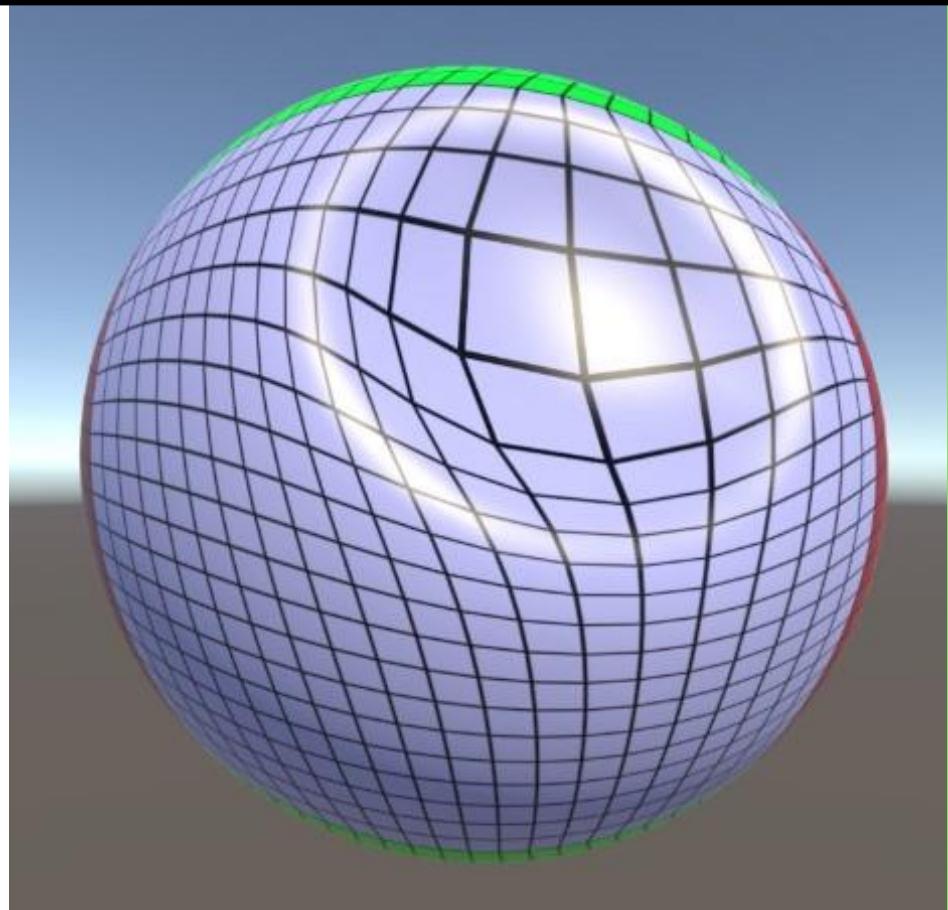


INFO 4900

Projet Final - Rapport



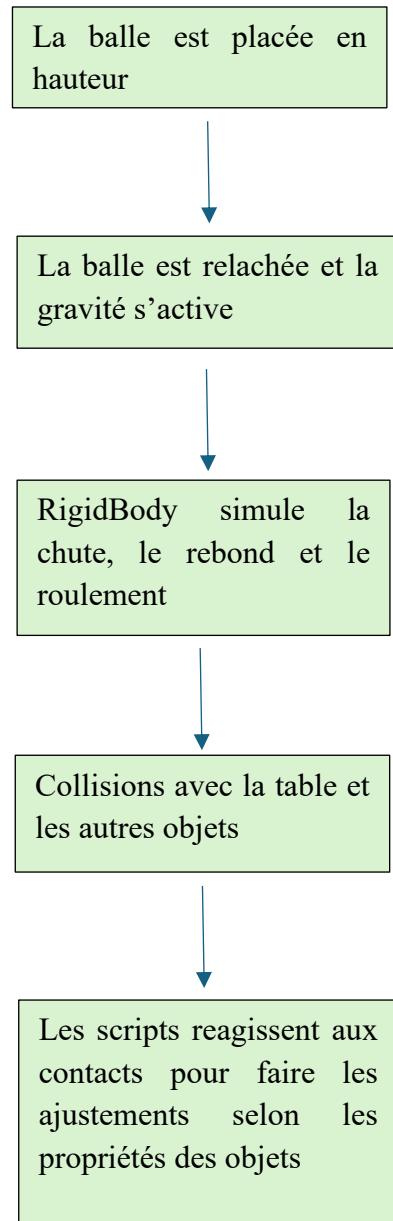
Loic Mutima – A00219227

RFKN Preval – A00217742

Tippi Razafindroriaka – A00224392

Zakari Rafik Chabi – A00219089

I. Architecture



II. Choix techniques

Physique

Le projet utilise la physique de Unity pour gérer les chutes, les rebonds ainsi que toutes les interactions entre objets au moment des impacts. Le ballon repose sur un Rigidbody dynamique associé à un SphereCollider. Son comportement physique est configuré par le script BallController.cs, qui définit sa masse, sa friction, son rebondissement ainsi que le mode de détection de collision continue. Le script gère également la déformation du ballon lorsqu'il subit un impact, ce qui fait qu'il se déforme puis retrouve progressivement sa forme initiale, permettant d'obtenir un rebond visuellement réaliste tout en conservant la cohérence du mouvement.

Le verre utilise lui aussi un Rigidbody dynamique, contrôlé cette fois par le script GlassController.cs. Celui-ci ajuste sa masse, sa friction, son rebondissement, sa stabilité et son comportement lorsqu'il est déplacé, relâché, ou en contact avec un autre élément de la scène. Le même mode de détection de collision continue est activé pour éviter les traversées, ce qui garantit un comportement physique réaliste, qu'il s'agisse d'un simple déplacement ou d'un choc suffisamment fort pour provoquer sa cassure.

Collision

Les collisions sont gérées automatiquement par Unity à travers les colliders attachés aux objets, mais leur comportement réel dépend des scripts qui réagissent aux impacts. Dans le cas du ballon, les contacts sont détectés par le SphereCollider et transmis au script, qui analyse l'intensité de la collision. Le verre repose sur un principe similaire où le résultat de sa collision dépend de la propriété des éléments avec lesquels il rentre en contact, il reste donc intact lorsqu'il reçoit un impact léger comme celui du ballon souple, mais se casse lorsqu'il subit un choc violent comme lorsqu'il tombe sur le sol. Dans ce cas, le verre intact est remplacé par une série de fragments générés par Blender à l'aide de l'outil Cell Fracture, et chacun de ces fragments calcule sa propre collision.

Déformation

La déformation concerne le ballon, qui réagit visuellement aux impacts en s'écrasant avant de retrouver progressivement sa forme initiale, mais le SphereCollider reste fixe pour une performance optimale. Ce système repose sur un système de ressort-ammortisseur avec manipulation des vertices en temps réels où le point d'impact est localisé et seuls les vertices autour de celui-ci sont affectés. Elles subissent une force de rappel proportionnelle à la distance de l'original et cette force est opposée au déplacement.

III. Limites et pistes d'amélioration

En grande partie, nos limites sont liées aux contraintes techniques des outils utilisés. Par exemple la lumière directionnelle à l'infini de Unity agit comme un soleil et rend visible tous les objets et leurs interactions, mais l'utiliser comme unique source de lumière cause une absence de variation dans le niveau d'éclairage des éléments de la scène et un manque de précision dans les ombres.

Une autre limite concerne la cassure du verre, que nous n'avons pas été en mesure de générer sans passer par Blender pour produire les fragments à l'aide de l'outil Cell Fracture. Cette solution fonctionne, mais elle reste entièrement prédéfinie, car peu importe la manière dont le verre tombe ou l'orientation du choc, la cassure se produit toujours selon le même schéma. Une amélioration possible aurait été d'intégrer un système de fracture procédurale dans Unity, permettant au verre de se briser différemment selon la force, l'angle ou la zone d'impact.

IV. Références et licences des assets

Dans notre scène, le verre à vin a été modélisé sur Blender et est donc libre de droit, tandis que les autres objets, notamment le verre (timbale), la table, le ballon et la fourchette, proviennent tous de la plateforme d'assets en ligne TurboSquid. Il s'agit d'assets gratuits mis à disposition sous licence royalty-free, ce qui autorise leur utilisation dans des projets interactifs, académiques ou visuels, tant qu'ils ne sont ni revendus ni redistribués en tant qu'assets indépendants. Les liens vers les modèles utilisés sont les suivants :

- Table : https://www.turbosquid.com/3d-models/3d-coffee-table-2230645?dd_referrer=1
- Ballon : <https://www.turbosquid.com/3d-models/generic-volleyball-ball-low-poly-pbr-2450512>
- Fourchette : <https://www.turbosquid.com/3d-models/old-vintage-fork-1747461>
- Verre: <https://www.turbosquid.com/3d-models/dof-glassware-2026510>