Aaron Lopez (301434253)
Brandon Quon (301545786)
Jordan McKenzie (301551839)
Sepehr Borji (301372450)

Overall Plan and Milestones

Milestone 1: Drawing objects to the screen ✓

Milestone 2: Class setup and hierarchy for overriding draw method (Custom sprites) ✓

Milestone 3: Fully functional grid system, complete with instantiating objects ✓

Milestone 4: Player movement and wall collision ✓

Milestone 5: Reading/Writing Map layouts to the game manager✓

Milestone 6: Allow the player to pick up collectables✓

Milestone 7: Enemy collision and movement✓

Milestone 8: Blackhole teleportation and collision✓

Milestone 9: Start screen and GUI✓

Milestone 10: Controls and instructions screens ✓

Milestone 11: Level/difficulty selection screen✓

Milestone 12: Leaderboard screen✓

Milestone 13: All 5 maps implemented with corresponding level game difficulty ✓

Milestone 14: Pause screen with a restart and main menu button✓

Milestone 15: End screen that displays a win or loss and the player's score with a play again and main menu button✓

Milestone 16: Game fully functional and accurately implemented✓

Milestone 17 (Extra): QoL changes and improved code/game functionality✓

What we intend to achieve by the halfway point:
- Full documentation of adjustments and modifications up to that point
- Ideally 25-50% of the milestones completed

Plan to reach these milestones:

Following the plan our team has constructed in phase one, we also plan on reaching each milestone incrementally one step at a time as each sequential milestone may build off another. It is also possible that we will work on up to two milestones at the same time if they are tasks that can be completed with lesser effort (1 person vs the entire group). We also agreed that we need to include JavaDoc comments so that we can each identify changes and understand each other's ideas/methodology in implementing the code.
We will likely to run into roadblocks and issues where we will create new plans to reach later milestones however, at this point in time no further plans are necessary.

Aaron Lopez (301434253)
Brandon Quon (301545786)
Jordan McKenzie (301551839)
Sepehr Borji (301372450)

Our Team's Approach

Our approach is working bottom up and adding one feature at a time. After each feature implementation, we will have bug testing and this will be much more efficient in finding them, as compared to working on many features and hoping they work together. This will ensure the creation of a robust and bug-free product.

Adjustments & Modifications

- Original plan for tile/grid was to be 32x32, now we realize that is too large and have now decided to have map size be small on earlier levels and increasingly large on later levels (Ex. Level 1 - 6x6 and Level 5 - 14x12)

No further adjustments or modifications as of the halfway submission. Changes after halfway submission are below:

- Map sizes revamped to follow that the map size stays relatively the same
- Added the feature of oxygen tanks disappearing within a shorter time frame based on level difficulty
- Changed scoring system: Obtaining an oxygen tank replenishes 25 points of oxygen instead of 75 and when calculating final score, it is now Base Score + Remaining Oxygen - Elapsed Time
- We've had to stray greatly from our original UML diagram; we added a bunch more attributes to certain classes in order to make the implementation easier. Looking back, our original UML diagram felt very simplified compared to our current implementation.
- Instead of utilizing the ControlP5 library for the GUI as originally planned, we made our own GUIManager class to handle a majority of the GUI and state transitions.

Management Process & Division of Roles and Responsibilities

Similar to what we had originally intended in Phase 1, everyone in our group has and will be working on various features, as opposed to splitting everything up. So far we have made a satisfactory effort in following this idea but there are areas of improvement that can be made. Some areas to address are proper time management and fair work distribution. However, our work so far has not yet negatively impacted our group so we will continue to work together on each milestone/segment together instead of directly each completing a specific or individual task.

Aaron Lopez (301434253)
Brandon Quon (301545786)
Jordan McKenzie (301551839)
Sepehr Borji (301372450)

External Libraries Used

- Processing:
    - This library was used mainly for allowing us to draw images to the screen, and detect keyboard/mouse input.

Measures Taken to Enhance Code Quality

- Using JavaDoc commenting so that all users can identify what the code is doing Accurately
- Using version control effectively for code collaboration and history tracking
- Incremental refactoring as the project and code progresses
- Using design principles such as the Singleton in order to make sharing game data between classes much easier (GameManager, GUIManager, etc.)
- Adhering to the 'single dot rule' as much as possible by creating wrapper methods in certain classes (GameManager.UpdateScoreboard, as opposed to GameManager.Scoreoard.Update)

Biggest Challenges During This Phase

Overall, a few of the biggest challenges we've faced (so far, as of the halfway deadline) was becoming adjusted to the processing library, since that was the library we decided to use for drawing sprites. We also struggled with github, making sure everyone could properly clone the repository, and push and pull to it with no issues.

However, one of the main issues we encountered was trying to figure out the wall collisions. We've rewritten how the movement worked several times, and finally got a solution we were happy with.

After the halfway deadline, we had to complete a number of milestones. Some were more difficult than others, but the most notable were

- Making the oxygen deplete at a rate proportional to the level (While maintaining a fair and playable game)
- Correctly applying a score multiplier based on "oxygen overflow" (you cannot obtain more than 100% oxygen and also note that it is no longer a multiplier to prevent large score numbers)
- Allowing the user to choose a level and based on their choice apply the correct variables specific to that level
- Creating multiple black holes and identifying which black hole will be "linked" to its counterpart

Aaron Lopez (301434253)
Brandon Quon (301545786)
Jordan McKenzie (301551839)
Sepehr Borji (301372450)

- Using text files to not only store different map layouts, but save player scores between sessions
- Creating and displaying the GUI and the transitions between the different states
- Keeping our classes organized, attempting to follow the 'single dot rule' as much as possible.