**[**Sprint 3-Agility**] Design Document**

November 15**, 2019**

CS 104-03

Holly Bernich

Nolan Beagell

Anthony Berardis

Professor Eckert

Fall 2019 Semester

**Table of Contents**

[**1.**](#_heading=h.1fob9te) **EXECUTIVE SUMMARY 3**

[1.1](#_heading=h.1fob9te) Project Overview 3

[1.2](#_heading=h.3znysh7) Purpose and Scope of this Specification 3

[**2.**](#_heading=h.2et92p0) **PRODUCT/SERVICE DESCRIPTION 3**

[2.1](#_heading=h.3as4poj) Product Context 3

[2.2](#_heading=h.3dy6vkm) User Characteristics 3

[2.3](#_heading=h.1t3h5sf) Assumptions 3

[2.4](#_heading=h.4d34og8) Constraints 3

[2.5](#_heading=h.2s8eyo1) Dependencies 4

[**3.**](#_heading=h.17dp8vu) **REQUIREMENTS 4**

[3.1](#_heading=h.1pxezwc) Functional Requirements 5

[3.2](#_heading=h.26in1rg) Security 5

[*3.2.1*](#_heading=h.lnxbz9) *Protection 5*

[*3.2.2*](#_heading=h.35nkun2) *Authorization and Authentication 6*

[3.3](#_heading=h.1ksv4uv) Portability 6

[**4.**](#_heading=h.49x2ik5) **REQUIREMENTS CONFIRMATION/STAKEHOLDER SIGN-OFF 6**

[**5.**](#_heading=h.z337ya) **SYSTEM DESIGN 6**

[5.1](#_heading=h.3j2qqm3) Algorithm 6

[5.2](#_heading=h.1y810tw) System Flow 6

[5.3](#_heading=h.4i7ojhp) Software 6

[5.4](#_heading=h.2xcytpi) Hardware 6

[5.5](#_heading=h.1ci93xb) Test Plan 7

[5.6](#_heading=h.3whwml4) Task List/Gantt Chart 7

[5.7](#_heading=h.2bn6wsx) Staffing Plan 7

# 

# 

# 1. Executive Summary

## ***Project Overview***

This software project is the first step of our software project. The entire project is to design software for our robot to complete a series of sprints; speed, agility, and speed. The intended audience of this project is the CS 104 class and Professor Eckert, along with anyone else in the Monmouth University Computer Science Department.

## ***Purpose and Scope of this Specification***

The purpose of this part of the project is to be able to complete the first sprint which is speed. The intended audience of the first sprint is Professor Eckert, the CS 104 class, and any other person who is a part of the Computer Science and Software Engineering Department.

**In scope**

This document addresses requirements related to phase 3 of the Robot Project:

* modification of javascript code along with sphero plug in code in order for the robot to be able to beat the agility test

**Out of Scope**

The following items in phase 1 and 2 of the Robot Project are out of scope:

* modification of javascript and plug sphero code to meet the expectations of the endurance test.
* modification of javascript and plug sphero code to meet the expectations of the speed test.

(Phases ( endurance and speed) will be considered in the development of the requirements for Phase 1 and 2, and these requirements will be documented separately.)

# Product/Service Description

The general factors that affect the agility code for the Robot Project include the ability to write the code for the sprint, and the ability of the robot to complete the agility track given to the class by the professor. The code for this level of the project should allow the robot to complete the agility test.

## ***Product Context***

This product relates to other products because for the next two sprints, we will be working on doing two more similar things with the robot, such as endurance and speed. This product is independent, because it does not rely on the other tests in order to run and function and complete the course.This product relates to other products because it has to pass a test in order to run.

***User Characteristics***

* Anyone interested in learning or studying programming
* Ability to use Sphero program
* Use of basic computer block programming

## ***Assumptions***

Equipment required for this project is a sphero robot, charging case, charging cable, robot cover, and robot case. Availability of Howard Hall 208 also affect the requirements of the project, since the course is in that room. Little to no user expertise required because it is a guess and check system using the angles of a protractor and course angles.

## ***Constraints***

Describe any items that will constrain the design options, including

* Deadline November 26, 2019
* Access to campus after normal school hours
* Availability of Howard Hall 208
* Group availability

## ***Dependencies***

* This new product will require a download of the program to the robot each time the code is changed in order to run it properly.
* Agility program must be completed before the robot can run, and must be downloaded to the robot in order to run it.

# Requirements

## ***Functional Requirements***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Req#** | **Requirement** | **Comments** | **Priority** | **Date Rvwd** | **SME Reviewed / Approved** |
| AGILITY\_01 | Robot should start on the X tape in order to complete the course. | This is the start of the start of the Agility test. | 3 | 11/22/19 | Holly Bernich  Nolan Beagell  Anthony Berardis |
| AGILITY\_02 | The robot should continue from the path and not stray from the course. | Speed and accuracy affect this part of the project. | 1 | 11/22/19 | Holly Bernich  Nolan Beagell  Anthony Berardis |
| AGILITY\_03 | Robot should not hit any obstacles on the course. | Important procedure for proper run of program. | 2 | 11/22/19 | Holly Bernich  Nolan Beagell  Anthony Berardis |
| AGILITY\_04 | Robot needs to land around the next starting point to knock pins down after binder jump ramp. | Important part for next start of the code, can adjust aim. | 3 | 11/22/19 | Holly Bernich  Nolan Beagell  Anthony Berardis |
| AGILITY\_05 | Robot will need to continue off landing point in a straight line to knock down the markers. | Almost end of program | 2 | 11/22/19 | Holly Bernich  Nolan Beagell  Anthony Berardis |
| AGILITY\_06 | Robot has completed the Agility test | Completion of test required for this project. | 1 | 11/22/19 | Holly Bernich  Nolan Beagell  Anthony Berardis |

## 

## 

## ***Security***

### **Protection**

* Private code on our accounts
* Robots are sign out only
* Robots are locked up
* Silicone cover on the outside of the robot to protect against damages

### **Authorization and Authentication**

Username and passcode to login to Sphero to access code. Access to GitHub to share code with group members.

## ***Portability***

If portability is a requirement, specify attributes of the system that relate to the ease of porting the system to other host machines and/or operating systems. For example,

* Percentage of components with host-dependent code;
* Percentage of code that is host dependent;
* Use of a proven portable language;
* Use of a particular compiler or language subset;
* Use of a particular operating system;
* Use of specific Block code
* Use of specific java code from block code
* Use of a specific Application (Sphero)
* The need for environment-independence - the product must operate the same regardless of operating systems, networks, development or production environments.

# 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** |
| 11/22/19 | Holly Bernich, Project Manager  Nolan Beagell, Code and Tester | Confirmed AGILITY\_01-AGILITY\_06 |
| 11/22/19 | Holly Bernich, Project Manager  Anthony Berardis, Document and Chart Manager | Worked on section 2 of Requirements Doc |
| 11/22/19 | Holly Bernich, Project Manager  Nolan Beagell, Code and Tester | Worked on Section 3 of Requirements Doc |
| 11/22/19 | Anthony Berardis Document Manager | Worked on Section 4 of Requirements Doc |
| 11/22/19 | Holly Bernich Project Manager | Worked on Section 5 of Requirements Doc |

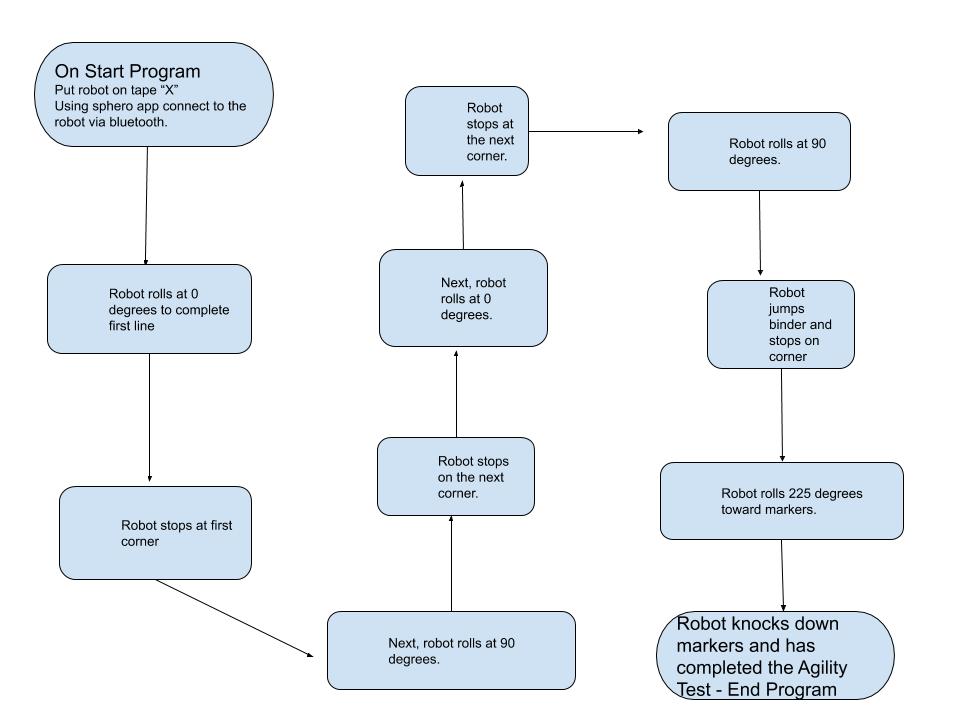
# System Design

## ***Algorithm***

Develop and describe here an algorithm that will be used to provide the required performance of your software

1. Put robot on tape “X”
2. Using sphero app connect to the robot via bluetooth.
3. On start program, robot rolls at 0 degrees.
4. Robot stops at first corner.
5. Next, robot rolls at 90 degrees.
6. Robot stops on the next corner.
7. Next, robot rolls at 0 degrees.
8. Robot stops at the next corner.
9. Robot rolls 90 degrees.
10. Robot jumps binder and stops on corner.
11. Robot rolls 225 degrees toward markers.
12. Robot knocks over all markers and stops.
13. Robot has completed the Agility Test.

## ***System Flow***



## ***Software***

Sphero block code, javascript, GitHub

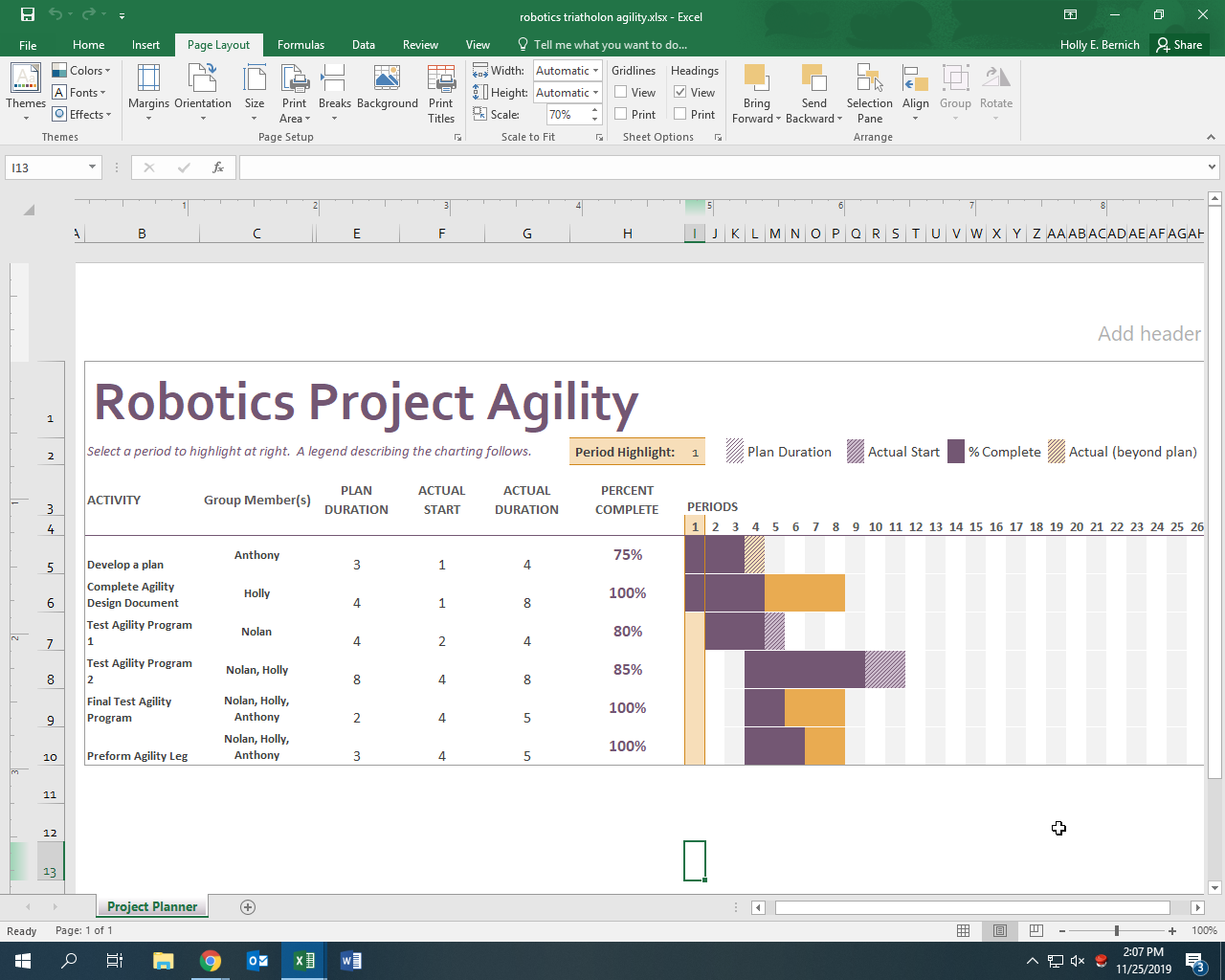
## ***Hardware***

Sphero robot, phone to run code, laptop computer to run code

## ***Test Plan***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Reason for Test Case** | **Test Date** | **Expected Output** | **Observed Output** | **Staff Name** | **Pass/Fail** |
| Testing start and obstacle avoidance of the Agility Test | 11/22/19 | Robot will attempt to perform first part of the Agility Test. | Robot was slightly off angle | Holly Bernich | Fail |
| Robot will attempt binder jump of Agility Test | 11/22/19 | Robot will attempt to land after the binder jump | Robot was off target | Nolan Beagell | Fail |
| Robot needs adjusted speed and angle for binder jump | 11/22/19 | Robot will perform whole first steps of Agility Test | Robot was slightly off due to speed. | Holly Bernich | Fail |
| Adjust speed for robot to go slower | 11/22/19 | Robot will use adjusted aim and speed to complete the Agility test. | Robot completed the first aspect of the Agility test | Nolan Beagell | Pass |
| Test last straight away of course | 11/22/19 | Robot has knocked down markers at the end of the course | Robot will have completed the Agility test with pass of this. | Holly Bernich  Nolan Beagell | Pass |
| Completion of course | 11/22/19 | Testing angles and speed for whole course | Robot has completed the course | Nolan Beagell | Pass |

## ***Task List/Gantt Chart***



## ***Staffing Plan***

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Responsibility** | **Reports To** |
| Holly Bernich | Project Manager  Code Tester | Code starter/implementer  Document writer  Calculate algorithms  Presenter | Professor Eckert |
| Anthony Berardis | Code Tester  Document Manager | Document leader  Code Tester  Adjust documents and gantt chart | Professor Eckert |
| Nolan Beagell | Code Tester  Document Manager | Figuring out aim  Writing document  Figuring out algorithms | Professor Eckert |