Sleepless Underwater Effects - Version 1.1

NOTE: Demo scene worked with bloom and HDR. For corrected bloom with "Forward" mode disable multisampling, or use "Deferred" mode. "Edit->Project Settings->Quality->Multisampling"
Scripts will work perfectly on Mobile / PC / Consoles and while this scene was created with Unity 5.5.0f3, it will work on all versions of Unity5 by setting it up in the same way.

Standard Assets to Import

For compatibility with as many Unity versions as possible, those both previous, current and in future, we have removed the Standard Assets from the Demo Scene, and you will need to Re-Import them in order to run the scene correctly.

1. Import Standard Asset Package - Characters - Can be any type, but we used a 1st Person prefab for our Demo Scene.
2. Import Standard Asset Package - Environment - To replace the Water. We used a scaled up Water4 Prefab
3. Import Standard Asset Package - Effects - For the BlobLightProjector and some of the Image Effects we used on the Main Camera.

We have included Screenshots of all Assets and Settings, in this document, to assist you in modifying the Standard Assets or to help you apply these changes to your own assets.

Changes to Standard Asset Defaults

1. In the demo scene, I have decoupled the Main Camera from the FPSController and put it in under a CameraAttachment GameObject. This allows you to move the CameraAttachment up slightly for a more realistic head placement. I left the Y as defaults, but just letting you know why I made this change.

2. Added a Crouch input key to the Input Settings. Crouchheight script is used to simulate what our bodies naturally do when we enter the water. The "c" key is used to crouch in first person, but it is also used to sink when under water.

3. FirstPersonController. Crouchheight requires that the FirstPersonController script have a boolean variable called m_IsSwimming and a method which returns this boolean called CanSwim() declared. I have included screenshots below of all changes to the current version of the FirstPersonController Class to assist you in making these changes. This will also help understand where and why we have made these changes in case you are using your own Character Controller.

4. I stretched the scale of X and Z on the Water4 object. This can cause weird wave and reflection effects sometimes, especially on a larger scale again. If this happens, you will need to modify the Water4 Material settings and reduce the Wave settings. Scaling causes the waves to also scale strangely, reducing these settings will correct this issue. Also make sure you change the layer for the water prefab to "Water"

5. You will need to set the LightProjector material on the BlobLightProjector Prefab once the Standard Assets have been imported

Relevant Changes to the FirstPersonController Class

```
namespace UnityStandardAssets.Characters.FirstPerson
{
    [RequireComponent(typeof (CharacterController))]
    [RequireComponent(typeof (AudioSource))]
    2 references
    public class FirstPersonController : MonoBehaviour
    {
        [SerializeField] public bool m_IsSwimming;
        [SerializeField] private bool m_IsWalking;
        [SerializeField] private float m_WalkSpeed;
        [SerializeField] private float m_RunSpeed;
        [SerializeField] [Range(0f, 1f)] private float m_RunstepLenghten;
        [SerializeField] private float m_JumpSpeed;
```

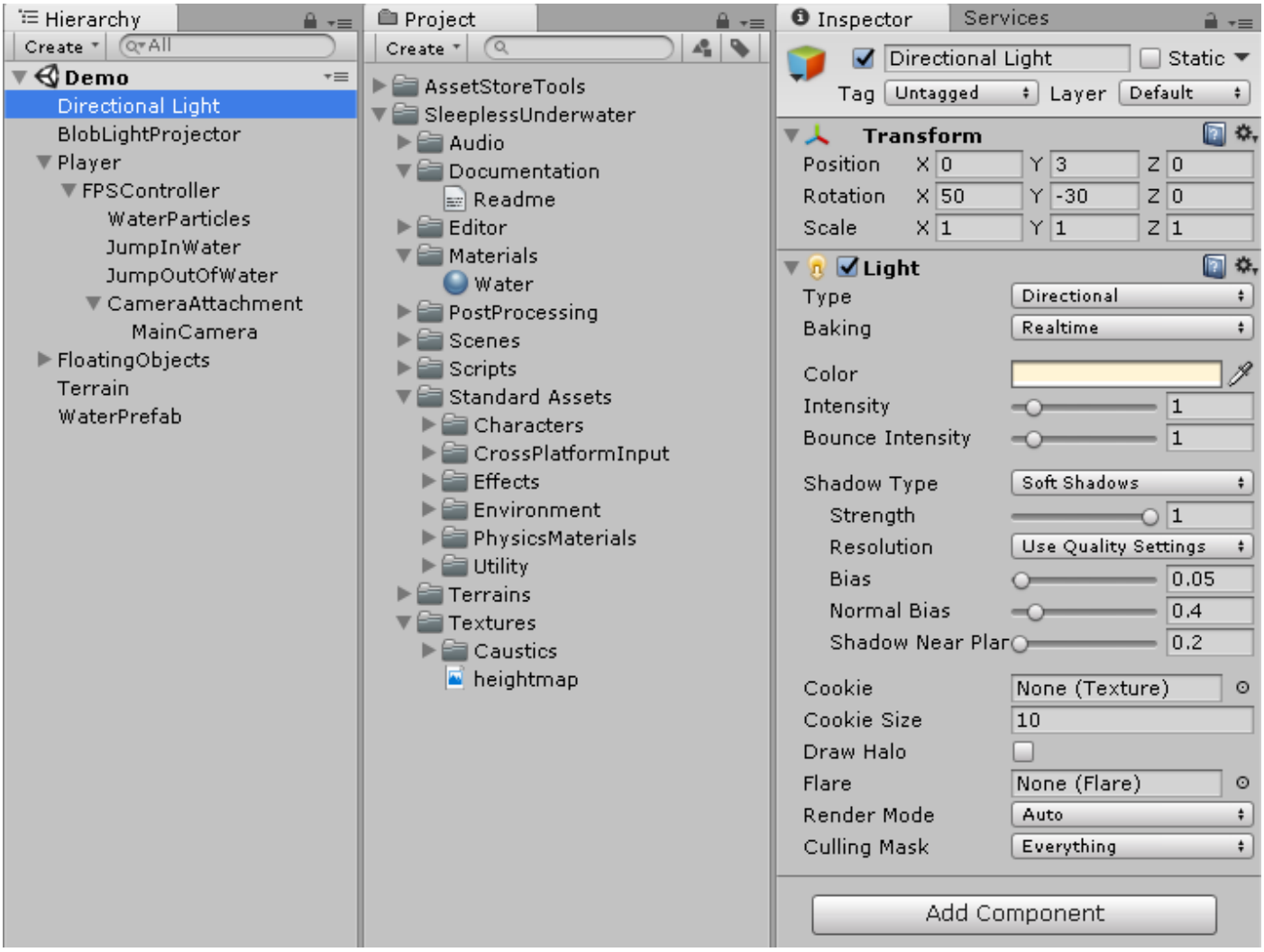
```
    // Use this for initialization
    0 references
    private void Start()
    {
        m_IsSwimming = false;
        m_CharacterController = GetComponent<CharacterController>();
        m_Camera = Camera.main;
        m_OriginalCameraPosition = m_Camera.transform.localPosition;
        m_FovKick.Setup(m_Camera);
        m_HeadBob.Setup(m_Camera, m_StepInterval);
        m_StepCycle = 0f;
        m_NextStep = m_StepCycle/2f;
        m_Jumping = false;
        m_AudioSource = GetComponent<AudioSource>();
        m_MouseLook.Init(transform , m_Camera.transform);
    }


    // Update is called once per frame
    0 references
    private void Update()
    {
        RotateView();
        // the jump state needs to read here to make sure it is not missed
        if (!m_Jump)
        {
            m_Jump = CrossPlatformInputManager.GetButtonDown("Jump");
        }

        if (!m_PreviouslyGrounded && m_CharacterController.isGrounded && m_IsSwimming == false) //Check to ensure not swimming
        {
            StartCoroutine(m_JumpBob.DoBobCycle());
            PlayLandingSound();
            m_MoveDir.y = 0f;
            m_Jumping = false;
        }
```

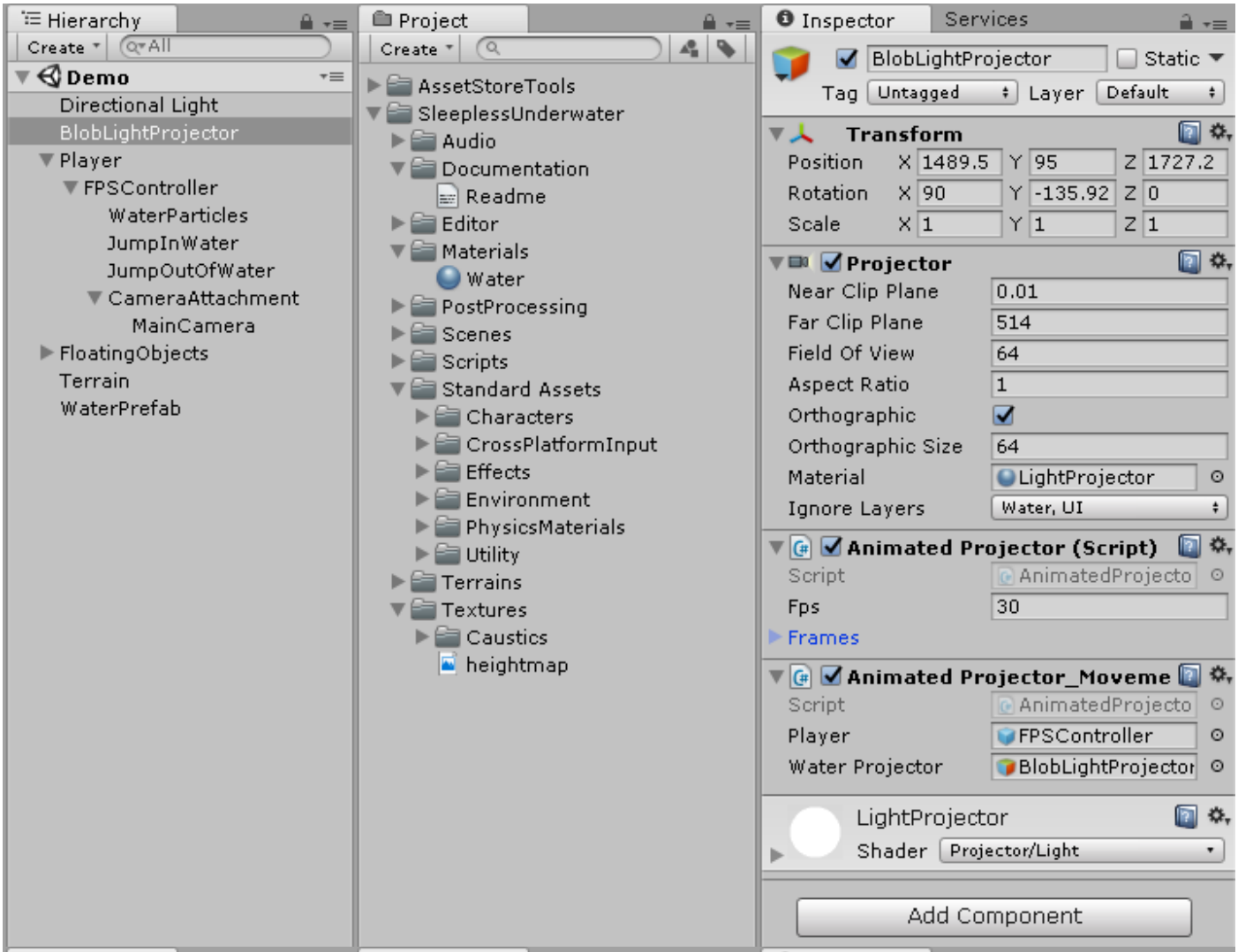
```
104             // get a normal for the surface that is being touched to move along it
105             RaycastHit hitInfo;
106             Physics.SphereCast(transform.position, m_CharacterController.radius, Vector3.down, out hitInfo,
107                                m_CharacterController.height/2f, Physics.AllLayers, QueryTriggerInteraction.Ignore);
108             desiredMove = Vector3.ProjectOnPlane(desiredMove, hitInfo.normal).normalized;
109
110             m_MoveDir.x = desiredMove.x*speed;
111             m_MoveDir.z = desiredMove.z*speed;
112
113
114             if (m_CharacterController.isGrounded && m_IsSwimming == false)
115             {
116                 m_MoveDir.y = -m_StickToGroundForce;
117
118                 if (m_Jump)
119                 {
120                     m_MoveDir.y = m_JumpSpeed;
121                     PlayJumpSound();
122                     m_Jump = false;
123                     m_Jumping = true;
124                 }
125             }
126             else
127             {
128                 if (m_IsSwimming == true)
129                 {
130                     m_Jump = false;
131                     m_Jumping = false;
132                     // Move slower in water - down to 70% - otherwise go normal speed
133                     m_MoveDir *= m_IsSwimming ? 0.7f : 1;
134                     m_MoveDir += Physics.gravity * (Time.fixedDeltaTime / 8);
135                 }
136                 else
137                 {
138                     m_MoveDir += Physics.gravity * m_GravityMultiplier * Time.fixedDeltaTime;
139                 }
140             }
141             m_CollisionFlags = m_CharacterController.Move(m_MoveDir*Time.fixedDeltaTime);
142             // If we hit space and in water swimming, move up
143             if (CrossPlatformInputManager.GetButton("Jump") && m_IsSwimming == true)
144             {
145                 m_MoveDir.y += 1 + Time.fixedDeltaTime;
146                 //rigidbody.AddForce(transform.up * jumpForce, ForceMode.Impulse);
147             }
148             if (CrossPlatformInputManager.GetButton("Crouch") && m_IsSwimming == true)
149             {
150                 m_MoveDir.y -= 1 + Time.fixedDeltaTime;
151                 //rigidbody.AddForce(transform.up * jumpForce, ForceMode.Impulse);
152             }
153             ProgressStepCycle(speed);
154             UpdateCameraPosition(speed);
```

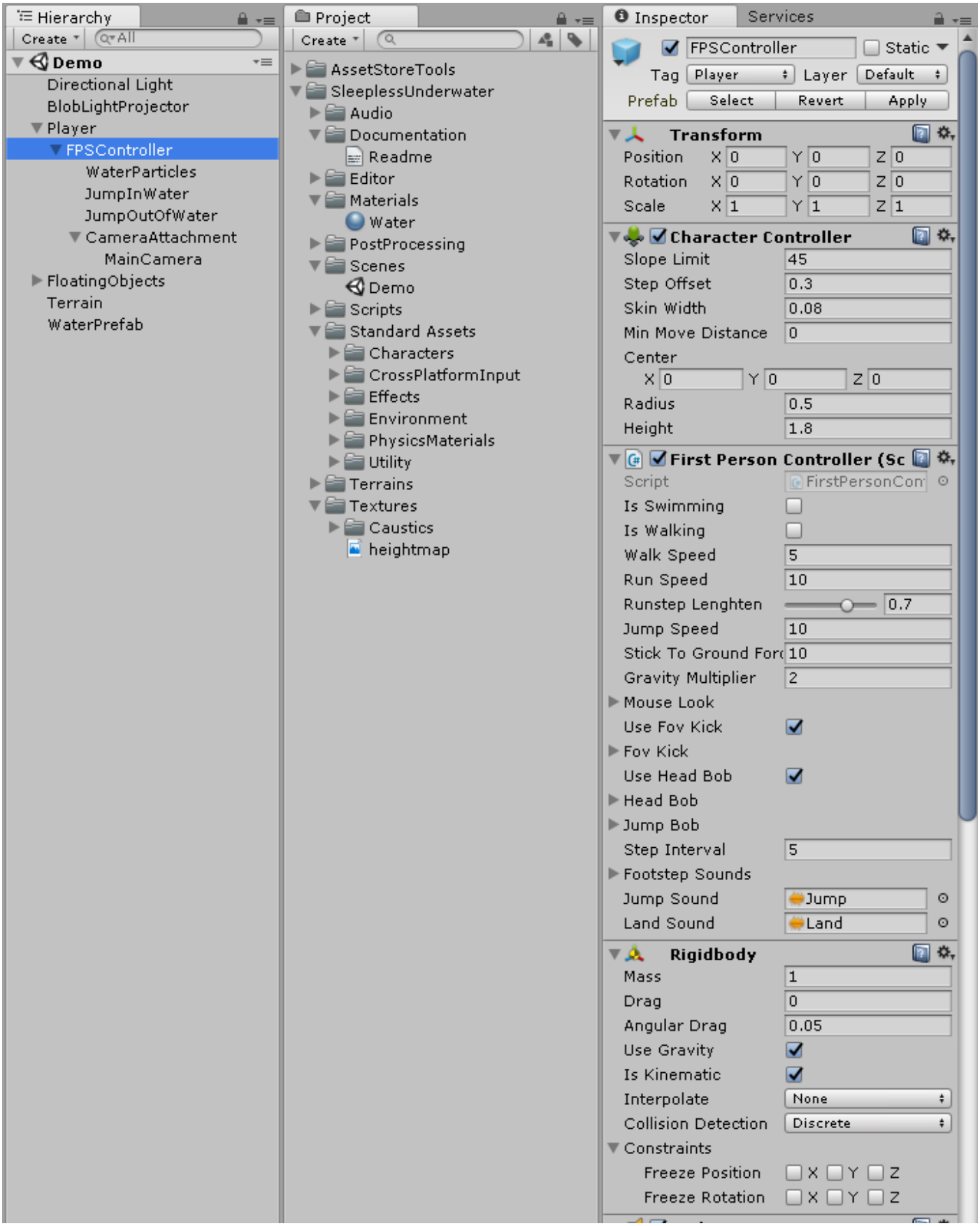
```
        1 reference
        private void PlayFootStepAudio()
        {
            if (!m_CharacterController.isGrounded || m_IsSwimming == true)
            {
                return;
            }
            // pick & play a random footstep sound from the array,
            // excluding sound at index 0
            int n = Random.Range(1, m_FootstepSounds.Length);
            m_AudioSource.clip = m_FootstepSounds[n];
            m_AudioSource.PlayOneShot(m_AudioSource.clip);
            // move picked sound to index 0 so it's not picked next time
            m_FootstepSounds[n] = m_FootstepSounds[0];
            m_FootstepSounds[0] = m_AudioSource.clip;
        }


        1 reference
        private void UpdateCameraPosition(float speed)
        {
            Vector3 newCameraPosition;
            if (!m_UseHeadBob)
            {
                return;
            }
            if (m_CharacterController.velocity.magnitude > 0 && m_CharacterController.isGrounded && m_IsSwimming == false)
            {
                m_Camera.transform.localPosition =
                    m_HeadBob.DoHeadBob(m_CharacterController.velocity.magnitude +
                                        (speed*(m_IsWalking ? 1f : m_RunstepLenghten)));
                newCameraPosition = m_Camera.transform.localPosition;
                newCameraPosition.y = m_Camera.transform.localPosition.y - m_JumpBob.Offset();
            }
            else
            {
                newCameraPosition = m_Camera.transform.localPosition;
                newCameraPosition.y = m_OriginalCameraPosition.y - m_JumpBob.Offset();
            }
            m_Camera.transform.localPosition = newCameraPosition;
        }
```

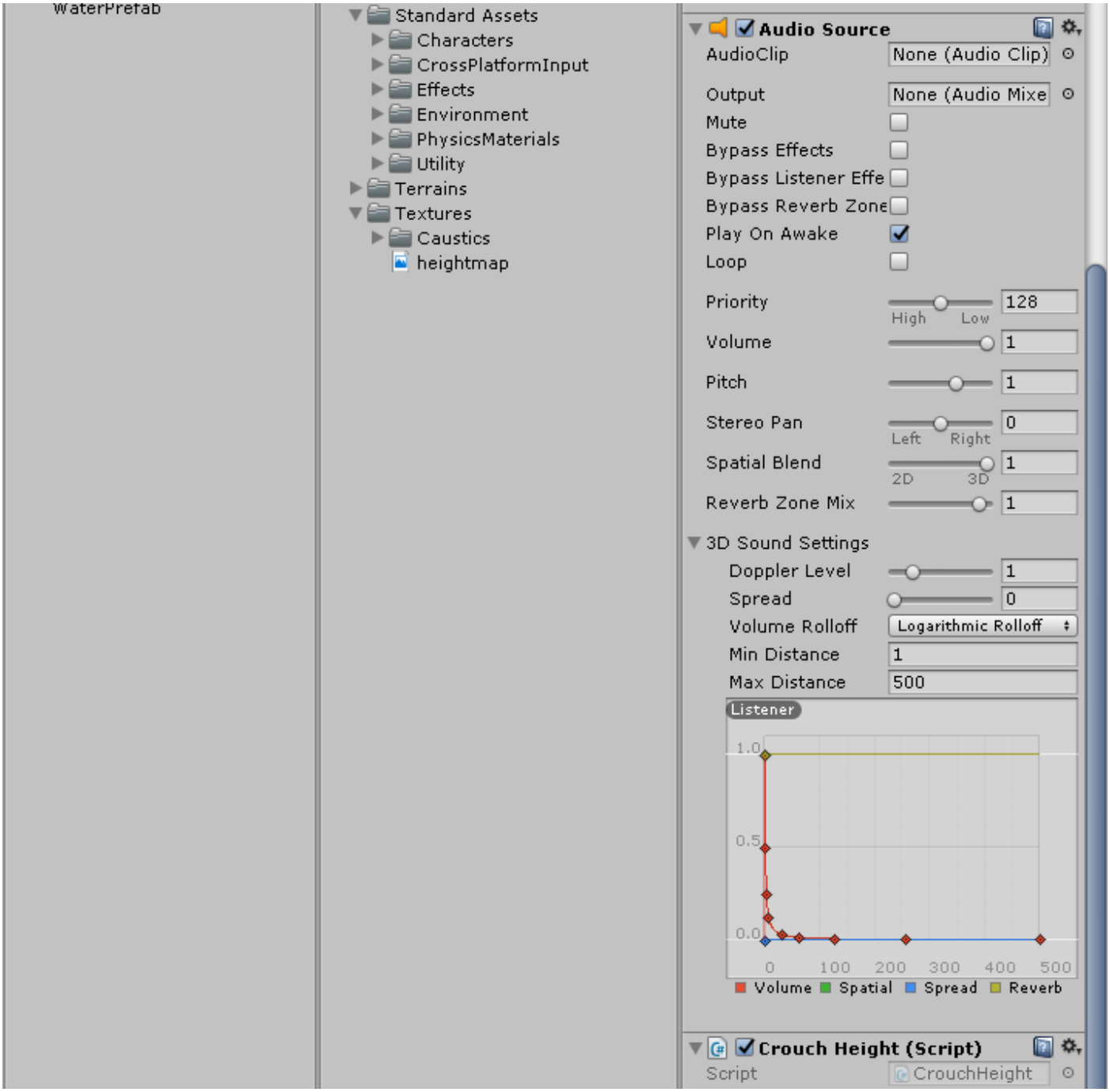
```
        1 reference
        private void RotateView()
        {
            m_MouseLook.LookRotation (transform, m_Camera.transform);
        }

        // Set Player to isSwimming true
        0 references
        public void inWater()
        {
            m_IsSwimming = true;
        }
        // Set Player to isSwimming false
        0 references
        public void outOfWater()
        {
            m_IsSwimming = false;
        }
        0 references
        private void OnControllerColliderHit(ControllerColliderHit hit)
        {
            Rigidbody body = hit.collider.attachedRigidbody;
            //dont move the rigidbody if the character is on top of it
            if (m_CollisionFlags == CollisionFlags.Below)
            {
                return;
            }

            if (body == null || body.isKinematic)
            {
                return;
            }
            body.AddForceAtPosition(m_CharacterController.velocity*0.1f, hit.point, ForceMode.Impulse);
        }
        1 reference
        public bool CanSwim
        {
            get { return m_IsSwimming; }
        }
    }
}
```

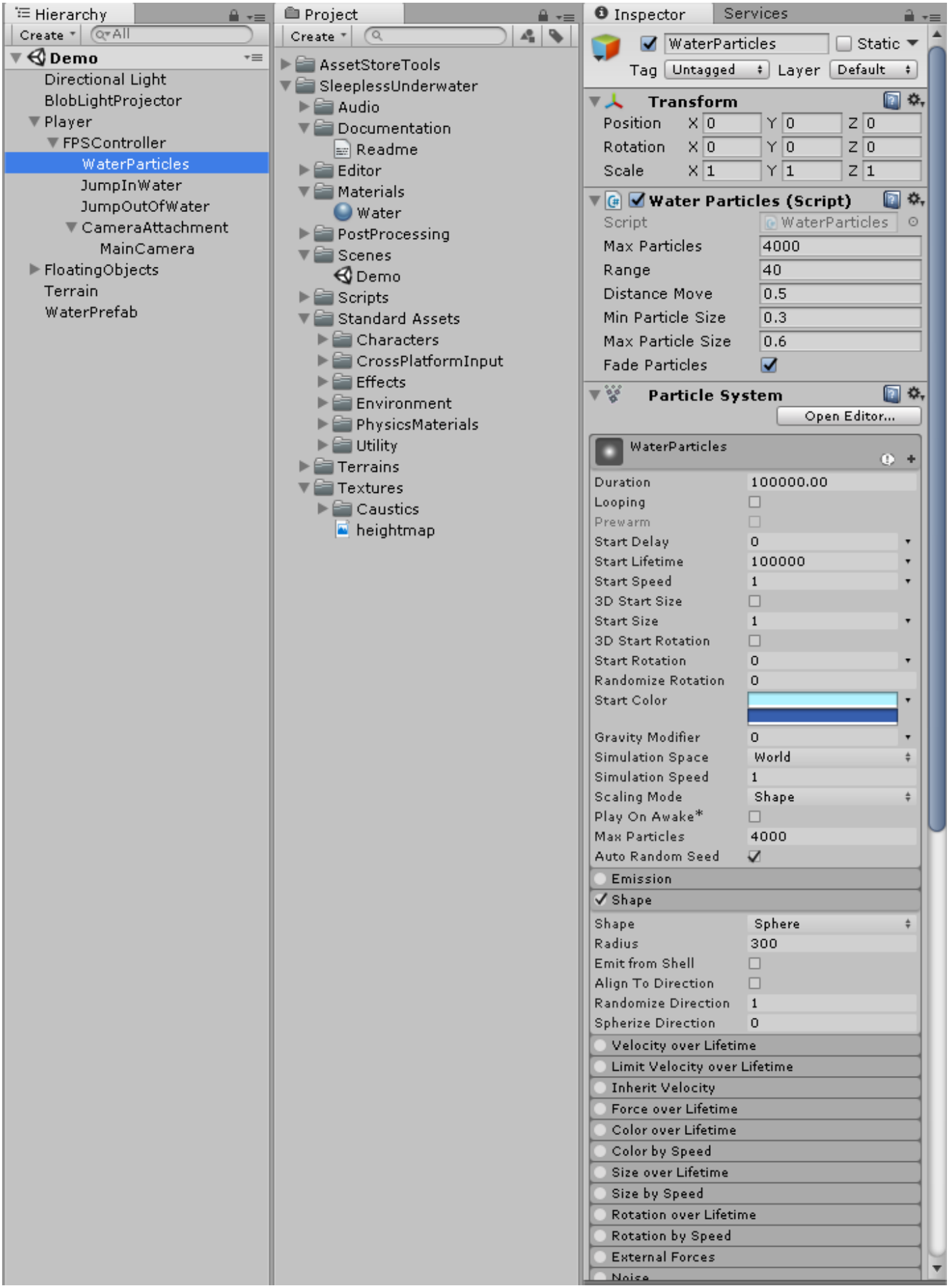
Relevant Asset Screenshots

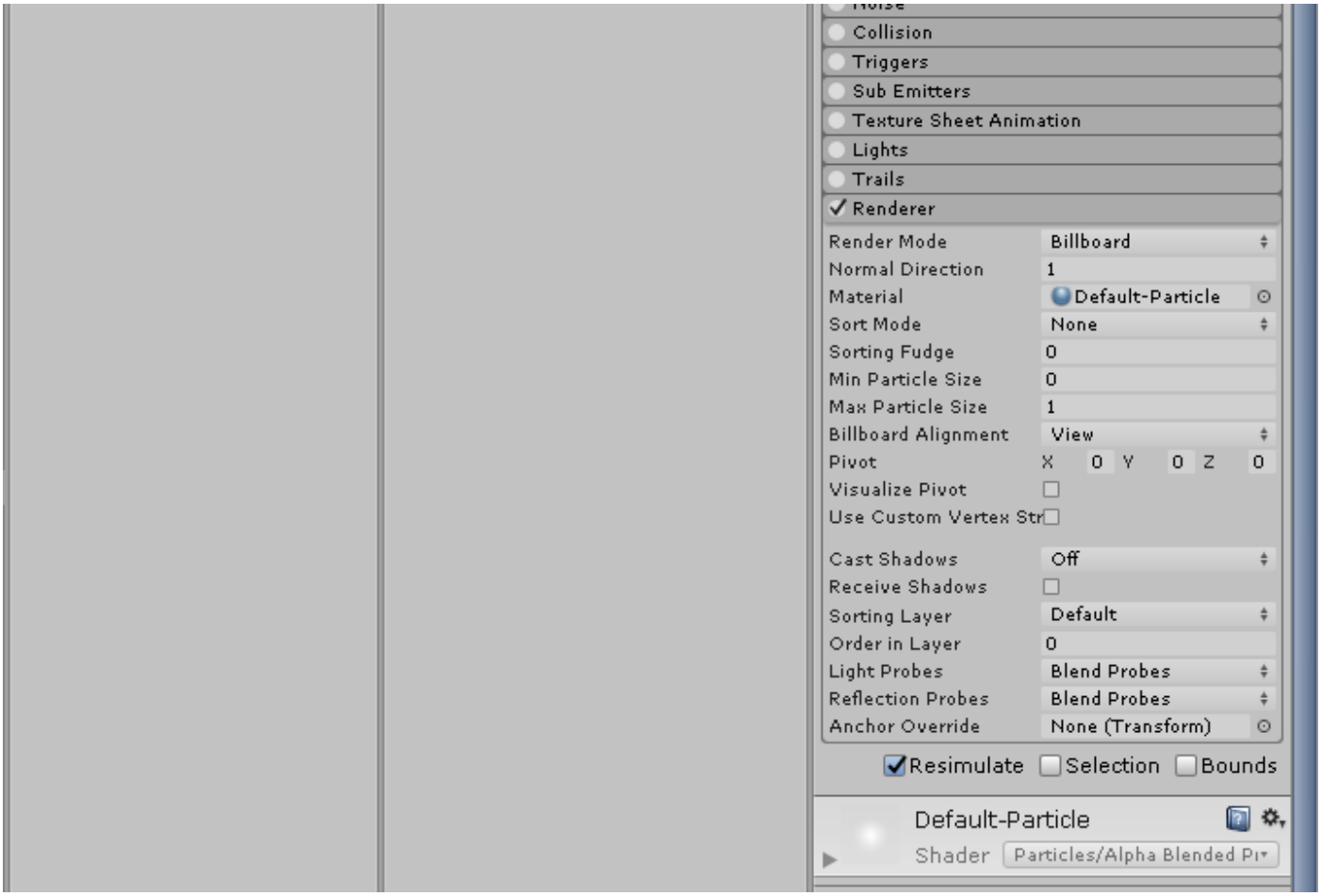


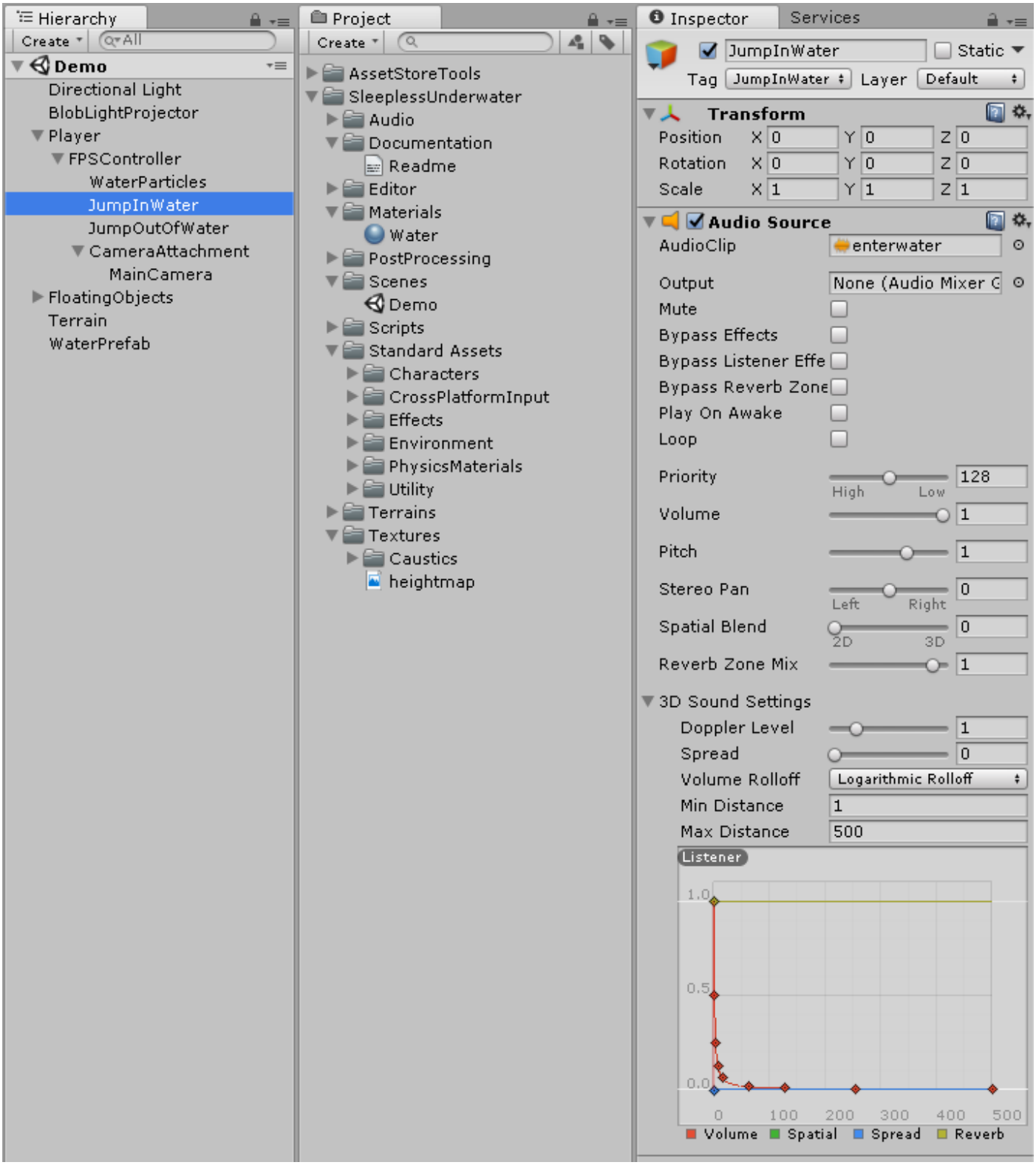

Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
BlobLightProjector
Static
Tag
Untagged
Layer
Default
Transform
Position X 1489.5 Y 95 Z 1727.2
Rotation X 90 Y -135.92 Z 0
Scale X 1 Y 1 Z 1
Projector
Near Clip Plane 0.01
Far Clip Plane 514
Field Of View 64
Aspect Ratio 1
Orthographic
Orthographic Size 64
Material LightProjector
Ignore Layers Water, UI
Animated Projector (Script)
Script AnimatedProjecto
Fps 30
Frames
Animated Projector_Moveme
Script AnimatedProjecto
Player FPSController
Water Projector BlobLightProjector
LightProjector
Shader Projector/Light
Add Component
Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
Player
Static
Tag
Untagged
Layer
Default
Transform
Position X 1476.1 Y 84.6 Z 1717.1
Rotation X 0 Y 163.423 Z 0
Scale X 1 Y 1 Z 1
Add Component

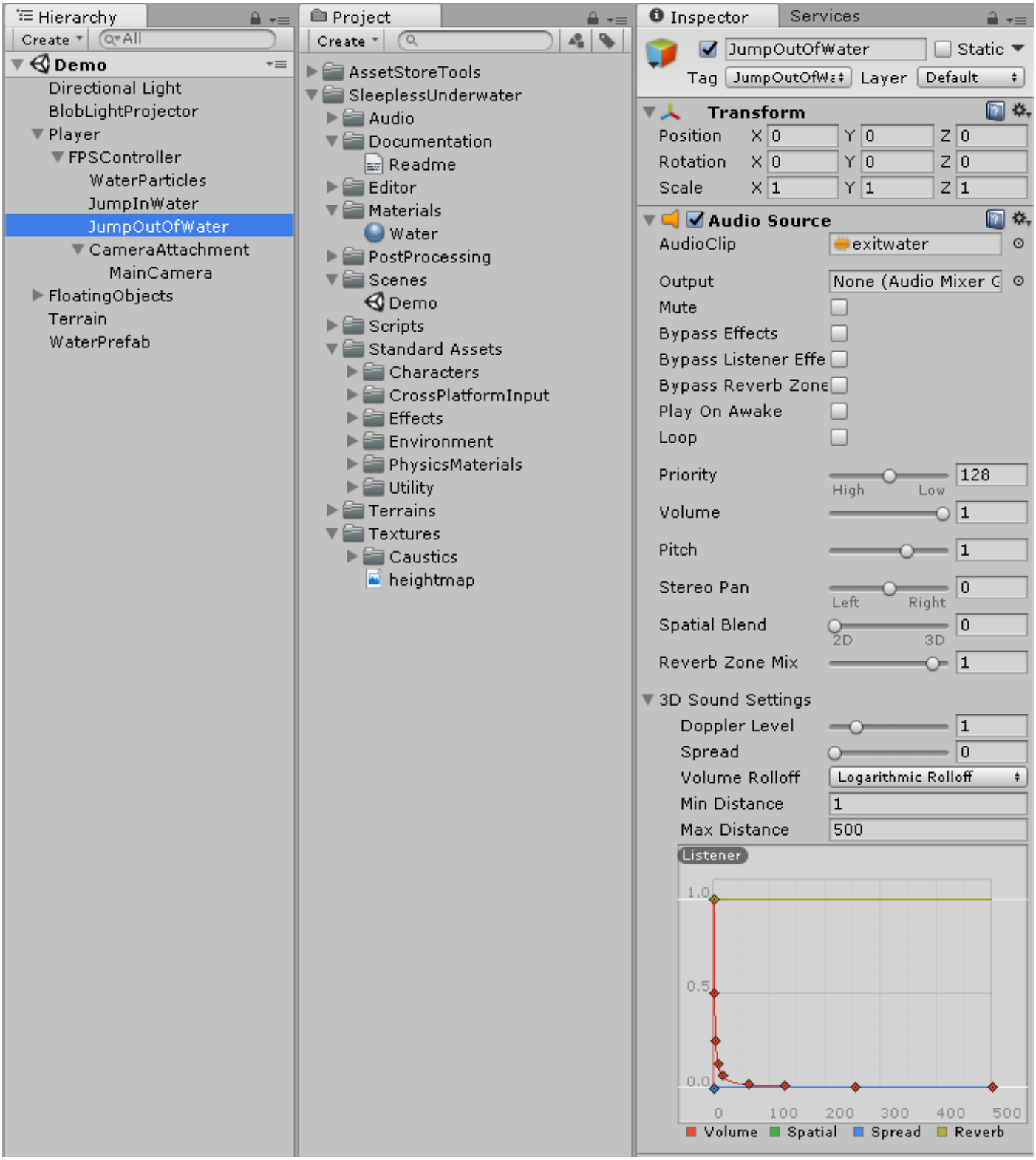

Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
FPSController
Static
Tag
Player
Layer
Default
Prefab
Select
Revert
Apply
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Character Controller
Slope Limit 45
Step Offset 0.3
Skin Width 0.08
Min Move Distance 0
Center
X 0 Y 0 Z 0
Radius 0.5
Height 1.8
First Person Controller (Sc
Script FirstPersonCon
Is Swimming
Is Walking
Walk Speed 5
Run Speed 10
Runstep Lenghten 0.7
Jump Speed 10
Stick To Ground For 10
Gravity Multiplier 2
Mouse Look
Use Fov Kick
Fov Kick
Use Head Bob
Head Bob
Jump Bob
Step Interval 5
Footstep Sounds
Jump Sound Jump
Land Sound Land
Rigidbody
Mass 1
Drag 0
Angular Drag 0.05
Use Gravity
Is Kinematic
Interpolate None
Collision Detection Discrete
Constraints
Freeze Position X Y Z
Freeze Rotation X Y Z

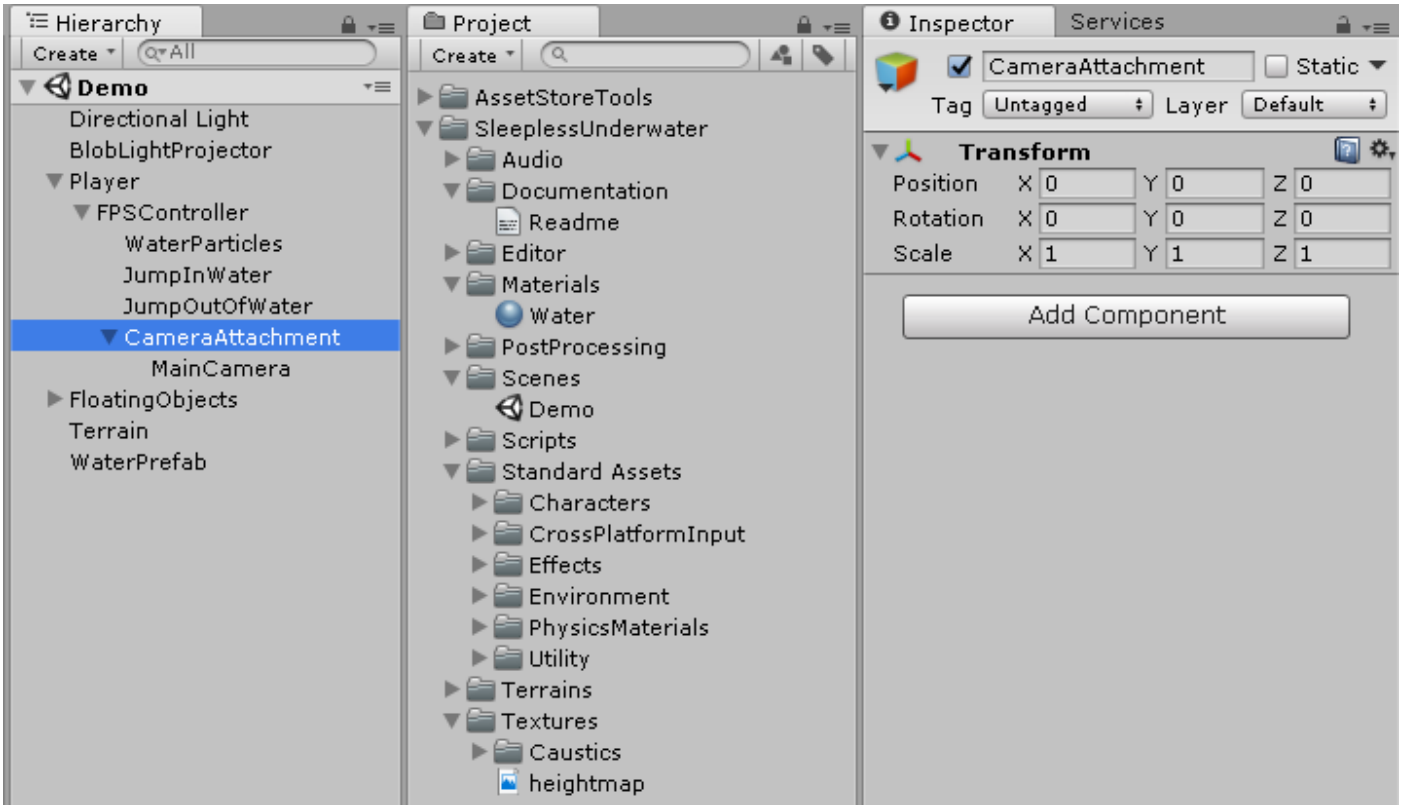

WaterPrefab
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Audio Source
AudioClip
None (Audio Clip)
Output
None (Audio Mixe
Mute
Bypass Effects
Bypass Listener Effe
Bypass Reverb Zone
Play On Awake
Loop
Priority
128
High
Low
Volume
1
Pitch
1
Stereo Pan
0
Left
Right
Spatial Blend
1
2D
3D
Reverb Zone Mix
1
3D Sound Settings
Doppler Level
1
Spread
0
Volume Rolloff
Logarithmic Rolloff
Min Distance
1
Max Distance
500
Listener
1.0
0.5
0.0
0
100
200
300
400
500
Volume
Spatial
Spread
Reverb
Crouch Height (Script)
Script
CrouchHeight

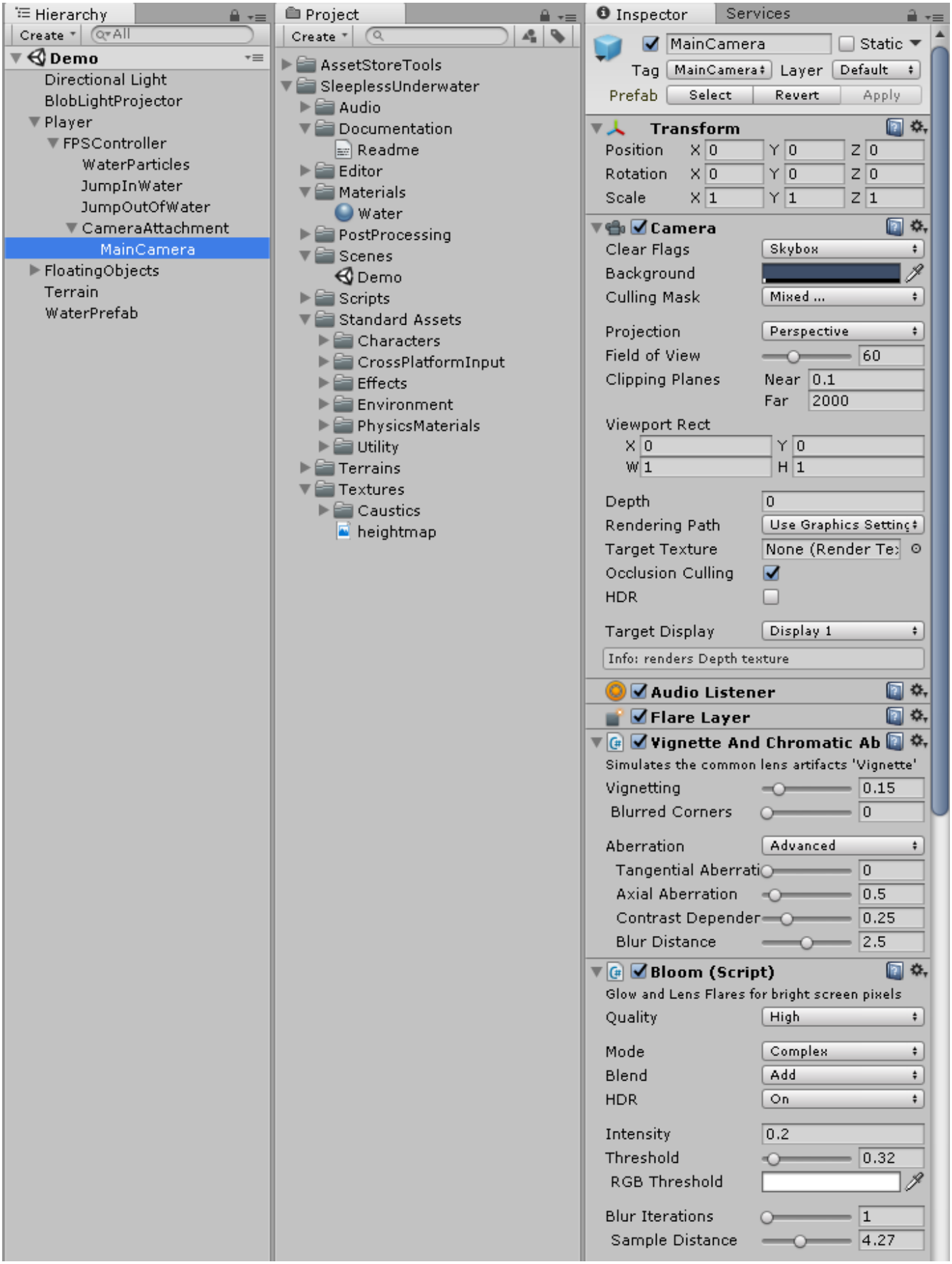

Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
WaterParticles
Static
Tag Untagged
Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Water Particles (Script)
Script WaterParticles
Max Particles 4000
Range 40
Distance Move 0.5
Min Particle Size 0.3
Max Particle Size 0.6
Fade Particles
Particle System
Open Editor...
WaterParticles
Duration 100000.00
Looping
Prewarm
Start Delay 0
Start Lifetime 100000
Start Speed 1
3D Start Size
Start Size 1
3D Start Rotation
Start Rotation 0
Randomize Rotation 0
Start Color
Gravity Modifier 0
Simulation Space World
Simulation Speed 1
Scaling Mode Shape
Play On Awake*
Max Particles 4000
Auto Random Seed
Emission
Shape
Shape Sphere
Radius 300
Emit from Shell
Align To Direction
Randomize Direction 1
Spherize Direction 0
Velocity over Lifetime
Limit Velocity over Lifetime
Inherit Velocity
Force over Lifetime
Color over Lifetime
Color by Speed
Size over Lifetime
Size by Speed
Rotation over Lifetime
Rotation by Speed
External Forces
Noise

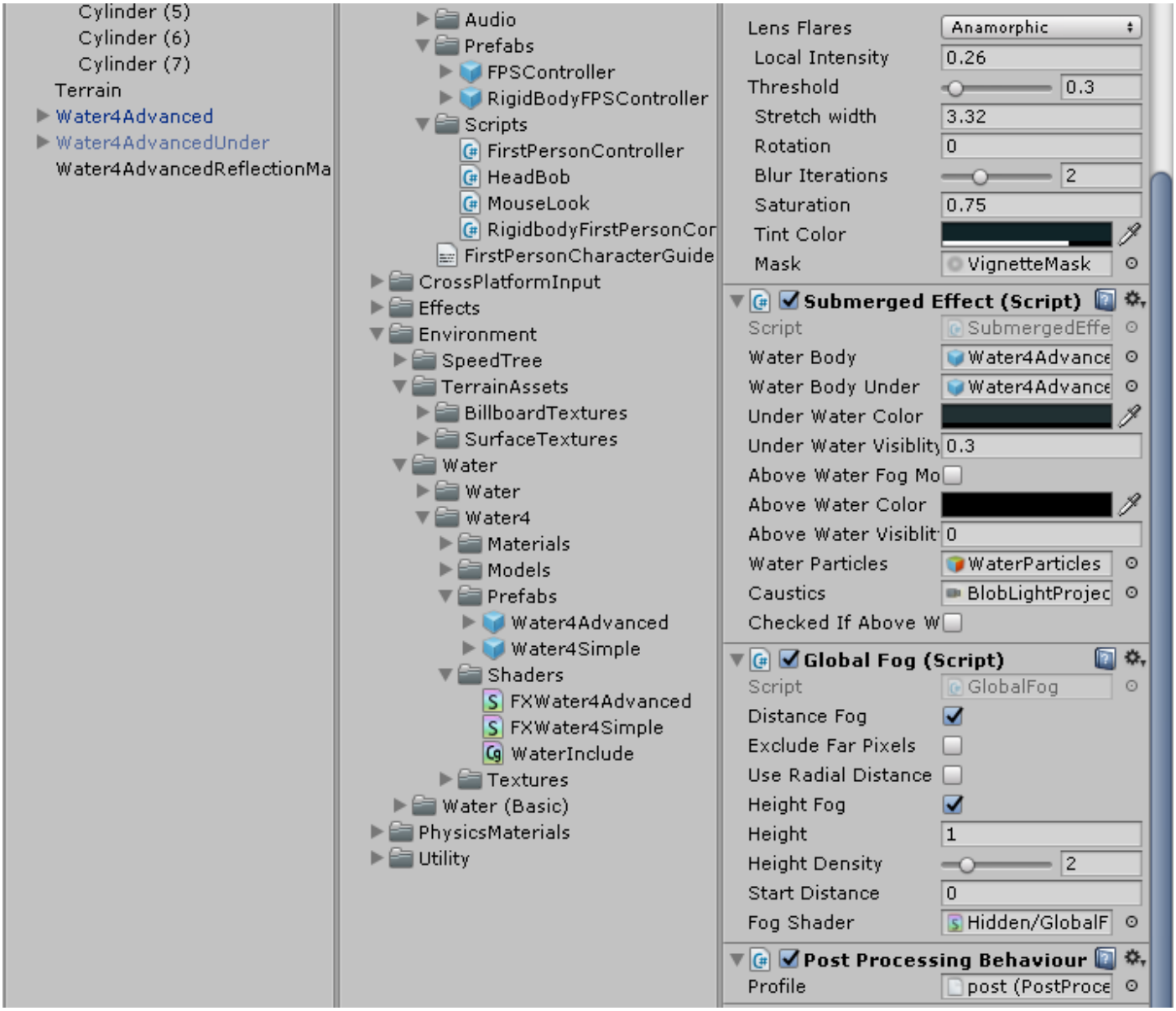

Collision
Triggers
Sub Emitters
Texture Sheet Animation
Lights
Trails
Renderer
Render Mode
Billboard
Normal Direction
1
Material
Default-Particle
Sort Mode
None
Sorting Fudge
0
Min Particle Size
0
Max Particle Size
1
Billboard Alignment
View
Pivot
X 0 Y 0 Z 0
Visualize Pivot
Use Custom Vertex Str
Cast Shadows
Off
Receive Shadows
Sorting Layer
Default
Order in Layer
0
Light Probes
Blend Probes
Reflection Probes
Blend Probes
Anchor Override
None (Transform)
Resimulate
Selection
Bounds
Default-Particle
Shader
Particles/Alpha Blended Pr

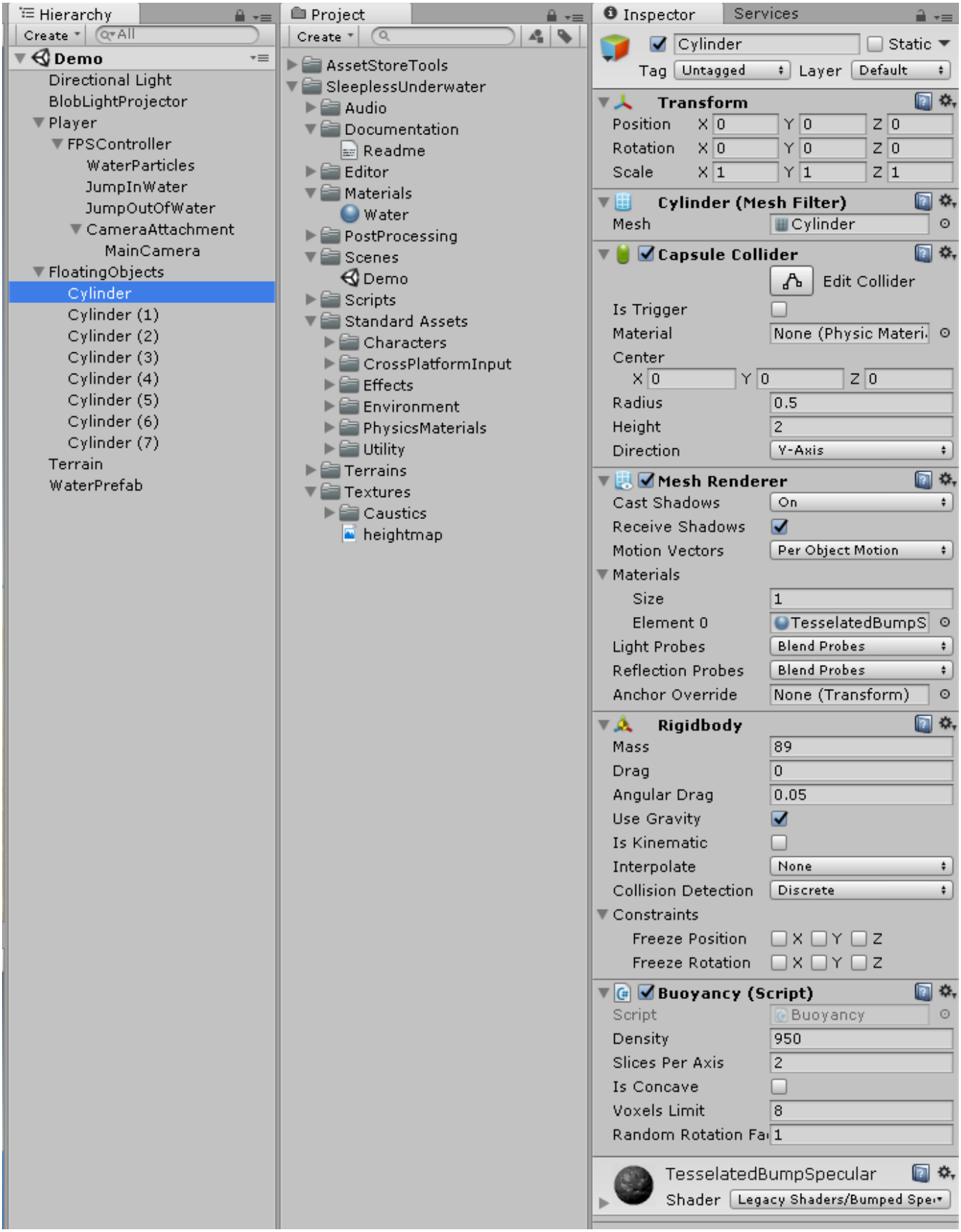

Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
JumpInWater
Static
Tag JumpInWater
Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Audio Source
AudioClip enterwater
Output None (Audio Mixer G
Mute
Bypass Effects
Bypass Listener Effe
Bypass Reverb Zone
Play On Awake
Loop
Priority 128
High Low
Volume 1
Pitch 1
Stereo Pan 0
Left Right
Spatial Blend 0
2D 3D
Reverb Zone Mix 1
3D Sound Settings
Doppler Level 1
Spread 0
Volume Rolloff Logarithmic Rolloff
Min Distance 1
Max Distance 500
Listener
1.0
0.5
0.0
0 100 200 300 400 500
Volume Spatial Spread Reverb

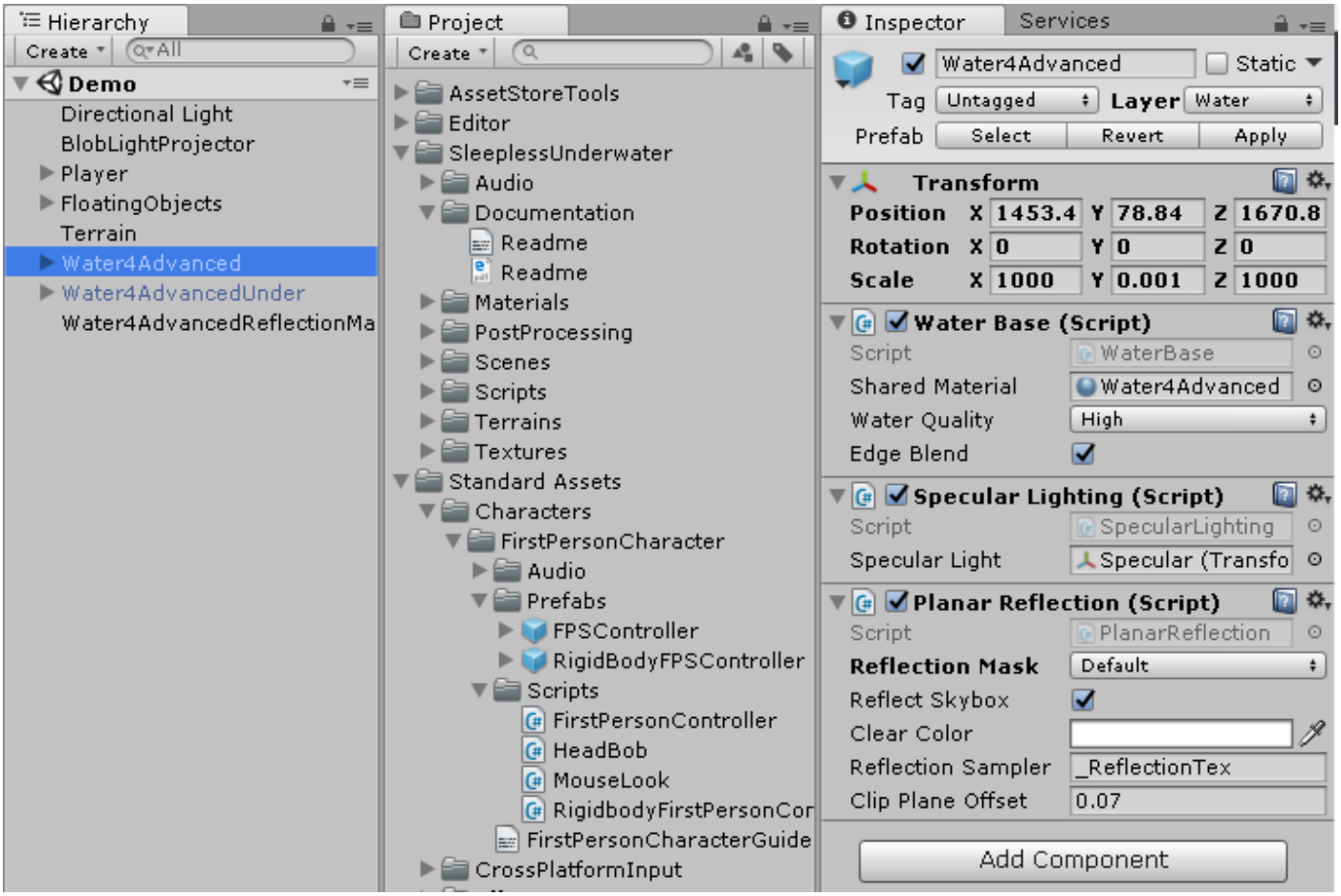

Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
JumpOutOfWater
Static
Tag
Layer
Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Audio Source
AudioClip
exitwater
Output
None (Audio Mixer G
Mute
Bypass Effects
Bypass Listener Effe
Bypass Reverb Zone
Play On Awake
Loop
Priority
128
High
Low
Volume
1
Pitch
1
Stereo Pan
0
Left
Right
Spatial Blend
0
2D
3D
Reverb Zone Mix
1
3D Sound Settings
Doppler Level
1
Spread
0
Volume Rolloff
Logarithmic Rolloff
Min Distance
1
Max Distance
500
Listener
1.0
0.5
0.0
0
100
200
300
400
500
Volume
Spatial
Spread
Reverb

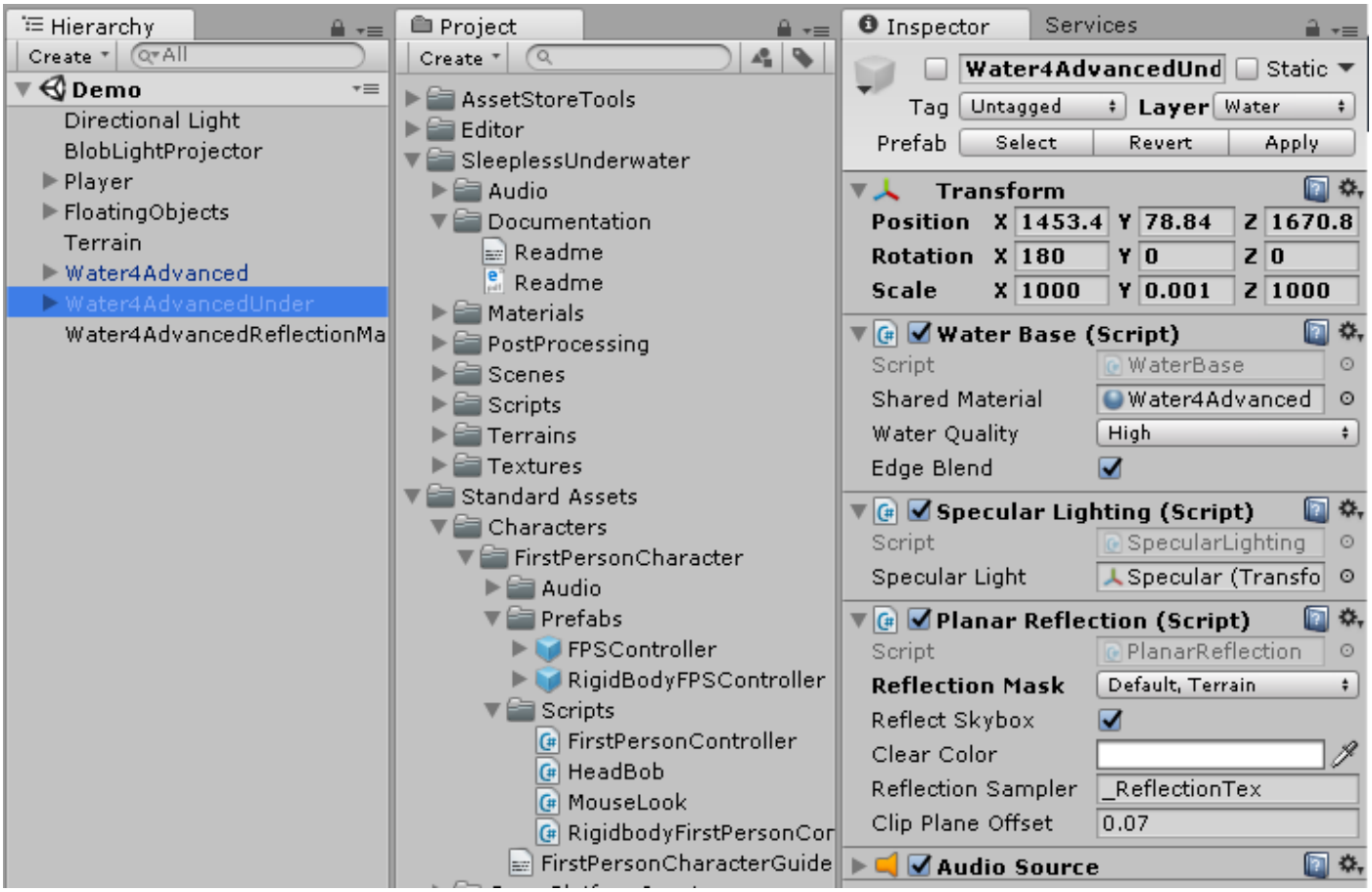

Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
CameraAttachment
Static
Tag
Untagged
Layer
Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Add Component


Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
MainCamera
Static
Tag
MainCamera
Layer
Default
Prefab
Select
Revert
Apply
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Camera
Clear Flags
Skybox
Background
Culling Mask
Mixed ...
Projection
Perspective
Field of View
60
Clipping Planes
Near 0.1
Far 2000
Viewport Rect
X 0 Y 0
W 1 H 1
Depth
0
Rendering Path
Use Graphics Setting
Target Texture
None (Render Te
Occlusion Culling
HDR
Target Display
Display 1
Info: renders Depth texture
Audio Listener
Flare Layer
Vignette And Chromatic Ab
Simulates the common lens artifacts 'Vignette'
Vignetting
0.15
Blurred Corners
0
Aberration
Advanced
Tangential Aberrati
0
Axial Aberration
0.5
Contrast Depender
0.25
Blur Distance
2.5
Bloom (Script)
Glow and Lens Flares for bright screen pixels
Quality
High
Mode
Complex
Blend
Add
HDR
On
Intensity
0.2
Threshold
0.32
RGB Threshold
Blur Iterations
1
Sample Distance
4.27


Cylinder (5)
Cylinder (6)
Cylinder (7)
Terrain
Water4Advanced
Water4AdvancedUnder
Water4AdvancedReflectionMa
Audio
Prefabs
FPSController
RigidBodyFPSController
Scripts
FirstPersonController
HeadBob
MouseLook
RigidbodyFirstPersonCor
FirstPersonCharacterGuide
CrossPlatformInput
Effects
Environment
SpeedTree
TerrainAssets
BillboardTextures
SurfaceTextures
Water
Water
Water4
Materials
Models
Prefabs
Water4Advanced
Water4Simple
Shaders
FXWater4Advanced
FXWater4Simple
WaterInclude
Textures
Water (Basic)
PhysicsMaterials
Utility
Lens Flares
Anamorphic
Local Intensity
0.26
Threshold
0.3
Stretch width
3.32
Rotation
0
Blur Iterations
2
Saturation
0.75
Tint Color
Mask
VignetteMask
Submerged Effect (Script)
Script
SubmergedEffe
Water Body
Water4Advance
Water Body Under
Water4Advance
Under Water Color
Under Water Visiblity
0.3
Above Water Fog Mo
Above Water Color
Above Water Visiblit
0
Water Particles
WaterParticles
Caustics
BlobLightProjec
Checked If Above W
Global Fog (Script)
Script
GlobalFog
Distance Fog
Exclude Far Pixels
Use Radial Distance
Height Fog
Height
1
Height Density
2
Start Distance
0
Fog Shader
Hidden/GlobalF
Post Processing Behaviour
Profile
post (PostProce


Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FPSController
WaterParticles
JumpInWater
JumpOutOfWater
CameraAttachment
MainCamera
FloatingObjects
Cylinder
Cylinder (1)
Cylinder (2)
Cylinder (3)
Cylinder (4)
Cylinder (5)
Cylinder (6)
Cylinder (7)
Terrain
WaterPrefab
Project
Create
AssetStoreTools
SleeplessUnderwater
Audio
Documentation
Readme
Editor
Materials
Water
PostProcessing
Scenes
Demo
Scripts
Standard Assets
Characters
CrossPlatformInput
Effects
Environment
PhysicsMaterials
Utility
Terrains
Textures
Caustics
heightmap
Inspector
Services
Cylinder
Static
Tag Untagged
Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Cylinder (Mesh Filter)
Mesh Cylinder
Capsule Collider
Edit Collider
Is Trigger
Material None (Physic Materi.
Center
X 0 Y 0 Z 0
Radius 0.5
Height 2
Direction Y-Axis
Mesh Renderer
Cast Shadows On
Receive Shadows
Motion Vectors Per Object Motion
Materials
Size 1
Element 0 TesselatedBumpS
Light Probes Blend Probes
Reflection Probes Blend Probes
Anchor Override None (Transform)
Rigidbody
Mass 89
Drag 0
Angular Drag 0.05
Use Gravity
Is Kinematic
Interpolate None
Collision Detection Discrete
Constraints
Freeze Position X Y Z
Freeze Rotation X Y Z
Buoyancy (Script)
Script Buoyancy
Density 950
Slices Per Axis 2
Is Concave
Voxels Limit 8
Random Rotation Fa 1
TesselatedBumpSpecular
Shader Legacy Shaders/Bumped Spe


Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FloatingObjects
Terrain
Water4Advanced
Water4AdvancedUnder
Water4AdvancedReflectionMa
Project
Create
AssetStoreTools
Editor
SleeplessUnderwater
Audio
Documentation
Readme
Readme
Materials
PostProcessing
Scenes
Scripts
Terrains
Textures
Standard Assets
Characters
FirstPersonCharacter
Audio
Prefabs
FPSController
RigidBodyFPSController
Scripts
FirstPersonController
HeadBob
MouseLook
RigidbodyFirstPersonCor
FirstPersonCharacterGuide
CrossPlatformInput
Inspector
Services
Water4Advanced
Static
Tag
Untagged
Layer
Water
Prefab
Select
Revert
Apply
Transform
Position X 1453.4 Y 78.84 Z 1670.8
Rotation X 0 Y 0 Z 0
Scale X 1000 Y 0.001 Z 1000
Water Base (Script)
Script
WaterBase
Shared Material
Water4Advanced
Water Quality
High
Edge Blend
Specular Lighting (Script)
Script
SpecularLighting
Specular Light
Specular (Transfo
Planar Reflection (Script)
Script
PlanarReflection
Reflection Mask
Default
Reflect Skybox
Clear Color
Reflection Sampler
_ReflectionTex
Clip Plane Offset
0.07
Add Component


Hierarchy
Create
All
Demo
Directional Light
BlobLightProjector
Player
FloatingObjects
Terrain
Water4Advanced
Water4AdvancedUnder
Water4AdvancedReflectionMa
Project
Create
AssetStoreTools
Editor
SleeplessUnderwater
Audio
Documentation
Readme
Readme
Materials
PostProcessing
Scenes
Scripts
Terrains
Textures
Standard Assets
Characters
FirstPersonCharacter
Audio
Prefabs
FPSController
RigidBodyFPSController
Scripts
FirstPersonController
HeadBob
MouseLook
RigidbodyFirstPersonCor
FirstPersonCharacterGuide
Inspector
Services
Water4AdvancedUnd
Static
Tag
Untagged
Layer
Water
Prefab
Select
Revert
Apply
Transform
Position X 1453.4 Y 78.84 Z 1670.8
Rotation X 180 Y 0 Z 0
Scale X 1000 Y 0.001 Z 1000
Water Base (Script)
Script
WaterBase
Shared Material
Water4Advanced
Water Quality
High
Edge Blend
Specular Lighting (Script)
Script
SpecularLighting
Specular Light
Specular (Transfo
Planar Reflection (Script)
Script
PlanarReflection
Reflection Mask
Default, Terrain
Reflect Skybox
Clear Color
Reflection Sampler
_ReflectionTex
Clip Plane Offset
0.07
Audio Source

Contact Us

If you have any questions, comments or improvement suggestions, you can email me: daceian@yahoo.com.au

Other contact details available on our website: www.sleeplessentertainment.com.au