**Sprint 3 Agility - SDD**

1. **Executive Summary**

**1.1 Project Overview**

This is part 3 of the Robot Triathlon project called “Agility”. This includes making the robot go through a course that has many different parts such as zig zags, a ramp, and the need to knock over objects. This project is to test our team’s ability to manage and code our robot for the Agility test.

**1.2 Purpose and Scope of this Specification**

The Sphero robot needs to path and complete a figure 8 track with the ability to blink 5 times after the course is over and say “I am the winner”.

1. **Product / Service Description**

**2.1 Product Context**

The product of our team’s work to generate and code a four part system of tests for the robot. First Endurance, then Agility, Then Accuracy.

**2.2 User Characteristics**

This product will be used by our team of students on each of the upcoming robotic tests.

**2.3 Assumptions**

* The code will work on the first or second try
* The room will be available for use
* Everybody will contribute their part and get their jobs done correctly and on time
* Robot will have no issues

**2.4 Constraints**

* Room not being available
* Coding errors
* Floor being made with tiles
* Very hard to have friction which creates pathing errors for the robot
* Not too much time to work on this
* Robot taking long to charge

**2.5 Dependencies**

The sphero will require a charger and also proper coding to perform well on the Agility track.

**3.** **Requirements**

The robot will need to zigzag around bottles without touching them then it will need to launch off a ramp and then finally knock over several markers standing up at the end of the course.

**Priority Definitions**

* Priority 1: The robot MUST zig zag without touching the bottles
* Priority 2: The robot will need to launch itself up a ramp
* Priority 3: After jumping the ramp the robot must realign itself and get a “Strike” with the markers

**3.1 Functional Requirements**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Req#** | **Requirement** | **Comments** | **Priority** | **Date Rvwd** | **SME Reviewed / Approved** |
| **ACCUR\_1** | **Must be able to perform a zig zag maneuver around bottles** | **This wasn’t to difficult to get down just took a bit of educated guessing.** | **1** | **11/29/22** | **Approved** |
| **ACCUR\_2** | **Needed to be able to make it up the ramp** | **This was a bit trickier because we couldn’t get the right combination of speed for a while.** | **2** | **11/29/22** | **Approved** |
| **ACCUR\_3** | **Needs to be able to readjust itself and get the strike after hitting the jump** | **T)his was also a little tricky because of the readjust and because we had to find a good speed that would get the strike.** | **3** | **11/29/22** | **Approved** |
|  |  |  |  |  |  |

**3.2 Security**

**We used a password to get into the sphero account**

**3.2.1 Protection**

To protect the robot’s code integrity we would use a private connection that would need an admin password to get approval to operate the robot. The robot will also have an activity log that would show the user’s ip address and location of the device that was used to activate and use the robot. We would also have at least 2 approved admins on set to watch the robot be used. The robot will also have constant data logging to make sure if anything goes wrong that there will be data to find out where the error occurred.

**3.2.2 Authorization and Authentication**

We would use PubCookie to authenticate users and grant them access.

**4. Requirements Confirmation / Stakeholder Sign-Off**

|  |  |  |
| --- | --- | --- |
| Meeting Date | Attendees | Comments |
| November 29th, 2022 | Richard K, Roman D, Shabbar S | After as few errors we were able to get the robot to successfully and accurately complete its tasks. |

**5. System Design**

**5.1 Algorithm**

Agility Algorithm

1. Start robot

4. Set the speed to 50 at 0 degrees for 2 seconds

5. delay for 1.5 seconds

6. Set speed to 50 at 92 degrees for 2.15 seconds

7. delay for 1.5 seconds

8. Set speed to 48 at 358 degrees for 2.7 seconds

9. delay for 1.5 seconds

10. Set speed to 118 at 91 degrees for 2.7 seconds

11. delay for 1.5 seconds

12. Set speed to 81 at 225 degrees for 5 seconds

13. end program

**5.2 System Flow**

**file:///C:/Users/radat/Downloads/Flowchart.pdf**

**5.3 Software**

We used the Sphero box coding and api to code the robot.

**5.4 Hardware**

We used the Sphero given to us and the blue tape shaped in a box in the CS room.

**5.5 Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Reasons for Test Case** | **Test Date** | **Expected Output** | **Observed Output** | **Staff Name** | **Pass / Fail** |
| **First test to get the robot on the correct path** | **11/29** | **The robot would follow the zig zag** | **Robot did the zig zag around the bottles** | **Richard k, Roman D, Shabbar S** | **Passed the first time** |
| **Test to get the robot over the ramp** | **11/29** | **Robot would make it over the ramp** | **Robot makes it over the ramp and readjust itself** | **Richard k, Roman D, Shabbar S** | **Failed several times but was able to successfully get the right speed** |
| **Test to get the robot to knock over the markers** | **11/29** | **Robot would be able to get a “Strike” and knock over all markers** | **Robot would move to slow to knock them all over we just needed to adjust speed** | **Richard k, Roman D, Shabbar S** | **Failed once then was able to pass** |

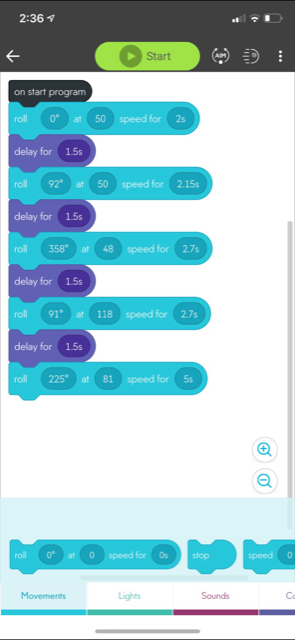
**5.6 Gantt Chart**



**5.7 Staffing Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Responsibility** | **Reports to** |
| **Richard** | **Writer/Designer** | **Filling info for design** | **Each other** |
| **Shabbar** | **Coder/Designer** | **Coding/Algo/FLchart** | **Each other** |
| **Roman** | **Manager/Designer** | **Helping with a bit of everything** | **Each other** |

**Block Code**

****

**Sensor Data**

