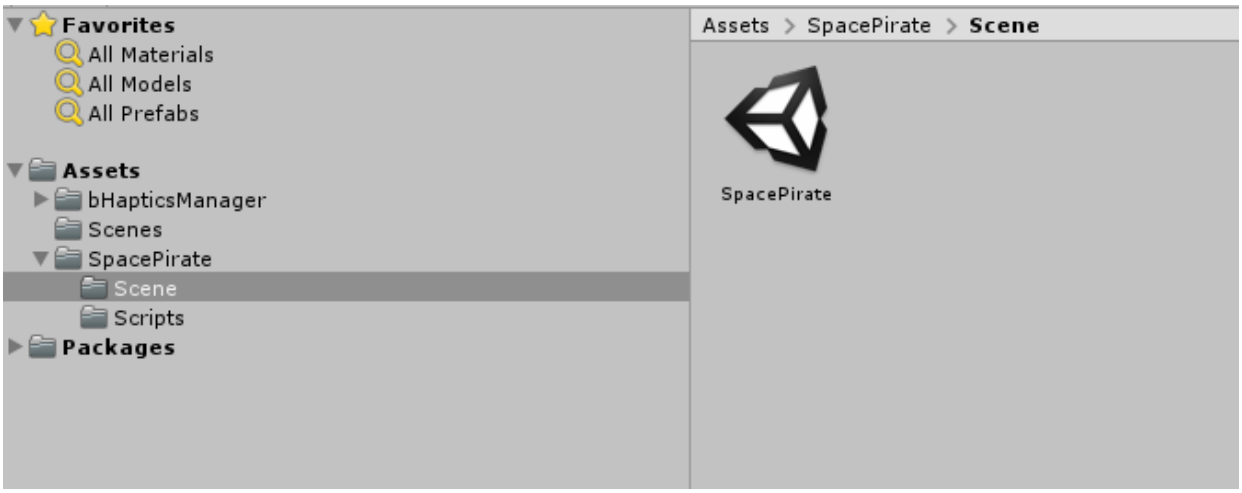
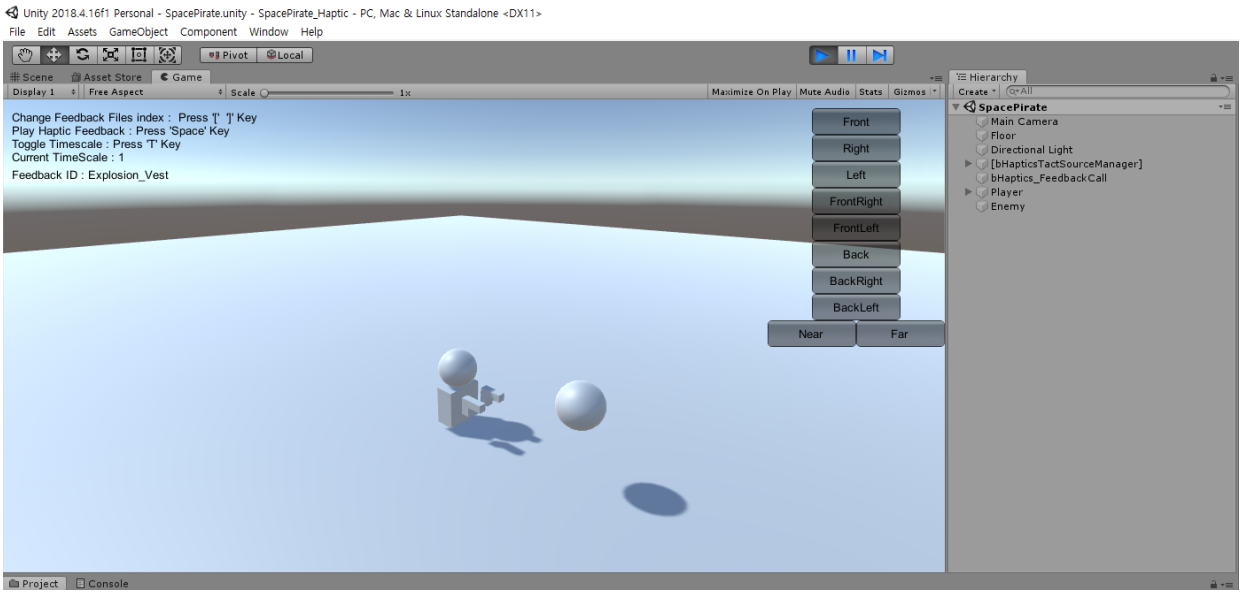
SpacePirate X bHaptics

Testing

For a starting point, you can check the SpacePirate scene.



From that scene, you can check most of the functionality.



You can move enemy position press a Position button right side on screen.

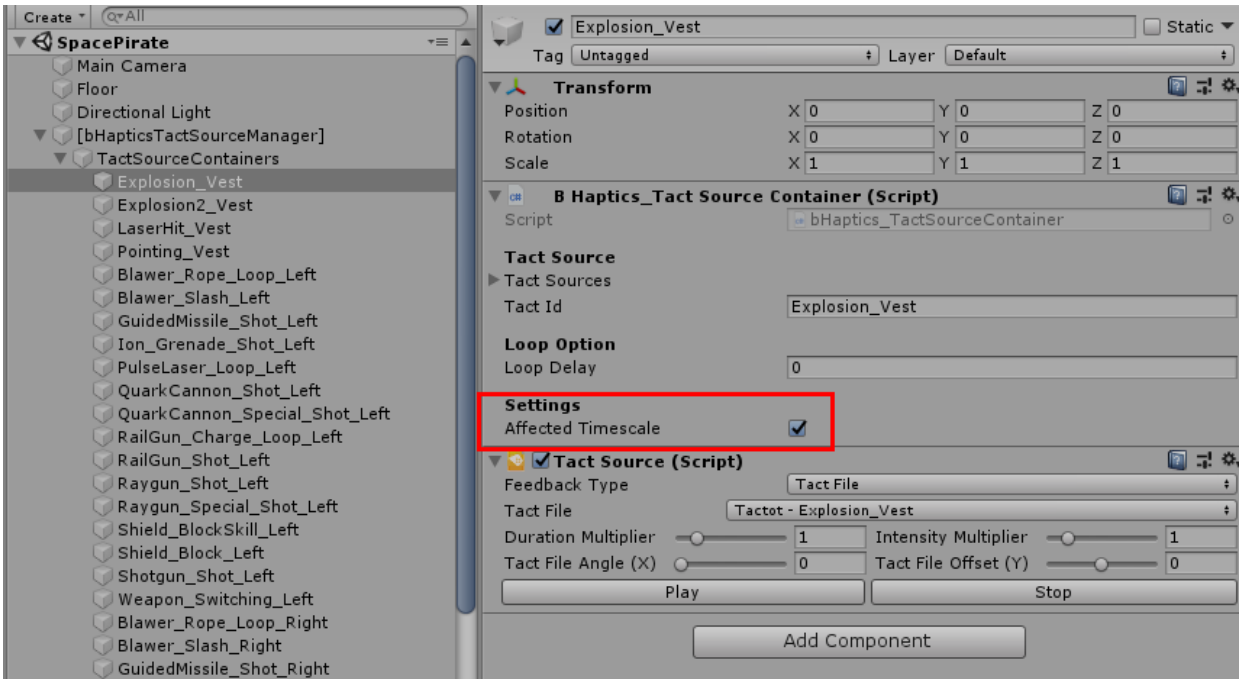
Change Haptic feedback Press “[” “]” key
Play Haptic feedback Press “Space bar” key
Toggle Time scale Press “T” key

Setting up in your scene

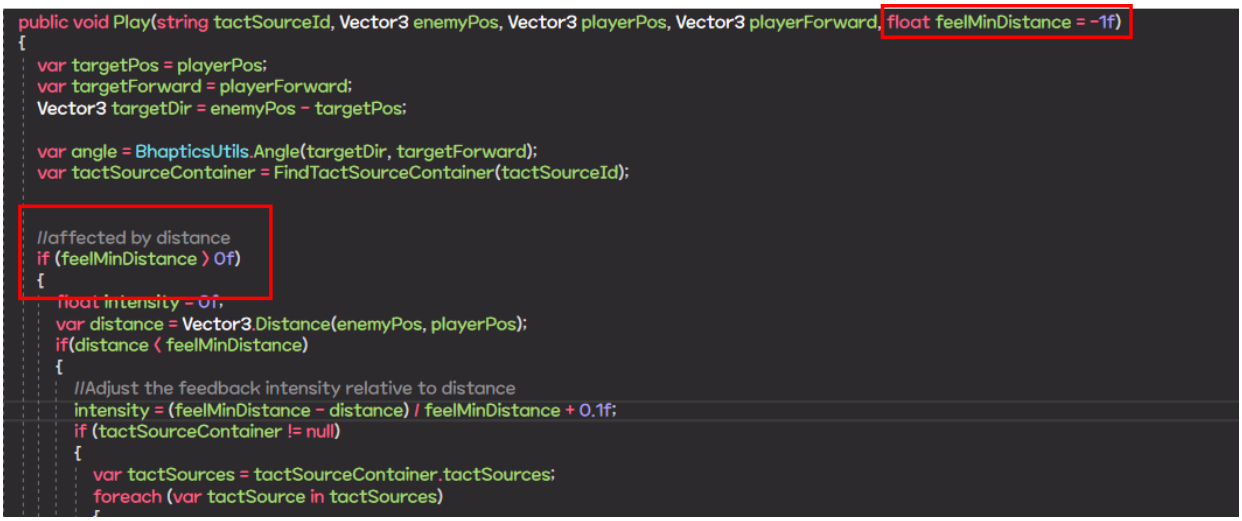
1. Add [bHapticsTactSourceManager].prefab on your scene or instantiate it in your game manager object
   (If you have to change the scene often, Don't destroy object would be better).

2. Then just call the method like this
   bHaptics_TactSourceManager.instance.Play("@tactIdString");
   bHaptics_TactSourceManager.instance.PlayLoop("@tactIdString");

3. If you want to have Haptic feedback affected by Time Scale, set true affectedTimescale variable in bHaptics_TactSourceContainer Component



Importantly, by using your player transform and explosion or bullet hit position, you can make directional feedback as below.
bHaptics_TactSourceManager.instance.Play("@tactIdString", explosion or bullet position(Vector3), player position(Vector3), player forward(Vector3), feelMinDistance(default parameter float));
(If you need using feedback affected by distance (like explosion), put in the feelMinDistance value)



```
public void Play(string tactSourceId, Vector3 enemyPos, Vector3 playerPos, Vector3 playerForward, float feelMinDistance = -1f)
{
    var targetPos = playerPos;
    var targetForward = playerForward;
    Vector3 targetDir = enemyPos - targetPos;

    var angle = BhapticsUtils.Angle(targetDir, targetForward);
    var tactSourceContainer = FindTactSourceContainer(tactSourceId);

    //affected by distance
    if (feelMinDistance > 0f)
    {
        float intensity = 0f;
        var distance = Vector3.Distance(enemyPos, playerPos);
        if(distance < feelMinDistance)
        {
            //Adjust the feedback intensity relative to distance
            intensity = (feelMinDistance - distance) / feelMinDistance + 0.1f;
            if (tactSourceContainer != null)
            {
                var tactSources = tactSourceContainer.tactSources;
                foreach (var tactSource in tactSources)
```

You will know if you check bHaptics_FeedbackCall.cs

Haptic Pattern Prepared

- Vest
    - Explosion_Vest //Has directional
    - Explosion2_Vest //Has directional
    - LaserHit_Vest //Has directional
    - Pointing_Vest //Has directional
- Left Arm
    - Blawer_Rope_Loop_Left
    - Blawer_Slash_Left
    - GuidedMissile_Shot_Left
    - IonGrenade_Shot_Left
    - PulseLaser_Loop_Left
    - QuarkCannon_Shot_Left
    - QuarkCannon_Special_Shot_Left
    - RailGun_Charge_Loop_Left
    - RailGun_Shot_Left
    - RayGun_Shot_Left
    - RayGun_Special_Shot_Left
    - Shield_BlockSkill_Left
    - Shield_Block_Left
    - Shotgun_Shot_Left

- Right Arm
    - Blawer_Rope_Loop_Right
    - Blawer_Slash_Right
    - GuidedMissile_Shot_Right
    - IonGrenade_Shot_Right
    - PulseLaser_Loop_Right
    - QuarkCannon_Shot_Right
    - QuarkCannon_Special_Shot_Right
    - RailGun_Charge_Loop_Right
    - RailGun_Shot_Right
    - RayGun_Shot_Right
    - RayGun_Special_Shot_Right
    - Shield_BlockSkill_Right
    - Shield_Block_Right
    - Shotgun_Shot_Right