc21616866)

[4. Requirements Confirmation/Stakeholder sign-off 6](#_Toc21616867)

[5. System Design 6](#_Toc21616868)

[5.1 Algorithm 6](#_Toc21616869)

[5.2 System Flow 6](#_Toc21616870)

[5.3 Block Code 6](#_Block_Code)

[5.4 Software 6](#_Toc21616871)

[5.5 Hardware 6](#_Toc21616872)

[5.6 Test Plan 7](#_Toc21616873)

[5.7 Sensor Data Diagram 6](#_Sensor_Data_Diagram)

[5.8 Task List/Gantt Chart 7](#_Toc21616874)

[5.9 Staffing Plan 7](#_Toc21616875)

# Executive Summary

## Project Overview

This project has been created to program a Sphero robot to travel along a blue-tape obstacle track. Within the track, the Sphero robot will avoid obstacles as well as travel over a ramp and knock over pins at the end. It has been created to test the applications of the Sphero robot and the ability of the robot to meet the specified requirements. The intended audience of this project is Professor Eckert and our fellow classmates.

## Purpose and Scope of this Specification

The purpose of this specification is to test the capabilities of the team members to collaborate and meet all the requirements specified. It is also to test the performance of the Sphero robot and application when completing an obstacle course with many obstacles to avoid and objectives such as knocking pins over after going over a ramp.

In scope

* *Traveling along a blue-tape obstacle course*
* *Programming robot to change angles and knock over pins*
* *Programming robot to avoid glass bottle obstacles*
* *Programming robot to travel over a ramp*

Out of Scope

* *Courses other than the current blue-tape obstacle course*
* *Actions not specified in the requirements*

# Product/Service Description

## Product Context

This product is the third part of the 3-part Robotics Triathlon. This product is self-contained and independent of the other parts of the Triathlon. All that is needed is the Sphero Edu application as well as the Sphero robot.

## User Characteristics

* *Team members- Students, two years of coding experience, beginner knowledge of Sphero programming*

## Assumptions

* *The classroom (HH-208) is available for students when necessary for testing purposes*
* *Team members have non-conflicting schedules and can meet up before the deadline*
* *The Sphero robot is provided by Professor Eckert*
* *Sphero robot is functional as well as the application used for programming*
* *Laptops and phones used for programming are available*
* *Glass bottle obstacles as well as pins and ramps are available for testing*

## Constraints

* *Team members are only able to meet up at specific times*
* *Parts of the tape course are missing*
* *Tiles on the ground of the classroom are uneven*
* *HH-208 is not always available/shared with other groups*
* *Quick deadlines*
* *Obstacles, pins, and ramp are not always* *available*

## Dependencies

* *Testing must occur in HH-208*
* *Requirements table and algorithm must be completed before coding can begin*
* *Sphero robot must be placed on blue-tape obstacle course and must knock over the pins as well as avoid obstacles*
* *Sphero Edu application must be used to program robot*

# Requirements

## Functional Requirements

| Req# | Requirement | Comments | Priority | Date Rvwd | SME Reviewed / Approved |
| --- | --- | --- | --- | --- | --- |
| AGILIT\_01 | The robot will start on the blue square. | The location of the robot will dictate where the robot will move. | 1 | 11/19/22 | Aaila Arif |
| AGILIT\_02 | The robot’s LED will turn green and say “Okay, let’s do this!” | Zaccery Tarver (Software Designer) added this. | 3 | 11/19/22 | Aaila Arif |
| AGILIT\_03 | The robot will move forward and stop while avoiding the 1st obstacle. | The robot must avoid the obstacles or else its movement will be affected. | 1 | 11/19/22 | Aaila Arif |
| AGILIT\_04 | The robot will turn, move forward and stop while avoiding the 2nd obstacle. | The robot must avoid the obstacles or else its movement will be affected. | 1 | 11/19/22 | Aaila Arif |
| AGILIT\_05 | The robot will turn, move forward and stop while avoiding the 3rd obstacle. | The robot must avoid the obstacles or else its movement will be affected. | 1 | 11/19/22 | Aaila Arif |
| AGILIT\_06 | The robot will turn and increase its speed to go over the ramp and stop after hitting the end of the ramp. | The ramp can change where the robot lands depending on multiple factors. | 1 | 11/19/22 | Aaila Arif |
| AGILIT\_07 | The robot will adjust itself by moving forward about an inch and turn itself. | This program will ensure the robot will move up to the last blue line. | 2 | 11/19/22 | Aaila Arif |
| AGILIT\_08 | The robot’s LED will turn red and say, “Time to destroy those pins!” | Zaccery Tarver (Software Designer) added this. | 3 | 11/19/22 | Aaila Arif |
| AGILIT\_09 | The robot will increase its speed and move forward fast to knock over as many pins as possible. | Points will be added to each pin that the robot knocks down. | 1 | 11/19/22 | Aaila Arif |
| AGILIT\_10 | The robot’s LED will turn yellow and say, “I did it!” | Zaccery Tarver (Software Designer) added this. | 3 | 11/19/22 | Aaila Arif |
| AGILIT\_11 | The robot’s program will end. | The robot’s location at the end doesn’t matter. | 1 | 11/19/22 | Aaila Arif |

## Security

### Protection

* Windows Laptops
* Monmouth Accounts
* Sphero Accounts
* GitHub Accounts

### Authorization and Authentication

* All accounts have some sort of authentication that verifies who someone is.
* GitHub has authorization to decide what files the user has access to.

## Portability

* Sphero is easily portable among the team members. If a member wants to use the robot, they can connect their phone or computer to Sphero. Then they can control the robot via a program on their device.
* To share a Sphero program, hit the share icon on the top right and send it to a member.

# Requirements Confirmation/Stakeholder sign-off

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

|  |  |  |
| --- | --- | --- |
| Meeting Date | Attendees (name and role) | Comments |
| 11/19/22 | Aaila Arif (Project Executive), Dominick Del Bene (Requirements Manager) and Zaccery Tarver (Software Designer) | Conversed on the basis for our robot |
| 11/21/22 | Aaila Arif (Project Executive), Dominick Del Bene (Requirements Manager) and Zaccery Tarver (Software Designer) | Confirmed all |

# System Design

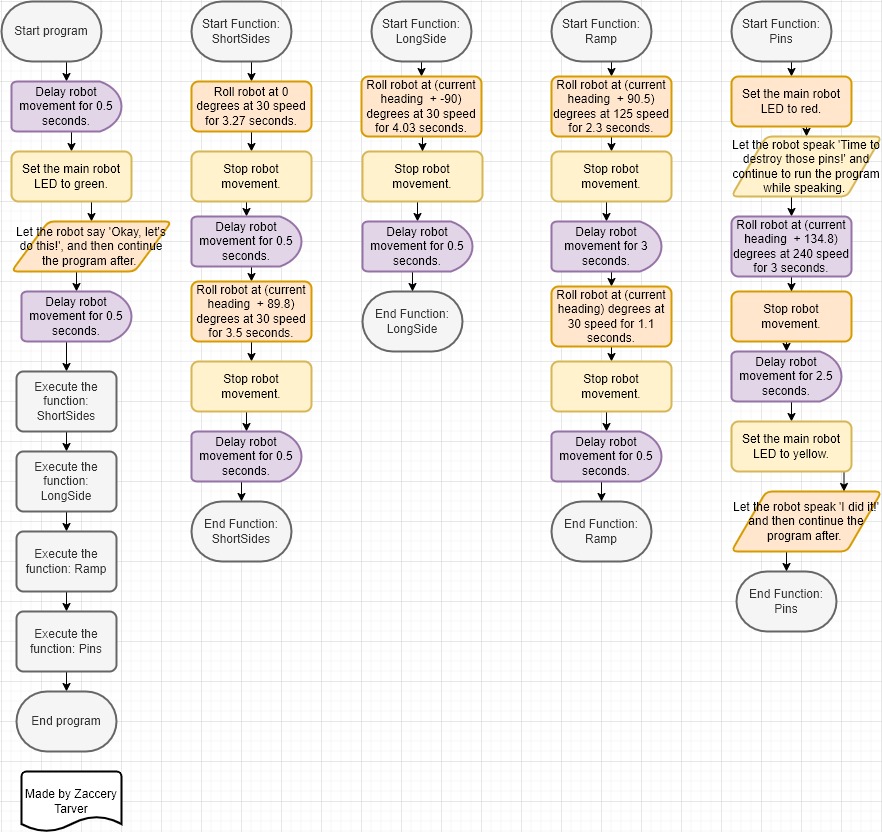
This section will provide all details concerning the technical design, staffing, coding, and testing the system

## Algorithm

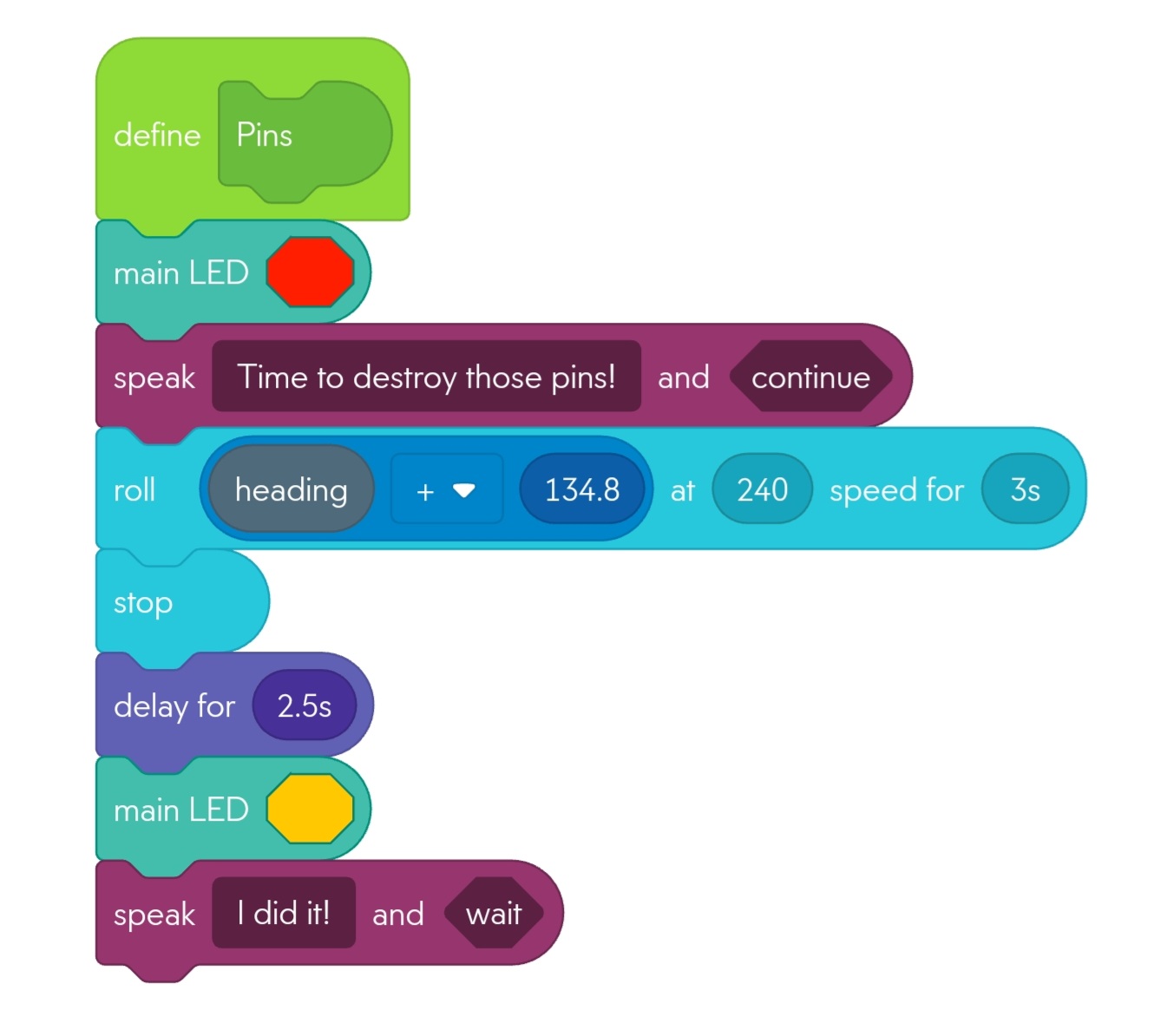
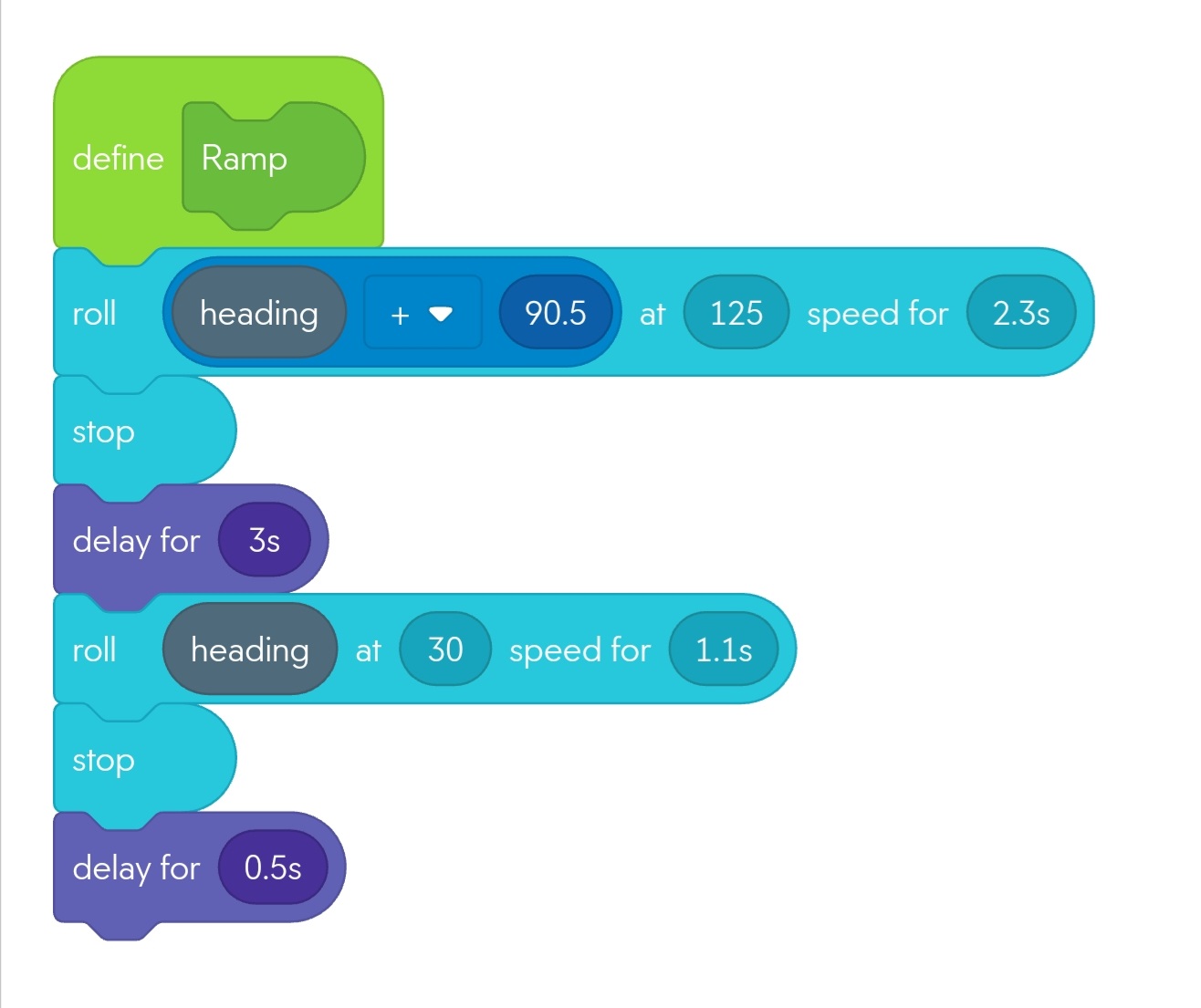
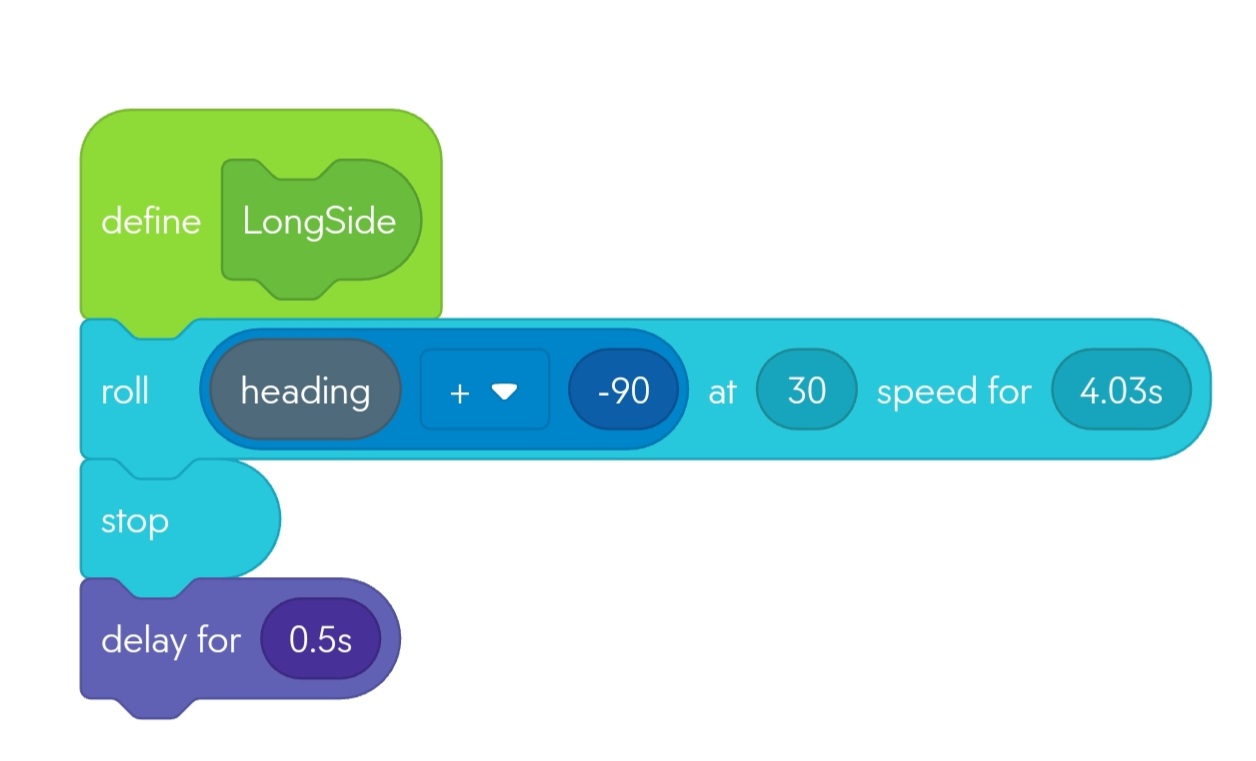
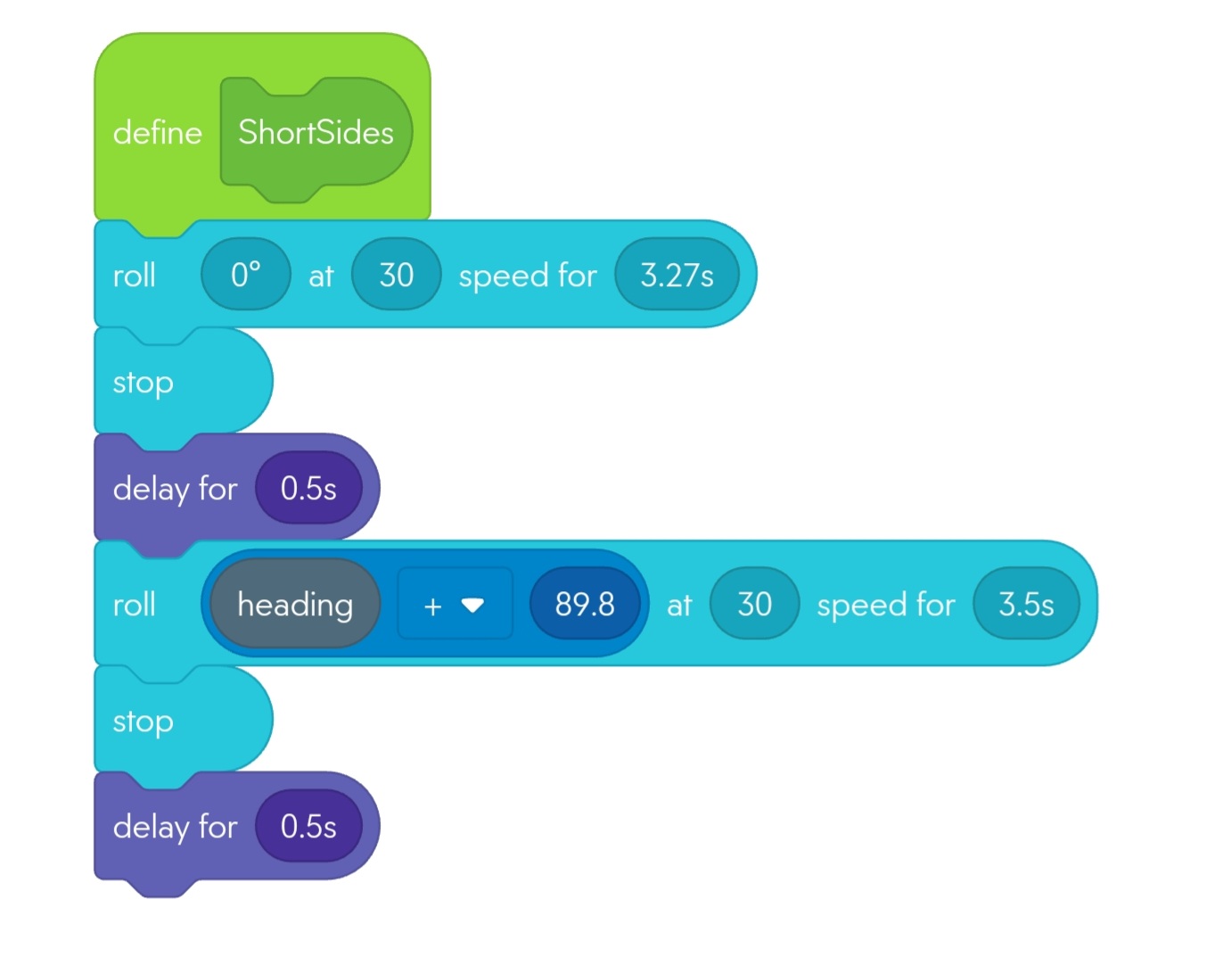
\*An initial experiment was done to determine an approximate formula for Sphero’s speed. Sphero was instructed to move a fixed distance with a fixed speed value, recording how long it took to move that fixed distance for six trials. Then, the average time was taken of these six trials. Dividing the fixed distance by the average time gives the speed in real units rather than Sphero’s artificial speed units. Dividing the real speed by whatever Sphero speed value used in the experiment will give the unit speed for Sphero and how it relates to a real speed. Of course, this is only true if increasing Sphero’s speed is a linear model – it could be quadratic, cubic, logarithmic, etc. However, it is more likely to be linear and still serves as a great approximation method.

1. Delay robot movement for 0.5 seconds.
2. Set the main robot LED to green.
3. Let the robot say ‘Okay, let’s do this!’, and then continue the program after.
4. Delay the robot movement for 0.5 seconds.
5. Execute a function to travel the first two paths:
   1. Roll robot at 0 degrees at 30 speed for 3.27 seconds.
   2. Stop robot movement.
   3. Delay robot movement for 0.5 seconds.
   4. Roll robot at (current heading + 89.8) degrees at 30 speed for 3.5 seconds.
   5. Stop robot movement.
   6. Delay robot movement for 0.5 seconds.
   7. End the function.
6. Execute a function to travel the third, longer path:
   1. Roll robot at (current heading + -90) degrees at 30 speed for 4.03 seconds.
   2. Stop robot movement.
   3. Delay robot movement for 0.5 seconds.
   4. End the function.
7. Execute a function to jump the ramp and stabilize afterwards:
   1. Roll robot at (current heading + 90.5) degrees at 125 speed for 2.3 seconds.
   2. Stop robot movement.
   3. Delay robot movement for 3 seconds.
   4. Roll robot at (current heading) degrees at 30 speed for 1.1 seconds.
   5. Stop robot movement.
   6. Delay robot movement for 0.5 seconds.
   7. End the function.
8. Execute a function to roll super-fast and knock down the pins.
   1. Set the main robot LED to red.
   2. Let the robot speak ‘Time to destroy those pins!’ and continue to run the program while speaking.
   3. Roll robot at (current heading + 134.8) degrees at 240 speed for 3 seconds.
   4. Stop robot movement.
   5. Delay robot movement for 2.5 seconds.
   6. Set the main robot LED to yellow.
   7. Let the robot speak ‘I did it!’, then continue the program after.
   8. End the function.
9. End program.

## System Flow



## Block Code



## Software

* Sphero Block Code
* Sphero Edu

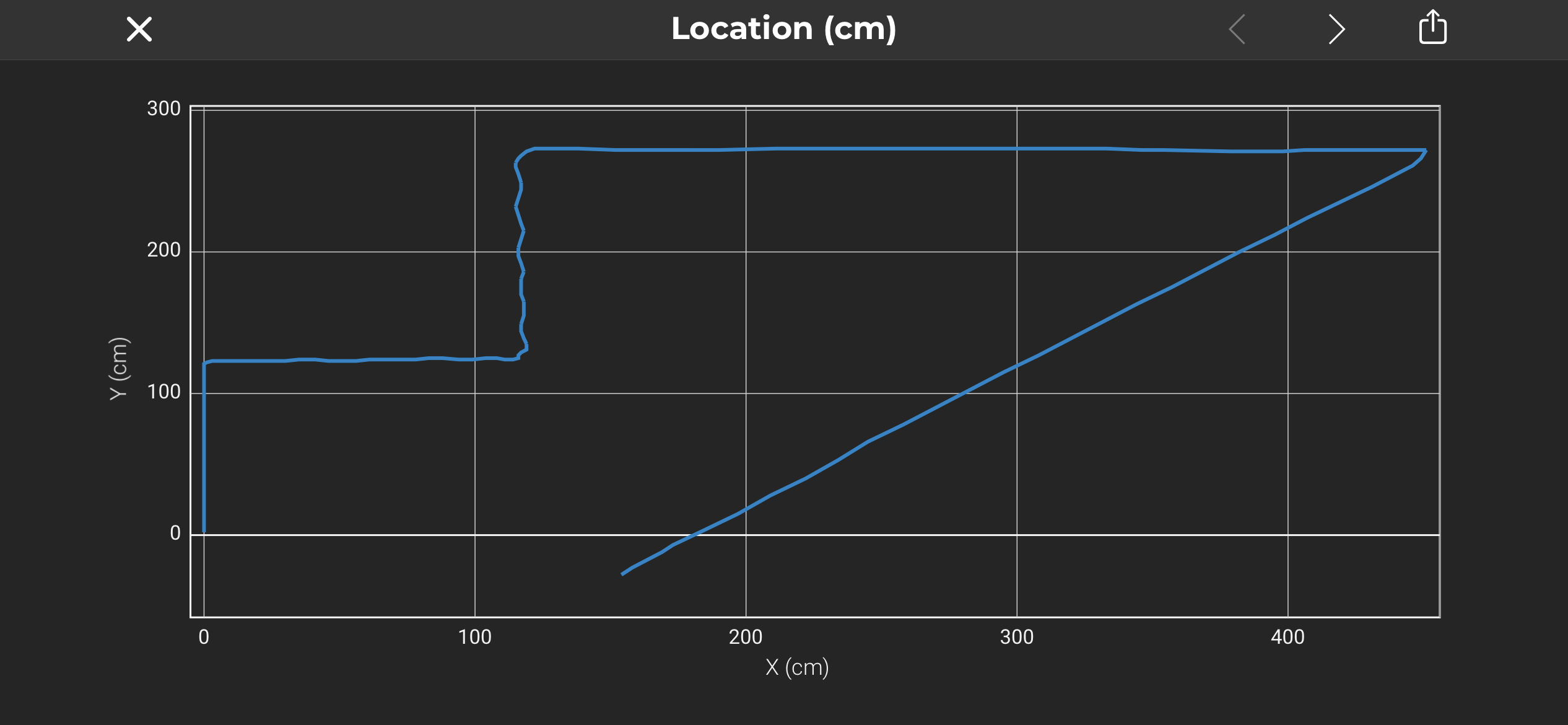
## Hardware

* Sphero Robot
* A laptop or phone can run the Sphero program.

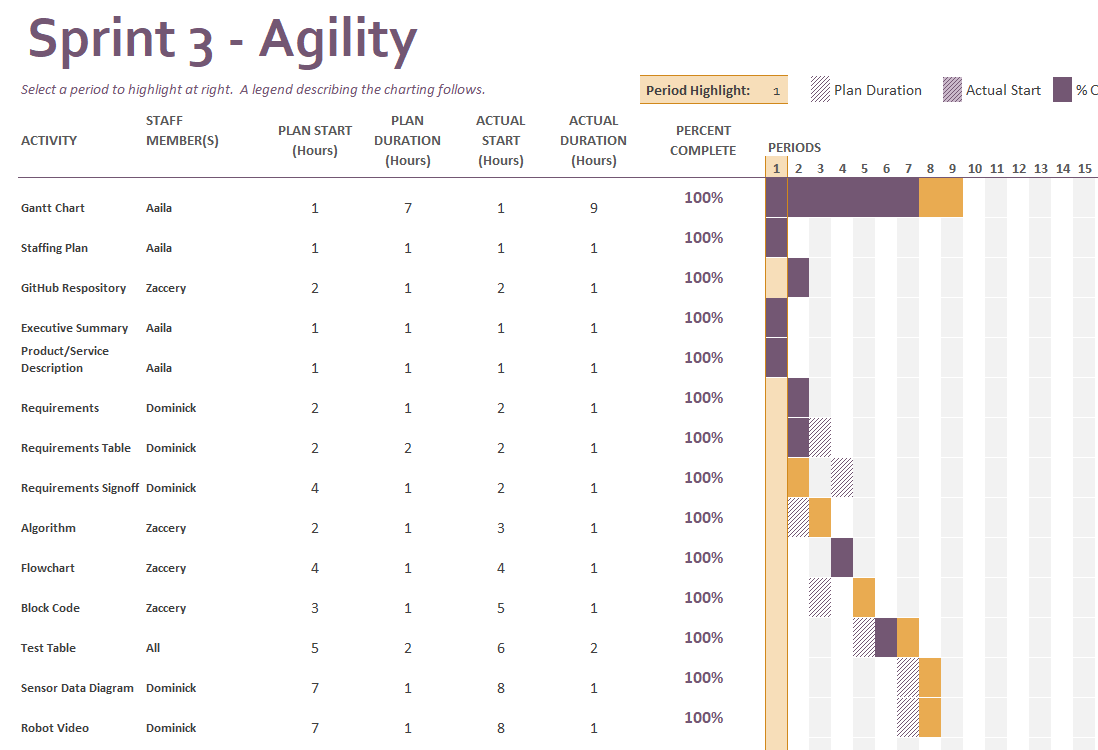
## Test Plan

| **Reason for Test Case** | **Test Date** | **Expected Output** | **Observed Output** | **Staff Name** | **Pass/Fail** |
| --- | --- | --- | --- | --- | --- |
| Testing for first travel time, as a part of an experiment to determine speed. | 11/18 | Sphero travels in an approximate straight line. | Sphero travels in an approximate straight line. | Zaccery Tarver | Pass |
| Testing for completion of the short sides. | 11/18 | Sphero should complete the two short sides of the track and avoid the bottles. | Sphero travels without turning. | Zaccery Tarver | Fail |
| Testing for completion of the short sides. | 11/18 | Sphero should complete the two short sides of the track and avoid the bottles. | Sphero completes the two short sides and avoids the glass bottles. | Zaccery Tarver | Pass |
| Testing for completion of the long side as well. | 11/18 | Sphero should be able to complete the short sides and longer side. | Sphero avoids the glass bottle on the third path. Time needs to be adjusted for distance traveled since it just grazed the second bottle. | Zaccery Tarver | Pass |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero ends travel right before ramp. This needs to be fixed. | Zaccery Tarver | Fail |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. Travel time before entering the ramp increased to 2 seconds, so Sphero should make the jump now. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero not centered enough to complete ramp. Time needs to be adjusted. | Zaccery Tarver | Fail |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. Travel time of the long side changed to 3.883 seconds make Sphero more in line with ramp. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero did not make it up the ramp. | Zaccery Tarver | Fail |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. Travel time before entering ramp increased to 2.7 seconds, so Sphero should be able to make the jump. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero cannot travel up the ramp due to a lack of speed. Sphero nudges the 2nd bottle. | Zaccery Tarver | Fail |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. Travel time before entering ramp decreased to 2.5 seconds, but speed increased to 115, so Sphero should be able to make the jump. The first path travel time changed to 3.27 seconds. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero does not have enough speed to go up the ramp. | Zaccery Tarver | Fail |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. Speed increased to 125 seconds, which should be enough. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero does the jump! Another trial will be done before starting data collection. | Zaccery Tarver | Pass |
| Adjusting the distance traveled along the long side from 3.883 seconds to 3.91, lowering the time of travel for the ramp jump from 2.5 to 2.3 seconds. | 11/18 | Sphero should land in the approximate area of the fourth checkpoint. | Sphero misses the ramp by its lip. | Zaccery Tarver | Fail |
| Adjusting the distance traveled along the long side from 3.883 seconds to 4.05, which ensures that Sphero remains centered. | 11/18 | Sphero should land in the approximate area of the fourth checkpoint. | Sphero veers too far left. | Zaccery Tarver | Fail |
| First trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. Long side travel changed to 4 seconds. | 11/18 | Sphero should land in approximate checkpoint area and be in-line with the ramp. | Sphero completes the first jump: 7.25 inches. | Zaccery Tarver | Pass |
| Second trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero completes the second jump: 8 inches. | Zaccery Tarver | Pass |
| Third trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero completes the third jump: 11 and 3/8 inches. | Zaccery Tarver | Pass |
| Fourth trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero completes the fourth jump: 16 inches. | Zaccery Tarver | Pass |
| Fifth trial to determine ramp data. The remaining time of travel needs to be approximated from test data of the jump. | 11/18 | Sphero should jump the ramp and end up a variable distance *x* from the end of the path. | Sphero completes the fifth jump: 12 inches. | Zaccery Tarver | Pass |
| The average time from the average distances can now be calculated. Testing for completion of the ramp segment. | 11/18 | Sphero should travel a short distance to reach the next checkpoint. | Sphero loses control after landing, not traveling straight. | Zaccery Tarver | Fail |
| Testing for completion of the ramp segment. Increased delay time from 0.5 to 3 seconds after completing ramp. | 11/18 | Sphero should maintain stability after ramp jump. | Sphero maintains control and completes the ramp segment, being in the approximate vicinity of the end location. | Zaccery Tarver | Pass |
| Testing the pins segments as well now. | 11/18 | Sphero should be able to knock down at least one pin. | Sphero stops before the pins. | Zaccery Tarver | Fail |
| Testing the pins segment again. Time of travel increased to 4 seconds. | 11/18 | Sphero should knock down at least one pin. | Sphero not only completes the course but knocks down multiple pins. Success! | Zaccery Tarver | Pass |
| Test to acquire the sensor data. | 11/21 | Sphero should remain in the approximate vicinity of the path. | Sphero stays along the path well for an accurate Location reading. | Dominick | Pass |
| Test to acquire the video. Increasing the final speed of the robot to knock down more pins. | 11/21 | Sphero should avoid all obstacles, jump the ramp, and knock down all of the pins. | 3 Pins knocked down, will test again for more. Sphero did the rest of the obstacle course just fine. | Zaccery, Dominick | Pass |
| Test to acquire the video. Sphero’s travel time to the fourth checkpoint will increase from 0.897 to 0.909 seconds so that it stays more centered. | 11/21 | Sphero should knock down most of the pins now. | Sphero missed the pins. | Zaccery, Dominick | Fail |
| Test to acquire the video. Sphero’s travel time to the fourth checkpoint will increase from 0.909 to 1.1 seconds so that it stays more centered. | 11/21 | Sphero should knock down all of the pins now. | Sphero knocks down 5 pins. | Zaccery, Dominick | Pass |
| Test to acquire the video. Re-testing with a faster final speed and shorter travel time so that Sphero has less of a chance to wobble. Adjusted the travel angle and time of the second route to avoid grazing the bottle. | 11/21 | Sphero should knock down all of the pins now. | Sphero knocks down 9 pins really fast. Successful! | Zaccery, Dominick | Pass |

## Sensor Data Diagram



## Task List/Gantt Chart



## Staffing Plan

| Name | Role | Responsibility | Reports To |
| --- | --- | --- | --- |
| Aaila Arif | Project Executive | - To specify the main goals of the project underlined in the executive summary and project service description.  -To check that the other teammates fulfill their responsibilities properly and on time.  - To take a video of the final product. | Team Members |
| Dominick Del Bene | Requirements Manager | - To specify the requirements at the beginning of the project, and to update them accordingly as needed.  - To specify the security measures surrounding the project.  - To acquire the sensor data of the robot. | Aaila |
| Zaccery Tarver | Software Designer | - To manage and organize the GitHub repository.  - To build the algorithm.  - To make the flowchart.  - To construct the block-code. | Aaila |