**Games Engineering Report**

Daniel Seavers, Jack Millar and Maja Heltowska

1. The assignment we were given was to create a game, this game needing to be of a playable state by a stranger coming into it with no experience of how it worked or what it entailed, from this we decided to create a game where the user would control a physics based ball that would move around the stage through controlled bounces, and combat enemies generated in the stage. The inspirations for this came from horde based games like vampire survivors and blockhead and the idea of using simplistic graphics came from games such as diep.io and agar.io.
2. In our GDD we had planned to have more levels available to the player than we did in the end, however due to the complexity of creating the level code repeatedly for new levels this did not end up feeling worth the time and thus was left out. The task of implementing a boss did not feel worth the required effort after making the rest of the game as we felt it would not add much and would cause far more complications. While the current implementation of the game is endless, the player cycles through a repeating set of three rooms. We were not able to add co-operative play to the game and felt that it would take away from the enjoyment the main player would have while playing. In the end we decided to not have the player able to shoot the enemies as between time constraints and difficulty implementing such systems, we in the end opted for bounces to be the sole source of damage. Reducing the number of edges the circle had and eventually turning it into a square, while a cool idea was impractical for the project scope. The speed of the player is inconsistent as is to be expected with a physics object thus at times it moves far slower than a triangle and at times it moves far faster. Squares were not implemented as the enemies ended up taking far too long to implement for it to be worth adding variety. The wave effect is invisible but triggers every time the player lands in the current implementation. Pentagons have also not been implemented at this time. Health and damage values have been changed since the original plan and are still being tuned as I write this report. There is no squangle or other shooting enemy currently. Due to a bug causing enemies to spawn in walls we currently open the exit on its own after 2 seconds, but if this bug is fixed it will be changed to when all enemies die. The player uses arrow keys and the space bar for ball control. Currently enemies do not pass through walls. There is no in game story currently. Currently there are no ui buttons implemented in the game and the title is displayed in a standard font. There is currently only a plain black background, and the platforms are plain white rectangles, the doors are green squares. None of the planned animations have been implemented. The game does not currently use any sound effects. The game runs at over 2000 fps when uncapped so should run on most devices. The game may not be complete to a state where it can be uploaded to itch.
3. The Enemy State Diagram at Appendix 1 shows how the enemy AI works and the states it uses. The enemy starts in its stationary state, then if the player is in range of the enemy, the enemy will enter the seek state and move towards the player and when it gets to closer distance it will enter the flee state which makes the enemy stop at a close distance to the player, it was planned to stop at a close distance and shoot at the player, with more time that would have been implemented.
4. F
5. F
6. Resources used: SFML library and Box2D library

Appendix

Appendix1:

A screenshot of a cell phone

Description automatically generated