

# Rapport TP 4 – Jeu de Bowling VR

Module : Réalité Virtuelle et Augmentée

Niveau : 2e Cycle – ESI Sidi Bel Abbès

Étudiant : Abdelillah Serghine

Groupe : 02 IASD

## Objectif du TP

L'objectif de ce TP est de créer une **scène de bowling en style VR simple dans Unity**, en utilisant les outils de **XR Interaction Toolkit** et le **XR Device Simulator**, pour simuler un environnement immersif sans casque VR réel.

## Étapes de Réalisation

### 1. Création du projet

- Un projet Unity 3D nommé VRBowling a été créé.

### 2. Installation des packages XR

- **XR Plugin Management, XR Hands** a été installé via le Package Manager.
- **XR Interaction Toolkit** a été ajouté, avec les exemples inclus (samples).
- Dans Project Settings > XR Plugin Management, **OpenXR** a été activé pour la plateforme "PC, Mac & Linux Standalone".
- Le système d'entrée (Input System) a été activé avec redémarrage de Unity.

### 3. Construction de la scène

#### Sol du bowling

- Un Plane a été ajouté pour servir de sol, renommé BowlingFloor.

#### XR Origin et simulateur

- Le prefab **XR Origin (Action-Based)** a été inséré dans la scène.
- Le prefab **XR Device Simulator** a été ajouté pour simuler les mouvements et interactions sans casque VR.

## 4. Création des objets du jeu

### La boule de bowling

- Un objet Sphere renommé BowlingBall a été ajouté.
- Composants ajoutés :
  - Rigidbody (avec Use Gravity)
  - Sphere Collider
  - XR Grab Interactable pour la rendre saisissable par les mains virtuelles

### Les quilles (pins)

- 10 objets Cylinder ont été ajoutés pour représenter les quilles.
- Ils ont été placés manuellement en forme triangulaire comme dans un vrai jeu de bowling.
- Chaque quille possède :
  - Un Rigidbody non cinématique
  - Un Collider

## Modifications supplémentaires réalisées :

- **Ajout du système de score :**
  - Création d'un objet GameManager contenant le script GameManager.cs.
  - Ajout d'un élément d'interface utilisateur (ScoreText) dans un Canvas pour afficher le score en temps réel.
  - Ajout du script Pin.cs à chaque quille pour détecter lorsqu'une quille est tombée.
  - Mise à jour automatique du score à chaque chute d'une quille détectée via l'angle d'inclinaison.
- **Ajout et intégration des mains XR :**
  - Création des objets LeftHand Controller et RightHand Controller sous le XR Origin.
  - Ajout des composants XR Controller (Action-based), XR Direct Interactor, et Collider (Sphere) sur chaque main pour permettre l'interaction directe avec les objets.
  - Intégration avec XR Device Simulator pour simuler la saisie et le lancer de la boule sans casque VR réel.

## 5. Test et Simulation

### Utilisation du XR Device Simulator :

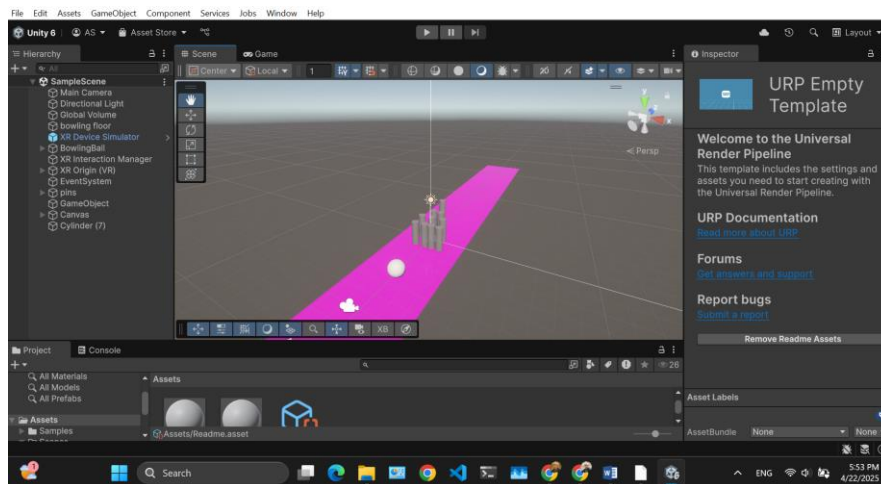
- **Déplacement** dans la scène avec les touches W, A, S, D
- **Regarder autour** : Shift + souris
- **Changer de main** : Q / E
- **Déplacer la main** : Ctrl / Alt + souris
- **Saisir/relâcher la boule** : G

## Résultats obtenus

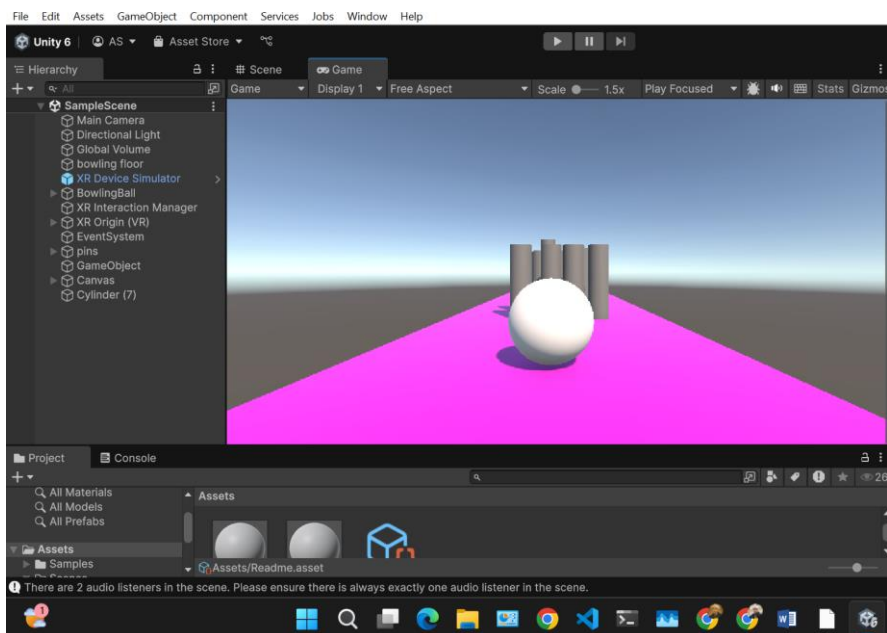
- Le joueur peut se déplacer dans la scène via le simulateur.
- Les quilles réagissent à la collision grâce aux composants physiques (Rigidbody + Collider).
- L'expérience immersive est simulée avec succès sans avoir besoin d'un casque VR.

## Captures d'écran

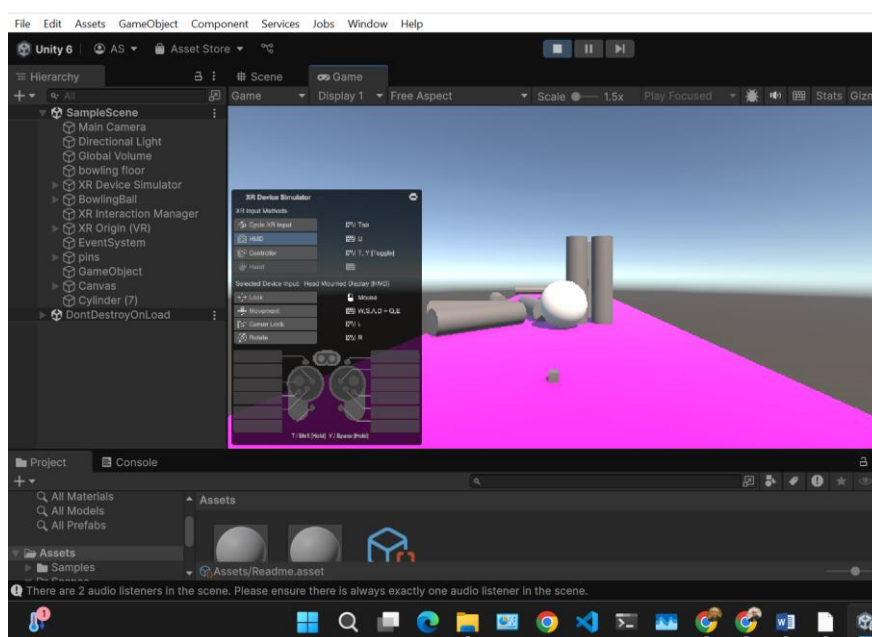
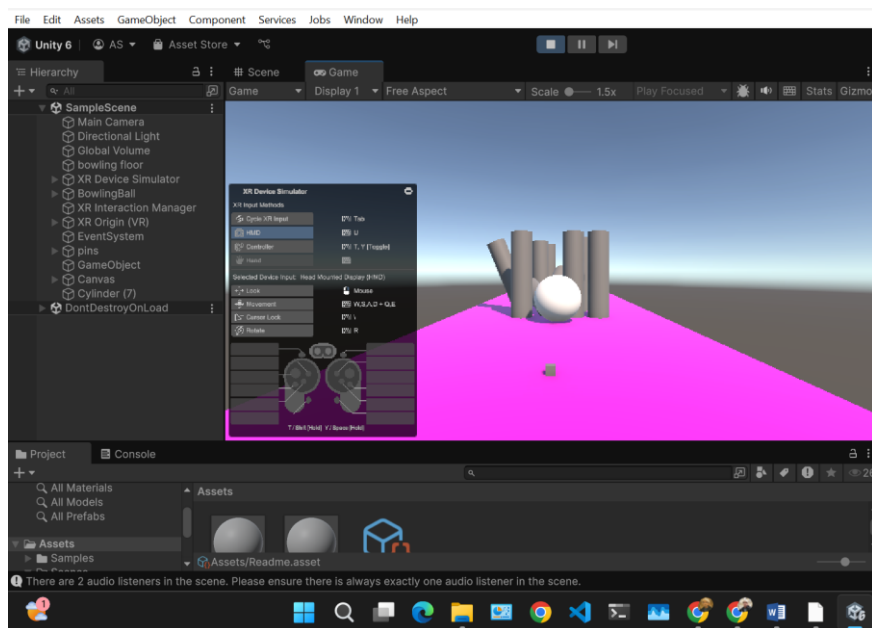
- Vue de la scène (éditeur)



- Vue en jeu (Game View)

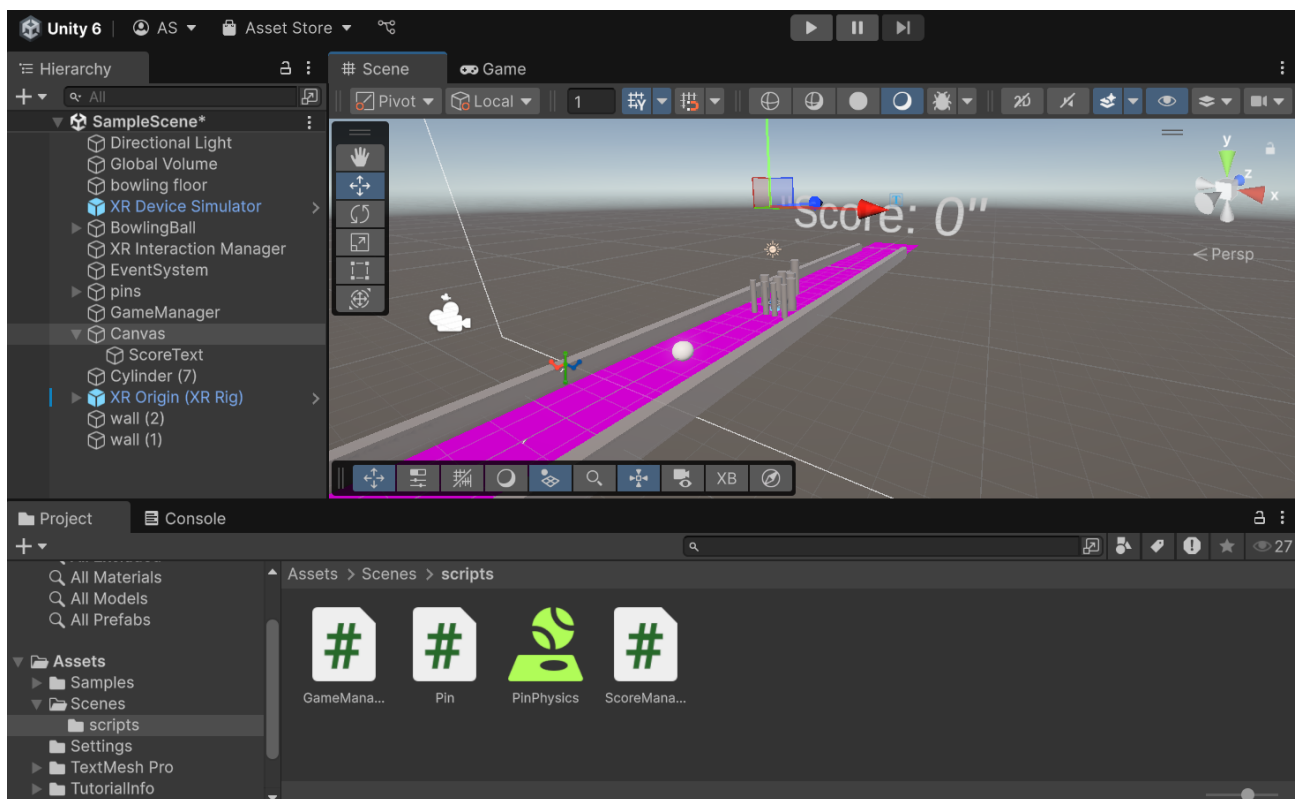
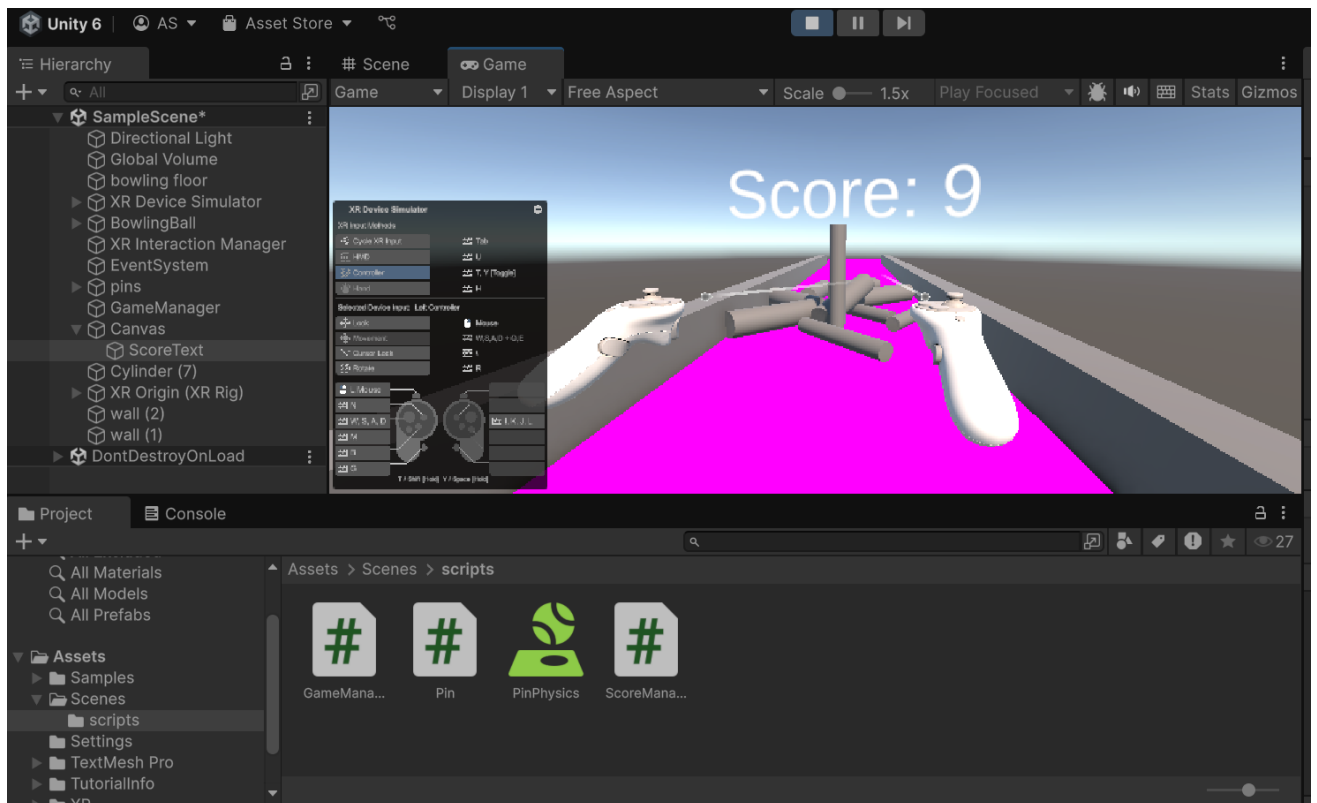


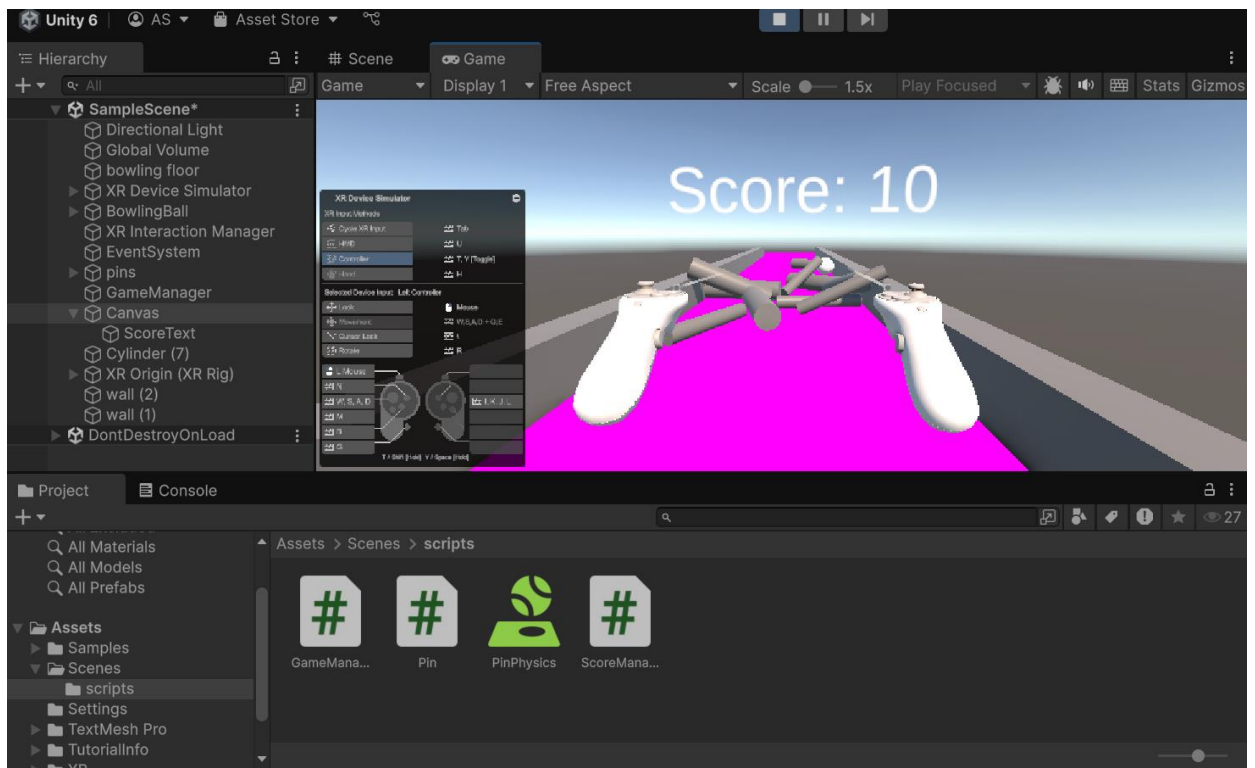
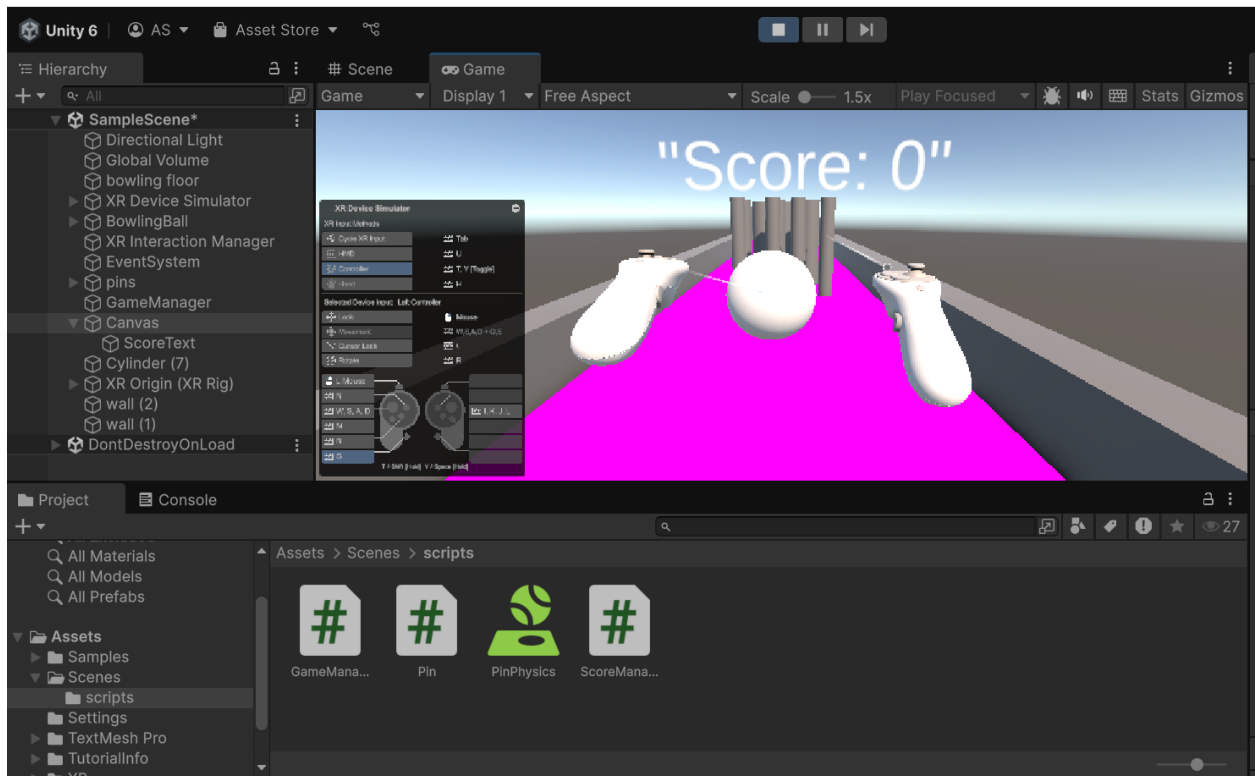
- Quilles en train de tomber



## Les nouvelles modifications :

- Ajout de mains pour contrôler la balle.
- Ajout du système de score.
- Ajout de murs autour du plan pour empêcher la balle de sortir des limites du terrain





## Conclusion

Ce TP a permis d'explorer la configuration d'un environnement XR dans Unity, en manipulant les interactions via le **XR Device Simulator**. La simulation du jeu de bowling a été réalisée avec succès en intégrant les composants nécessaires pour gérer les mouvements, la saisie, et les collisions physiques dans un environnement 3D immersif.

---

## Fichiers utilisés

- Unity 6000.0.46f1
- XR Plugin Management
- XR Interaction Toolkit v3.0.8
- Input System
- XR Device Simulator (sample)