

Sprint 2 – Séance de TP [0.5em] **Projet : LO3BA** (ACL-ISN 2025 – Mines Rabat)

1. Backlog Sprint 2

Objectif global

Développer la première version **jouable** du jeu : ajout du personnage principal, de la logique de déplacement, des premières interactions et du système de niveaux (*Level System*).

User Stories

| ID | En tant que... | Je veux... | Afin de... | Priorité |
|-----|----------------|---|--|----------|
| US1 | Joueur | Contrôler le dinosaure avec ← → et ESPACE | Pouvoir me déplacer dans le niveau | Haute |
| US2 | Joueur | Voir une zone de jeu avec un sol et des plateformes | Évoluer dans un environnement clair | Haute |
| US3 | Joueur | Ramasser un objet (ex. pièce, cœur) | Gagner des points / vie | Moyenne |
| US4 | Développeur | Mettre à jour le diagramme UML avec les nouvelles classes (Player, Entity, Level) | Structurer le code | Moyenne |
| US5 | Développeur | Ajouter une boucle de jeu simple (rafraîchissement du rendu) | Gérer les animations / mouvements | Moyenne |
| US6 | Joueur | Voir l'interface HUD (score, vie) | Avoir un retour visuel sur l'état du jeu | Basse |
| US7 | Joueur | Passer d'un niveau à un autre après avoir atteint un objectif | Découvrir une progression de jeu | Haute |
| US8 | Développeur | Créer un système de niveaux (chargement, transition, réinitialisation) | Gérer la structure du jeu sur plusieurs cartes | Haute |

2. Conception (Prévision UML – Sprint 2)

Nouvelles classes prévues

| Classe | Rôle principal | Détails |
|----------------------------------|--|--|
| Player | Représente le personnage du joueur | Gère la position, le saut, la gravité, les collisions. |
| Entity (abstraite) | Base pour tous les objets du jeu | Contient position, taille et méthode <code>update()</code> . |
| GameLoop | Boucle principale d'animation | Appelle <code>update()</code> sur les entités du jeu. |
| HUDController (optionnel) | Gère l'affichage du score et de la vie | Intégré dans la scène de jeu. |
| Level | Représente un niveau du jeu | Contient les entités, la carte, les obstacles, les points d'apparition. |
| LevelManager | Gère la progression entre les niveaux | Charge le prochain niveau, réinitialise les données et gère les transitions. |

Relations UML simplifiées

voir image.png

3. Répartition des tâches (Sprint 2)

| Membre | Rôle / Tâches principales |
|-----------|--|
| Yassine 1 | Implémentation de la classe Player (déplacement, saut, gravité). |
| Yassine 2 | Mise en place du GameLoop et du système de mise à jour continue. |
| Oualid | Création des classes Level et LevelManager + intégration dans le GameController . |
| Wadie | Mise à jour du diagramme UML, documentation et préparation du système de transitions entre niveaux. |

4. Livrables de la séance

- **Backlog Sprint 2** (ce fichier)
- **Conception UML mise à jour** (/docs/uml/LO3BA_sprint2_UML.png)
 - Répartition des tâches (tableau ci-dessus)
 - Ajout du Level System (premiers niveaux jouables)

5. Prochaine étape (préparation Sprint 3)

- Intégration des **ennemis**, **collisions avancées** et **détection de fin de partie**.
- Amélioration du **HUD** (score, vies, niveau actuel).
- Création du **tag Git v2.0** à la fin du sprint.