Sprint 3- Agility Design Document

April 22, 2020

# Executive Summary

### 1.1 Project Overview

The Accuracy sprint is intended for use in CS 104-01. The project calls for a program written in block code in the Sphero Edu app. The intended function of the program is to guide a robot to complete an obstacle course. **Specific requirements from Prof. Eckert:**

Your robot will run the obstacle course. The course will start in a square. Then the robot will encounter 3 objects which it must avoid.. Next, the robot will go over the ramp. Finally, the robot will knock over as many pins as possible. Points added for each obstacle the robot completes, for each obstacle avoided and, for each pin the robot topples.

1.2 Purpose and Scope of this Specification

**In scope**

This module includes the following:

* A program for the SPRK robot provided for classroom use
* Block code compatible with the Sphero Edu app
* Completion of the Agility sprint from the overarching Robotics Project

**Out of Scope**

This module excludes the following:

* Code compatible with non-Sphero robots
* Specifications listed outside of the Agility sprint

# Product/Service Description

### 2.1 Product Context

The project relates to other products because a SPRK robot is required to run the program. A Bluetooth connection is also necessary for the robot to communicate with the device running the program.

### 2.2 User Characteristics

* CS 104-01 students or Prof. Eckert
* Basic understanding of the Sphero Edu app and block code
* No technical expertise required

### 2.3 Assumptions

The equipment necessary to run the program is available, including: Sphero Edu app, SPRK robot, HH208. The user has a general understanding of how to use the Sphero Edu app and the SPRK robot.

### 2.4 Constraints

* Block code is only compatible with the Sphero products
* The program cannot be used for robots not affiliated with the Sphero brand

### 2.5 Dependencies

The functionality of the program depends on access to the SPRK robot. Without the robot, the program cannot have the desired output. The program also depends on the Bluetooth connection between the computer and the robot so that the computer can communicate with the robot.

# Requirements

1. Priority 1
   1. Robot must avoid three objects.
   2. Robot must go over the ramp.
   3. Robot must collide with pins.
2. Priority 2
   1. Robot must not collide with anything but the pins during the circuit.
3. Priority 3
   1. Robot should knock down more than one pin.
   2. Robot should generally perform the same each time the program runs.

### 3.1 Functional Requirements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requirement Number** | **Requirement** | **Comments** | **Priority** | **Date Reviewed** | **Reviewed/Approved** |
| 1a | Robot must avoid three objects. | Only Neil has access to the robot, so he will need to ensure that the robot successfully completes this requirement. | 1 | 4/21/2020 | AN, NS, CV, AW |
| 1b | Robot must go over the ramp. | Slower speeds ensure that the robot executes the program with more precision, but speed is required to get the robot over the ramp. | 1 | 4/21/2020 | AN, NS, CV, AW |
| 1c | Robot must collide with pins. | The robot needs enough momentum to hit the pins and hopefully knock them down. | 1 | 4/21/2020 | AN, NS, CV, AW |
| 2a | Robot must not collide with anything but the pins during the circuit. | Accuracy is required to make sure the robot does not deviate from its path. | 2 | 4/21/2020 | AN, NS, CV, AW |
| 3a | Robot should knock down more than one pin. | Needs to move fast enough and be centered on the pins to knock them down. | 3 | 4/21/2020 | AN, NS, CV, AW |
| 3b | Robot must flash multicolored lights at the end of the program. | Carpet may need to be used, which could make the program less accurate. | 3 | 4/5/2020 | AN, NS, CV, AW |

### 3.2 Security

#### 3.2.1 Protection

Protection is not a concern with this project because our program cannot be misused or cause harm to anything. Malicious access is highly unlikely and the program will be made private on the Sphero Edu app so that it is protected against plagiarism by other students.

#### 3.2.2 Authorization and Authentication

The program is only accessible through our GitHub repository and one of our accounts on Sphero Edu. There is no authorization or authentication required to access our repository, but there is authentication required to access the program on Sphero Edu because only the owner can access it.

### 3.3 Portability

The program is only portable across Sphero’s line of robots as it is written in block code only compatible with Sphero robots. Any operating system can be used to access the Sphero website. The robot can execute this program anywhere, so long as all of the equipment is available and there is ample space for the robot to move in.

# Requirements Confirmation/Stakeholder sign-off

|  |  |  |
| --- | --- | --- |
| Meeting Date | Attendees | Comments |
| 4/21/2020 | Anna Nardelli  Neil Swagger  Connor Vidnansky  Andrew Welde | Approved all requirements |

1. System Design

### 5.1 Algorithm

1. The robot rolls for 2 seconds forward
2. The robot stops turns 270 degrees and rolls for 2 seconds
3. It stops, turns back to 0 degrees and then begins to roll again
4. The robot turns 90 degrees and begins to roll for 2 seconds
5. Repeat steps 3-4
6. The robot returns to facing 0 degrees and moves forward for 2 seconds
7. It stops and then increases speed for 5 seconds
8. Finally stops

### 5.2 System Flow

Uploaded to GitHub

### 5.3 Software

The software used to develop the program was the Sphero Edu app and Mac OS.

### 5.4 Hardware

The hardware used to develop the program was a personal computer and a SPRK robot.

### 5.5 Test Plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Reason for Test Case** | **Test Date** | **Expected Output** | **Observed Output** | **Staff Name** | **Pass/Fail** |
| To go straight then stop before hitting obstacle then go around obstacle | 4/21/20 | Avoids obstacle | Avoided obstacle | Neil | Pass |
| Avoid second obstacle | 4/21/20 | Avoids obstacle | Avoided obstacle | Neil | Pass |
| Avoid second obstacle | 4/21/20 | Avoids obstacle | Avoided obstacle | Neil | Pass |
| Go over ramp | 4/21/20 | Goes over ramp | Made it over | Neil | Pass |

### 5.6 Task List/Gantt Chart

Uploaded to GitHub

### 5.7 Staffing Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Responsibility** | **Reports To** |
| Anna Nardelli | Leader | Sections 1-4 of Design Document, Gantt Chart, making sure everyone else contributes | Everyone |
| Neil Swagger | Owner of robot | Type the program for the robot | Everybody |
| Connor Vidnansky | Design Document | Create the flowchart, approve of changes to code/Design Document | Everyone |
| Andrew Welde | Design Document | Write algorithm | Everyone |