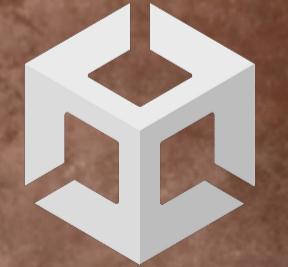




MADE WITH



PITCH VIDEO

<https://www.youtube.com/watch?v=Rv3MXZEU47U>

#VRGame #Simulator

Bartender Simulator is a VR game that simulates bartending. Players can experience both the fun of bartending and how the world of alcoholics is.

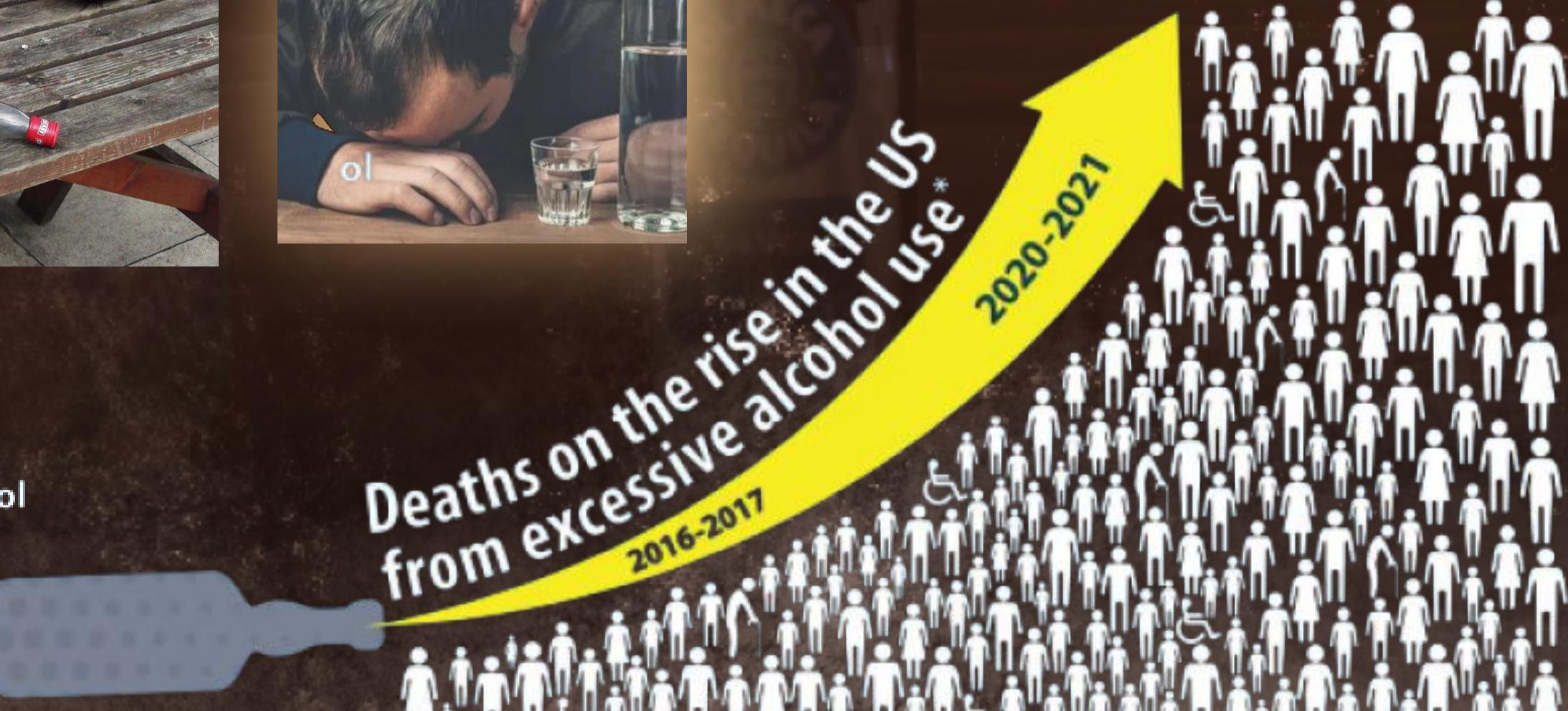
BARTENDER  
SIMULATOR

# INSPIRATION & RESEARCH

Wine has a long history dating back to the Neolithic Age. Since ancient times, it has been used as a ritual object and banquet drink. Nowadays, many people use alcoholic beverages to relieve worries, release stress or seek excitement. The number of alcoholics has been increasing year by year. According to a survey released by WHO in 2019, about 2.6 million people died worldwide due to alcohol consumption. Of these, 1.6 million died from non-communicable diseases, 700,000 from injuries and 300,000 from communicable diseases. This shows that alcoholism has serious negative consequences for alcoholics and others.



[www.cdc.gov/alcohol](http://www.cdc.gov/alcohol)



\*178,000 deaths each year in the US during 2020-2021, compared to 138,000 deaths each year during 2016-2017.

# BACKGROUND

Alcoholism can cause a lot of harm to both the alcoholic and those around them.

Hence, a game is needed to warn and prevent such things. This game uses the theme of mixing drinks to let players experience the interactive fun. Then, after drinking, hallucinogenic effects are produced to warn players to drink in moderation instead of indulging in alcohol, lest they bring irreversible consequences to others and themselves.

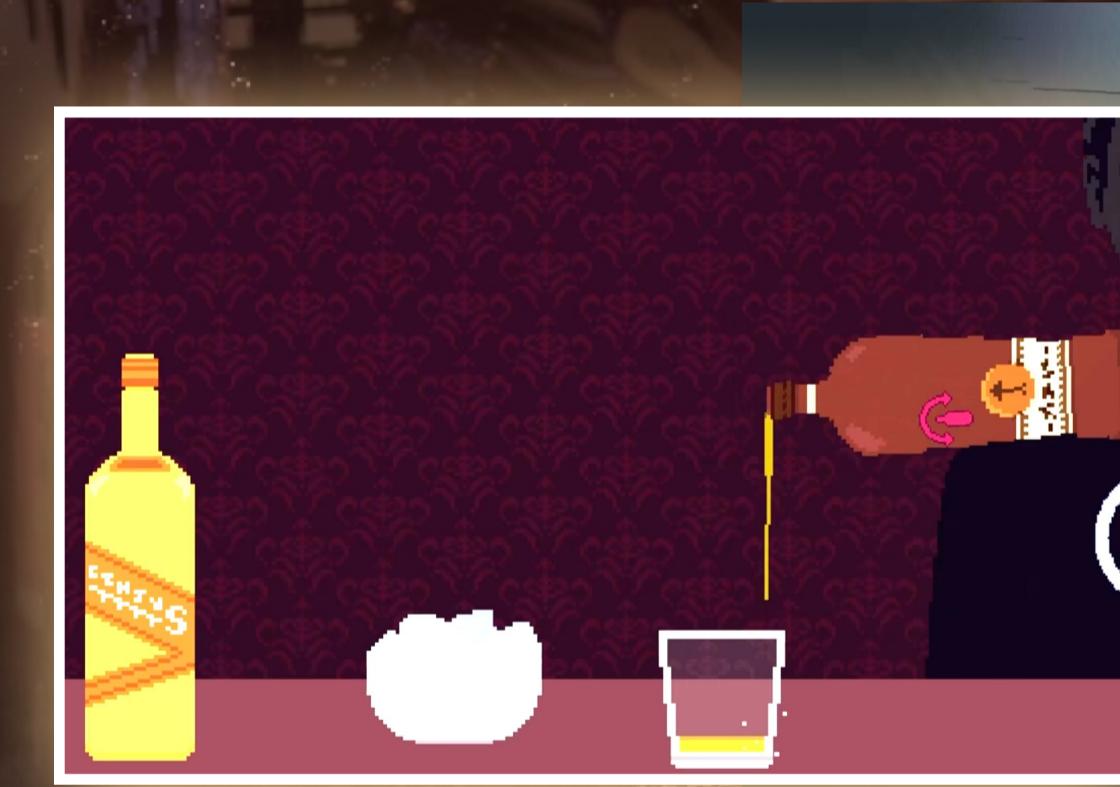
# ART REFERENCE



Cooking Simulator



Half-Life: Alyx



RELEVANT GAMES



This game references the bartending gameplay in **Red String Club**. In it, bartender Donovan mixes drinks like Greedy and Arrogant ones to get info about Supercontinent executives using their emotions. Executives drink and reveal info, reflecting their human emotions and psychology.  
Player chooses ingredients according to recipe, affecting customer's state of mind. Bartending interaction is interesting and resulting drinks are beautiful, inspiring me to make a bartending game.

# GAME FLOW



Strawberry  
Gin  
Tonic



Prepare Highball cups

Place ice cube

Pour 1 jigger of  
strawberry gin

Pour 2 jigger of tonic  
water

Agitation

Margarita



Prepare Margarita cups

Prepare Shakers

Dip the rim of the  
glass in salt

Place ice cube

Pour 1 jigger of  
Tequila

Pour 1 jigger of  
Cointreau

Juice of half a lime

Shake  
Strainer filter and fill  
into cups

Garnish with lime slices

Flame  
Tequila



Prepare Shot cups

Pour 1 jigger of Tequila

Cover with a slice  
of lime

Take a spoonful of ground  
coffee and put it on (bar  
spoon)

Top with Spicy  
minced millet

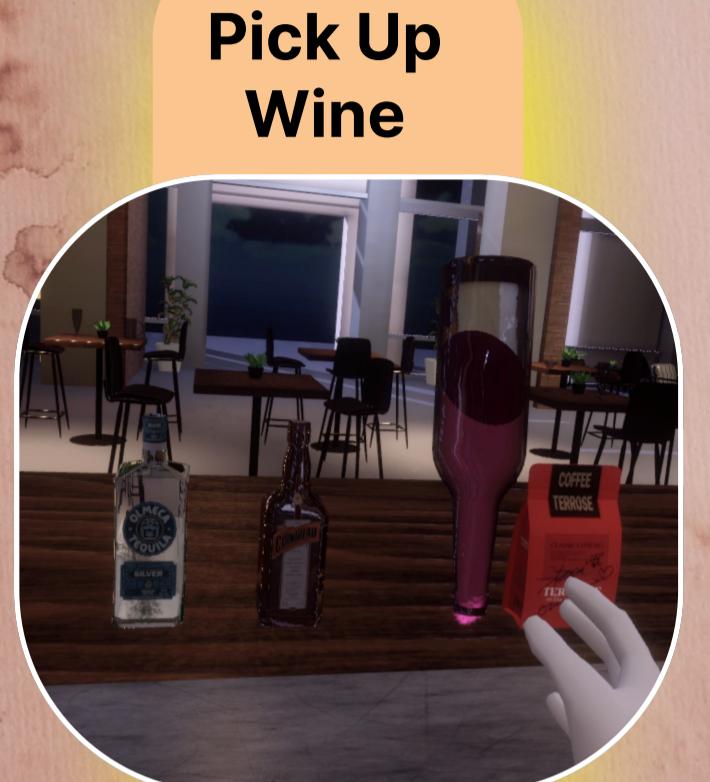
Pour a few drops of  
Tequila

Strike a match

Light the ground  
coffee



# MENU DEVELOP



Pick Up Wine



Agitation



Filtration



Shake



Strike A Match



# DIRECT GRAB

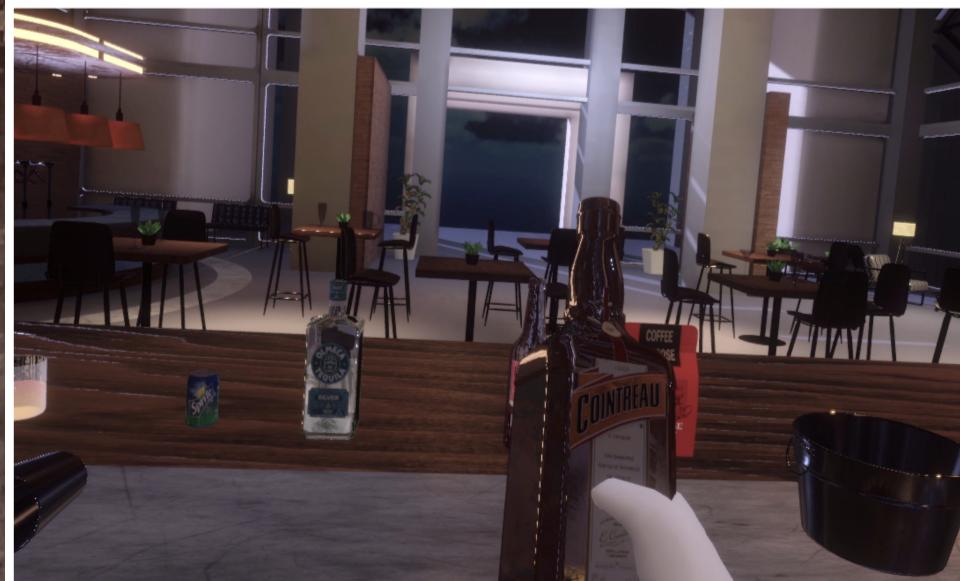
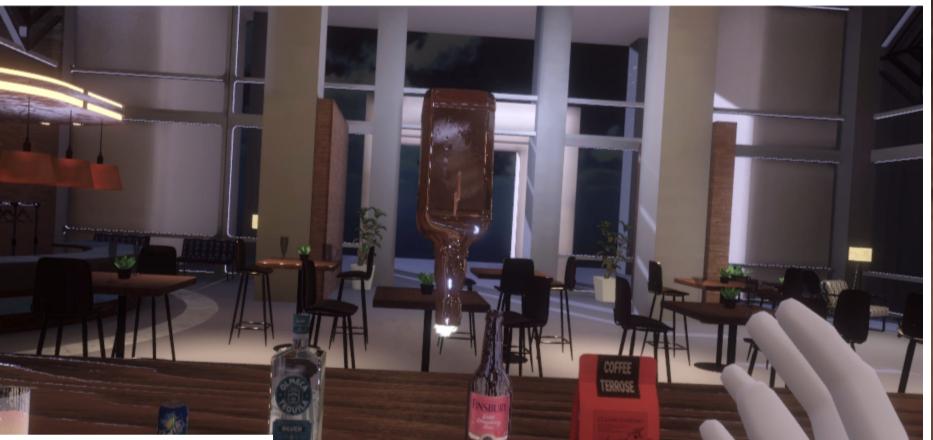
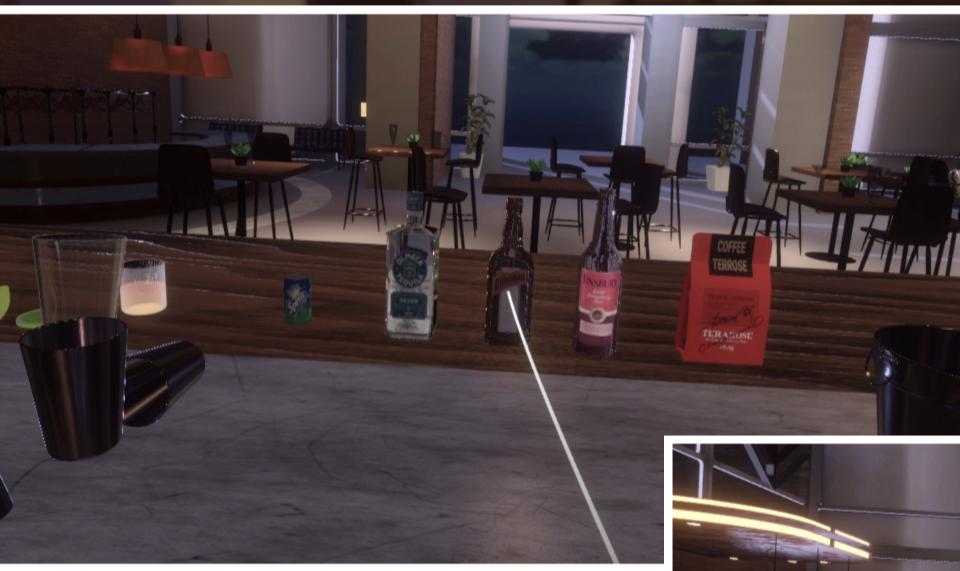
When there is an overlap between the player's hand and an interactable object, the player can press the GrabTrigger in the VR grip to pick up the object. Using this, players can pick up bottles to pour wine, shake glasses and strike matches, among other actions.



# DISTANT GRAB

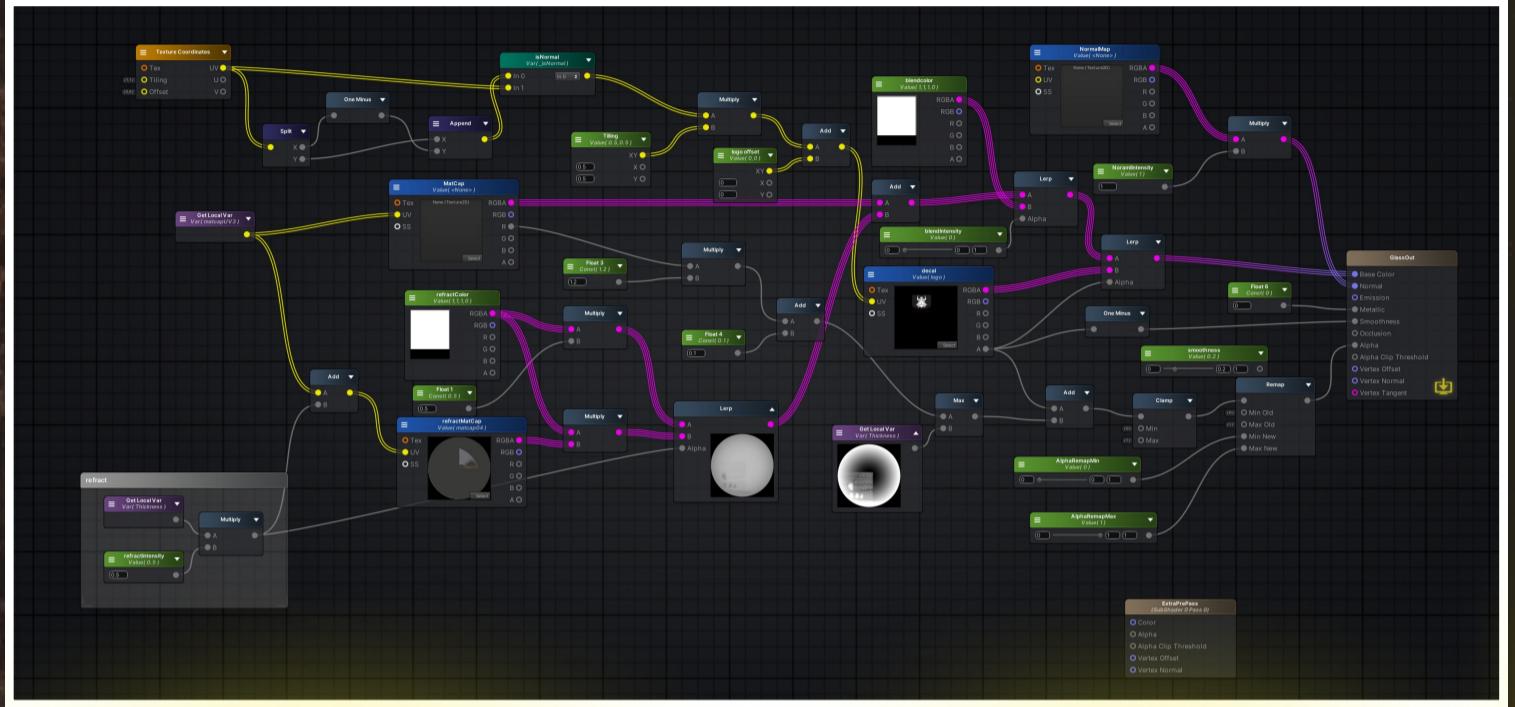
When the player's finger points at the interactive object, a white ray can be seen pointing at the object. At this point, the player can press and hold the GrabTrigger button in the VR grip to lift it up and release it, and within a certain distance, the interactive object will fly to the player in a parabolic motion with precision.

The player simply holds the object in a "direct pickup" position. This eliminates the need for the player to walk to the object every time they want to interact with it, and allows the player to focus on the interaction itself, reducing the chance of being distracted by other things.



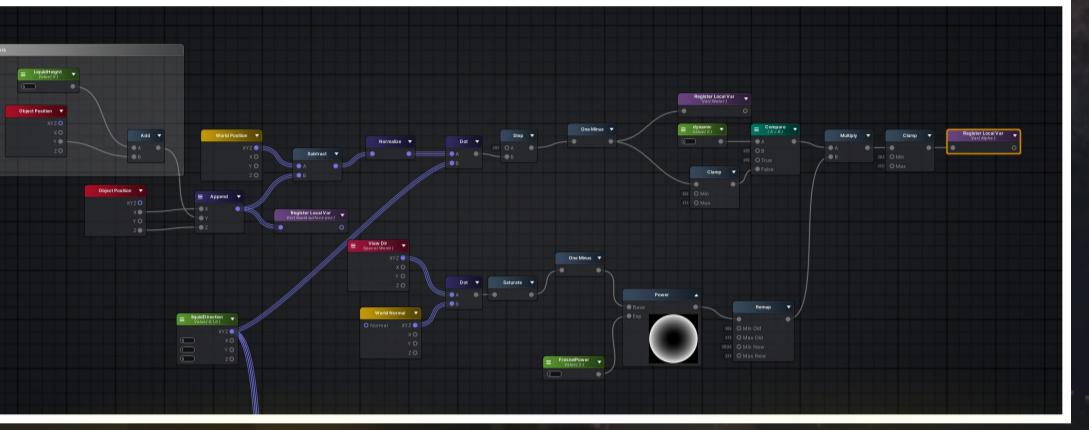
# SHADER EFFECT

## Glass Bottle With Liquid(Out)



For the outside of the glass, the MatCap technique is applied to use only one texture, reducing the performance footprint. In addition, for ASE Shader, the Texture Coordinates node is used to change the tiling size of the UVs, sample a stain map, and overlay it to add realism to the glass.

## Glass Bottle With Liquid(In)

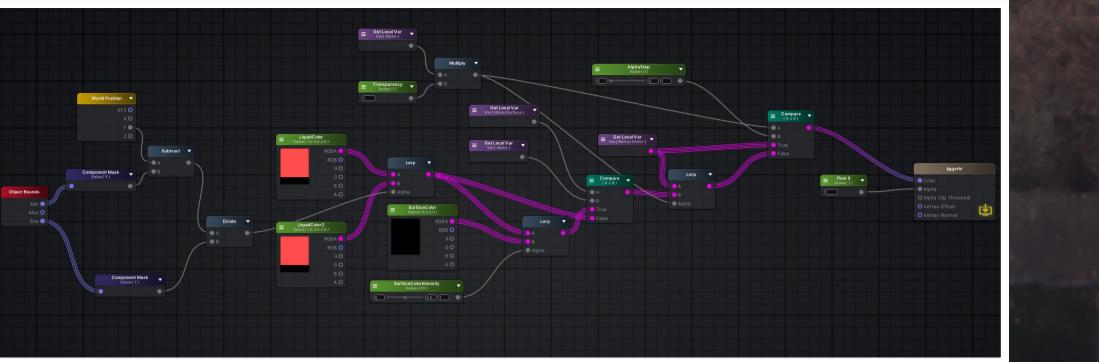


The inside of the glass is divided into three parts, the empty liquid part, the liquid surface part and the liquid part.

The more complicated part is the calculation of the liquid part. (As the sketch shows I simulated math functions in my code.)

For the liquid itself, I applied the Fresnel effect equation

```
float fresnel = pow(1 - saturate(dot(viewDir, normalDir)), _FresnelPower);
```



```
void Update()
{
    var positionWorld = transform.position;
    Vector3 localAxis = isLocal ? transform.forward : transform.up;
    Vector3 worldAxis = Vector3.up;
    float angle = Vector3.Angle(worldAxis, worldAxis);
    var staticLiquidHeight = anisotropicCurve.Evaluate((angle / 180) + dynamicLiquidHeight);
    currentvalue = CalculateSpring(position);
    var liquidheight = staticLiquidheight + Mathf.Clamp(currentvalue - position.y, -heightOffset, heightOffset);
    material.SetFloat("_LiquidHeight", liquidheight);
}
```

```
Vector3 CalculateSpring(Vector3 targetValue)
{
    float deltaTime = Time.deltaTime;
    _oscillationX = Mathf.Lerp(_oscillationX, (targetValue.x - currentValueXYZ.x) * springinessXY, t.deltaTime * springinessXY);
    currentValueXYZ.x += _oscillationX * deltaTime * (1f - dampingXY);
    _oscillationY = Mathf.Lerp(_oscillationY, (targetValue.y - currentValueXYZ.y) * springinessXY, t.deltaTime * springinessXY);
    currentValueXYZ.y += _oscillationY * deltaTime * (1f - dampingXY);
    _oscillationZ = Mathf.Lerp(_oscillationZ, (targetValue.z - currentValueXYZ.z) * springinessXY, t.deltaTime * springinessXY);
    currentValueXYZ.z += _oscillationZ * deltaTime * (1f - dampingXY);
    return currentValueXYZ;
}
```

And as a movable glass, the liquid level physical simulations need to follow the movement of the object and generate the corresponding state.

The information such as liquid surface orientation and liquid surface height obtained from the state of the object is computed and passed in by the C# script.

$$\vec{R} \cdot \vec{j} = |\vec{x}| |\vec{j}| \cos \theta$$

$$\cos \theta = \frac{\vec{n}_1 \cdot \vec{n}_2}{\sqrt{|\vec{n}_1|^2 - |\vec{n}_2|^2}}$$

$$\cos \theta = \frac{q}{t}$$



# PROCESS

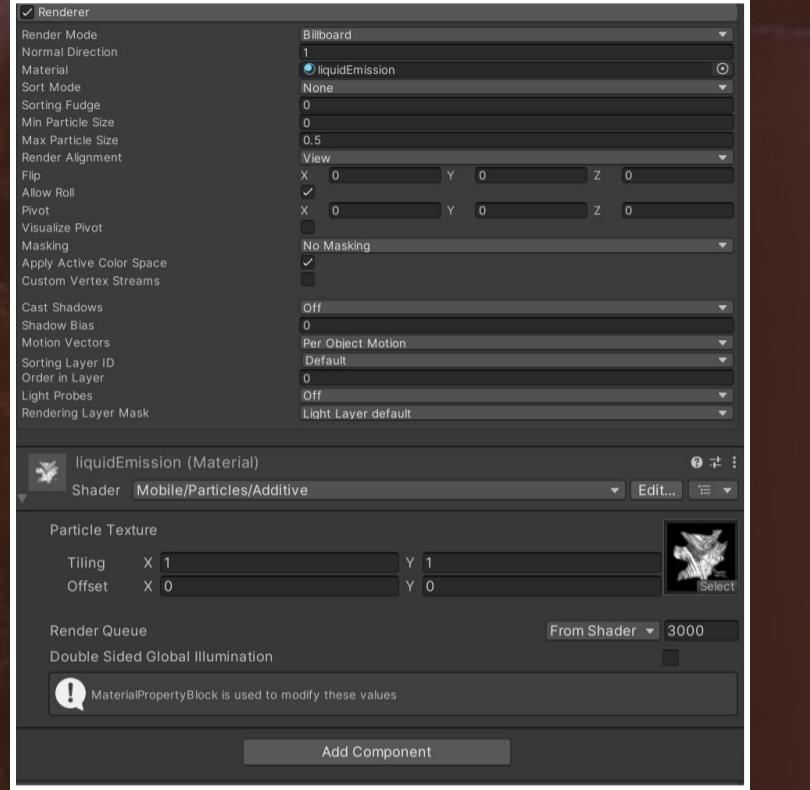


As for refraction Effect, I used hlsl's refract function Result = refract(View, Normal, IOR); and use this to sample the color of the back of the object's occlusion to get the refraction effect.

Also, Cocktail shades are mixed and the liquid will have a color gradient effect.

# VFX DESIGN

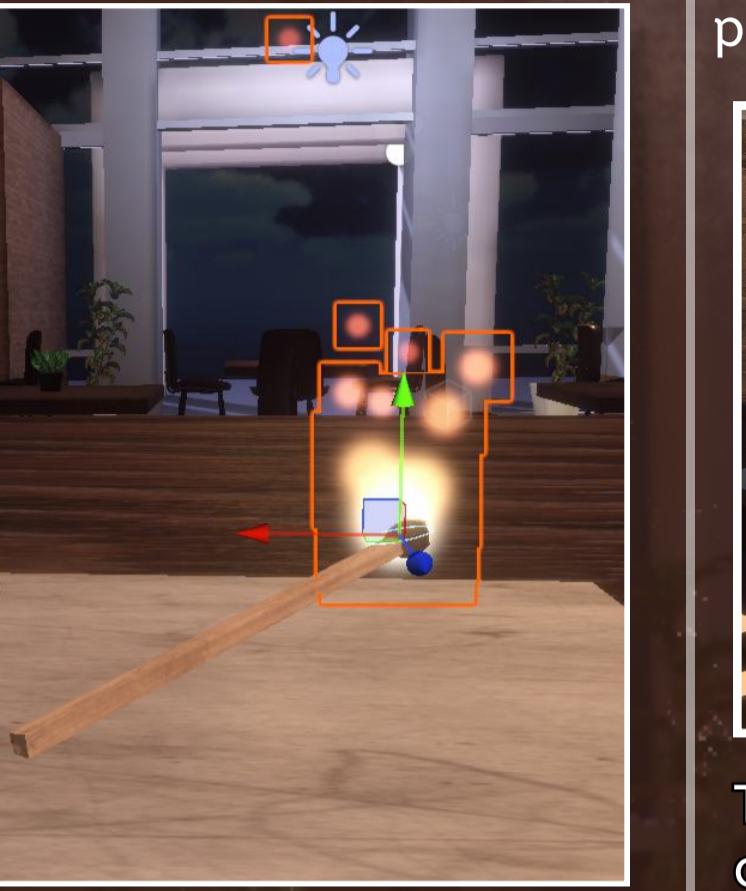
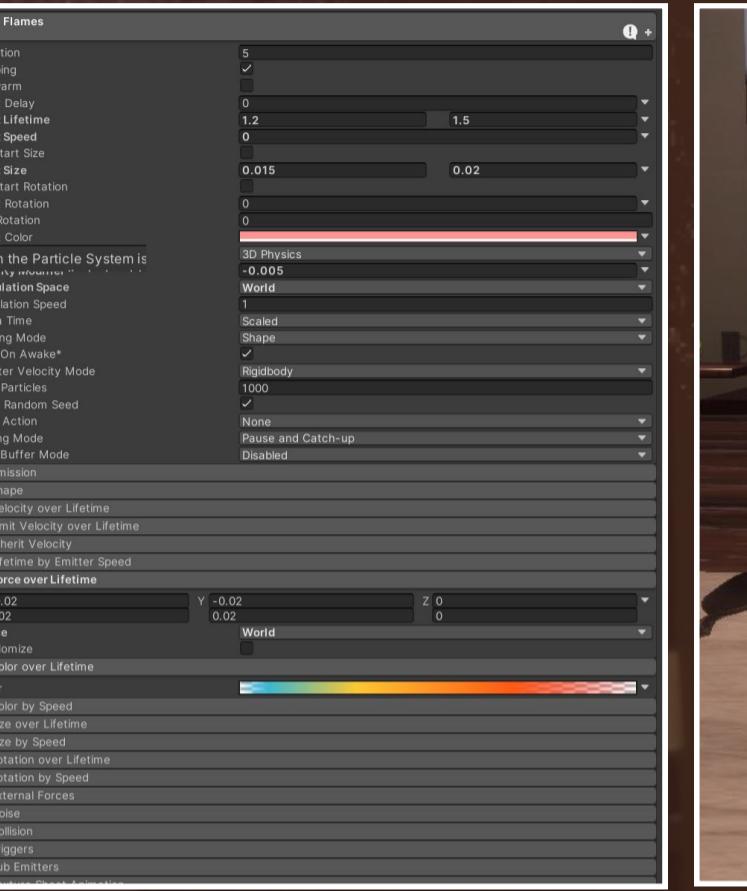
## Fluid effects of wine



Using Unity's Particle System, I can simulate the features of liquid by changing the particle's shader to Additive type. Besides, I added gravity, initial velocity size and direction to the particles, so that the liquid flowing out effect when pouring wine will be more realistic.



## Flame effect

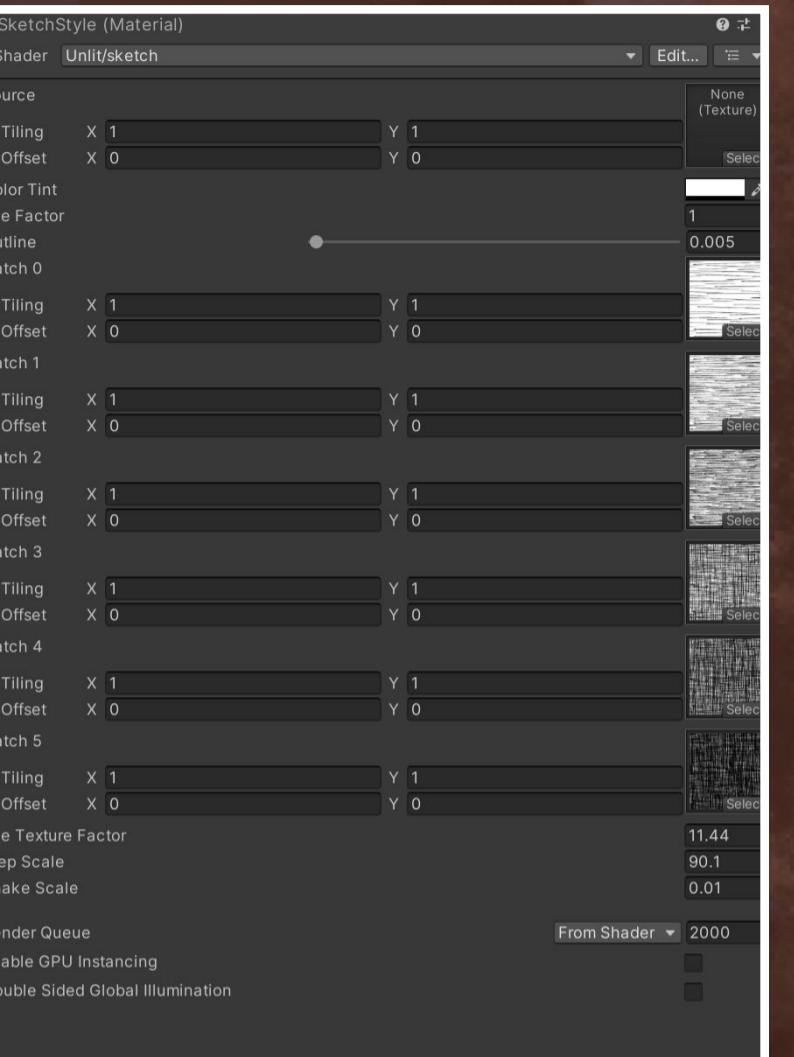
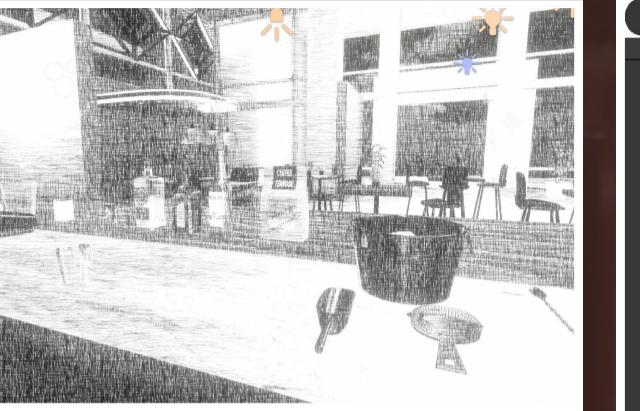


## Post-processing Effect

When intoxicated, the player gets a feeling of drunken vertigo. To create this feeling, this game adds some full-screen post-processing effects.



Sketch style shows that the player is in a state of completely black out. The player no longer has a sense of color in front of him, as if the whole world is made of black and white, and no longer has the ability to perceive color.



The oil painting texture is a representation of tipsy. The whole world seems to be painted over, with a very vivid brush texture.



Sketch style is based on the grayscale grading of the screen color, respectively, to add different sketch layers mapping to achieve the post-process screen effect of sketch texture.

## TEST v1.0-v2.0



After adjusting the hue, brightness and contrast, the player had the feeling of being drunk. All kinds of colors are no longer sharp and real, and the colors presented in the brain are also very different from the real world. The player's senses begins to gradually move towards fainting.