

# **Sprint 1 - Endurance Design Document**

**November 12, 2020**

## **Table of Contents**

<b>1. Executive Summary</b>	<b>3</b>
1. Project Overview	3
2. Purpose and Scope of this Specification	3
<b>2. Product/Service Description</b>	<b>3</b>
1. Product Context	3
2. User Characteristics	3
3. Assumptions	3
4. Constraints	3
5. Dependencies	4
<b>3. Requirements</b>	<b>4</b>
2. Functional Requirements	4
3. Security	5
1. <i>Protection</i>	5
2. <i>Authorization and Authentication</i>	5
4. Portability	5
<b>4. Requirements Confirmation/Stakeholder sign-off</b>	<b>5</b>
<b>5. System Design</b>	<b>6</b>
1. Algorithm	6
2. System Flow	6
3. Software	6
4. Hardware	6
5. Test Plan	6
6. Task List/Gantt Chart	6
7. Staffing Plan	6

# 1. Executive Summary

## 1. *Project Overview*

This product is intended to test our knowledge on software development. It is specifically designed to be an early project where we can gain a better understanding at coding and the development process as a whole.

## 2. *Purpose and Scope of this Specification*

### **In scope**

Our project is meant to control a machine through software only.

### **Out of Scope**

Our project can not obey other commands, as we develop its code the robot responds only to management.

# 2. Product/Service Description

Our product is designed for the sole purpose of overcoming environmental obstacles. In this first instance, the robot must navigate the perimeter of HH208, while at the same time, say a few phrases. These requirements were outlined by our instructor.

## 1. *Product Context*

Compared to other products, ours is very easy to use. The specific block code was created in an app called Sphero Edu. In order for the user to use our particular code, they will need to download the app and look up the code we created. The user then simply selects the "Run" option at the top of the screen.

## 2. *User Characteristics*

- Student/faculty
- Experience: Beginner level
- Technical expertise:
  - Must know how to navigate the Sphero Edu application on either Windows/IOS

## 3. *Assumptions*

- User has some understanding/knowledge of navigating the Sphero Edu app
- User possesses the Sphero robot
- User has some basic knowledge on the type of block code consistent with the Sphero app
- User has a computer/phone

## 4. *Constraints*

- Must have IOS 10.0 or higher
- Must have space available for the robot to operate (5.5 ft x 3 ft).
- Must have a Sphero account to access the code.
- Must have available space (67.4 MB of existing storage).
- Smooth surface is needed for the robot path.

## 5. **Dependencies**

- Requires space for occasional updates to the applications
- Requires specific version of the Sphero robot (SPRK +)

## 3. **Requirements**

- Robot must travel in a rectangle with length:5.5ft and width:3ft. (space constrictions)
- Robot should start with a green light and say "Ready, set, go!"
- Robot should travel 5.5ft and take a right.
- Robot should travel 3ft and continue on the rectangle's length and back up it's width.
- Robot should not collide with any objects as it goes around the room.
- Robot must return to it's starting location.
- Robot must say "I'm done and I need water."

See the sample requirements in Functional Requirements, and **Error! Reference source not found.**, as well as these example priority definitions:

### **Priority Definitions**

The following definitions are intended as a guideline to prioritize requirements.

- Priority 1 – The requirement is a "must have" as outlined by policy/law
- Priority 2 – The requirement is needed for improved processing, and the fulfillment of the requirement will create immediate benefits
- Priority 3 – The requirement is a "nice to have" which may include new functionality

It may be helpful to phrase the requirement in terms of its priority, e.g., "The value of the employee status sent to DIS **must be** either A or I" or "It **would be nice** if the application warned the user that the expiration date was 3 business days away". Another approach would be to group requirements by priority category.

- A good requirement is:
- Correct
- Unambiguous (all statements have exactly one interpretation)
- Complete (where TBDs are absolutely necessary, document why the information is unknown, who is responsible for resolution, and the deadline)
- Consistent
- Ranked for importance and/or stability
- Verifiable (avoid soft descriptions like "works well", "is user friendly"; use concrete terms and specify measurable quantities)
- Modifiable (evolve the Requirements Specification only via a formal change process, preserving a complete audit trail of changes)
- Does not specify any particular design
- Traceable (cross-reference with source documents and spawned documents).

### 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					
ENDUR_02					
ENDUR_03					
ENDUR_XX					

### Security

- All users are protected by the Sphero edu app's data protection software
- Users can only access code with their private account
- Uses Bluetooth
  - Can be accessed and remembered by the owners ios or windows device

### Authorization and Authentication

User must link their device with the robot via bluetooth to be authorized access

### Portability

- Robot can be used across multiple platforms, as long as it supports bluetooth
- Sphero app must be loaded on such device
- Code can be pulled up quickly via the Sphero app

## 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
MM/DD/YY	My group member names	confirmed all except ENDUR_XX
MM/DD/YY	My group member names	confirmed.....

## 5. System Design

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

### 1. Algorithm

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

-

## 2. **System Flow**

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

## 3. **Software**

Describe software languages/platforms/api's used to develop and deploy this application

## 4. **Hardware**

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

## 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
Rectangle	11/10	Straight for 5.5ft	Wrong way	Vincent	Fail
Rectangle	11/10	Straight for 5.5ft	Right way	Vincent	Pass
Rectangle	11/10	Straight for 3ft	Right way	Vincent	Pass
Rectangle	11/10	Straight for 5.5ft	Hit wall	Vincent	Fail
Rectangle	11/10	Straight for 3ft	Hit wall	Vincent	Fail
Rectangle	11/10	Straight for 5.5ft	Hit wall	Vincent	Fail
Rectangle	11/10	Straight for last 3 ft	Did not finish	Vincent	Fail
Rectangle	11/10	Straight for last 3 ft	Went too far	Vincent	Fail
Rectangle	11/10	Makes rectangle with proper proportions	Went for 6 ft instead of 5.5 ft	Vincent	Fail
Rectangle	11/10	Makes rectangle with proper proportions	Went for 3.4 ft instead of 3	Vincent	Fail
Rectangle	11/10	Straight for 5.5 ft	Went for 5.5 ft	Vincent	Pass
Rectangle	11/10	Makes rectangle with proper proportions	Went for 3.2 ft instead of 3	Vincent	Fail
Rectangle	11/10	Go forward	Hit wall	Vincent	Fail
Rectangle	11/10	Makes rectangle with proper proportions	Went for 3 ft	Vincent	Pass
Rectangle	11/10	Speak at the end and turn red	Did not speak at the end or turn red	Vincent	Fail
Rectangle	11/10	Makes rectangle with proper proportions	Went for 3 ft and spoke	Vincent	Pass
Rectangle	11/10	Make a 5.5 ft by 3 ft rectangle with correct speaking commands and led changes.	Went for 5.5 ft by 3 ft rectangle with correct speaking commands and led changes.	Vincent	Pass

6. ***Task List/Gantt Chart***

Embed your gantt chart here

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