Sprint 3 - Agility Design Document

December 3rd, 2020

Table of Contents

1. Executive Summary

1.1 Project Overview

This project is the Agility leg of the triathlon. The goal of this leg is to get our Sphero robot to follow the zig-zag pattern around and not hit any of the bottles. It then needs to go up and over the ramp, once over it must make a turn and follow the straight away, knocking down the pins at the end. The intended audience is our classmates in CS-104 and Professor Eckert.

1.2 Purpose and Scope of this Specification

In scope

This document addresses requirements related to phase 3 of Project A:

- The program we create is designed to follow the path in Howard Hall
- The software is designed to run on smooth surfaces

Out of Scope

- The program will not work on leg 1 or leg 2 of the triathlon
- The robot will not run on sand

2. Product/Service Description

2.1 Product Context

The product (our program) relies on the Sphero Edu app to run its code. The program also relies on a connected Sphero robot in order for the program to run. The program relies on the app and the robot in order to exist. The app and robot can run other code, but our code can not run on other apps or different brands of robot. This is the third of three sprints we will have done.

2.2 User Characteristics

1. Computer Science Professor: Has experience using the robot in the past, has expertise in the field
2. Student: Limited experience in the field, has an interest in learning the field, has a background of using technology throughout their life
3. Child: No experience in the field, has an interest in technology, has a background using technology almost their whole life

2.3 Assumptions

- Test room in Howard Hall availability
- Sphero Edu app is available to run create our code on
- Robot is available and in possession of the group
- Assume that we have enough expertise to get the program running well

2.4 Constraints

- Restricted to remote work instead of collaborating in person
- Restricted to certain times to test in Howard Hall

2.5 Dependencies

- Algorithm needs to be complete before creating a flowchart
- Flowchart needs to be complete before programming the robot
- Program needs to have at least one line of code before the robot can move

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
AGILITY_01	Robot must be placed at the start of the course		1	12/2	Approved
AGILITY_02	Robot must start		1	12/2	Approved
AGILITY_03	Robot must move straight for 3 feet and 3 inches		1	12/2	Approved
AGILITY_04	Robot must turn 90 degrees to the right		1	12/2	Approved
AGILITY_05	Robot must move straight for 3 feet and 4 inches		1	12/2	Approved
AGILITY_06	Robot must turn 90 degrees to the left		1	12/2	Approved
AGILITY_07	Robot must move straight for 3 feet and 11 inches		1	12/2	Approved
AGILITY_08	Robot must turn 90 degrees to the right		1	12/2	Approved
AGILITY_09	Robot must move straight for 7 feet and 6 inches		1	12/2	Approved
AGILITY_10	While on that 7 foot 6 inch path, the robot must make a jump over a ramp			12/2	Approved
AGILITY_11	Robot must make a sharp right turn immediately after finishing the 7 foot 6 inch path		1	12/2	Approved
AGILITY_12	Robot must go straight for 9 feet		1	12/2	Approved
AGILITY_13	Robot must continue after that 9 feet to knock down all of the pins		1	12/2	Approved
AGILITY_14	Change numbers in the program to follow the path more accurately	Numbers will change based on multiple factors including the floor, distance, imperfections in the course.	2	12/2	Approved
AGILITY_15	Inspect the floor of Howard Hall to check for anything that would disrupt the path	The Tape on the floor of Howard Hall causes bumps in the track, important to take it into consideration.	2	12/2	Approved

3.2 Security

3.2.1 Protection

- Locked Zoom meetings to discuss project
- Password protected computers
- Github editing locked to the members of our group
- Files sent through locked Zoom meetings

3.2.2 Authorization and Authentication

No Authorization or Authentication was used for this project

3.3 Portability

- We used Sphero Edu for the code which is using javascript. Javascript is a very portable language per internet research.
- The program is easily uploadable to Github

4. Requirements Confirmation/Stakeholder sign-off

Meeting Date	Attendees (name and role)	Comments
12/2/20	Jordan, Chelsea, Emily	confirmed all

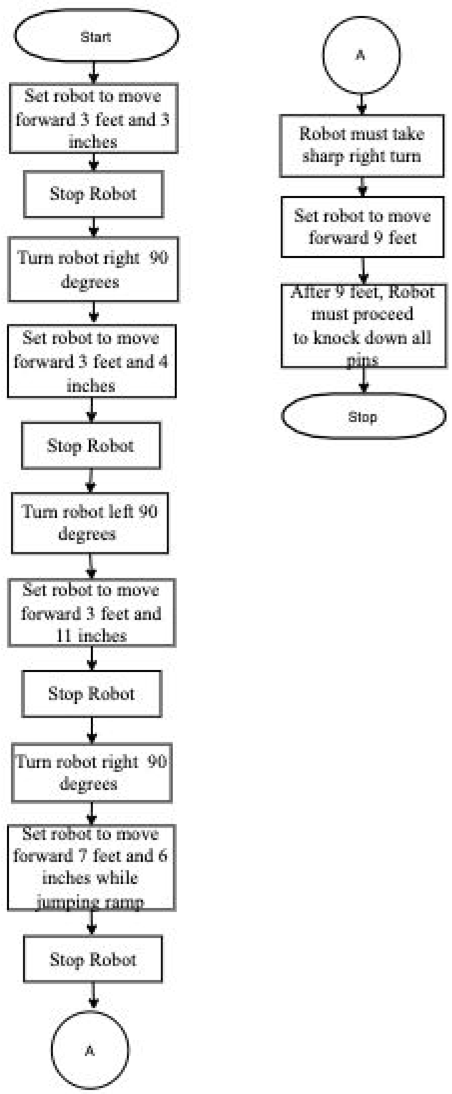
5. System Design

5.1 Algorithm

Algorithm for Agility Sprint

1.Place robot at start of course
2.Start robot
3.Set robot to move forward 3 feet and 3 inches
4.Stop robot
5.Turn robot 90 degrees right
6.Set robot to move forward 3 feet and 4 inches
7.Stop robot
8.Turn robot 90 degrees left
9. Set robot to move forward 3 feet and 11 inches
10.Stop robot
11.Turn robot 90 degrees to the right
12.Set robot to move forward 7 feet 6 inches while going over ramp
13.Stop robot
14.Robot must take sharp right turn
15. Set robot to move forward 9 feet
16. Robot must proceed to knock down all pins after 9 feet

5.2 System Flow



5.3 Software

- Zoom: We used Zoom to communicate during the project
- Sphero Edu: We used Sphero Edu to program the robot
- Google Docs: We used Google Docs for this document and to create the algorithm, it allowed for collaboration
- Microsoft Excel: Used for the Gantt Chart
- Draw.io: Used to create the flowchart
- Github: Used to upload important files to our repository
- Text Messages: We used text to communicate during the project as well

5.4 Hardware

- Sphero Robot
- Jordan's Macbook Pro

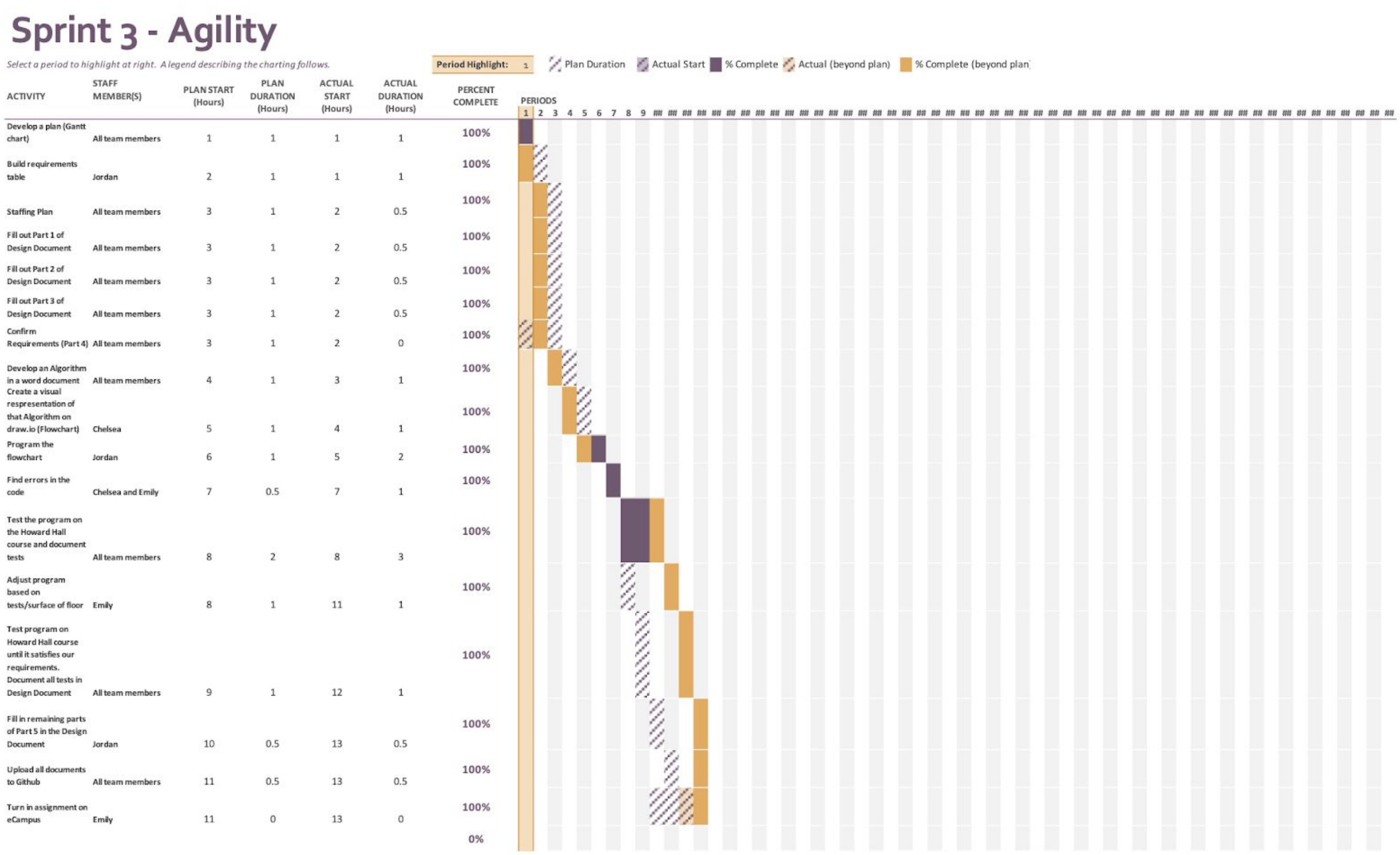
- Chelsea's Macbook Air
- Emily's Macbook Air
- Emily's iPhone 11

5.5 Test Plan

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
To start robot	12/2/2020	Robot will start	Robot started	Emily	Pass
Set robot to move the first length	12/2/2020	Robot will stop at the end of the first length	Robot went too short	Emily	Fail
Set robot to move the first length	12/2/2020	Robot will stop at the end of the first length	Robot hit the bottle	Emily	Fail
Set robot to move the first length	12/2/2020	Robot will stop at the end of the first length	Robot went correct length without hitting the bottle	Emily	Pass
Set the robot to turn 90 degrees to the right	12/2/2020	The robot will turn in the right direction	The robot turned in the right direction	Emily	Pass
Set robot to move the second length	12/2/2020	Robot will stop at the end of the second length	Robot hit bottle	Emily	Fail
Set robot to move the second length	12/2/2020	Robot will stop at the end of the second length	Robot went correct length without hitting bottle (adding delay helped)	Emily	Pass
Set the robot to turn 90 degrees to the left	12/2/2020	The robot will turn in the right direction	Robot turned in the correct direction	Emily	Pass
Set robot to move the third length	12/2/2020	Robot will stop at the end of the third length	Robot went past the end of the third length	Emily	Fail
Set robot to move the third length	12/2/2020	Robot will stop at the end of the third length	Robot went past the end of the third length	Emily	Pass
Set robot to turn 90 degrees to the right	12/2/2020	The robot will turn in the right direction	Robot turned in the correct direction	Emily	Pass
Make the robot go up the ramp	12/2/2020	Robot will go up the ramp	Robot did not even touch the ramp	Emily	Fail
Make the robot go up the ramp	12/2/2020	Robot will go up the ramp	Robot robot did not go all the way up	Emily	Fail
Make the robot go up the ramp	12/2/2020	Robot will go up the ramp	Robot landed off center	Emily	Fail
Make the robot go up the ramp	12/2/2020	Robot will go up the ramp	Robot landed correctly	Emily	Pass
Make the sharp right turn towards the last length	12/2/2020	Robot will make the turn	Robot's turn was wide	Emily	Fail

Make the sharp right turn towards the last length	12/2/2020	Robot will make the turn	Robot made the correct turn	Emily	Pass
Robot will complete the final length	12/2/2020	Robot will go the final length and stop before the pins	Robot came up short	Emily	Fail
Robot will complete the final length	12/2/2020	Robot will go the final length and stop before the pins	Robot went the correct distance	Emily	Pass
Robot will knock over the pins	12/2/2020	Robot knocks over all the pins	Robot did not knock all of them down	Emily	Fail
Robot will knock over the pins	12/2/2020	Robot knocks over all the pins	Robot knocked all pins down	Emily	Pass

5.6 Task List/Gantt Chart



Sprint 3 - Agility

Select a period to highlight at right. A legend describing the charting follows.

Period Highlight: 1

Plan Duration · Actual Start · % Complete · Actual (beyond plan) · % Complete (beyond plan)

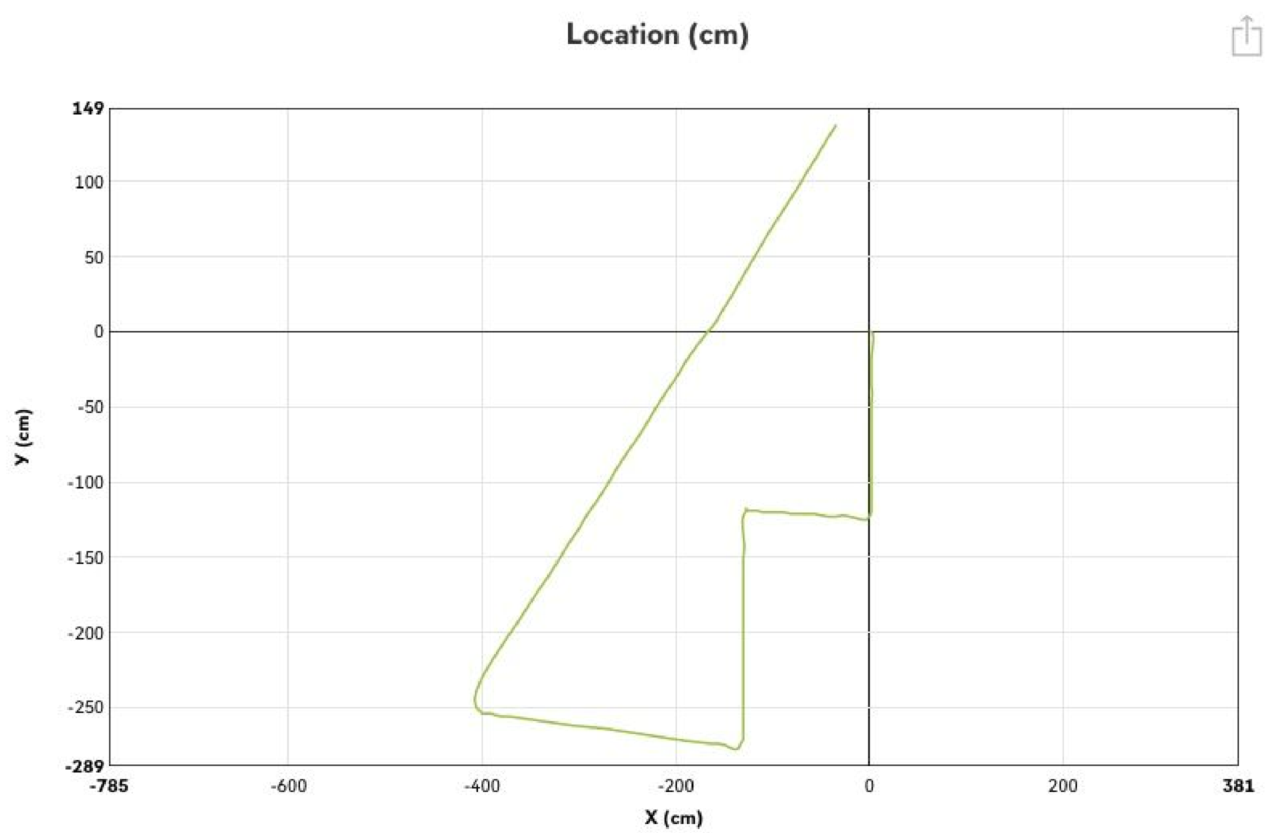
ACTIVITY	STAFF MEMBER(S)	PLAN START (Hours)	PLAN DURATION (Hours)	ACTUAL START (Hours)	ACTUAL DURATION (Hours)	PERCENT COMPLETE
Develop a plan (Gantt chart)	All team members	1	1	1	1	100%
Build requirements table	Jordan	2	1	1	1	100%
Staffing Plan	All team members	3	1	2	0.5	100%
Fill out Part 1 of Design Document	All team members	3	1	2	0.5	100%
Fill out Part 2 of Design Document	All team members	3	1	2	0.5	100%
Fill out Part 3 of Design Document	All team members	3	1	2	0.5	100%
Confirm Requirements (Part 4)	All team members	3	1	2	0	100%
Develop an Algorithm in a word document	All team members	4	1	3	1	100%
Create a visual respresentation of that Algorithm on draw.io (Flowchart)	Chelsea	5	1	4	1	100%
Program the flowchart	Jordan	6	1	5	2	100%
Find errors in the code	Chelsea and Emily	7	0.5	7	1	100%
Test the program on the Howard Hall course and document tests	All team members	8	2	8	3	100%
Adjust program based on tests/surface of floor	Emily	8	1	11	1	100%
Test program on Howard Hall course until it satisfies our requirements. Document all tests in Design Document	All team members	9	1	12	1	100%
Fill in remaining parts of Part 5 in the Design Document	Jordan	10	0.5	13	0.5	100%
Upload all documents to Github	All team members	11	0.5	13	0.5	100%
Turn in assignment on eCampus	Emily	11	0	13	0	100%
						0%

5.7 Staffing Plan

Name	Role	Responsibility	Reports To

Jordan	Team Leader, Staff Manager, Gantt Chart Manager, Programmer, Software Developer	To lead and oversee overall projects, to work with team members to develop programs, algorithms, and software, contribute ideas to overall project and sprint 1 document. Create the Gantt chart.	Professor Eckert
Emily	Team member, Software Tester, Idea Contributor, Software Developer, Repository owner	To work with team members to develop software/algorithms, to test robots on site (HH208), contribute ideas to the overall project.	Jordan
Chelsea	Team Member, Develop algorithm, Idea Contributor, Flowchart Developer, Algorithm Developer	To work with team members to develop algorithms, develop software, and contribute ideas to the overall project and sprint 1 document. To create the flowchart.	Jordan

5.8 Sensor Data



5.9 Block Code