

Game Programming

Lecture VIII Shield

Izzet Fatih Senturk

Create the Shield Prefab

- Drag the first shield sprite from powerups folder to the Hierarchy window
- Change the sorting layer to foreground
- Rename it: "Shield_Powerup"
- Set scale to 0.5 0.5 0.5
- Add a Circle Collider 2D
 - Set isTrigger true
- Add a Rigid Body 2D
 - Set gravity scale to 0
- Add "Powerup_sc" component
 - Set PowerupID = 2
- Make it a prefab under "Powerup" folder
 - Delete it from the Hierarchy

Animate the Shield Powerup

- Drag the Shield Powerup prefab to Hierarchy Window
- Make sure the Shield Powerup object is selected
- Open Window -> Animation -> Animation (or add tab -> Animation)
- Click Create and go to Animations folder
 - Save as Shield_Powerup_anim
- Go to record mode
 - Drag and drop Shield sprites from Powerups folder to Animation Window
 - Disable record mode
- Override changes to prefab
- Delete the Shield Powerup object from the Hierarch Window

Update Spawn Manager

- Currently, Spawn Manager script has an array of powerup prefabs with a size of 2
 - Add Shield Powerup as the third element of the array
- Update SpawnPowerupRoutine in the Spawn Manager script to consider 3 elements while selecting the next powerup randomly

```
IEnumerator SpawnPowerupRoutine()
{
    while (_stopSpawning == false)
    {
        Vector3 position = new Vector3(Random.Range(-8f, 8f), 7, 0);
        int randomPowerUp = Random.Range(0, 3);
        Instantiate(_powerupPrefabs[randomPowerUp], position, Quaternion.identity);
        yield return new WaitForSeconds(Random.Range(3,8));
    }
}
```

Shield Powerup Behavior

- Player gets damaged in the Damage function of the Player script
- When the Shield powerup is collected
 - Do not decrease the lives
 - Deactivate the shield
- Basically, Player is not damaged for one hit when the shield is active

Shield Powerup Implementation

- Define a Boolean variable to check if the shield powerup is activated

```
private bool _isTripleShotActive = false;  
2 references  
private bool _isSpeedBoostActive = false;  
0 references  
private bool _isShieldPowerupActive = false;
```

- Define a public function to activate shield powerup
 - Set the Boolean variable true

```
public void ShieldPowerupActive()  
{  
    _isShieldPowerupActive = true;  
}
```

Shield Powerup Implementation

- Update the Damage function
 - Set the Boolean variable false to deactivate shield
 - Do nothing and return

```
public void Damage()
{
    if (_isShieldPowerupActive == true)
    {
        _isShieldPowerupActive = false;
        return;
    }
    _lives--;
```

Shield Powerup Implementation

- Update "Powerup_sc" to activate the shield powerup

```
if (player != null)
{
    switch(powerupID)
    {
        case 0:
            player.TripleShotActive();
            break;
        case 1:
            player.SpeedBoostActive();
            break;
        case 2:
            player.ShieldPowerupActive();
            break;
        default:
            Debug.Log("Default Value");
            break;
    }
}
```


Shield Visualizer Behavior

- Use “Player Shield” to denote that Shield Powerup is active
 - When the shield powerup is collected, show “Player Shield”
 - When the shield powerup is deactivated, hide “Player Shield”
- Drag and drop the first “Player Shield” to Hierarchy window
 - Rename it as “Shield”
- “Player Shield” needs to start in the off position
 - Uncheck the enable field for “Player Shield”
- “Player Shield” must move with the “Player”
 - Set the “Player Shield” as the child of the “Player”

Shield Visualizer Implementation

- Define a variable to store the shield visualizer game object
 - In the Inspector window assign the shield visualizer object

```
[SerializeField]  
0 references  
private GameObject _shieldVisualizer;
```

- Enable shield visualizer when the shield powerup is activated

```
public void ShieldPowerupActive()  
{  
    _isShieldPowerupActive = true;  
    _shieldVisualizer.SetActive(true);  
}
```

Shield Visualizer Implementation

- Update Damage function to disable shield visualizer

```
if (_isShieldPowerupActive == true)
{
    _isShieldPowerupActive = false;
    _shieldVisualizer.SetActive(false);
    return;
}
_lives--;
```

Shield Visualizer Animation

- Select the visualizer game object that we want to animate
- Go to animation window
- Create a new animation
 - Under Animations folder: "Shield_Visualizer_anim"
- Start record mode
- Drag and drop shield visualizer sprites in order
- Stop record mode