

Project 2 Retrospective Write-up for “Big SegFault Energy”

Team Members:

Chance Penner
Haonan Hu
Markus Becerra
Sarah Scott
Thomas Gardner

Git Link:

<https://github.com/h192h407/Battleship-1>

Date finished: October 20, 2019

Group Meeting log:

Meeting 0:

Location: Eaton Hall Lawr2

09/27/2019 @ 11:30AM to 11:50AM

All in attendance

Agenda:

- *Brainstorming for extra features that we need to add

- *Nuke on hit streak

- *UAV radar(probably)

- *Trying to solve Compiler Error

Meeting 1:

Location: Eaton Hall Lawr2

09/30/2019 @ 11:35AM to 11:50AM

All in attendance

Agenda:

- *Fixed compile Error(Thanks to Markus)

- *Scheduled for next meeting(10/2 Wednesday)

Meeting 2:

Location: LEEP2 1328

10/05/2019 @ 12:00PM to 5:00PM

Haonan, Chance, Thomas

Agenda:

- *Class diagram for the project

- *Bug fixes

- *Get program ready for implementing extra features

Meeting 3:

Location: LEEP2 1328

10/06/2019 @ 12:00PM to 5:00PM

Chance, Thomas

Agenda:

- *AI(easy mode) is done

- *Start with hard mode

Meeting 4:

Location: LEEP2 1322, Eaton 1005C

10/09/2019 @ 9:00AM to 10:50AM

All in attendance

Agenda:

- *Help group members have a good understanding of the classes function

Meeting 5:

Location: LEEP2 1324, Leep2 Ground Floor

10/11/2019 @ 9:00AM to 10:50AM

All in attendance

Agenda:

- *Fixed median difficulty AI would stack on while loop if you put three horizontal battleships together

Meeting 6:

Location: LEEP2 1324, Leep2 Ground Floor

10/12/2019 @ 2:30PM to 5:02PM

Thomas, Chance

Agenda:

- *Fixed popups for all gamemodes
- *Attempted to fix button highlight bug

Meeting 7:

Location: LEEP2 1324

10/14/2019 @ 12:00PM to 4:00PM

Sarah, Haonan, Chance, Markus

Agenda:

- *Fixed nuke AI
- *Reformatted files

Meeting 8:

Location: LEEP2 1322

10/16/2019 @ 9:00AM to 10:50AM

Sarah, Haonan, Chance, Thomas

Agenda:

- *Add nuke text for player 1
- *Add radar button

Meeting 9:

Location: Leep2 ground floor Alcove

10/18/2019 @ 4:00PM to 5:00PM

Haonan, Chance, Thomas

Agenda:

*Fixed Nuke text bug for both players

Meeting 10:

Location LEEP2 1326

10/18/2019 @ 6:30PM to 8:30PM

Markus, Haonan, Chance, Thomas

Agenda:

*Radar implementation

Meeting 11:

Location LEEP2 1326

10/18/2019 @ 8:30PM to 10:00PM

Markus, Chance, Thomas

Agenda:

*Compete radar implementation

Meeting 12:

Location LEEP2 1324

10/19/2019 @ 4:00PM to 6:00PM, 7:00PM to 9:00PM

Chance, Thomas

Agenda:

*Radar bug fix

*Begin commenting

Meeting 13:

Location LEEP2 1324

10/19/2019 @ 11:00AM to 1:15PM

Haonan, Thomas, Chance

Agenda:

*Commenting/documentation

*Resolve merge conflicts

Meeting 14:

Location LEEP2 1324

10/19/2019 @ 1:15PM to 3:30PM

All in attendance

Agenda:

*Finish retrospective/documentation

*Turn in final version of project

Work Distribution:

For this project, we needed to add an AI game mode with three difficulty modes (easy, medium, and hard). We chose to add two additional features, a nuke shot, and a radar. To get the program to compile on our machines, we had to downgrade our Java versions to JavaFX 8. Haonan and Markus took responsibility for figuring out the necessary steps to get our machines to compile the project. Chance, Thomas, and Haonan were in charge of the versusAIGUI. Chance and Thomas then completed the easy and hard mode for the AI option. All of the group then met and planned out and worked on the medium difficulty. Markus began the nuke, and the rest of the team joined in to help. The radar was planned by Haonan, Markus, Chance, and Thomas. The radar got finished later that night by Markus, Chance, and Thomas. When each addition was created, Sarah and Thomas added/fixed the popups for them. We did not split the work into classes, as the project we received was in a mega class. Overall, each member did a fair amount of work and we were able to finish the project on time.

Challenges:

Our first was compiling the project we received. The project was coded using JavaFX8, which is a vastly outdated version of Java. We each had to downgrade our machines to this Java version, which took more time than expected. Haonan and Markus did hours of online research to figure out how to remove the current version of Java and replace it with an older one. Haonan and Markus then simplified the steps so that Sarah, Chance, and Thomas could get it to compile on their machines. This process was a total of around 5 hours between the two.

Another challenge we faced was figuring out how “e.getSource()” functions, since we needed to get an AI player to work. Since the entire handle function was waiting for an input as a mouse click (clicking on a location on the board), we attempted to hard code mouse clicks for the AI player. This was not possible for JavaFX8. We then had to copy all of the code for human player two functions and paste them into an AI function that did not expect a mouse click for input.

The overarching challenge throughout the whole project was coding in a language we were not familiar with (Java) and doing so without a single comment given to us besides “#megaclass”. We spent the first three or four meetings trying to understand how the program actually worked and getting familiar with GUI’s, buttons, labels, and stages.

For this project, we struggled more with meeting with the whole group. People had plans made for weekends and fall break, so we could not always meet altogether. Though the meetings were split up with different groups of people, we were still able to understand and put in enough work to finish the project.

Unfinished Features:

- 1) Remove printing "Press R to rotate" when game starts
- 2) Player1 cannot rotate his first ship (length 1).
- 3) Once 2 players placed their ships, player2's board has a highlighted block that shows player2's ship location, it disappears next turn
- 4) When you win by using a Nuke, it prints 9 "you win" messages behind "you used a nuke" message

Features to be Added After Demo:

- 1) Sound effects, animations
- 2) Change color of ships or themes
- 3) Game mode for having another turn after you hit a ship
- 4) Basic battleship game mode that has no extra powerups, plain battleship game

What We Would Have Done Differently

If we could redo this project, we would have split the code we were given into more functions. Since we were given a mega class, it would have been more helpful to have functions that would remove repeat code and improve the “smell” of the code. We also would have tried to restructure the program by splitting things into different classes, but we did not have a strong understanding of how classes work in Java. Other than these changes, we would have liked to meet more often like we did for the first project.

Works Cited

Icon image:

<https://publicdomainvectors.org/en/free-clipart/Battleship-colorful-sketch/69364.html>

Randomize numbers:

<https://stackoverflow.com/questions/363681/how-do-i-generate-random-integers-within-a-specific-range-in-java>

Setting image size:

<https://stackoverflow.com/questions/27894945/how-do-i-resize-an-imageview-image-in-javafx>

Setting button position:

<https://stackoverflow.com/questions/30641187/position-javafx-button-in-a-specific-location?rq=1>