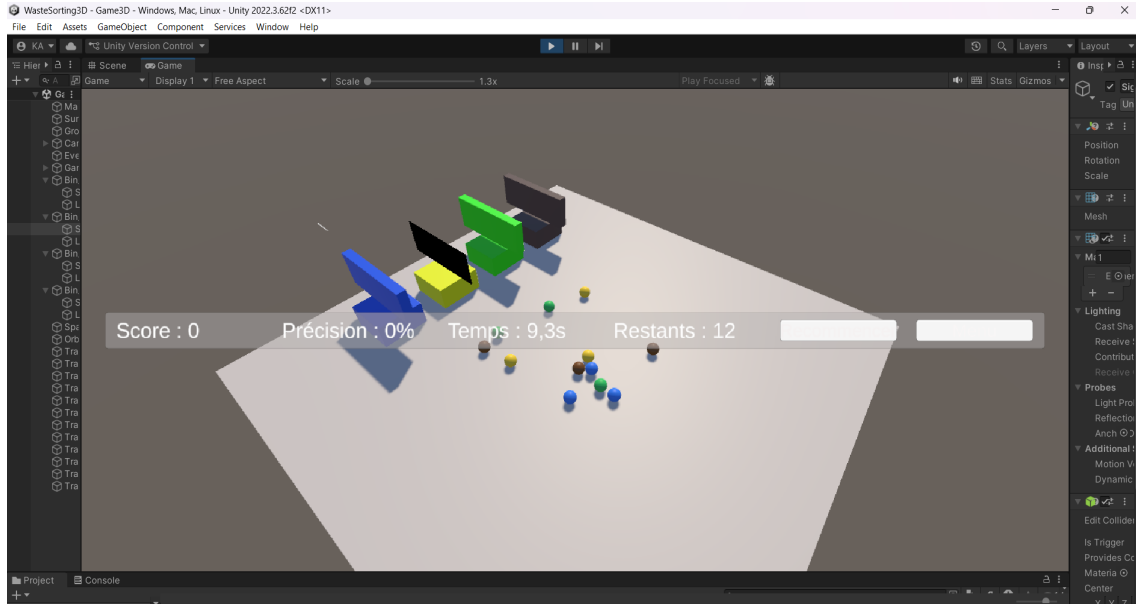


# ■ Rapport professionnel

## Mini-projet Unity 3D : Simulation interactive de tri des déchets



Nom :	AITLBIZ Kaoutar
Logiciel :	Unity 2022.3.6f1
Année :	2025
Thème :	Simulation 3D éducative du tri des déchets

## 1. Introduction

Ce mini-projet a pour objectif de créer une simulation 3D interactive éducative réalisée avec Unity, destinée à sensibiliser les étudiants au tri sélectif des déchets. L'utilisateur interagit avec des objets 3D colorés représentant différents types de déchets et doit les placer dans les bacs correspondants. Ce projet combine apprentissage scientifique, manipulation d'objets 3D et évaluation automatique des performances.

## 2. Objectifs pédagogiques

Concevoir une scène Unity complète et interactive.

Mettre en œuvre une interaction directe utilisateur-objet (cliquer, déplacer, déposer).

Créer un système de score, précision et temps intégré.

Structurer un mini-projet réutilisable avec des dossiers clairs (Scenes, Scripts, Materials, Prefabs, UI).

Évaluer automatiquement les actions de l'utilisateur (bonne ou mauvaise catégorie).

### 3. Description fonctionnelle

L'utilisateur doit déplacer à la souris des sphères colorées (déchets) et les déposer dans les bacs correspondants :

Papier (bleu), Plastique (jaune), Verre (vert), Organique (marron).

À chaque action :

Un message de feedback s'affiche ("■ Bravo !" ou "■ Mauvais bac").

Le score, la précision et le temps sont mis à jour en temps réel.

Une note finale est affichée à la fin de la simulation.

### 4. Conception technique

#### 4.1. Scène principale

Une seule scène a été utilisée : Game3D, qui contient le plateau principal (Ground), les bacs de tri, les objets à déplacer (TrashItems), le Canvas d'interface (score, temps, boutons) et le GameManager3D qui centralise toute la logique du jeu.

#### 4.2. Éléments 3D

Sol (Plane) : surface principale (couleur gris clair).

Bacs (Cubes) : 4 boîtes colorées : Bleu (Papier), Jaune (Plastique), Vert (Verre), Marron (Organique).

Chaque bac a : BoxCollider (Is Trigger = ON), script BinZone.cs, pancarte avec texte TextMeshPro.

Objets "déchets" (Prefab TrashItem) : Rigidbody (Use Gravity ■, Drag = 2), Collider actif (Is Trigger ■), script Draggable3D.cs, matériau coloré selon la catégorie.

### 5. Interface utilisateur (UI)

#### 5.1. Composants du Canvas

TextScore → Score : 0

TextAccuracy → Précision : 0%

TextTime → Temps : 0.0s

TextRemaining → Restants : 0

FeedbackText → affiche "Bravo !" ou "Mauvais bac"

Boutons : Recommencer (relance la scène actuelle), Quitter (ferme l'application).

#### 5.2. Script UIManager.cs

Ce script actualise le score, la précision, le temps écoulé et affiche les messages de feedback.

## 6. Gestion du jeu – Logique principale

### 6.1. GameManager3D.cs

Gère la génération automatique des objets (GenerateItems()), attribue la catégorie à chaque sphère, détecte si un objet est déposé dans le bon bac, met à jour le score, joue des sons et sauvegarde les actions dans un fichier CSV via CsvLogger.cs.

### 6.2. ScoreManager.cs

Calcule les taux de réussite, le nombre d'objets restants et la précision finale.

### 6.3. CameraOrbit.cs

Fait tourner la caméra autour de la scène pour toujours placer la vue devant l'utilisateur. L'utilisateur peut aussi pivoter la scène avec la souris (clic droit + glisser).

## 7. Procédure de mise en œuvre

Création du plateau 3D : Ground avec un matériau gris clair.  
Ajout des 4 bacs colorés avec BoxCollider et BinZone.cs.  
Création du Prefab TrashItem : sphère + Rigidbody + Draggable3D.cs.  
Mise en place de l'UI (Canvas, Texts, Buttons).  
Câblage des références dans GameManager3D et UIManager.  
Test des interactions : clic, déplacement, détection correcte.  
Ajout de la caméra orbitale pour une meilleure visualisation.  
Finalisation de l'expérience utilisateur : feedback visuel et sonore.

## 8. Résultat final

À l'exécution : la scène Game3D s'affiche directement. Des sphères colorées apparaissent sur le plateau. L'utilisateur les attrape à la souris et les dépose dans le bon bac. À chaque dépôt : ■ "Bravo !" s'affiche si la catégorie est correcte, ■ "Mauvais bac" sinon. L'interface met à jour le score, la précision, le temps et le nombre restant. Une fois terminé, un résumé final est affiché (score global, précision, durée). Le tout reste visible grâce à la caméra orbitale qui tourne légèrement autour de la scène.

## 9. Conclusion

Ce mini-projet a permis de découvrir les principales notions de développement Unity 3D : création d'un environnement 3D cohérent, utilisation des colliders, rigidbodies et triggers, programmation d'interactions utilisateur avec C#, construction d'une interface utilisateur dynamique et gestion de la logique globale avec un GameManager. Le projet final est une simulation éducative complète, immersive et réutilisable dans le cadre d'autres activités scientifiques ou environnementales. Il offre une base solide pour développer d'autres simulations d'apprentissage interactives.