Sprint 3 - Agility Design Document

November 29, 2022

Table of Contents

1. Executive Summary

1.1 Project Overview

In this project we will design a robot that is able to complete a predetermined obstacle course set up inside of HH208. The robot will need to speak at certain intervals, travel to the specified areas, light up the correct color when required, and perform other tasks to the best of its ability. There will be three tests the robot goes through, testing its skills in Endurance, Accuracy and Agility.

1.2 Purpose and Scope of this Specification

In scope

This document addresses the intended audience of the project:

- The intended audience of this project will be Prof. Eckert, who will judge its performance.
- Students in the classroom will also listen to our presentation and are also part of the intended audience.

Out of Scope

The following items address how this project relates to technology outside of its scope:

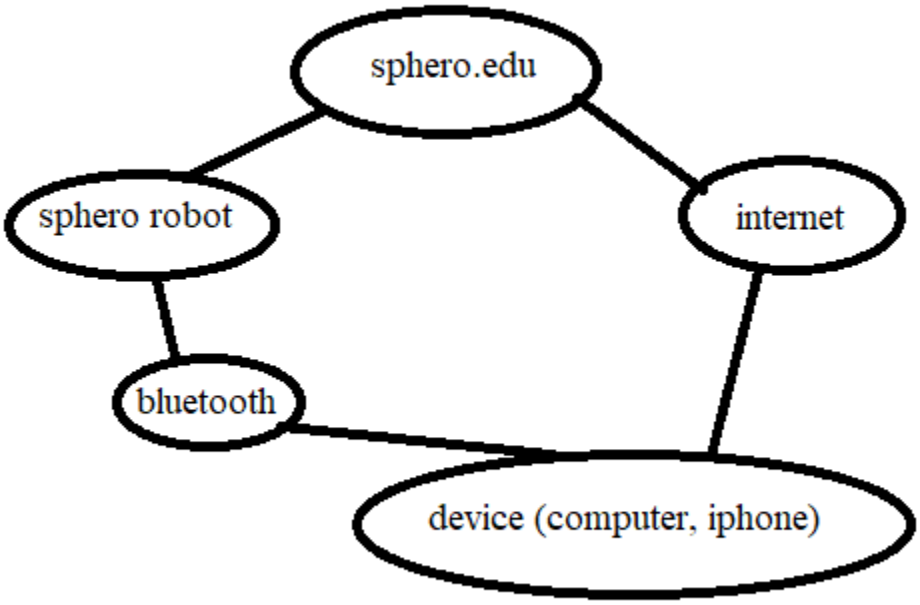
- Controlling a robot is something that is useful and applied in many factories in modern day.
- Products such as self-driving cars also use algorithms to control themselves.

2. Product/Service Description

2.1 Product Context

This product is similar to millions of other autonomous robots that are programmed to perform a specified task multiple times. Modern factories and assembly lines make heavy use of robots to complete certain parts of a process repeatedly. This project is independent and self-contained since it does not have any application outside of the classroom. Perhaps the robot itself could be reprogrammed to complete other tasks, so in that way, it is not self-contained. It does have limitations in that it doesn't have much interaction with a variety of related systems. A web application or database can be used in many ways by many systems, but this robot has a single programming application and a

single way of outputting data and actions.



2.2 User Characteristics

- Student
  - Very experienced with product
  - Above average technical expertise
  - Familiarity with computer science
- Faculty/staff
  - Much experience with product
  - Vast technical Expertise
  - Mastery of computer science
- Non-Comp Sci Student
  - No experience with product
  - Average technical expertise
  - Familiarity with technology
- Non-Comp Sci Faculty/Staff
  - Zero experience with product
  - Little technical expertise
  - Unfamiliarity with technology

2.3 Assumptions

We assume that the robot is performing in ideal conditions - flat ground, indoors with no wind, no outside interference at any point while it is running, and so on. We assume that the robot will do exactly

what it is told, exactly the same way every time we tell it to do something. There should be a sustained wifi connection to allow the robot to communicate to its controller.

2.4 Constraints

- The project will run on a limited time schedule.
- There is only one programming interface with no alternatives to code the robot.
- We can only work with one robot among three group members.
- The testing room has limited accessibility and tests can only be conducted there during certain times.
- We must use technology that is compatible with the robot. (windows computers cannot be used)
- Limitations within the IDE itself that is used to program the robot.
- External factors such as friction, sliding, and uneven ground may result in unintended results as for the movement of the robot.

2.5 Dependencies

- The robot must be placed properly in the square so that it is facing the correct direction for it to then follow the correct path. Otherwise, it may not follow the correct path and hit objects in the room which is a violation of a requirement.
- The robot will finish the course by knocking over the pins in the final portion of the course. The more pins knocked over, the more points awarded.

3. Requirements

Requirements:

1. The robot must begin in the designed square.
2. The robot will encounter three objects which all must be avoided, still following the designed path.
3. The robot must go over the ramp
4. The robot will go straight and knock over as many pins as possible.

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
AGIL_01	Robot must start in the square provided	robot must be placed within the square, or it will not follow the correct path	1	11/29/22	Approved
AGIL_02	Robot must stay within the path provided	robot must be aimed precisely when it is placed down to make sure it travels in the correct direction	1	11/29/22	Approved
AGIL_03	Robot must avoid all three obstacles in the zig zag pattern	make sure the robot does not collide with any of the objects	1	11/29/22	Approved

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
AGIL_04	Robot must successfully go over the ramp while still staying on course	make sure the robot follows the straight line over the ramp and keeps moving, and doesn't change direction on the fall down off the ramp.	1	11/29/22	Approved
AGIL_05	When the robot finishes the course, it will follow the final line, and knock over as many pins (markers) as possible. It is extra points awarded for any pin it hits.	The robot does not really have a final position to end in, just aim to follow the line and knock over the pins, but it is only extra points.	2	11/29/22	Approved
AGIL_06	Robot must not collide with anything in the room	If the robot does not follow the tape accurately enough, it may run into objects in the room	1	11/29/22	Approved
AGIL_XX					

3.2 Security

3.2.1 Protection

- The users who are verified by PubCookie are the only ones able to access the system.
- By clicking on expand windows log, the history of the system is stated
- Able to see if changes have been made to the system

3.2.2 Authorization and Authentication

The only people allowed to use this system are the people who are verified by PubCookie and are able to sign on.

4. Requirements Confirmation/Stakeholder sign-off

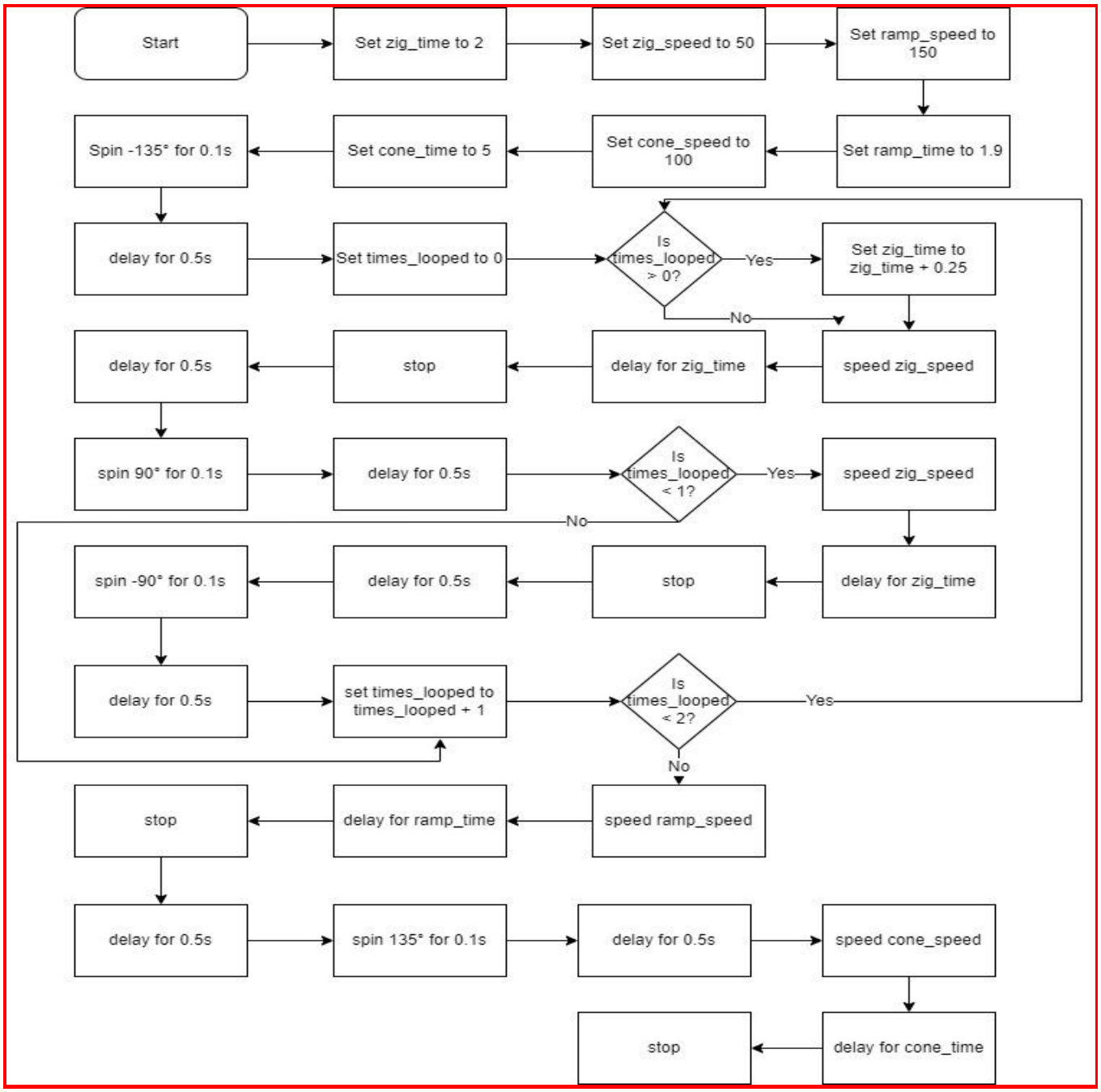
Meeting Date	Attendees (name and role)	Comments
11/29/22	Bryan Kahl	confirmed all
11/29/22	Jason French	confirmed all
11/29/22	Aidan Sacci	confirmed all

5. System Design

5.1 Algorithm

- Robot must start in the square provided.
- Robot must stay in the path provided, without hitting the obstacles designed on the course.
- Robot must go over the ramp
- Robot must go straight after turning off the ramp, and hit as many pins as possible that are on the path.

5.2 System Flow



5.3 Software

To develop and deploy this application, SpheroBlock Code was used.

5.4 Hardware

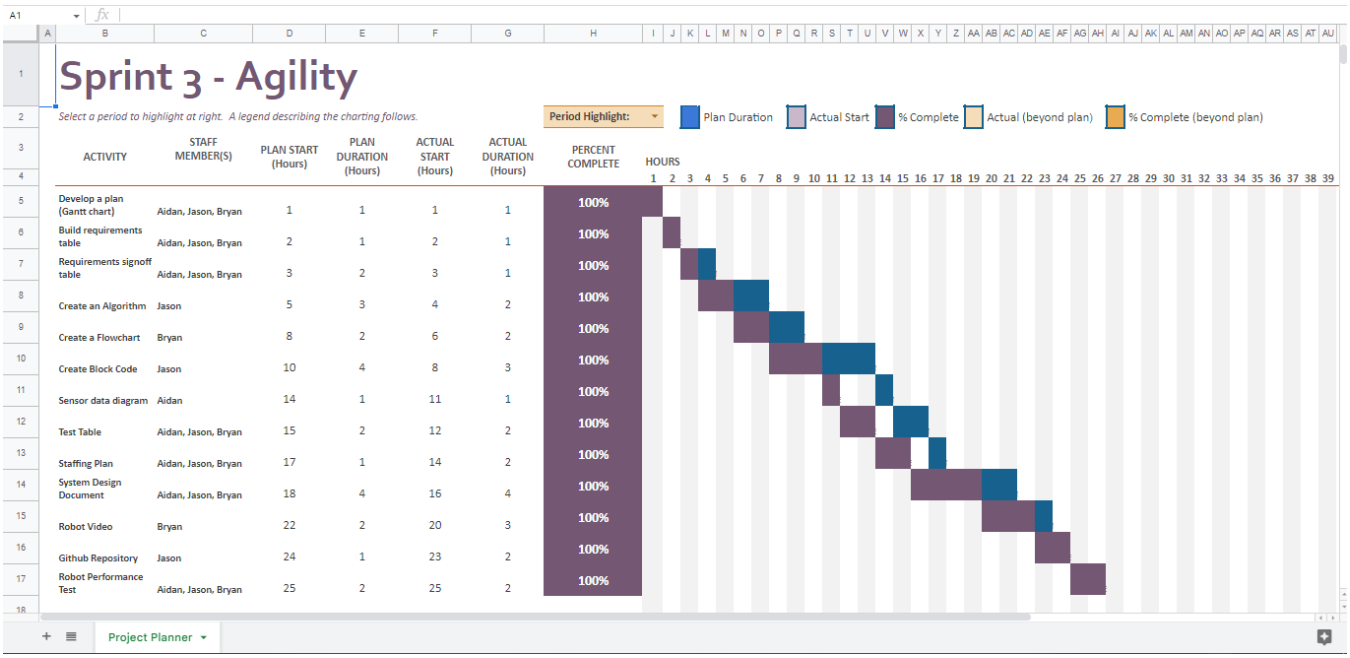
The hardware platforms used to develop, test and demonstrate this application were a laptop and the robot. An iphone was used to program and control the robot, as well as collect some sensor data.

5.5 Test Plan

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail
Test the blockcode to see if robot follows the course of avoiding the obstacles	11/29/22	Robot runs the course without hitting the objects and follows correct path	the robot wasn't following the turns, wasn't following the correct path.	Bryan	Fail
Test the blockcode to see if robot follows the course of avoiding the obstacles	11/29/22	Robot runs the course without hitting the objects and follows correct path	Robot hit the obstacles, throwing it off course	Aidan	Fail
Test the blockcode to see if robot follows the course of avoiding the obstacles	11/29/22	Robot runs the course without hitting the objects and follows correct path	Robot followed the desired path and avoided the obstacles	Aidan	Pass
Test ability to go over the ramp and follow the remaining path	11/29/22	Robot goes over the ramp, and follows the remaining path without going off course	Robot went over the ramp from the spot after it avoided the obstacles, and successfully went in a straight line, following the course	French	Pass
Test ability to follow the path after the ramp jump and hit the pins.	11/29/22	Robot follows the path and hits the pins	Robot did not end up following the path and hitting the pins after the ramp jump	French	Fail
Test ability to follow the path after the ramp jump and hit the pins.	11/29/22	Robot follows the path and hits the pins	Robot followed the path and hit all the pins after the ramp jump.	French	Pass
Test the all of the requirements of the robot for Agility Sprint	11/29/22	Robot successfully completed the obstacle course without hitting the obstacles, and went over the ramp successfully, and hit some pins all without going offcourse.	Robot was successful in completing the course and all requirements	Bryan	Pass

Reason for Test Case	Test Date	Expected Output	Observed Output	Staff Name	Pass/Fail

5.6 Task List/Gantt Chart



Sprint 3 - Agility

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

Period Highlight: | 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)	Aidan, Jason, Bryan	1	1	1	1	100%
Build requirements table	Aidan, Jason, Bryan	2	1	2	1	100%
Requirements signoff table	Aidan, Jason, Bryan	3	2	3	1	100%
Create an Algorithm	Jason	5	3	4	2	100%
Create a Flowchart	Bryan	8	2	6	2	100%
Create Block Code	Jason	10	4	8	3	100%
Sensor data diagram	Aidan	14	1	11	1	100%
Test Table	Aidan, Jason, Bryan	15	2	12	2	100%
Staffing Plan	Aidan, Jason, Bryan	17	1	14	2	100%
System Design Document	Aidan, Jason, Bryan	18	4	16	4	100%
Robot Video	Bryan	22	2	20	3	100%
Github Repository	Jason	24	1	23	2	100%
Robot Performance Test	Aidan, Jason, Bryan	25	2	25	2	100%

HOURS 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

Project Planner

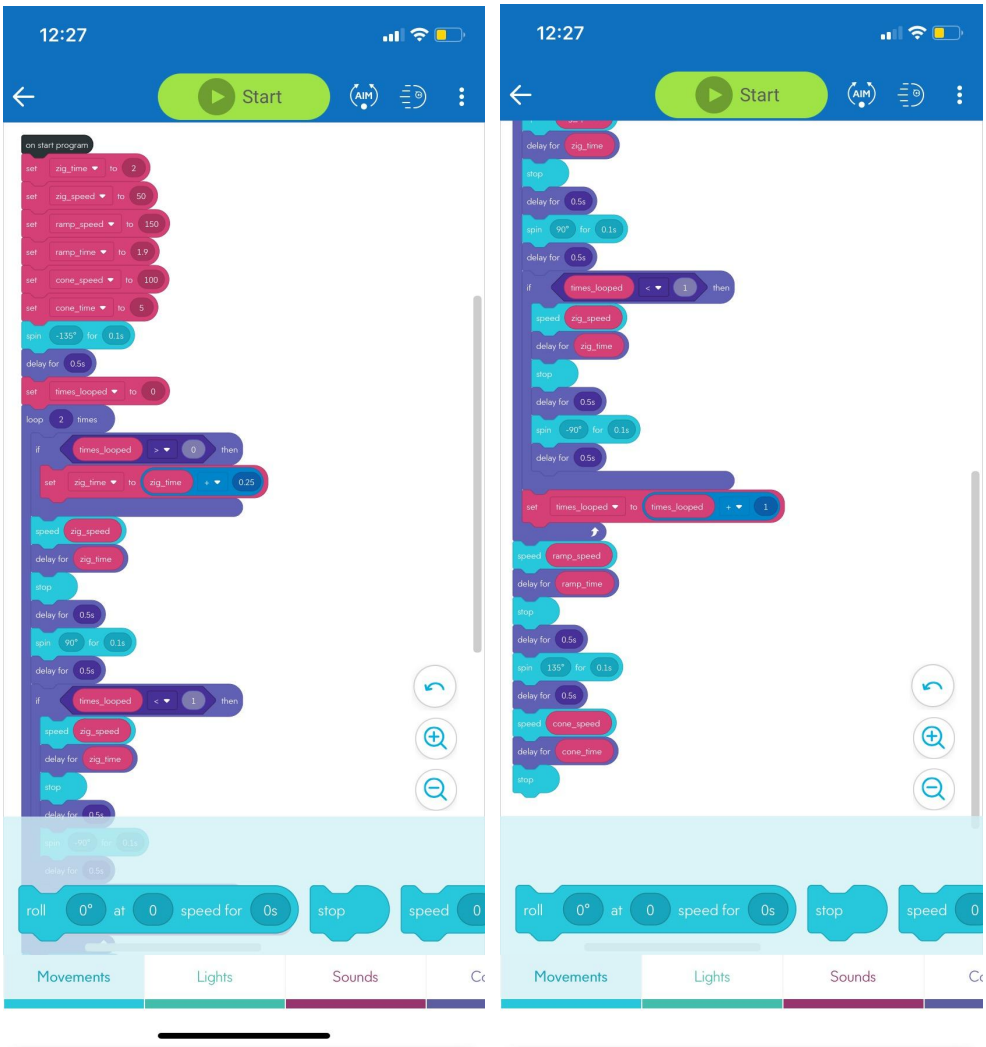
5.7 Staffing Plan

Name	Role	Responsibility	Reports To
French	Programmer	Code the logic to make the robot go around the track, avoid the obstacles and generally follow the requirements specified.	Group

Name	Role	Responsibility	Reports To
Bryan	Flowchart, robot video	Shows how the robot program works in a flowchart, and explains what the robot will do in a video	Group
Aidan	Sensor Data Diagram, System design documenter	Collects data, and fills out all the requirements for the system design document. States all the parts involved with the project	Group

Block Code and Sensor Data Diagram

Block Code:


12:27
Start
on start program
set zig_time to 2
set zig_speed to 50
set ramp_speed to 150
set ramp_time to 1.9
set cone_speed to 100
set cone_time to 5
spin -135° for 0.1s
delay for 0.5s
set times_looped to 0
loop 2 times
if times_looped > 0 then
set zig_time to zig_time + 0.25
speed zig_speed
delay for zig_time
stop
delay for 0.5s
spin 90° for 0.1s
delay for 0.5s
if times_looped < 1 then
speed zig_speed
delay for zig_time
stop
roll 0° at 0 speed for 0s
stop
speed 0
Movements
Lights
Sounds
12:27
Start
delay for zig_time
stop
delay for 0.5s
spin 90° for 0.1s
delay for 0.5s
if times_looped < 1 then
speed zig_speed
delay for zig_time
stop
delay for 0.5s
spin -90° for 0.1s
delay for 0.5s
set times_looped to times_looped + 1
speed ramp_speed
delay for ramp_time
stop
delay for 0.5s
spin 135° for 0.1s
delay for 0.5s
speed cone_speed
delay for cone_time
stop
roll 0° at 0 speed for 0s
stop
speed 0
Movements
Lights
Sounds

Sensor Data Diagram: