

Université Abdelhamid Ibn Badis Mostaganem  
Faculté des Sciences Exactes et de l'Informatique  
Département d'Informatique



## Mémoire de fin d'étude en Master informatique

Présentée par :

**Mohammed Touati Sabour**

---

### Conception et développement d'une application web pour Serious Game Collaboratif

---

Filière : Informatique

Option : Ingénierie des Systèmes d'Information

Soutenue le 15 juin 2025 devant un jury composé de :

---

|              |                                    |                            |
|--------------|------------------------------------|----------------------------|
| Président    | <b>M. Sofiane Khat</b>             | (Université de Mostaganem) |
| Examineur    | <b>M. Mohammed Walid Krakallah</b> | (Université de Mostaganem) |
| Encadant     | <b>M. Mohamed Habib Zahmani</b>    | (Université de Mostaganem) |
| Co-Encadrant | <b>M. Wassim Bouazza</b>           | (Université de Nantes)     |

## **Résumé**

Ce projet s'inscrit dans le domaine de l'apprentissage à travers les serious games. Il a pour objectif de développer une application web dédiée à la gestion d'un jeu sérieux intitulé Digital Factory Challenge, conçu pour la formation en logistique et en industrie 4.0. Ce jeu simule un environnement industriel réaliste dans lequel les participants sont amenés à prendre des décisions stratégiques influençant divers paramètres de gestion.

L'application permet au maître de jeu de créer et d'animer des sessions interactives, tandis que les joueurs participent de manière collaborative et immersive. Ce projet vise à renforcer l'engagement des apprenants, à faciliter la compréhension de concepts industriels complexes et à encourager la prise de décision dans un cadre à la fois ludique et structuré.

## **Mots-clés :**

Jeux sérieux, apprentissage par le jeu, industrie et logistique, application web.

## **Abstract**

This project falls within the field of learning through serious games. Its objective is to develop a web application for managing a serious game called Digital Factory Challenge, designed for training in logistics and Industry 4.0. The game simulates a realistic industrial environment where participants are required to make strategic decisions that impact various management parameters.

The application allows the game master to create and run interactive sessions, while players participate in a collaborative and immersive way. This project aims to enhance learner engagement, facilitate the understanding of complex industrial concepts, and encourage decision-making within a structured and game-based environment.

## **Keywords:**

Serious games, game-based learning, industry and logistics, web application..

Je dédie ce travail à toutes les personnes chères à mon cœur  
À mes parents, pour leur amour inconditionnel et leur soutien constant,  
À mes frères et sœurs, compagnons de vie et de souvenirs,  
À l'inconnu connu par mon cœur,  
Et à mes amis.

# Remerciements

La rédaction de cette partie n'est pas une chose facile. En pensant aux personnes qui ont contribué d'une manière ou d'une autre à l'élaboration de ce travail de mémoire de fin d'étude, de nombreux noms me viennent à l'esprit. Seulement, je n'arrive pas à trouver les bons mots pour exprimer ma gratitude et ma reconnaissance envers eux. Je m'excuse auprès des personnes qui j'ai oublié de mentionner, merci pour tout ce que vous avez fait pour moi.

Tout d'abord, j'exprime ma profonde gratitude à mes encadrants, M. Mohamed Habib Zahamani et M. Wassim Bouazza, pour leur accompagnement constant, leur suivi rigoureux, leurs orientations précieuses et les efforts qu'ils ont déployés pour mener ce travail à bien. J'adresse mes remerciements particuliers à M. Mohamed Habib Zahamani pour la confiance qu'il m'a toujours témoignée, ainsi que pour la richesse de ses conseils. Je lui suis également reconnaissant pour sa qualité humaine, sa bienveillance, son hospitalité et le soutien qu'il m'a apporté tout au long de ces deux dernières années.

Je tiens également à remercier chaleureusement l'équipe TechFley, notamment Abdelkader Benzoura, Amine et mon tuteur de stage Ahmed Bahamni, pour leur disponibilité, leurs conseils avisés et leur soutien continu. Leur accompagnement m'a permis d'améliorer mon travail et de progresser tout au long de cette expérience professionnelle.

Je ne saurais finir sans exprimer mes remerciements aux personnes qui me sont les plus proches et les plus chères. À mes parents pour lesquels je suis éternellement reconnaissant pour m'avoir constamment soutenu, pour avoir cru en moi pour mener à bien ce travail, pour avoir mis à ma disposition tous les moyens dont ils disposaient, mais surtout pour leur amour. Merci infiniment !

À mon frère Oussama, pour son soutien indéfectible et ses encouragements constants. À mon frère Ayoub et à mes sœurs, pour leur présence bienveillante.

À une personne très spéciale pour moi, pour son amour infini, sa présence précieuse et son soutien dans les moments les plus importants de ma vie.

Et enfin, merci à tous ceux qui ont participé de près ou de loin à l'aboutissement de ce travail. Je remercie Dieu, le Tout Puissant.

# Liste des figures

|                                                                                               |    |
|-----------------------------------------------------------------------------------------------|----|
| FIGURE 1.1 – ANALOGIE DE LA VINAIGRETTE ET DE L'APPRENTISSAGE PAR LE JEU. ....                | 7  |
| FIGURE 1.2 - LES MECANIKES DE JEU DANS UN SERIOUS GAME .....                                  | 8  |
| FIGURE 1.3 - COPIE D'ECRAN DU SERIOUS GAME "TECHNOCITY" .....                                 | 9  |
| FIGURE 1.4 – JEU DE PLATEAU WARGAMES .....                                                    | 11 |
| FIGURE 1.5 – CRITERES DE CLASSIFICATION DU MODELE G/P/S .....                                 | 13 |
| FIGURE 1.6 - LE SERIOUS GAME THE FRESH CONNECTION.....                                        | 18 |
| FIGURE 1.7 LE SERIOUS GAME ERPSIM.....                                                        | 19 |
| FIGURE 1.8 CYCLE DU DFC .....                                                                 | 21 |
| FIGURE 1.9 – DGC ACTIVITES .....                                                              | 23 |
| FIGURE 3.1 DESCRIPTION DE PROCESSUS UNIFIE. ....                                              | 38 |
| FIGURE 3.2 – PRESENTATION DES DIAGRAMMES UML.....                                             | 39 |
| FIGURE 3.3 – HERITAGE DES ACTEURS .....                                                       | 40 |
| FIGURE 3.4 – DIAGRAMME DE CONTEXTE STATIQUE .....                                             | 41 |
| FIGURE 3.5 – DIAGRAMME DE CAS D’UTILISATION – GESTION DE L’AUTHENTIFICATION .....             | 43 |
| FIGURE 3.6 – DIAGRAMME DE CAS D’UTILISATION – GESTION D'UNE SESSION DE JEU.....               | 44 |
| FIGURE 3.7 – DIAGRAMME DE CAS D'UTILISATION – JOUEUR .....                                    | 44 |
| FIGURE 3.8 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – S’INSCRIRE.....                     | 45 |
| FIGURE 3.9 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – S’AUTHENTIFIER .....                | 46 |
| FIGURE 3.10 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – DEMARRER UNE SESSION .....         | 47 |
| FIGURE 3.11 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – CLOTURER UNE SESSION.....          | 48 |
| FIGURE 3.12 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – REJOINDRE UNE SESSION DE JEU ..... | 49 |
| FIGURE 3.13 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – LANCER LE JEU .....                | 50 |
| FIGURE 3.14 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – LANCER LE JEU .....                | 51 |
| FIGURE 3.15 – DIAGRAMME DE CLASS DE L’APPLICATION .....                                       | 54 |
| FIGURE 4.1– DIAGRAMME DE DEPLOIEMENT DE LA PLATEFORME.....                                    | 62 |
| FIGURE 4.2 PAGE D’ACCUEIL .....                                                               | 63 |
| FIGURE 4.3 INTERFACE D’INSCRIPTION .....                                                      | 63 |
| FIGURE 4.4 INTERFACE DE CONNEXION .....                                                       | 64 |
| FIGURE 4.5 INTERFACE 2FA .....                                                                | 64 |
| FIGURE 4.6 L’INTERFACE D’ACCUEIL APRES AUTHENTIFICATION. ....                                 | 65 |
| FIGURE 4.7 INTERFACE DE DEMARRAGE D’UNE SESSION .....                                         | 66 |
| FIGURE 4.8 INTERFACE D’ATTENTE DES JOUEURS .....                                              | 67 |
| FIGURE 4.9 INTERFACE DASHBOARD OBSERVATION.....                                               | 68 |
| FIGURE 4.10 INTERFACE DE SAISIR UN CODE D’ADHESION .....                                      | 68 |
| FIGURE 4.11 INTERFACE DE SAISIR UN PSEUDONYME.....                                            | 69 |
| FIGURE 4.12 INTERFACE DE QUIZ .....                                                           | 70 |
| FIGURE 4.13 INTERFACE DE DECISION .....                                                       | 70 |
| FIGURE 4.14 INTERFACE DE SIMULATION .....                                                     | 71 |

|                                                                                              |    |
|----------------------------------------------------------------------------------------------|----|
| FIGURE 4.15 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – PARTICIPER AU QUIZ .....          | 75 |
| FIGURE 4.16 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – PARTICIPER A UNE DECISION.....    | 76 |
| FIGURE 4.17 – DIAGRAMME DE SEQUENCE DU CAS D’UTILISATION – PARTICIPER A UNE SIMULATION ..... | 77 |

# Liste des tableaux

|                                                                  |    |
|------------------------------------------------------------------|----|
| TABLEAU 1-1 JEUX SERIEUX DANS L'INDUSTRIE ET LA LOGISTIQUE ..... | 16 |
| TABLEAU 2-1 ACTEURS DE LA SEL .....                              | 25 |
| TABLEAU 2-2 CARACTERISTIQUES DES UTILISATEURS.....               | 29 |
| TABLEAU 3-1 LIST DES CAS D'UTILISATION.....                      | 43 |
| TABLEAU 3-2 PRESENTATION DES CLASSES DE L'APPLICATION .....      | 53 |

## Liste des abréviations

|       |                                                   |
|-------|---------------------------------------------------|
| SGs   | Serious Games                                     |
| DGC   | Digital Factory Challenge                         |
| MJ    | Maître de jeu                                     |
| SEL   | Spécification Exigences Logicielles               |
| IEEE  | Institute of Electrical and Electronics Engineers |
| MVC   | Modèle-Vue-Contrôleur                             |
| HTTPS | Hypertext Transfert Protocol Secure.              |
| UML   | Unified Modeling Language.                        |
| UP    | Unified Process                                   |
| PHP   | Unified Process                                   |
| HTML  | HyperText Markup Language                         |
| CSS   | Cascading Style Sheets                            |
| IDE   | Integrated Developement Environnement             |

# Table des matières

|                                                                                 |           |
|---------------------------------------------------------------------------------|-----------|
| <b>Introduction Générale .....</b>                                              | <b>4</b>  |
| <b>Chapitre 1 Concepts autour du Serious Games.....</b>                         | <b>6</b>  |
| <b>1.1 Introduction .....</b>                                                   | <b>6</b>  |
| <b>1.2 L'apprentissage par le jeu .....</b>                                     | <b>6</b>  |
| <b>1.3 Les mécaniques de jeu.....</b>                                           | <b>8</b>  |
| <b>1.4 Le concept de ludo-éducatif.....</b>                                     | <b>8</b>  |
| 1.4.1 Définition.....                                                           | 8         |
| 1.4.2 Limites .....                                                             | 9         |
| <b>1.5 Le concept de gamification .....</b>                                     | <b>10</b> |
| 1.5.1 Définition.....                                                           | 10        |
| 1.5.2 Gamification et éducation.....                                            | 10        |
| <b>1.6 Les Serious Game .....</b>                                               | <b>11</b> |
| 1.6.1 Définition.....                                                           | 11        |
| 1.6.2 Proposition de définition.....                                            | 12        |
| 1.6.3 Classification des Serious Games .....                                    | 12        |
| <b>1.7 Simulation en milieu industriel .....</b>                                | <b>14</b> |
| <b>1.8 Présentation des jeux sérieux dans l'industrie et la logistique.....</b> | <b>14</b> |
| 1.8.1 Industrie .....                                                           | 14        |
| 1.8.2 Logistique .....                                                          | 15        |
| 1.8.3 The Fresh Connection.....                                                 | 17        |
| 1.8.4 ERPsim.....                                                               | 18        |
| <b>1.9 Présentation de notre jeu DFC .....</b>                                  | <b>20</b> |
| 1.9.1 Objectifs Pédagogiques .....                                              | 20        |
| 1.9.2 Contexte et finalité.....                                                 | 20        |
| 1.9.3 Activités intégrées .....                                                 | 20        |
| 1.9.4 Mécaniques de Jeu intégrées .....                                         | 23        |
| 1.9.5 Apports spécifiques de notre jeu .....                                    | 24        |

|                                                                |                                             |           |
|----------------------------------------------------------------|---------------------------------------------|-----------|
| 1.10                                                           | Conclusion .....                            | 24        |
| <b>Chapitre 2 Spécifications d'exigences de logiciel .....</b> |                                             | <b>25</b> |
| 2.1                                                            | Introduction .....                          | 25        |
| 2.1.1                                                          | Objet.....                                  | 25        |
| 2.1.2                                                          | Portée.....                                 | 25        |
| 2.1.3                                                          | Définitions, acronymes et abréviations..... | 26        |
| 2.1.4                                                          | Vue d'ensemble.....                         | 26        |
| 2.2                                                            | Description générale .....                  | 26        |
| 2.2.1                                                          | Environnement.....                          | 26        |
| 2.2.2                                                          | Fonctions .....                             | 28        |
| 2.2.3                                                          | Caractéristiques des utilisateurs .....     | 29        |
| 2.2.4                                                          | Contraintes .....                           | 29        |
| 2.2.5                                                          | Hypothèses et dépendances .....             | 32        |
| 2.3                                                            | Exigences spécifiques .....                 | 32        |
| 2.3.1                                                          | Exigences des interfaces externes.....      | 33        |
| 2.3.2                                                          | Exigences fonctionnelles .....              | 33        |
| 2.3.3                                                          | Exigences de performance .....              | 35        |
| 2.3.4                                                          | Contraintes de conception .....             | 35        |
| 2.3.5                                                          | Attributs .....                             | 35        |
| 2.3.6                                                          | Autres exigences .....                      | 36        |
| 2.4                                                            | Conclusion .....                            | 36        |
| <b>Chapitre 3 Analyse des besoins et conception .....</b>      |                                             | <b>37</b> |
| 3.1                                                            | Introduction .....                          | 37        |
| 3.2                                                            | Présentation de processus unifié .....      | 37        |
| 3.3                                                            | Présentation de langage UML .....           | 38        |
| 3.3.1                                                          | Définition d'UML .....                      | 38        |
| 3.3.2                                                          | Les diagramme UML.....                      | 38        |
| 3.4                                                            | Analyse des besoins .....                   | 40        |
| 3.4.1                                                          | Acteurs du système .....                    | 40        |
| 3.4.2                                                          | Diagramme de contexte statique .....        | 41        |
| 3.4.3                                                          | Diagramme des cas d'utilisation .....       | 41        |

|                                    |                                            |           |
|------------------------------------|--------------------------------------------|-----------|
| 3.4.4                              | Diagramme de séquence de système .....     | 45        |
| <b>3.5</b>                         | <b>Analyse de conception.....</b>          | <b>51</b> |
| 3.5.1                              | Diagramme de classe de conception.....     | 52        |
| 3.5.2                              | Passage au modèle relationnel .....        | 55        |
| <b>3.6</b>                         | <b>Conclusion .....</b>                    | <b>58</b> |
| <b>Chapitre 4 Réalisation.....</b> |                                            | <b>59</b> |
| <b>4.1</b>                         | <b>Introduction.....</b>                   | <b>59</b> |
| <b>4.2</b>                         | <b>Environnement de travail .....</b>      | <b>59</b> |
| 4.2.1                              | HTML .....                                 | 59        |
| 4.2.2                              | CSS.....                                   | 59        |
| 4.2.3                              | Tailwind CSS .....                         | 59        |
| 4.2.4                              | Java Script.....                           | 60        |
| 4.2.5                              | Alpine JS .....                            | 60        |
| 4.2.6                              | PHP .....                                  | 60        |
| 4.2.7                              | Laravel .....                              | 60        |
| 4.2.8                              | Livewire .....                             | 60        |
| 4.2.9                              | Visual Studio Code.....                    | 61        |
| 4.2.10                             | Draw.io .....                              | 61        |
| <b>4.3</b>                         | <b>Diagramme de déploiement .....</b>      | <b>61</b> |
| <b>4.4</b>                         | <b>Présentation de l'application .....</b> | <b>62</b> |
| 4.4.1                              | Page d'accueil.....                        | 62        |
| 4.4.2                              | Utilisateur.....                           | 63        |
| 4.4.3                              | Maitre de jeu .....                        | 65        |
| 4.4.4                              | Joueur.....                                | 68        |
| <b>4.5</b>                         | <b>Conclusion .....</b>                    | <b>71</b> |
| <b>Conclusion Générale .....</b>   |                                            | <b>72</b> |
| <b>Bibliographie .....</b>         |                                            | <b>73</b> |
| <b>Annexe .....</b>                |                                            | <b>75</b> |

# Introduction Générale

Depuis toujours, l'homme utilise le jeu, d'une manière tout à fait naturelle, comme un moyen pédagogique pour contribuer à son développement affectif, sensoriel, moteur, cognitif, intellectuel et social. De nos jours, l'apprentissage par le jeu a gardé tout son sens et se décline sous des formes particulièrement variées avec l'avènement des jeux vidéo.

De nombreux travaux de recherche ont démontré que les jeux vidéo conçus avec un but éducatif explicite, appelés Serious Games, pouvaient se montrer plus efficaces que les méthodes traditionnelles pour l'acquisition de compétences complexes dans la formation supérieure et en entreprise. Cependant, malgré leurs nombreux atouts, les Serious Games ne sont aujourd'hui que très peu utilisés dans l'enseignement, et ce principalement en raison de leur coût élevé mais également en raison des difficultés souvent rencontrées pour concevoir de tels outils et ensuite les intégrer dans une formation pédagogique.

Face à ce constat, nous avons souhaité apporter une contribution concrète à travers le développement d'une application web dédiée à la gestion d'un Serious Game collaboratif, ainsi que la conception d'un jeu sérieux. Ce jeu, inspiré des enjeux de l'industrie 4.0 et de la logistique, permet aux apprenants d'évoluer dans un environnement simulé où ils doivent prendre des décisions stratégiques et collaboratives.

Ce mémoire s'inscrit dans une démarche à la fois pédagogique et technologique, visant à explorer les apports des jeux sérieux dans l'apprentissage, tout en proposant une solution opérationnelle adaptée aux besoins des formateurs et des apprenants.

Il est structuré en quatre chapitres :

- **Chapitre 1** : Introduction aux Serious Games et à leur usage dans l'industrie et la logistique, avec une présentation de notre solution pédagogique.
- **Chapitre 2** : Concerne la documentation des spécifications des exigences logicielles.

- **Chapitre 3** : Concerne les différentes étapes de la modélisation et la conception de notre solution proposée à travers les différents diagrammes UML.
- **Chapitre 4** : Décrit les différentes étapes de la réalisation et l'implémentation de la solution proposée suivi de l'inventaire des technologies utilisées et des captures d'écran.

Nous clôturons ce mémoire par une conclusion générale qui résume l'ensemble de la réalisation du projet et exposant quelques perspectives de développement de notre plateforme.

# **Chapitre 1**

## **Concepts autour du Serious Games**

### **1.1 Introduction**

Dans un monde en constante évolution, où la formation doit s'adapter à des défis toujours plus complexes, les Serious Games s'imposent comme une solution pédagogique innovante.

Dans ce chapitre, nous présenterons tout d'abord les fondements théoriques et conceptuels essentiels à la compréhension de notre travail, ainsi que des exemples de Serious Games appliqués à l'industrie et à la logistique. Dans un second temps, nous décrirons notre propre solution pédagogique : un Serious Game développé dans le cadre de ce projet.

### **1.2 L'apprentissage par le jeu**

L'apprentissage par le jeu, aussi appelé Game-based learning, s'appuie sur le potentiel ludique des jeux pour renforcer l'implication des apprenants dans le processus pédagogique. Cette approche pédagogique vise à attirer l'attention des élèves et à les encourager à participer activement à l'apprentissage en exploitant l'attrait naturel et l'engagement inhérents aux jeux [1]. Contrairement à l'enseignement traditionnel, où les élèves sont souvent placés dans une position passive, le jeu l'invite à devenir un véritable acteur de sa formation. Par le biais du jeu, le rôle actif de l'apprenant est primordial. L'apprenant est placé dans une situation où ses choix, ses actions et ses décisions ont un impact direct, favorisant ainsi une meilleure appropriation des savoirs et compétences. De plus, comme le montre une étude du National Research Council (2000), l'apprenant se retrouve engagé émotionnellement dans le jeu, ce qui facilite l'imprégnation de ses actions et décisions dans sa mémoire. Par ailleurs, cette méthode pédagogique s'avère particulièrement efficace pour développer certaines compétences qui résistent aux approches plus traditionnelles tel que telles que la gestion de la chaîne logistique, l'optimisation des flux de production, la prise de décision en environnement incertain, ou encore la gestion des stocks et la planification des ressources. Elle permet, entre autres, de travailler sur des savoir-faire pratiques, des capacités de résolution de

problèmes ou encore des compétences en communication et en collaboration [2]. Mais, pour que l'apprenant ressente une motivation intrinsèque et soit totalement engagé dans sa formation, les activités pédagogiques doivent être soigneusement tissées avec le scénario du jeu. La notion de motivation intrinsèque implique que l'apprenant est tellement absorbé par le jeu qu'il ressent une stimulation personnelle et intérieure qui le pousse à progresser, sans l'intervention de récompenses ou de punitions externes. [3]

“ Le jeu éducatif fait diminuer la notion d'effort en cachant la partie éducative d'activité. Il devrait être distrayant, sans contrainte perceptible, axé sur les apprentissages. Le jeu est alors vu comme vecteur de motivation. “

La Figure 1 présente l'analogie entre l'apprentissage par le jeu et l'élaboration d'une bonne vinaigrette. Les contraintes liées à l'apprentissage (acquisition de compétences, entraînement, évaluation) et les caractéristiques du jeu (action, émotion, engagement, attractivité) doivent être bien dosées et mélangées pour stimuler l'appétit d'apprendre et ainsi inciter les élèves à travailler par eux-mêmes. Et tout comme pour l'élaboration d'une bonne vinaigrette, la conception d'une formation ludique est loin d'être évidente.

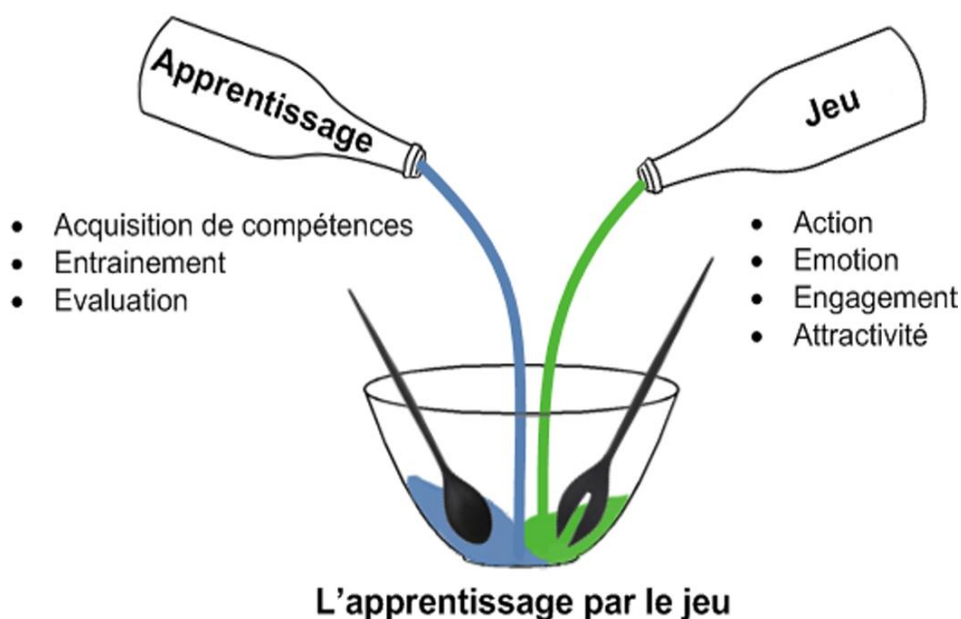


Figure 1.1 – Analogie de la vinaigrette et de l'apprentissage par le jeu.

## 1.3 Les mécaniques de jeu

Les mécaniques de jeu sont constituées des règles, principes et systèmes interactifs qui déterminent le fonctionnement du jeu et la façon dont le joueur interagit avec lui. Elles englobent toutes les actions envisageables, les buts à atteindre, les défis à surmonter et les prix à gagner dans le cadre du jeu. Parmi les mécaniques courantes, on peut citer :

- La progression (ex : passer des niveaux, accumuler des points d'expérience).
- Les récompenses (ex : gagner des points, obtenir des badges ou des objets).
- Les défis (ex : surmonter des obstacles, résoudre des énigmes).
- La coopération ou la compétition entre joueurs.
- La gestion de ressources (ex : temps, argent virtuel, énergie).



Figure 1.2 - Les mécaniques de jeu dans un serious game

## 1.4 Le concept de ludo-éducatif

### 1.4.1 Définition

Le terme ludo-éducatif fait référence à des méthodes pédagogiques qui incorporent des éléments ludiques (issus du jeu) pour encourager la motivation et simplifier le processus d'apprentissage. Dès les années 1980 et 1990, ces logiciels ont vu le jour dans le but de rendre l'apprentissage plus attractif, en utilisant le jeu comme levier d'engagement.

La plupart des logiciels ludo-éducatifs reposent sur des théories behavioristes, qui postulent que l'apprentissage est favorisé par des systèmes de récompense. Dans ce cadre, l'apprenant effectue une tâche d'apprentissage (exercice, quiz, etc.) et obtient en retour un moment de jeu perçu comme une récompense extrinsèque. L'aspect ludique est alors séparé de l'aspect éducatif. [4]

### 1.4.2 Limites

Ces logiciels ont permis d'intégrer le jeu dans l'éducation, mais leur efficacité est limitée :

- Le fait de séparer les activités considérées comme utiles (apprentissage) et celles perçues comme plaisantes (jeu) peut conduire les apprenants à se désengager des contenus éducatifs.
- Les élèves peuvent se focaliser sur la partie ludique, au détriment de l'objectif pédagogique.

Un exemple représentatif de cette approche est le serious game Technocity (2006), en distinguant nettement le temps de jeu récréatif de la phase d'apprentissage (visionnage d'une vidéo et réponse à des questions). Ce type de conception est dit extrinsèque, car l'élément ludique n'est pas directement impliqué dans le processus d'apprentissage.



Figure 1.3 - Copie d'écran du serious game "Technocity"

## **1.5 Le concept de gamification**

### **1.5.1 Définition**

La gamification, ou ludification en français, représente l'intégration de mécanismes de jeu dans des contextes non ludiques afin de stimuler la motivation, l'engagement ou l'apprentissage. A l'inverse du serious game qui associe une visée utilitaire à du jeu. La gamification ajoute des éléments de jeu à une activité existante, qui n'est pas un jeu à l'origine. À titre d'exemple, une agence de communication a transformé un escalier de métro en piano géant pour inviter les usagers à en gravir les marches et faire de l'exercice, plutôt que d'emprunter l'escalier mécanique.

### **1.5.2 Gamification et éducation**

Dans le secteur éducatif, la gamification est de plus en plus utilisée pour augmenter l'implication et la motivation des apprenants, en intégrant des logiques issues du jeu.

Un exemple emblématique de gamification dans le milieu éducatif est Classcraft. Inspiré des jeux de rôle en ligne comme World of Warcraft, cette application permet aux élèves de former des guildes, où chacun incarne un personnage avec des pouvoirs (chevalier, mage, guérisseur, etc.). L'enseignant joue le rôle de maître du jeu : il attribue ou retire des points en fonction des comportements et des résultats scolaires. Par exemple :

- Aider un camarade ou poser une question pertinente fait gagner des points.
- Arriver en retard ou interrompre la classe peut en faire perdre, affectant toute l'équipe.

Ce système favorise la coopération, valorise les attitudes positives et régule les comportements perturbateurs.

Éric Sanchez, spécialiste des jeux sérieux, note que cette approche peut renforcer la motivation, au point que certains élèves acceptent de venir en cours le samedi pour rattraper leur retard, ce qu'ils refusaient auparavant. [5]

## 1.6 Les Serious Game

### 1.6.1 Définition

Il existe de nombreuses définitions du serious game. L'une des plus synthétique est celle proposée par les concepteurs de jeux vidéo Chen et Michael (2005) : « Tout jeu dont la finalité première est autre que le simple divertissement. » D'après cette définition, un Serious Game s'applique potentiellement à une multitude de secteurs : éducation, santé, publicité, communication, politique, humanitaire, défense, religion, art..., ect. Et peut s'appuyer sur tout type de support.

Les SGs peuvent avoir un support papier. à l'image des premiers jeux qualifiés ainsi dans le domaine éducatif, dès les années 1960-1970. Ces jeux prenaient alors la forme de jeux de cartes, de plateau, de rôle, voire de jeux en plein air [2]. Bien entendu, l'origine des jeux sérieux non numériques est bien plus ancienne. Dès le XIXe siècle, les wargames étaient utilisés pour former les officiers à la stratégie militaire, tandis que le jeu d'échecs, apparu il y a plus de 1500 ans.



Figure 1.4 – Jeu de plateau Wargames

Les SGs aussi se présenter sous une forme numérique. C'est de loin le cas le plus fréquent aujourd'hui. L'effervescence actuelle autour du serious game tient, d'ailleurs, très probablement au support numérique qui ouvre de nombreuses perspectives en termes d'innovations, que ce soit dans la manière de jouer, d'enseigner, d'apprendre, d'échanger des données.

### 1.6.2 Proposition de définition

On peut définir le serious game comme une est un dispositif, numérique ou non, dont l'intention initiale est de combiner, avec cohérence, à la fois des aspects utilitaires (serious) tels, de manière non exhaustive et non exclusive, l'enseignement, l'apprentissage, la communication, ou encore l'information, avec des ressorts ludiques issus du jeu, vidéoludique ou non ("game"). Une telle combinaison qui s'opère par la mise en place d'un scénario utilitaire, qui sur le plan informatique correspond à implémenter une interface graphique, un habillage sonore, une narration et des règles adaptées, a donc pour but de s'écarter du simple divertissement. Cette définition peut être résumée par la mise en relation suivante :

$$\text{Serious game} = \text{scénario utilitaire} + \text{jeu}$$

Dans le cadre de ce mémoire, nous nous intéressons uniquement aux serious games numériques appliqués au domaine de l'éducation.

### 1.6.3 Classification des Serious Games

Parmi les modèles de classification des serious games, le modèle G/P/S constitue une synthèse issue d'une analyse comparative des systèmes de classification des jeux vidéo et des serious games [6]. Ce modèle repose sur trois sur trois critères qui permettent d'appréhender à la fois les aspects ludiques et sérieuses des SGs :

1. **Gameplay** : ce critère porte sur la nature du jeu. S'agit-il d'un jeu avec ou sans objectif précis ? Quelles en sont les règles ?... On peut distinguer en particulier deux catégories d'artefacts ludiques : les jeux et les jouets.

Les « jeux » sont caractérisés par des règles et des objectifs permettant de déterminer si le joueur a « gagné » ou « perdu » à la fin de la partie. Par exemple, Monopoly (Charles Darrow, 1935) est un jeu emblématique : il s'agit d'un objet qui est accompagné de règles

claires, décrites dans un livret, qui expliquent la manière de jouer, ainsi que les conditions permettant de déterminer la victoire ou la défaite de chaque joueur.

À l’opposé, les « jouets » se caractérisent par des règles plus ou moins informelles, mais sans objectif explicite à atteindre. Ils ne permettent donc pas de déterminer une victoire ou une défaite, ni même, dans certains cas, d’identifier une véritable fin de partie. Par exemple, la poupée Barbie (Ruth Handler, 1959) n’est accompagnée d’aucune règle de jeu c’est le joueur lui-même qui définit les actions et les scénarios qu’il souhaite explorer, en toute liberté.

2. **Permet de** : ce critère renseigne sur les fonctions visées par le jeu (fonctions, détaillées dans le schéma ci-contre, qui ne sont pas exclusives et peuvent se combiner) : - diffuser un message (éducatif, informatif, marketing...) ; - dispenser un entraînement (physique ou cognitif) ; - favoriser l’échange de données.
3. **Secteur** : ce critère informe sur le type de public et de marché ciblés par le jeu.

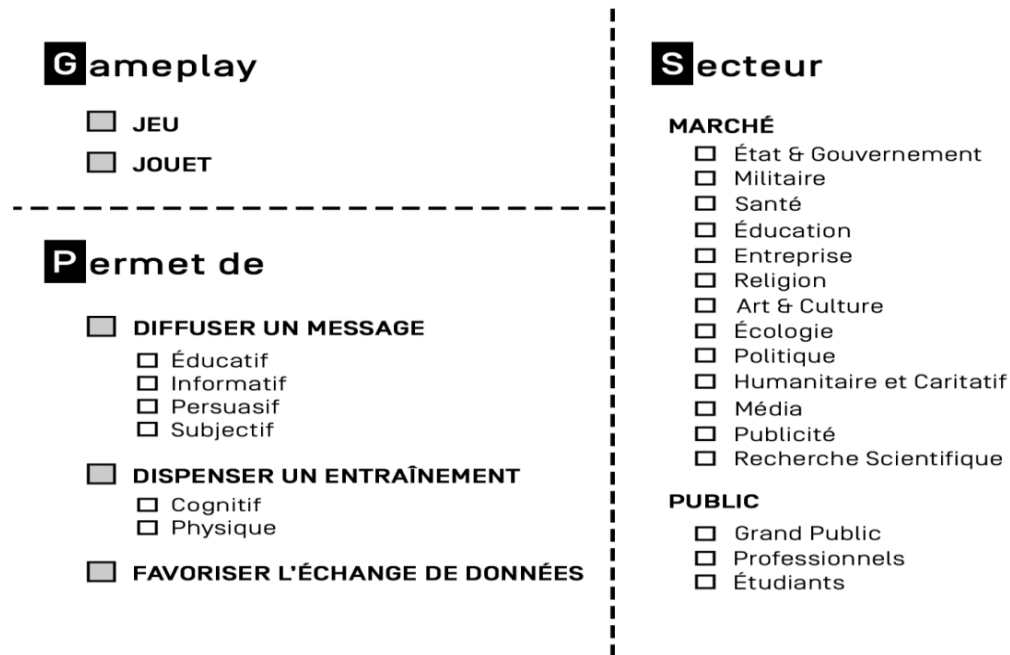


Figure 1.5 – Critères de classification du modèle G/P/S

## **1.7 Simulation en milieu industriel**

La simulation est une méthode pédagogique qui vise à reproduire un environnement réel ou fictif, afin de permettre aux participants d'interagir avec ce système, en prenant des décisions et de voir leurs conséquences. Elle est basée sur un modèle simplifié de la réalité, intégrant des variables et des règles qui déterminent le fonctionnement du système simulé.

Dans le cadre des jeux sérieux appliqués au milieu industriel, la simulation occupe une place centrale en tant qu'outil d'apprentissage expérientiel. Elle immerge les apprenants dans des situations proches de la réalité professionnelle, où ils sont amenés à gérer des ressources, faire face à des contraintes opérationnelles et prendre des décisions stratégiques. En recréant des environnements industriels complexes tels que la production, la logistique, la maintenance ou la qualité, la simulation permet de mieux appréhender les dynamiques systémiques, d'anticiper les conséquences des actions entreprises et d'adapter les stratégies en temps réel.

## **1.8 Présentation des jeux sérieux dans l'industrie et la logistique**

Dans cette section, avant de présenter une sélection de serious games largement utilisés dans les domaines industriel et logistique, nous commencerons par définir ces deux concepts.

### **1.8.1 Industrie**

L'industrie se réfère à l'ensemble des activités économiques qui transforment des matières premières en biens manufacturés grâce à des procédés techniques et des moyens humains et matériels.

Elle joue un rôle central dans le développement économique en produisant des biens destinés à la consommation ou à l'investissement.

L'industrie regroupe :

- Les activités de production de biens matériels (ex : fabrication de voitures, textiles, produits pharmaceutiques),
- Les industries extractives (ex : extraction de minerais),
- Les industries de transformation (ex : agroalimentaire, chimie),

- Les industries énergétiques. [7]

## 1.8.2 Logistique

La logistique fait référence à toutes les opérations qui assurent la gestion efficace des flux de matériaux, d'informations et de produits, allant des fournisseurs jusqu'aux clients finaux.

Elle vise à optimiser les opérations de transport, de stockage, d'approvisionnement, et de distribution tout en réduisant les coûts et les délais.

La logistique couvre :

- La gestion des stocks,
- La planification des approvisionnements,
- L'optimisation des transports,
- La gestion des entrepôts,
- La coordination entre les différents acteurs de la chaîne d'approvisionnement (supply chain). [8]

Le tableau ci-dessous donne un aperçu comparatif de ces jeux :

| <b>Jeu</b>           | <b>Type</b>     | <b>Thème</b>            | <b>Public</b>               | <b>Description brève</b>                                                           |
|----------------------|-----------------|-------------------------|-----------------------------|------------------------------------------------------------------------------------|
| The Fresh Connection | Numérique (web) | Supply chain management | Étudiants et professionnels | Simulation d'entreprise en ligne axée sur la coordination de la chaîne logistique. |
| The Cool Connection  | Numérique (web) | Supply chain finance    | Étudiants et professionnels | Jeu de gestion qui relie décisions supply chain et impacts financiers.             |
| The Blue Connection  | Numérique (web) | Économie circulaire     | Étudiants et professionnels | Simulation visant à transformer un modèle linéaire en modèle circulaire rentable.  |

|                        |                       |                                 |                             |                                                                                       |
|------------------------|-----------------------|---------------------------------|-----------------------------|---------------------------------------------------------------------------------------|
| Beer Distribution Game | Hybride (plateau/web) | Effet coup de fouet             | Tous publics                | Illustration des dynamiques de stocks et coordination dans une chaîne logistique.     |
| Business on the Move   | Plateau               | Logistique globale              | Jeunes adultes              | Jeu de société sur le transport multimodal et l’empreinte carbone.                    |
| Littlefield Labs       | Numérique (web)       | Gestion de production           | Étudiants et professionnels | Simulation d’usine virtuelle pour apprendre files d’attente et planification.         |
| Plantville             | Numérique (en ligne)  | Gestion d’usine                 | Étudiants et professionnels | Jeu Siemens simulant la direction et l’amélioration continue d’usines.                |
| SCM Globe              | Numérique (web)       | Conception de réseau logistique | Étudiants et professionnels | Simulation en ligne pour concevoir et optimiser des supply chains mondiales.          |
| Factory I/O            | Numérique (3D)        | Automatisation industrielle     | Étudiants et professionnels | Environnement 3D pour pratiquer la programmation et le pilotage de lignes virtuelles. |
| ERPsim                 | Numérique (SAP ERP)   | Gestion intégrée d’entreprise   | Étudiants et professionnels | Simulation utilisant SAP ERP pour gérer une entreprise manufacturière.                |

Tableau 1-1 Jeux sérieux dans l’industrie et la logistique

Parmi ces jeux, deux en particulier ont retenu notre attention en raison de leur richesse pédagogique et de leur proximité avec les objectifs de notre projet :

### **1.8.3 The Fresh Connection**

The Fresh Connection est un serious game collaboratif développé par Inchainge, largement utilisé à l'échelle internationale dans les formations en gestion de la chaîne logistique. Il place les participants dans la peau des dirigeants d'une entreprise fictive spécialisée dans les produits alimentaires périssables (jus de fruits), confrontée à un déclin de sa performance financière. Le but principal est de redresser cette situation en prenant des décisions stratégiques coordonnées dans différents départements de l'entreprise. [9]

#### **1.8.3.1 Objectifs pédagogiques**

Ce jeu vise à :

- Rompre avec un fonctionnement en silos pour promouvoir une approche transversale et collaborative.
- Améliorer la communication entre les départements de l'entreprise et favoriser la remontée d'informations critiques.
- Développer les compétences en prise de décision collective et en gestion de compromis.
- Traduire une stratégie globale en actions opérationnelles concrètes.
- Appliquer les principes du MRP II (Manufacturing Resource Planning) ;
- Mettre en évidence l'impact direct de la Supply Chain sur la rentabilité de l'entreprise.

#### **1.8.3.2 Fonctionnement du jeu**

Les participants sont répartis en équipes de quatre, chaque membre occupant un poste stratégique:

- Directeur des achats,
- Directeur des ventes,
- Directeur de la production
- Directeur de la chaîne logistique.

Ensemble, ils doivent collaborer pour améliorer la rentabilité de l'entreprise, mesurée à travers l'indicateur financier du ROI (Return On Investment).

Le jeu se déroule en plusieurs cycles (appelés rounds), au cours desquels les équipes analysent les résultats précédents, ajustent leurs choix stratégiques et observent l'effet de leurs décisions. Ce processus itératif permet d'ancrer les concepts théoriques dans une dynamique de simulation réaliste. La Figure 1.6 illustre une capture d'écran du serious game The Fresh Connection.



Figure 1.6 - Le Serious game The Fresh Connection

## 1.8.4 ERPsim

ERPsim est un jeu sérieux développé par HEC Montréal, conçu pour initier les étudiants et les professionnels à l'usage concret des systèmes ERP, en particulier SAP ERP, qui propose plusieurs scénarios sectoriels (industrie alimentaire, énergie, distribution, etc.) et niveaux de complexité (initiation, avancé, expert), dans un environnement dynamique et compétitif. Il s'agit d'un des rares serious games intégrant un véritable progiciel de gestion intégré, permettant une immersion totale dans les processus de gestion d'entreprise. [10]

### 1.8.4.1 Objectifs pédagogiques

Ce jeu vise à :

- Rompre avec un fonctionnement en silos pour promouvoir une approche transversale et collaborative.
- Améliorer la communication entre les départements de l'entreprise et favoriser la remontée d'informations critiques.

- Développer les compétences en prise de décision collective et en gestion de compromis.
- Traduire une stratégie globale en actions opérationnelles concrètes.
- Appliquer les principes du MRP II (Manufacturing Resource Planning) .
- Mettre en évidence l'impact direct de la Supply Chain sur la rentabilité de l'entreprise.

#### 1.8.4.2 Fonctionnement du jeu

Les participants sont répartis en équipes, chacune gérant une entreprise virtuelle dans un marché concurrentiel. Grâce à l'interface de SAP ERP, chaque équipe doit :

- Planifier la production,
- Gérer les approvisionnements et les stocks,
- Fixer les prix de vente,
- Analyser les données financières en temps réel,
- Réagir aux évolutions du marché.

Les décisions sont prises en temps réel, et chaque action dans le système ERP a des conséquences immédiates. Cette dynamique permet de comprendre les interdépendances entre les fonctions d'une entreprise, et de développer une vision globale de la gestion d'entreprise. La Figure 1.6 illustre une capture d'écran du serious game ERPsim.

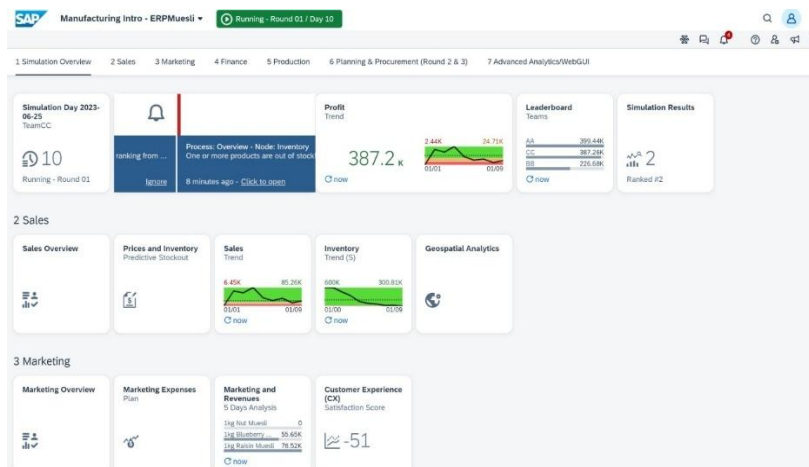


Figure 1.7 Le Serious game ERPsim

## **1.9 Présentation de notre jeu DFC**

Après avoir défini les concepts clés liés à l'industrie, à la logistique et aux serious games, et présenté deux jeux de référence dans ce domaine (The Fresh Connection et ERPsim), nous introduisons ici notre propre solution pédagogique : Digital Factory Challenge (DFC).

### **1.9.1 Objectifs Pédagogiques**

Le Digital Factory Challenge poursuit plusieurs objectifs éducatifs :

- Consolider les connaissances théoriques des étudiants dans le domaine de la logistique et de l'industrie 4.0.
- Développer les compétences en prise de décision stratégique,
- Favoriser la collaboration et le travail en équipe,
- Permettre une mise en pratique des connaissances dans un environnement simulé proche du réel.

### **1.9.2 Contexte et finalité**

Le DFC est un jeu sérieux conçu pour initier les apprenants aux enjeux stratégiques de l'Industrie 4.0 dans un environnement immersif et interactif. Destiné à un public académique diversifié (BUT, Master, Doctorat), il combine connaissances théoriques, prises de décision concrètes et résultats simulés, dans le but de renforcer les compétences opérationnelles et décisionnelles des participants.

### **1.9.3 Activités intégrées**

Le jeu se structure autour de trois activités principales, organisées selon un cycle pédagogique répété un certain nombre de fois.



Figure 1.8 Cycle du DFC

### 1.9.3.1 Activité 1 : Quiz

L'activité Quiz a pour objectif de consolider les connaissances théoriques des joueurs dans des domaines clés de l'industrie 4.0 et logistique. Les participants doivent répondre à des questions à choix unique (QCU) et à choix multiples (QCM) portant sur :

- Le contrôle de qualité,
- La gestion de la production,
- La maintenance industrielle
- La logistique industrielle.
- L'optimisation des processus.
- La durabilité environnementale

Chaque bonne réponse récompense le joueur en augmentant son cash disponible pour les futures décisions. Ainsi, les connaissances théoriques ont un impact direct sur la capacité d'investissement et de gestion dans le jeu.

### 1.9.3.2 Activité 2 : Décision

Lors de l'activité Décision, les joueurs doivent allouer leur capital à l'amélioration de huit technologies clés de l'usine numérique, chacune divisée en quatre niveaux de développement :

- Robotics

- Artificial Intelligence
- Augmented Reality
- Horizontal Integration
- Vertical Integration
- Big Data
- Green Energy
- Cybersecurity

Chaque investissement impacte un ou plusieurs paramètres industriels critiques, tels que :

- Le temps de production,
- Le coût de la main-d'œuvre,
- Le taux de pannes,
- La qualité de production,
- Les émissions de CO<sub>2</sub>,
- La résilience aux cyberattaques, etc.

Cette activité développe la capacité d'analyse et la projection stratégique des joueurs.

### **1.9.3.3 Activité 3 : Simulation**

L'activité Simulation confronte les joueurs aux conséquences réelles de leurs décisions et connaissances. Le simulateur modélise :

- Les flux de production,
- Les pannes machines,
- La gestion des stocks,
- Les incidents qualité,
- Les cyberattaques,
- Les indicateurs financiers et environnementaux.

Les résultats affichés incluent par exemple :

- Produits fabriqués, vendus, stockés ou jetés,
- Nombre de pannes et durée totale d’arrêt,
- Nombre de cyberattaques et taux de réussite,

Les joueurs analysent ensuite ces retours et sont redirigés vers l’activité Décision pour ajuster leur stratégie.

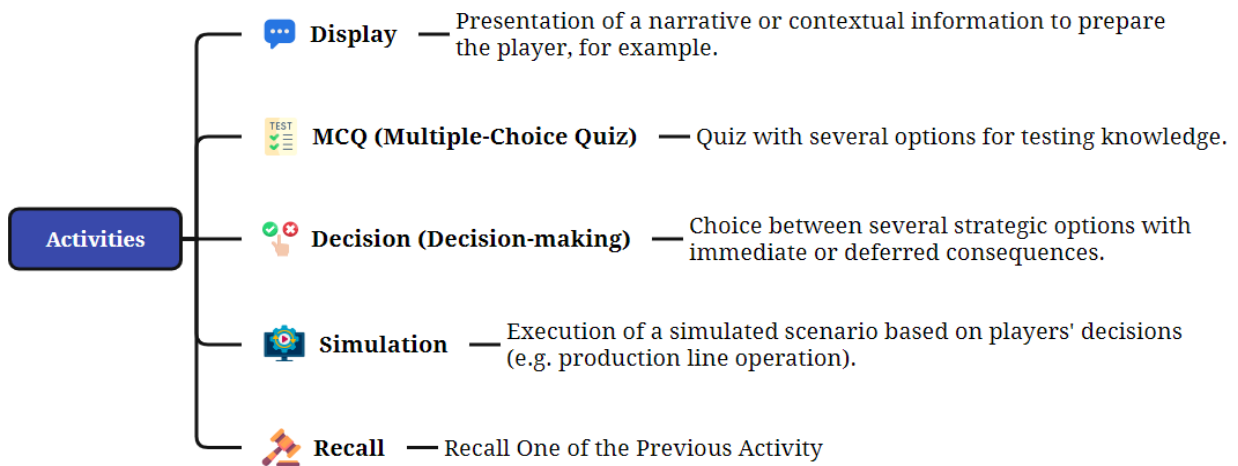


Figure 1.9 – DGC activités

### 1.9.4 Mécaniques de Jeu intégrées

Les mécanismes de jeu intégrés sont conçus pour encourager, engager et orienter les joueurs durant toute la partie :

- Système de points et de récompenses (classements),
- Choix à conséquences influençant la progression du jeu,
- Feedback en temps réel pour renforcer l'engagement.

### 1.9.5 Apports spécifiques de notre jeu

Le DFC propose une approche intégrée de l'apprentissage, où chaque phase du jeu renforce les compétences transversales des apprenants :

- **Théorique** : Validation des connaissances par le quiz.
- **Stratégie** : Prise de décision raisonnée selon les contraintes industrielles.
- **Pratique** : Visualisation concrète des impacts via la simulation.

De plus, en liant directement la réussite au quiz à la capacité d'investissement, il crée un lien immédiat entre savoir, action et performance ce qui favorise :

- L'engagement actif,
- La compréhension des conséquences industrielles,
- La responsabilisation des joueurs dans leur parcours d'apprentissage.

## 1.10 Conclusion

Dans ce chapitre, nous avons défini les concepts fondamentaux liés aux jeux sérieux, en précisant leurs caractéristiques, leurs objectifs pédagogiques, ainsi que les différentes approches utilisées dans ce domaine. Nous avons également clarifié la distinction entre serious game, jeu éducatif et gamification, tout en présentant les mécaniques de jeu intégrées et leur impact sur la motivation et l'apprentissage. Enfin, nous avons présenté notre serious game DGC, en détaillant ses activités pédagogiques et ses objectifs éducatifs.

Dans le chapitre suivant, nous aborderons la documentation des spécifications des exigences logicielles, essentielle pour définir les besoins et les fonctionnalités attendues du système.

## Chapitre 2

# Spécifications d'exigences de logiciel

### 2.1 Introduction

Dans ce chapitre, nous allons définir les spécifications des exigences relatives à notre projet.

#### 2.1.1 Objet

L'objet de ce SEL est de définir l'ensemble des exigences fonctionnelles et non fonctionnelles de l'application web pour Serious Game Collaboratif, destiné aux étudiants en logistique et en industrie 4.0.

La présente SEL est destiné aux acteurs suivants :

| Chef de projet |                       | Développeur            |
|----------------|-----------------------|------------------------|
| Wassim Bouazza | Mohamed Habib Zahmani | Mohammed Touati Sabour |

Tableau 2-1 Acteurs de la SEL

#### 2.1.2 Portée

PlayLearn est une application web pour destinée à la gestion du serious game Digital Factory Challenge. Conçue pour offrir une expérience utilisateur riche et interactive, intégrant l'ensemble des fonctionnalités nécessaires à un apprentissage actif.

Elle permet notamment :

- La création et la configuration de sessions de jeu par un maître de jeu.
- La participation en temps réel de joueurs à des sessions de jeu.

- L’analyse des performances des joueurs, avec des feedbacks.

L’objectif principal de PlayLearn est de proposer une expérience éducative ludique, immersive et complète, spécifiquement adaptée aux besoins des étudiants en logistique et en industrie.

### **2.1.3 Définitions, acronymes et abréviations**

Les termes suivants sont utilisés tout au long de cette SEL avec les définitions associées :

- SEL : Spécification d'Exigences de Logiciel.
- Maître de jeu : Utilisateur responsable du lancement et du suivi d’une session.
- Player : Participant à une session de jeu.
- Session : Partie active du Digital Factory Challenge lancée par un maître de jeu.
- DFC : Digital Factory Challenge, le jeu sérieux implémenté sur la plateforme.

### **2.1.4 Vue d’ensemble**

Ce document décrit les méthodes recommandées pour préparer les spécifications des exigences logicielles. Il comprend une description générale de cette plateforme ainsi que la définition des exigences spécifiques.

## **2.2 Description générale**

### **2.2.1 Environnement**

PlayLearn est une application web autonome, mais elle peut être intégrée dans un environnement pédagogique plus large tel qu’un système de gestion de l’apprentissage (LMS) ou un dispositif de formation universitaire.

#### **2.2.1.1 Interfaces avec le système**

PlayLearn est conçue pour fonctionner de manière autonome, mais elle propose également des interfaces permettant son intégration avec des systèmes externes d’authentification.

- Une interface est prévue avec les applications d’authentification à deux facteurs (2FA) basées sur le protocole TOTP, telles que Google Authenticator.
- Une interface est également prévue avec le système d’authentification de Google, via le protocole OAuth 2.0.

Ces interfaces sont prévues mais non essentielles au fonctionnement de base.

### **2.2.1.2 Interfaces avec les utilisateurs**

La plateforme propose plusieurs interfaces selon le contexte et le rôle de l'utilisateur. Chaque utilisateur peut cumuler les rôles de joueur et de maître de jeu.

- Interface Visiteur (non authentifié) : page d'accueil publique présentant les options de connexion ou de participation.
- Interface de connexion : accès via email/mot de passe ou compte Google.
- Interface d'inscription : création d'un compte personnel pour maître de jeu.
- Interface de participation : accessible pour tous les joueurs, permet de rejoindre une session en saisissant un code.
- Interface session : permet de créer, paramétrer et lancer des sessions du DFC, suivre les joueurs, et analyser les résultats.
- Interface Joueur : permet de participer activement à une session en cours à travers des activités en temps réel.

Les interfaces sont conçues pour être intuitives, accessibles (responsive design) et adaptées à tous niveaux techniques.

### **2.2.1.3 Les interfaces avec le matériel**

Aucune exigence matérielle spécifique. Le système s'exécute sur un navigateur moderne avec connexion Internet.

### **2.2.1.4 Les interfaces avec les logiciels**

Les logiciels et services utilisés ou intégrés comprennent :

- Base de données MySQL pour la persistance des données.
- Web sockets via Laravel Echo pour la communication temps réel entre joueurs et maître du jeu.

### **2.2.1.5 Les interfaces de communication**

Le système PlayLearn met en œuvre différentes interfaces de communication reposant sur des protocoles standards afin d'assurer la sécurité, la fiabilité et la réactivité des échanges de données.

Ces interfaces incluent :

- Une interface client–serveur reposant sur le protocole HTTPS, garantissant la confidentialité et l'intégrité des échanges.
- Une interface avec la base de données MySQL, utilisant le protocole TCP/IP pour les transactions SQL.
- Une interface temps réel, basée sur le protocole WSS (WebSocket Secure), permettant des échanges bidirectionnels entre les clients et le serveur.

### **2.2.1.6 Les contraintes de mémoire**

Cette section décrit les exigences minimales en matière de mémoire, tant pour l'environnement serveur (mémoire primaire et secondaire) que pour les clients utilisateurs, afin d'assurer un fonctionnement optimal de la plateforme.

- Mémoire RAM recommandée serveur : 2 Go minimum, 4 Go idéal.
- L'espace de stockage requis (mémoire secondaire) pour le serveur doit être d'au moins 1 Go, afin d'assurer l'enregistrement des données applicatives, des sessions de jeu, des fichiers générés et des éventuels journaux système.
- Du côté client, la plateforme doit rester légère : le volume de données transféré par session ne doit pas dépasser 2 Mo en moyenne, afin d'optimiser les performances, même sur des connexions à faible débit ou sur des appareils avec peu de mémoire disponible.

## **2.2.2 Fonctions**

Les fonctions principales de la platform sont :

- Utilisateur invité :
  - S'inscrire : créer un compte personnel via formulaire classique ou authentification Google.
  - Se connecter : accéder à la plateforme avec un compte existant.

- Maîtres du jeu :
  - Le lancement de sessions de jeu DFC.
  - La configuration des sessions.
  - Le suivi en direct de la session.
- Joueurs :
  - Rejoindre une session via code d'adhésion.
  - Participer aux activités interactives du DFC.
  - Recevoir des feedbacks personnalisés.

### 2.2.3 Caractéristiques des utilisateurs

| Utilisateur          | Niveau d'instruction                                     | Expérience                                              | Connaissances techniques                                                      |
|----------------------|----------------------------------------------------------|---------------------------------------------------------|-------------------------------------------------------------------------------|
| <b>Joueur</b>        | Étudiant de niveau<br>Licence ou Master (BUT,<br>M1, M2) | Familiarisé avec les<br>outils numériques<br>éducatifs  | Utilisation régulière de<br>plateformes web ou LMS                            |
| <b>Maître du jeu</b> | Enseignant, formateur,<br>encadrant pédagogique          | Expérience en<br>animation de groupe ou<br>enseignement | Bonne maîtrise des outils<br>numériques (navigation,<br>configuration, suivi) |

Tableau 2-2 Caractéristiques des utilisateurs.

### 2.2.4 Contraintes

Nous examinons les contraintes qui doivent être prises en considération lors du développement et de la réalisation du projet

a) Politiques réglementaires

On met en place quelque politique de réglementation pour avoir une bonne efficacité de la platform :

Politique de protection des données : Garantir la confidentialité et la sécurité des données personnelles des utilisateurs (stockage, export, suppression).

Politique d'utilisation : Assurer que les utilisateurs comprennent comment utiliser l'application et comment elle peut les aider.

b) Limites imposées par le matériel

- Aucune contrainte matérielle spécifique côté client.
- Côté serveur, l'architecture doit garantir la prise en charge des connexions simultanées (sessions temps réel avec plusieurs joueurs).
- L'hébergement doit prévoir un serveur compatible Web sockets.

c) Interfaces avec les autres applications

La plateforme doit permettre l'intégration avec des services tels que **Google** pour l'authentification.

d) Exploitation en parallèle

- Plusieurs sessions de jeu peuvent être ouvertes simultanément
- Le système doit gérer les connexions simultanées de plusieurs utilisateurs sans perte de performance.

e) Fonctions de vérification

- Vérification des données saisies (codes de session, réponses aux activités, formulaires).
- Des messages d'erreur explicites doivent guider les utilisateurs en cas de mauvaise saisie.

f) Fonctions de contrôle

Le contrôle sur la Clôturer une session à tout moment.

g) Exigences relatives aux langages évolués

Le logiciel doit être développé à l'aide de technologies modernes, assurant maintenabilité, extensibilité et compatibilité avec l'écosystème web actuel.

- Le backend de l'application doit être développé en PHP 8 ou version supérieure, en s'appuyant sur le framework Laravel, pour bénéficier de sa structure MVC, de ses outils de sécurité intégrés et de son écosystème riche.
- Le frontend doit utiliser Livewire 3 pour la réactivité côté serveur, avec l'appui de composants JavaScript et HTML pour l'interactivité côté client.

h) Protocoles d'échange de signaux

- Le système utilise HTTPS pour les échanges client-serveur.
- La communication temps réel repose sur Web sockets sécurisés

i) Exigences de fiabilité

- Le système doit garantir une haute disponibilité lors d'une session de jeu en cours.
- Les données critiques (paramètres, Cash, choix de décisions) doivent être sauvegardées automatiquement en cas d'interruption.

j) Niveau d'importance de l'application

Le niveau d'importance de la plateforme PlayLearn est jugé élevé dans son contexte d'utilisation, en particulier pour les établissements d'enseignement supérieur et les formations liées à l'industrie 4.0 ou logistique. Plusieurs éléments justifient ce niveau :

- Popularité et adoption.
- Utilité et fonctionnalités.
- Impact sur les utilisateurs.

#### k) Considérations relatives à la sécurité et à la sûreté

- La plateforme doit exiger une authentification des utilisateurs pour accéder aux fonctionnalités avancées, telles que la création d'une session de jeu ou la participation authentifiée à une session.
- Des mesures de protection contre les attaques courantes doivent être mises en place:
  - Protection CSRF / XSS / injection SQL
  - Chiffrement des mots de passe (bcrypt)
  - Gestion des tentatives de connexion abusives
- La plateforme doit garantir la confidentialité et l'intégrité des données utilisateurs, y compris pendant la phase de transmission (via HTTPS) et lors du stockage en base de données.
- La plateforme doit intégrer une authentification à deux facteurs (2FA) afin de renforcer la sécurité des comptes. Cette méthode d'authentification repose sur la vérification de deux éléments distincts :
  - Un identifiant connu (mot de passe)
  - Un code temporaire généré par une application d'authentification mobile (ex. : Google Authenticator)
- Une procédure de désactivation sécurisée de la 2FA et une option de récupération d'accès doivent également être prévues pour les cas de perte d'accès au second facteur.

### 2.2.5 Hypothèses et dépendances

- Nécessité d'une connexion Internet stable.
- Maître de jeu est responsable de l'ouverture et la clôture des sessions.
- Les utilisateurs disposent d'un appareil compatible et d'un navigateur moderne.
- L'hébergement est réalisé sur un serveur supportant PHP 8+, MySQL et Web sockets.

## 2.3 Exigences spécifiques

La section des exigences spécifiques du projet met en évidence les fonctionnalités attendues du logiciel, les interfaces, les performances, les contraintes techniques ainsi que les attributs de qualité clé que l'application doit respecter.

### **2.3.1 Exigences des interfaces externes**

Nous détaillons les spécifications et les exigences relatives aux interactions de la plateforme avec les autres systèmes ou applications externes.

#### **2.3.1.1 Interfaces avec le matériel**

La plateforme est entièrement web-based et ne nécessite aucun matériel spécifique.

- Compatible avec : ordinateurs de bureau, tablettes, smartphones.
- Systèmes d'exploitation supportés : Windows, Mac OS, Linux, Android, iOS.
- Recommandation : navigateur moderne (Chrome, Firefox, Edge).

#### **2.3.1.2 Interfaces avec les logiciels**

La plateforme doit être compatible avec les navigateurs web courants.

#### **2.3.1.3 Interfaces de communication**

- La plateforme doit utiliser le protocole HTTPS pour assurer des échanges sécurisés entre le client et le serveur.
- Les communications en temps réel (ex. : début d'une session, validation des réponses, etc.) doivent être assurées à l'aide de Livewire Events, ou Web Socket si les performances l'exigent.

### **2.3.2 Exigences fonctionnelles**

Les exigences fonctionnelles définissent les fonctionnalités clés que la plateforme doit inclure pour répondre aux besoins des utilisateurs.

#### **2.3.2.1 Classe d'utilisateur 1 : Les invites**

##### **2.3.2.1.1 EF – Inscription**

La plateforme doit permettre à un invité de créer un compte en fournissant un nom complet et un adresse e-mail et un mot de passe.

#### **2.3.2.1.2 EF – Connexion**

L'utilisateur doit pouvoir se connecter à son compte à l'aide de ses identifiants (e-mail et mot de passe) pour accéder au Dashboard.

#### **2.3.2.2 Classe d'utilisateur 1 : Les maîtres de jeu**

Les maîtres de jeu sont responsables de la création, la configuration et le lancement des sessions de jeu.

##### **2.3.2.2.1 EF – Lancer une session**

La plateforme doit permettre à un maître de jeu de configurer et lancer une nouvelle session avec des paramètres associée au jeu DFC.

##### **2.3.2.2.2 EF – Lancer le jeu**

La plateforme doit permettre au maître de jeu de démarrer le jeu une fois que les joueurs ont rejoint la session.

##### **2.3.2.2.3 EF – Visualiser les performances des joueurs**

La plateforme doit permettre au maître de jeu de consulter les performances des joueurs pendant et après la session (cash, classements).

#### **2.3.2.3 Classe d'utilisateur 2 : Les joueurs**

Les joueurs participent aux sessions de jeu et interagissent avec les différentes activités proposées.

##### **2.3.2.3.1 EF – Rejoindre une session**

La plateforme doit permettre à un joueur (invité ou connecté) de rejoindre une session en saisissant un code d'adhésion valide.

##### **2.3.2.3.2 EF – Saisir un pseudonyme**

Après avoir entré un code d'adhésion valide, la plateforme doit afficher une interface demandant au joueur de saisir un pseudonyme avant de rejoindre la session.

##### **2.3.2.3.3 EF – Participer aux activités**

La plateforme doit permettre aux joueurs d'interagir avec les différentes activités du jeu DFC.

### **2.3.3 Exigences de performance**

- Le système doit supporter jusqu'à 50 connexions simultanées par session de jeu.
- Le délai de réponse à une action utilisateur (clic, soumission) ne doit pas dépasser 1 seconde dans des conditions normales.
- Les opérations critiques (démarrage de session, enregistrement des réponses, calcul des impacts) doivent être traitées en moins de 2 secondes.
- Le système doit synchroniser en temps réel les actions des joueurs au sein d'une même session de jeu.
- La plateforme doit garantir un fonctionnement continu sans défaillance critique, même en cas de pics de trafic ou d'utilisation intensive.

### **2.3.4 Contraintes de conception**

- L'interface utilisateur doit être intuitive et facile à utiliser pour les maîtres de jeu et les joueurs.
- Aucune formation préalable ne doit être nécessaire pour naviguer sur la plateforme.
- Le design doit guider l'utilisateur naturellement vers les actions attendues.

### **2.3.5 Attributs**

- Disponibilité : La plateforme doit être disponible 24h/24 sauf période de maintenance planifiée.
- Fiabilité : La plateforme doit être fiable et ne doit pas planter ou se bloquer.
- Maintenabilité :
  - L'architecture du code suit le modèle MVC avec des composants Livewire modulaires.
  - Le code est documenté pour faciliter sa reprise.
- Sécurité :
  - Le système doit empêcher les accès non autorisés
  - La plateforme doit garantir la sécurité des données des utilisateurs.

- Le système doit intégrer une authentification à deux facteurs (2FA), basée sur une application mobile d'authentification, afin de renforcer la sécurité des comptes utilisateurs.
- Transférabilité : La plateforme doit pouvoir être déployé facilement sur un autre serveur.

### **2.3.6 Autres exigences**

#### **2.3.6.1 Exigences fonctionnelles liées à la sécurité**

- La plateforme doit permettre à un utilisateur de sécuriser son compte en activant une authentification à deux facteurs (2FA) via une application mobile (ex. : Google Authenticator, Microsoft Authenticator).
- L'utilisateur doit pouvoir désactiver la 2FA depuis son tableau de bord, après une vérification d'identité.
- Le système doit fournir un mécanisme de récupération sécurisé en cas de perte du second facteur via un code de secours.

## **2.4 Conclusion**

Dans ce chapitre, nous avons défini avec précision les exigences logicielles afin de fournir une description détaillée des fonctionnalités et des performances nécessaires pour répondre aux besoins des utilisateurs. Ces exigences sont essentielles dans le processus de développement logiciel, car elle garantit que l'application réponde pleinement aux attentes des utilisateurs tout en respectant les normes de qualité requises. Ainsi, en accordant une attention particulière à la spécification des exigences, nous nous assurons de la réussite de notre projet logiciel et de la satisfaction des utilisateurs. Dans le chapitre qui suit on va entamer l'analyse des besoins et de conception, dans le but de visualiser l'architecture de notre plateforme.

# Chapitre 3

## Analyse des besoins et conception

### 3.1 Introduction

Après avoir défini les différents besoins fonctionnels et non fonctionnels de notre application, nous passons à la phase d'analyse des besoins et conception.

Dans cette phase, nous utilisons le langage UML pour modéliser le système, conformément aux activités définies dans le processus UP. Nous définissons les différents acteurs de système, ainsi que les diagrammes de contexte, les diagrammes de cas d'utilisation, diagramme de séquence et le diagramme de classes pour passer en fin à réaliser le modèle relationnel associé au diagramme de classe de notre plateforme.

### 3.2 Présentation de processus unifié

Un processus unifié est un processus de développement logiciel construit sur UML, il est itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques [11].

La gestion d'un tel processus est inception, élaboration, construction et transition. Ses activités de développement sont définies par 5 disciplines fondamentales qui décrivent :

- Capture des besoins,
- l'analyse et la conception,
- l'implémentation,
- le test,
- le déploiement.

La figure suivante décrit les différentes activités de processus :

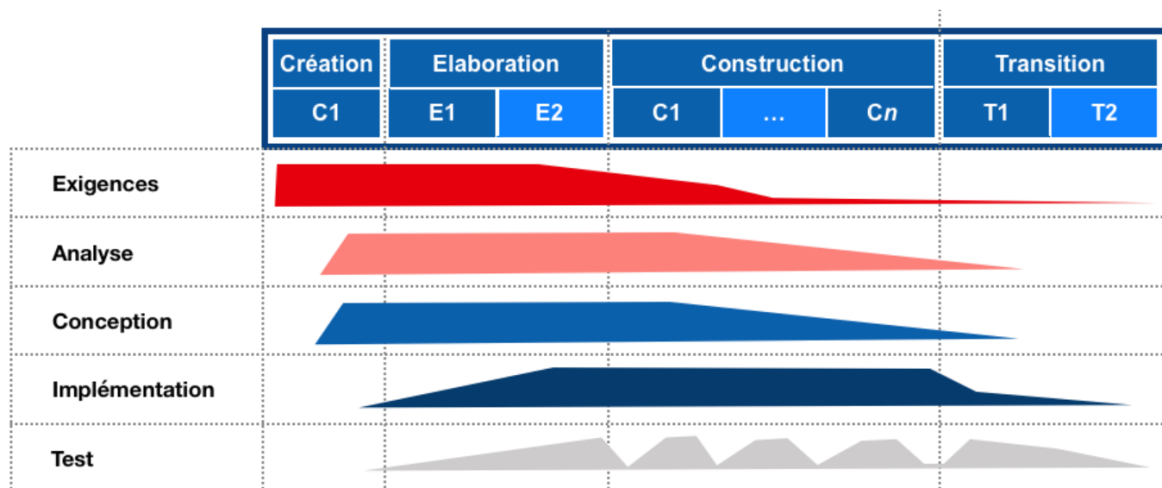


Figure 3.1 Description de processus unifié.

## 3.3 Présentation de langage UML

### 3.3.1 Définition d'UML

UML (Unified Modeling Language) se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue [12].

### 3.3.2 Les diagramme UML

UML 2 s'articule autour de quatre types de diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont répartis en deux grands groupes [12].

### 3.3.2.1 Les diagrammes structurels

Ces diagrammes, au nombre de six, ont vocation à représenter l'aspect statique d'un système (classes, objets, composants).

### 3.3.2.2 Les diagrammes de comportement

Ces diagrammes représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs.

L'ensemble des treize types de diagrammes UML peut ainsi être résumé sur la figure suivante :

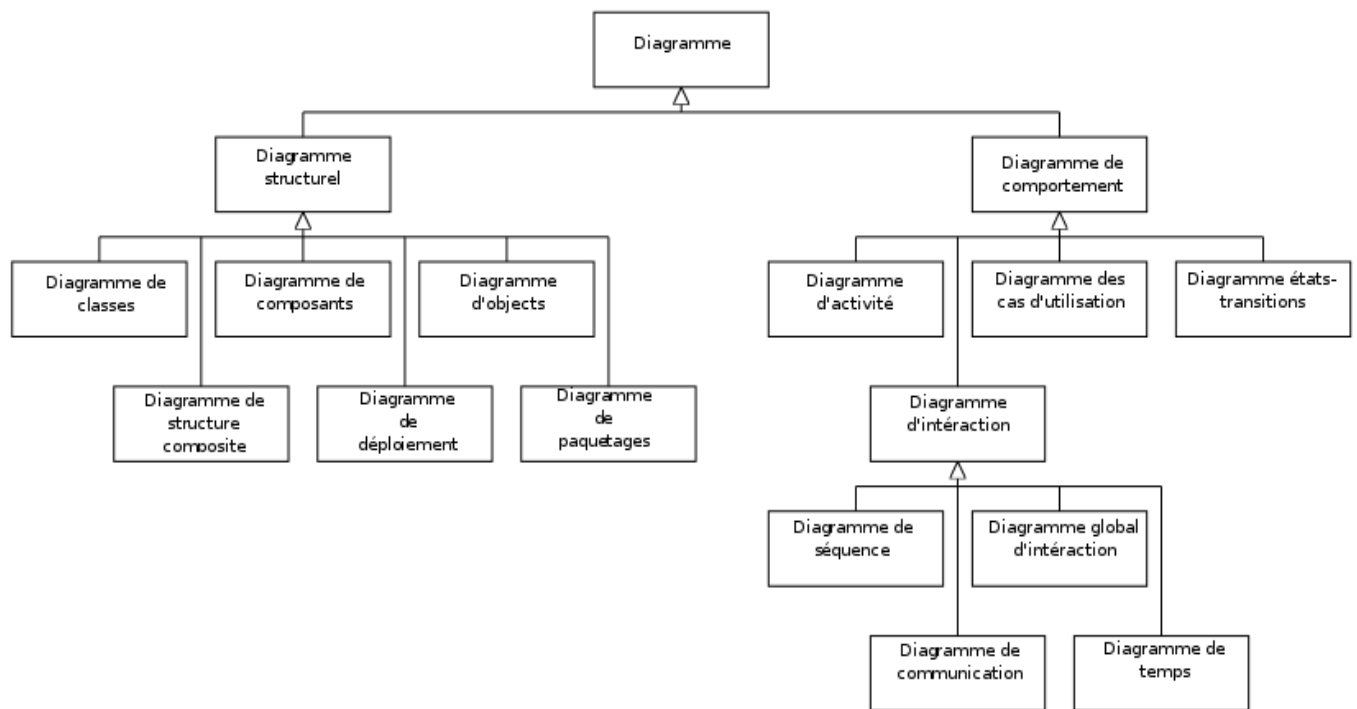


Figure 3.2 – Présentation des diagrammes UML.

## 3.4 Analyse des besoins

Après l'étude des besoins fonctionnels cités précédemment, on peut passer à l'étape de formalisation de ces besoins à l'aide du diagramme de cas d'utilisation.

### 3.4.1 Acteurs du système

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié, autrement dit un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données [13].

Dans le cadre de notre plateforme playlearn, nous avons distingué deux acteurs principaux :

- Maître du jeu : responsable du lancement et de la gestion des sessions de jeu basées sur le serious game DGC.
- Joueur : participant qui rejoint une session de jeu *DGC*.

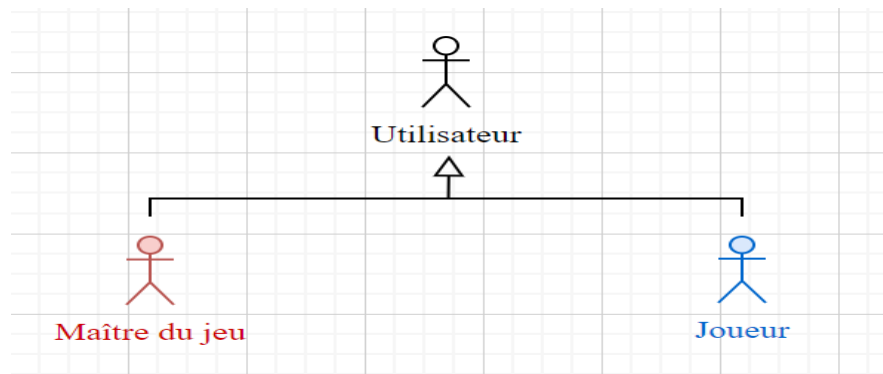


Figure 3.3 – Héritage des acteurs

### 3.4.2 Diagramme de contexte statique

Le diagramme de contexte statique constitue une représentation globale du système d'information étudié, en en délimitant les frontières fonctionnelles. Il permet d'identifier ce qui relève de la responsabilité du système et ce qui appartient à son environnement externe. Ce diagramme est un outil essentiel pour comprendre les interactions entre le système et les entités extérieures avec lesquelles il communique.

Ses composants sont :

- Les acteurs externes, qui sont des entités extérieures (utilisateurs, organisations, services ou systèmes tiers) interagissant avec le système. Ils peuvent envoyer ou recevoir des informations.
- Un processus unique symbolisant le Système Information étudié.
- Les flux d'échange, qui représentent les interactions entre les acteurs externes et le système étudié.

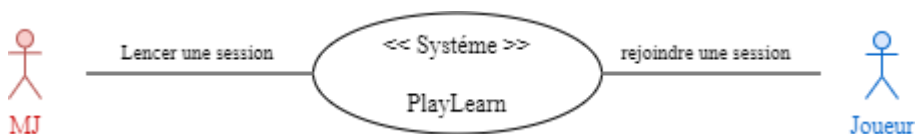


Figure 3.4 – Diagramme de contexte statique

### 3.4.3 Diagramme des cas d'utilisation

Le diagramme de cas d'utilisation décrit les grandes fonctions d'un système du point de vue des acteurs, mais n'expose pas de façon détaillée le dialogue entre les acteurs et les cas d'utilisation. Bien que de nombreux diagrammes d'UML permettent de décrire un cas, il est recommandé de rédiger une description textuelle, car c'est une forme souple qui convient dans bien des situations [13].

### 3.4.3.1 Identification des cas d'utilisation

Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service [13].

Le tableau suivant décrit les différents cas d'utilisation associées pour chaque acteur :

| Acteur        | Cas d'utilisation         | Désignation                                                                                                      |
|---------------|---------------------------|------------------------------------------------------------------------------------------------------------------|
| Utilisateur   | S'inscrire                | Utilisateur crée un compte sur la plateforme afin d'interagir avec le système.                                   |
|               | S'authentifier            | Utilisateur saisit son login et son mot de passe pour accéder à son interface.                                   |
| Maître de Jeu | Démarrer une session      | Le maître de jeu lance une nouvelle session du jeu DGC et génère un code d'adhésion à partager avec les joueurs. |
|               | Lancer le jeu             | Le maître de jeu démarre le jeu après le lancement de la session.                                                |
|               | Observer les performances | Le maître de jeu suit en temps réel ou après la session les performances des joueurs.                            |
|               | Clôturer une session      | Le maître de jeu peut clôturer la session à tout moment.                                                         |



- Diagramme de cas d'utilisation – Gestion d'une session de jeu :

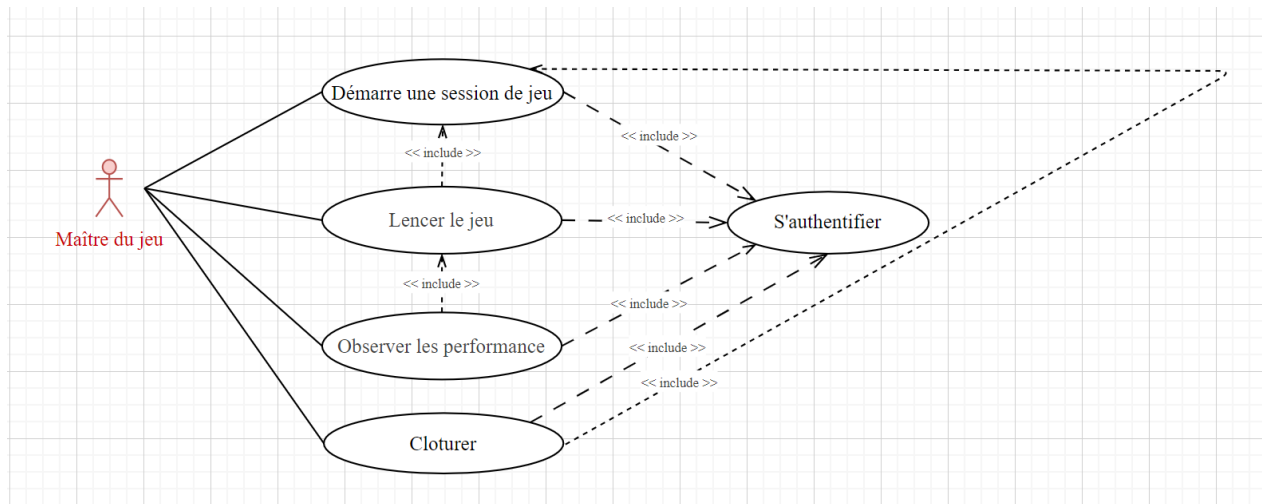


Figure 3.6 – Diagramme de cas d'utilisation – Gestion d'une session de jeu

- Diagramme de cas d'utilisation – Joueur

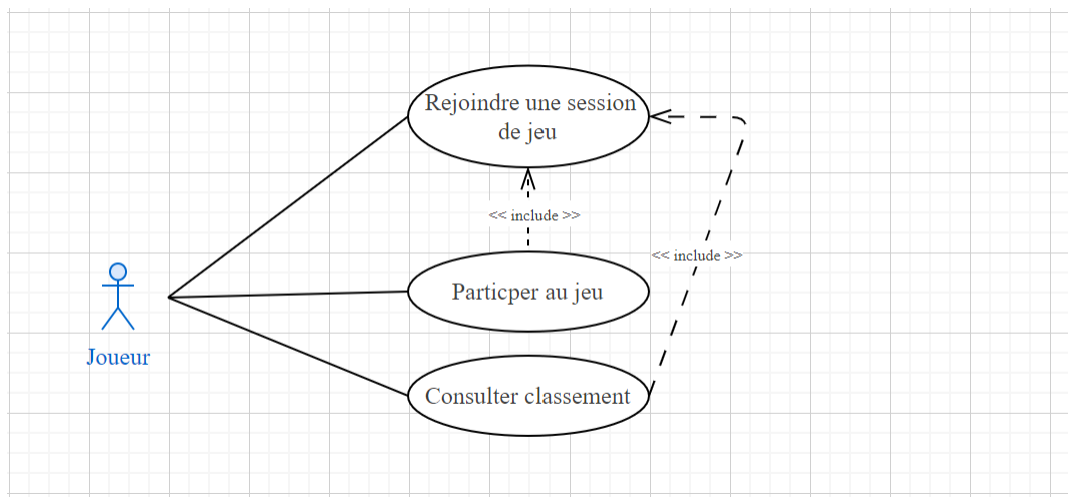


Figure 3.7 – Diagramme de cas d'utilisation – Joueur

### 3.4.4 Diagramme de séquence de système

Les diagrammes de séquences mettent en valeur les échanges de messages (déclenchant des événements) entre acteurs et objets (ou entre objets et objets) de manière chronologique, l'évolution du temps se lisant de haut en bas [14].

- Diagramme de séquence du cas d'utilisation – s'inscrire

Lorsqu'un client souhaite créer un compte, il envoie une demande d'inscription au système. Ce dernier affiche les formulaires d'inscription que l'utilisateur doit remplir. Une fois le formulaire soumis, le système procède à une vérification. Si le formulaire est incomplet ou comporte une erreur, un message d'erreur est retourné. Sinon, si toutes les informations sont correctes, le système affiche l'interface Dashboard pour les utilisateurs authentifiés.

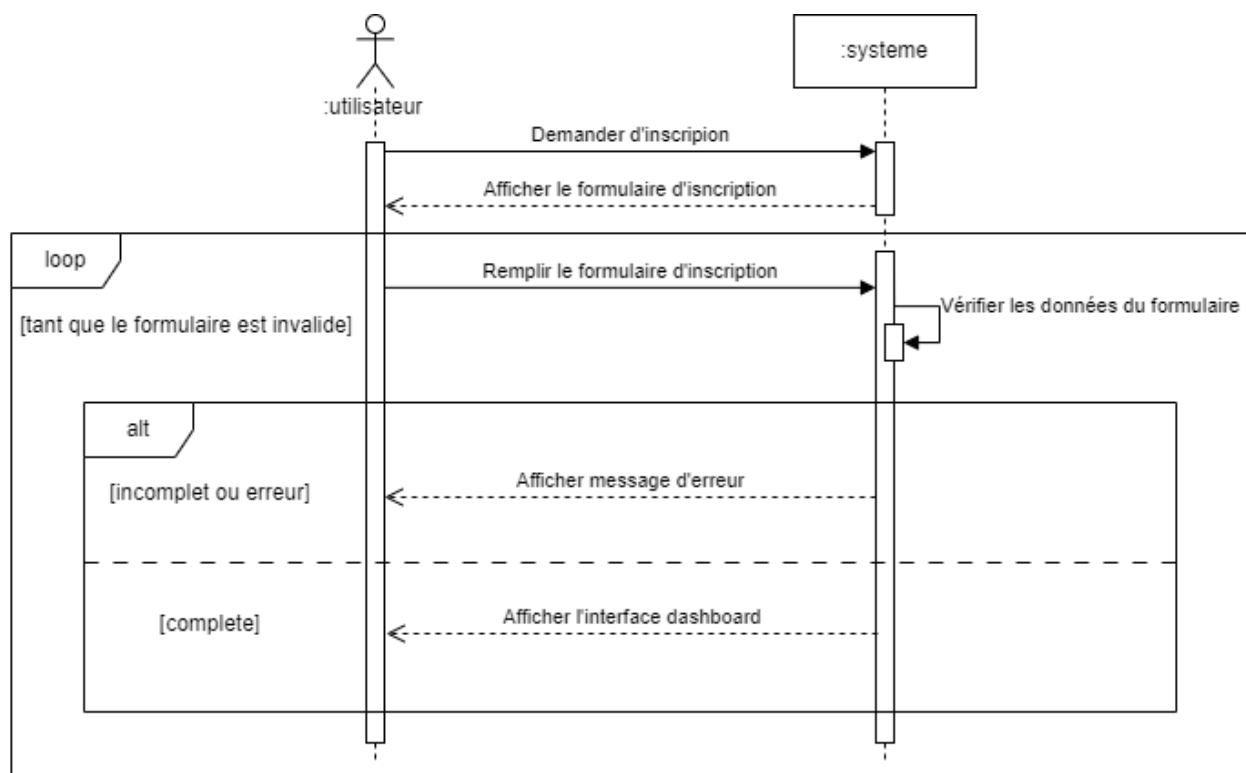


Figure 3.8 – Diagramme de séquence du cas d'utilisation – s'inscrire

- Diagramme de séquence du cas d'utilisation – s'authentifier

L'authentification consiste à assurer la confidentialité des données, elle se base sur la vérification des informations associées à un utilisateur (un email et un mot de passe). Ces informations sont préétablies dans une base de données. Lors d'une authentification deux cas se présentent : les informations introduites par l'utilisateur sont incomplètes ou pas valides, dans ce cas un message d'erreur s'affiche, ou les informations saisies sont complètes et le système procède à leur vérification. Ceci explique l'utilisation de l'opérateur « alt ». Le même opérateur illustre les deux réactions du système, après la vérification des informations saisies par l'utilisateur, soit par l'affichage d'un message d'erreur, ou de l'interface Dashboard pour les utilisateurs authentifier.

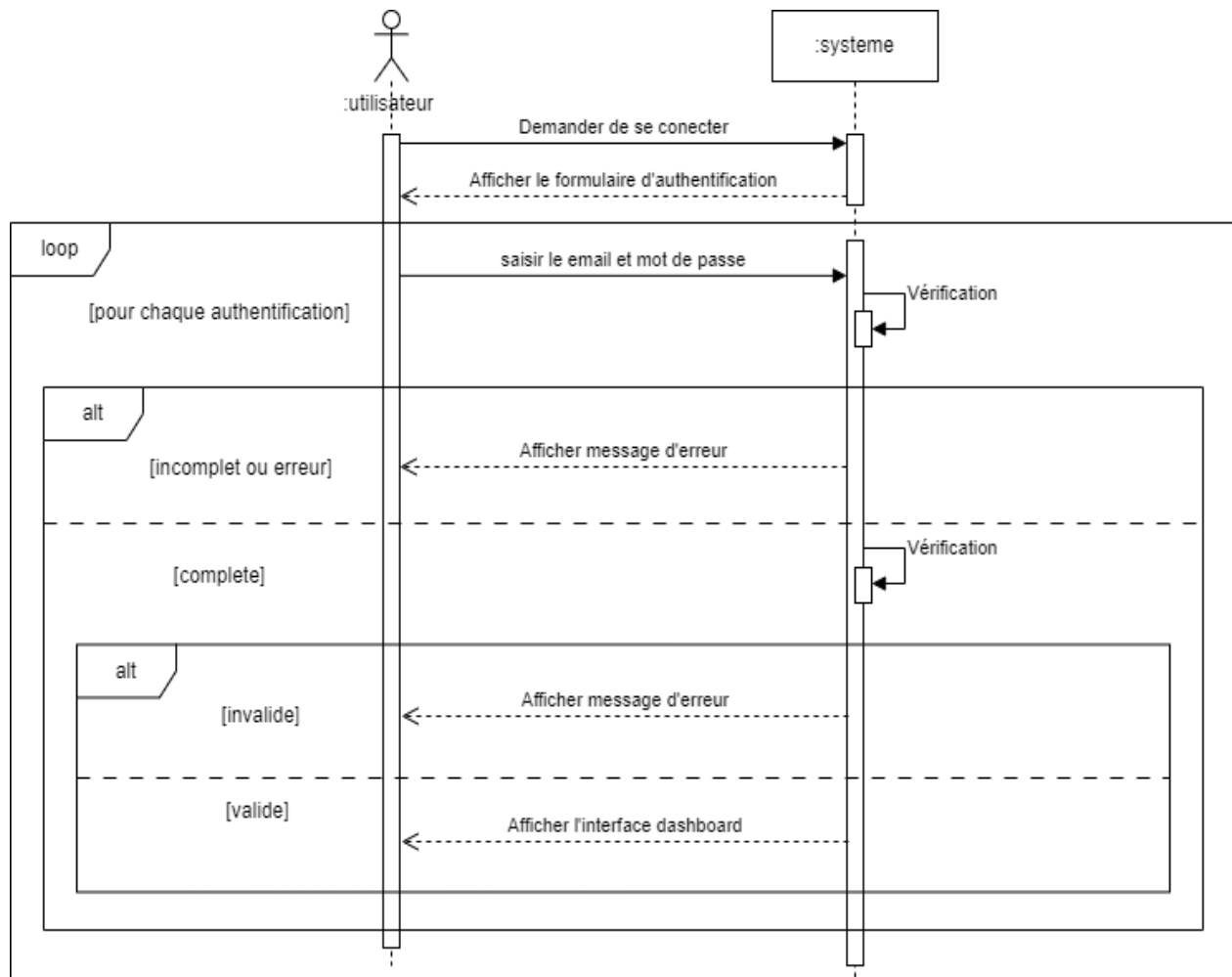


Figure 3.9 – Diagramme de séquence du cas d'utilisation – s'authentifier

- Diagramme de séquence du cas d'utilisation – Démarrer une session

Lorsque le maître de jeu envoie une demande de lancement d'une session de jeu, le système lui répond en affichant l'interface de configuration de la session. Une fois la session configurée, le maître de jeu peut la démarrer, ce qui conduit le système à afficher l'interface d'attente des joueurs.

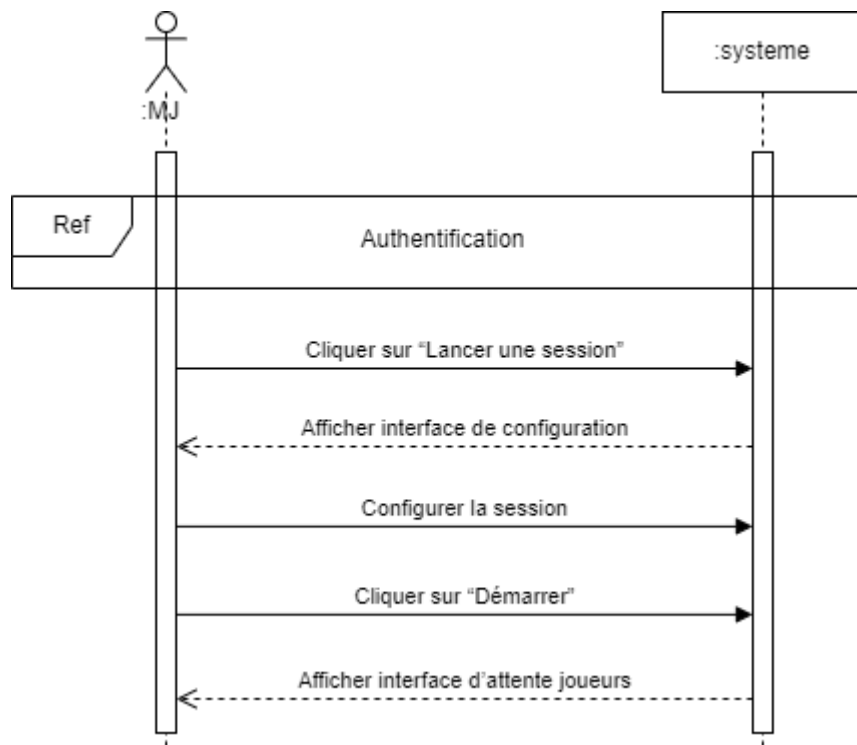


Figure 3.10 – Diagramme de séquence du cas d'utilisation – Démarrer une session

- Diagramme de séquence du cas d'utilisation – clôturer une session

Lorsque le maître de jeu envoie une demande de clôture d'une session de jeu, le système traite cette action en mettant fin à la session en cours. Une fois la session clôturée, le système redirige le maître de jeu vers l'interface dashboard pour les utilisateurs authentifiés, tandis que les joueurs sont redirigés vers l'interface de fin de session.

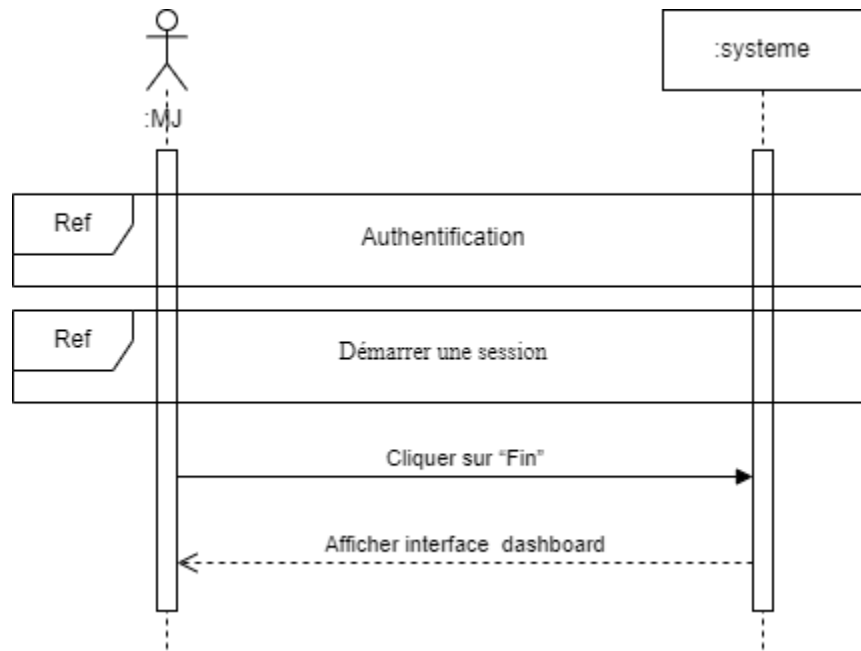


Figure 3.11 – Diagramme de séquence du cas d'utilisation – clôturer une session

- Diagramme de séquence du cas d'utilisation – Rejoindre une session de jeu

Lorsque le joueur saisit le code d'adhésion fourni par le maître de jeu, le système vérifie la validité de ce code ainsi que l'état de la session. Si le code est valide et que la session est ouverte, le système affiche l'interface de pré-jeu afin que le joueur puisse saisir son pseudonyme. Après validation du pseudonyme, le joueur est ajouté à la session et redirigé vers l'interface d'attente.

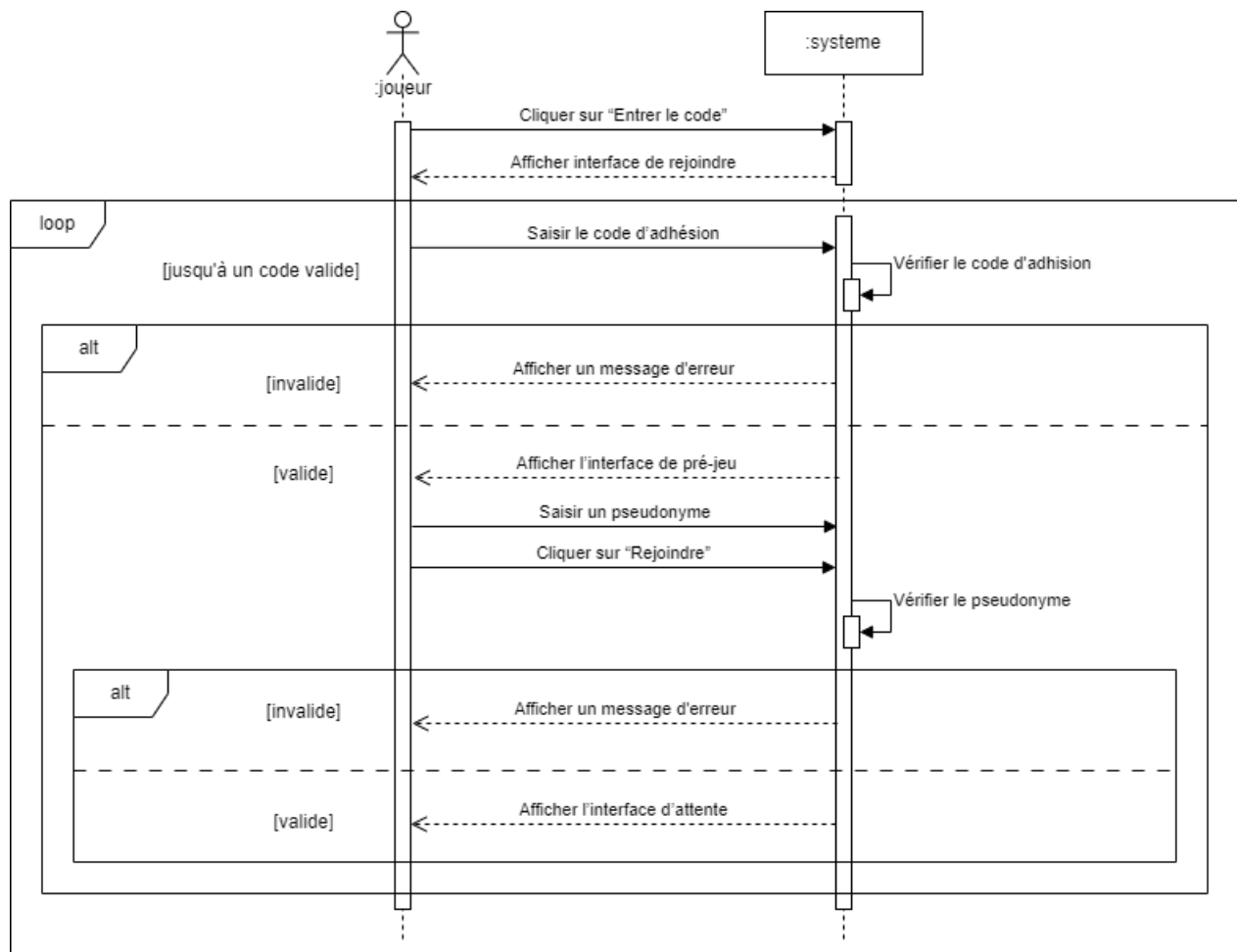


Figure 3.12 – Diagramme de séquence du cas d'utilisation – Rejoindre une session de jeu

- Diagramme de séquence du cas d'utilisation – Lancer le jeu

Lorsque le maître de jeu envoie une demande de lancement du jeu, le système vérifie si au moins un joueur a rejoint et si le nombre maximal de joueurs défini dans la configuration a été atteint pour démarrer la session est atteint. Si le nombre est suffisant, la session est lancée et le système redirige le maître de jeu vers l'interface de contrôle de jeu et les joueurs vers l'interface de jeu.

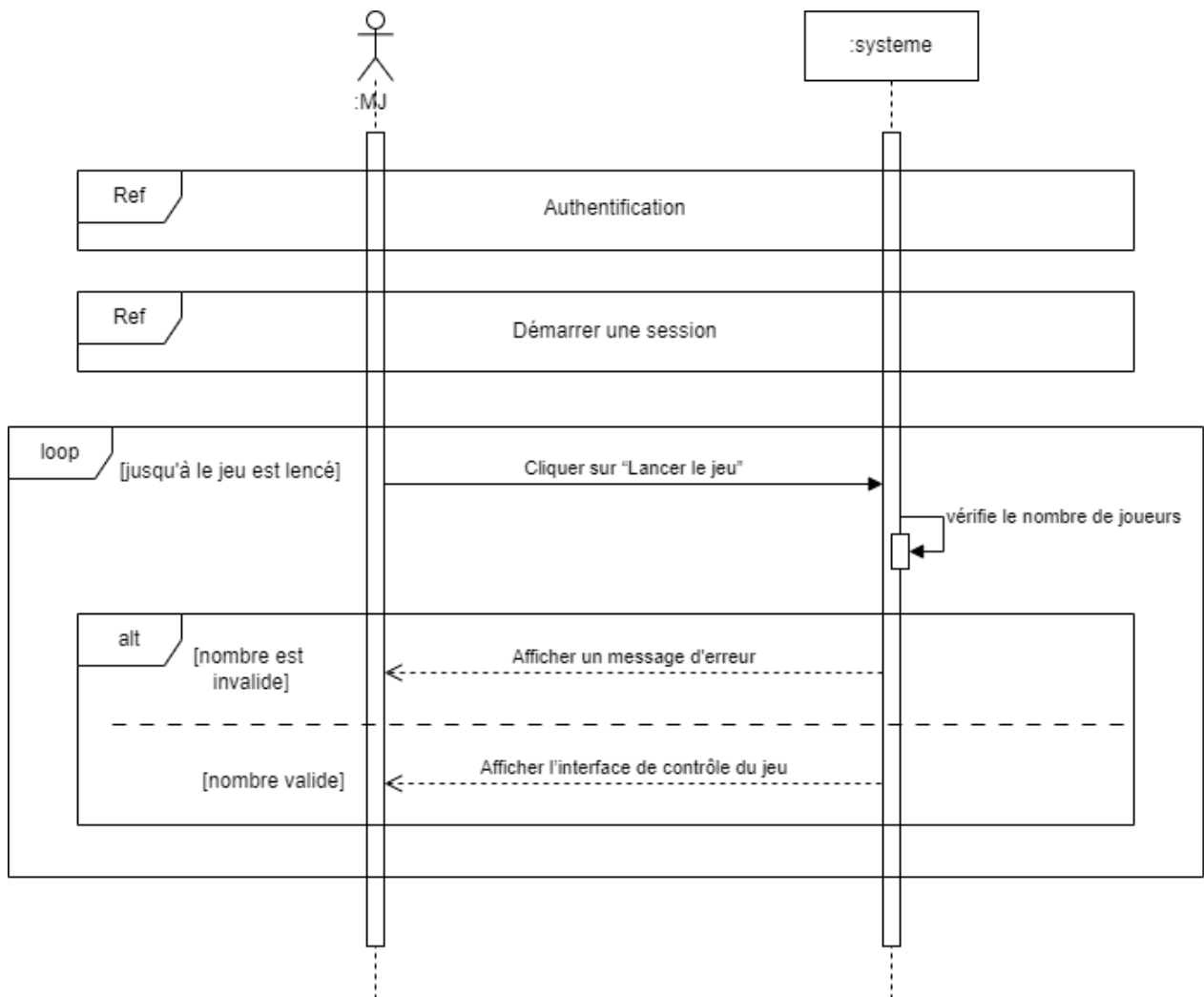


Figure 3.13 – Diagramme de séquence du cas d'utilisation – Lancer le jeu

- Diagramme de séquence du cas d'utilisation – Participer au jeu

Lorsque la session lancée par le maître de jeu Cette condition est représentée par un bloc opt, le système permet au joueur de commencer les activités du jeu. Le joueur participe d'abord à un quiz, puis enchaîne trois cycles composés chacun d'une activité de prise de décision suivie d'une simulation. À l'issue des trois cycles, le système affiche l'interface de classement.

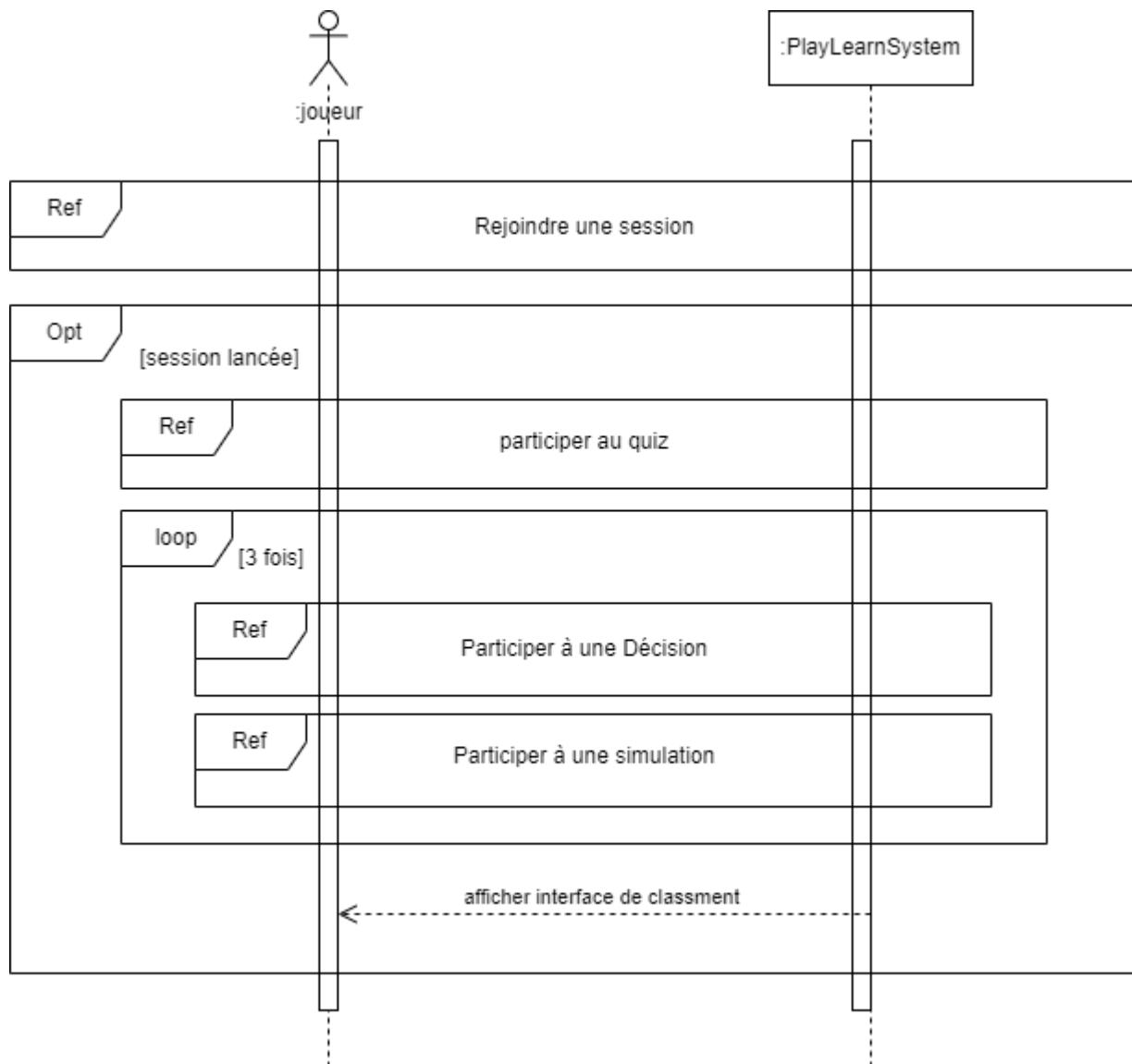


Figure 3.14 – Diagramme de séquence du cas d'utilisation – Lancer le jeu

### 3.5 Analyse de conception

Dans cette partie, nous présentons le diagramme de classe de conception ainsi que le modèle relationnel.

### 3.5.1 Diagramme de classe de conception

Le diagramme de classe constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est centrée sur les concepts de classe et d'association. Chaque classe se décrit par les données et les traitements dont elle est responsable pour elle-même et vis-à-vis des autres classes. Les traitements sont matérialisés par des opérations. Le détail des traitements n'est pas représenté directement dans le diagramme de classe, seul l'algorithme général et le pseudo-code correspondant peuvent être associés à la modélisation [15].

La description du diagramme de classe est fondée sur :

- Le concept d'objet.
- Le concept de classe comprenant les attributs et les opérations.
- Les différents types d'association entre classes.

#### 3.5.1.1 Présentation des classes

| Classe  | Désignation                                                                                                                                                                                                |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| User    | Contient les informations qui concernant un utilisateur, ce dernier peut être un maitre de jeu ou un joueur selon le contexte de la session.                                                               |
| Session | Désigne les informations liées à une session de jeu. Une session correspond à une partie en cours de <b>jeu</b> , dirigée par un Maître de jeu, à laquelle les joueurs participent via un code d'adhésion. |

|            |                                                                                                                                                                                                           |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Jeu        | Représente un scénario de jeu composé de plusieurs activités, et un ensemble de paramètres à gérer par les joueurs durant la partie.                                                                      |
| Paramètre  | Définit un indicateur ou une variable que le joueur doit gérer au cours du jeu. Exemples : cash, satisfaction client, stock, etc. Les paramètres évoluent en fonction des décisions prises par le joueur. |
| Quiz       | Contient une activité de type questionnaire à choix multiples ou unique, cette dernière impacte un ou plusieurs paramètres selon les réponses du joueur.                                                  |
| Decision   | Contient une activité dans laquelle le joueur doit faire un choix stratégique, chaque choix ayant un impact sur un ou plusieurs paramètres du jeu.                                                        |
| Simulation | Contient une activité qui simule un processus dynamique, cette dernière permet d'évaluer les effets de différents facteurs (pannes, qualité, etc.) sur les paramètres de joueur.                          |

Tableau 3-2 Présentation des classes de l'application

### 3.5.1.2 Diagramme de classe de la plateforme



### 3.5.2 Passage au modèle relationnel

Le concepteur d'une base de données relationnelle doit élaborer ce qu'il est convenu d'appeler le schéma relationnel de la base de données. Cette activité consiste à définir toutes les relations normalisées de la base de données et les domaines de leurs attributs. Théoriquement cela consisterait à décrire par intention chaque relation et définir les domaines de chaque attribut de la relation. Les relations du schéma doivent être toutes posséder les propriétés suivantes [16] :

- 1) Une relation a un nom distinct de toutes les autres relations du même schéma.
- 2) Chaque attribut d'une relation ne peut recevoir qu'une seule valeur atomique (type de données simple).
- 3) Chaque attribut a un nom unique.
- 4) Toutes les valeurs d'un attribut font parties du même domaine, même type de données et même contraintes d'intégrité.
- 5) Chaque tuple de la relation est distinct, pas de tuple en double.
- 6) L'ordre des tuples n'a pas d'importance.
- 7) L'ordre des attributs n'a pas d'importance.

#### 3.5.2.1 Les règles du passage

Règle1 : Transformation des classes

Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle de l'identifiant. Dans le cas où aucun attribut ne convient en tant qu'identifiant, il faut en ajouter un de telle sorte que la relation dispose d'une clé primaire.

Règle2 : Association un-à-plusieurs

Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.

#### Règle3 : Associations plusieurs-à-plusieurs

L'association devient une relation dont la clé primaire est composée par la concaténation des identifiants des classes connectées à l'association. Les attributs de l'association doivent être ajoutés à la nouvelle relation.

#### Règle4 : Association un-à-un

Il faut ajouter un attribut clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de la classe connectée à l'association. Si les deux cardinalités (multiplicités) minimales sont à zéro, le choix est donné entre les deux relations dérivées de la règle R1. Si les deux cardinalités minimales sont à un, il est sans doute préférable de fusionner les deux entités (classes) en une seule.

#### Règle5: Transformation de l'héritage

Trois décompositions sont possibles pour traduire une association d'héritage en fonction des contraintes existantes :

- Décomposition par distinction : il faut transformer chaque sous-classe en une relation. La clé primaire de la sur-classe, migre dans la (les) relation(s) issue(s) de la (des) sous-classe(s) et devient à la fois clé primaire et clé étrangère.

- Décomposition descendante(push-down) : s'il existe une contrainte de totalité ou de partition sur l'association d'héritage, il est possible de ne pas traduire la relation issue de la sur-classe. Il faut alors faire migrer tous ses attributs dans la (les) relation(s) issue(s) de la (des) sous classe(s).

- Décomposition ascendante (push up) : il faut supprimer la (les) relation(s) issue(s) de la (des) sous classe(s) et faire migrer les attributs dans la relation issue de la sur-classe. L'héritage de la décomposition par distinction et décomposition ascendante sont ceux qui correspondent à notre diagramme de classe.

## Règle6 : Agrégation

L'agrégation partagée de UML se traduit au niveau logique comme une simple association, il n'est pas de même pour la composition

### **3.5.2.2 Le modèle relationnel associé au diagramme de classe**

Après l'application de règles de passage sur le diagramme de classe, on a obtenu le modèle relationnel suivant :

- User (id-user, Name, Email, Password, is-guest, two\_factor\_secret, two\_factor\_recovery\_codes);
- Jeu (id-jeu, titre, description);
- Paramètre (id-param, nom, initialValue, unite, id-jeu#);
- Session (id-session, code\_adhesion, status, mode, max\_players, id-jeu#, id-user#) ;
- sessions\_joueurs (id-join, pseudonym, id-session#, id-user#) ;
- Paramètre\_session\_joueurs (id, valeur, id-param#, id-join#) ;
- Simulation (id-sim, titreSim, descriptionSim, durée, id-jeu#) ;
- Decision (id-decision, titre, description, durée, id-jeu#) ;
- ChoixDecision (id-choix, choice\_text, id-decision#);
- LevelChoice (id-choix, level, description, cout, id-choix#);
- EffetDecision (id-effetDecision, valeur\_impact, id-param#, id-choix#);
- Quiz (id-quiz, titre, description, cach, id-jeu#) ;
- Question (id-question, texte, type\_question, temp, id-quiz#) ;
- ChoixReponse (id-choixReponse, texte, is\_correct, variation, id-question#) ;
- Soumission (id-soumission, is\_correct, id-join#, id-choixReponse#);

- EffetQuiz (id-effetQuiz, valeur\_impact, id-param#, id-soumission #);

## 3.6 Conclusion

Dans ce chapitre nous avons présenté l'étape de conception et d'analyse des besoins, que nous avons décrit d'une façon détaillée la modélisation en utilisant le langage de modélisation « UML », ainsi que le processus unifié qui nous a aidé de faire une conception détaillée avec les différents diagrammes.

# Chapitre 4

## Réalisation

### 4.1 Introduction

Dans ce chapitre, Nous commençons par présenter l’environnement, les outils et les langages utilisés pour implémenter et réaliser notre plateforme. Ensuite, nous introduisons le diagramme de déploiement pour représenter l’architecture de l’application dans son environnement d’exécution. Enfin, nous allons illustrer les fonctionnalités de notre système à travers des captures d’écrans.

### 4.2 Environnement de travail

Un environnement de travail est l’ensemble d’outils et technologies nécessaires à l’élaboration d’un système donné. Dans notre travail, nous avons utilisé les outils et les langages suivants :

#### 4.2.1 HTML

C’est un langage qui permet la construction et la structuration des sites web. Ce n’est pas un langage de programmation car il ne permet pas de créer des fonctionnalités dynamiques [17]. Nous l’avons utilisé pour définir la structure de notre application web.

#### 4.2.2 CSS

CSS est utilisé pour décrire les styles de pages HTML, par exemple, les couleurs, la taille de texte, etc [18]. Nous l’avons utilisé pour ajouter du style à nos page HTML.

#### 4.2.3 Tailwind CSS

C’est un framework CSS utilitaire qui permet de concevoir rapidement des interfaces web en appliquant directement des classes prédéfinies dans le code HTML. Contrairement aux feuilles de style classiques [19]. Nous l’avons utilisé pour styler notre application de manière réactive et moderne sans écrire beaucoup de CSS personnalisé.

#### **4.2.4 Java Script**

C'est un langage de programmation orienté objet principalement utilisé côté client pour rendre les pages web interactives. Il permet gérer les événements utilisateur et communiquer avec le serveur sans recharger la page [20]. Nous l'avons utilisé pour compléter certaines interactions dynamiques dans notre application, en complément de Livewire et Alpine.js.

#### **4.2.5 Alpine JS**

C'est une librairie JavaScript légère qui permet d'ajouter des interactions dynamiques directement dans le HTML, de manière simple et réactive. Idéale pour enrichir les composants Livewire sans complexité excessive [21]. Nous l'avons utilisée pour gérer de petites interactions côté client, comme l'affichage conditionnel ou le changement d'état d'un élément dans l'interface.

#### **4.2.6 PHP**

C'est un langage de script côté serveur principalement utilisé pour le développement web. Il permet de créer des site web dynamiquement, interagir avec des bases de données et gérer la logique côté serveur [22]. Dans notre projet, nous l'avons utilisé comme langage principal pour le développement backend.

#### **4.2.7 Laravel**

C'est un framework PHP open source basé sur le modèle MVC (Modèle-Vue-Contrôleur), qui facilite la création d'applications web robustes et bien structurées. Il fournit une syntaxe claire et de nombreux outils intégrés, comme la gestion des routes, des migrations, de l'authentification et des contrôleurs [23]. Nous l'avons utilisé pour organiser notre application et accélérer son développement tout en assurant sa maintenabilité.

#### **4.2.8 Livewire**

C'est un framework full-stack pour Laravel qui permet de créer des interfaces dynamiques sans avoir besoin d'écrire du JavaScript. Il fonctionne en combinant PHP et JavaScript de manière transparente : les composants sont écrits en PHP et les interactions côté client sont automatiquement gérées via AJAX [24]. Nous l'avons utilisé pour développer des composants interactifs dans notre application tout en restant dans l'environnement Laravel.

### 4.2.9 Visual Studio Code

C'est un éditeur de code source multiplateforme, disponible sur Windows, macOS et Linux, qui prend en charge de nombreux langages de programmation, notamment PHP, JavaScript, etc. Parmi ses fonctionnalités, nous en citons : la prise en charge du débogage, la complétion intelligente du code, la refactorisation du code et Git intégré [25]. Nous l'avons utilisé pour écrire le code de notre application (HTML, CSS, JavaScript, PHP), et ses nombreuses extensions nous ont grandement facilité le développement.

### 4.2.10 Draw.io

Draw.io est un logiciel polyvalent de conception des diagrammes avec des caractéristiques qui facilitent la création d'une illustration scientifique, des diagrammes UML, etc. [26]. Il est destiné à tous les types d'utilisateurs. Nous l'avons utilisé pour concevoir différents diagrammes lors de la phase de modélisation de notre projet.

## 4.3 Diagramme de déploiement

Un diagramme de déploiement décrit la disposition physique des ressources matérielles qui composent le système et montre la répartition des composants sur ces matériels. Chaque ressource étant matérialisée par un noeud, le diagramme de déploiement précise comment les composants sont répartis sur les noeuds et quelles sont les connexions entre les composants ou les nœuds. Les diagrammes de déploiement existent sous deux formes : spécification et instance [27].

Chaque ressource est matérialisée par un nœud représenté par un cube comportant un nom. Un nœud est un classeur et peut posséder des attributs (quantité de mémoire, vitesse du processeur...etc).

- Diagramme de déploiement de notre plateforme

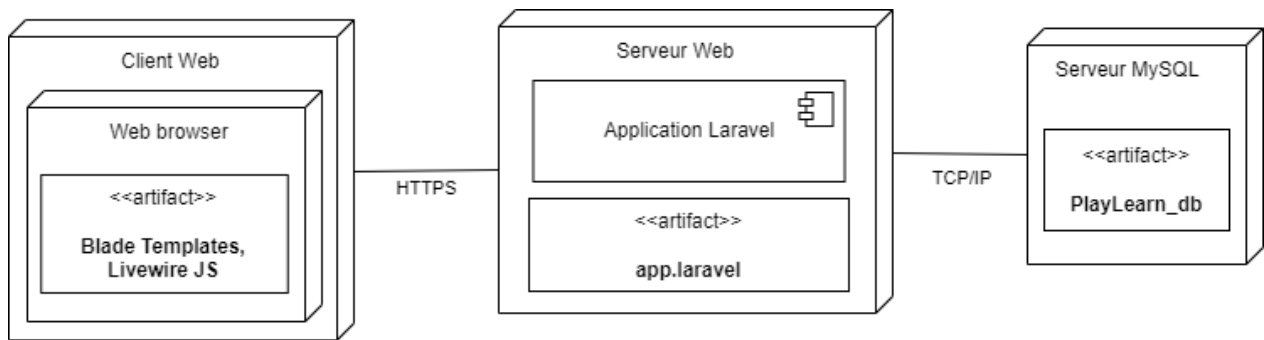


Figure 4.1– Diagramme de déploiement de La plateforme.

## 4.4 Présentation de l'application

Dans cette partie, nous allons illustrer le fonctionnement de notre application à travers des captures d'écrans des interfaces par acteur.

### 4.4.1 Page d'accueil

La page d'accueil de PlayLearn est la première interface visible par l'utilisateur. L'objectif est d'inviter les utilisateurs à découvrir la plateforme de manière engageante.

En haut à droite, trois boutons permettent de :

- S'inscrire à la plateforme,
- Se connecter pour accéder à son espace personnel,
- Entrer un code pour rejoindre une session de jeu existante en tant que joueur.

Deux boutons incitent à l'action :

- Get started : pour commencer rapidement.
- Learn more : pour en savoir plus sur le fonctionnement du jeu.

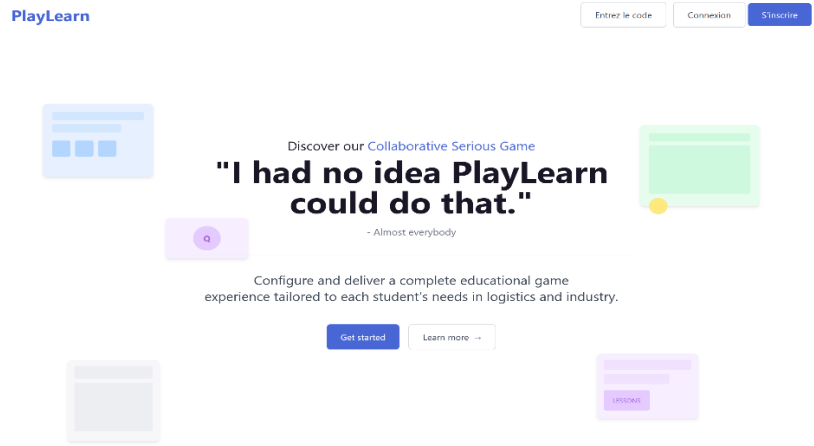


Figure 4.2 Page d'accueil

## 4.4.2 Utilisateur

### 4.4.2.1 S'inscrire

Afin d'accéder aux fonctionnalités avancées de l'application PlayLearn, chaque utilisateur doit disposer d'un compte. Lors de l'inscription, il est invité à renseigner les informations suivantes : nom, adresse e-mail et mot de passe. Une fois les champs complétés, il peut cliquer sur le bouton "Register" pour finaliser la création de son compte.

Figure 4.3 Interface d'inscription

#### 4.4.2.2 Se connecter

Lorsqu'un utilisateur souhaite accéder à son compte sur la plateforme PlayLearn, il doit renseigner les informations d'authentification requises, à savoir l'adresse e-mail et Mot de passe. Une fois que les informations de connexion ont été saisies, il peut cliquer sur le bouton "Log in" afin de pouvoir accéder à son interface. En cas d'oubli du mot de passe, il a la possibilité de cliquer sur le lien "Mot de passe oublié" pour réinitialiser celui-ci. La figure 4.4 montre la page de connexion.



Figure 4.4 interface de connexion

Si la 2FA est activée par l'utilisateur, une étape supplémentaire est requise lors de la connexion. Après avoir saisi son adresse e-mail et son mot de passe, il doit fournir un code temporaire à 6 chiffres (TOTP) généré par une application d'authentification mobile telle que Google Authenticator, sinon doit fournir code de récupération d'urgence préalablement généré et sauvegardé lors de l'activation de la 2FA. La figure 4.5 montre la page de 2FA.



Figure 4.5 interface 2FA

### 4.4.3 Maître de jeu

Une fois connecté, le maître de jeu est redirigé vers l'interface d'accueil de la plateforme PlayLearn. L'interface propose ensuite deux actions principales :

- **Lancer une partie** : permet de démarrer une nouvelle session du DGC.
- **Découvrir notre jeu** : permet d'en apprendre davantage sur le contenu du jeu.

Cette page constitue le point de départ pour toute interaction du maître de jeu avec la plateforme. La figure 4.6 montre l'interface d'accueil du maître de jeu après authentification.

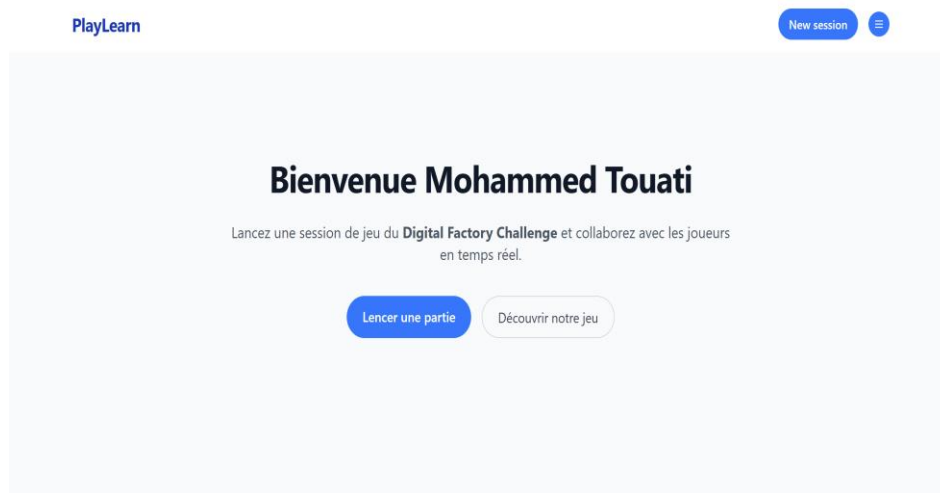


Figure 4.6 l'interface d'accueil après authentification.

#### 4.4.3.1 Démarrer une session

Lorsqu'il clique sur « Lancer une partie » depuis son interface d'accueil, le MJ accède à l'écran de configuration de la session de jeu. Cette étape lui permet de personnaliser le déroulement du DGC avant de le lancer en direct avec les joueurs.

Une fois la configuration terminée, le maître de jeu clique sur le bouton « Commencer » en haut à droite pour générer la session et le code d'adhésion que les joueurs utiliseront pour rejoindre la partie. La figure 4.7 présente l'interface de démarrage d'une session de jeu par le maître de jeu.

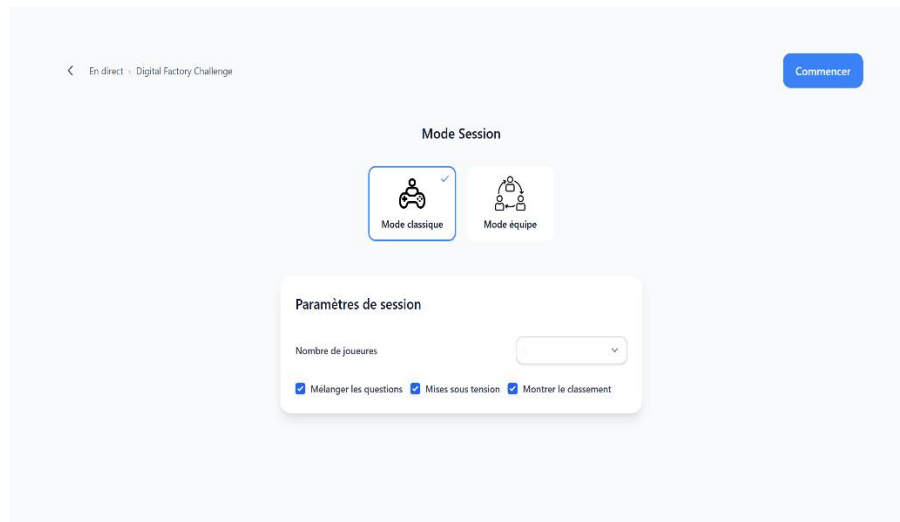


Figure 4.7 Interface de démarrage d'une session

#### 4.4.3.2 Lancer le jeu

Après avoir configuré et démarré la session, une page d'attente s'affiche. Celle-ci contient :

- Le code d'adhésion unique de la session, que les joueurs peuvent utiliser pour se connecter via n'importe quel appareil.
- La liste des participants ayant rejoint la session, mise à jour en temps réel.
- Le bouton « Commencez maintenant » qui permet au maître de jeu de démarrer la session dès qu'au moins un joueur a rejoint.

La figure 4.8 montre l'interface d'attente et de lancement du jeu après le démarrage de la session.

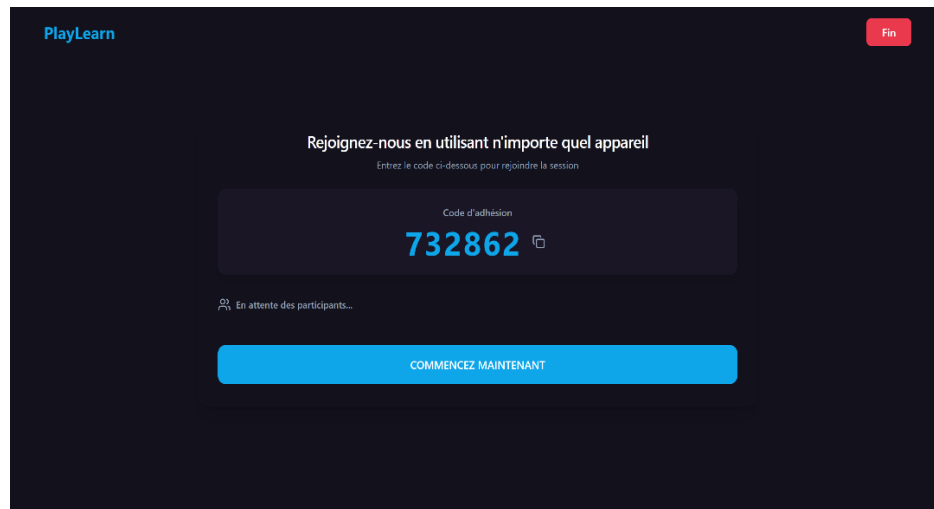


Figure 4.8 Interface d'attente des joueurs

#### 4.4.3.3 Observer les performances

Après avoir lancé le jeu, le MJ accède à une interface dédiée permettant d'observer en temps réel les performances des participants. Cette fonctionnalité est essentielle pour suivre la progression individuelle ou collective des joueurs tout au long de la session.

L'interface de suivi des performances affiche :

- Le classement des participants, mis à jour dynamiquement en fonction des performances de chaque joueur.
- Les informations clés telles que le rang, le nom du joueur et le montant de cash, initialement attribué et mis à jour automatiquement selon les réponses et les actions du joueur au cours du jeu.
- Un bouton « Fin » permettant au MJ de clôturer la session à tout moment.

La figure 4.9 présente l'interface de classement en direct utilisée pour visualiser les performances des joueurs pendant le déroulement du jeu.

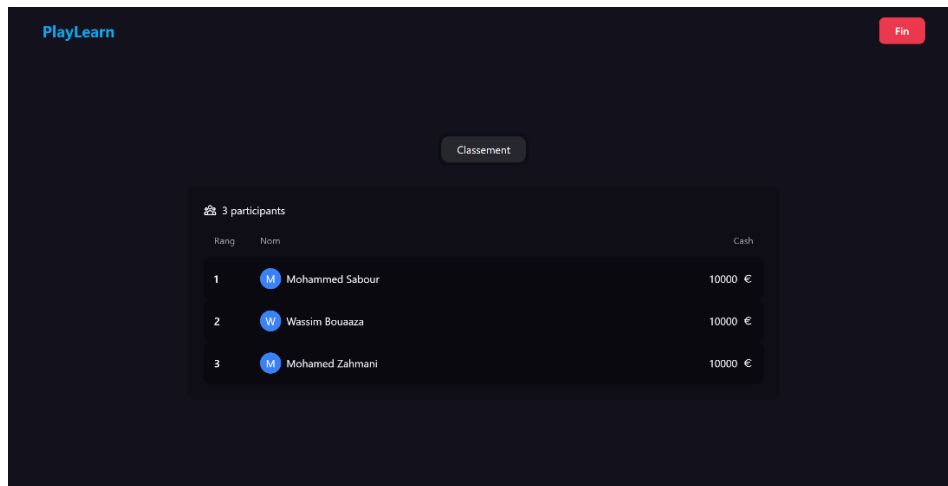


Figure 4.9 interface Dashboard observation

## 4.4.4 Joueur

### 4.4.4.1 Rejoindre une session

Pour participer à une session de jeu, le joueur doit effectuer deux étapes successives, illustrées dans les interfaces suivantes :

#### – Saisie du code d’adhésion

Lorsque le joueur accède à la plateforme, il est invité à entrer le code d’adhésion à six chiffres qui lui a été fourni par le maître de jeu (Voir Figure 4.10). Ce code permet d’identifier la session active à rejoindre. Une fois le code est correct et que la session est accessible, l’utilisateur est redirigé vers l’étape suivante.

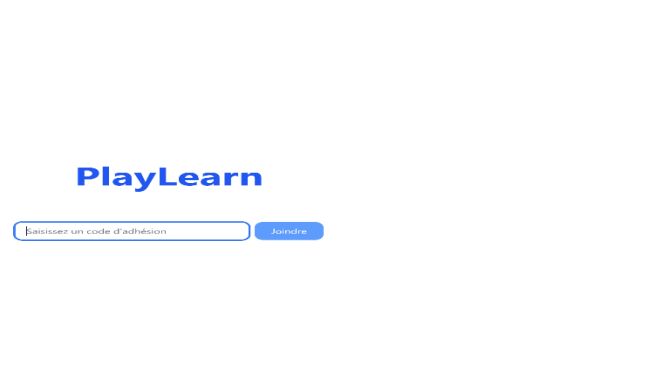


Figure 4.10 Interface de saisir un code d’adhésion

### – Saisie du pseudonyme

Une fois le code validé, le joueur est invité à entrer un pseudonyme. (Voir Figure 4.11)

Ce nom sera utilisé pour l'identifier dans la session. Après validation du pseudonyme :

- Le joueur est enregistré comme participant à la session.
- Il est automatiquement redirigé vers l'interface d'attente, en attendant que le maître de jeu lance le début de la partie.

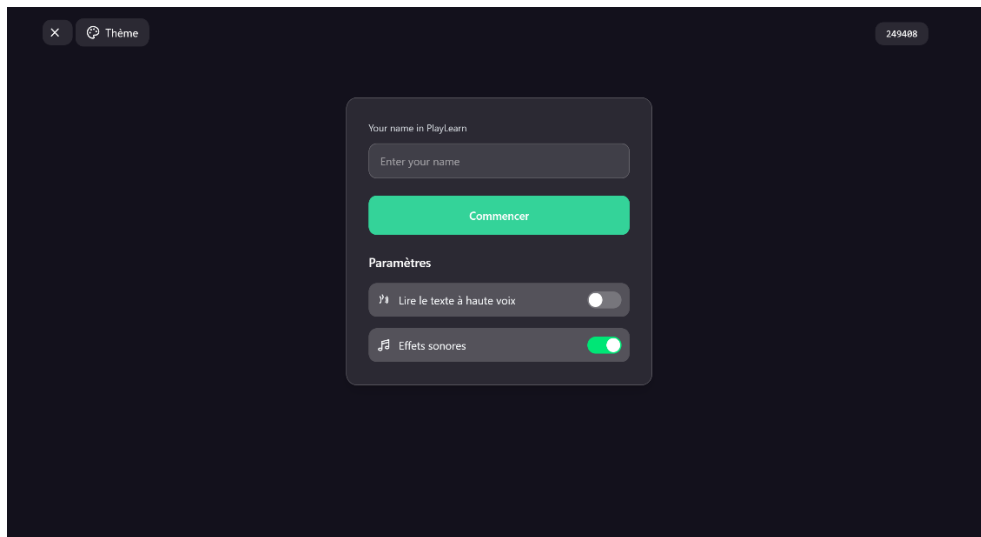


Figure 4.11 Interface de saisir un pseudonyme

#### 4.4.4.2 Participer au jeu DFC

Une fois le jeu est lancé par le MJ, le joueur accède progressivement aux différentes activités du jeu DFC. Le parcours se déroule en deux grandes phases :

- **Activité initiale : Quiz** Le joueur commence par un quiz destiné à évaluer ses connaissances de base. L'interface présente des questions à choix multiples (QCM) et à choix unique, Les réponses permettent d'accumuler un cash initial.

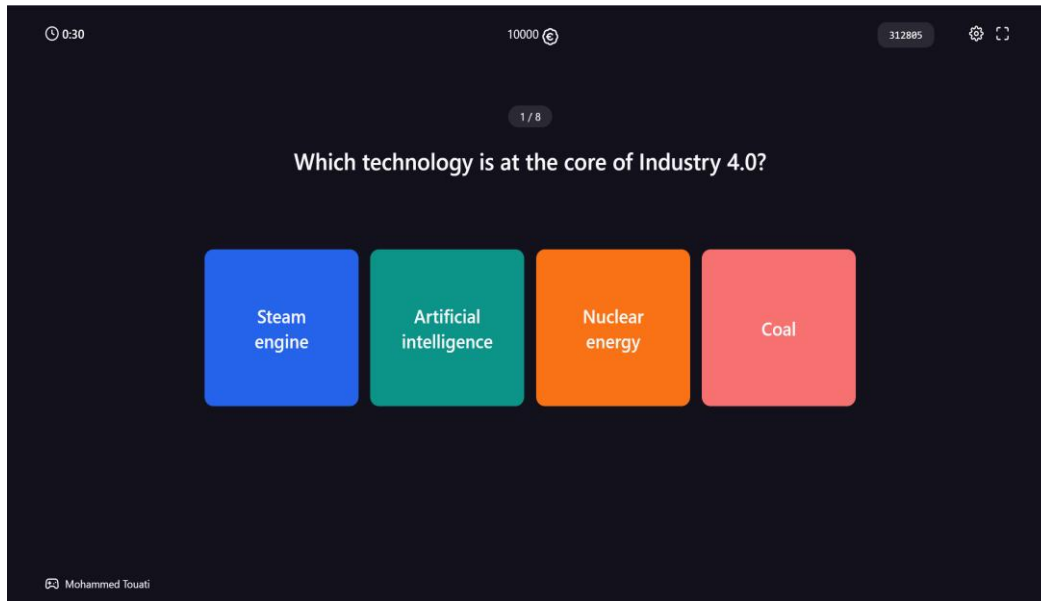


Figure 4.12 Interface de quiz

- **Cycles de jeu :** Après le quiz, le joueur entre dans un cycle pédagogique répété plusieurs fois, composé de deux étapes :

- **Activité de Prise de Décision**

Le joueur est invité à investir dans de nouveaux équipements.



Figure 4.13 Interface de décision

### ○ Activité de Simulation

Une fois toutes les décisions soumises, une simulation est générée.

- L'interface affiche les résultats obtenus suite aux décisions.
- Le joueur analyse ces résultats pour ajuster sa stratégie au prochain cycle.

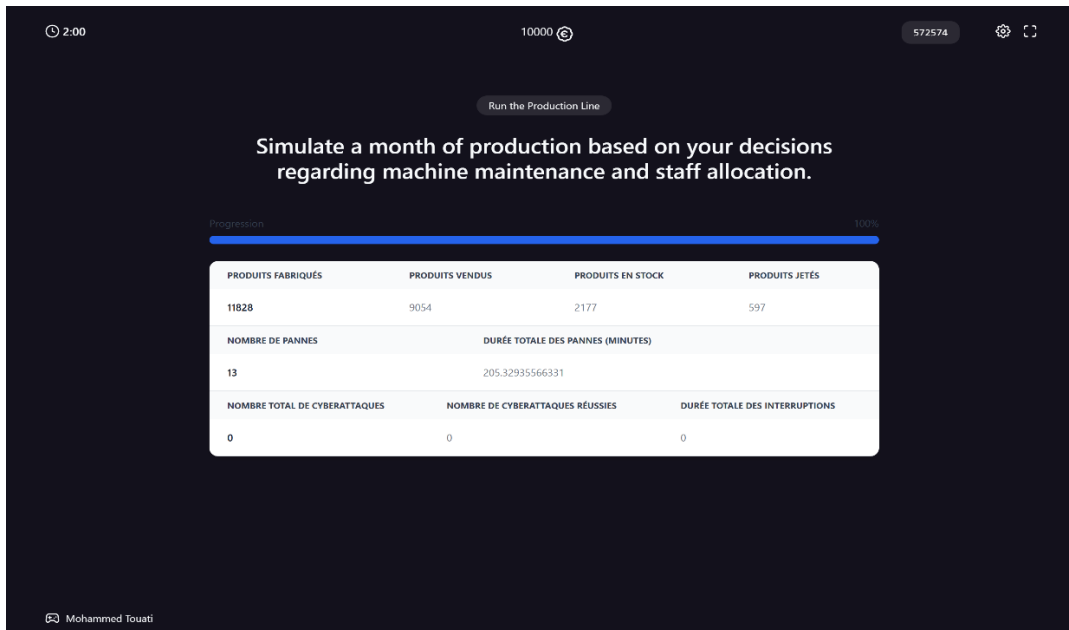


Figure 4.14 Interface de simulation

## 4.5 Conclusion

Dans ce chapitre nous avons présenté les aspects pratiques liés à la réalisation de notre application, à savoir les outils de développement nécessaires pour le fonctionnement de notre application suivi du diagramme de déploiement associé à notre système. En dernier, nous avons illustré quelques interfaces de l'application.

# Conclusion Générale

À travers ce mémoire, nous avons exploré l'univers des Serious Games en mettant en lumière leur potentiel pédagogique, notamment dans les domaines de l'industrie et de la logistique. Nous avons constaté que, malgré leur efficacité reconnue, les jeux sérieux restent encore peu intégrés dans les dispositifs de formation, principalement en raison de leur coût, de leur complexité de conception, et du manque d'outils adaptés.

Face à ces constats, nous avons proposé une solution concrète et accessible : PlayLearn, une plateforme web dédiée à la gestion d'un Serious Game collaboratif. Conçue pour faciliter l'animation de sessions pédagogiques interactives, cette plateforme intègre le jeu Digital Factory Challenge, un scénario original inspiré des problématiques réelles de l'usine numérique et des des jeu sérieux existe, où les participants prendre des décisions stratégiques, tout en gérant les impacts sur divers paramètres industriels.

Ce travail nous a permis de mobiliser des compétences en ingénierie logicielle, en conception pédagogique et en développement web, tout en répondant à une problématique actuelle : comment rendre les Serious Games plus accessibles, adaptables et efficaces dans le cadre de la formation.

Bien que la solution développée soit pleinement fonctionnelle, certaines perspectives d'amélioration peuvent être envisagées, telles que l'intégration d'une intelligence artificielle pour analyser les décisions des joueurs, l'ajout de scénarios personnalisables.

En conclusion, ce projet illustre l'intérêt et la faisabilité de concevoir des environnements numériques ludiques pour l'apprentissage, tout en ouvrant la voie à de nouvelles pratiques pédagogiques plus interactives, immersives et collaboratives.

# Bibliographie

- [1] Dondlinger, «Educational video game design: A review of the literature.,» *Journal of Applied Educational Technology*, pp. 41-48, 2007.
- [2] Abt, C. C. (1970). *Serious games*. New York: Viking Press.
- [3] Ryan, R. M., & Deci, E. L. (2000). Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being. *American Psychologist*, 55(1), 68–78.
- [4] J. Alvarez, Du jeu vidéo au serious game : approches culturelle, pragmatique et formelle, thèse de doctorat, Université de Toulouse II – Le Mirail, 2007.
- [5] Sanchez, É. (2015, 7 avril). Quels transferts de connaissances par le jeu numérique dans l’enseignement ? Communication présentée à la Journée EHESS : Les objets communicants dans l’écologie des enfants de 0-12 ans.
- [6] Alvarez, J., & Djaouti, D. (2012). Introduction au serious game. Paris : Questions Théoriques.
- [7] INSEE. (n.d.). Définition - Industrie. Institut national de la statistique et des études économiques. <https://www.insee.fr/fr/metadonnees/definition/c1426>
- [8] Christopher, M. (2016). Logistique et supply chain management: Stratégies pour réduire les coûts et améliorer les services (5<sup>e</sup> éd.). Pearson.
- [9] <https://inchainge.com/business-games/tfc/>
- [10] <https://erpsim.hec.ca/index.php/en/erpsim>
- [11] Roques, P., & Vallée, F. (2006). UML en action : de l’analyse des besoins à la conception. Paris : Eyrolles.

- [12] Roques, P., & Vallée, F. (2006). Les cahiers du programmeur UML 2 : Modéliser une application Web. Paris : Eyrolles.
- [13] Audibert, L. (2007). UML 2 : De l'apprentissage à la pratique. Paris : Eyrolles.
- [14] O. Sigaud. Introduction à la modélisation orientée objets.
- [15] Gaby, J., & Gaby, D. (2007). UML 2 : Analyse et conception. Saint-Herblain : Éditions ENI.
- [16] Gilles et Roy. UML2 modéliser une application web. EYROLLES. Paris, 4ème édition. 2008.
- [17] [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp)
- [18] <https://developer.mozilla.org/fr/docs/Web/CSS>
- [19] <https://tailwindcss.com/docs>
- [20] <https://www.futura-sciences.com/tech/definitions/internet-javascript-509/>
- [21] Alpine.js Documentation. "Getting Started." *Alpine.js*, <https://alpinejs.dev/start>
- [22] <https://www.php.net/docs.php>
- [23] Laravel Documentation. "The PHP Framework for Web Artisans." *Laravel*, <https://laravel.com/docs>
- [24] Livewire Documentation. "What is Livewire?" *Laravel Livewire*, <https://livewire.laravel.com/docs/introduction>
- [25] Visual Studio Code Documentation. "Visual Studio Code – Code Editing. Redefined." *Visual Studio Code*, <https://code.visualstudio.com/docs>
- [26] <https://www.computerhope.com/jargon/d/drawio.htm>
- [27] <https://www.ibm.com/docs/fr/rational-soft-arch/9.5.0?topic=diagrams-deployment>

# Annexe

- Diagramme de séquence du cas d'utilisation – Participer au quiz

Lorsque le joueur accède à l'activité de quiz, le système lui affiche les questions une par une. Chaque question peut être à choix multiple ou à choix unique. Le joueur répond à chaque question individuellement. Après chaque réponse, le système enregistre la réponse dans la base de données, puis affiche la question suivante. Ce processus se répète jusqu'à la dernière question. Une fois toutes les réponses enregistrées, le système passe à l'activité suivante du jeu.

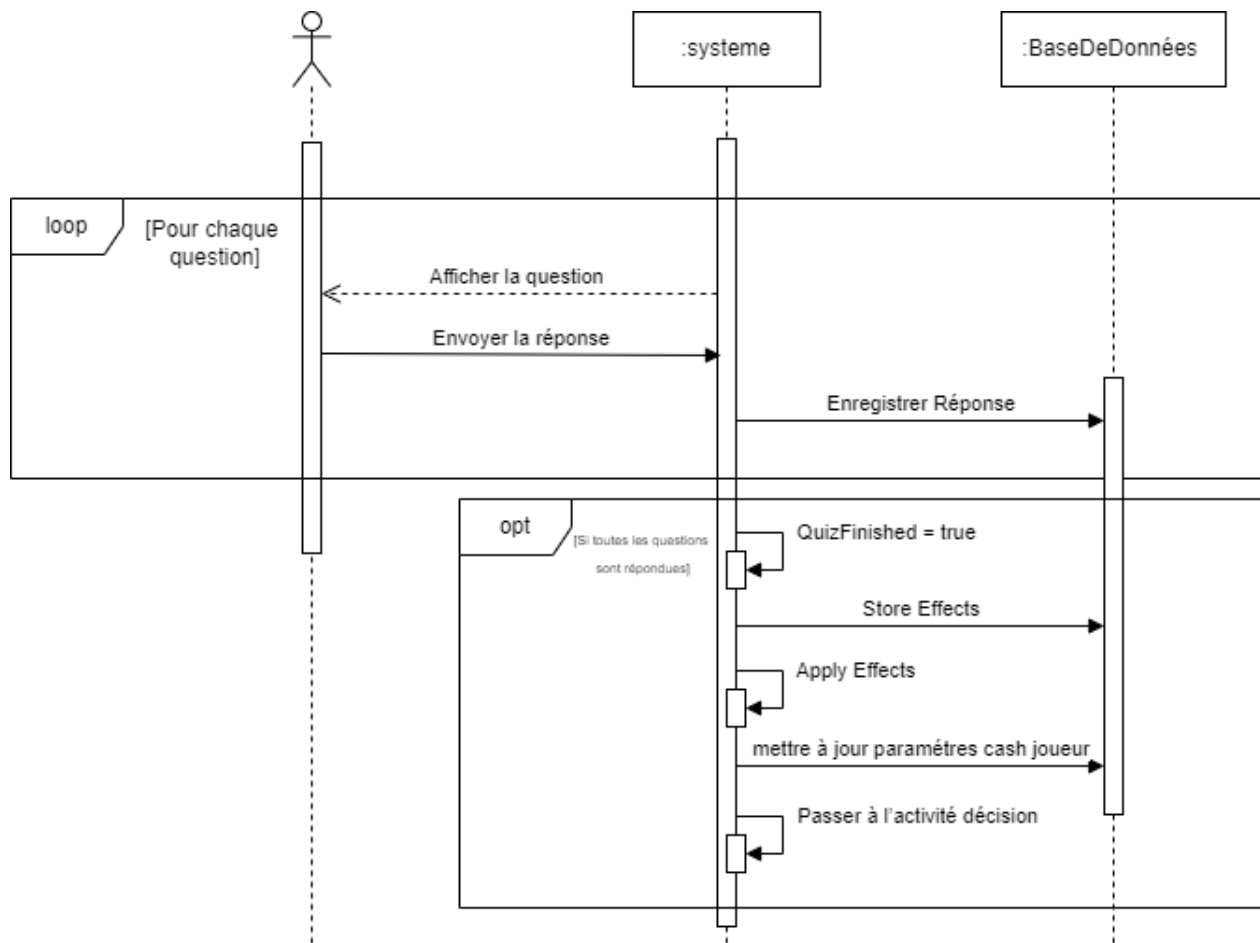


Figure 4.15 – Diagramme de séquence du cas d'utilisation – Participer au quiz

- Diagramme de séquence du cas d'utilisation – Participer à une décision

Lorsque le joueur accède à l'activité de décision, le système affiche l'interface de décision et démarre un minuteur de 2 minutes. Le joueur choisit une technologie, puis un niveau d'investissement. Le système vérifie si le joueur dispose de suffisamment de cash. Si c'est le cas, les effets de la décision sont appliqués, les paramètres du joueur sont mis à jour, et l'investissement est enregistré dans la base de données. Un message de confirmation est affiché. Sinon, un message d'erreur informe le joueur de l'échec de la décision. À l'issue du délai, le système passe automatiquement à l'activité suivante : la simulation.

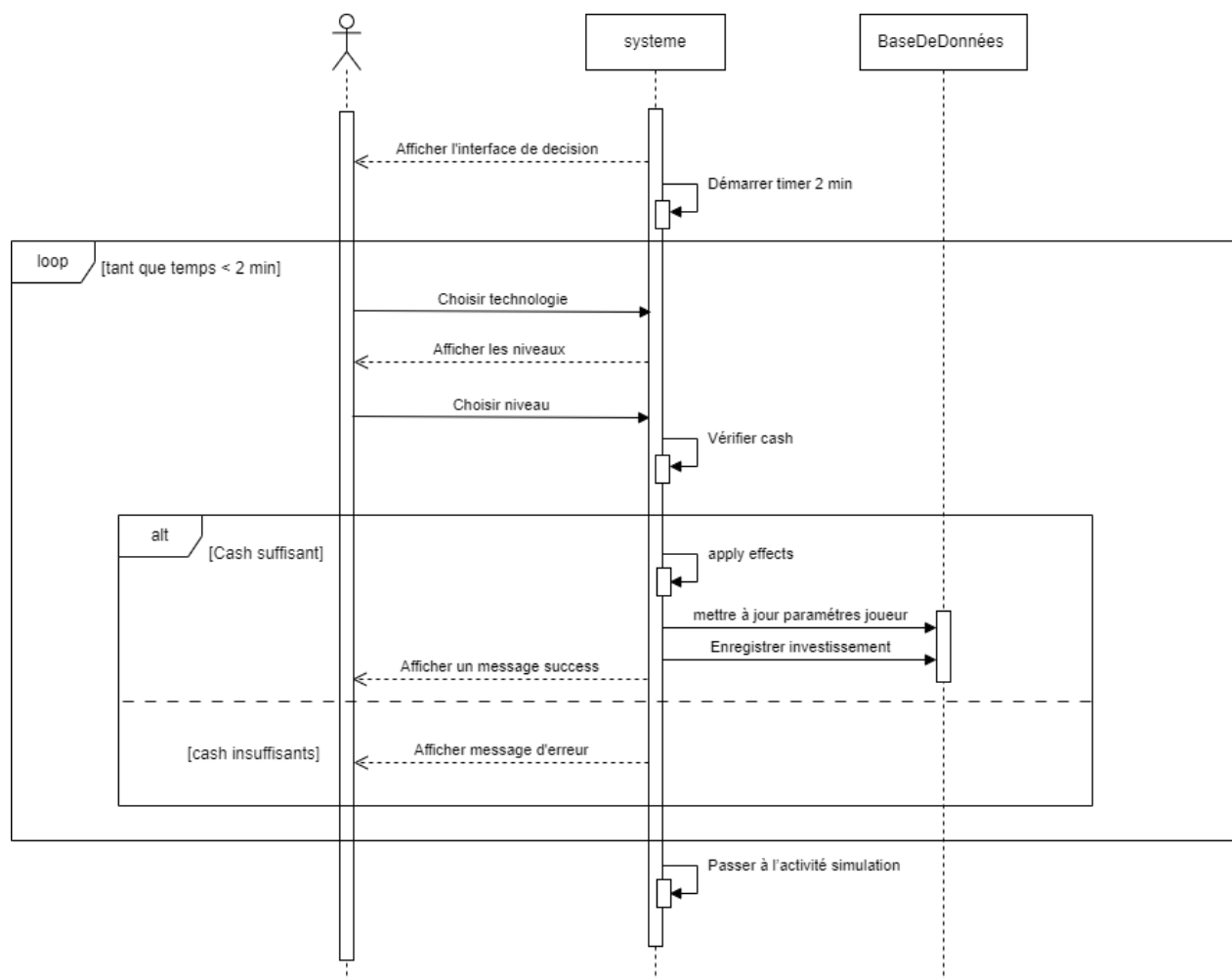


Figure 4.16 – Diagramme de séquence du cas d'utilisation – Participer à une décision

- Diagramme de séquence du cas d'utilisation – Participer à une simulation

Lorsque le joueur accède à l'activité de simulation, le système affiche l'interface de simulation et démarre un minuteur de 2 minutes. Le système récupère alors les paramètres du joueur depuis la base de données, puis lance le simulateur avec ces paramètres. Pendant ces 2 minutes, la simulation s'exécute, et le système affiche les résultats de la simulation à l'écran. À la fin du temps imparti, la simulation prend fin automatiquement.

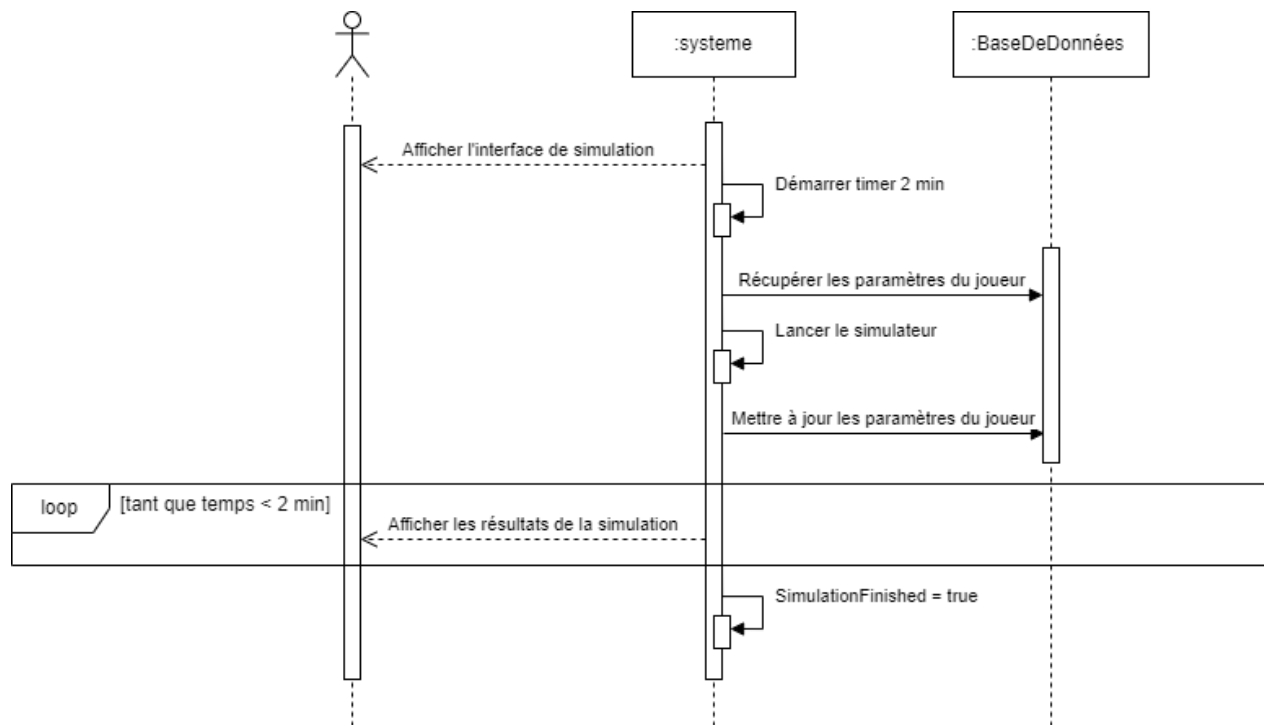


Figure 4.17 – Diagramme de séquence du cas d'utilisation – Participer à une simulation

الجمهورية الجزائرية الديمقراطية الشعبية  
وزارة التعليم العالي والبحث العلمي

جامعة عبد الحميد بن باديس - مستغانم -  
كلية العلوم الدقيقة والإعلام الآلي

تصريح شرفي خاص بالالتزام بقواعد النزاهة العلمية  
لإنجاز البحث

أنا الممضي أدناه،

الطالب(ة): صبور محمد تواتي رقم التسجيل الجامعي: 202037033343  
الحامل لبطاقة التعريف الوطنية رقم: 408376106 والصادرة بتاريخ: 2024/02/07  
عن بلدية سيدي لخضر - مستغانم

المسجل بـ: كلية العلوم الدقيقة والإعلام الآلي / قسم: الرياضيات والإعلام الآلي  
شعبة الإعلام الآلي / التخصص: هندسة أنظمة المعلومات.

والمكلف بإنجاز مذكرة ماستر بعنوان: تصميم وتطوير تطبيق ويب للعبة تعاونية جادة.

أصريح بشرفي أنني ألتزم بمراعاة المعايير العلمية والمنهجية ومعايير الأخلاقيات العلمية والنزاهة الأكاديمية  
المطلوبة في إنجاز البحث، وأتحمل المسؤولية الشخصية عن كل المحتوى المتضمن في البحث المذكور أعلاه.

التاريخ: 2025/06/22

إمضاء المعني



\* ملحق القرار الوزاري رقم 933 المؤرخ في 28 جويلية 2016 الذي يحدد القواعد المتعلقة بالوقاية من السرقة العلمية ومكافحتها.