

**[CS 490] Sprint 1 Report, Team 6 [mccullougha]**

shelbyemailrelay@gmail.com &lt;shelbyemailrelay@gmail.com&gt;

Sat 4/6/2024 10:11 AM

To: McCullough, Alysha &lt;amccullough5@ewu.edu&gt;

Alysha,

This report describes the activities of your EWU Senior Project team over the previous self-evaluation period (usually Saturday through Friday). It contains only public information. Private information and comments, etc. are available only to the instructor. If you notice any discrepancies or have questions, please contact Dan Tappan at [dtappan@ewu.edu](mailto:dtappan@ewu.edu).

**Sprint 1 Team Report**

Team 6: Argumentative Writing Game

- Alysha McCullough
- Curtis Melton
- Nicholas Parkman
- Clark Rabe
- Bryan Snyder

**Logged Hours**

The team is generally free to work whenever they want during the sprint. The expectation for a team of five members is 75 hours total (15 per member) on average. However, this number will vary throughout the course.

Individual Hours:

Member	Hours	All Sprints							
		Total	Min	Max	Avg <sup>1</sup>	Avg <sup>2</sup>	Std <sup>2</sup>	Count <sup>1</sup>	Missed
McCullough	2.0	2.0	2.0	2.0	2.0	2.0	0.0	1	0 (0%)
Melton	6.0	6.0	6.0	6.0	6.0	6.0	0.0	1	0 (0%)
Parkman	2.0	2.0	2.0	2.0	2.0	2.0	0.0	1	0 (0%)
Rabe	17.0	17.0	17.0	17.0	17.0	17.0	0.0	1	0 (0%)
Snyder	2.0	2.0	2.0	2.0	2.0	2.0	0.0	1	0 (0%)
Team Total:	29.0								

<sup>1</sup>including and <sup>2</sup>excluding missed submissions for required sprints

Team Hours:

Sprint					
1	Total	Min	Max	Avg	Std
29.0	29.0	29.0	29.0	29.0	0.0

The following is optional descriptions of daily work that is not captured as activities below:

McCullough:

- Looked into HackBox

Rabe:

- updated README on github and Uploaded project via git to separate github repository
- Found and researched a plug in for Unity called Hackbox
- Implemented Hackbox plugin into project and pitched the idea to teammembers

Snyder:

- Revising and submitting abstract to symposium.
- Group meeting discussing multiplayer implementation.

## Activities

---

Activities are member-defined units of work that are formally tracked from sprint to sprint (unlike the optional descriptions above). Every activity must be accounted for from its creation until it is completed or abandoned.

## New Activities

---

These activities were created by during this sprint.

McCullough

### Activity 2: Sprites

10-15 sprites need to be created for the game. (one sprint expected)

Melton

### Activity 1: Learn more about Hackbox

Learn more about Hackbox, as we may be using it for our networking solution (one sprint expected)

Parkman

### Activity 7: reviewing hackbox git asset

becoming familiar with a networking asset for our project (one sprint expected)

Rabe

### Activity 3: Hackbox

Unity plugin to allow multiplayer functionality using phones as controllers (three sprints expected)

#### **Activity 4: README**

Updated README.md as instructed by professor (one sprint expected)

Snyder

#### **Activity 5: Abstract**

Revise and submit project abstract to creative symposium (one sprint expected)

#### **Activity 6: Review hackbox git assets**

study how to use the hackbox github framework (one sprint expected)

### **Team Reflection**

---

This section refers to the team's collective perception of and reflection on the project over this sprint.

The instructions are: Consider the following four pairs of questions hierarchically. They are not the same question. If you think they are, then you are likely not using an appropriate breadth and depth of software-engineering thought. This course is a practical application of the aspects of product, process, and people. We are trying to account for everything: not just to create a good product, but also to learn from the process to improve the people. Reflect on the experience of the entire team collectively over this sprint. You do not need to account for all work, just two examples that are most representative of easiest and hardest.

For reference, *understand* relates to the comprehension of what needs to be done; *approach* to how you think it should be solved; *solve* to implementing the actual solution; and *evaluate* to demonstrating to yourself and your team (if applicable) that the performance of your solution is consistent with everything else in the project. Remember [The Cartoon](#) from CS 350.

### **Understand**

---

- |                 |   |
|-----------------|---|
| <b>Easiest:</b> | Our user interface design is easy to understand. We have been able to quickly and effectively brainstorm different themes and styles we intend to use. Our client has also been able to provide feedback on UI/UX because of its less-technical nature.   |
| <b>Hardest:</b> | The networking aspects of our project are the hardest to understand. We are looking into using Hackbox (a Jackbox style networking solution package for Unity) to aid some of the networking aspects of our project such as: real-time data synchronization, facilitation of individual player states and interactions within the game, and a hosting server. |

### **Approach**

---

- Easiest:** User interface and design has been easiest to approach. We have all been experimenting with Unity, and we feel confident about creating menu pages and basic scenes to begin designing UI/UX components to fit our theme and functionality requirements.
- Hardest:** Our networking solution has been the hardest to approach. Now that we have found a possible package to help aid the process (Hackbox), we feel more confident about approaching the networking functionality. By having access to the Hackbox GitHub repository, we feel confident we can either use Hackbox as a package to aid our design, or study their design to create our own networking solution.

## Solve

---

- Easiest:** User interface and design has been easiest to solve. Being that we already have a strong idea of how we want our game to look and function, we can begin implementing these aspects. We all have different levels of competency with Unity, so we will be collaborating and learning more about the software individually.
- Hardest:** Networking has been the hardest to solve. We are on the right track with researching and possibly implementing tools such as Hackbox to facilitate individual player states and interactions within the game.

## Evaluate

---

- Easiest:** The UI/UX design aspects are easiest to evaluate. We can quickly convene and make UX/UI decisions because the problem is also easy for us to understand and approach. It is also easy to have our client evaluate the UI/UX design because it is less technical than other aspects of the project, thus our client can easily provide feedback for us.
- Hardest:** The networking aspects are the hardest to evaluate. This is because the networking design encompasses many aspects which can also be challenging to understand and approach. Networking aspects include performance/stability, scalability, security and accessibility/compatibility.

**Completion:** 5%. Yes this pace seems likely to succeed.

**Contact:** No. March 12.

**Comments:** N/A

---

Report generated on Sat Apr 06 10:11:52 PDT 2024

**CAUTION** This email originated from outside your organization. Please exercise caution when clicking any links or opening attachments.