

```
1 using CustomProject.GameObjects;
2 using SplashKitSDK;
3
4 namespace CustomProject.StrategyDesign
5 {
6     public class AdvancedCollect : ICollectStrategy
7     {
8         public void Collect(Player player)
9         {
10             if (!player.IsDead)
11             {
12                 List<CollectibleGold> collectedGolds = new List<CollectibleGold>();
13                 List<Sword> collectedSwords = new List<Sword>();
14                 List<SpeedPotion> collectedSpeedPotion = new List<SpeedPotion>();
15                 List<JumpPotion> collectedJumpPotion = new List<JumpPotion>();
16
17                 foreach (CollectibleGold gold in player.GameMap.Golds)
18                 {
19                     if (SplashKit.BitmapCollision(player.Image, player.X,
20 player.Y, gold.Image, gold.X, gold.Y))
21                     {
22                         collectedGolds.Add(gold);
23                     }
24                 }
25
26                 foreach (Sword sword in player.GameMap.Swords)
27                 {
28                     if (SplashKit.BitmapCollision(player.Image, player.X,
29 player.Y, sword.Image, sword.X, sword.Y))
30                     {
31                         collectedSwords.Add(sword);
32                     }
33                 }
34
35                 foreach (SpeedPotion potion in player.GameMap.SpeedPotion)
36                 {
37                     if (SplashKit.BitmapCollision(player.Image, player.X,
38 player.Y, potion.Image, potion.X, potion.Y))
39                     {
40                         collectedSpeedPotion.Add(potion);
41                     }
42                 }
43
44                 foreach (JumpPotion potion in player.GameMap.JumpPotion)
45                 {
46                     if (SplashKit.BitmapCollision(player.Image, player.X,
```

```
        player.Y, potion.Image, potion.X, potion.Y))
    {
        collectedJumpPotion.Add(potion);
    }
}

foreach (SpeedPotion potion in collectedSpeedPotion)
{
    player.GameMap.SpeedPotion.Remove(potion);
    player.IncreaseSpeed = true;
}

foreach (JumpPotion potion in collectedJumpPotion)
{
    player.GameMap.JumpPotion.Remove(potion);
    player.IncreaseJump = true;
}

foreach (Sword sword in collectedSwords)
{
    player.GameMap.Swords.Remove(sword);
    player.CollecteSwords += 1;
    player.Notify("SwordCollected");
}

foreach (CollectibleGold gold in collectedGolds)
{
    player.GameMap.Golds.Remove(gold);
    player.CollecteGolds += 1;
    player.Notify("GoldCollected");
}
}
}
}
```