Sprint 3 - Agility

System Design Document

December 1, 2024

Anna Pitera, Isuhi Acosta, & Joseph Ventimiglia

CS-104-01

Table of Contents

1. EXECUTIVE SUMMARY 3

1.1 Project Overview 3
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 4

3. REQUIREMENTS 4

3.1 Functional Requirements 4
3.2 Security 4
3.2.1 Protection 4
3.2.2 Authorization and Authentication 4
3.3 Portability 5

4. REQUIREMENTS CONFIRMATION/STAKEHOLDER SIGN-OFF 5

5. SYSTEM DESIGN 5

5.1 Algorithm 5
5.2 System Flow 5
5.3 Software 6
5.4 Hardware 7
5.5 Test Plan 7
5.6 Task List/Gantt Chart 8
5.7 Staffing Plan 8

1. Executive Summary

1.1 Project Overview

Utilizing a Sphero SPRK robot, the goal of this project is to develop a block code algorithm that will allow the robot to successfully and accurately navigate an obstacle course around the CS-104-01 classroom without any errors.

1.2 Purpose and Scope of this Specification

The purpose of this project is to test and evaluate the agility skills of the Sphero SPRK robot as well as the effectiveness of our block code algorithm. The project's intended audience is the CS-104-01 class and professor.

The following requirements listed are in the scope of this project:

- A GitHub repository must be created and maintained for this project
- A Gantt chart made in Excel must be created and consistently updated to reflect progress made on the project
- A flow chart made in draw.io must be created to effectively display the robot's algorithm in a step-by-step format
- A video of the robot successfully completing the obstacle course must be recorded and uploaded to the collaborative GitHub repository

The following items listed are outside the scope of this project:

- The robot does not need to be as fast as possible because accuracy (i.e. staying on the blue tape) and successfully navigating the obstacle course without hitting any obstacles is of higher importance

2. Product/Service Description

2.1 Product Context

The Sphero SPRK robot is dependent on a computer or laptop to be controlled by the Sphero EDU application's block code.

2.2 User Characteristics

The users of the Sphero SPRK robot and the audience for this project are students and professors of CS-104-01. This audience has a higher technical expertise, as the vast majority of students in the class are computer science or software engineering majors at Monmouth University.

2.3 Assumptions

For this project, it is assumed that the user will be utilizing the Windows operating system on their computer to run the block code. It is also assumed that the user will have all the necessary parts in order to operate the Sphero SPRK robot.

2.4 Constraints

The design of the algorithm for this project may be constrained by some aspects of the Sphero SPRK robot and its application. For example, the robot itself is slightly unstable and has difficulty with moving straight in one direction. It also has difficulty moving smoothly due to its spherical design. In terms of the Sphero EDU application, the block code format may constrain how the algorithm is developed.

2.5 Dependencies

This project's algorithm is dependent on a laptop running the Windows 11 operating system. It is also dependent on a charger and charging adapter that will power the robot.

3. Requirements

Top Priority Requirements:

- An algorithm must be designed to make the Sphero SPRK robot navigate the obstacle course marked by blue tape on the floor of the CS-104-01 classroom
- The robot must be able to fully complete the obstacle course all in one go, meaning it cannot bump into any obstacles or encounter any errors in the block code
- The robot must be able to roll up the ramp in the obstacle course
- The robot must be able to knock down as many of the ten markers standing up at the end of the obstacle course as possible
- The robot's sensor data as shown in the Sphero EDU application must accurately reflect the obstacle course as marked by the blue tape on the floor of the classroom

Low Priority Requirements:

- The robot does not need to complete the obstacle course as fast as possible, but speed and timing will be of slight importance

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
1	Navigate the obstacle course marked with blue tape on the floor of the classroom	Must complete without bumping into any obstacles	Top Priority	11/20/24	Approved
2	Sensor data must reflect the obstacle course	Must be as exact as possible	Top Priority	11/20/24	Approved
3	Roll up the ramp in the obstacle course	Must be quick and make a smooth landing	Top Priority	11/20/24	Approved
4	Knock down the markers at the end of the obstacle course	Must knock down as many markers as possible	Top Priority	11/20/24	Approved

3.2 Security

3.2.1 Protection

There are no major security restrictions or requirements for this project, except for making sure to store the robot in a safe location when it is not in use.

3.2.2 Authorization and Authentication

There are no major authorization or authentication requirements for this project, except for making sure the robot is connected to the correct computer when working on the Sphero EDU application during class time.

3.3 Portability

The block code can be transferred to other computers as a Java code file.

4. Requirements Confirmation/Stakeholder sign-off

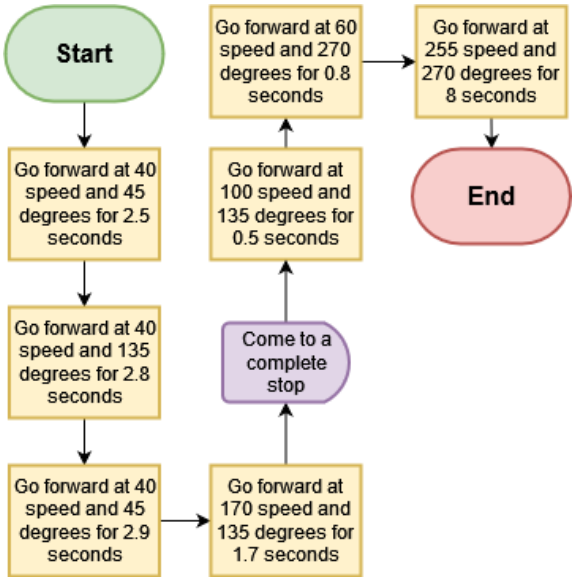
Meeting Date	Attendees (name and role)	Comments
11/20/24	Anna Pitera (Team Leader) Isuhi Acosta (Team Member) Joseph Ventimiglia (Team Member)	Confirmed all requirements

5. System Design

5.1 Algorithm

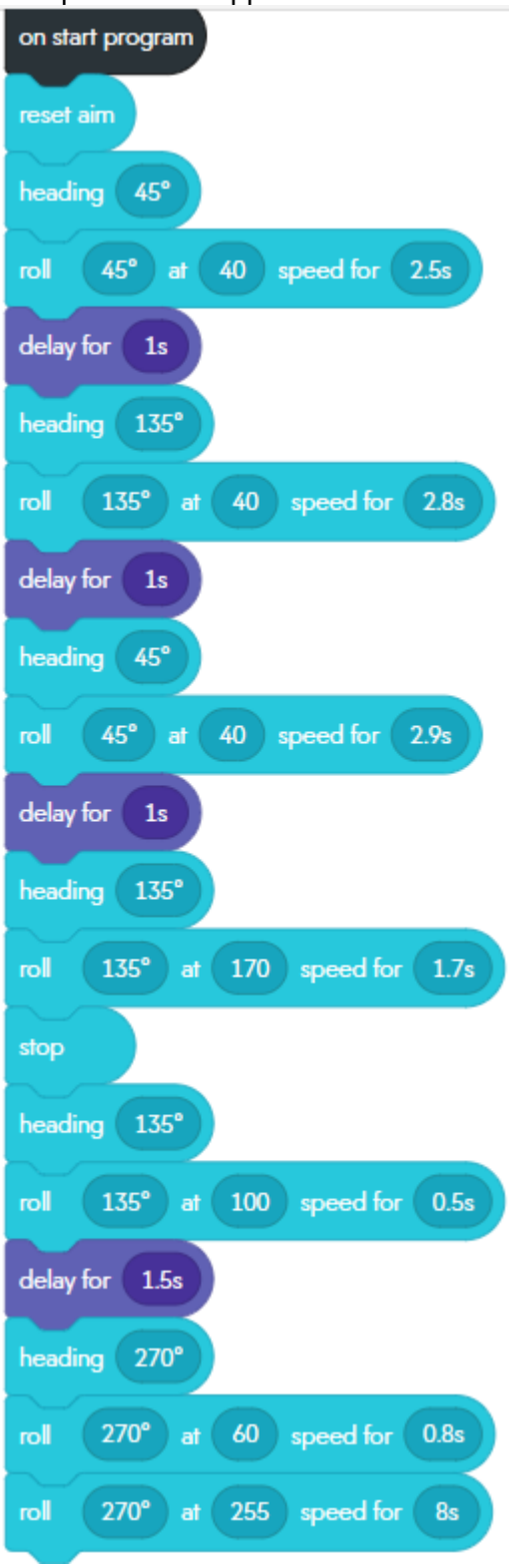
- When the program starts, the robot will move 45 degrees forward along the first line of blue tape, avoiding the obstacles
- Once it reaches the first corner, the robot will move 135 degrees forward along the second line of blue tape, avoiding the obstacles
- Once it reaches the second corner, the robot will move 45 degrees forward along the third line of blue tape, avoiding the obstacles
- Once it reaches the third corner, the robot will move 135 degrees forward along the third line of blue tape, avoiding the obstacles
- Once it reaches the ramp, the robot will increase its speed and go the ramp
- Once the robot lands on the ground after the ramp, the robot will move 135 degrees forward along the fourth line of blue tape
- Once the robot reaches the fourth corner, the robot will increase to maximum speed and move 270 degrees forward along the last line of blue tape
- Once the robot reaches the ten markers, the robot will be at maximum speed and knock as many down as possible, which will complete the program

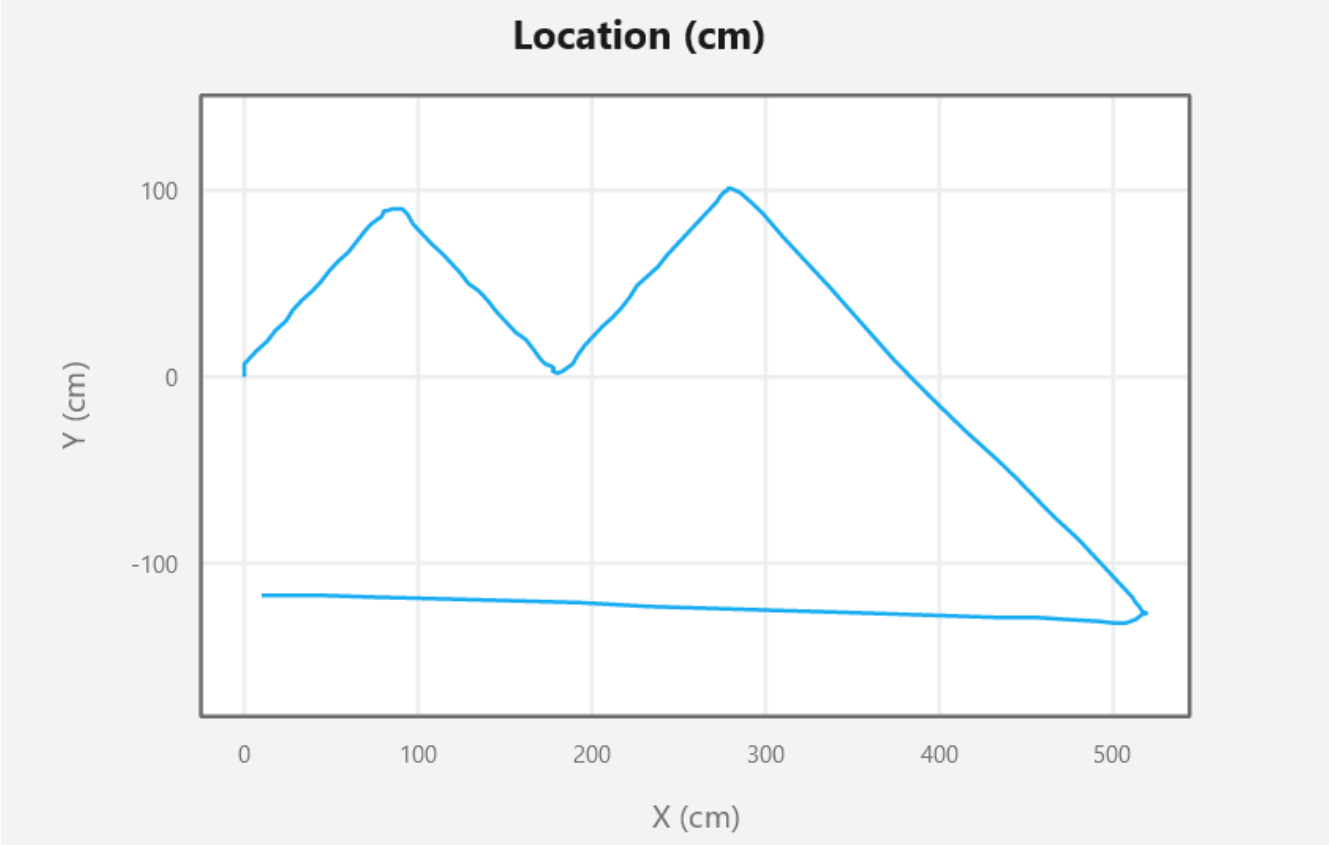
5.2 System Flow



5.3 Software

The Sphero EDU application was used to develop the block code for this project.





5.4 Hardware

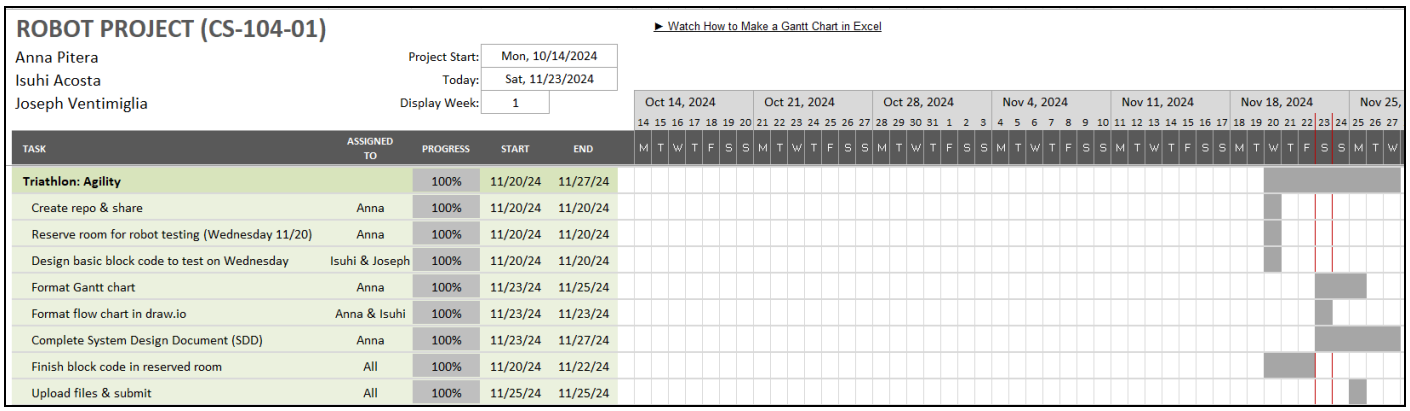
A laptop running the Windows 11 operating system was used to run the Sphero EDU application in order to control the Sphero SPRK robot. A charging wire, an adapter, and a Sphero charging pad were also used to power the robot.

5.5 Test Plan

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot lost aim before first corner	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot lost aim before second corner	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot lost aim before second corner and hit obstacle	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot hit obstacle at first corner	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot went too far on third corner	All team members	Fail

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot did not go all the way up the ramp	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot went too fast up the ramp and landed too far away	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot did not go all the way up the ramp	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot went all the way up the ramp, but lost aim and fell off	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot went all the way up the ramp and landed properly, but lost aim and did not hit the markers	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot only hit down one marker	All team members	Fail
Testing block code	11/20/24	Robot will successfully navigate the obstacle course	Robot hit down nine markers out of ten	All team members	Pass

5.6 Task List/Gantt Chart



ROBOT PROJECT (CS-104-01)

Anna Pitera
Isuhi Acosta
Joseph Ventimiglia

► Watch How to Make a Gantt Chart in Excel

Project Start: Mon, 10/14/2024
Today: Sat, 11/23/2024
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Triathlon: Agility		100%	11/20/24	11/27/24
Create repo & share	Anna	100%	11/20/24	11/20/24
Reserve room for robot testing (Wednesday 11/20)	Anna	100%	11/20/24	11/20/24
Design basic block code to test on Wednesday	Isuhi & Joseph	100%	11/20/24	11/20/24
Format Gantt chart	Anna	100%	11/23/24	11/25/24
Format flow chart in draw.io	Anna & Isuhi	100%	11/23/24	11/23/24
Complete System Design Document (SDD)	Anna	100%	11/23/24	11/27/24
Finish block code in reserved room	All	100%	11/20/24	11/22/24
Upload files & submit	All	100%	11/25/24	11/25/24

5.7 Staffing Plan

Name	Role	Responsibility	Reports To
Anna Pitera	Team Leader	- Maintaining Gantt chart - Room reservations - Block code - Robot storage/charging - Writing SDD	Prof Qu
Isuhi Acosta	Team Member	- Flow charts	Prof Qu

Name	Role	Responsibility	Reports To
		- Block code - Updating GitHub repository - Record videos	
Joseph Ventimiglia	Team Member	- Block code - Updating GitHub repository - Calibrating robot aim during testing	Prof Qu