Criterion E: Evaluation

Meeting the Criteria

- The game is playable, and score based, including a High Score that saves for separate instances of the game locally.
    - This version of the game is very playable and keeps track of scores.
- The player can move around easily and intuitively within the space.
    - The keys to move around are "wasd", a common configuration.
- The player can shoot and aim the blasters of the spaceship easily.
    - To shoot, simply point and click with a mouse
- The asteroids spawn in a way that requires eventual response from the player.
    - It is very common for a generated Asteroid to be on a collision course with the player if they stand still, meaning the player has to move in order to keep a life.
- The game finishes eventually, leaving a game won indication
    - If you make it to the end of the seed, a congratulations screen appears, along with a progression tracker at the bottom of the UI.
- Has basic controls over menus
    - P to Pause, R to Restart, Escape to Exit the game.
- The game is seed able, where in, I can create the same instance of the game if a certain "seed" or String is typed in, and the placement of where asteroids spawn, or placement of entities is the same.
    - In seed select play, simply type in a seed and every time it will spawn the same asteroids in the same locations.
- The game keeps track of lives and bullets shot able by the player.
    - As shown by animations on the screen.
- The game has the ability for multiple game modes
    - Currently there is Play with a seed, and play with music
- The implementation of multiple game modes is documented and simple to do, with the spawning of the Asteroids simple to change.
    - The way the Spawn Controller handles spawning is by reading a list of strings, from a level, so if someone just rewrites how the strings are made, and redirects what path the program is looking at for it's level strings, the game would be changed to the new type of spawning method.

Recommendations for Future Improvements

The game itself was a challenge to make, since dealing with the game loop was a tricky niche that isn't explored in a standard Computer Science class. It was hard making a UI around an Iterative loop, instead of just using a scanner.

I plan to keep working on this game in many ways. One way is to add music based spawning and graphics. From the Music play mode, which currently lets you select a song and play it in the background while playing the game I had bigger plans for. I plan to have the mode decide when to spawn entities and modify them based on the sound and beat detection. That way, if I select a certain kind of song that has lots of strong beats, the game would be spawning asteroids accordingly. This would make the game more challenging and fun. I also plan to have more colors and visualizations of the music during the game play, that way it becomes more inviting to play.

In the end, I learned a lot about developing games, and other coding practizzse's through Java. There were many challenges I had to face with this game. Like learning to think with the Iterative game loop. However, in the end I am glad I chose Java for the language, since it was so supported, and easily expandable.