Sprint 1 - Endurance Design Document

November 7, 2023

Table of Contents

1. Executive Summary

1.1 Project Overview

This project's goal is to make a Bluetooth Sphero robot run the perimeter of a rectangle in the classroom.

1.2 Purpose and Scope of this Specification

In scope

This document addresses requirements related to phase 2 of Project A:

- Modification of robot programming
- Measurement of tape on room floor
- Frequent communication via text, virtual, or in person

Out of Scope

The following items in phase 3 of Project A are out of scope:

- Managing requirements and time
- Fixing colors and distance

(Phase 3 will be considered in the development of the requirements for Phase 2, but the Phase 3 requirements will be documented separately.)

2. Product/Service Description

2.1 Product Context

This robot functions similarly to other programs. It's like a normal computer program, it requires a set of instructions to perform a task. When the robot is hooked up to someone's computer, they can use the blocks to code the robot to move in a certain direction. When the code is provided to speak, the voice will come out of the computer rather than the robot.

2.2 User Characteristics

The professor has worked with the robots before. He is proficient in how they work since he's used them before. For us, we only had one previous experience with the robots prior to receiving instructions.

2.3 Assumptions

The room might not be available to us, so we might not get accurate measurements for how far the robot should move. A Mac computer is also required for the sensor data diagram. There will also be lots of trial and error since we don't have lots of experience with the Sphero robots and this kind of technology. Also, we might have trouble connecting the Bluetooth to enable the Sphero robots.

2.4 Constraints

Describe any items that will constrain the design options, including

- Windows computers won't work
- One audit trail (test table) and don't know all of computer's functions
- Classes and other activities take up access time, need to manage time
- Don't know how to do professional computer evaluation, only evaluate at a normal level
- Possible limits of storage space
- Computer lag is predictable

2.5 Dependencies

- This product requires a computer to function
- The first distance must be measured correctly before the other distances are embedded in

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
ENDUR_01	Turn light green		1	10/25	Approved
ENDUR_02	Speak “ready set go”		1	10/25	Approved
ENDUR_03	Travel to corner of rectangle	moving in the west direction	1	10/25	Approved
ENDUR_04	Turn right		1	10/25	Approved
ENDUR_05	Travel to corner of rectangle	moving in the south direction	1	10/25	Approved
ENDUR_06	Turn right		1	10/25	Approved
ENDUR_07	Travel to corner of rectangle	moving in the east direction	1	10/25	Approved
ENDUR_08	Turn right		1	10/25	Approved
ENDUR_09	Travel to corner of rectangle	moving in the north direction	1	10/25	Approved
ENDUR_10	Turn right		1	10/25	Approved
ENDUR_11	Turn light red		1	10/25	Approved
ENDUR_XX	Speak “I’m done and I need water”		1	10/25	Approved

3.2 Security

3.2.1 Protection

- Security and privacy on computer protects data from being stolen
- Activity logging helps keep track of what we’ve done up to that point
- Robot isn’t always with computer, restricting inter module communication
- Data integrity checks when necessary

3.2.2 Authorization and Authentication

For authorization, Sphero has a privacy policy verifying that the users know how to use the product safety and effectively. For authentication, the computers and logging in to the coding software require usernames and passwords to enter, validating user’s identities.

3.3 Portability

- The robot must be connected to the computer the code is being made on
- A Mac is required for the sensor data diagram

4. Requirements Confirmation/Stakeholder sign-off

Meeting Date	Attendees (name and role)	Comments
10/25/23	Andrew	Requirements Approved

5. System Design

5.1 Algorithm

On start

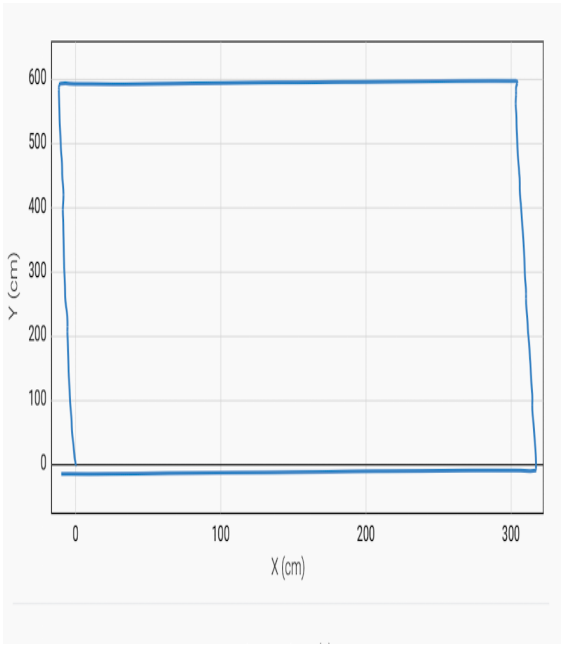
1. Turn the light green on the robot.
2. Have the robot say “Ready set go.”
3. Roll 0° at 80 speed for 11.75 s.
4. Delay for 1 second and rotate 90 degrees.
5. Roll 90° at 80 speed for 6.55 s.
6. Delay for 1 second and rotate 90 degrees.
7. Roll 180° at 80 speed for 11.8 s.
8. Delay for 1 second and rotate 90 degrees.
9. Roll 270° at 80 speed for 6.5 s.
10. Delay for 1 second and rotate 90 degrees.
11. Turn the light red on the robot.
12. Have the robot say “I’m done and I need water.”

5.2 System Flow

https://drive.google.com/file/d/1q5gWLBT5wd0yCjG47D9MI6mlEh_2Cu4B/view?usp=sharing

5.3 Software

Made with Sphero. Note: the initial 3 second delay was used to aid the creation of the video



5.4 Hardware

Laptops and the Sphero EDU program were used to code this robotics program.

5.5 Test Plan

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail (Fail* = some Pass, some Fail)
travel length of rectangle (22 ft.)	**10/25**	robot travel straight	robot went on a slight diagonal and didn't go the correct distance	Alex	Fail
" "	**10/25**	" "	robot traveled too far	Alex	Fail
" "	**10/26**	" "	robot went off the track and went against the wall	Alex	Fail
" "	**10/26**	" "	robot traveled too far	Alex	Fail
" "	**10/26**	" "	robot went off the track and traveled too short	Alex	Fail
travel width of rectangle (12 ft.)	**10/26**	robot travel straight	robot was off the track	Alex	Fail
travel length of rectangle	**10/26**	robot travel straight	robot followed the path	Alex	Pass
travel full rectangle path (22 ft., turn, 12 ft., turn, 22 ft., turn, 12 ft.)	**10/27**	robot stay on the tape on the floor	robot veered off the path to the left, cut a corner, but largely followed the width; it went farther than 22 ft. and I had to stop the program to retrieve the robot	Alex	Fail
" "	**10/27**	" "	robot veered off the path to the left; the robot went past the next 3 paths it was supposed to take	Alex	Fail
wanted to see if the robot could travel the whole course	**10/31**	turn in line with the track	turned too early, went too fast	Alex	Fail
" "	**10/31**	turn in line with the track (Alex added 0.5s to each movement as an adjustment)	went over the bounds	Alex	Fail
" "	**11/6**	stay on the course	first leg was too long	Kiumbura	Fail
wanted to see if the robot could complete the course accurately	**11/6**	stay on the course	veered off the first leg and went too far	Kiumbura	Fail
" "	**11/6**	stay on the course (tape)	first leg and second leg distance was good, but hit chair wheel on third leg	Andrew	Fail

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail (Fail* = some Pass, some Fail)
“ ”	**11/6**	stay on the tape	overcorrected for the second leg	Andrew	Fail
“ ”	**11/6**	stay on the tape	veered to the right off course for the first leg, went way too far on the second leg, traveled to the left and parallel of the third leg, was on track but died suddenly on the fourth leg	Andrew	Fail
“ ”	**11/7**	don’t go off the course	veered to the left off course and went a bit too far on first leg, was slightly too short on the second leg but corrected for the third, slightly too short on the third leg, finished at a good distance on the final leg	Alex	Fail*
“ ”	**11/7**	“ ”	veered slightly to the left and went slightly too far on the first and second leg, went slightly too far on the third leg, but perfectly completed the fourth leg	Alex	Fail*
“ ”	**11/7**	“ ”	first half was perfect, second half was too short and then too far	Alex	Fail*
complete the course accurately	**11/7**	“ ”	Although it was just off the course for parts of the third leg and the fourth, it was overall accurate.	Alex	Pass

5.6 Task List/Gantt Chart

Sprint 1 Endurance Gantt project plan Template.xlsx

5.7 Staffing Plan

Name	Role	Responsibility	Reports To
Alex	planning, developing, fine-tuning, submitting	SDD, gantt chart developer, creating code, test table, uploading to Github repository	Professor Eckert
Andrew	planning, listing requirements, outlining overview of code	SDD, requirements table, flowchart	Professor Eckert

Name	Role	Responsibility	Reports To
Kiumbura	planning, interpreting, documenting	SDD, building algorithm, creating sensor data diagram, creating code, robotics video	Professor Eckert