



Robotics Triathlon

By : Chelsea Spencer, Emily Williamson, and Jordan Strobing



Challenges We Faced

- Struggling to find times where each group member to meet via zoom (due to busy class schedules)
- Issues with Robot
 - Faulty robot (had to get a new one)
 - Robot would not pair to Emily's phone multiple times
 - Robot had weak battery (consistently died even though app said it was fully charged)
 - Robot was having issues with stabilization (would veer in wrong direction although app showed accurate aim)
 - Initially had issues with trying to make robot spin in a complete circle
 - Initially had issues making robot stop even though we had a stop block in block code
- Coronavirus
 - Not being able to all meet in person in Howard hall
 - Communicating only via Zoom
- Had an issue getting collaboration on Github to work

Roles of each Team Member

- Jordan

Team Leader, Staff Manager, Gantt Chart Manager, Programmer, Software Developer

Led and oversaw group in all operations, worked closely with team members to develop programs, algorithms, requirements and software. Recorded the test table while Emily was testing on site and developed the plan in which the projects were executed.

- Emily

Team member, Software Tester, Idea Contributor, Software Developer, Repository owner, Robot owner

Worked with team members to develop the program, led the testing portion of the sprints in Howard Hall, owned the robot, owned the Github repository for the 3 sprints, and contributed ideas to the overall project.

- Chelsea

Team Member, Algorithm Developer, Idea Contributor, Flowchart Developer

Led the team in developing the algorithm and flowchart, helped with the program due to knowledge of algorithm, contributed ideas to the overall project, helped fill out the System Design Document..

What we have Learned about Software Engineering

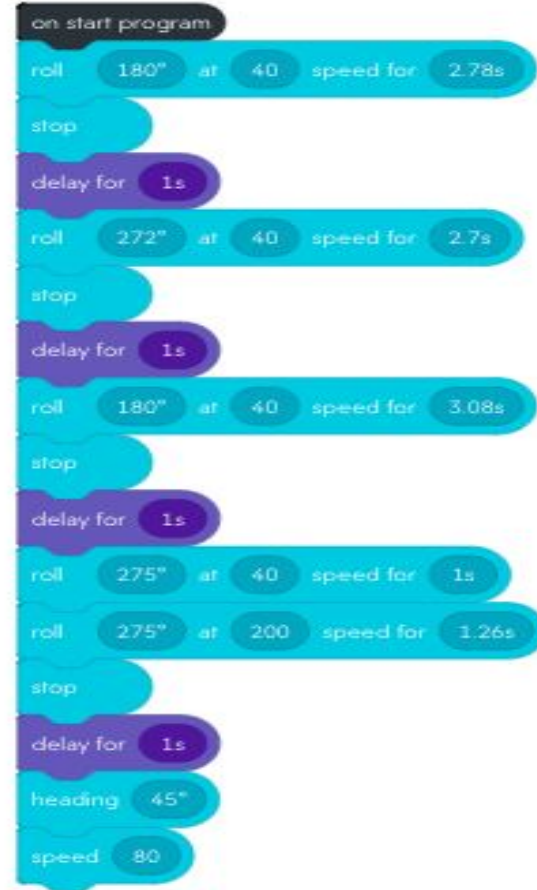
- It is important to stay organized.
- Always start with a set plan and requirements.
- Using the Gantt chart is important to help us stay on track, and keep track of what we have done.
- Testing is important to reach desired outcome. What we thought would work sometimes would be the exact opposite.
- Its okay to fail, observe what happened, and adjust based on prior outcome.
- Collaboration is important, having the ability to work well together and divide tasks to reach a common goal is beneficial for meeting deadlines .
- To share ideas, and discuss different point of views of the same plan to see which is most efficient. Communicating throughout helped each of us.
- How detailed you need to be before even starting the project. You need to explain exactly what you want to do in the System Design Document before getting to the practical part

What We Would do Differently

- Having multiple/different sets of code ready before testing in case of failure to prevent wasted time of redoing the entire code.
- Having set weekly meeting times on days we were all available, to prevent stress/issues of figuring out a day that all members were available.
- Being able to meet and work in person to allow better communication.
- Read more tutorials on how the robot worked, instead of experimenting and using trial and error.

Block Code for Sprint 3-Agility

- Adding the delays in the block code helped the robot stay more accurately on its course.
- The code was similar for the first 3 lengths were similar due to them all being straight and a similar distance.
- There was a steady speed of 40 until ramp, due to robot struggling to get over it.
- Instead of using general roll function for last length, we used to the heading function, because we felt that it allowed program to run smoother and more accurate.



Sprint 3 Video-Agility

- Emily was the group member to go to Howard Hall, while she was at Howard Hall the rest of the group was on zoom observing the testing and equally collaborated on the work.

