**EECS 448**

**Project 2: Minesweeper Inheritance!**

**Log of all Meetings**

Date: 03/04/2019

Location: Spahr 1326

Members: All were present

Briefings/Notes:

What to add

How to split up project

What is everyone’s schedule:

Requirements

Functionality:

Fix end-game RYAN

Not a win condition to reveal all non-mine spaces

Cheat Mode: TAYLOR

Toggle board

Change mine color by button call

Custom Addition:

Wilhelm home screen MYRA

Clock/timer? MYRA

Fix flag counter (goes negative currently) MEGANA

Possible power up – maybe allow BOGO mine MARCUS

“pretty-up” the UI MEGANA

Scoreboard (maybe a standard competitive button) MARCUS

Something similar to reset button? MARCUS

Next meeting to be on Thursday around 5:00 pm

Try to figure out your plan by then and be able to explain or reassign

Date: 03/06/2019

Location: Spahr 1326

Members: Ryan, Marcus, Taylor, Megana

Briefings/Notes:

How to make more modular

Sent links in our groupchat to plan modularity

Links to plan to use javascript multiple files or other files (requires, grunt, or brunch)

Look at code to see where to split up possible codes

Date: 03/13/2019

Location: Decade Lawrence

Members: Ryan, Marcus, Taylor

Briefings/Notes:

Sat to code together

Merged branches and hashed out any bugs that arose

Date: 03/18/2019

Location: Spahr 1326

Members: Ryan, Marcus, Taylor, Megana, Myra

Briefings/Notes:

Checked last minute bugs

Found errors to fix

Worked on powerup function

Discussed documentation

Date: 03/19/2019

Location: Decade Lawrence

Members: Ryan, Megana

Briefings/Notes:

Coded the powerup function bug together to try to fix problems arising

**Work Delegation Among Members**

The project was split between the group members within the first meeting. Our plan of attack was to determine what features we wanted to add, how we wanted modify the original code (we weren’t a huge fan of the setup of the original code that we inherited) and the timeline we thought would be suited for achieving these goals. Therefore we decided one person should take charge of changing the overall functionality (there was a bug with the ending of the game to restart the game versus continuing the game even after losing the game). Ryan was in charge of this. Taylor felt comfortable creating a cheat toggle mode for this game as per the project’s requirements. Myra wanted to add sound effects to the game as our additional function to the game. Marcus and Megana decided to work on user interface and any other powerups that would be necessary. However, deciding the powerup that we wanted was changed throughout the implementation of this new and improved game.

**Challenges**

The biggest challenge of this game was the fact that the inherited code was not easy to read, nor was it modular. There were originally only 3 classes (board, userinterface, and graphics). However, the functionality of the game was solely focused on one class, that was difficult to control. In order to add features or change any of the code, we had to change the overall structure of the program. Otherwise, the program would crash as a lot of things were hardcoded mathematically. Myra actually went through and changed the overall structure of the JavaScript files and made multiple files to make this program more modular. Once this was taken care of, it was easier to change things to the code – whether it was adding a cheat mode or adding powerups. Additionally, the overall readability of the program was very difficult (upon first glance, we did not like the variable names and documentation to find what we were messing with. For that reason, changing things required a lot of coding and checking on the actual server.) We did not want to change all variable names, but tried to add documentation to make it easier to understand and work through in case another group were to make edits to our program.

**Things not in Final Product**

When we were assigned the project, we had multiple ideas for project power-ups or additions. Most of the features to add were already requirements that we needed to change as we changed structure for modularity. We had ideas to hold a leadership board to test different high scores, but got rid of this idea. We also had an idea to a “press one mine=get one free” type game feature. That means once a mine powerup tile was hit, maybe another mine would automatically be flagged. We got rid of this idea as well. Eventually, we went with the sound effects (Wilhelm Scream) and also an additionally another power up (although we only needed to add one total). We ended up creating both these powerups, and decided to stop adding even more to the product, with the time we have.

**Retrospective on Things to do Differently**

Overall, we believe the project was taken care of effectively. We had an efficient timeline to make sure all our code was running properly and the features to be added were taken care of. The biggest issue was making sure the code could be modified to best suit requirements, when the code originally was not easy to work with. During spring break, our schedules got busy and it was difficult to meet up, but we had a good plan as to how much we wanted to get done and where we should be when we reconvened. Possibly one thing that we could improve on, more than do differently, is to be able to merge without hesitation. We were nervous about merging until at least one other group member was present just in case. The other thing, more so in general rather than specific to our group, is making sure code is readable and extensible. In order to add new features to this code, we had to basically refactor the design, almost entirely, to create an efficient and working model.