Sprint 2 - Accuracy Design Document

November 14, 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 7](#_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 figure 8 track five times. 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 a figure 8 track.

In scope

* *Traveling along a blue-tape figure 8 track multiple times*
* *Changing LED colors*
* *Programming robot to “speak” certain phrases*
* *Programming directional changes as well as delays*

Out of Scope

* *Courses other than the blue-tape figure 8*
* *Actions not specified in the requirements*

# Product/Service Description

## Product Context

This product is the second 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*

## 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*

## 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 figure 8 track* *and must complete the course 5 times*
* *Sphero Edu application must be used to program robot*

# Requirements

## Functional Requirements

| Req# | Requirement | Comments | Priority | Date Rvwd | SME Reviewed / Approved |
| --- | --- | --- | --- | --- | --- |
| ACCUR\_01 | The robot will start and end in the center of the Figure 8 on the blue tape. | The location of the robot is important to ensure the robot will stay on track. | 1 | 11/9/22 | Aaila Arif |
| ACCUR\_02 | A variable will be created for the robot’s speed (robotSpeed), robot’s spin angle (robotSpinAngle), robot’s spin time (robotSpinTime), and robot’s color fade (fadeTime). | These variables will help us manage and troubleshoot our robot. The measurements will be determined during the test plan. | 2 | 11/9/22 | Aaila Arif |
| ACCUR\_03 | Set Sphero’s speed to robotSpeed. | Measurements will be determined during the tests. | 2 | 11/9/22 | Aaila Arif |
| ACCUR\_04 | Spin Sphero for robotSpinAngle degrees for robotSpinTime seconds. | To make the robot spin in a circle, the degrees will be set to 360°. | 1 | 11/9/22 | Aaila Arif |
| ACCUR\_05 | Sphero will stop for about a second. | Stopping the robot will ensure the robot doesn’t go off track. | 2 | 11/9/22 | Aaila Arif |
| ACCUR\_06 | Spin Sphero in negative robotSpinAngle degrees for robotSpinTime seconds. | To make the robot spin in a circle, the degrees will be set to -360°. | 1 | 11/9/22 | Aaila Arif |
| ACCUR\_07 | Sphero will stop again for about a second. | Stopping the robot will ensure the robot doesn’t go off track. | 2 | 11/9/22 | Aaila Arif |
| ACCUR\_08 | Repeat steps ACCUR\_03 through ACCUR\_7 five times. | A loop will be more straightforward instead of re-coding this program 5 times. | 1 | 11/9/22 | Aaila Arif |
| ACCUR\_09 | When Sphero is finished with the loop, the robot will begin fading its lights to colors of the rainbow. Each color will fade to the next color for about 0.125 seconds equaling 1 second. | Added collectively by team members. | 3 | 11/9/22 | Aaila Arif |
| ACCUR\_10 | Repeat step ACCUR\_9 five times. | A loop will repeat the color fade program 5 times so it will equal 5 seconds. | 3 | 11/9/22 | Aaila Arif |
| ACCUR\_11 | At the end, Sphero will speak, “I am the winner!” and will end the program. | The device’s volume must be on, so we can hear Sphero. | 1 | 11/9/22 | 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

|  |  |  |
| --- | --- | --- |
| Meeting Date | Attendees (name and role) | Comments |
| 11/09/22 | Dominick Del Bene (Requirements Manager), Aaila Arif (Project Executive), and Zaccery Tarver (Software Designer) | Confirmed all. |
| 11/11/22 | Dominick Del Bene (Requirements Manager) and Aaila Arif (Project Executive) | Confirmed all after revising it. |

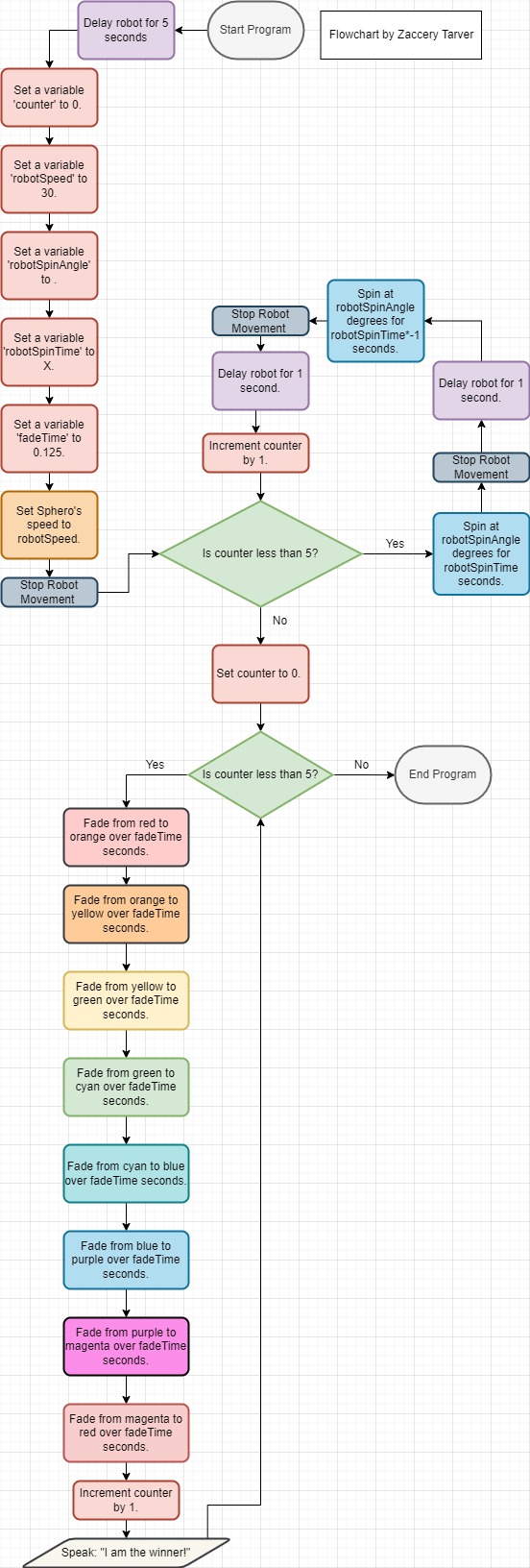
# System Design

## Algorithm

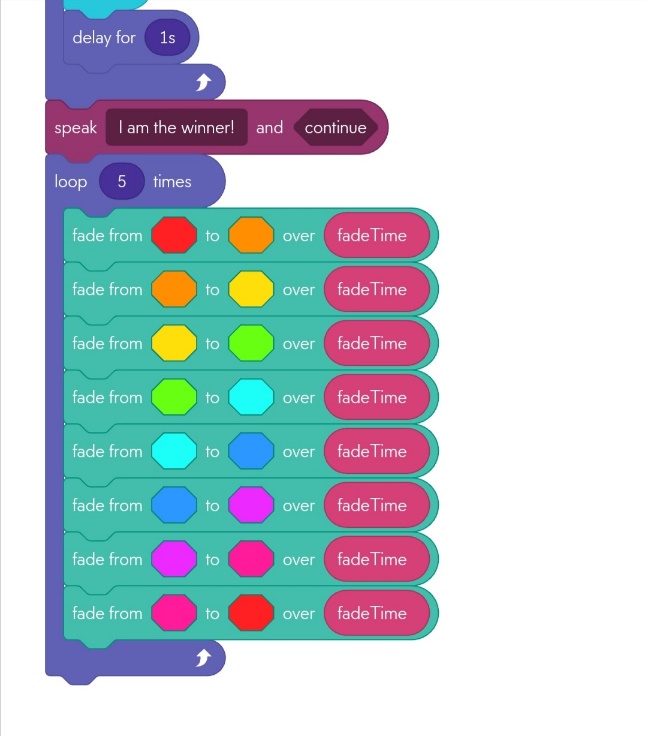
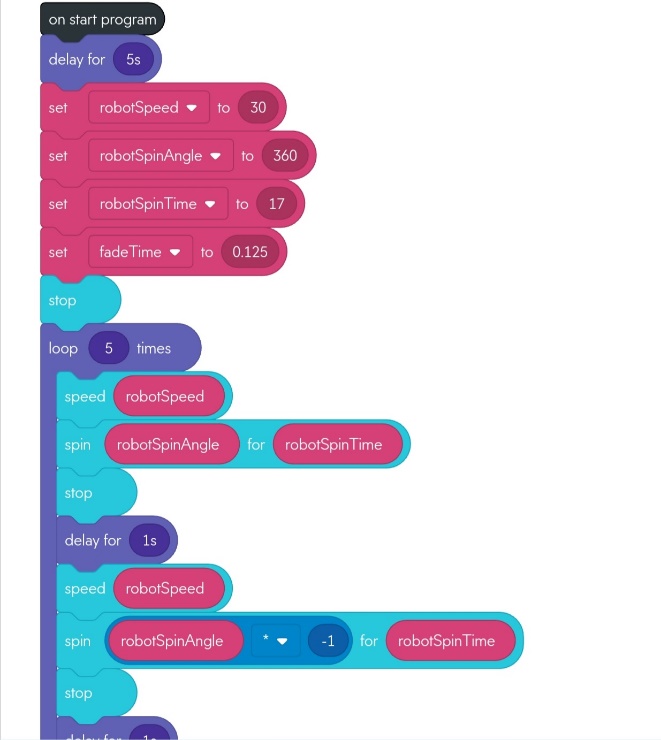
* Note: Use Spin and Speed blocks to make the robot go in a circle

1. Delay robot for 5 seconds.
2. Set a variable ‘robotSpeed’ to 30.
3. Set a variable ‘robotSpinAngle’ to 360.
4. Set a variable ‘robotSpinTime’ to 17.
5. Set a variable ‘fadeTime’ to 0.125.
6. Set Sphero’s speed to robotSpeed.
7. Stop robot movement.
8. Repeat this step 5 times:
   1. Spin at robotSpinAngle degrees for robotSpinTime seconds.
   2. Stop robot movement.
   3. Delay robot for a full second.
   4. Spin at robotSpinAngle \*-1 degrees for robotSpinTime seconds.
   5. Stop robot movement.
   6. Delay robot for a full second.
9. Speak ‘I am the winner!’
10. Repeat this process 5 times:
    1. Fade from red to orange over fadeTime seconds.
    2. Fade from orange to yellow over fadeTime seconds.
    3. Fade from yellow to green over fadeTime seconds.
    4. Fade from green to cyan over fadeTime seconds.
    5. Fade from cyan to blue over fadeTime seconds.
    6. Fade from blue to purple over fadeTime seconds.
    7. Fade from purple to magenta over fadeTime seconds.
    8. Fade from magenta to red over fadeTime seconds.
11. End the program.

## System Flow



## Block Code



## Software

* Sphero Block Code
* Sphero Edu

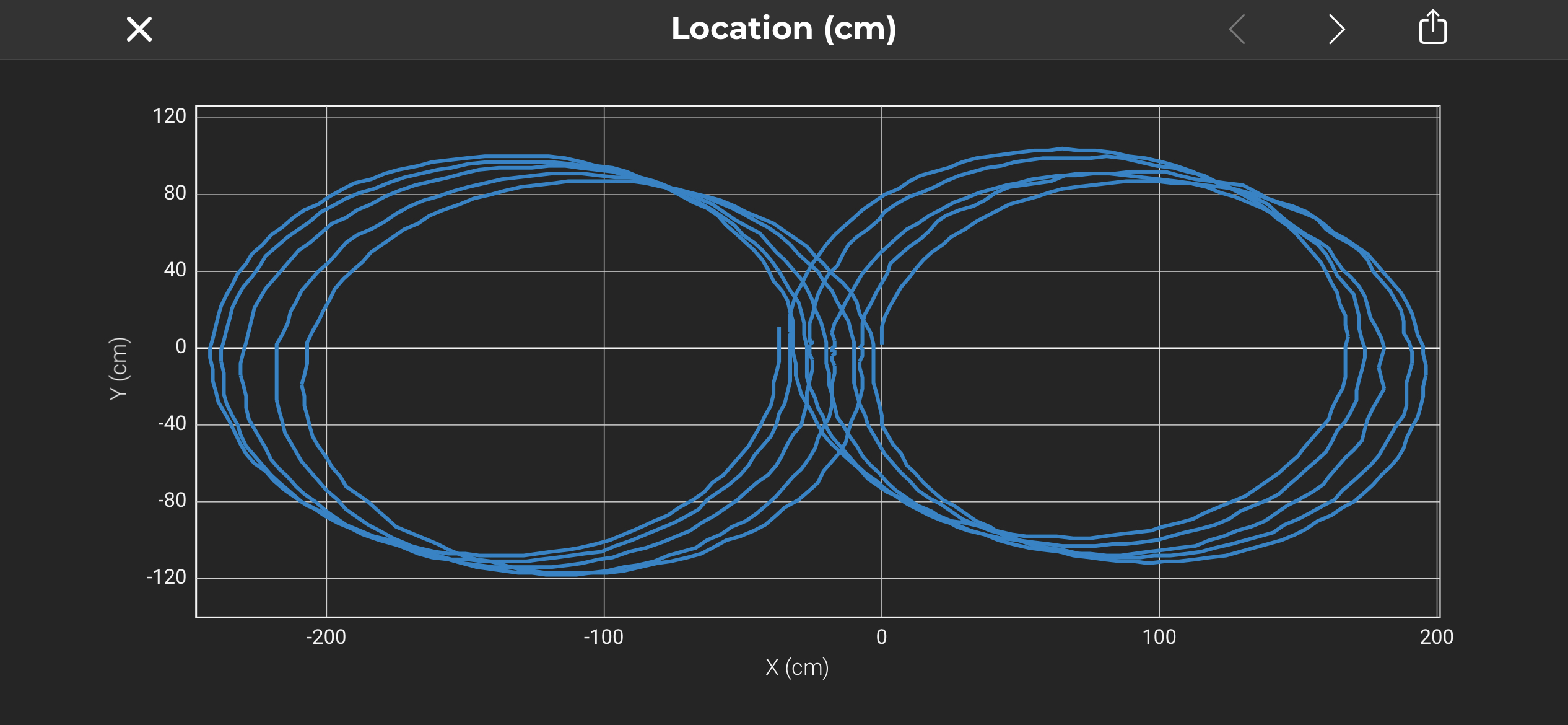
## Hardware

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

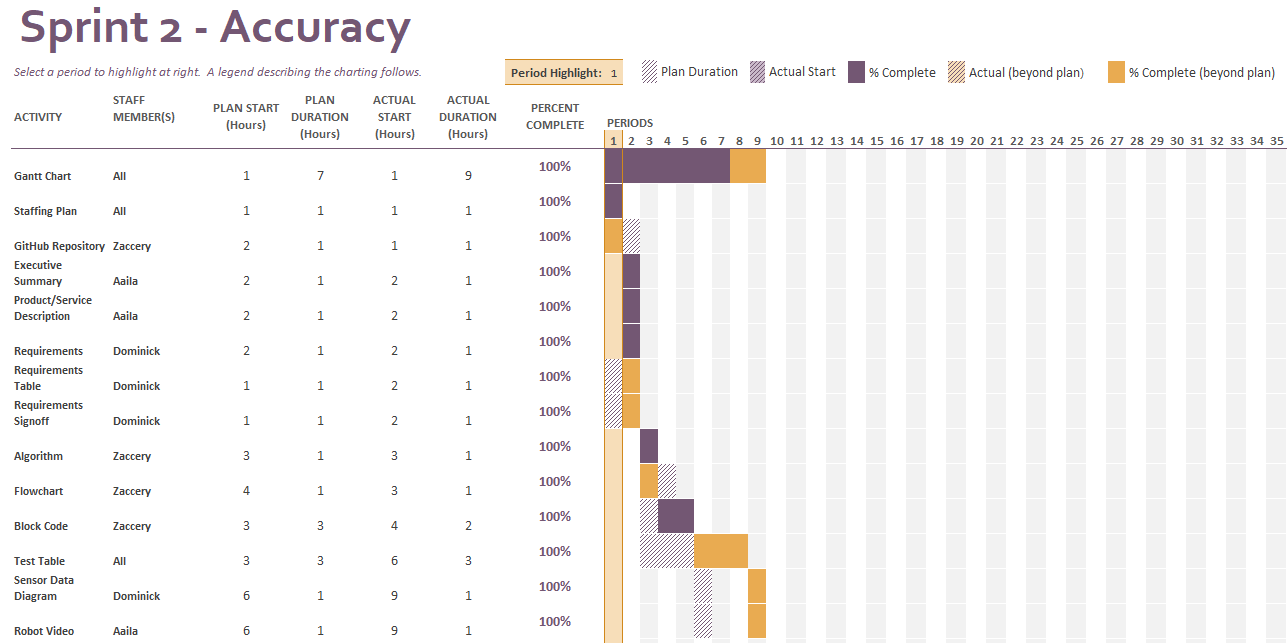
## Test Plan

| **Reason for Test Case** | **Test Date** | **Expected Output** | **Observed Output** | **Staff Name** | **Pass/Fail** |
| --- | --- | --- | --- | --- | --- |
| Initial trial and error test to determine the variables of speed, angle, and time. Testing 50, 200, and 8 respectively. | 11.9 | No level of accuracy is expected, this test is just a benchmark. | The initial heading greatly determines the path of travel, which will be adjusted accordingly. It also seems that the speed must be set before every instance of a spin, which will be fixed. Much was learned from this test, so it was successful. | Zaccery Tarver | Pass |
| Second test to determine the variables of speed, angle, and time. Testing 50, 200, and 8 respectively. Initial heading and speed problems are fixed. | 11.9 | The robot should complete an inaccurate figure-eight. | The robot did not fully complete a figure-eight but rather moved in strange ‘S’ like shapes.  It stuck to the path quite accurately at the start, though. It veered off to the left while the program went on, however. | Zaccery Tarver | Pass |
| Third test to determine the variables of speed, angle, and time. Testing 60, 180, and 16 respectively. I want the robot to move a little faster, which is why the speed increased. It needs to complete the first circle of the figure-eight, which is why the time for the movement was doubled. | 11.9 | The robot should complete a somewhat accurate figure eight.  We need to see what effect does lowering the angle has on the path accuracy. | The robot went way off course, it seems that lowering the angle made the robot veer extremely left. The robot should therefore be given an angle above 200. | Zaccery Tarver | Fail |
| Fourth test to determine the variables of speed, angle, and time. Testing 60, 240, and 16 respectively. By increasing the angle, the robot should not veer less anymore since it was previously veering left. | 11.9 | The robot should complete a rough figure eight, and potentially end where it started. | The robot still veered left. The angle may have increased by too little since the speed was also raised. | Zaccery Tarver | Fail |
| Fifth test to determine the variables of speed, angle, and time. Testing 60, 540, and 16 respectively. By increasing the angle extremely, we will be able to test the extremities of the change in behavior caused by this instruction. | 11.9 | Robot should veer right instead of left, given the drastic increase in angle. | The robot was very accurate, staying on the line, but it eventually veered left.  We now understand that angle needs to be raised dramatically to get the result we want.  The robot also stayed on its path for too long. | Zaccery Tarver | Pass |
| Sixth test to determine the variables of speed, angle, and time. Testing 60, 800, and 9 respectively. An extreme increase in angle should make robot be accurate. Also, lowering the time should make it end approximately where it started. | 11.9 | Robot should end initial circle at its start point and not veer left. | The robot went at extreme angles right. Angle needs to be lowered. | Zaccery Tarver | Fail |
| Seventh test. Old values are the same, testing a new angle: 600. | 11.9 | Robot should not veer extremely right. | The robot still veers right. | Zaccery Tarver | Fail |
| Eight test. New angle, 570, which is in between 540 and 600. This should be a medium between two slightly inaccurate values, hopefully giving a correct | 11.9 | Robot should stay on course. | The robot does not stay on the path of travel too well. | Zaccery Tarver | Fail |
| After several tests, an issue was found with the code. The angle being used was not 360 degrees. This is the only angle to get the robot to spin in a singular circle. This test case is meant to prove that by switching the angle to 360. | 11.11 | Robot should end up where it started, because if it moves in one circle, then it should not be displaced from the origin at all. | The robot ends up in its approximate original start location. | Aaila Arif | Pass |
| Now, it was only a matter of getting the correct speed and time. It was noted that by decreasing time, the radius of the circular motion was shortened. But when speed decreased, the radius would not change. It was a matter of trial-and-error approximation to find the correct ratio between time and speed so that Sphero would complete the motion as accurately and quickly as possible. Testing for certain values of speed and time. | 11.11 | Sphero should move in its circular motion faster than the previous test. | The robot does not stay on course well enough but does move in a circle still. | Aaila Arif | Fail |
| After making several adjustments to speed and time, the correct ratio was found between them respectively: 30 and 17. Testing for the video. | 11.11 | The robot should be able to complete a figure-eight motion, say ‘I am the winner,’ and flash multicolored lights. | The robot fulfills the requirements and is fully functional. It completes the proper motion and end sequence. | Zaccery Tarver | 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 robot’s sensor data. | Aaila |
| Zaccery Tarver | Software Designer | - To manage and organize the GitHub repository.  - To build the algorithm.  -To construct the flowchart.  - To use the algorithm to construct the block-code. | Aaila |