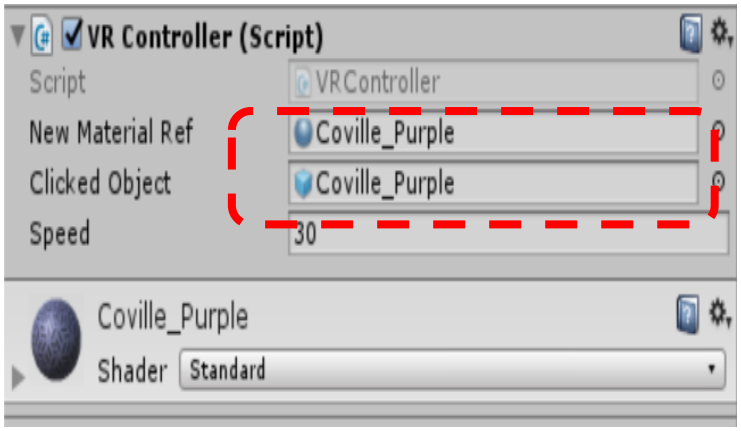
VR App Manual using Unity

Introduction

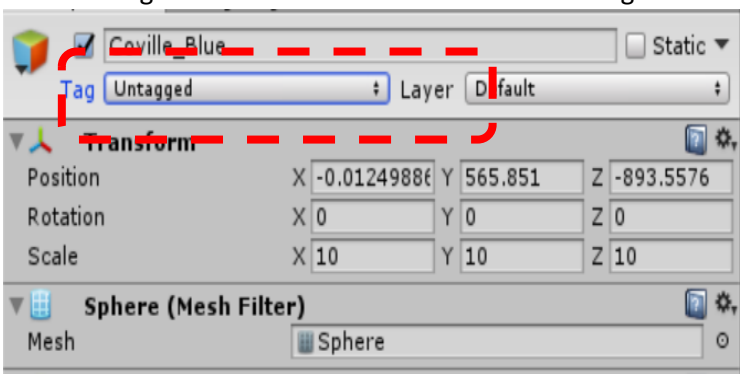
This document will guide you on how to add new patterns and new colours to VR app.

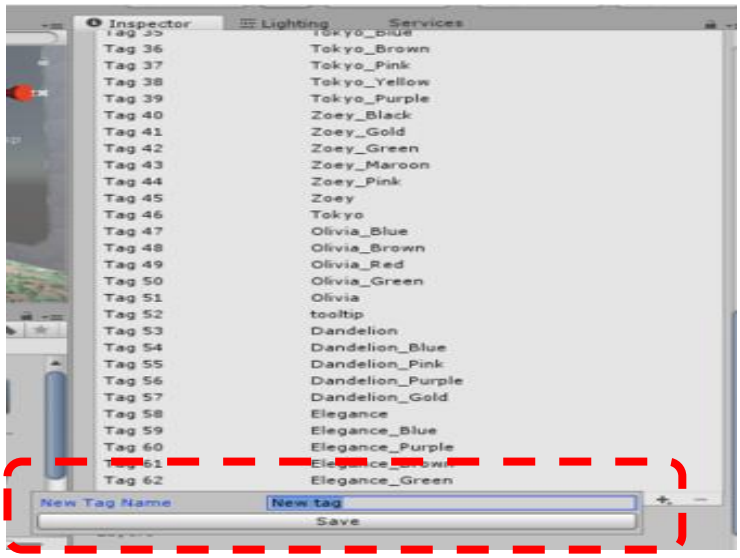
Add new pattern and new colour

1. **Prepare pattern prefab**
    a. Import image into Patterns folder, create a new material and attach the image to the material
    b. Create a sphere gameObject named with pattern/colour name with scale of 10*10*10
    c. Attach the created material and VRController script to the gameObject
    d. Drag the corresponding material to “New Material Ref” and the gameObject to “Clicked Object”



    e. Create a new Tag with gameObject name and assign the newly created tag
        i. Click the Tag and scroll to the bottom to add new tag





f. Create a prefab and attach the gameObject to the prefab

2. Modify the scripts

a. VRController

Add new pattern name into patterns array

```
public static string[][] patterns = new string[][]
{
  //  new string[] {"Mimosa", "Flower", "Coconut"},
 //   new string[] {"Forest", "Lotus", "Bird"}
     new string[] {"Coville", "Olivia", "Tokyo"},
     new string[] { "Dandelion", "Zoey", "Elegance" },
};
```

Create a new colour array for the new pattern

```
public static string[][] eleganceColors = new string[][]
{
   new string[] { "Elegance_Blue", "Elegance_Brown", "Elegance_Green"},
   new string[] { "Elegance_Purple"},

};
```

Modify OnGazeTrigger() to instantiate colours once the pattern is clicked

```
if (ClickedObject.name == "Mimosa" || ClickedObject.name == "Mimosa(Clone)")
{

    createPatterns(mimosaColors[colorId]); // create colours



}
```

3. Modify MorePatternsToLeft and MorePatternsToRight to scroll to more colours

Scroll colour selection to the left

```
if (VRController.Clicked.Contains("Mimosa"))
{
    print("colorId" + VRController.colorId);
    if (VRController.colorId > 0)
    {

        string[] preColors = VRController.mimosaColors[VRController.colorId];
        for (int i = 0; i < preColors.Length; i++)
        {
            VRController.DestoryPattern(preColors[i]);

        }
        VRController.colorId -= 1;
        //print(patternId);
        string[] nexColors = VRController.mimosaColors[VRController.colorId];

        VRController.createPatterns(nexColors);
    }


}
```

Scroll colour selection to the right

```
if (VRController.Clicked.Contains("Mimosa"))
{
    print("colorId" + VRController.colorId);

    if (VRController.colorId < (VRController.mimosaColors.Length - 1))
    {

        string[] preColors = VRController.mimosaColors[VRController.colorId];
        for (int i = 0; i < preColors.Length; i++)
        {
            //print(preColors[i]);
            VRController.DestoryPattern(preColors[i]);

        }
        VRController.colorId += 1;
        //print(patternId);
        string[] nexColors = VRController.mimosaColors[VRController.colorId];

        VRController.createPatterns(nexColors);
    }


}
```

4. **Modify BackToPatterns to destroy colour gameObjects when back to pattern menu**

```
for (int i = 0; i < GameObjects.Length; i++)
{
    if (GameObjects[i].name.Contains("Coville") || GameObjects[i].name.Contains("Sofia") || GameObjects[i].name.Contains("Tokyo") || G
        Destroy(GameObjects[i]);
}



Instantiate(Resources.Load("SelectPattern"), new Vector3(-15, 585, -882), Quaternion.Euler(3, -56, 1));
Instantiate(Resources.Load("Reset"), new Vector3(-13, 561, -869), Quaternion.Euler(0, -60, 0));
string[] patterns = VRController.patterns[VRController.patternId];
```

5. **Modify ClickBed to destroy new patterns when resetting the scene**

```
public void resetPatterns()
{
    clickedBed = false;
    GameObject[] gameObjects = UnityEngine.Object.FindObjectsOfType<GameObject>();

    for (var i = 0; i < gameObjects.Length; i++)
    {
        if (gameObjects[i].name.Contains("Coville") || gameObjects[i].name.Contains("Sofia") || gameObjects[i].name.Contains("Tokyo"))
        {
            Destroy(gameObjects[i]);
        }
    }

    Instantiate(Resources.Load("Tooltip"), new Vector3(-2, 567, -897), Quaternion.Euler(0, -60, 0));

}
```