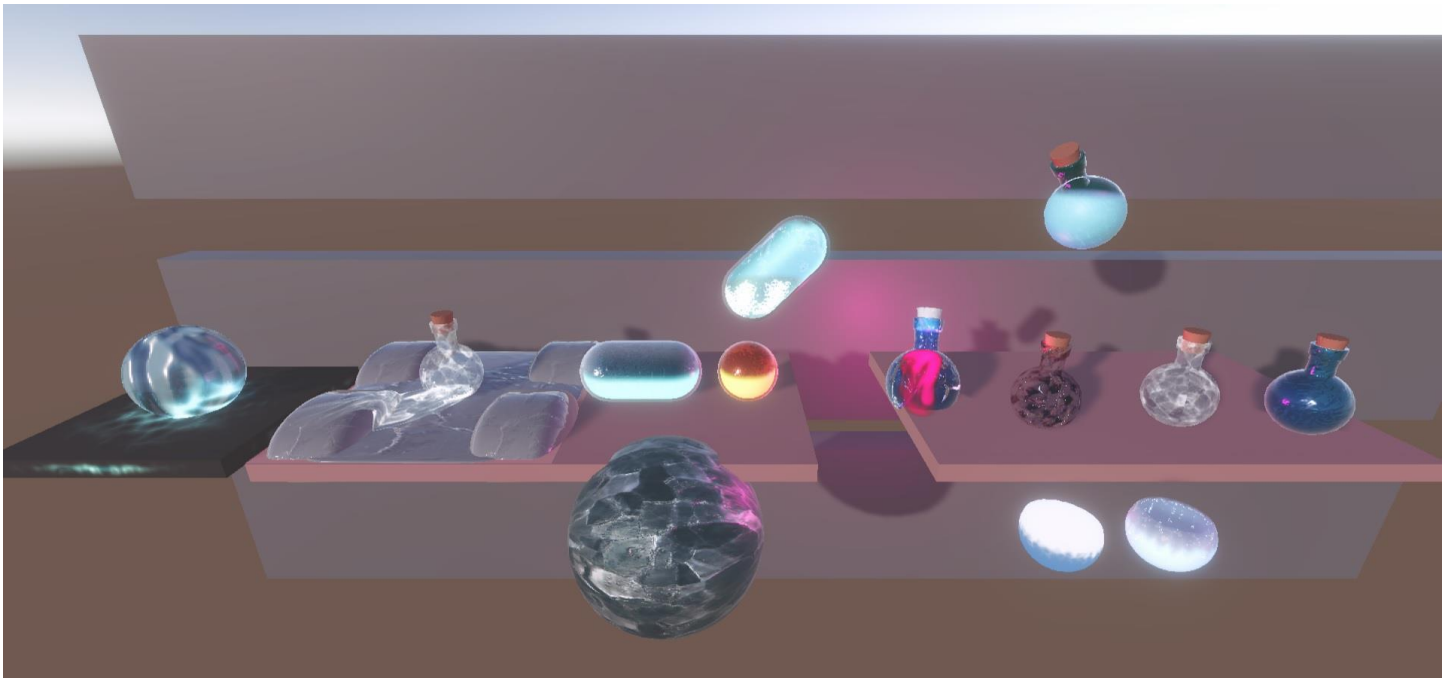
# LIQUA Volumetric Fluids



**v1.1**

**LIQUA** is a next generation tool from **ARTnGAME** for fluid emulation inside containers. The system allows various volumetric effects of fluids and liquids to be presented inside containers and interact in various ways.

Some of the system shaders are planned to be compatible with the **Sky Master ULTIMATE** Weather system, and is planned to be able to receive snow gradually in snow weather, this will be added in next versions of the system.

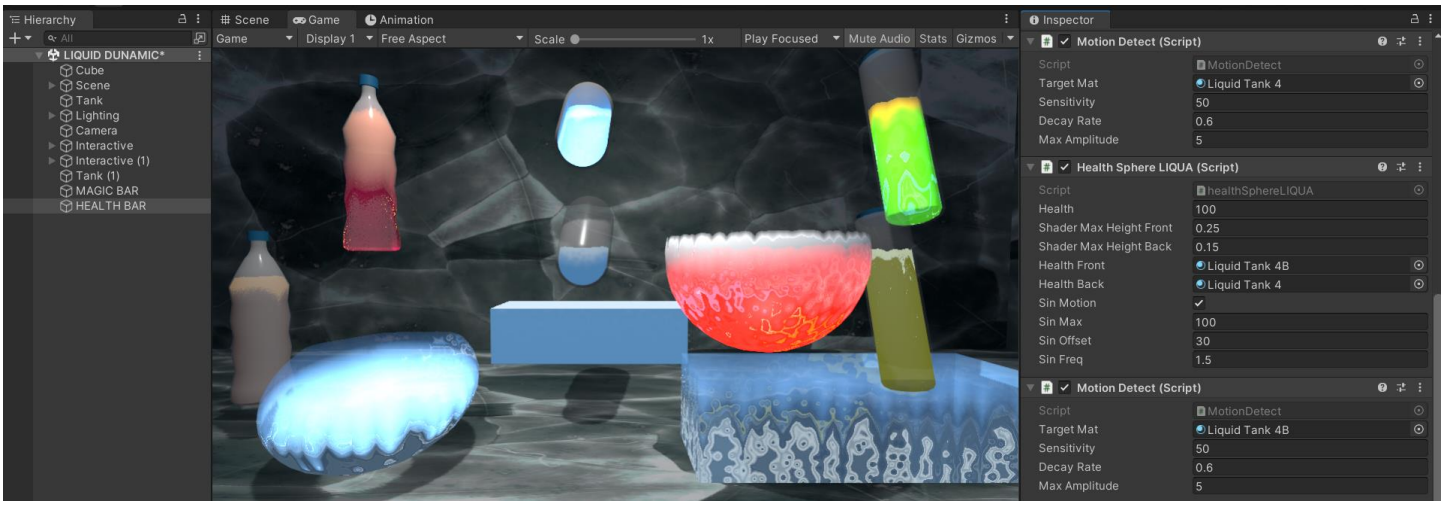
For help with using the system, please visit **ARTnGAME** **Discord channel**, use the **included Manual PDF** and post in the **LIQUA Forum thread**. The system works in URP pipeline.

**The system can also be upgraded to Sky Master ULTIMATE Suit for $39**, which allows access to the full URP weather suit, that includes **Ethereal** volumetric lighting module for URP, plus various types of volumetric clouds and water reflection.

# LIQUA Setup

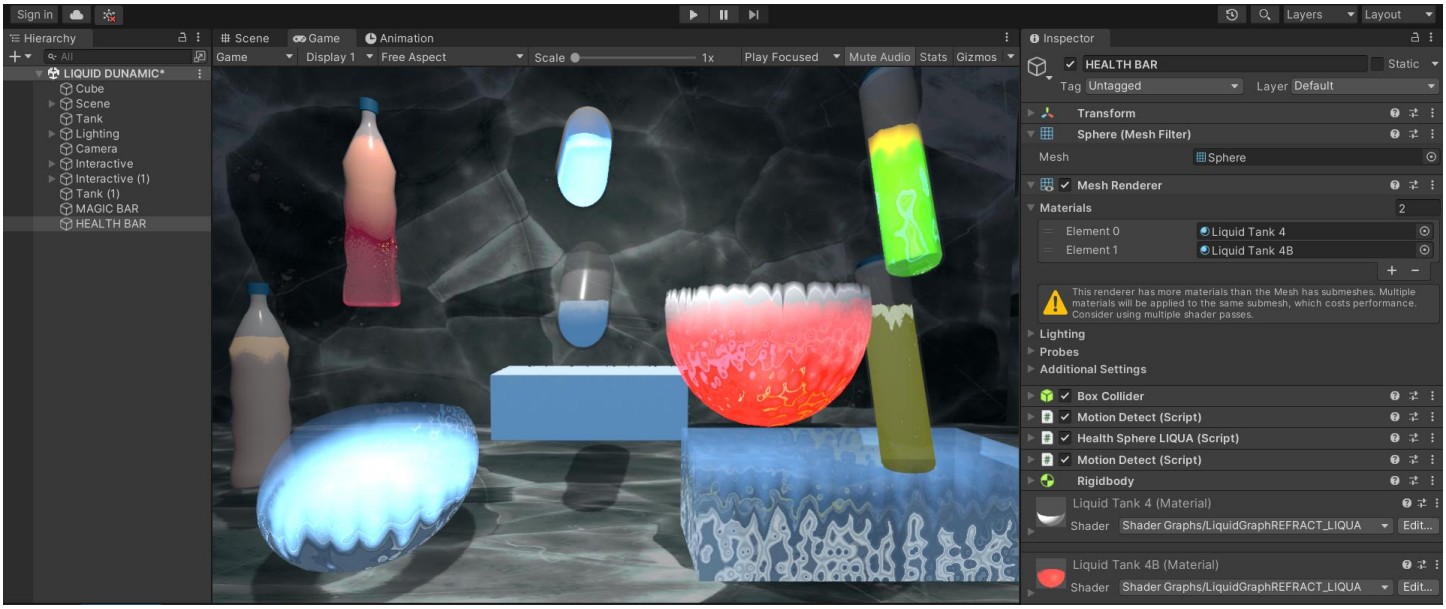
**1. Interactive Liquid setup.**

The core of the **LIQUA** Interactive Liquid inside a container system is the **“Motion Detect”** script. This script reads the motion of the Transform it is attached to and assigns an amplitude to the shader of the liquid container material. The material must be referenced in the **“TargetMat”** slot. The three other parameters define the intensity and sensitivity to motion of the effect and the rate it goes away after the motion stops.



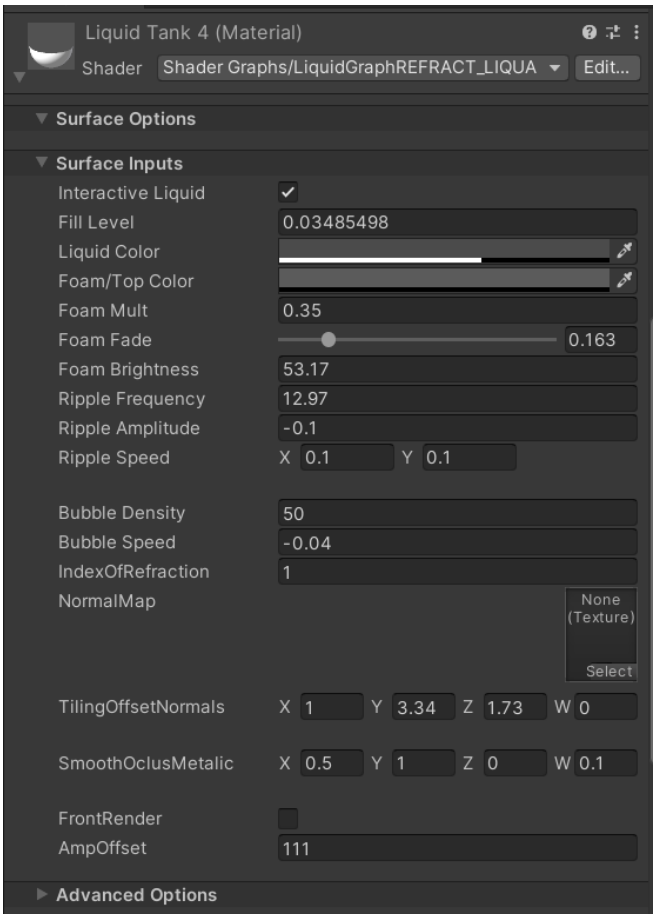

The container can be setup using two material – shader options. One option is using two materials in the same mesh, one for front and one for back side and use the refractive shader or use a single material and the non refractive shader.

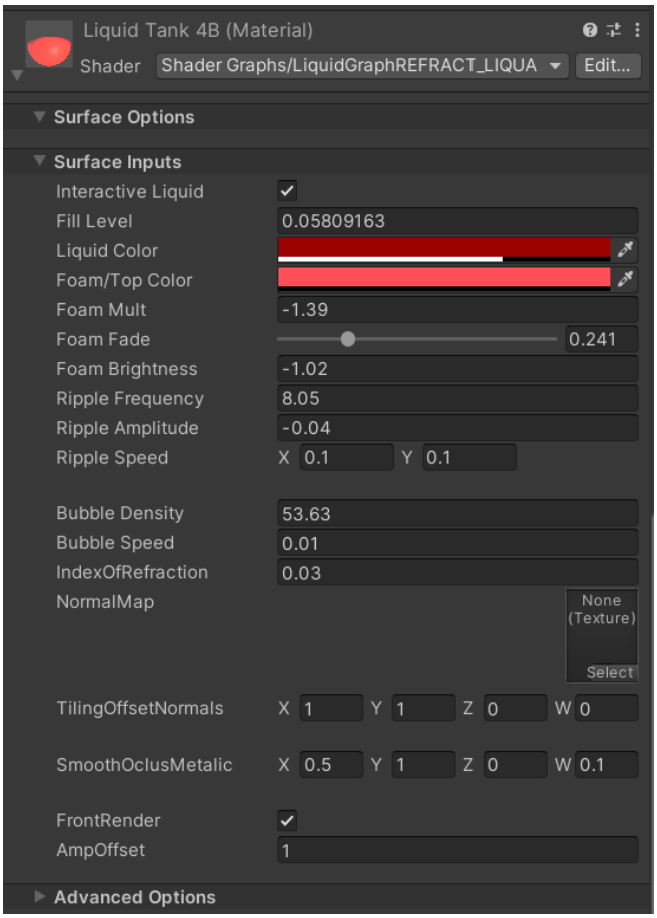
The refractive shader setup is shown below, with two materials in the mesh renderer.



**Refractive materials parameters (“LiquidREFRACT_LIQUA” Shader)**

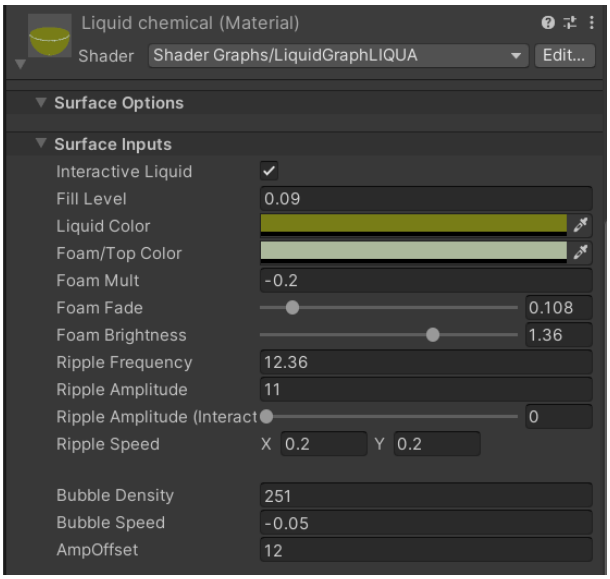
The two materials used for the Refractive setup are shown below. One should be setup with the checkbox **“FrontRender”** on and the other with the option off. The **“Interactive Liquid”** checkbox allows the materials to react to object motion using the **“Motion Detect”** script. The “**Fill Level**” defines how full the container will be with fluid and the **“Amp Offset”** defines the constant amplitude of the bubbles and motion effect, so can see the fluid move even if the object is not moving. This is additional offset to any motion induced by the **“Motion Detect”** script. The other parameters define the liquid bubble features like intensity and frequency and the ripples created by the motion amplitude and **“Amp Offset”** offset to that amplitude**.** The normal map and tiling – offset of it define the refractive properties of the material.



**Non Refractive material parameters (“LiquidLIQUA” Shader)**

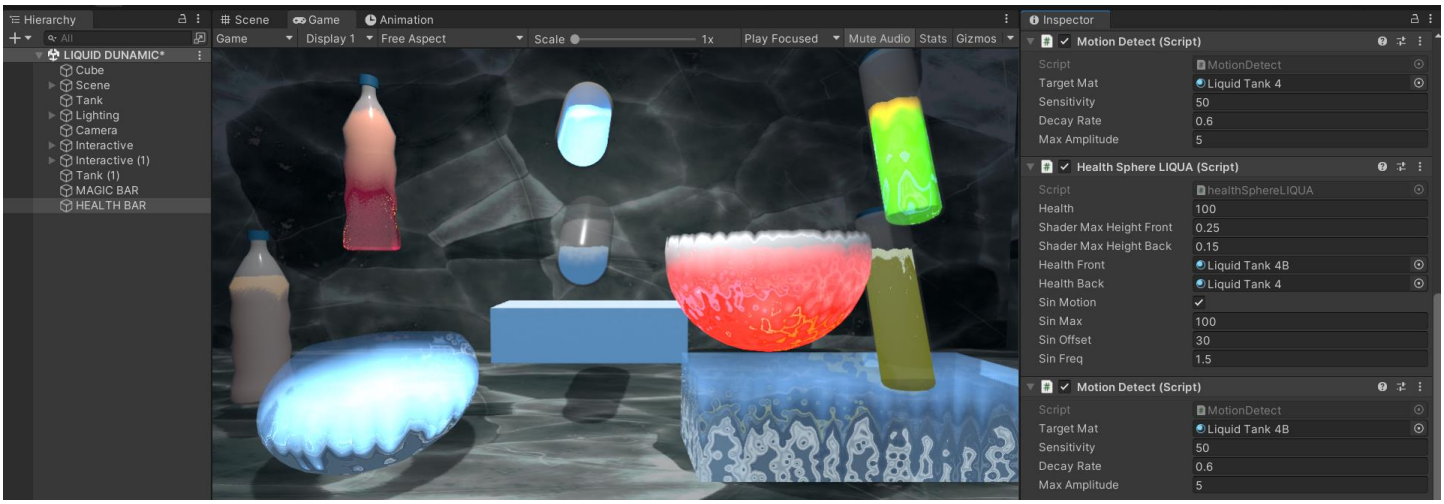
The material used for the Non-Refractive setup is shown below. The parameters are similar to the refractive one, without the refraction and normal map options and the front render option.



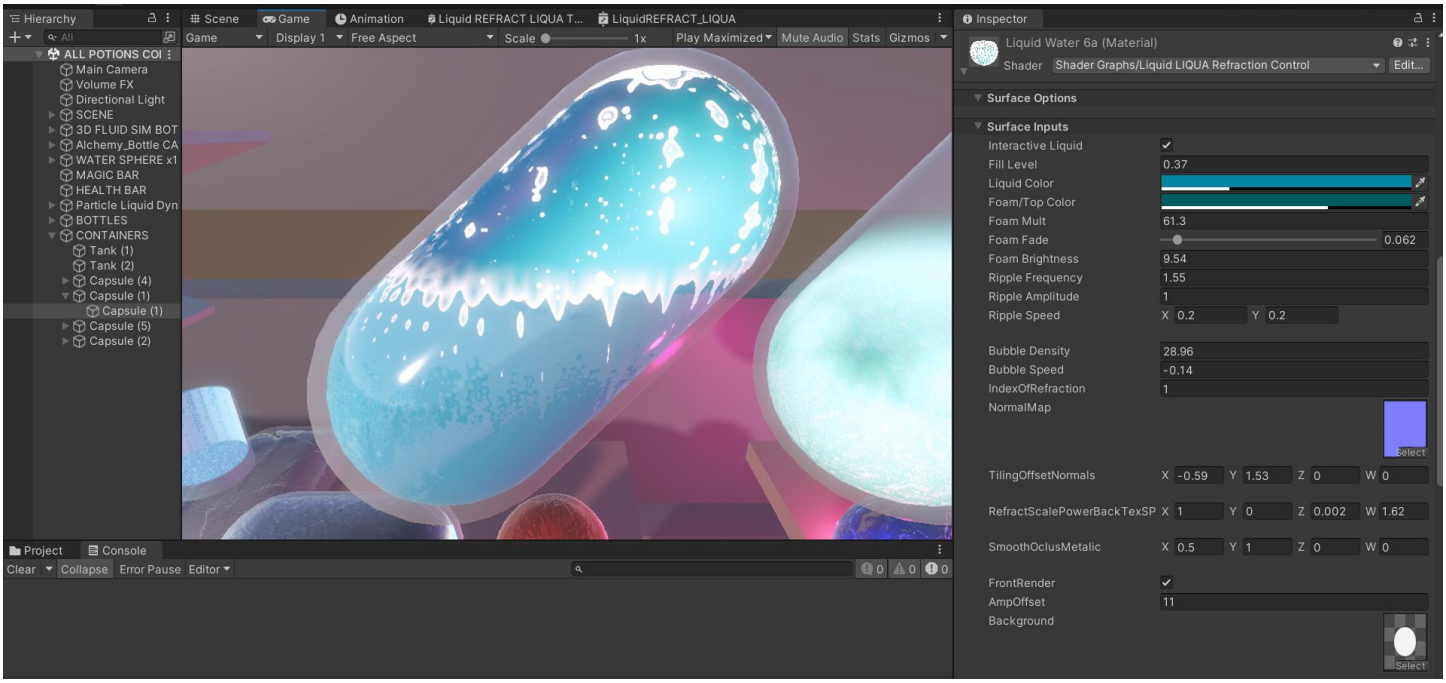

This effect is shader based and does not use any scripts for the fluid simulation other than the **“Motion Detect”** script to add motion amplitude.

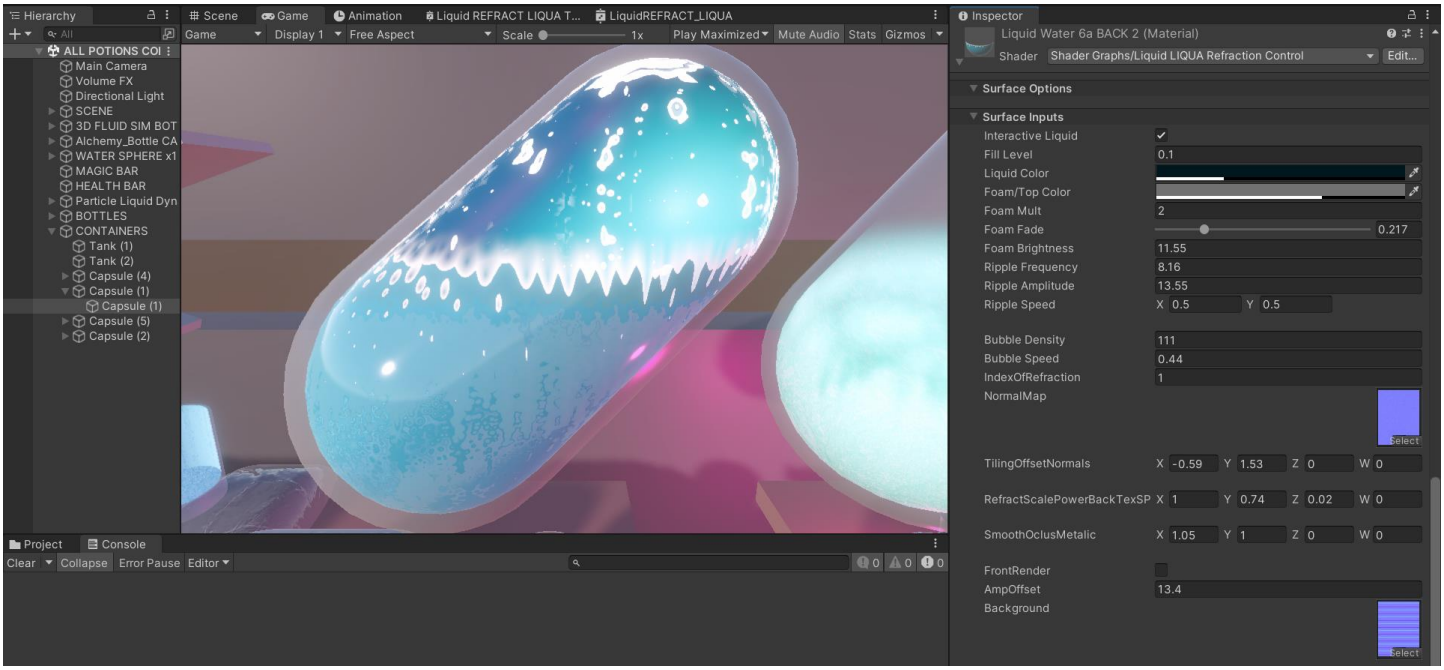
**Health bar sample script.**

The system contains a sample script that allows lower and raise the health bar sphere, as shown below, depending on the Health parameter. Also can enable a sample “Sin Motion” to auto lower and raise the bar to showcase the effect in motion. The script requires the materials to be referenced in the Health front and back material slots.



## 1A. Advanced Refractive shader (“Liquid LIQUA Refraction Control” Shader)

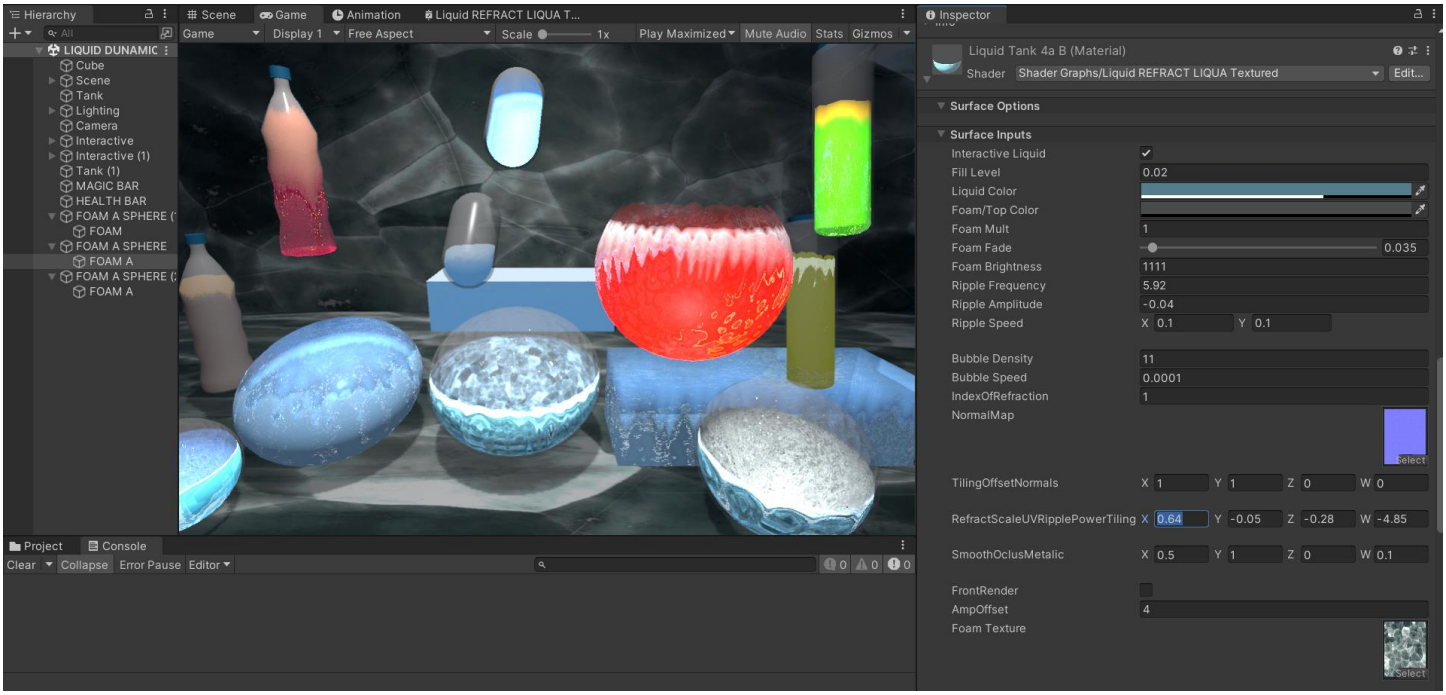


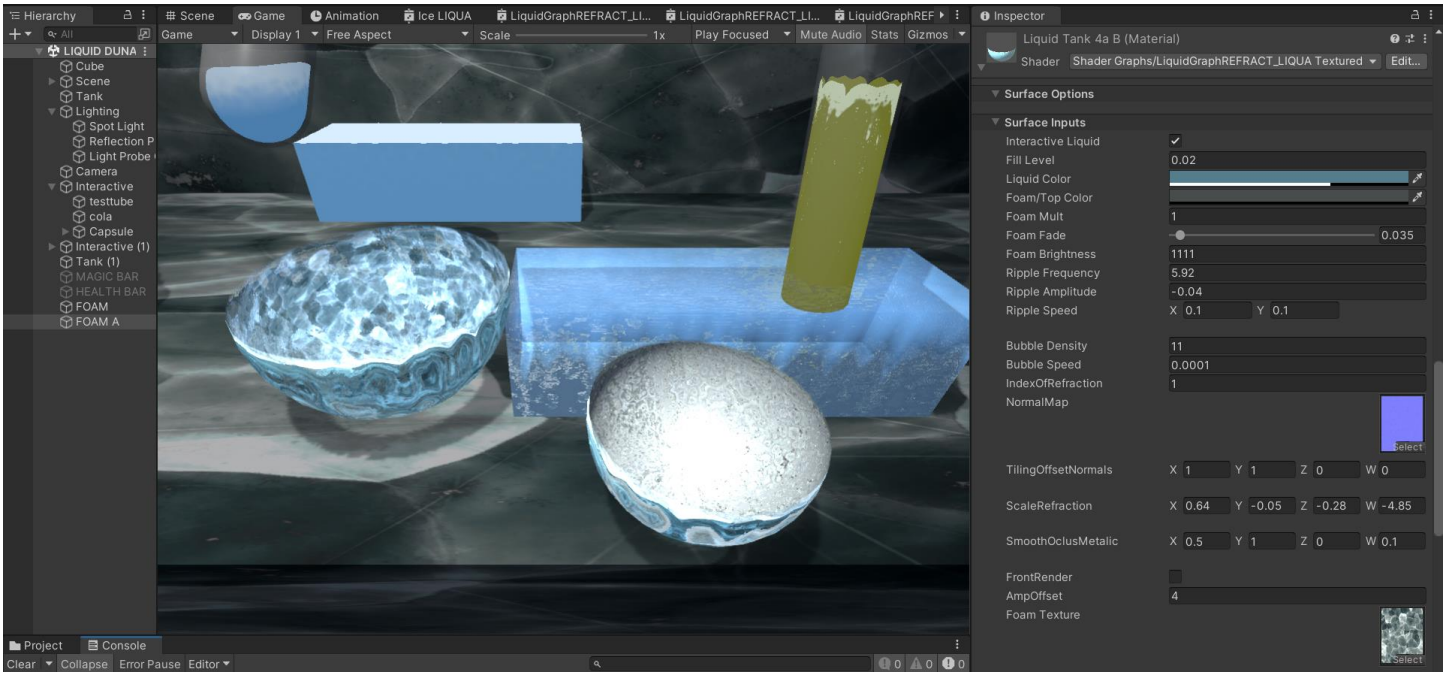




The advanced refraction shader can **control the refraction separately from the bubbles with a background texture** for finer control. This is done through the **“RefractScalePowerBackTexSP”** vector, **the X value** defines the scale of the refraction solution that grabs the background behind the liquid, the **Y value** defines the strength of the refraction, **the Z value** defines the scaling of the background custom texture defined in the **“Background”** slot and **the W value** defines the strength of the custom background texture.

**1B. Texture refractive shader** (”**Liquid REFRACT LIQUA Textured**” **Shader**)

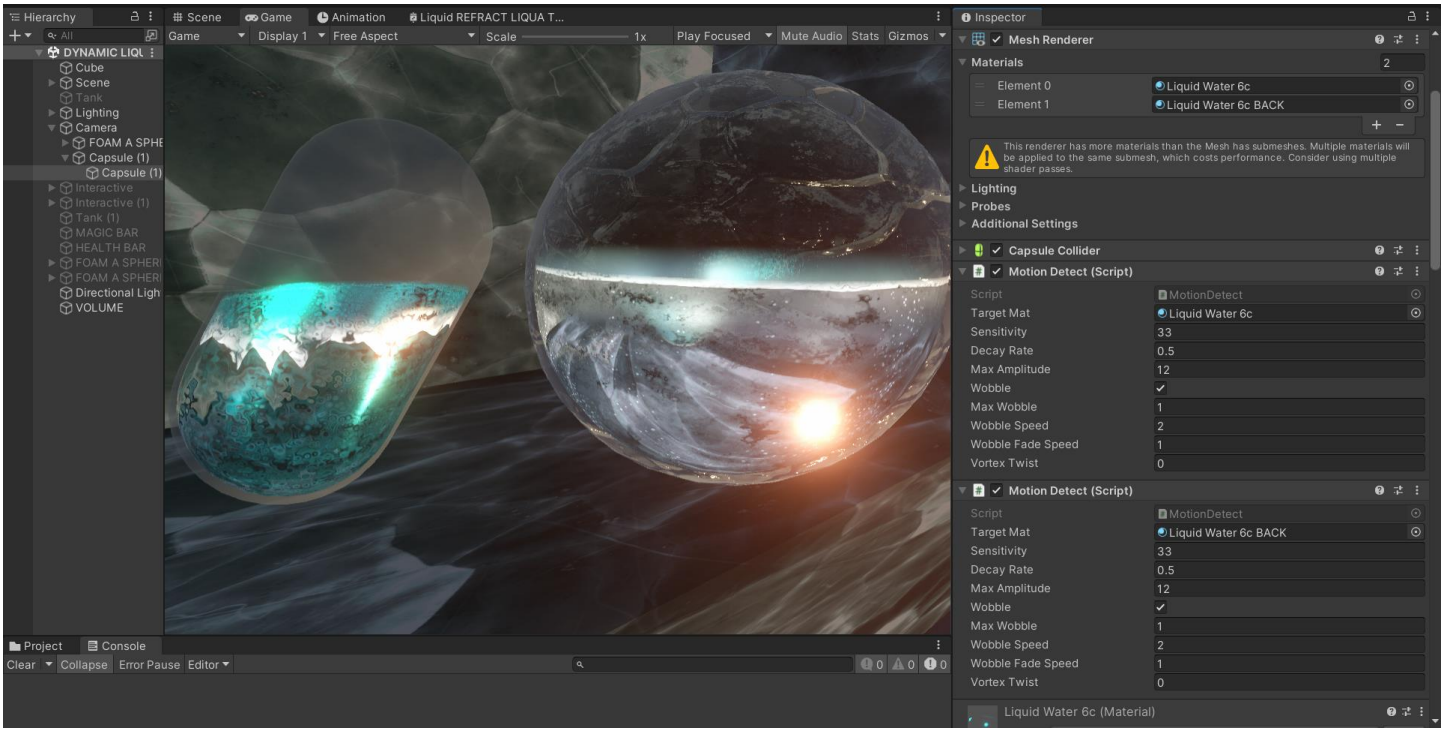
The advanced textured  shader can **control the texture of the liquid surface separately from the liquid side with a Foam Texture** for finer control. This is done through the **“RefractScaleUVRipplePowerTiling”** vector, **the X and Y values** defines the scale of the refraction solution that grabs the background behind the liquid, the **Z value** defines the strength of the Foam Texture and **the W value** defines the tiling of the Foam Texture.



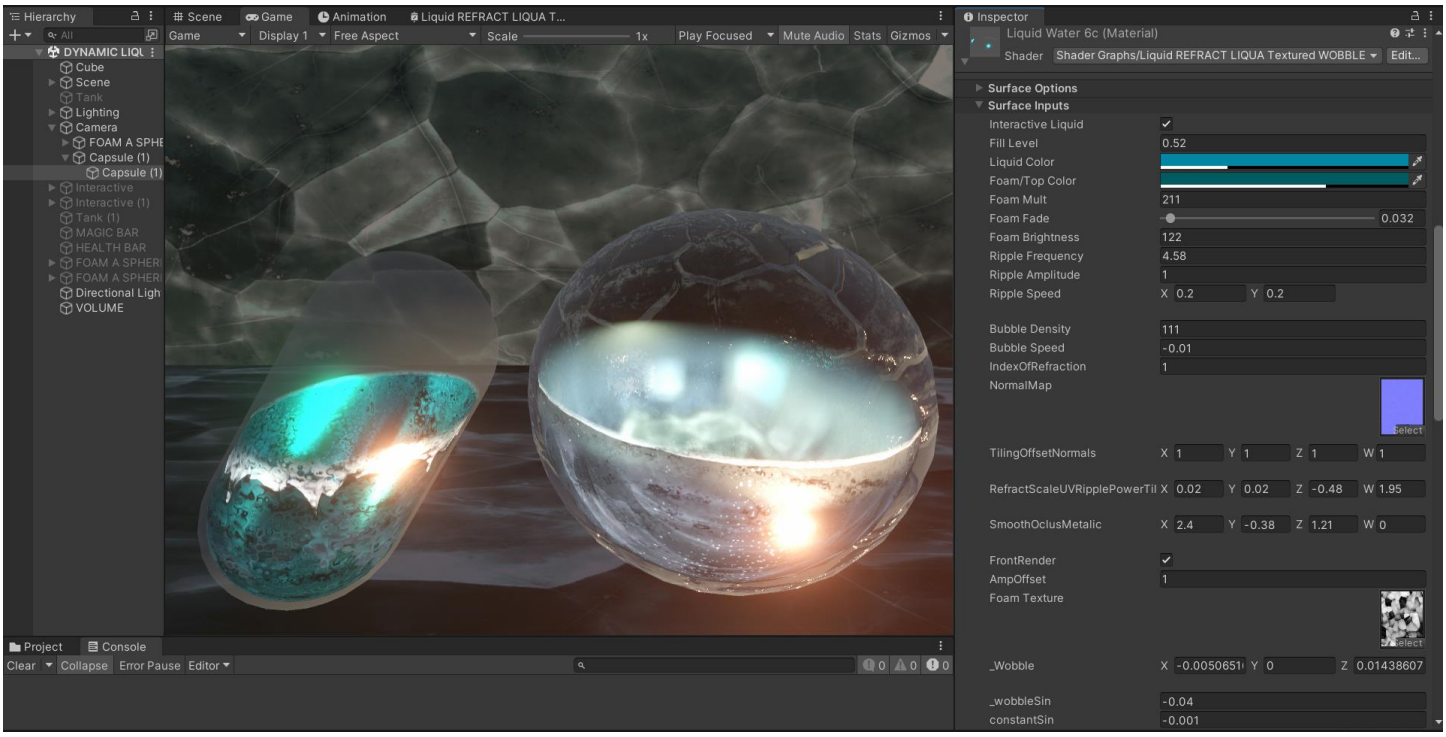



**1C. Texture refractive shader with Wobble** (”**Liquid REFRACT LIQUA Textured WOBBLE**” **Shader**)

The “**Motion Detect**” script integrates a liquid wobble system that makes water react to object motion. It is activated by the “**Wobble**” checkbox and has two main parameters, the **max wobble** power and **wobble speed**. The **wobble fade** defines how fast the effect will fade after motion stops. The **Vortex Twist** is an experimental feature and currently adds a simple sine motion to the overall effect.



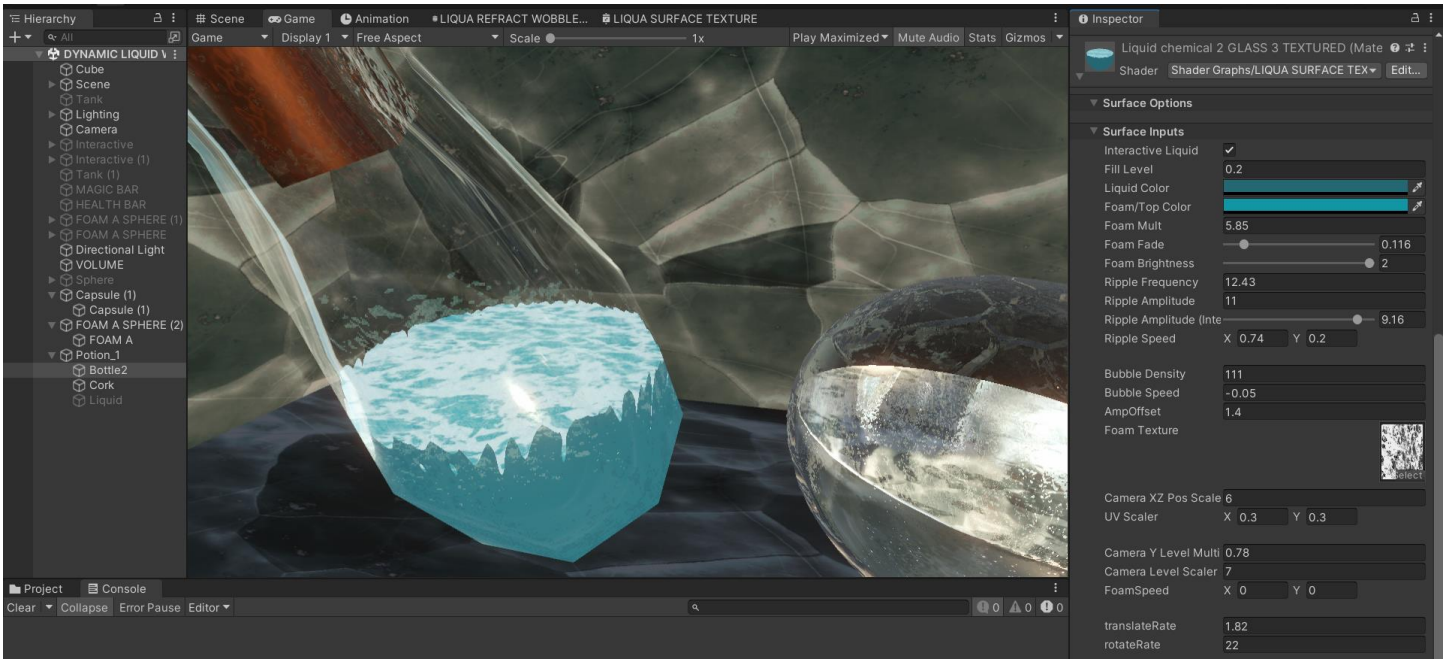

The system requires the ”**Liquid REFRACT LIQUA Textured WOBBLE**” **Shader** (Or its “REFL” variant that adds the rippling to the Reflection Probe reflected image), and is important to set the “**WobbleSin**” parameter to a desired value, this value will add a sinusoid ripple effect additionally to the liquid lateral motion. The “**ConstantSin**” is an experimental parameter and currently adds an extra sine motion to the overall effect.

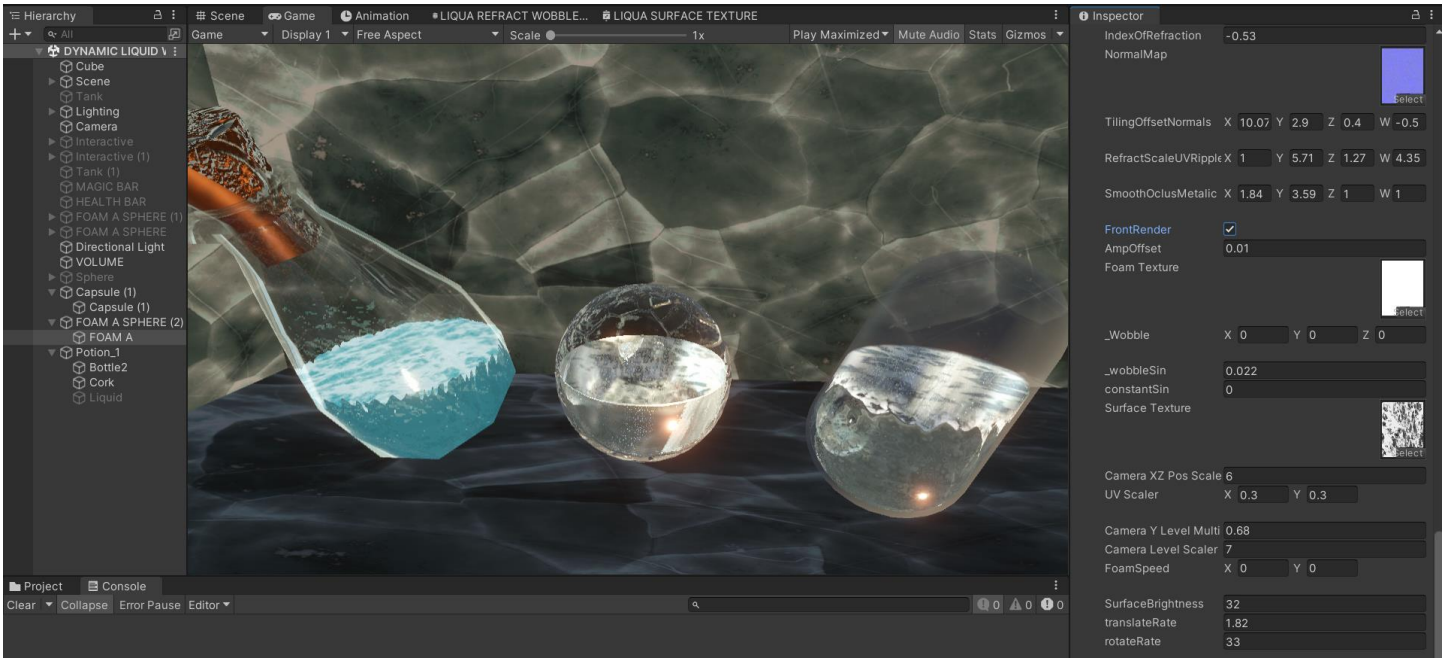


**1D. Textured Liquid Surface Option with and without Wobble**
**(”LIQUA SURFACE TEXTURE” and “LIQUA REFRACT WOBBLE REFL SURFACE TEXTURE” Shaders)**

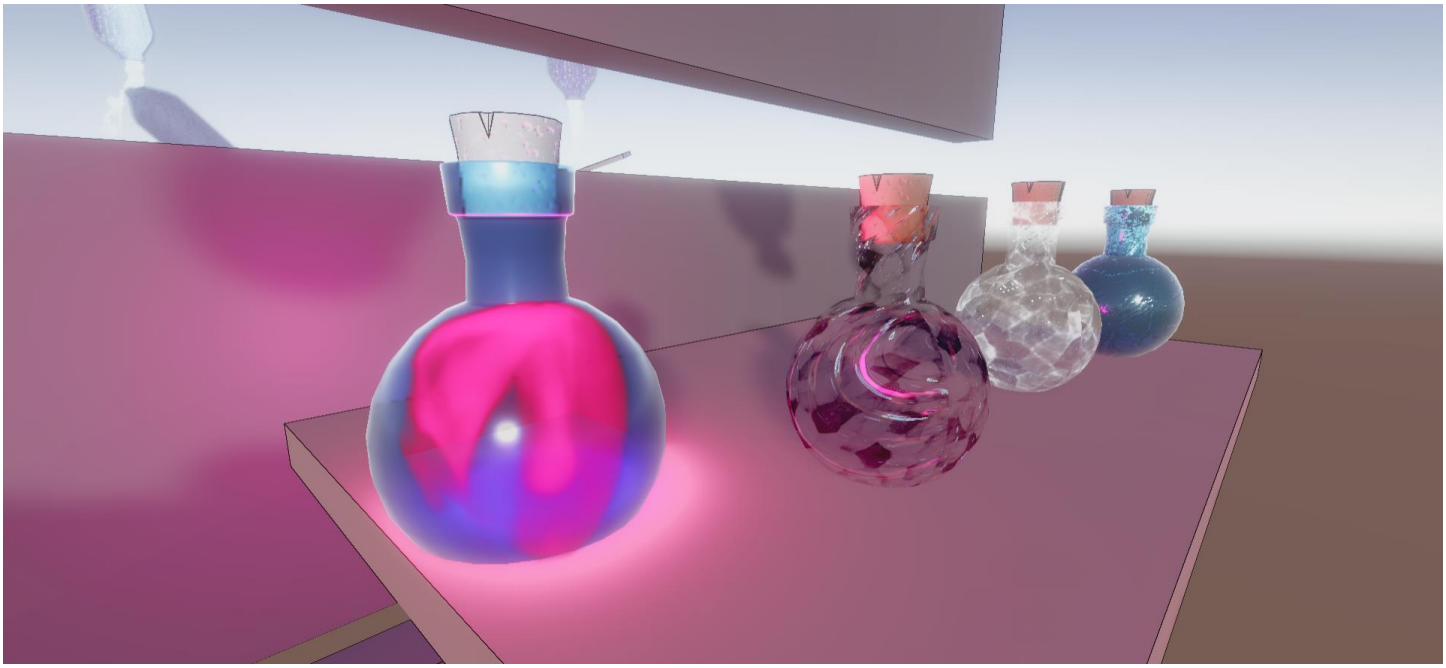
The new shaders in v1.1 allow for the application and control of a user defined texture on the liquid surface, using raytracing to place the texture on a virtual liquid surface space, independent of the bottle mesh.

The “**Camera XZ Pos scale**” scales the motion of the liquid when camera moves horizontally. The “**UVscaler**” scales the surface texture. The “**Camera Y Level Multiplier**” variable is not used. The “**Camera Level Scaler**” controls the motion-scaling of the liquid when the camera moves vertically. The “**Foam Speed**” adds a XZ motion to the foam. The “**translate Rate**” controls the motion of the liquid as the bottle moves around the scene. The “**rotate Rate**” controls the rotation of the liquid.

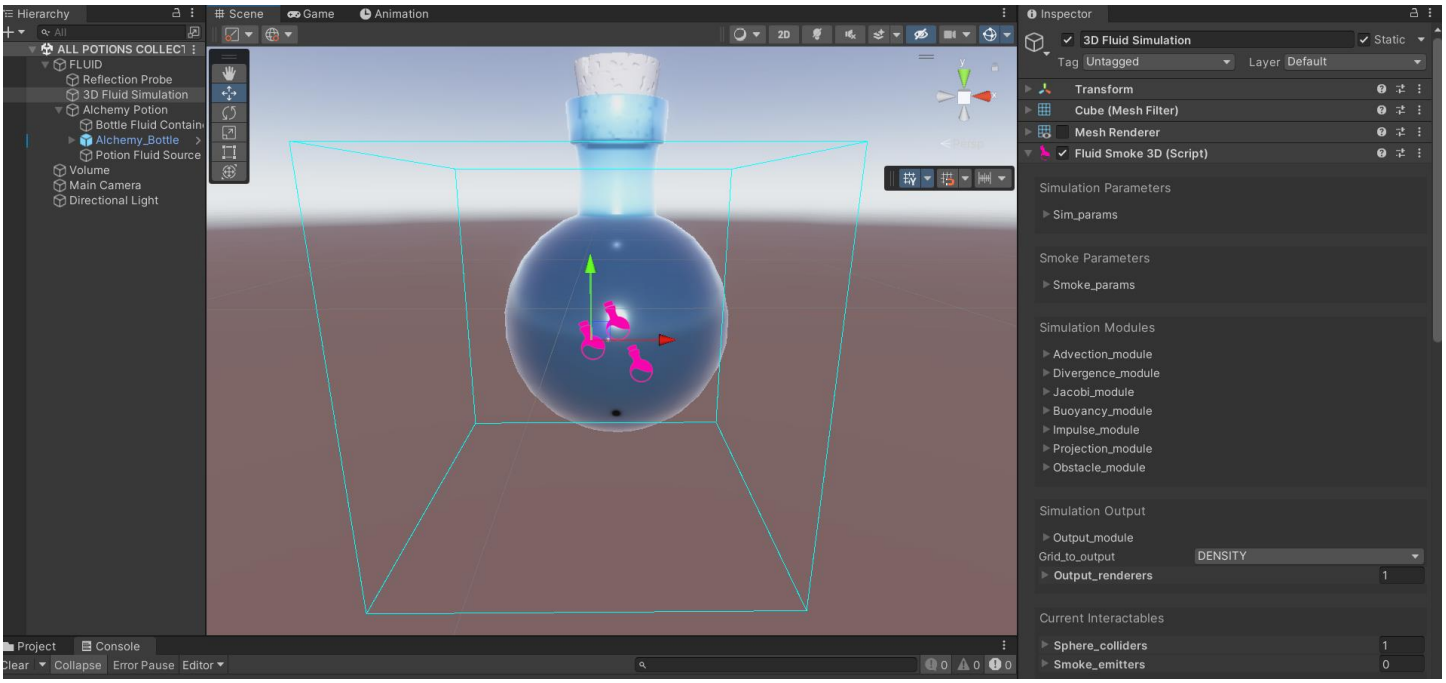


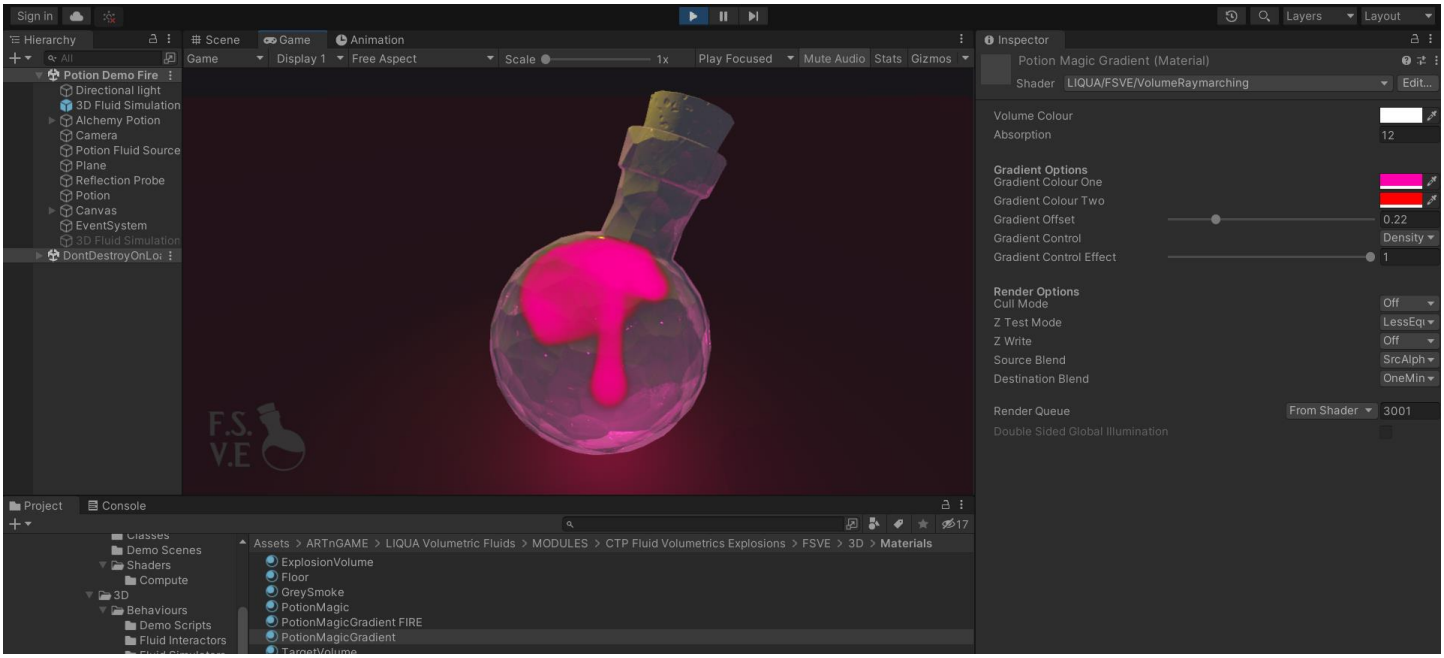
**2. 3D Fluid simulation setup.**



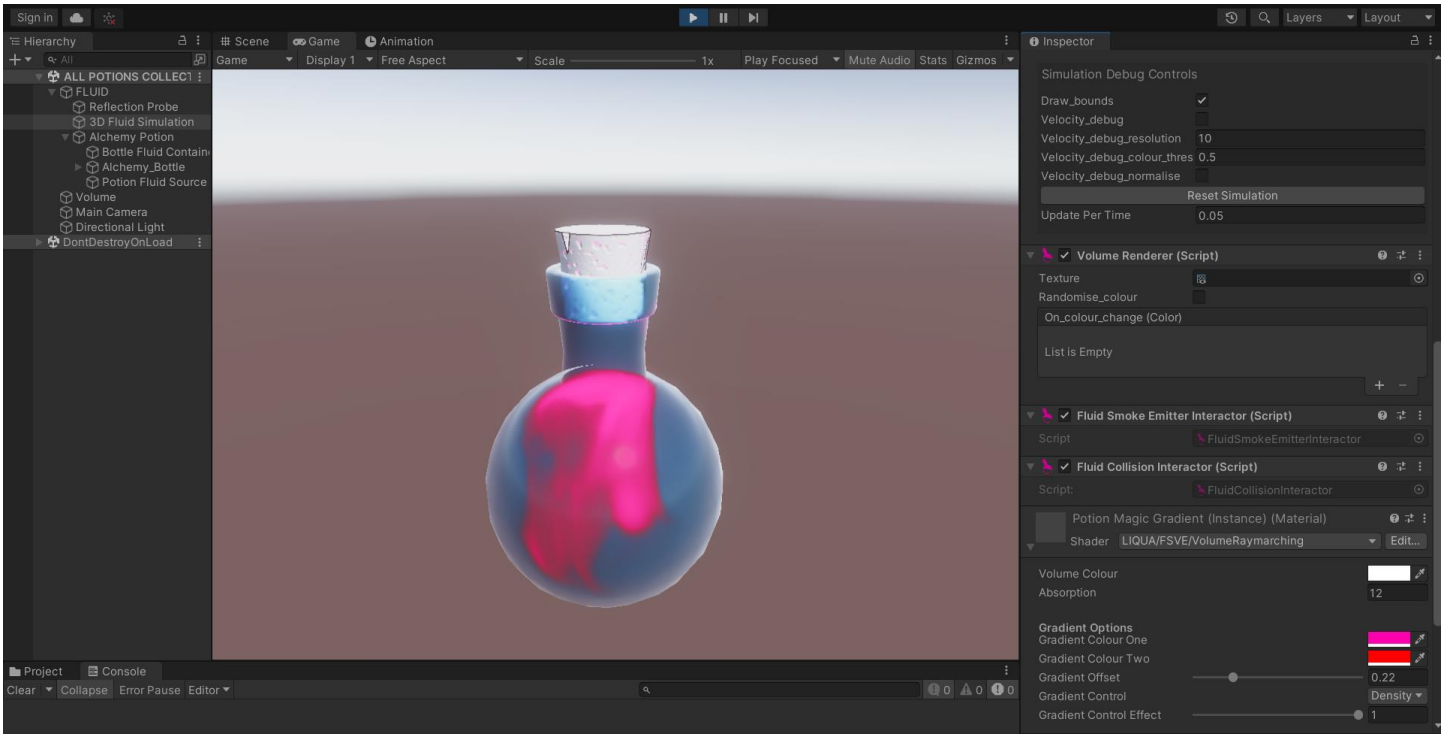
The core of the LIQUA **3D Fluid simulated liquids** inside a container system is the **“Fluid Smoke 3D”** script, which simulates a smoke based fluid. This should be attached to a box object and the Mesh Renderer should be disabled and enabled at game start using the “**enableWithDelayLIQUA**” script, where the renderer must be referenced in the **“RendererA”** slot.



The mesh renderer must have the following material – shader.

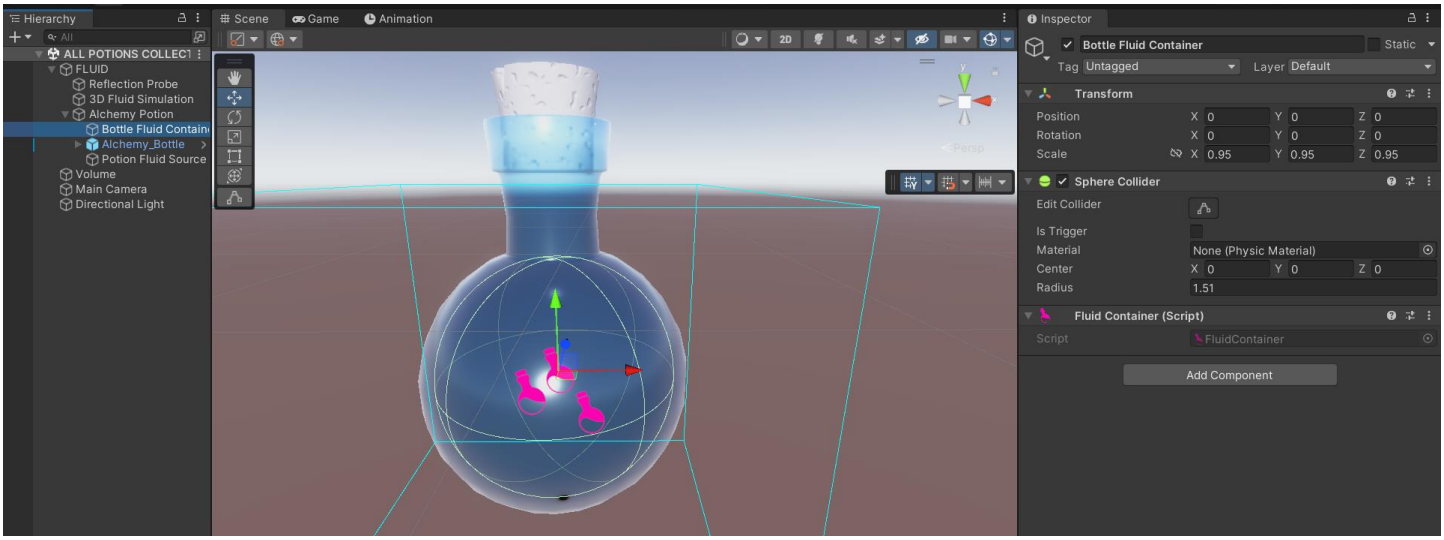



The other required module for the fluid rendering is the **“Volume Renderer”** script.

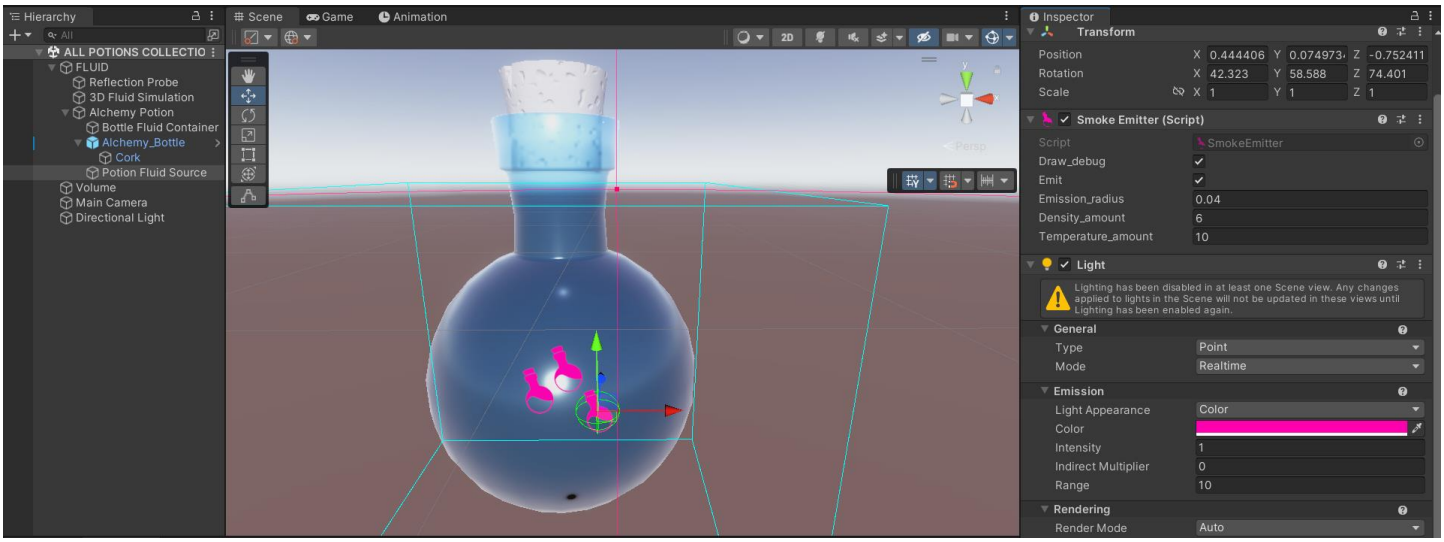



The **“Fluid Smoke Emitter Interactor”** and **“Fluid Collisions Interactor”** scripts are responsible for the emission of the fluid and the interaction with the fluid respectively.

In order to contain the fluid from the box space to a spherical space, a sphere collider must be used together with the **“Fluid Container”** script as shown below.



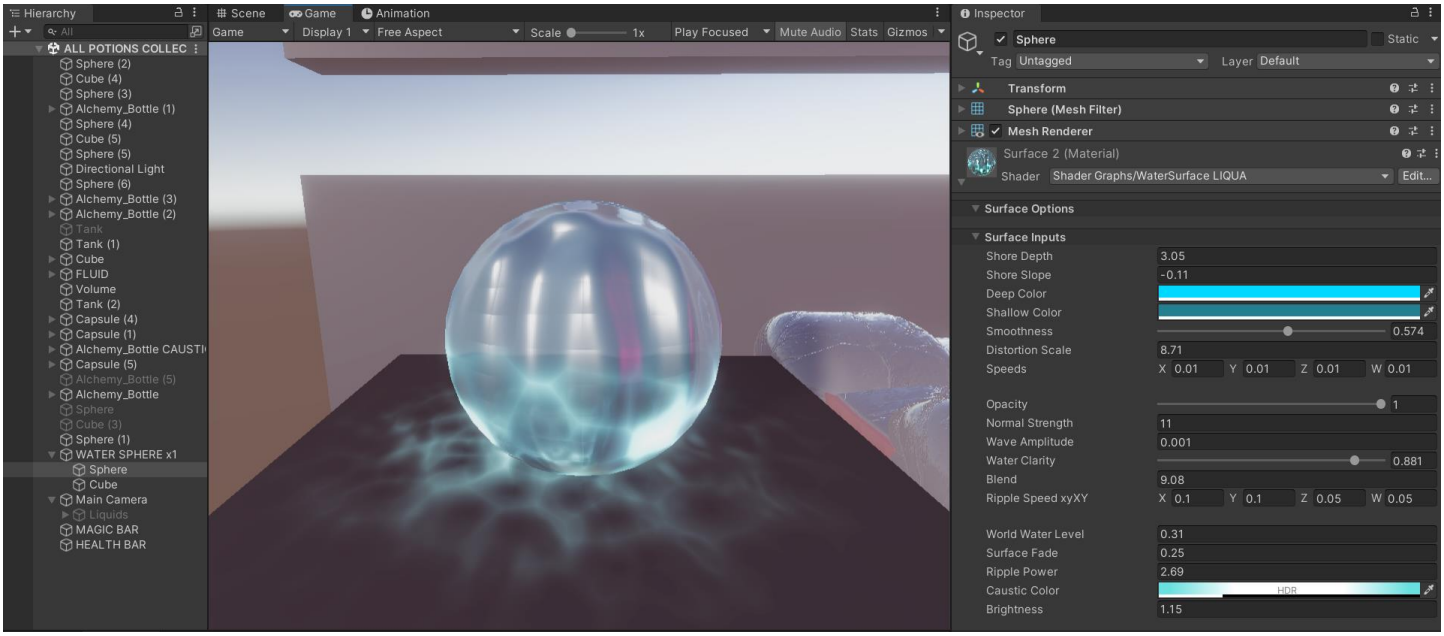

In order to emit from a specific place, use the “Smoke Emitter” script on a Transform positioned in the desired place.



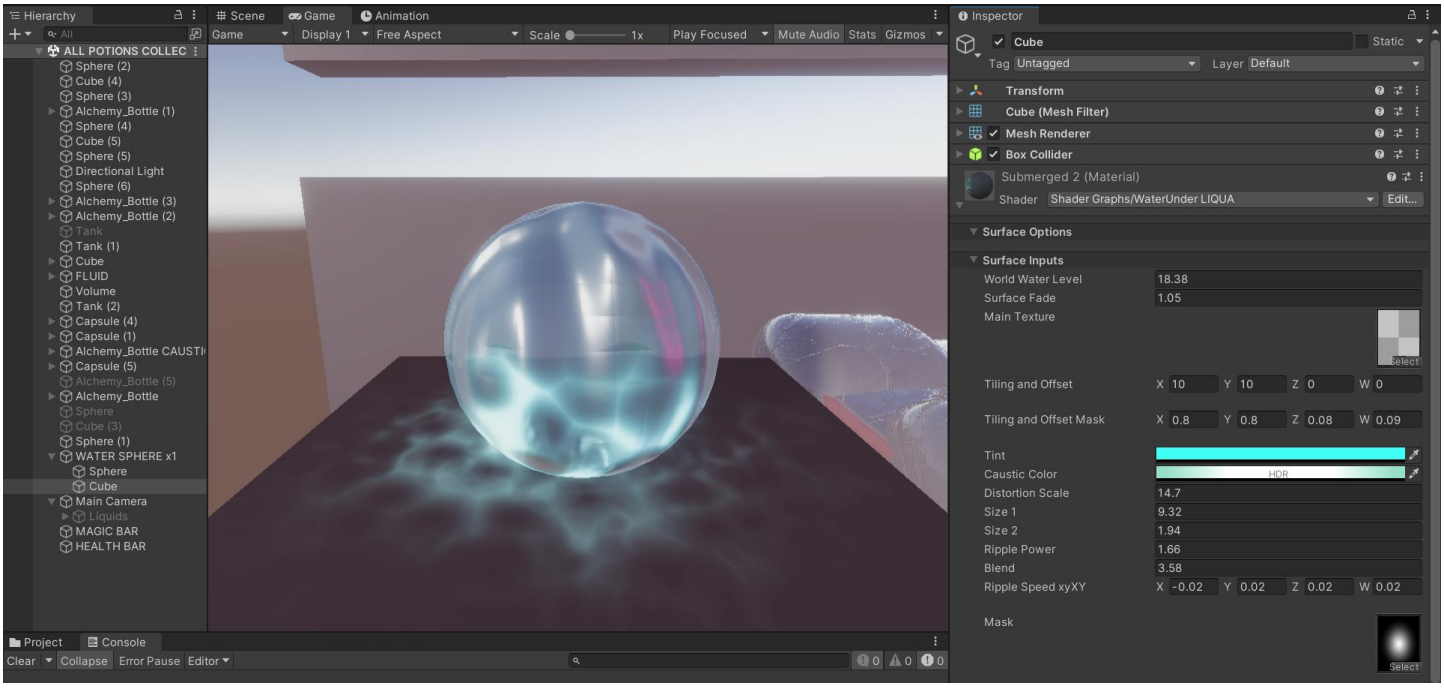

**NOTE:** There is also **a Fire variant of the Smoke material and scripts above**, this is experimental and currently replicates the smoke functionality and is planned to be extended to emulate fire inside the containers in next versions.

### 3. Water shader setup.

The water shader is presented in the following image.

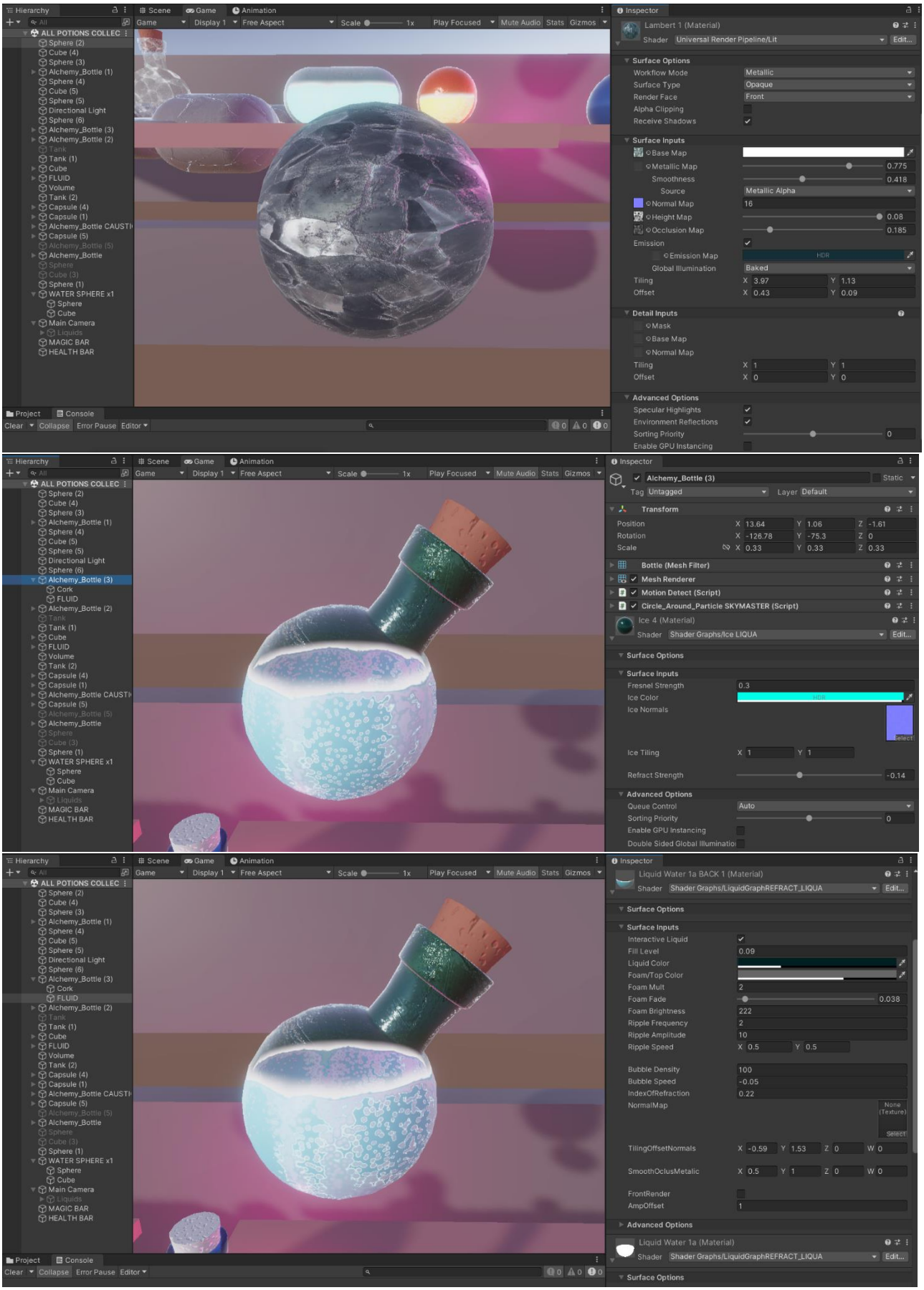



Floor caustics



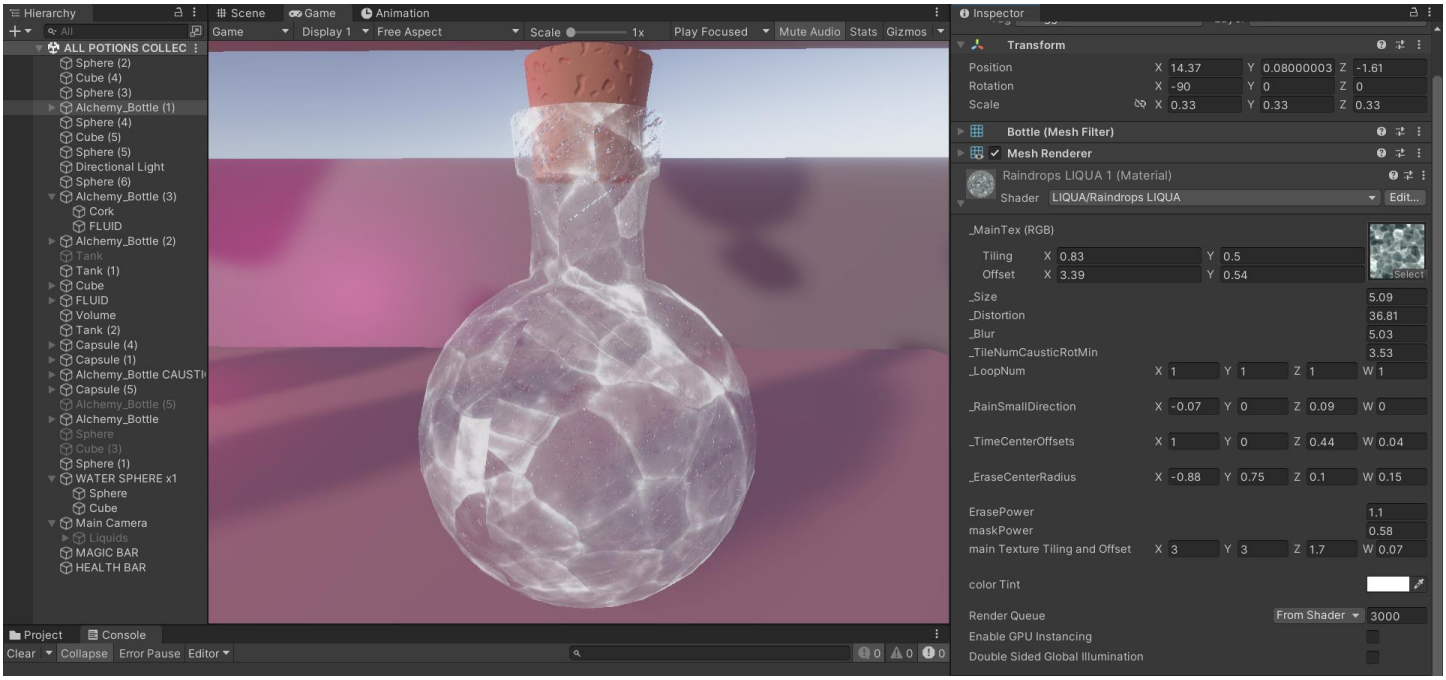
**4. Ice shader setup.**

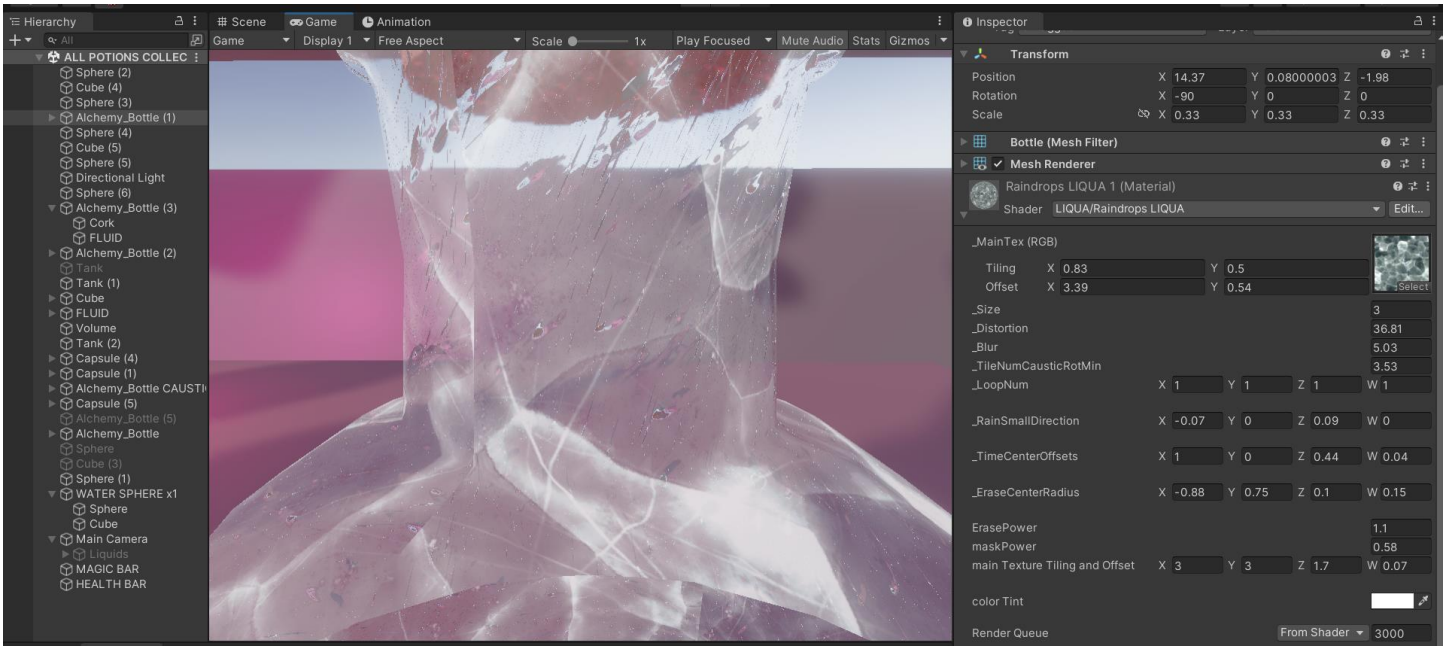
The ice shader is presented in the following image.



**5. Raindrops shader setup.**

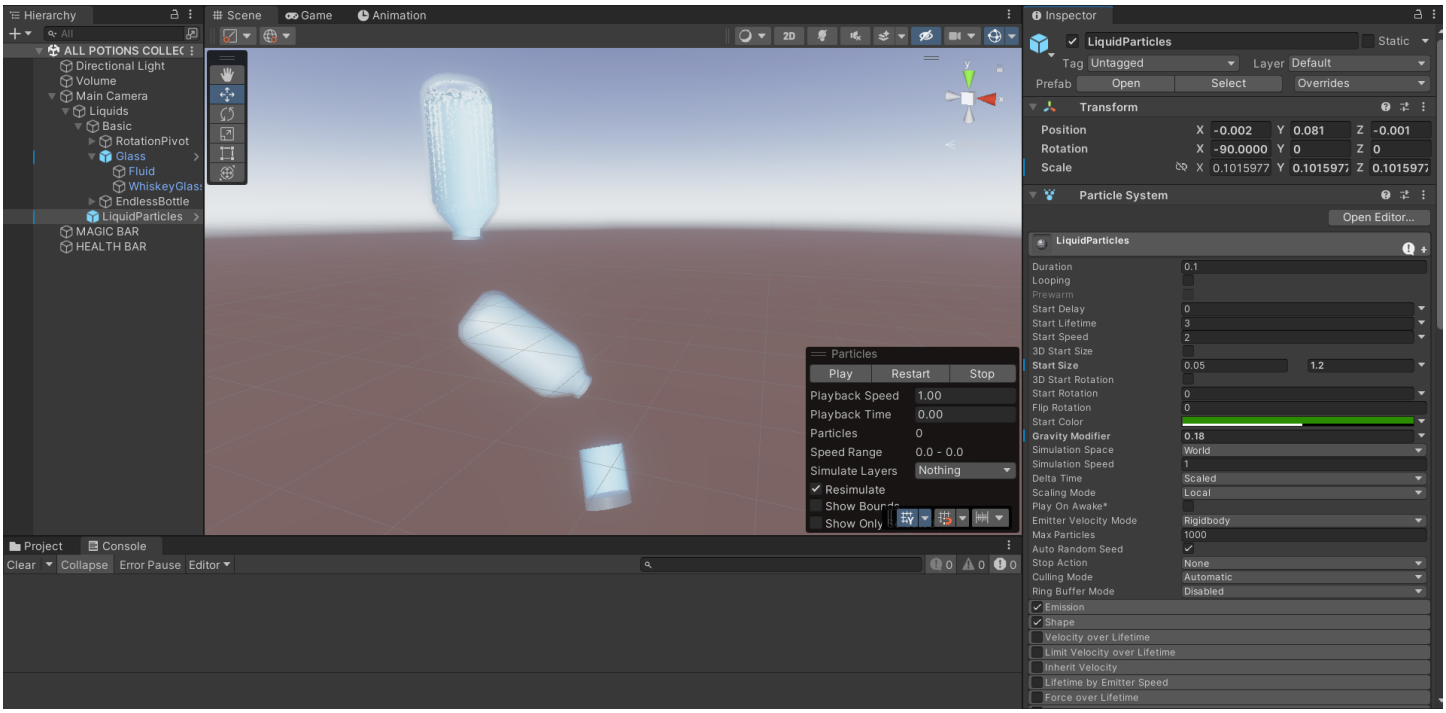
The dynamic infinite detail raindrops shader is presented below.

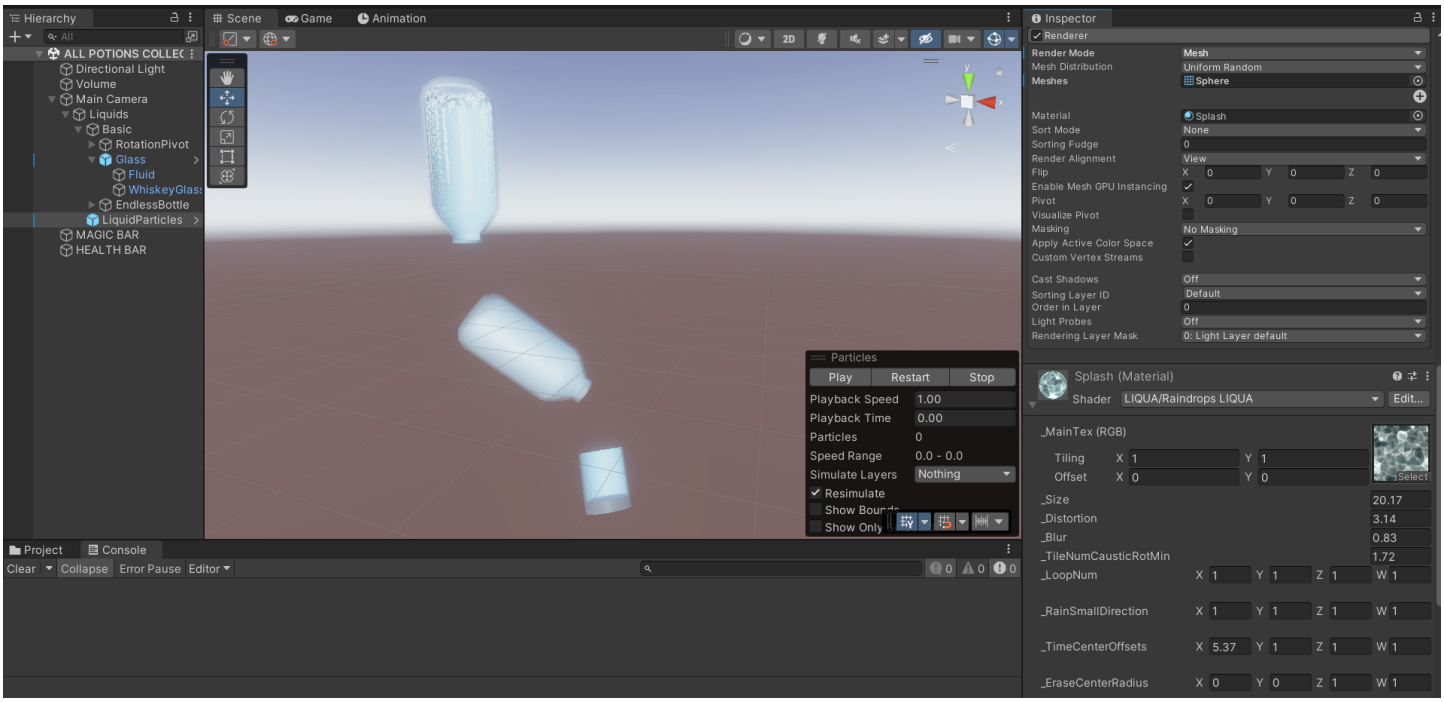


**6. Dynamic particles setup.**

The dynamic particles system for filling bottles is presented in this section. The system uses a particle system as shown below to render the fluid particles as drop from one container to another.

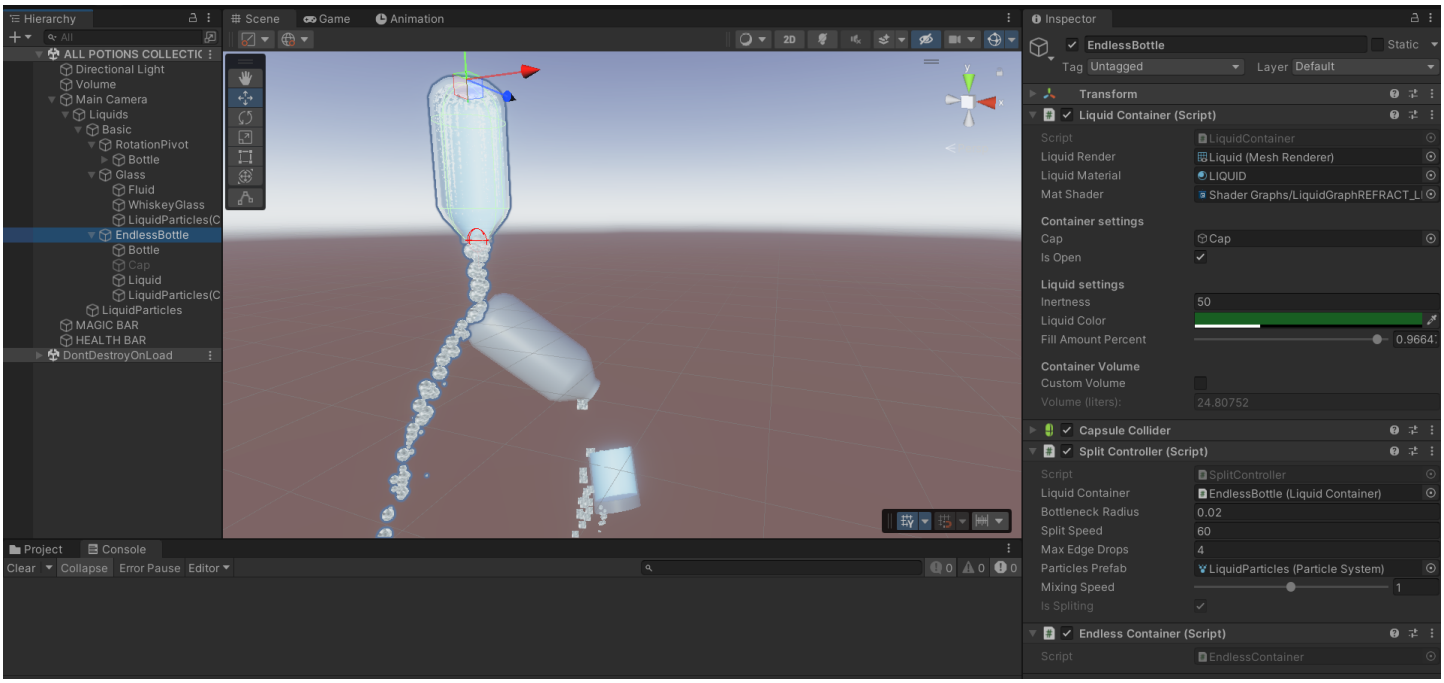



The particle uses a refractive material and sphere particles to represent the water particles.

**Water source – endless bottle setup**

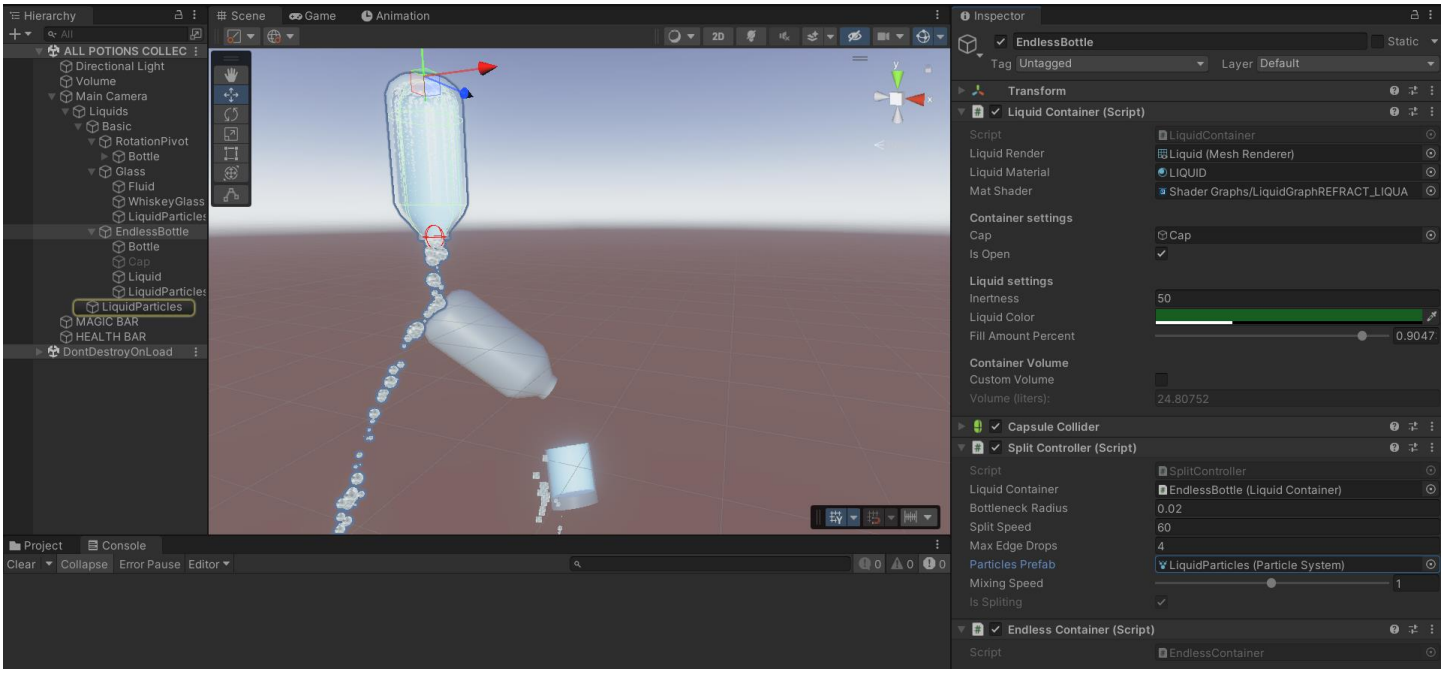
The system starts from an endless source bottle that cast the water to other containers.




For that purpose the **“Endless Container”** script must be used that keeps the fill of the **“Liquid container”** script to 100 percent. The **“Liquid container”** handles the container and water material properties. The “**Split Controller**” script calculates liquids splitting effect and transfer to other liquid containers. The system also requires a collider to sense when water particles fall on a container opening and start filling it accordingly.

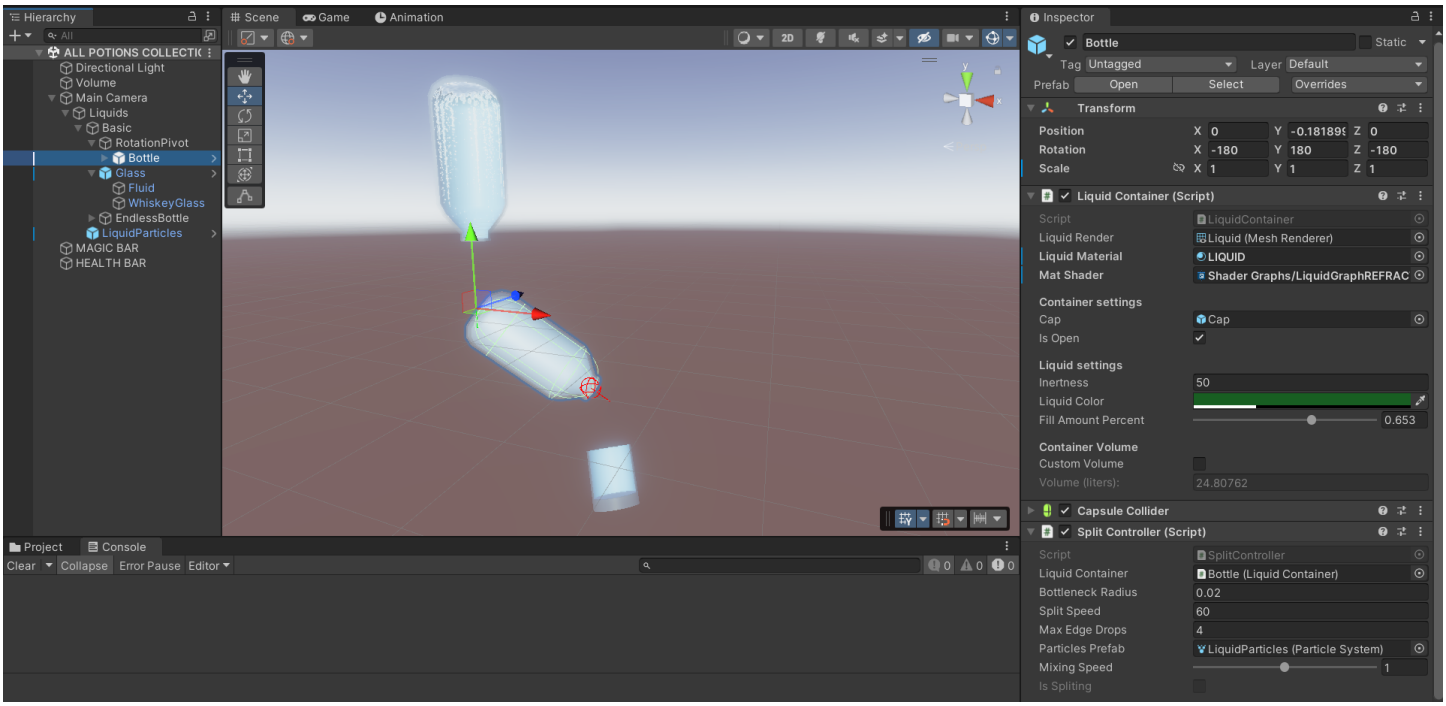
**Particle setup**

The water particle system must be referenced in the **“Particles Prefab”** slot in the “**Split Controller**” script.



**Rotating bottle – water filling and removal**

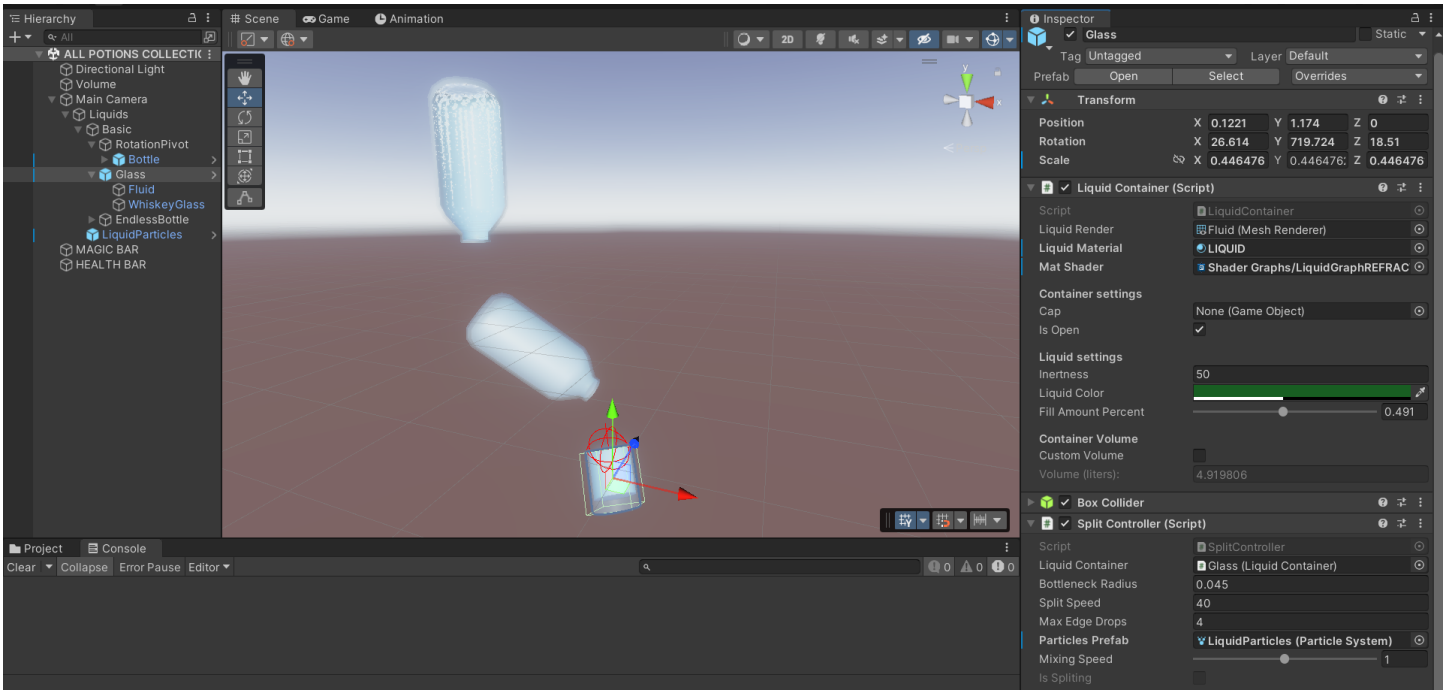
The demo implements a middle container, which fills from the endless one and removes its content periodically.

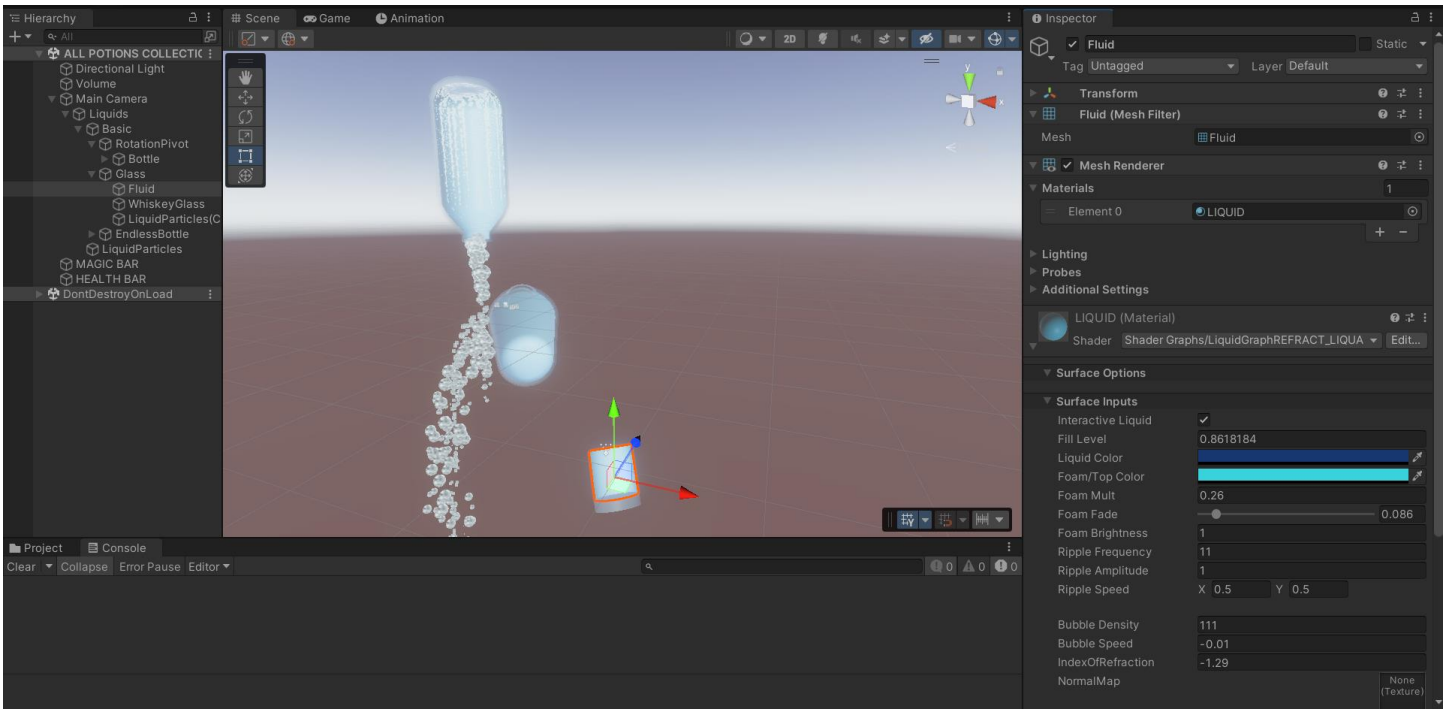



The fill will begin when the collider touches the water particles near the bottle opening. Then on rotation will start spill the liquid and empty its fill reducing the fill amount in the **“Liquid container”** script.

**Glass receiver**

Finally a glass receives the dropped liquid and fills its content.




Hierarchy
Scene
Game
Animation
Inspector
ALL POTIONS COLLECTIC
Directional Light
Volume
Main Camera
Liquids
Basic
RotationPivot
Bottle
Glass
Fluid
WhiskeyGlass
LiquidParticles(C
EndlessBottle
LiquidParticles
MAGIC BAR
HEALTH BAR
DontDestroyOnLoad
Fluid
Static
Tag Untagged
Layer Default
Transform
Fluid (Mesh Filter)
Mesh Fluid
Mesh Renderer
Materials 1
Element 0 LIQUID
Lighting
Probes
Additional Settings
LIQUID (Material)
Shader Shader Graphs/LiquidGraphREFRACT_LIQUA
Edit...
Surface Options
Surface Inputs
Interactive Liquid
Fill Level 0.8618184
Liquid Color
Foam/Top Color
Foam Mult 0.26
Foam Fade 0.086
Foam Brightness 1
Ripple Frequency 11
Ripple Amplitude 1
Ripple Speed X 0.5 Y 0.5
Bubble Density 111
Bubble Speed -0.01
IndexOfRefraction -1.29
NormalMap
None (Texture)
Project
Console
Clear
Collapse
Error Pause
Editor