Sprint 2 - Accuracy Design Document

April 11th, 2024

Table of Contents

1. Executive Summary

1.1 Project Overview

This project is to showcase the new and improved functionality of the Sphero robots. This product is intended for anyone who is learning to code either as a profession or a hobby. In the following sections you will find the general makeup of the testing on the Sphero robots that have been conducted onsite, as well as external research into the making of these robots. There will also be a proposal of unique ways to use this software outside of learning moving forward in the future.

1.2 Purpose and Scope of this Specification

The purpose of this specification is to provide clear guidelines and requirements for the design and development of our team's Sphero robot project. It outlines the functionality, constraints, and considerations that need to be addressed during this project's lifecycle.

In scope

Functional Requirements: Ensure the robot completes the obstacle course.

Design Constraints: All programming is done through the Sphero Robot application.

Hardware/Software: This document explains the difference in using different software and the most compatible as well as the required hardware.

Testing: Document provides an in-depth test table with each test our team has run.

Out of Scope

Coding Application: This document does not go into depth with how the Sphero Application coding was developed.

Robot Design: This document does not go into the Sphero robots' design or how they are made.

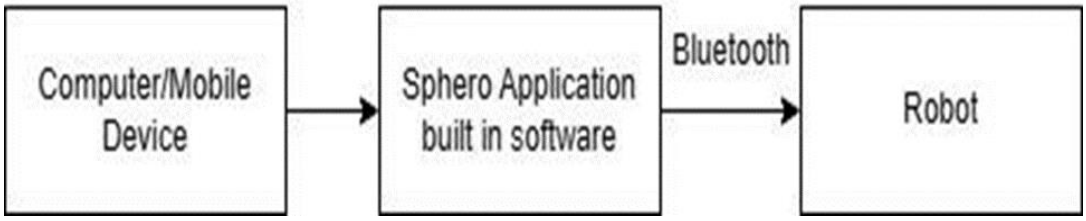
Sales Pitch: It is not our intention to sell the reader on anything, this document is simply research conducted by our team utilizing Sphero Robots and their software.

2. Product/Service Description

To ensure proper functionality of the robots, the proper operating system is paramount. Due to the developments made by the Sphero team, the Robots work best with Mac OS but do offer operating capabilities with Windows or Linux OS. The Sphero program utilizes block coding for easy-to-use development which reduces development time exponentially.

2.1 Product Context

The Sphero robots utilize state of the art technology for easy-to-use coding with preset "blocks" that can be used to set the robots function as the user sees fit. The robots utilize similar but simplified software that the Roomba Vacuum or RC cars utilize. The Sphero robots are self-contained in that each component is designed to operate that single individual robot. Sphero does work with a variety of related systems including Computer/Mobile Devices via Bluetooth. These computer/mobile devices link directly to the robot and can control the robot as coding is done via the Sphero application and not onto the robot itself.



2.2 User Characteristics

Student:

Role: Student

Experience: Varies based upon users' knowledge.

Technical Expertise: Depends on if the student has experience/is familiar with operating a smartphone or laptop. Typically ranges from low to moderate.

Other Characteristics: Creativity and willingness to learn. Students may use the Sphero Robots to understand creating an algorithm and putting their on-paper tests into physical ones.

Faculty/Staff:

Role: Educator

Experience: Varies, but generally more experienced compared to students. Faculty may have some background in education, technology, or STEM fields.

Technical Expertise: Ranges from moderate to high. Faculty and staff may have experience with educational technology, programming, and curriculum development.

Other Characteristics: Interest in incorporating innovative teaching methods and technologies into their courses. Allows freedom for students to learn proper project management with little interference.

Other:

Role: Individuals either in STEM related fields or enjoy computing as a hobby.

Experience: Varies ranging from complete beginners to experienced individuals with a passion for technology.

Technical Expertise: Varies based upon individuals background, interest, and experience with programming or robotics.

Other Characteristics: Interest in technology and learning. Users may use the Sphero robots for recreational purposes or to learn more about programming and robotics.

2.3 Assumptions

Internet Connection- The user must ensure they are connected to the internet via Wi-Fi, hotspot, or cellular data to ensure proper Bluetooth connection to the robot.

Operating System- The Sphero Robots work best with Mac OS but are compatible with Windows, Linux, Android, and Chromebook as well with reduced functions.

Physical Environment- The robot is set to traverse a certain obstacle course and if the course were to change then the code would have to be adjusted to fit.

Age and Experience of User- The simple and easy to use application provides users with an efficient way to design the robots' functions, erasing many abstractions that can be present in coding.

2.4 Constraints

Old System Compatibility: While the Sphero application works best on Mac OS, it can be run on older systems with limited functions. For the best operation you must ensure the most up to date OS and application are downloaded.

Audit Functions: The original code is posted to Github for public viewing, but the SDD is embedded so that every change to the document found can be sourced back to who made the change.

Access and Security: Due to the block code and research being posted to Github, all this information is open source for anyone to use.

Importance of the Project: While not groundbreaking, our research is to show the simplicity of the Sphero application. As our team is new to the world of coding, this project shows how easy the Sphero robots can be to grasp the concept of coding.

Resource Limits: The Sphero Robots utilize a basic software that can be run on a variety of devices if it has the ability to download external applications and connect to wife/Bluetooth.

Design Standards: The Sphero Robots must be programmed within the Sphero application. The Sphero application utilizes block coding and presets that apply directly to Sphero robots. The built program can be used on any Sphero Robot assuming proper Wi-Fi connection to connect the robot via Bluetooth.

2.5 Dependencies

The Sphero Robots act based on the code provided to it via the Sphero application. To ensure proper functionality, the user must ensure the Sphero application is up to date. The proper program can either be downloaded via Github or the web or created from scratch based on the user's preference and tasking.

3. Requirements

To begin this project, all requirements must be described in adequate detail to satisfy further steps of the project. The following are the requirements in detail which satisfy the project guidelines:

A.) Robot must start on the provided square.
B.) Robot must complete one figure eight
C.) Robot must stay within the desired path
D.) Robot must repeat step b for a total of 5 laps
E.) Robot must finish on the provided starting square.
F.) Robot must speak "I am the winner!"
G.) Robot must flash multicolored for a total of 5 seconds.

Requirement #	Priority	Input	Function	Output
1	**1**	Physically placing the robot on the starting square	Physically place robot on starting square	Robot begins on yellow square
2	**1**	Robot must be able to move	Set robots speed to 80	Robot will move at a speed of 80
3	**1**	Code robot to complete a figure eight movement	Robot must spin 360° for 9.1 seconds	Robot completes a full 360° spin
4	**1**	Code robot to complete a figure eight movement	Robot must spin -360° for 9.2 seconds	Robot completes a full 360° spin the opposite direction
5	**1**	Code robot to repeat both spins for a total of 5	Set loop = 0, adding 1 after each full figure eight	Robot continues to repeat movements until a total of 5 figure eights are made
6	**1**	Code robot to stop and say “I am the winner”	Robot makes its final stop in which it started at, and says “I am the winner”	Robot completes final stop and speaks
7	**1**	Code robot to move flash multicolored	Robot flashes multicolored	Robot flashes multicolored

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
ENDUR_01	Robot must start on provided square	Start	1	02APR24	Chris K
ENDUR_02	Robot must complete one figure eight	First lap	1	02APR24	Chris K
ENDUR_03	Robot must stay within desired path	Ensure Accuracy of robot’s movements	1	02APR24	Chris K
ENDUR_04	Robot must repeat same movements as ENDUR_02	Repeat movements for a total of 5 laps	1	02APR24	Chris K
ENDUR_05	Robot Must finish on starting square	End	1	02APR24	Chris K
ENDUR_06	Robot must speak “I’m the winner!”	Speak	1	02APR24	Chris K
ENDUR_07	Robot must flash multi-colored	Flash colors	1	02APR24	Chris K

3.2 Security

3.2.1 Protection

The use of GitHub, Google Docs, and Sphero.edu incorporates protective measures like encryption, activity logging, and data integrity checks. GitHub encrypts code repositories to prevent unauthorized access, while activity logging in both GitHub and Google Docs tracks changes for detection of unauthorized modifications. These measures collectively defend against malicious or accidental threats, bolstering system security.

3.2.2 Authorization and Authentication

To fortify security measures for our project, we utilized GitHub and Google Docs to enforce strict authorization and authentication protocols. GitHub served as our primary platform for version control and collaboration, allowing only authorized team members to modify our code repositories, ensuring maximum safety. Meanwhile, our System Design Document on Google Docs featured authentication features such as timestamps and user verification, providing clear accountability for any changes made. By integrating these measures, we bolstered security, safeguarding our data and ensuring the integrity of our project.

3.3 Portability

GitHub, Google Docs, and Sphero.edu are built with portability in mind. They use code that can easily work on different machines and operating systems. These platforms prioritize simple languages that are known to work well everywhere. Sphero.edu is especially good at working the same way no matter where it's used. The robots are made to be portable as they come with a carrying case, allowing for easy access to bring anywhere. This also means they can be used on different computers without any problems, making them flexible and easy to use across various setups.

4. Requirements Confirmation/Stakeholder sign-off

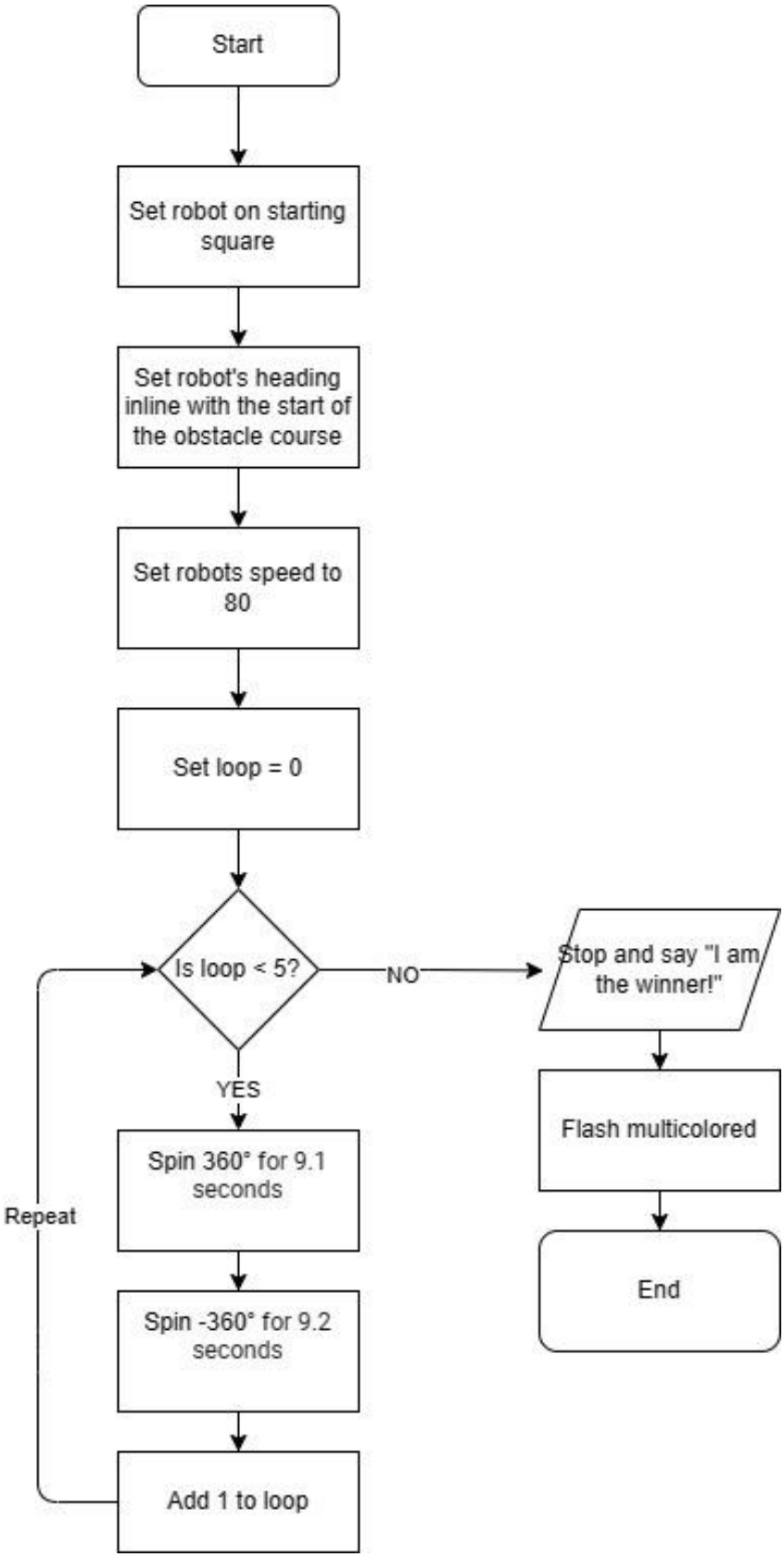
Meeting Date	**Attendees (name and role)**	**Comments**
04/02/24	Chris K and Aaron G	Confirmed all requirements
04/05/24	Chris K and Aaron G	Confirmed algorithm and flowchart
04/0924	Chris K and Aaron G	Confirmed code and recorded video

5. System Design

5.1 Algorithm

A.) Robot must start on the provided square
B.) Set robot’s heading in line with the start of the obstacle course
C.) Set Robots Speed to 90
D.) Spin 360 for 9.1s
E.) Spin -360 for 9.2s
F.) Loop Steps D-F for a total of 5 laps
G.) Speak “I am the winner!”
H.) Flash Multicolored

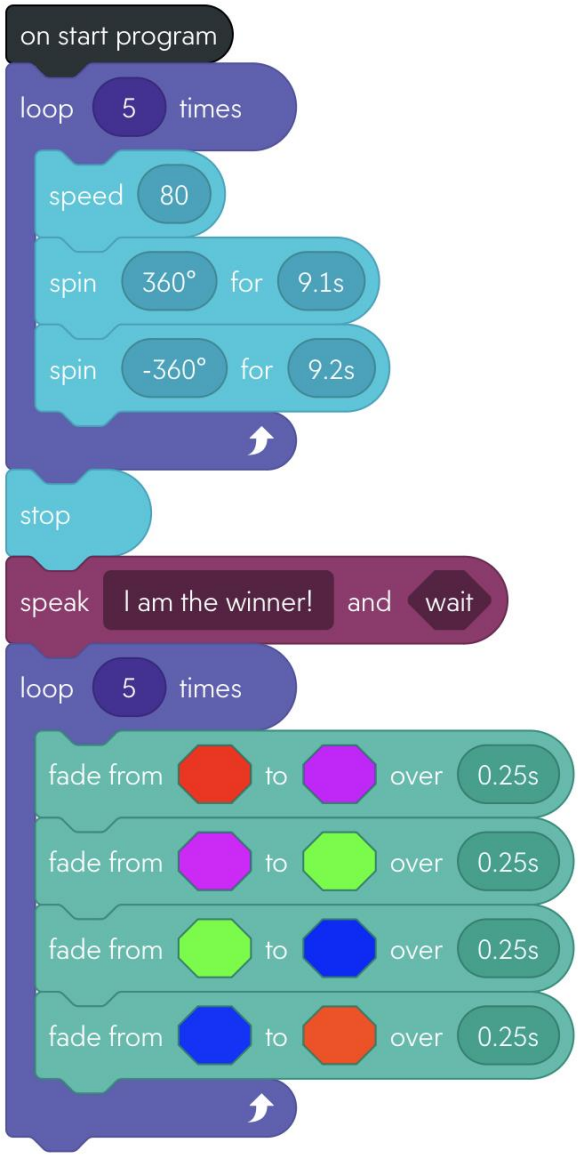
5.2 System Flow



5.3 Software

Mac Operating System for optimal results and feedback

Sphero Robot application for coding



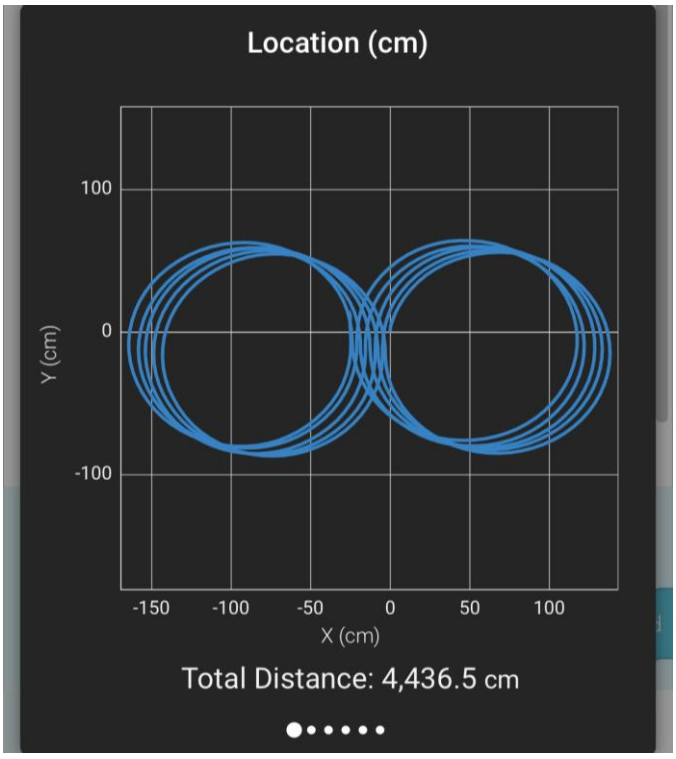
5.4 Hardware

Apple Mac computer for coding and accurate results following tests

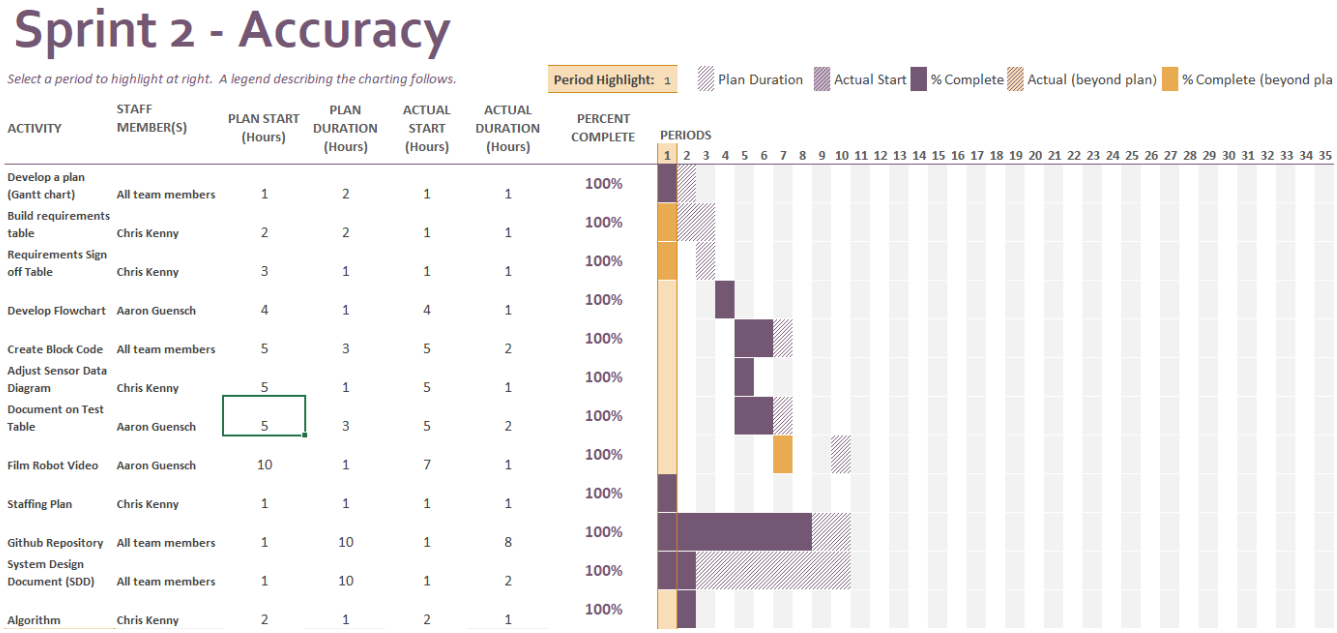
Sphero robot

5.5 Test Plan

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Ensure spins in circle	05APR24	Completes full circle	Completes full circle	Chris K Aaron G	Pass
Ensure spins full circle in opposite direction	05APR24	Completes full circle in opposite position	Completes full circle in opposite position	Chris K Aaron G	Pass
Ensure completes full figure eight	05APR24	Completes circle back and forth creating a figure eight	Robot died mid testing	Chris K Aaron G	Fail
Ensure completes full figure eight	09APR24	Completes circle back and forth creating figure eight.	Completes circle back and forth creating figure eight	Chris K Aaron G	Pass
Ensure speaks “I am the winner”	09APR24	Says “I am the winner!”	Says “I am the winner!”	Chris K Aaron G	Pass
Robot flashes multi-colored	09APR24	Robot flashes a multitude of colors	Robot flashes a multitude of colors	Chris K Aaron G	Pass
Record full video	09APR24	Robot completes full 5 laps and speaks/turns different colors	Robot skewed off course by the 3rd loop	Chris K Aaron G	Fail
Record full video	09APR24	Robot completes full 5 laps and speaks/turns different colors	Robot completes full 5 laps and speaks/turns different colors	Chris K Aaron G	Pass



5.6 Task List/Gantt Chart



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	1	100%
Build requirements table	Chris Kenny	2	2	1	1	100%
Requirements Sign off Table	Chris Kenny	3	1	1	1	100%
Develop Flowchart	Aaron Guensch	4	1	4	1	100%
Create Block Code	All team members	5	3	5	2	100%
Adjust Sensor Data Diagram	Chris Kenny	5	1	5	1	100%
Document on Test Table	Aaron Guensch	5	3	5	2	100%
Film Robot Video	Aaron Guensch	10	1	7	1	100%
Staffing Plan	Chris Kenny	1	1	1	1	100%
Github Repository	All team members	1	10	1	8	100%
System Design Document (SDD)	All team members	1	10	1	2	100%
Algorithm	Chris Kenny	2	1	2	1	100%

5.7 Staffing Plan

Name	Role	Responsibility	Reports To
Chris Kenny	Leader/Developer	Develop, Ensure smooth operations, complete work	Professor Eckert
Aaron Guensch	Co-Leader/Developer	Develop and complete work	Chris
Skylyn	Quit	None	Themselves