Sprint 3 – Agility Design Document

December 4th, 2023

Table of Contents

1. EXECUTIVE SUMMARY 2

1.1 PROJECT OVERVIEW 2
1.2 PURPOSE AND SCOPE OF THIS SPECIFICATION 3

2. PRODUCT/SERVICE DESCRIPTION 3

2.1 PRODUCT CONTEXT 3
2.2 USER CHARACTERISTICS 3
2.3 ASSUMPTIONS 3
2.4 CONSTRAINTS 3
2.5 DEPENDENCIES 3

3. REQUIREMENTS 3

3.1 FUNCTIONAL REQUIREMENTS 3
3.2 SECURITY 4
3.2.1 Protection 4
3.2.2 Authorization and Authentication 4
3.3 PORTABILITY 4

4. REQUIREMENTS CONFIRMATION/STAKEHOLDER SIGN-OFF 4

5. SYSTEM DESIGN 5

5.1 ALGORITHM 5
5.2 SYSTEM FLOW 5
5.3 SOFTWARE 5
5.4 HARDWARE 6
5.5 TEST PLAN 6
5.6 TASK LIST/GANTT CHART 8
5.7 STAFFING PLAN 9

1. Executive Summary

1.1 Project Overview

The intended audience of our project is Professor Eckert and our class.

This project serves to demonstrate our ability to function as a team when faced with the challenge of making a robot move around the room. The multitude of facets and moving pieces require us to organize our time and workload. We intend to show this project to the professor instructing our class.

1.2 Purpose and Scope of this Specification

The purpose of this specification is to place the robot in the starting position of a course and make the robot traverse the obstacles without contact, rolling up a ramp, and traveling to the end while knocking over as many “pins” (represented by markers) as possible. This is phase three, accuracy out of three sprints, endurance, and agility. The intended audience is my professor and CS 104-02 class.

Inside scope:

- For this phase we will have to adjust our program to be able to accurately run the course while staying on the path efficiently and accurately.

Outside scope:

- Outside of the scope of these specifications include a sprint phase where the robot will have to run through an obstacle course and avoid the objects and then knock over as many pins as possible.

2. Product/Service Description

One of the general factors influencing the product and its requirements is the need to coordinate around our group members’ schedules, while other key factors encompass the program’s development based on the algorithm, as well as the hardware and software utilized during the project’s past phases.

2.1 Product Context

Our product is entirely self-contained in the coding application it was made in. It is run inside of and fills out its sensor data diagram inside the Sphero EDU app.

2.2 User Characteristics

Our product can be used by anyone. It does not require any special knowledge aside from having to navigate the Sphero EDU application and where the robot needs to be placed.

2.3 Assumptions

The code being run on a computer with MacOS so that the sensor data diagram can be exported as an easily readable image instead of raw data.

The robot starts in the center of the square at the start point of the course.

The track being clear of debris.

2.4 Constraints

Constraints of our design options is resetting the robot’s position and angle so that it perfectly navigates the track.

2.5 Dependencies

- A clear room to test our robot.
- Having the robot itself.

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
Agility_1	Start the robot in the middle of square to begin.	Robot needs to start in the center of the square.	3	11/27/23	11/27/23
Agility_2	Make robot go in a straight line and turn to the right without hitting first object.	Need robot to angle horizontally straight to go straight and turn to the right.	1	11/27/23	11/2723
Agility_3	Make robot go straight past first object, reach second corner, and turn to the left without hitting second object.	After turning right, robot needs to go straight and turn to the left.	1	11/27/23	11/27/23
Agility_4	Go straight and turn to the right without hitting third object.	After turning left, robot needs to go straight and turn to the right.	1	11/27/23	11/27/23
Agility_5	Go straight up and over the ramp, move forward a bit to reach the corner.	After turning right, robot needs to go straight up and over ramp to land on corner after the ramp.	1	11/27/23	11/27/23
Agility_6	Turn robot to the right and speed towards pins.	Once reaching the corner, robot needs to turn to the right.	2	11/27/23	11/27/23
Agility_7	Knock down as many pins as possible with robot.	When the robot turns to the right, it needs to go straight at fast speed to knock down as many pins as possible.	2	11/27/23	11/27/23

3.2 Security

3.2.1 Protection

- Password protected laptop.
- Making backups of the code.

3.2.2 Authorization and Authentication

There will be none.

3.3 Portability

- 100% of the project's components and code are host dependent.
- The product is environment independent.

4. Requirements Confirmation/Stakeholder sign-off

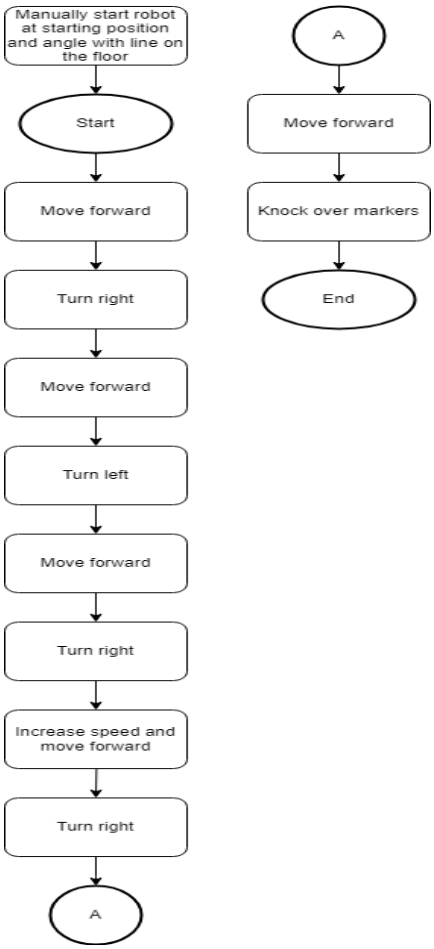
11/27/2023	Luca Bertinelli, Delvis Rodriguez, Xochitl Martinez	confirmed all except AGILITY__
11/27/2023	Luca Bertinelli, Delvis Rodriguez, Xochitl Martinez	confirmed…

5. System Design

5.1 Algorithm

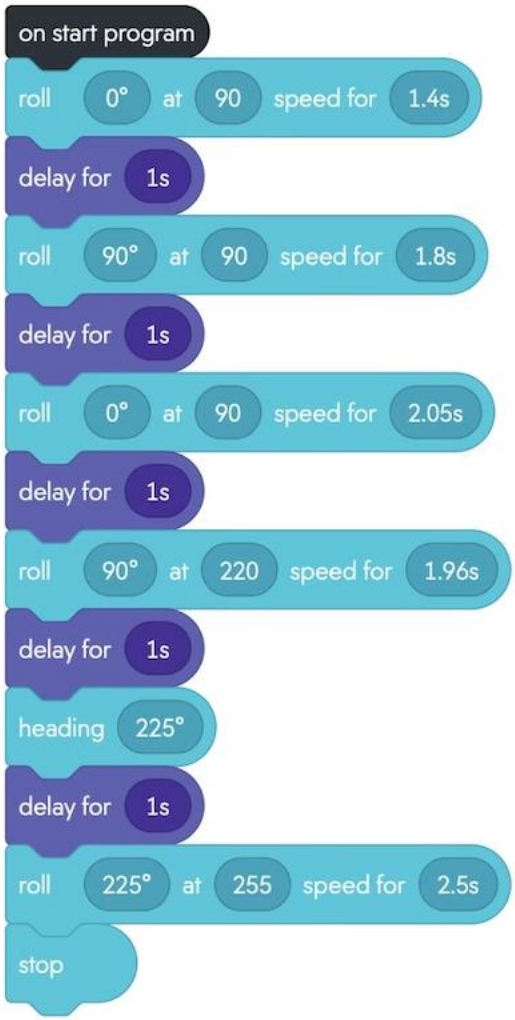
1. Start robot at starting point.
2. Aim robot with the line on the floor.
3. Move forward, stop, and turn to the right.
4. Move forward, stop, and turn to the left.
5. Move forward, stop, and turn to the right.
6. Increase speed and move forward past ramp.
7. Land and turn to the right.
8. Move forward.
9. Knock over markers.

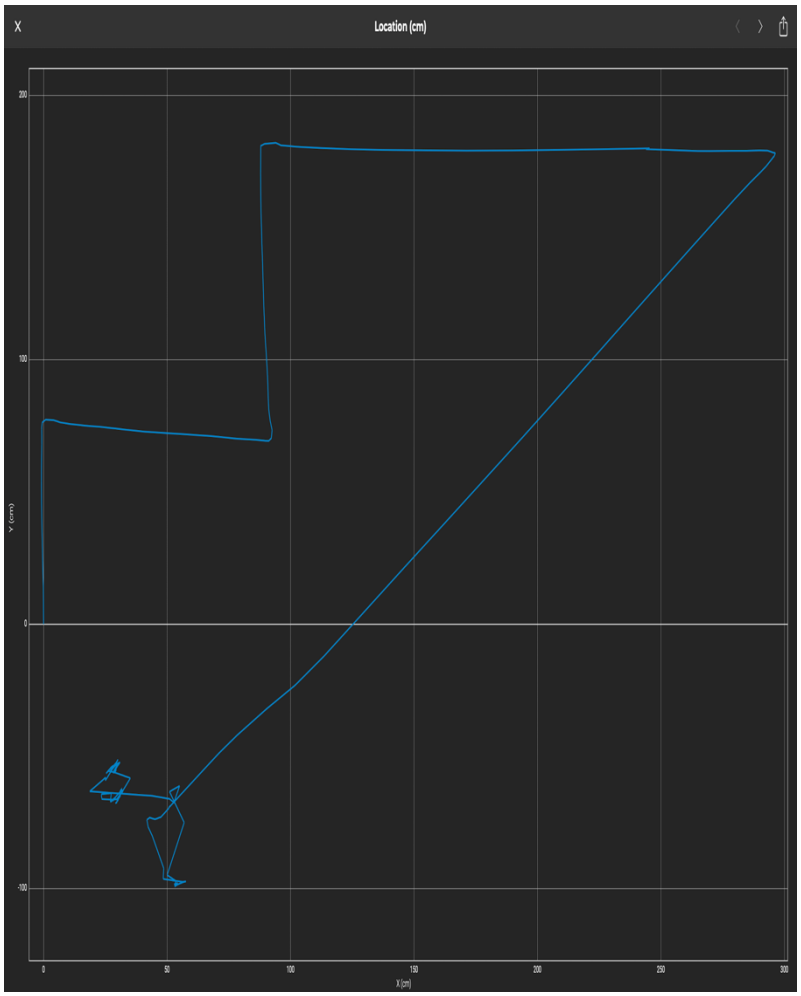
5.2 System Flow



5.3 Software

We use the Sphero EDU platform and block code to make our robot move throughout the course.





5.4 Hardware

The hardware platforms used for this project were the Sphero SPRK 2.0 robot and our laptops.

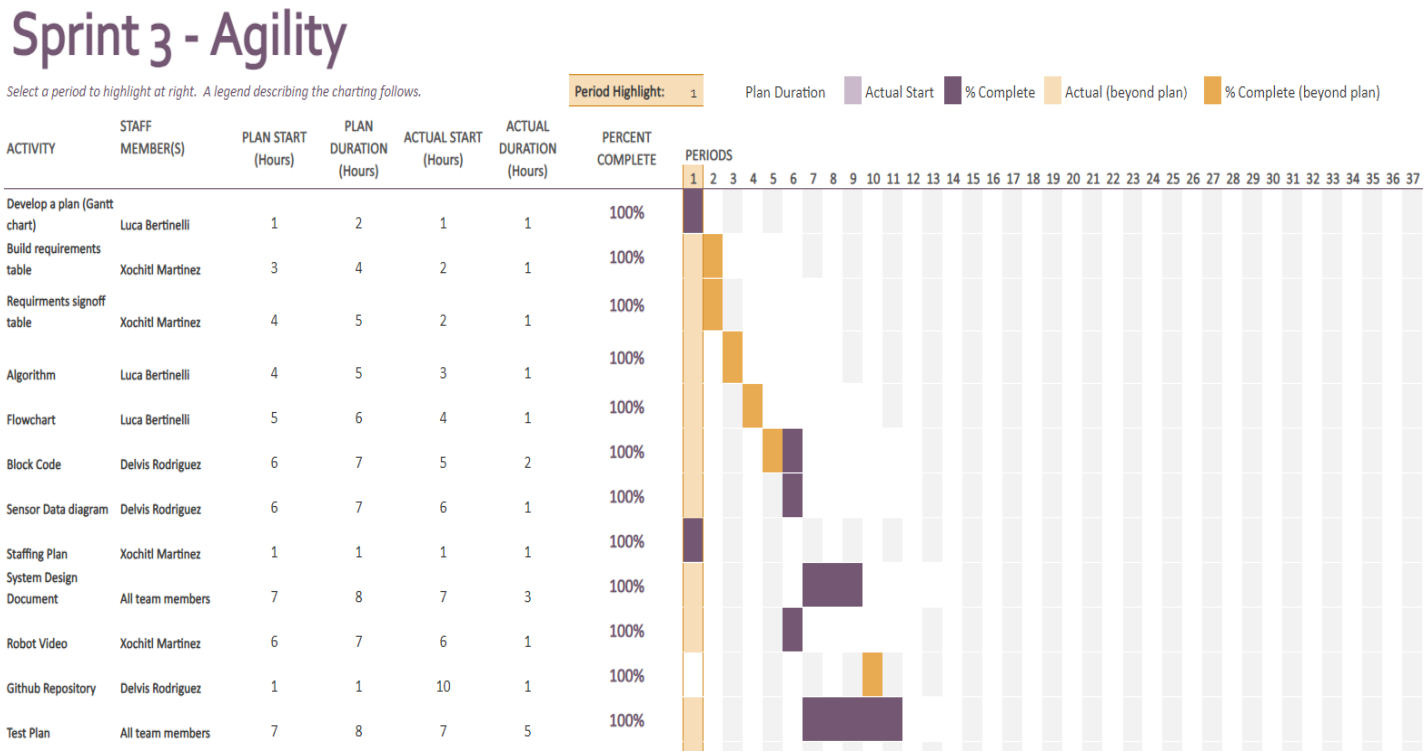
5.5 Test Plan

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Make robot go in straight line to reach first corner.	11/27/23	For robot to go in straight line.	Went in straight line but didn't reach corner.	Delvis & Xochitl	FAIL
Make robot go in straight line to reach first corner.	11/27/23	For robot to go in straight line.	Went in straight line and reached corner.	Delvis & Xochitl	PASS
Make robot go in straight line to reach first corner and turn 90 degrees to reach second corner.	11/27/23	For robot to go in straight line, turn 90 degrees and go straight to touch second corner.	Went in straight line, reached first corner, turned 90 degrees and hit second glass bottle.	Delvis & Xochitl	FAIL

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Make robot go in straight line to reach first corner and turn 90 degrees to reach second corner.	11/27/23	For robot to go in straight line, turn 90 degrees and go straight to touch second corner.	Went in straight line, reached first corner, turned 90 degrees and reached second corner.	Delvis & Xochitl	PASS
Make robot go in straight line to reach first corner and turn 90 degrees to reach second corner, then 0 degrees to reach third corner.	11/27/23	For robot to go in straight line, turn 90 degrees, go in straight line to touch second corner and then 0 degrees to go straight for third corner.	Went in straight line, reach first corner, turned 90 degrees to reach second corner and then 0 degrees to go straight and reach third corner.	Delvis & Xochitl	PASS
Make robot go up the ramp.	11/27/23	For robot to go up the ramp.	Robot went halfway up the ramp and rolled back.	Delvis & Xochitl	FAIL
Make robot go up the ramp.	11/27/23	For robot to go up the ramp.	Robot went up and over the ramp but went straight to the wall.	Delvis & Xochitl	PASS
Make robot go up the ramp and stop at a corner shortly after landing.	11/27/23	For robot to go up the ramp and stop shortly after landing.	Robot went up and over the ramp and stopped at the corner after.	Delvis & Xochitl	PASS
Make robot go up the ramp, stop at the corner and turn right to go straight and knock down pins.	11/28/23	For robot to go up the ramp, travel to corner, turn to the right and go full speed to knock pins.	Robot went up and over the ramp, stopped shortly after on the corner but turned to the left.	All	FAIL
Make robot go up the ramp, stop at the corner and turn right to go straight and knock down pins.	11/28/23	For robot to go up the ramp, travel to corner, turn to the right and go full speed to knock pins.	Robot went up and over the ramp, stopped shortly after on the corner but went in a different direction.	All	FAIL
Make robot go up the ramp, stop at the corner and turn right to go straight and knock down pins.	11/28/23	For robot to go up the ramp, travel to corner, turn to the right and go full speed to knock down pins.	Robot went up and over the ramp, stopped shortly after on the corner, turned to the right but by too much.	All	FAIL
Make robot go up the ramp, stop at the corner and turn right to go straight and knock down pins.	11/28/23	For robot to go up the ramp, travel to corner, turn to the right and go full speed to knock down pins.	Robot went up and over the ramp, stopped shortly after on the corner, turned to the right but just barely missed the pins.	All	FAIL
Make robot hit the pins after landing on the corner after the ramp.	11/30/23	For robot to stick the landing on the corner after going up the ramp and go full speed to knock down pins.	Robot barely missed the pins by 1cm to the right.	All	PASS

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Make robot hit the pins after landing on the corner after the ramp.	11/30/23	For robot to stick the landing on the corner after going up the ramp and go full speed to knock down pins.	Robot knocked down 2 pins.	All	PASS
Do the whole course without hitting any obstacle, land on corners, go up ramp and knock down pins.	11/30/23	For robot to do the whole course without making any mistakes.	Robot was placed wrong, hit an object and did not stay along the course.	All	FAIL
Do the whole course without hitting any obstacle, land on corners, go up ramp and knock down pins.	11/30/23	For robot to do the whole course without making any mistakes.	Robot did the whole course without hitting any obstacles, went up the ramp, landed shortly after and knocked down five pins.	All	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)	Luca Bertinelli	1	2	1	1	100%
Build requirements table	Xochitl Martinez	3	4	2	1	100%
Requirments signoff table	Xochitl Martinez	4	5	2	1	100%
Algorithm	Luca Bertinelli	4	5	3	1	100%
Flowchart	Luca Bertinelli	5	6	4	1	100%
Block Code	Delvis Rodriguez	6	7	5	2	100%
Sensor Data diagram	Delvis Rodriguez	6	7	6	1	100%
Staffing Plan	Xochitl Martinez	1	1	1	1	100%
System Design Document	All team members	7	8	7	3	100%
Robot Video	Xochitl Martinez	6	7	6	1	100%
Github Repository	Delvis Rodriguez	1	1	10	1	100%
Test Plan	All team members	7	8	7	5	100%

5.7 Staffing Plan

Name	Role	Responsibility	Reports To
Luca Bertinelli	Manager	Agility Design Document Gantt Chart Algorithm Flowchart Test Plan	All
Delvis Rodriguez	Manager	Agility Design Document Block Code Sensor Data Diagram Test plan GitHub repository	All
Xochitl Martinez	Manager	Agility Design Document Functional Requirements Test Plan Video recording Staffing plan Requirements signoff table	All