

# Documentation for code

The project consists of three script files that together comprise the code for the project.

1. FaceSwap.cs
2. ObstacleSpawner.cs
3. PlayerScore.cs

## FaceSwap.cs

```
// Switch face material for all faces to next material in the list of materials.
public void SwitchFaceMaterial()
{
    foreach(ARFace face in faceManager.trackables)
    {
        face.GetComponent<MeshRenderer>().material = faceMaterials[faceMaterialIndex];
    }
    faceMaterialIndex++;
    // If at the end of list, go back to first element.
    if(faceMaterialIndex == faceMaterials.Count)
    {
        faceMaterialIndex = 0;
    }
}
```

FaceSwap.cs is used to change the AR face filter of the player. It is the script that is part of the face detection and AR Filter module. It uses the ARFaceManager from XRCore to get the detected faces. For all faces, it swaps the face material using the list of materials available. The script is executed, every time the swap button in the game UI is pressed.

## ObstacleSpawner.cs

```
// Function to spawn randomly
IEnumerator SpawnObject(float time)
{
    yield return new WaitForSecondsRealtime(time); // For now in level 1, the time of spawning is taken as input.

    Vector3 temp = transform.position;
    temp.x = Random.Range(x1, x2); // Randomly select boundary between the bounds.
    Instantiate(obj[Random.Range(0, obj.Length)], temp, Quaternion.identity); // The objects to be spawned can be customized.
    StartCoroutine(SpawnObject(Random.Range(1f, 2f))); // To loop this code.
}
```

ObstacleSpawner.cs is the script responsible for spawning 3D falling objects onto the screen. The script runs throughout the game until the game over condition is satisfied, and then it restarts. It randomly selects which object to spawn and randomly spawns it at top of the screen.

## PlayerScore.cs

```
void OnTriggerEnter2D(Collider2D target)
{
    if(target.tag == "fireball") // If collision is with fireball.
    {
        transform.position = new Vector2(0, 100);
        target.gameObject.SetActive(false);
        StartCoroutine(RestartGame()); // Call function to restart the game.
    }
    if(target.tag == "coin") // If collision is with coin.
    {
        target.gameObject.SetActive(false);
        score++; // Increment score.
        ScoreText.text = score.ToString();
    }
}
```

PlayerScore.cs is the script responsible for the reward system of the game. It uses the Collider objects to detect collision and depending on the kind of collision, increments the score or causes the Game Over, in which case the game restarts.