

Individual Lessons

- Megan Postava:
 - I learned the importance of version control, and how essential time management is. I also learned that working with a team this large is tough, not only because of schedule management, but also because everyone has different coding styles. A lot of it came down to successful communication, which I also learned a lot about. There were also many things about spriting that I learned while creating the enemies.
- David Moore:
 - I learned how important it is to agree on standards and conventions for your project before any work is actually done. This allows for everyone to know how to name new files and objects and where to put them so that there is some logic in attempting to understand the code someone else wrote. Communication and shared vision is key to a successful product.
- Jonathan Bennett:
 - This project was a good reminder to have each team member work on tasks they are best suited for, so the team will benefit from their expertise in their respective fields. Our team also benefitted by having good communication, which involved meeting often and taking detailed notes we could reference later. I learned a lot about networking between a desktop client and a web service when creating the scoreboard system.
- William Adkins:
 - The main lessons that I learned are that Clear, Frequent communication with teammates, especially in a large group situation is of the utmost importance. On top of that, having the team meet frequently is helpful in making sure that everyone is on the same page, and keeping up with their individual tasks.
- Katie Jurek:
 - Version control is nice when implemented early and simply, though it can be a true pain if something goes wrong or if you're confused on how to use it. Programming in a large team, while not always seeming so unwieldy at the beginning, can be annoying. I prefer to take the reins, and it admittedly drives me a little crazy when I can't. Programming time is often increased with multiple people due to the time needed to delegate tasks to each member. I also learned a lot about documentation, which I can see as being necessary as the group increases in size such as at massive companies like Microsoft.
- Miguel Corona:
 - The most important lesson that I can take away from this project is how important it is to maintain concise documentation. In order to produce great documentation, everyone in the group must understand the ideas that the group has discussed. Being able to refer to the initial documents later in the project development allowed our group to stick to a set of ideas that we decided on early on.

Team Lessons

- Importance of Communication:
 - It was incredibly important to keep in regular contact with our team to know which tasks were being done by whom and how much progress each member was making. Also, if there was a problem, or a feature ended up seeming too daunting to put in the final game, we were able to discuss it with the members who were most available first, which influenced how much of a role each had in the end.
- Importance of Clear Documentation:
 - Clear documentation helped the team to be able to focus on implementation details while simultaneously not losing track of the bigger picture. If the documentation was vague in an area, it would cause contention when group members attempted to reason out what was meant by it.
- Importance of Meeting Regularly:
 - Meeting regularly was definitely important, and it was something we did from the beginning. It allowed us to clearly defined what we were doing for each week, and it led to the creation of a successful final product.