CS104 - Sprint 1 Endurance – System Design Document

Kathryn Schauer, Nicholas Smythe, Joseph Scerbo

Kathryn Schauer

**Algorithm:**

* Start the program
* Input any color
* Input speed to 63
* Loop set spin to 360 degrees for 7.5 seconds and set spin to -360 degrees for 7.5 seconds 5 times so the robot goes around in a figure 8 five times
* Set spin to 360 degrees for 7.5 seconds
* Set spin to -360 degrees for 7.5 seconds
* Set spin to 360 degrees for 7.5 seconds
* Set spin to -360 degrees for 7.5 seconds
* Set spin to 360 degrees for 7.5 seconds
* Set spin to -360 degrees for 7.5 seconds
* Set spin to 360 degrees for 7.5 seconds
* Set spin to -360 degrees for 7.5 seconds
* Set spin to 360 degrees for 7.5 seconds
* Set spin to -360 degrees for 7.5 seconds
* Input stop robot
* Input speak “I am the winner” and wait
* Input color fade to another color for 1 second
* Input color fade to another color for 1 second
* Input color fade to another color for 1 second
* Input color fade to another color for 1 second
* Input color fade to another color for 1 second
* End the program

**Executive summary:**

The products intended audience is the professor and our classmates and maybe even some outsiders just observing and watching the project. This project is a robot and it can roll around at different speeds, talk, change color for a certain amount of time, and much more.

The purpose of the project in the scope, is to show that the robot can go around in a figure 8 direction following the arrows five times. It cannot go off course like straying from the arrows or the arrows direction, bump into any objects, and it has to end at the starting point. It also has to glow multiple colors for five seconds and it has to say “I am the winner” when the robot stops moving at the end of the course. Outside the scope, the project has to be coded in such a way that no errors happen like bumping or going off course and everything needs to run into order so no malfunctions or mistakes happen while the robot is running.

**Product description:**

This product which is the robot does relate to other products out in the world. One of the products would be the Roomba which a robot or machine that rolls around and sucks up dirt and dust like vacuums. The robot is self-contained and is independent as long someone programs the robot to do certain actions. It has sensors that make the robot change directions whenever it bumps into any object. The robot can interface with any computer systems like laptops, desktops, computers, etc. which are the larger systems. The interconnections would probably be the computer system and coding controlling the robot’s actions. The external interfaces would be the Wi-Fi that is connecting the computer system and the robot.

The customer profile for a student would be that they would have to have some experience and technical expertise in how to write an algorithm, flowchart, syntax language, coding and some knowledge in software engineering to use the robot. They would also have to practice and experiment using the robot a couple of times to know how to use it well.

The customer profile for the faculty/staff would be having complete knowledge and a lot of experience and technical expertise on how to set up this program and how the program will work to use the robot. They will have to know the ins and outs of the robot’s characteristics completely. The customer profile for people that do not know how to use the robot or what is does would probably learning more about computer science, software engineering and observing the students and faculty/staff.

Some assumptions that may affect the requirements would be the equipment availability. If you cannot access the equipment then you will have to change everything around. You will have to contact the person in charge of the robots and see when you can get the robot as soon as possible and then instead of working on the code first, change the Gantt chart and work on the staff plan, executive summary, product description and requirements chart if you have not already done the algorithm and flow chart in the meanwhile. Also, the same would go if we still need to use the robot even after we give it back or need to use Howard Hall room to test robots and it isn’t available.

Some items that will cause constraints would be if the app or even the computer system device itself that is being used to control the robot is malfunctioning. For example, if it is lagging or it is just not working or letting you do certain commands as normal. Perhaps the code is too long and there is an overload on the device so its robot will not be able to perform the activity that it is supposed to do. The people working on the project may also just be having a hard time figuring out the coding or how to work the robot as well.

The dependencies that will affect the requirements are doing the project in the right order. First of all, you will need to sign out the robot to even just experiment and test it. Before even testing or writing the code, you will need to understand what you are trying to do with the program and then write the algorithm and then flowchart. Then you will need to understand the program that you are using to control the robot. After that, you will need to try testing the code that is written to see if it works multiple times.

**Requirements:**

The (#) show the requirement number. The first requirement (1) will be to contact the person that is in charge of the robots to set a date to sign out a robot in Howard Hall in order to use that robot later to create and test the codes. Next, we should assign the staffing plan (1) to make sure everyone has their own roles and responsibilities so that they will know that they will have to work on for the project. The Gantt chart (1) will be created to know exactly when everyone should start working on their part, and what parts they are working on in the project. It shows when people have finished their parts and how many tasks there has to be completed in the project. The requirements (1) describe the system requirements in much detail to satisfy and verify the system requirements. The algorithm (1) are the extremely important instructions of the program that show what exactly what should be in the code and how the code should look like the exact units/numbers, equations and what it is that is being written. The flow chart (1) is the algorithm converted into a chart like picture that is easier to understand since it visually shows what the code of the project should look like in order. The coding (1) affects how the robot will move around and what actions the robot is doing as long as everything is programmed correctly. The test plan (1) shows what mistakes have occurred and the corrections to that mistake. The actual testing (1) is the trial and error that made sure the code was working and running smoothly. The requirements confirmation (2) shows the group confirming what has been done on the days that we were working together on the project. The hardware (3) describes the hardware like the computer systems that created the code and application. The software (3) describes the software like the apps and syntax language used that created the code and application. The executive summary (3) described the objective and purpose of the program. The product description (3) compares the product to other products out there, the customer profiles and the constraints affecting the requirements. The presentation (2) at this point was worked on a little.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Req #** | **Requirement** | **Comments** | **Priority** | **Date Rvwd** | **Reviewed/**  **Approved** |
| 1. | Sign out robot | Properly signed out robot | 1 | 11/6/19 | Nicholas |
| 2. | Staffing plan | Assigned people tasks | 1 | 11/6/19 | Kathryn |
| 3. | Gantt chart | Saw if everything was completed | 1 | 11/6/19 | Kathryn |
| 4. | Requirements | Verify system requirements | 1 | 11/6/19 | Joseph |
| 5. | Algorithm | The instructions of how exactly the code and project should run | 1 | 11/6/19 | Kathryn |
| 6. | Flow Chart | Made algorithm into a chart | 1 | 11/6/19 | Joseph |
| 7. | Coding | Created final code | 1 | 11/7/19 | Nicholas |
| 8. | Test Plan | Tested code to see errors | 1 | 11/7/19 | Nicholas |
| 9. | Actual testing | Made sure it ran smoothly | 1 | 11/7/19 | Nicholas |
| 10. | Requirements confirmation | We made sure we did all the documents correctly | 2 | 11/11/19 | Kathryn |
| 11. | Hardware | Described hardware used to demonstrate application | 3 | 11/11/19 | Joseph |
| 12. | Software | Described software used to demonstrate application | 3 | 11/11/19 | Joseph |
| 13. | Executive summary | Summarized objective of robot and project | 3 | 11/10/19 | Kathryn |
| 14. | Product description | Described the robot itself and its requirements | 3 | 11/10/19 | Kathryn |
| 15. | Presentation | Worked on presentation a little | 2 | 11/10/19 | Kathryn  Nicholas  Joseph |

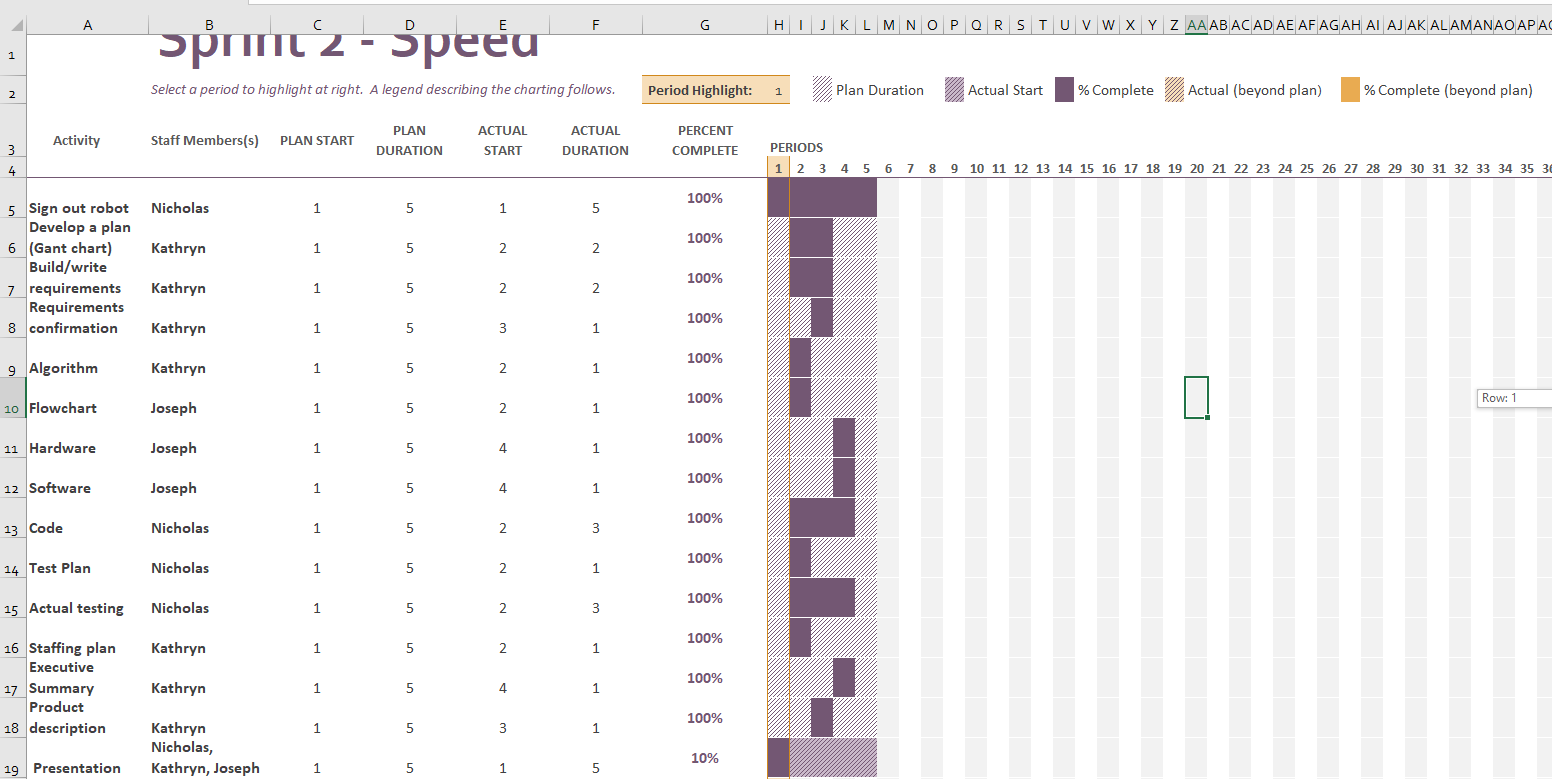
In order to protect the code, one could set up some type of program or defense system to pop up whenever an unknown user is trying to access the code. Also, a password could be set to access the code. Make sure to save the code in multiple locations as well in case the code gets altered or deleted by accident.

In order to be authorized to use and or see the code, the app Sphero Edu requires people to create an account to access their own code. Like protecting the code, one could also download or create multiple programs for someone to be able to access their own code.

It is possible to port the robot’s builder code into JavaScript language on the Sphero Edu app. It is also possible to convert the code created from the robot to a phone from a computer or vice versa as long as you have a Sphero Edu account. It is easy to move the robot because it is small and portable and it is not hard to charge it either.

**Requirements confirmation:**

|  |  |  |
| --- | --- | --- |
| **Meeting Date** | **Attendees** | **Comments** |
| 11/7/19 | Nicholas - Code writer  Joseph – Planner | Confirmed  Started working on project and code |
| 11/10/19 | Kathryn - Information gatherer  Nicholas - Code writer | Confirmed  Testing robot and finalizing code. |
| 11/11/19 | Kathryn - Information gatherer  Nicholas - Code writer  Joseph - Planner | Confirmed  Finalizing all documents and reviewing before submitting |

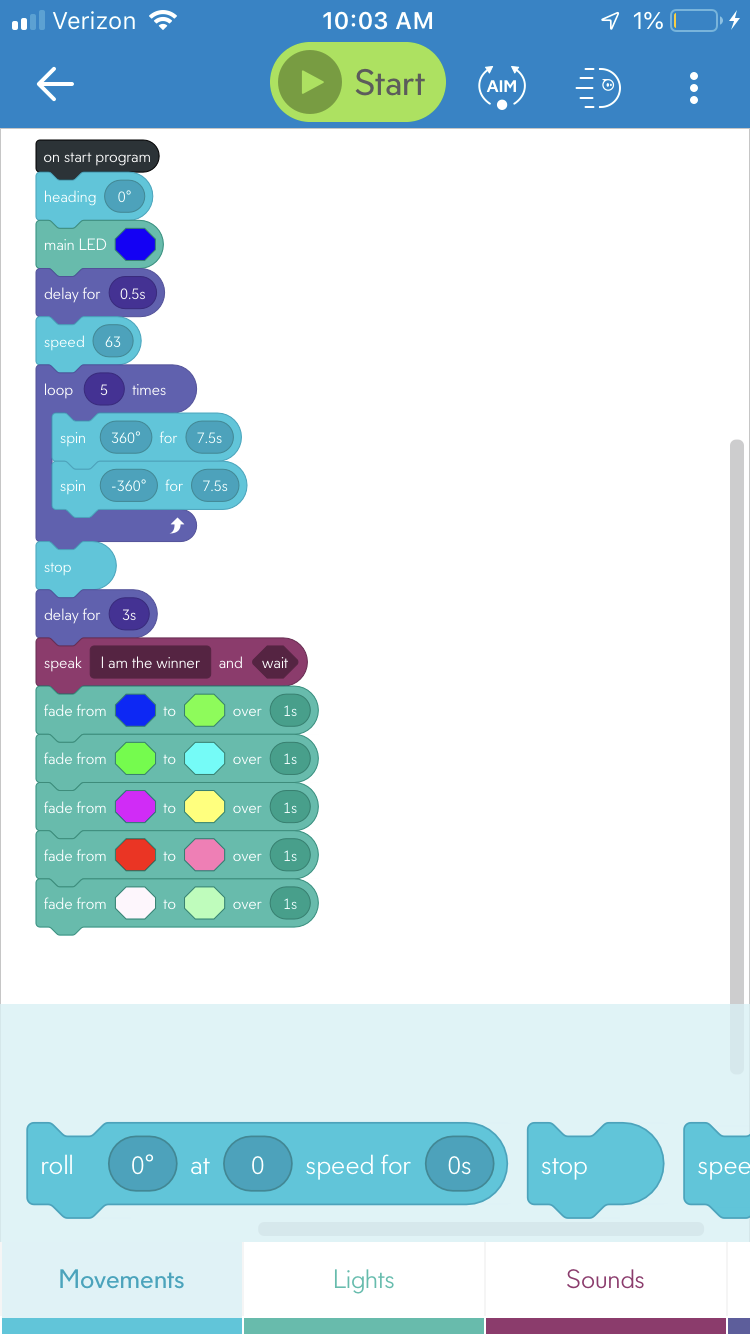
**Gantt chart: **

**Staffing plan:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Responsibility** | **Reports To** |
| Kathryn | Information gatherer | Submitting all documents together on GitHub, reviewing documents, & making team collaborate | 11/11/19 submitting all of the documents |
| Nicholas | Code writer | Creating the code, testing the code out, making adjustments to the code | 11/11/19 finalizing code |
| Joseph | Planner | Planning the flowchart and software and hardware information. | 11/11/19 making sure everything is ok |

Nicholas Smythe

**Code:**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reason For Test Case | Test Date | Expected Output | Observed Output | Pass/Fail |
| First test to see if it completes the requirement of the speed test loop infinite times to see if it continually goes in a figure 8 | 11/7 | Goes in figure 8 cleanly all the time will change the number of times to 5 at later time | Goes in a figure 8 sideways | Fail |
| Reaim the robot to the path instead of the person | 11/7 | Goes in figure 8 many times to stay on path will change numbers if off a bit | Now somewhat on path but offshooting a bit when on the massive turns needs to be tighter and need to reduce the loop to 5 times | Pass |
| Redo the loop to 5 times instead of infinite | 11/7 | Goes in figure 8 5 times | Went in figure 8 five times | Pass |
| Speed was at 150 overshooting way too far decrease to 70 and increase time turn from 6 to 8 | 11/7 | Cleanly goes in figure 8 | Figure 8 was overshot since it was too fast for the time amounted | Fail |
| Decrease speed to 63 and decrease the time to 7.5 | 11/7 | By decreasing the speed and the time it will turn at the right time and go in a figure 8 cleanly | It went in the figure 8 cleanly sometimes on the basis of not having a complete flat surface | Pass |

**Tested robot**

**Signed out robot**

Joseph Scerbo

**Hardware:** The hardware that the robot uses is the computer and the phone or any computer system that can use the website/app Sphero Edu that is used to be able to work on the program. The robot itself is also part of the hardware that is used in the testing and development part of the project/product.

**Software:** The software that the robot uses is the Sphero Edu app which you can manually program the actions and functions or you can write JavaScript code since it is also part of the app. It is also possible to convert the programs functions and actions into JavaScript. As long as there is software, the program will be able to run its course.

**Flowchart:**

