

Space Shooter Patch Notes

- Update overview.

- Broad overview of patch
- New:
 - This update contains some massive new additions for the game as a whole. This includes the addition of the enemy and bullet class which will be elaborated more on below. In addition to this the update adds shooting, enemy spawning, upgrades.
- Change:
 - This update also changed the original code extensively. This most notable changes was the conversion of player variables from the main code to the class and the addition of helper functions to make the code more readable and streamlined. This is also paired with some changes to the space shooter games main to have an adjustable screen size and the code to help manage the new components(EnemyClass, BulletClass).
- Removed:
 - This update didn't remove anything Horray!

- Game update

- Updates made to SpaceShooter.py
- Helper Functions/features:
 - There are currently no helper functions in SpaceShooter.py however they might be needed in later updates to make the code more readable and to add logic segmentation to handle an issue when the player moves and shoots and stops one of those actions.
- Code changes:
 - The main changes to the SpaceShooter.py class is the addition of two more keypresses and the base logic to handle the enemy feature and the bullet feature. For the key presses we added space and escape which handles shooting and a fast exit respectively. The base logic for the bullet is the addition of spawning the bullet when proper key is pressed and handling the bullet instances moving them forward and detecting for collisions with either the player or the enemy, The detection feature can be changed as needed to allow friendly fire or to remove friendly fire. More changes on the bullets can be found later in the document. For the enemy logic this update just handles spawning and a regular consistent maintenance on the enemies individually, This will need to be changed later to have the enemy act in a coherent way to make the game fun. More updates on the enemy features can also be found later in this document.

- Player class

- Updates made to PlayerClass.py
- Helper Functions/features:

- The player class updates include the addition of upgrades, referred to as augments, the addition of more player variables
 - Features:
 -
 - Functions:
 - The functions added in this update are get functions for the model, cooldown, last shoot, hitbox, score, and upgrade. This also includes a set function last shoot and change function for both augments. There are also helper functions which include a player Update which manages additional features in the player class which include: a purchase function which will aid in buying augments, a move function, a bullet function called pew which creates a new bullet, and a upgrade function which manages upgrades.
 - Code changes:
 - Beyond the addition of the above functions there is new variables and logic in the INIT function to help set the play function adequately.
 - What's next:
 - some additional helper functions such as for model instantiation would be nice depending on if we set upgrades to also be skins for the player.
- Enemy class
 - Updates made to EnemyClass.py
 - Helper Functions/features:
 - The enemy class is used to create an enemy class instance that can be personalized later for different enemy types this includes many of the features found in the player class however it has less features overall
 - Features:
 - the features include a means to get and set data for the enemy class along side useful features to help streamline managing the enemy class in the game
 - Functions:
 - The functions include get functions such as: model, position, cooldown, last shoot, and hitbox. It also includes a set function which is for last shoot. It has some helper functions such as move and valid move, and a bullet function pew.
 - Code changes:
 - Since the code is new all the above mentioned components are new.
 - What's next:
 - The class could stand to benefit from the addition of an additional function to help determine enemy action depending on player, alongside a feature similar to augments in the player class to help differentiate for different enemy's.

- **Bullet class**

- Updates made to BulletClass.py
- Helper Functions/features:
 - This class is used to create a bullet instance that is customizable depending on who shot it.
 - Features:
 - This class includes features to help manage a bullet as a whole to facilitate its use in the game.
 - Functions:
 - This class includes some get functions including model, position, hitbox, and origin. It has two helper functions offscreen and move.
- Code changes:
 - This is a new class so all code within is new
- What's next:
 - The only possible change foreseeable in the future is to move the movement to be a variable to help create potential for different augments.

- **Graphic changes**

- There was a lot of change to the graphics of the game. This includes a change to 128-bit models which can hold more detail and are simple and easy to manage. The models that are added are found in the 128-bit folder which needs some additional management before the game launch. These models are two for enemies, one for the player, and two models for enemy and player bullets. In the future we may want to consider adding more player models and use those as upgrades alongside a better background for the game as a gif if pygames can handle it.

- **Conclusion**

- This concludes this patch, in summary there was an addition of two classes which is used in conjunction to additional code to help add more functions to the game and there were some code changes to help add unique features to the game. In the future we would have to focus on finishing of the enemy class functions and converting the game to a mobile game, after this we may want to add additional graphics and features to the game to help make it more personalized but that is if time permits.