# EECS 448 Project 2 Retrospective

*Runtime Terrors (Abby Davidow, Anissa Khan, Grant Schnettgoecke, Jacob Swearingen, Chongzhi Gao)*

**Meeting Log**

| | | | |
|---|---|---|---|
| 9/27 | Spahr | Everyone | Talked about possible ideas |
| 9/30 | Spahr | Everyone | Determined animations and sound |
| 10/2 | Spahr | Everyone | Talked about how to split up AI |
| 10/2 | Spahr library | Everyone | Split up work load between members |
| 10/7 | Spahr | Everyone | Asked about changing additions |
| 10/9 | Spahr | Everyone | Discussed bug |
| 10/9 | 1131 Learned | Everyone | Fixed bug, worked on animations, sound, and AI |
| 10/11 | Study room | Abby, Anissa, Grant, Chongzhi | Finished baseline AI functions |
| 10/16 | Study room | Everyone | Worked on merging branches, Anissa and Grant fixed ai medium bugs |
| 10/18 | Spahr | Abby, Anissa, Grant, Chongzhi | Fixed bug and planned finishing project |
| 10/18 | Anissa's house | Anissa and Jacob | Fixed scoreboard bug |

**Division of Labor**

For this project, we split the work up into five clear sections. Abby and Jacob coded the additions made to the game. Abby created the animations played after a hit, miss, sunken ship, and once a player wins. Jacob was originally going to create the sound addition to the project, but the team later decided to switch to a scoreboard addition instead. As a result, Jacob then created a scoreboard that took a winner's initials and their score and displayed it on a leaderboard. Anissa, Grant, and Chongzhi were each responsible for an AI difficulty.

They all met together multiple times to organize gameManager.cpp and Player.cpp to understand how the project was organized and how to extend it to support an AI. Together they wrote the code to support the AI. Once the structural code was in place, Anissa wrote AI easy, Grant wrote AI medium, and Chongzhi wrote the AI hard. Chongzhi also wrote aiTurn, which provided the basis for all three AI difficulties, and placeAIships which randomly places the ships on the board. Together the three of them made some minor changes to those methods to make them more suited to the AI. Anissa and Grant later spent time together debugging and completing AI medium.

**Challenges**

Some of the challenges of this project included reading another team's code and debugging. It took awhile to figure out how the other team organized their project. Their documentation was helpful, but it was hard to get out of the mindset of how we set up our game versus how they handled the project. Their program had a bug that caused the program to crash, so we ended up having to learn their code more thoroughly in order to find the problem.

However, this leads into our other challenge of debugging. Only one person on our team had a debugger they felt comfortable using, so figuring out issues was inefficient because we had to depend on one person to use their debugger.

**Features**

The two additions made to the HackStreet Boys' Battleship game were animations and a scoreboard. Something we would have liked to have changed with the animations is how long it takes between pictures. The sleep function that we used only accepts integer parameters, so we had to pause for at least one second between animation changes. A different library may have had a slightly different function that would pause the program for less time.

Originally, our team wanted to add sound to the program, so the animations would have corresponding sound effects. However, after researching and attempting to add sound into a C++ program, we determined that we would not be able to include sound within the time allotted for the project or without attaining aid from the IT department.

Also, if we would have had a little more time, we would have liked to redesign the UI elements of the menu and player boards. Right now they are fairly generic, so we would have made the board look more like a grid and added some word art to the menu.

**Retrospective**

Overall, our team improved a lot from last project. Our communication greatly improved as we put all problems and questions into our group chat. Team meetings were communicated well and were extremely productive in getting work done and figuring out bugs. We also started reserving study rooms in Spahr library, which improved productivity as we had our own space to talk and draw on white boards. The team found a time that seems to work well for everyone to meet, and for the next project, we intend to make this time a weekly meeting to make sure the project stays on track.

Also, having a written code base to branch off of made it a lot easier to split up work and create branches on GitHub. The work was divided much more evenly for this project, and everyone finished their work in a timely manner. It will be crucial to

continue to divide the work evenly and strategically in future projects. Some things we could keep in mind while dividing up work are people's abilities and availability to meet outside of normal team meetings if their part requires collaboration with another team member.

Our team's GitHub ability greatly improved during this project. We each created our own branches allowing us to work individually, which greatly prevented merge conflicts and prevented an overlap of work. With the team's better understanding of GitHub, we were able to merge branches with ease and worked through merge conflicts quickly. We also figured out how to rebase or pull changes from master to our branches so that we could utilize newly written code in our additions.

For the next project, we each should try to download a good debugger on our devices, so when we run into an issue, each of us has something to help us work through the code. Another option would be to go into the KU computer labs to utilize GDB or another debugger provided on the computers. However, this might not be plausible if we are not on campus so having one on our personal devices is a necessity.

Also, some of the planning techniques we learned in class would be helpful to implement for the next project in order to prepare a well-laid out blueprint before starting. We made a simple plan for this project but fully drawing out a plan will definitely help us organize ourselves especially if we are starting from scratch.