Abir Razzak, abir@drexel.edu, CS338:GUI, Project Stage #1

Project Description

1) Minesweeper Ultimate

2) The users could range from casual gamers to Minesweeper professionals. Persona of the casual gamer is to look for a relatively simple game to play, that is not too difficult in terms of gameplay, and also not too complicated in terms of mechanics. These personas would enjoy the simple user interface as well as quality-of-life improvements to the game such as sound effects, and ways to achieve a higher score. On the other end of the spectrum the persona of the Minesweeper advocate is to look for something that reminds them of the classical Minesweeper game they grew up with, but with an interesting twist in terms of mechanic to make the game feel fresh, yet also different. After all, what's the point in playing this game if they can just return to their original Minesweeper game? Adding challenging obstacles, while maintaining the fundamental foundations of the game's look and feel is what this persona is searching for.

3) The system should of course allow the user to play Minesweeper as they would on the original PC's. The users should click tiles that will reveal themselves to show clues, mines, and new power-up's that the game has to offer. The system should include a timer, to please the hardcore Minesweeper persona and should include a counter for the number of remaining bombs left to allow the users to keep track of their progress. There should be a way to exit from the game itself, as well as an about page, should anyone come across this code and would like to credit me.

Abir Razzak, abir@drexel.edu, CS338:GUI, Project Stage #1

Project Plan

Week 1 -> UI sketch work, should be relatively simple since it should emulate the original Minesweeper game. Begin researching on various UI elements that would be needed (icons, sound effects, etc.). Also formulate a definite list of various power-ups to be added and explore how to implement these power-ups via Java code (don't have to start writing any code, but just get a sense of how to implement it when the time comes.)

Week 2 -> UI development work, make a simple UI design of the game itself. This way, tweaks can be simpler to make and can give peer reviewers an actual look at what the UI would look like once it is actually implemented in Swing. Make tweaks according to peer suggestions and if remaining time is available, continue gathering icons, sound effects, etc. If that is done already, then can begin writing logic code if spare time remains.

Week 3 -> Begin writing logic for mines, power-ups, and number tiles. By the end of this week, the program should be able to load a new game where the board is generated properly. Also, begin collecting various UI components like icons and sound effects to be used in the final project.

Week 4 -> Attaching UI and logic. First and foremost, this week should ensure that the logic of tiles gets mapped to their UI counterpart. The tiles should load as un-revealed tiles and when flipped over should show their respective icons. Remaining time should be dedicated to adding sound effects and other quality-of-life improvements to the game.