ProgramTest (Shooting)



Please improve the attached Unity project "ProgramTestShooting".

This project was created with Unity 2020.3.9.

There is only one scene included in the project, SampleScene.

■ Description of SampleScene

This section describes the main objects placed in SampleScene.

|  |  |
| --- | --- |
| ◇MainCamera  There is only one camera. In this project, the camera is set to "Orthographic".  ◇GameFlow  This is where the main code of this project is located.  Two transitions, "Title screen" and "Stage screen" are set up here.  ◇UI  All the UI displayed in this project is placed under this object.  The UI is set to "ScreenSpace-Camera”.  ◇StageObject  This is used as a location for temporary objects created during the stage. |  |

■ Explanation of the game flow

* When you launch this project, it will start with the title screen (TitleLoop.cs).
* Press the "Space" key on the title screen to start the stage (StageLoop.cs)
* During the stage, you control your ship (Player.cs) to defeat the enemies (Enemy.cs) coming at you from the top of the stage.
* You can fire bullets (PlayerBullet.cs) by pressing the "Z" key.
* If you hit an enemy with a bullet, you will defeat the enemy and get a point.
* Enemies will appear periodically from Spawners (EnemySpawner.cs) placed throughout the stage.
* Press the "ESC" key to exit the stage and return to the title screen.

■ Test Content

This project is "less than a game" at the moment, but we want you to improve some aspects of it to make it more interesting.

There are many ways to do this, but here are some basic fixes that we want you to consider for the test.

* The way the enemies appear is monotonous and uninteresting. Improve the enemy's movement, actions and spawning so they are more dynamic, fun and pleasing to look at and play against.
* The current movement and behavior of the player is simple and uninteresting. Please improve it. You may add some new features for this purpose.
* Please add some visual and audio effects for bullets, enemy collisions, etc.
* Please add a playable end condition to this stage. In other words, you will need to add a rule that the game ends when the player fails in some way, or when the game ends based on a time limit.

■ Improvements to current code

- Added generic object pooling (with Generics and interfaces) (based on unitys object pooling in unity version 2021 > and higher);

- Base spawner class with an object pool

- Object pool bullets

- Object pool enemies

- Added generic layers for the spawners of enemies and bulllets that make use of the object pooling

- Made most public variables [serializedfields] or private

- Converted some of the object data to scriptable objects so we can easily create new settings for the game (like enemies, bullets and so on)

- Seperated some functionality into different classes like the view logic (main screen and score into its own classes)

- Added interfaces for testability and implementation templates

- Added generics for extenability and reuseabillity of code

- Added messaging

■ New mechanics

The way the enemies appear is monotonous and uninteresting. Improve the enemy's movement, actions and spawning so they are more dynamic, fun and pleasing to look at and play against.

* When enemies spawn give change of different setting [x]
* Change on different movement [x]
* Spawn within radius [x]
* Start from outside the screen and move in before [x]
* Increasing difficulty over timer by the use of SO as game settings [x]

The current movement and behavior of the player is simple and uninteresting. Please improve it. You may add some new features for this purpose.

* Added rotation based on movement direction [x]
* Added the option to keep the player within the screen (playing field of the game) [x]
* Added barrel roll (dodge) ability Q/E [x]

Please add some visual and audio effects for bullets, enemy collisions, etc.

* VFX effects [x]
* Player model [x]
* Enemy model [x]
* Bullet effect/model [x]
* UI [x]
* Background [x]
* Add hit feedback [x]
* Audio [x]

Please add a playable end condition to this stage. In other words, you will need to add a rule that the game ends when the player fails in some way, or when the game ends based on a time limit.

* Player loses the players lives reaches zero [x]
* Player wins is he survives a certain amount of time [x];

Other mechanics added:

* Save score locally [x];
* Added UI elements [x]
* Added Life bar [x]