

## TPIntra : Jeu de pool

### Programmation et jeux 420-455-AL

---

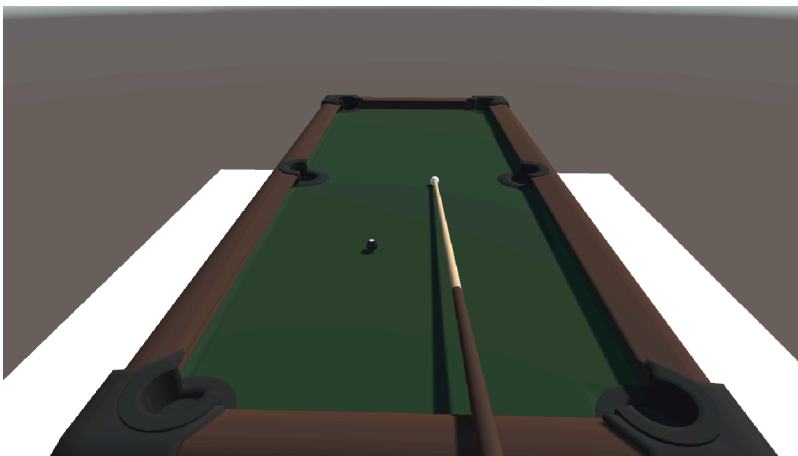
#### 1. Objectifs

- Approfondissement des notions vues en classe sur les jeux en 3D avec Unity
- Bonne utilisation des composantes de physiques de Unity (Rigidbody, Colliders, Physic Materials)
- Bonne utilisation des méthodes utilitaires de Unity (Vector3, Transform, ...)
- Continuer de s'amuser!

---

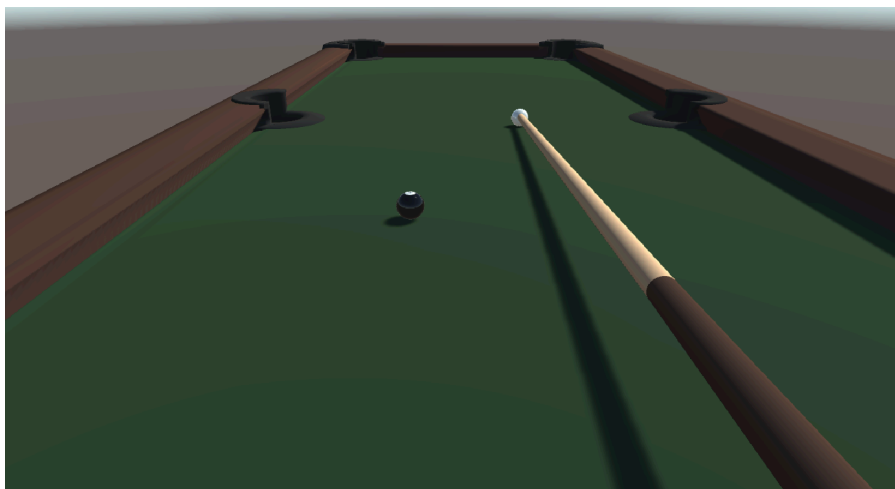
#### 2. Présentation du problème

Vous devez ajouter de la fonctionnalité à un jeu de pool de base. Une table vous est donnée ainsi qu'une queue de billard et 2 boules soit une blanche et une noire.

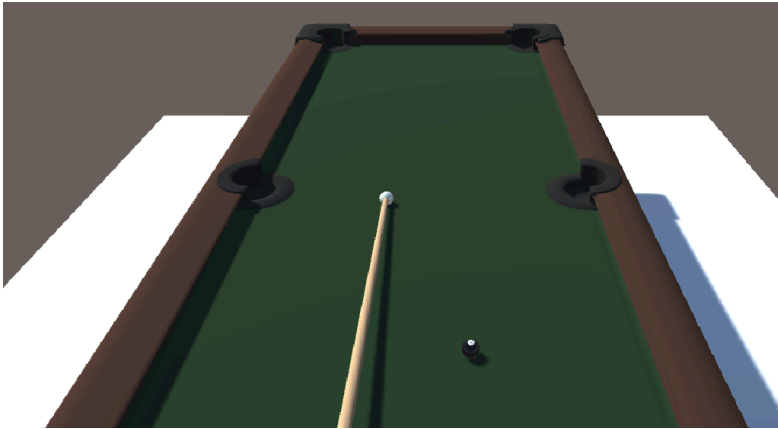


Voici la fonctionnalité attendue de votre jeu

- Vous devez pouvoir vous rapprocher et vous éloigner de la balle blanche à l'aide des clés 'up, down' comme le montre la photo suivante. La méthode static 'Transform.translate()' devrait vous aider à accomplir cela.



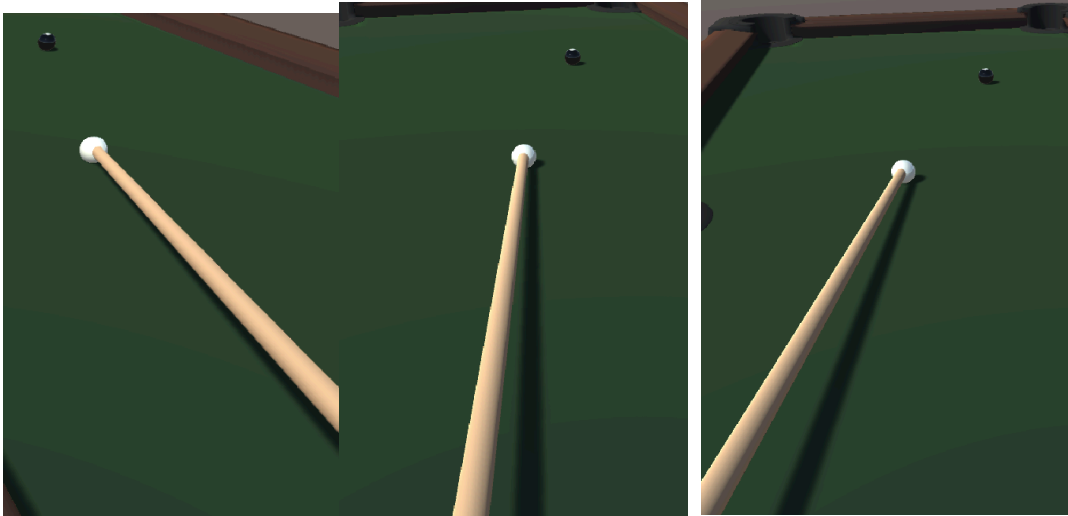
- De plus, vous devriez mettre un modificateur de déplacement (la clé 'shift' ou 'Maj' par exemple) afin que le déplacement n'altère pas la composante 'Y' de la translation. Ceci permettrait d'avancer la vue sans se rapprocher de la table comme le montre la photo ci-dessous.



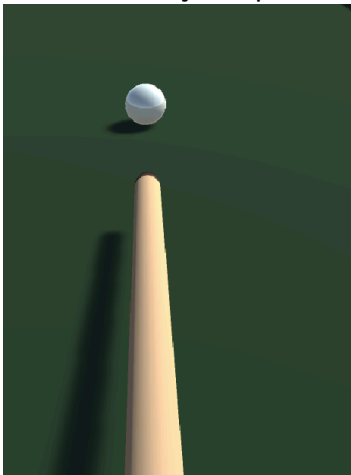
- Vous devez pouvoir tourner autour de la balle blanche à l'aide des clés 'gauche, droite' . Le méthode static 'Transform.RotateAround()' devrait pouvoir faire l'affaire. Voici quelques photos qui montre le résultat attendu.



- Vous devez également mettre un modificateur sur les clés 'gauche, droite' (Par exemple 'shift' ou 'Maj') afin de déplacer de gauche à droite la caméra sans tourner autour de la balle blanche



- 
- Vous devez pouvoir frapper la balle blanche à l'aide de la 'barre d'espacement'
- Plus longtemps vous appuyez sur la barre d'espacement, plus fort sera le coup donné sur la balle. Vous pouvez implanter comme vous voulez cette fonctionnalité mais je vous recommande d'utiliser un 'Rigidbody' et la méthode 'AddForce'. Regardez les différentes façons qu'il est possible d'appeler cette méthode. Cela vous sera utile.



- 
- Le roulement des balles doit être bien simulé. C'est-à-dire que la balle doit pouvoir ralentir sur le tapis et aussi rebondir sur les bandes. On a vu en classe comment simuler ces caractéristiques.



### 3. Critères de correction

Voici le barème de correction qui sera utilisé

<b>Qualité de la scène et du code (script C#)</b>	
Utilisation des scripts adéquats (clean-code, utilisation de script de type 'managers' au besoin)	/25
<b>Fonctionnement</b>	
Bonne utilisation des 'composantes' Unity pour réaliser le jeu	/25
Déplacement de la vue de la caméra adéquate	/20
Tirer sur la balle blanche à l'aide de la barre d'espacement	/20
Conclure la partie (gagnant/perdant)	/10
Bonus - Afficher le score et la force relative du coup sur la balle	/5
Bonus - Utilisation des sons	/5
<b>Total</b>	<b>/100</b>



#### 4. Remise

Le TP est à remettre le 15 Mai à 23h55.

##### POLITIQUE DE RETARD

Tout retard dans la remise d'un travail entraîne une pénalité de 10% par jour de retard jusqu'à concurrence de 5 jours de calendrier scolaire (donc, fin de semaine et congé non compris). Après cette date, la note de 0 sera attribuée au travail.

#### 5. Mot de la fin

Bonne chance à tous! Et comme d'habitude, n'oubliez pas de vous amuser lorsque vous ferez ce travail. Il s'agit de faire un jeu après tout.