1.1 Project Overview

The purpose of this project is to program our robot to travel in an obstacle course shaped in a figure eight 5 times. Our robot must start and finish on the spaces provided. When finished the robot must speak "I am the winner" and flash different colored lights for 5 seconds. If the robot goes off the path then points will be deducted.

1.2 Purpose and Scope of this Specification

The purpose is to program the robot so it follows an obstacle course shaped in a figure eight.

In scope

- **The robot**
- **The obstacle course**
- **The figure eight shape**

Out of Scope

- **The actual code**
- **The software that allows the robot to run**
- **sphero.edu**

2. Product/Service Description

In this section, describe the general factors that affect the product and its requirements. This section should contain background information, not state specific requirements (provide the reasons why certain specific requirements are later specified).

2.1 Product Context

This general factor that affects this product is the obstacle course that has been set up. Also the code is another factor since that's what allows the robot to travel in a figure eight. The background information is the code that makes the robot work.

2.2 User Characteristics

Create general customer profiles for each type of user who will be using the product. Profiles should include:

- **Students**
- **Faculty**
- **Those involved with programming**
- **developers**

2.3 Assumptions

List any assumptions that affect the requirements, for example, equipment availability, user expertise, etc. For example, a specific operating system is assumed to be available; if the operating system is not available, the Requirements Specification would then have to change accordingly.

Sphero.edu needs to be available because that is where we program the robot. Also the obstacle course has to be available because the robot will follow the course. If the robot is not available then the project cannot be done, therefore the robot is also essential.

2.4 Constraints

Describe any items that will constrain the design options, including

- **A working computer**
- **The robot**
- **The code that allows the robot to move**
- **Access toSphero.edu**
- **A computer with proper battery life**

2.5 Dependencies

List dependencies that affect the requirements. Examples:

- **Code need to match requirements**
- **Intended audience needs to be accurate**
- **Sprint 1 needs to be completed before Sprint 2**
- **Sprint 2 needs to be completed before Sprint 3**

3. Requirements

- Requirements
  - **Robot must travel in a figure eight 5 times**
  - **Robot must stay on path provided**
  - **Robot must stay on path provided**
  - **Robot must speak upon completion "I am the winner" and flash different color lights for 5 seconds**

3.1 Functional Requirements

In the example below, the requirement numbering has a scheme - BR_LR_0## (BR for Business Requirement, LR for Labor Relations). For small projects simply BR-## would suffice. Keep in mind that if no prefix is used, the traceability matrix may be difficult to create (e.g., no differentiation between '02' as a business requirement vs. a test case)

The following table is an example format for requirements. Choose whatever format works best for your project.

For Example:

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
ENDUR_01	Robot must travel in a figure eight 5 times	none	yes	4/5	Everyone
ENDUR_02	Robot must stay on path provided	none	yes	4/5	Everyone
	Robot must speak upon completion "I am the winner"	must speak "I am the winner"	yes	4/5	Everyone
	Robot must flash different color lights for 5 seconds	no	yes	4/5	Everyone
	Robot cannot collide with any other objects	no	yes	4/5	Everyone

3.2 Security

3.2.1 Protection

Specify the factors that will protect the system from malicious or accidental access, modification, disclosure, destruction, or misuse.

- **Encryption**
- **Proper passwords**
- **multiple back up saves**
- **peer review from group members**
- **known locations where project is saved**

3.2.2 Authorization and Authentication

3.3 Portability

If portability is a requirement, specify attributes of the system that relate to the ease of porting the system to other host machines and/or operating systems. For example,

- **Email project to group members so everyone has it**
- **Screenshot code to share with group members**
- **Send project file to group members**
- **Make sure sphero.edu can run on multiple OS**

4. Requirements Confirmation/Stakeholder sign-off

Include documentation of the approval or confirmation of the requirements here. For example:

Meeting Date	Attendees (name and role)	Comments
4/5/20	Tom	confirmed
4/5/20	Meghan	confirmed
4/5/20	Tyrone	???

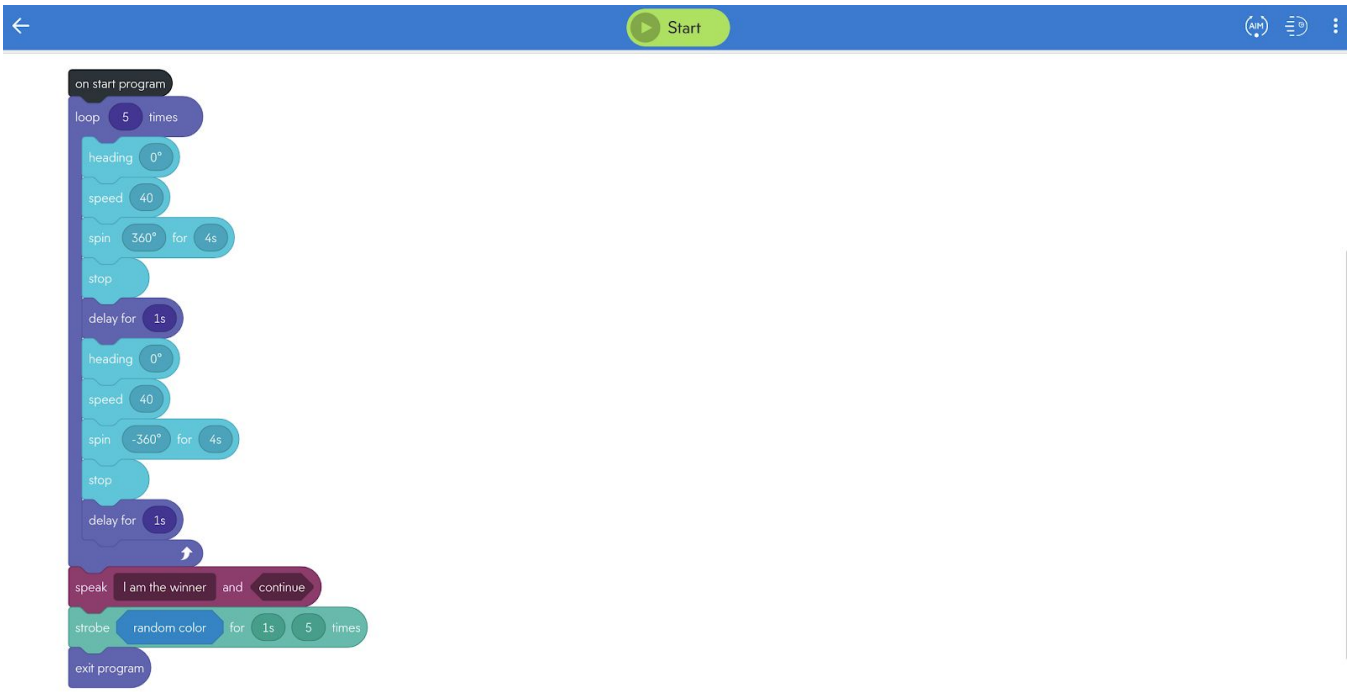
5. System Design

This section will provide all details concerning the technical design, staffing, coding, and testing the system

5.1 Algorithm

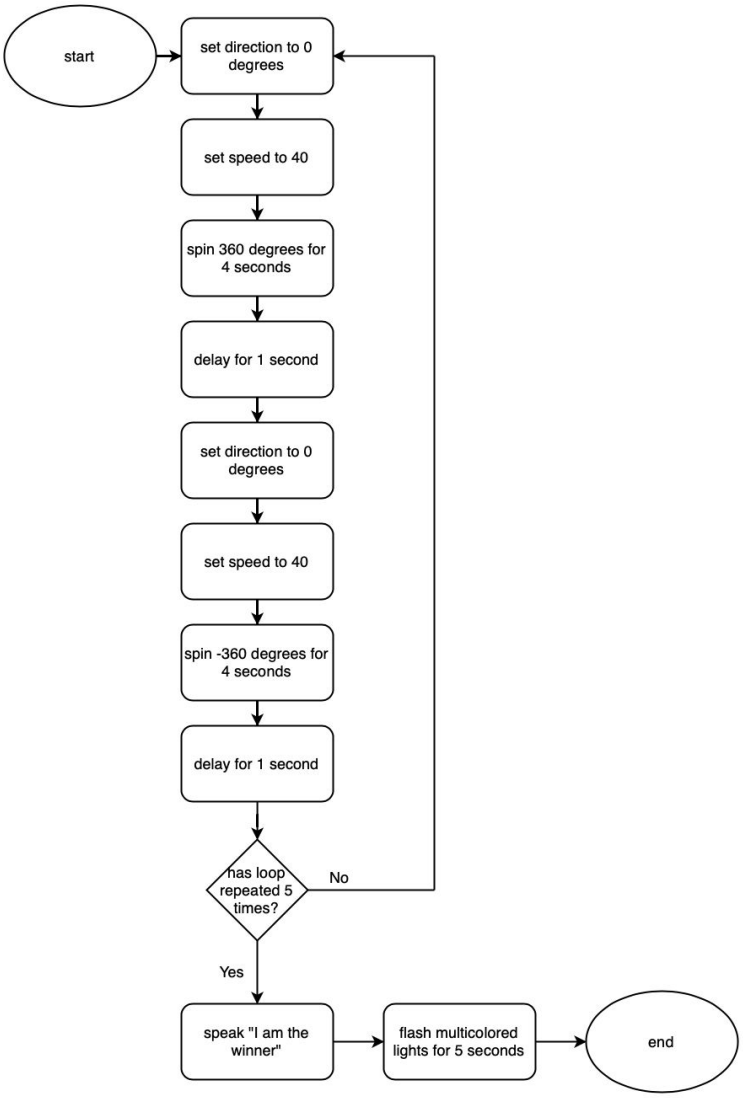
Develop and describe here the algorithm that will be used to provide the required performance of your software

- set direction to 0 degrees
- set speed to 40
- spin 360 degrees for 4 seconds
- delay for 1 second
- repeat at -360 degrees for 4 seconds
- loop program 5 times
- command robot to say “I am the winner”
- flash multicolored lights for 5 seconds
- exit program



5.2 System Flow

Develop a flowchart (and show here) that accurately depicts how your software application will act to fulfill the algorithm



5.3 Software

Sphero.edu is the software being used. It is a drag and drop programming software

5.4 Hardware

Describe hardware platforms that were used to develop, test and demonstrate this application

- **MacBook pro**

5.5 Test Plan

Include a test plan showing all unit tests performed for this application, Include test rational, test date, staff member, pass/fail status

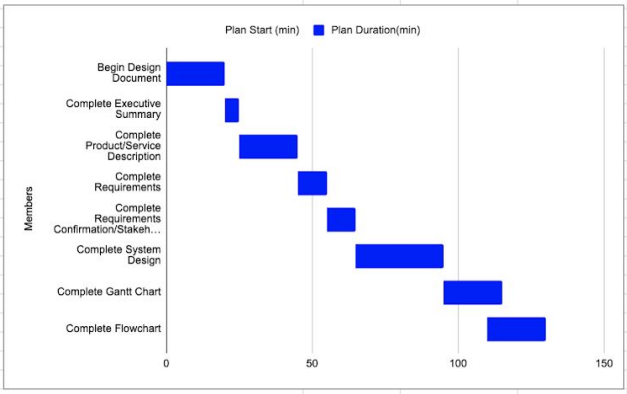
Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
speed	4/5/20	constant speed	too slow	Meghan	fail
shape	4/5/20	figure 8	figure 8	Meghan	pass
length of figure 8	4/5/20	sufficient distance	too small of a path	Meghan	fail
voice command	4/5/20	reads phrase	read correctly	Meghan	pass
loop 5 times	4/5/20	loop 5 times	loop 5 times	Meghan	pass

endpoint	4/5/20	end at start point	slightly off	Meghan	fail

5.6 Task List/Gantt Chart

Embed your gantt chart here

Members	Activity	Plan Start (min)	Plan Duration(min)	Actual Start(min)	Actual Duration(min)	Percent Complete
Tom, Meghan	Begin Design Document	0	20			100%
Tom and Meghan	Complete Executive Summary	20	5			100%
Tom and Meghan	Complete Product/Service Description	25	20			100%
Tom and Meghan	Complete Requirements	45	10			100%
Tom and Meghan	Complete Requirements Confirmation/Stakeholder sign-off	55	10			100%
Tom and Meghan	Complete System Design	65	30			100%
Tom	Complete Gantt Chart	95	20			100%
Meghan	Complete Flowchart	110	20			100%



5.7 Staffing Plan

Insert a chart/table that depicts the roles and responsibilities of each team member that worked on this project

Name	Role	Responsibility	Reports To
Meghan	code	writing and testing program code, system design template	Tom
Tom	system design	algorithm, system design template	Tom
Tyrone	??	??	everyone