

# **Intégrer la réalité virtuelle dans les formations en Micro-Nano Electronique**

## **Guide du développeur**



**Maxime Tlili - Alexis Briens**  
**Tuteurs : Justine Harmel et Marc Respaud**

**Département du Génie Physique - AIME**

**Année 2023 - 2024**

# Guide du développeur

## SOMMAIRE

<b>Introduction</b>	<b>2</b>
<b>La scène globale</b>	<b>2</b>
<b>Le XRPlayer</b>	<b>3</b>
<b>La Paillasse</b>	<b>3</b>
<b>Le support</b>	<b>8</b>
<b>Le SAS</b>	<b>10</b>

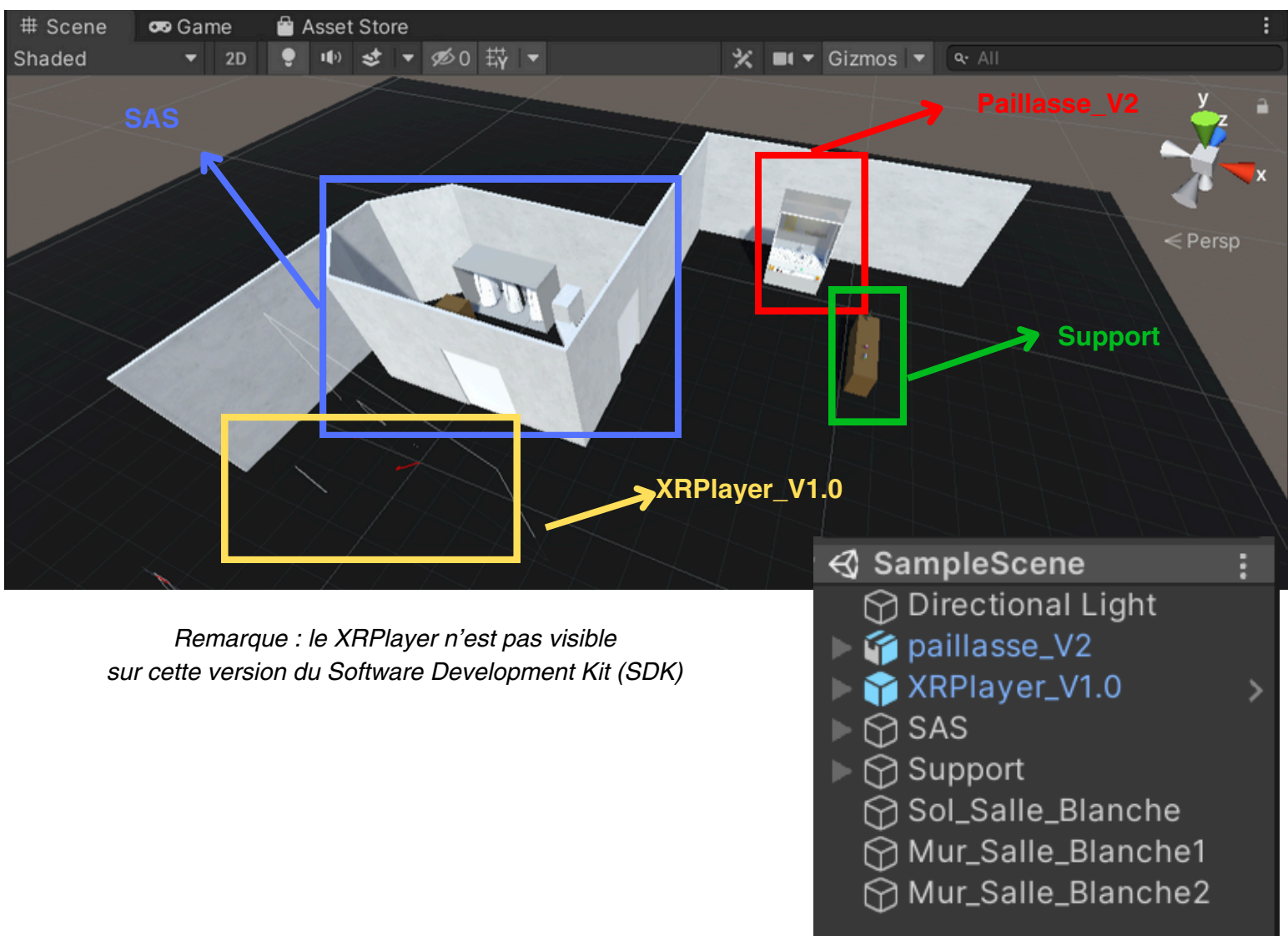
# Introduction

L'année 2023-2024 a été la première année du projet : “ intégrer la réalité virtuelle dans les formations en Micro-Nano Electronique”. C'est dans le cadre du projet Multidisciplinaire du cursus de Génie Physique que nous avons réalisé une scène sur Unity 2019.4.40.

L'objectif de ce projet est la modélisation 3D d'une salle blanche, principalement une paillasse de chimie, avec ajout d'interaction afin de manipuler des produits chimiques par la réalité virtuelle. Il a pour vocation à former et sensibiliser les intervenants avant leur entrée en salle blanche.

Ce manuel présente les fonctionnalités de la scène construite.

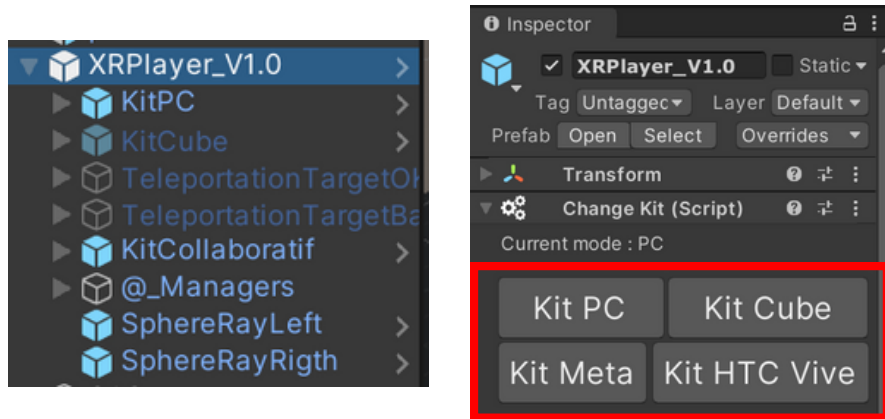
## La scène globale



# Le XRPlayer

Le XRPlayer représente un avatar jouable et compatible avec la version du cube et de Unity. Lors de la simulation, l'utilisateur l'incarne pour se déplacer dans la scène et interagir avec les différents objets.

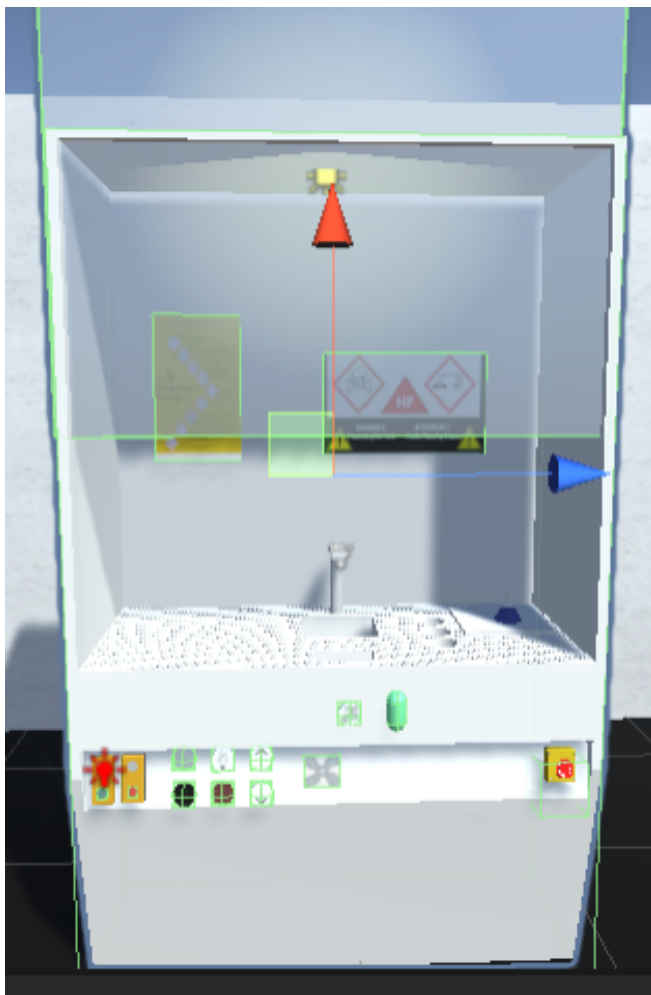
Sur cette version du SDK, le XRPlayer n'est pas visible sur la scène



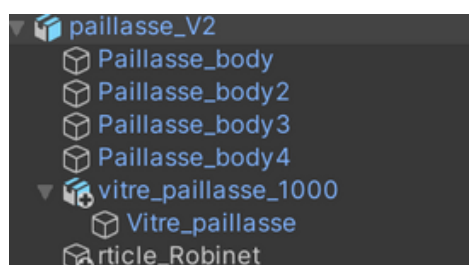
Choix du kit pour lancer le programme. Lors de la création du Build, sélectionner le kit approprié à la technologie utilisée

## La Paillasse

### Présentation générale

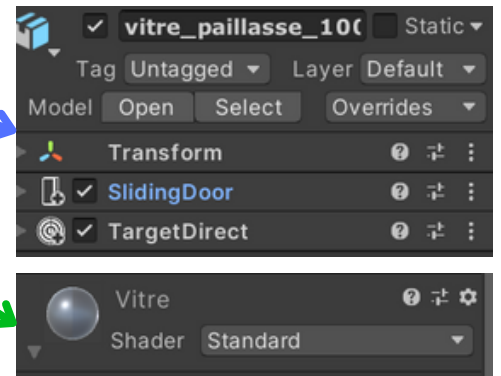
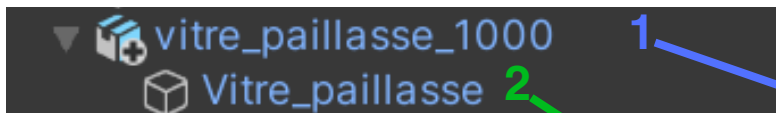


Le modèle 3D de la paillasse a été créé sur Fusion, puis importé dans Unity. Il y a eu assemblage des différents éléments (Corps de la paillasse, vitre, particule robinet, boutons, accessoires, lumières et affiches).

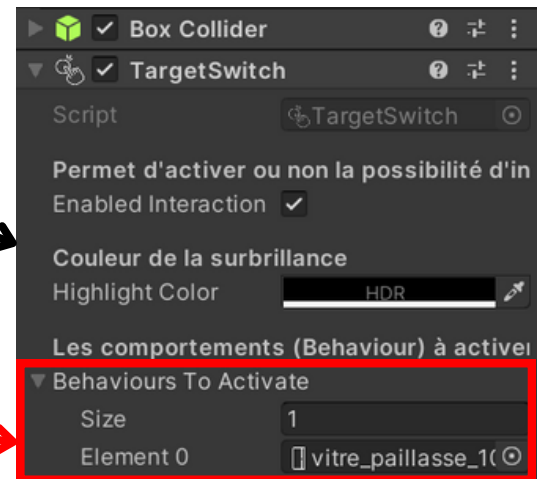
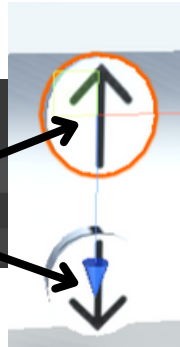
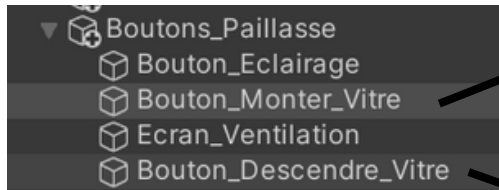


Modèles 3D  
formant  
la paillasse vierge

## Ouverture et fermeture de la vitre

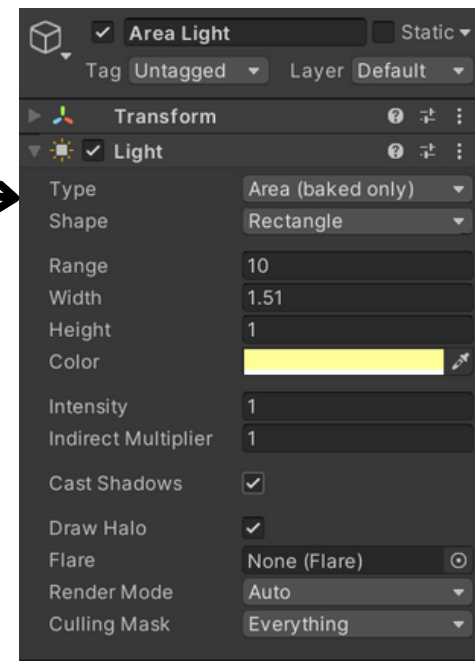
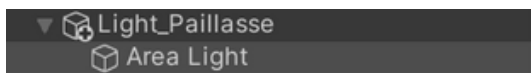


Les deux boutons possèdent le script *TargetSwitch* relié au script *SlidingDoor* de la vitre. Ils ont donc le même fonctionnement

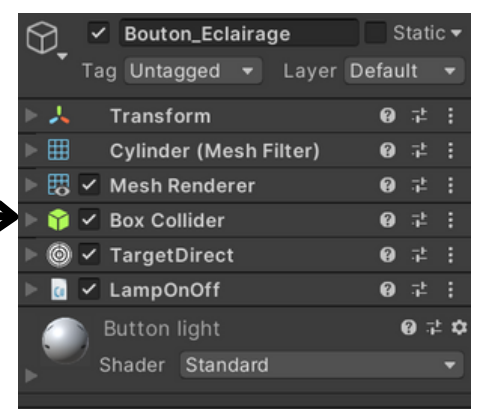


Partie permettant de lier le Target à l'ouverture / fermeture de la vitre

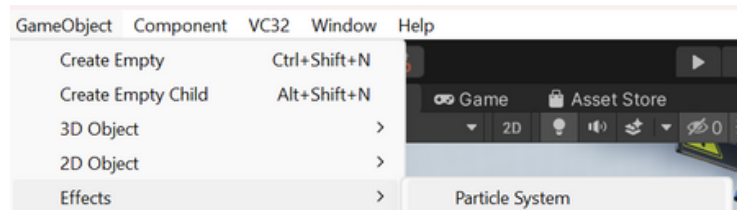
## Les lumières



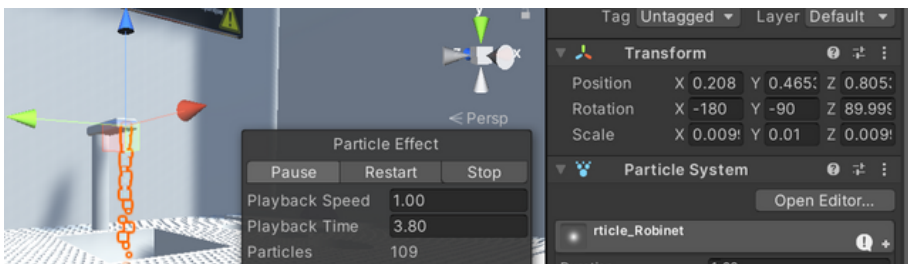
La lumière est de type "Area". Lors de l'interaction avec le bouton lumière, la lumière doit pouvoir s'éteindre. Avec cette version du SDK, le paramétrage du comportement n'a pas été réalisé



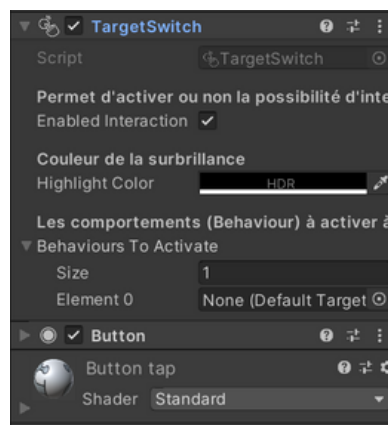
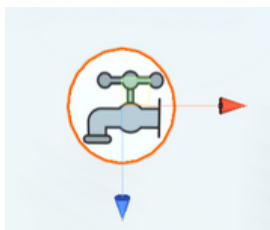
## Le robinet



Création d'un effet Particle System pour simuler le jet d'eau du robinet.



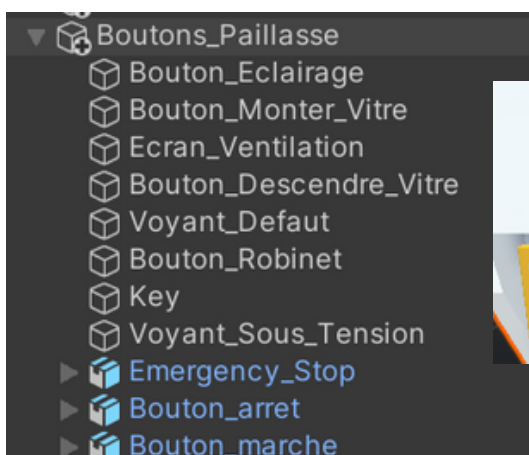
L'origine de l'effet est placée au niveau du robinet



Sur le bouton est placé un TargetSwitch. Il n'agit sur aucun comportement. Une fois que le comportement pour activer l'effet sera paramétrer, glisser l'élément dans la case "Element 0"

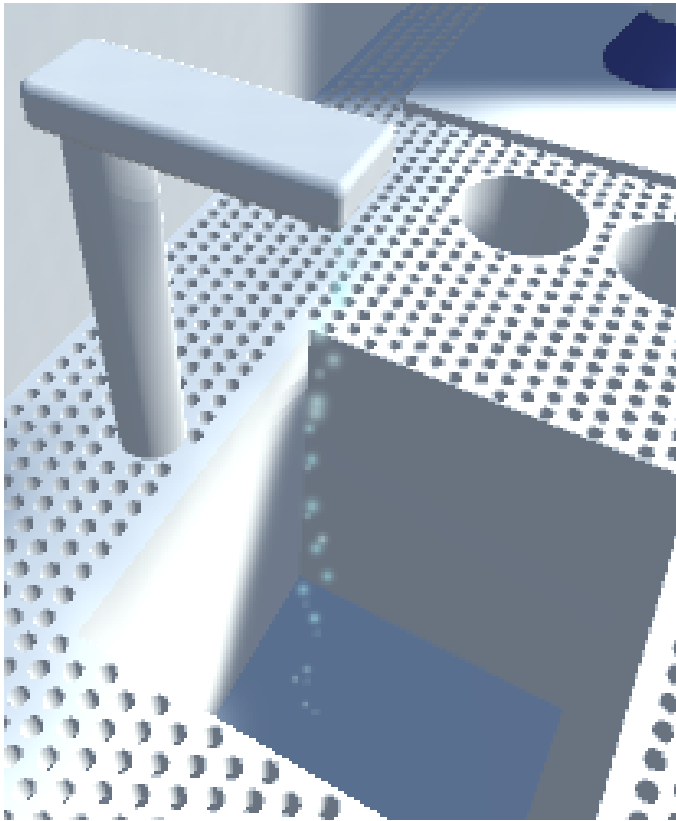
Sur la page suivante se situe le guide pour configurer cet effet

## Les autres boutons





## Paramètres de l'effet "rticle\_Robinet"



Inspector

rticle\_Robinet Static

Tag Untagged Layer Default

Transform

Position X 0.208 Y 0.4653001 Z 0.8053999

Rotation X -180 Y -90 Z 89.99999

Scale X 0.009999999 Y 0.01 Z 0.009999999

Particle System

Open Editor...

rticle\_Robinet

Duration 1.00

Looping ☒

Prewarm ☐

Start Delay 0

Start Lifetime 1 1.2

Start Speed 12 16

3D Start Size 1 3

Start Size 1 3

3D Start Rotation 0

Start Rotation 0

Flip Rotation 0

Start Color

Gravity Modifier 1

Simulation Space World

Simulation Speed 1

Delta Time Scaled

Scaling Mode Local

Play On Awake\* ☒

Emitter Velocity Rigidbody

Max Particles 1000

Auto Random Seed ☒

Stop Action None

Culling Mode Automatic

Ring Buffer Mode Disabled

☒ Emission

☒ Shape

Velocity over Lifetime

Limit Velocity over Lifetime

Inherit Velocity

Force over Lifetime

☒ Color over Lifetime

Color

☐ Color by Speed

☒ Size over Lifetime

Separate Axes ☐

Size

☐ Size by Speed

☐ Rotation over Lifetime

☐ Rotation by Speed

☐ External Forces

☐ Noise

☐ Collision

☐ Triggers

☐ Sub Emitters

☐ Texture Sheet Animation

☐ Lights

☐ Trails

☐ Custom Data

☒ Renderer

Render Mode Billboard

Normal Direction 1

Material Default - ParticleSystem

Sort Mode None

Sorting Fudge 0

Min Particle Size 0

Max Particle Size 0.5

Render Alignment View

Flip X 0 Y 0 Z 0

Allow Roll ☒

Pivot X 0 Y 0 Z 0

Visualize Pivot ☐

Masking No Masking

Apply Active Color Space ☒

Custom Vertex Streams ☐

Cast Shadows Off

Receive Shadows ☐

Shadow Bias 0

Motion Vectors Per Object Motion

Sorting Layer ID Default

Order in Layer 0

Light Probes Off

Reflection Probes Off

Default - ParticleSystem

Shader Particles/Standard Unlit

Controls the wind zones that each particle is affected by.

☒ Emission

Rate over Time 100

Rate over Distance 0

Bursts

Time	Count	Cycles	Interval	Probability
List is Empty				

☒ Shape

Shape Cone

Angle 15

Radius 0.1

Radius Thickness 0.47

Arc 360

Mode Random

Spread 0

Length 5

Emit from: Base

Texture None (Texture 2D)

Position X 0 Y 0 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

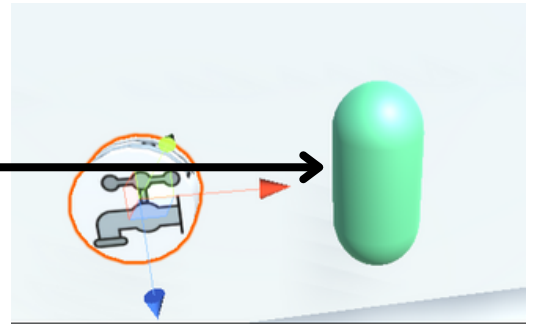
Align To Direction ☐

Randomize Direction 0

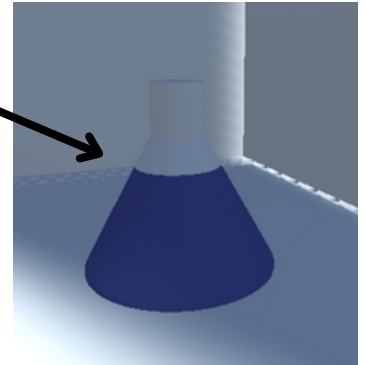
Spherize Direction 0

Randomize Position 0

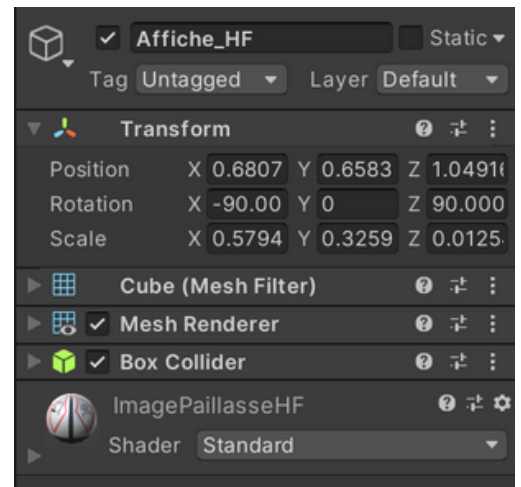
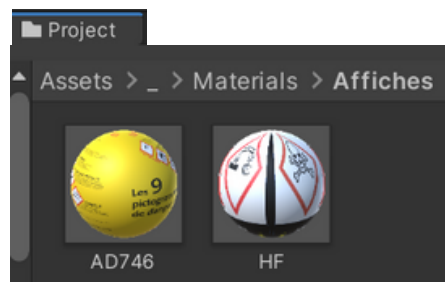
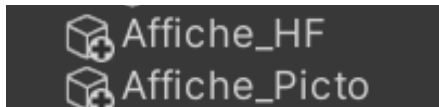
## Les accessoires



Ces éléments ont été positionnés pour la suite du projet. Ils ne possèdent pas de scripts pour interagir avec eux



## Les affiches

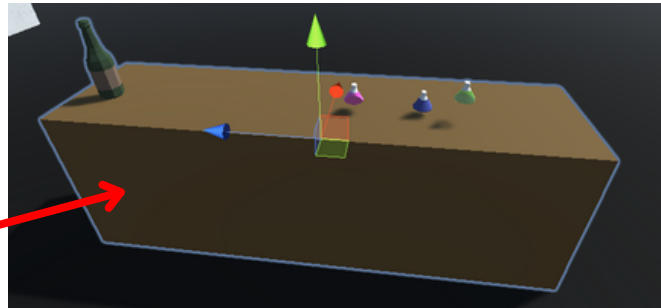


Les deux affiches *Affiche\_HF* et *Affiche\_Picto* sont des cubes 3D redimensionnés où a été ajouté un matériau. La texture du matériau est une image .png



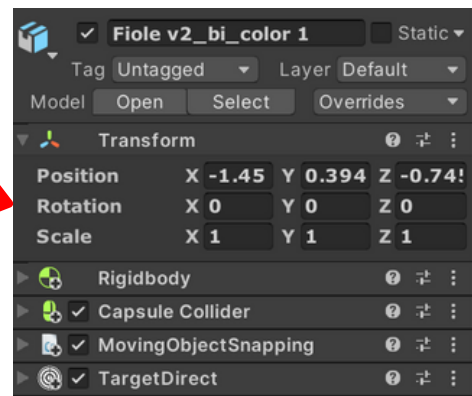
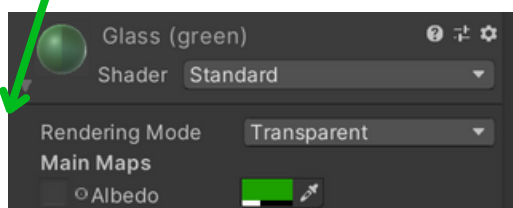
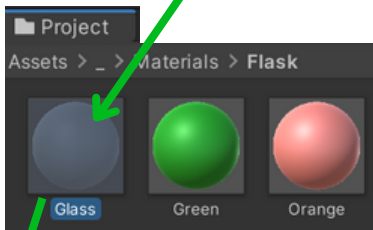
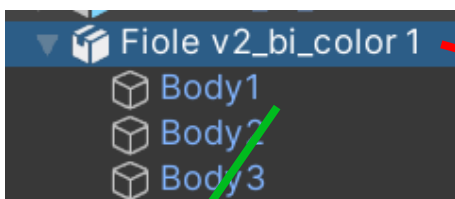
# Le support

## Présentation générale



Le support est un cube redimensionné sur lequel sont placées différentes fioles ainsi qu'une bouteille possédant le script PourEffect. Il permet de tester les scripts de versement

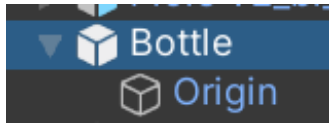
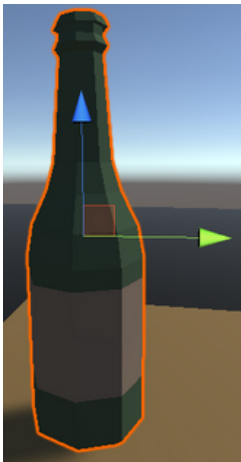
## La fiole



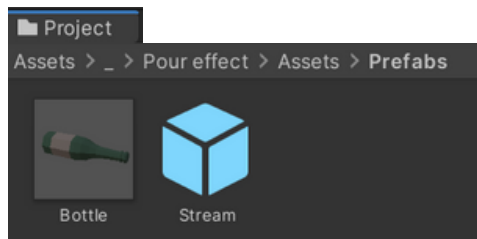
Seule la fiole verte possède les scripts *MovingObjectSnapping* (pour pouvoir être saisie par l'utilisateur), *TargetDirect* (pour permettre l'interaction) et *RigidBody* (qui simule la gravité)

Sur le support, il y a trois fioles (rose, bleue, verte). Ces modèles ont été retravaillés sur fusion pour représenter la partie liquide. Le corps a été séparé en trois parties. Le Body1 possède un matériau "glass" de Rendering Mode "Transparent". Pour la Fiole v2\_bi\_color1, la couleur choisie (Albedo) est le vert.

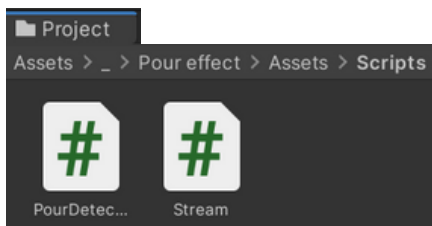
## La bouteille



La bouteille est un modèle Prefab, c'est-à-dire qu'il a été assemblé avec ses scripts puis importé sur cette scène.



La bouteille permet de tester le script *PourDetector*. Les deux scripts sans détails sont fournis avec le PreFab. Les mêmes scripts détaillés ne sont pas disponible dans Unity mais sur le GitHub associé.



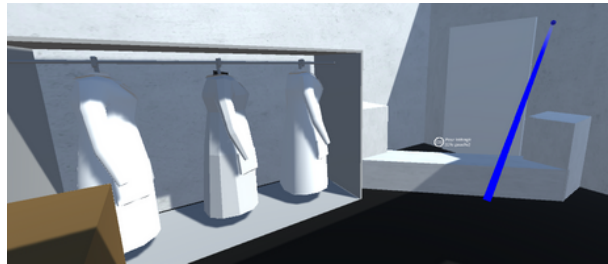
Le prefab *Bottle* contient la bouteille et l'origine de l'effet de versement. On lui a rajouté les scripts *MovingObjectSnapping* et *TargetDirect* pour se saisir de l'objet, comme pour les fioles

# Le SAS

## Présentation générale

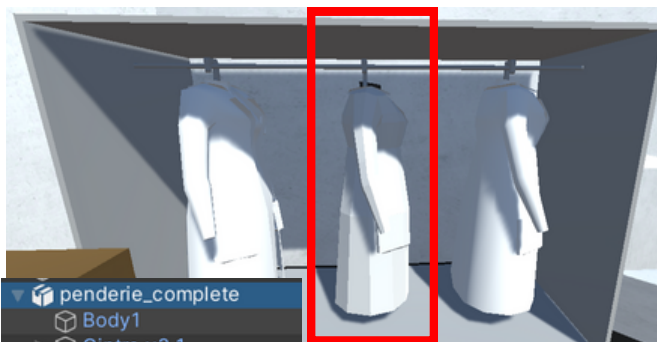


Le SAS a été construit à partir de cubes redimensionnés pour faire des murs (*Sas\_Mur*). Il possède une jonction (*Jonction\_Du\_Sas*) ainsi qu'une table (*Table\_Sas*) et une penderie (*penderie\_complete*) où se situent des blouses

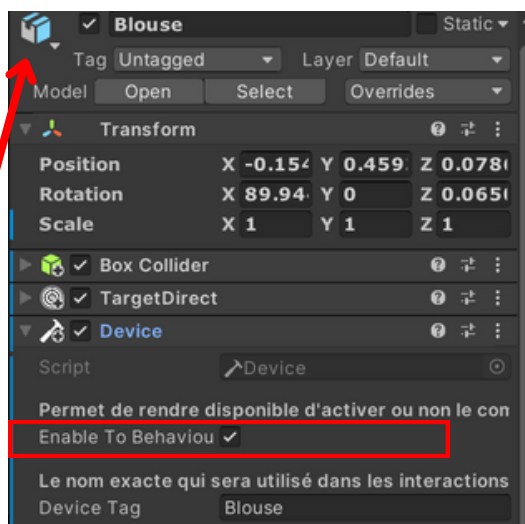
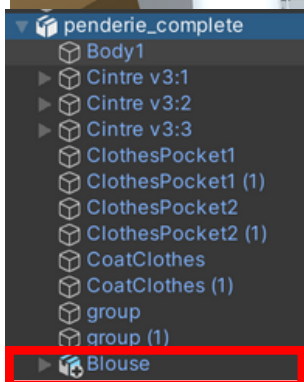


Vue à l'intérieur du SAS

## La blouse

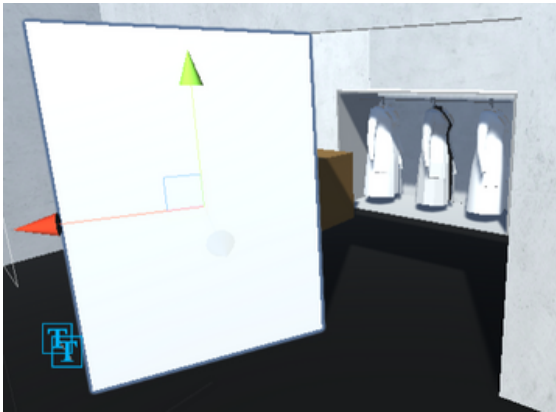


La penderie est un Prefab importé sur Unity. Nous avons modifié le fichier existant pour laisser une place vide au milieu pour le modèle de blouse portable. Dans la hiérarchie, cela correspond au prefab *Blouse*.

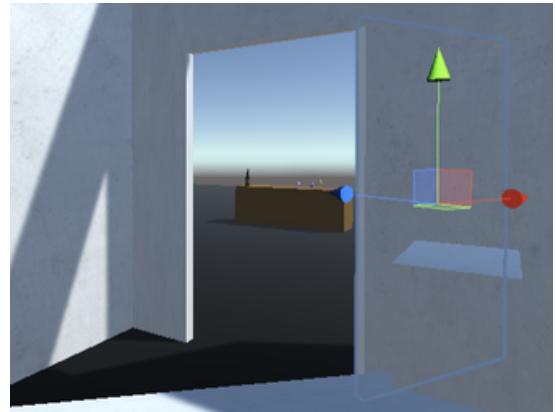


La blouse du milieu est la seule contenant les scripts *TargetDirect* et *Device* qui rendent son interaction possible. Le script *Device* est associé au *Device Tag* "Blouse". Ce *Device Tag* permet de configurer la porte de sortie du SAS

## Les portes



Porte Entrée



Porte Sortie

▼

PorteEntréeSas

1

▼

PorteEntréeSasObj

2

▼

PorteSortieSas

▼

PorteSortieSasObj

Empty

3D Object

Une porte est composée de deux objects : un *3D Object* qui représente la porte, et un *empty* qui représente l'axe de rotation pour le script *RotatingDoor*. Ainsi, les scripts d'interaction sont sur l'*empty*

La différence entre la porte d'entrée et la porte de sortie est la suivante :

La porte de sortie possède le *Device Tag* "Blouse" dans le script *RotatingDoor*. L'interaction est activée si et seulement si l'utilisateur possède la blouse (avec son script *Device* de même *Device Tag*) dans sa main.

Porte Entrée

Le nom exacte qui sera utilisé dans les interactions  
Device Tag

Porte Sortie

Le nom exacte qui sera utilisé dans les interactions  
Device Tag Blouse

# Guide du développeur



**Année 2023 - 2024**