

Technical Documentation

Modification Description:

- Changed the Blocks to UFOs approaching the bottom of the screen
 - This is to allow a more engaging gameplay, as the player is not racing against the UFOs and trying to destroy them.
- Added Powerup that duplicates the ball.
 - Allows the player to engage more by aiming for powerups as well.
 - Allow more strategic gameplay as the player now has multiple balls to utilize.

Structural Changes:

- Classes Added:
 - Levels
 - Allowed the game to change what components are being shown on the screen.
 - All level classes derived from the main Level Class share the same common methods, allowing each level to be easily tailored or modified if need be.
 - The reason for this is to allow the player to change levels, as well as being able to manage what the player sees and engages with based on the game. Furthermore, to allow more control on game assets.
 - Manager Classes
 - Allows more control over a particular class if there are more instances of the class.
 - The reason for this is to have more control over those multiple instances throughout the game and to also represent state calls that Levels can call upon.

State Changes:

- Though the original states of the original classes have not been changed, several of the newly added classes utilize state changes of their own.
 - Levels uses states to check and see if it is running or disabled.
 - Several managers utilize state changes to see if they have any instances of their target Class.
 - If they do not have a single instance, they may change their state from Running to Disabled.

- This will allow levels to call upon each management class and alter their own state depending on the management class's states.

Maintainability

- Due to the amount of classes the project has accumulated, a lot of technical debt was accrued.
 - Examples being how in some cases, in order to change levels, some management classes would have to be called upon from a waterfall of other classes.
- Some Classes did manage to have a more maintainable baseline.
 - Example being the Levels, where regardless, the levels are still able to perform their basic methods, such as clearing components.