Bodacious Battle Bots

PROJECT

RETROSPECTIVE

Project Retrospective

Table of Contents

SUCCESS……………………........................................................................................................2

LESSONS LEARNED…................................................................................................................4

PROJECT TIMELINE.....................................................................................................................5

Sprints Summary.................................................................................................................5

CONCLUSION……………………………………………………………………………………1

**Success**

We had good team meetings, everyone helped and collaborated during each meeting proving valuable feedback. We used GitHub to share, update, and monitor code, which made it quite simple to collaborate and work together. Our team was able to not let negative things impact us. The team worked hard and always tried to help others. We learned a lot of how to use APIs and sending requests. Our team worked extremely hard in all the tasks assigned and almost all the deadlines established were fulfilled. Our team was also able to stablish a good and efficient communication with the Misty Mountain group.

**Lessons Learned**

In the beginning of the project, we did not know much of what to expect and how hard a lot of things could be, and how easy some others could do. We ended up taking a lot of skill to complete, which made us a little bit overloaded. Preventing us from not being able to finish all the interaction and implementations that we planned to do.

This project based on research, such as what the robots could do and how to do those things. Unfortunately, there are not a lot of resources on Misty, since it is a new product and not a lot of people have it. Therefore, most part of the problems that we encountered were not faced by other people before, which made the work of trying to find solutions harder. Sometimes we could not even find a solution to the problem. Some of these problems made us spend lots of time in things that did not lead to a solution.

We were not able to implement the use of sockets in order to allow direct communication between two robots. The methods for sockets provided by the Misty developers seemed to have a lot of bugs and did not have proper documentation on how to use them. Furthermore, since misty does not allow us to install any libraries or use different programming languages, we could not implement it through JavaScript, neither try to use other programming languages.

Lastly, the fact that we did not have a lot of information about Misty before starting the project made it hard to estipulate how much time things would take to be completed. Some parts that we thought that would take around a month, ended up taking the whole semester and not being a hundred percent done, such as the dashboard and tic tac toe. While some other tasks that we anticipated to take around a month were done in couple of hours, such as security cameras. We also did not calculate the work that we would need for the documentation before it was mentioned in class. We were thinking only about comments and read.me files.

**Project Timeline**

**Sprints Summary:**

* Sprint 0 – Start:

We defined what our team was going to do based on the client’s preference list. We chose which skills we wanted to do, talked to our client to get her approval. We also defined the team roles and responsibilities. Lastly, we established the points and time needed to implement each part.

* Sprint 1 – Week 1:

We researched a lot about the robot. We also had to teach/learn the details of the robots before we started to code it. The first skill was developed: Dancing.

* Sprint 2 – Week 2-3:

We researched more. We also thought some simple commands and how to run the code to the Misty Mountain members that had not worked with Misty before. Our group started the dashboard. Our designs were approved by the client, and we were able to start the implementations of more skills. We also moved forward in the detecting objects and face recognition skills.

* Sprint 3 – Week 4:

Our team worked on the first status report. There was a lot of work done on the backend of the Dashboard. We finished the detecting objects skill. We also worked on reaction and rock paper scissors skills.

* Sprint 4 – Week 5:

We mainly worked on the presentation for the status report. We tried to get everyone on the same page and working/communicating a little more to get back on track.

* Sprint 5 – Week 6:

We had great progress in this week. We were able to complete text to sound and sound to text skills. Dashboard was progressing, being able to send POST requests to misty and pull data from it.

* Sprint 6 – Week 7:

Our group finished the translation skill and there were a lot of progress with the conversation skill. We were ahead of schedule and a lot of things were working very well. We had great progress too.

* Sprint 7 – Week 8:

Our team worked on presentation for status report 2. We were almost completely done with rock paper scissors, tic-tac-toe, and security cameras skills. Dashboard was almost done. Good work on designing and interacting with robots.

* Sprint 8 – Week 9:

We worked on documentation.

* Sprint 9 – Week 10:

Our group worked on final presentation.

**Conclusion**

There are only a couple of things that we would change if we were to start all over, but now knowing what to do, how long things would take, what works and what does not.

To begin with, and probably the most important thing, we would change how we approached the project and decided what we wanted to do. The main reason why we chose to do things the way we did was because there was a possibility of finding something that we could not do, and that thing could hold us of from delivering any skill at all. Instead, we decided to create a dashboard and do a lot of skills for Misty. However, the most important thing for the client was translation, which was a skill that we were afraid of not being able to deliver properly. Knowing now that translation is possible to do using the robots, we would have changed to only do a translation application and make that a big skill with lots of different functionalities. This would solve a lot of problems that we faced during the project and make the workload of the whole project a little fairer than what it turned out to be.

In addition to that, we would have worked more closely in the communication between robots. It took a lot of time and work to try to make the communication between robots work and we just managed to do it in some of the skills. Even though it was on our priority list, it turned out to be way harder than any other part in the project.

Lastly, we would also have changed the timeline a bit. With complete knowledge of how long things would take, changing the timeline to match more closely the resources needed for each task would have helped us to be a little more precise in our sprint division and work division.