



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de
HONORIS UNITED UNIVERSITIES

PROJET DE FIN D'ANNEE

4^{ème} Année en Ingénierie Informatique et Réseaux

Conception et réalisation d'une Plateforme E-Games: Viking Gaming

Réalisé par :

BOUAZZA Yassine

ALIL Mouad Ali

Encadré par :

Mme. **OATIQ** Amina

M. ABG HOUR

Membre du jury :

Mme. **OATIQ** Amina

Filière :

Ingénierie informatique et réseaux

Année universitaire 2023/2024

Dédicace

À nos familles,

Pour votre soutien inébranlable et vos encouragements constants. Votre amour et votre compréhension ont été les piliers sur lesquels nous nous sommes appuyés tout au long de ce projet.

À nos amis,

Pour votre patience et votre motivation. Vous avez toujours su trouver les mots pour nous encourager et nous inspirer.

À notre encadrante Amina

Pour vos conseils avisés, votre disponibilité et votre soutien. Votre expertise et votre dévouement ont été essentiels à la réussite de ce projet.

À tous ceux qui ont contribué, de près ou de loin, à la réalisation de ce projet,

Votre aide précieuse et vos conseils ont été des éléments clés de notre succès. Nous vous dédions ce travail avec une profonde gratitude.

Remerciement

Nous tenons à exprimer notre profonde gratitude envers toutes les personnes qui ont contribué à la réalisation de ce projet.

En premier lieu, nous remercions chaleureusement notre encadrant, [Nom de l'encadrant(e)], pour ses conseils avisés, sa disponibilité et son soutien constant tout au long de ce projet. Ses orientations précieuses et son expertise ont été déterminantes pour la réussite de cette plateforme de gestion de jeux vidéo.

Nous remercions également toute l'équipe pédagogique de l'Ecole Marocaine des Sciences de l'Ingénieur pour la qualité de l'enseignement dispensé et les ressources mises à disposition des étudiants. Leur engagement à fournir une formation de haut niveau a été une source d'inspiration et de motivation.

Nous sommes également reconnaissants envers nos familles et nos amis pour leur soutien inébranlable et leurs encouragements tout au long de cette aventure. Leur compréhension et leur présence ont été des piliers essentiels dans la réalisation de ce travail.

Enfin, nous tenons à remercier toutes les personnes qui, de près ou de loin, ont contribué à la réussite de ce projet. Vos conseils, vos encouragements et vos retours constructifs ont été d'une aide précieuse.

Ce projet a été une expérience enrichissante, tant sur le plan professionnel que personnel, et nous sommes reconnaissants envers tous ceux qui ont participé à cette réussite.

Résumé

Ce rapport présente une application web permettant la gestion complète de jeux vidéo en ligne. Le projet a été réalisé en utilisant les technologies Django pour le backend et Bootstrap pour le frontend, offrant ainsi une interface utilisateur moderne et responsive.

L'objectif principal de cette plateforme est de fournir aux utilisateurs une expérience fluide et intuitive pour parcourir, acheter et gérer leurs jeux vidéo. Les principales fonctionnalités de l'application incluent :

- **Authentification des utilisateurs** : Les utilisateurs peuvent s'inscrire, se connecter, et gérer leur profil.
- **Gestion des produits** : Les administrateurs peuvent ajouter, modifier, et supprimer des jeux vidéo disponibles sur la plateforme.
- **Panier d'achat** : Les utilisateurs peuvent ajouter des jeux à leur panier, visualiser le total des achats, et procéder à la commande.
- **Gestion des commandes** : Les utilisateurs peuvent consulter l'historique de leurs commandes et les administrateurs peuvent gérer les commandes reçues.

L'analyse des besoins a permis d'identifier les exigences fonctionnelles et non fonctionnelles de l'application, ainsi que les acteurs principaux et leurs interactions avec le système. La conception de l'application a été réalisée en utilisant les diagrammes UML, incluant des diagrammes de cas d'utilisation, de classes, et de séquences pour illustrer les processus clés.

La Réalisation du projet a impliqué l'utilisation de plusieurs outils et technologies, notamment Python pour le développement backend, Django comme Framework web, Bootstrap pour le design frontend, et SQLite pour la gestion de la base de données. L'environnement de développement utilisé était Visual Studio Code.

En conclusion, ce projet a été une expérience enrichissante, permettant d'acquérir de nouvelles compétences en développement web, en gestion de projet, et en collaboration d'équipe.

Liste des abréviations

- ✓ **IDE** : Integrated Development Environment (Environnement de développement intégré)
- ✓ **ORM** : Object-Relational Mapping (Mapping Objet-Relationnel)
- ✓ **HTML** : HyperText Markup Language (Langage de balisage hypertexte)
- ✓ **CSS** : Cascading Style Sheets (Feuilles de style en cascade)
- ✓ **JS** : JavaScript
- ✓ **SQL** : Structured Query Language (Langage de requête structuré)
- ✓ **HTTP** : HyperText Transfer Protocol (Protocole de transfert hypertexte)
- ✓ **HTTPS** : HyperText Transfer Protocol Secure (Protocole de transfert hypertexte sécurisé)
- ✓ **UI** : User Interface (Interface utilisateur)
- ✓ **CRUD** : Create, Read, Update, Delete (Créer, Lire, Mettre à jour, Supprimer)
- ✓ **JSON** : JavaScript Object Notation (Notation d'objet JavaScript)
- ✓ **API** : Application Programming Interface (Interface de programmation d'application)
- ✓ **MVC** : Model-View-Controller (Modèle-Vue-Contrôleur)
- ✓ **MVT** : Model-View-Template (Modèle-Vue-Template)
- ✓ **CSS3** : Cascading Style Sheets Level 3 (Feuilles de style en cascade niveau 3)
- ✓ **SQL** : Structured Query Language (Langage de requête structuré)
- ✓ **SQLite** : SQL Light (Base de données légère intégrée)
- ✓ **VS Code** : Visual Studio Code
- ✓ **VCS** : Version Control System (Système de contrôle de version)
- ✓ **CI/CD** : Continuous Integration/Continuous Deployment (Intégration continue/Déploiement continu)
- ✓ **DOM** : Document Object Model (Modèle d'objet de document)

Table des matières

INTRODUCTION GENERALE.....	1
Chapitre 1 : Contexte général du projet.....	3
1. Présentation du projet.....	4
2. Problématique	4
3. Objectifs du projet.....	5
4. Méthodologie de gestion de projet.....	5
Chapitre 2 : Modélisation & Conception de l'application.....	8
1. Analyse des besoins	9
1.1 Identification des acteurs.....	9
1.2 Besoins Fonctionnel	9
1.3 Besoins non Fonctionnel.....	10
2. Diagramme UML	10
2.1 Diagrammes de Cas d'utilisation.....	11
2.2 Diagramme de Classes.....	12
2.3 Diagrammes de Séquence.....	13
Chapitre 3 : Mise en œuvre	17
1. Outils et technologies utilisés	18
1.1 Django.....	18
1.2 SQLite.....	18
1.3 JavaScript.....	19
1.4 Bootstrap.....	19
1.5 Visual Studio Code.....	19
1.6 Git.....	20
1.7 GitHub.....	20
1.8 Jira.....	20
2. Interfaces graphiques.....	21
CONCLUSION GENERALE	25

INTRODUCTION GÉNÉRALE

Le secteur des jeux vidéo connaît une croissance exponentielle et constitue aujourd'hui l'une des industries les plus dynamiques du monde du divertissement. La gestion des jeux vidéo en ligne, ainsi que l'expérience utilisateur, sont devenues des aspects cruciaux pour répondre aux attentes des joueurs et des administrateurs. C'est dans ce contexte que le projet E-Games Shop a été conçu.

L'objectif de ce projet est de développer une plateforme web permettant la gestion et l'achat de jeux vidéo en ligne. Cette application vise à offrir aux utilisateurs une expérience intuitive et fluide, tout en permettant aux administrateurs de gérer efficacement les produits et les commandes. Pour ce faire, nous avons choisi de développer cette application en utilisant Django pour le backend et Bootstrap pour le frontend, deux technologies robustes et éprouvées.

Le projet E-Games Shop s'articule autour de plusieurs fonctionnalités clés :

- **Authentification des utilisateurs** : Permet aux utilisateurs de s'inscrire, se connecter, et gérer leur profil de manière sécurisée.
- **Gestion des produits** : Les administrateurs peuvent ajouter, modifier et supprimer des jeux vidéo.
- **Panier d'achat** : Les utilisateurs peuvent ajouter des jeux à leur panier, visualiser le total des achats et procéder à la commande.
- **Gestion des commandes** : Les utilisateurs peuvent consulter l'historique de leurs commandes, et les administrateurs peuvent gérer les commandes reçues.

Ce projet se distingue par son approche centrée sur l'utilisateur et son souci de fournir une interface moderne et responsive. La mise en œuvre de ce projet a impliqué une analyse approfondie des besoins, la conception de l'architecture logicielle via des diagrammes UML, et l'utilisation de technologies avancées pour garantir performance et sécurité.

Le présent rapport est structuré en plusieurs chapitres. Nous commencerons par une présentation détaillée du contexte et des objectifs du projet. Ensuite, nous analyserons les besoins fonctionnels et techniques de l'application. Nous décrirons la modélisation et la conception de l'application en utilisant des diagrammes UML. Puis, nous exposerons les outils et technologies utilisés pour la mise

en œuvre du projet, en détaillant les interfaces graphiques développées. Enfin, nous concluons avec une évaluation des résultats obtenus, les compétences acquises et les perspectives d'amélioration.

Ce projet a représenté une opportunité unique de mettre en pratique les compétences techniques et de gestion de projet acquises tout au long de notre formation. Il a également permis d'explorer de nouvelles technologies et de développer une application répondant aux besoins réels des utilisateurs et des administrateurs de jeux vidéo.

Chapitre 1 :

Contexte général du projet.

Introduction

Dans ce chapitre, nous allons présenter le contexte général du projet E-Games Shop, les objectifs spécifiques de ce projet, et l'importance de ce dernier dans le cadre de notre formation. Nous aborderons la problématique que ce projet vise à résoudre, puis nous décrirons les objectifs définis et la méthodologie adoptée pour mener à bien ce projet.

1. Présentation du projet

L'industrie des jeux vidéo est en constante évolution, avec une croissance rapide de la demande pour des plateformes en ligne permettant l'achat et la gestion de jeux. E-Games Shop s'inscrit dans cette dynamique en offrant une solution complète et moderne pour les utilisateurs et les administrateurs.

Le projet consiste en la création d'une application web permettant la gestion de jeux vidéo en ligne. Les principales fonctionnalités de l'application incluent :

- **Authentification des utilisateurs** : Un système sécurisé permettant aux utilisateurs de s'inscrire, de se connecter et de gérer leur profil.
- **Gestion des produits** : Une interface pour les administrateurs afin d'ajouter, de modifier et de supprimer des jeux vidéo.
- **Panier d'achat** : Une fonctionnalité permettant aux utilisateurs de sélectionner des jeux, de visualiser le total des achats et de finaliser leur commande.
- **Gestion des commandes** : Un historique des achats pour les utilisateurs et un outil de gestion des commandes pour les administrateurs.

2. Problématique

La gestion des jeux vidéo en ligne présente plusieurs défis, notamment en termes de sécurité, de gestion des produits et de fluidité de l'expérience utilisateur. Les utilisateurs recherchent des plateformes intuitives et sécurisées où ils peuvent facilement parcourir et acheter des jeux. De leur côté, les administrateurs ont besoin d'outils efficaces pour gérer les produits et les commandes.

La problématique principale que ce projet vise à résoudre est la complexité de la gestion des jeux vidéo en ligne tout en garantissant une expérience utilisateur optimale et une sécurité renforcée.

Les principaux défis identifiés sont :

- **Sécurité des données** : Assurer la protection des informations personnelles et des transactions des utilisateurs.
- **Gestion efficace des produits** : Fournir aux administrateurs des outils intuitifs pour ajouter, modifier et supprimer des jeux vidéo.
- **Expérience utilisateur fluide** : Offrir une interface conviviale et responsive pour améliorer la satisfaction des utilisateurs.
- **Gestion des commandes** : Permettre aux utilisateurs de suivre facilement leurs achats et aux administrateurs de traiter les commandes efficacement.

3. Objectifs du projet

Les objectifs spécifiques du projet E-Games Shop sont les suivants :

- **Développer une plateforme intuitive et sécurisée** permettant aux utilisateurs de gérer leurs jeux vidéo.
- **Fournir aux administrateurs des outils efficaces** pour la gestion des produits et des commandes.
- **Assurer une expérience utilisateur fluide et agréable** grâce à une interface moderne et responsive.
- **Mettre en place des fonctionnalités de sécurité robustes** pour protéger les données des utilisateurs et des transactions.

Ces objectifs se traduisent par des fonctionnalités spécifiques à développer, telles que :

- **Système d'authentification** : Inscription, connexion, et gestion du profil utilisateur.
- **Interface de gestion des produits** : Ajout, modification et suppression de jeux vidéo.
- **Fonctionnalité de panier d'achat** : Sélection de jeux, calcul du total des achats, et passage de commande.
- **Historique des commandes** : Suivi des achats par les utilisateurs et gestion des commandes par les administrateurs.
- **Mesures de sécurité** : Cryptage des données, authentification sécurisée, et protection contre les attaques.

4. Méthodologie de gestion de projet

Pour mener à bien ce projet, nous avons adopté une méthodologie de gestion de projet agile, en particulier la méthode Scrum, permettant une flexibilité et une adaptation rapide aux besoins changeants du projet. Cette approche nous a permis de développer et d'itérer rapidement sur les

fonctionnalités, en intégrant les retours des utilisateurs et des parties prenantes tout au long du processus.

Les principales étapes de notre méthodologie incluent :

1. **Backlog Produit** : Création et priorisation des fonctionnalités et des tâches à réaliser.
 - Le Product Owner est responsable de la gestion du backlog produit, qui contient toutes les fonctionnalités souhaitées pour l'application.
 - Les éléments du backlog sont priorisés en fonction de leur importance et de leur valeur ajoutée pour les utilisateurs.
2. **Sprint Planning** : Sélection des éléments du backlog produit à réaliser pendant le sprint et planification de leur implémentation.
 - Lors de la réunion de sprint planning, l'équipe Scrum sélectionne les éléments du backlog produit à réaliser pendant le sprint.
 - L'équipe planifie ensuite la façon dont ces éléments seront implémentés, en définissant des tâches spécifiques pour chaque membre de l'équipe.
3. **Cycle de Sprint** : Chaque sprint dure généralement 2 à 4 semaines et inclut des réunions quotidiennes (Daily Scrum) pour suivre l'avancement et ajuster le plan si nécessaire.
 - Pendant le sprint, l'équipe travaille sur les tâches planifiées et participe à des réunions quotidiennes pour discuter de l'avancement, des obstacles et des ajustements nécessaires.
 - À la fin de chaque sprint, l'équipe présente les fonctionnalités développées lors d'une réunion de revue de sprint.
4. **Tableau Scrum (Kanban)** : Utilisation d'un tableau visuel pour suivre l'avancement des tâches pendant le sprint, avec des colonnes pour les tâches à faire, en cours, et terminées.
 - Le tableau Scrum (ou Kanban) est utilisé pour visualiser l'avancement des tâches pendant le sprint.
 - Les tâches sont déplacées d'une colonne à l'autre (à faire, en cours, terminées) en fonction de leur état d'avancement.
5. **Sprint Review et Rétrospective** : Présentation des fonctionnalités terminées aux parties prenantes et analyse de ce qui a bien fonctionné ou non pour s'améliorer lors du prochain sprint.
 - Lors de la revue de sprint, l'équipe présente les fonctionnalités terminées aux parties prenantes et recueille leurs retours.
 - Lors de la rétrospective de sprint, l'équipe analyse ce qui a bien fonctionné et ce qui peut être amélioré pour les prochains sprints.

Conclusion

Ce premier chapitre a posé les bases du projet E-Games Shop en présentant le contexte général, la problématique, les objectifs et la méthodologie adoptée. La méthodologie Scrum a permis une gestion efficace du projet grâce à son approche itérative et flexible. Dans le chapitre suivant, nous entrerons dans le détail de la modélisation et de la conception de l'application, en utilisant des diagrammes UML pour illustrer les différents aspects du système.

Chapitre 2 :

Modélisation de l'application &
Conception

Introduction

Dans ce chapitre, nous allons détailler le processus de modélisation et de conception de l'application E-Games Shop. Nous commencerons par l'analyse des besoins pour identifier les exigences fonctionnelles et non fonctionnelles. Ensuite, nous présenterons les diagrammes UML pour illustrer l'architecture et les interactions du système, y compris les diagrammes de cas d'utilisation, de classes et de séquences.

1. Analyse des besoins

L'analyse des besoins est une étape cruciale pour s'assurer que toutes les fonctionnalités nécessaires sont bien identifiées et planifiées. Cette section décrit les besoins fonctionnels et non fonctionnels de l'application.

1.1. Identification des acteurs

Les acteurs principaux identifiés pour l'application sont :

- **Client** : Utilisateur de la plateforme qui achète et gère des jeux vidéo.
- **Administrateur** : Utilisateur avec des privilèges supplémentaires pour gérer les produits et les commandes.
- **Système** : Composant logiciel qui gère les différentes opérations de l'application.

1.2. Besoins fonctionnels

Les besoins fonctionnels décrivent les fonctionnalités spécifiques que l'application doit offrir aux utilisateurs et aux administrateurs.

- **Authentification des utilisateurs** :
 - Inscription des nouveaux utilisateurs.
 - Connexion des utilisateurs existants.
 - Gestion des profils utilisateurs (modification des informations personnelles, changement de mot de passe).
- **Gestion des produits** :
 - Ajout de nouveaux jeux vidéo.

- Modification des jeux existants.
- Suppression de jeux vidéo.
- **Panier d'achat :**
 - Ajout de jeux au panier.
 - Visualisation du contenu du panier.
 - Calcul du total des achats.
 - Passage de la commande.
- **Gestion des commandes :**
 - Consultation de l'historique des commandes par les utilisateurs.
 - Traitement et gestion des commandes par les administrateurs.

1.3. Besoins non fonctionnels

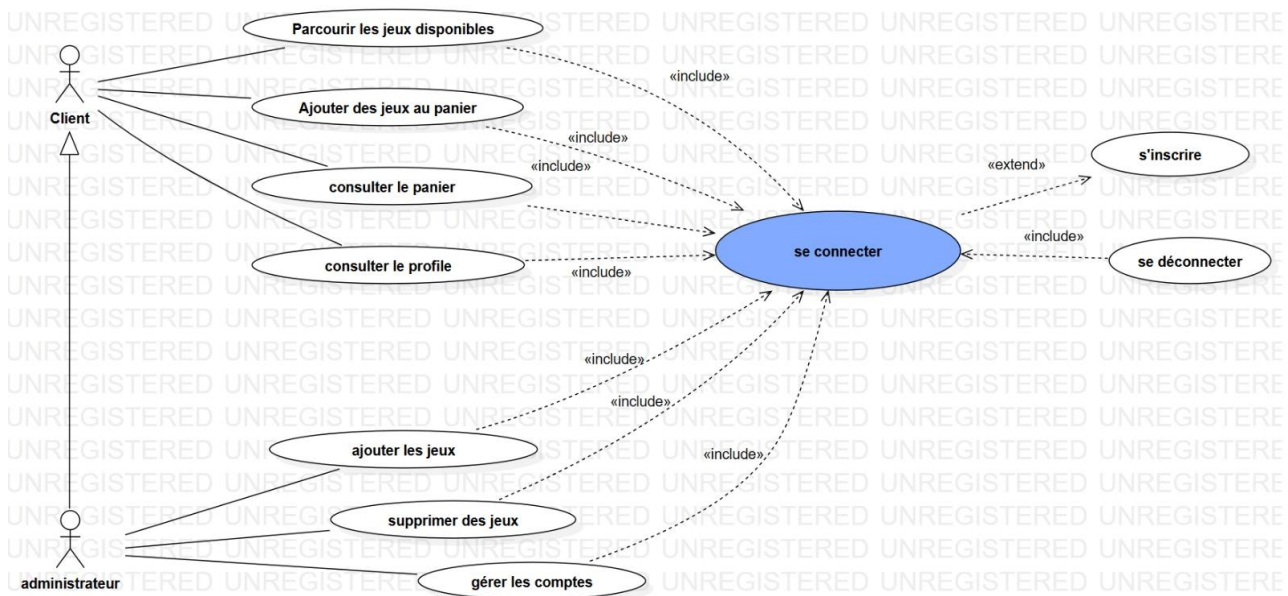
Les besoins non fonctionnels concernent les critères de performance, de sécurité et de convivialité de l'application.

- **Sécurité :**
 - Utilisation du HTTPS pour les communications sécurisées.
 - Cryptage des mots de passe.
 - Authentification sécurisée avec gestion des sessions.
- **Performance :**
 - Réponse rapide des pages web.
 - Optimisation des requêtes à la base de données.
- **Convivialité :**
 - Interface utilisateur intuitive et responsive.
 - Navigation simple et claire.
- **Scalabilité :**
 - Capacité à gérer un nombre croissant d'utilisateurs et de transactions.

2. Diagrammes UML

Les diagrammes UML (Unified Modeling Language) sont utilisés pour visualiser et documenter les différents aspects du système. Les principaux diagrammes utilisés dans ce projet sont les diagrammes de cas d'utilisation, de classes et de séquences.

2.1. Diagramme de Cas d'Utilisation

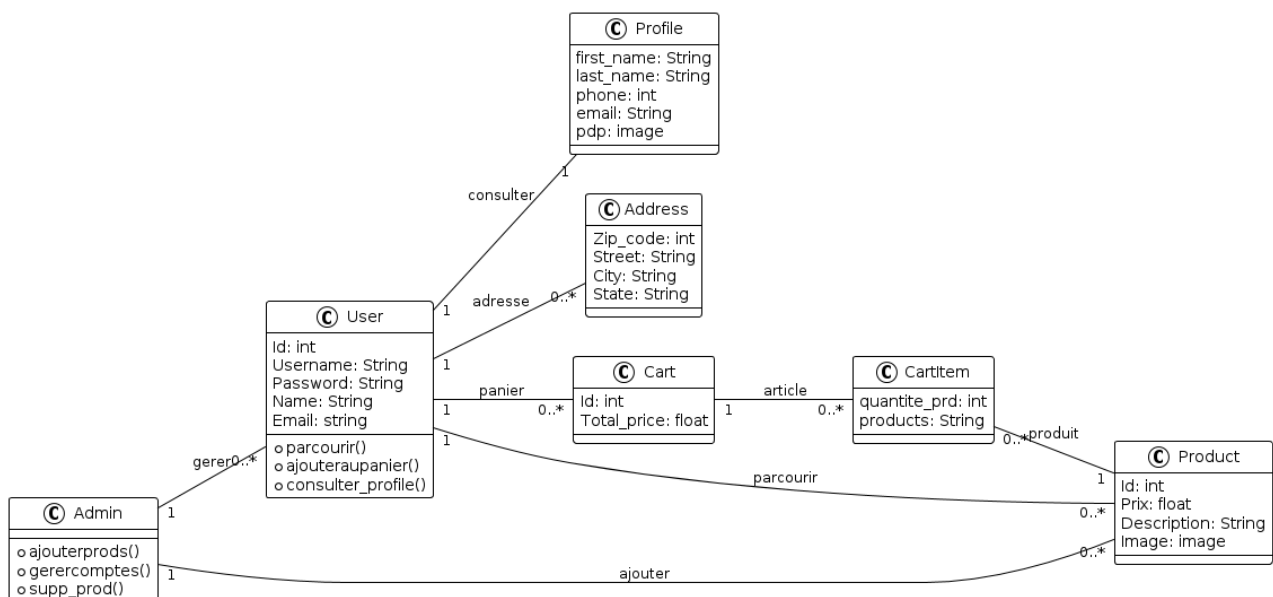


Explication du diagramme :

- Gérer Utilisateurs (Administrateur) :** Dans ce cas d'utilisation, l'administrateur est autorisé à effectuer différentes actions liées à la gestion des utilisateurs. Cela inclut :
 - Ajouter Utilisateur :** L'administrateur peut créer de nouveaux comptes utilisateurs en fournissant les informations nécessaires telles que le nom, l'identifiant et le mot de passe.
 - Modifier Utilisateur :** L'administrateur peut mettre à jour les informations des utilisateurs existants, par exemple en changeant leurs rôles ou leurs autorisations.
 - Supprimer Utilisateur :** L'administrateur peut désactiver ou supprimer des comptes utilisateurs qui ne sont plus nécessaires.
- Gérer Produits (Administrateur) :** L'administrateur a la possibilité de gérer les jeux vidéo disponibles dans le système.
 - Ajouter Produit :** L'administrateur peut ajouter de nouveaux jeux au système en fournissant les détails pertinents.
 - Modifier Produit :** L'administrateur peut mettre à jour les informations des jeux existants.
 - Supprimer Produit :** L'administrateur peut retirer des jeux du système qui ne sont plus nécessaires.
- Panier d'Achat (Client) :** Le client peut gérer son panier d'achat, y compris ajouter, modifier ou supprimer des articles.

- **Ajouter au Panier** : Le client peut ajouter des jeux à son panier.
 - **Modifier Panier** : Le client peut modifier les quantités des jeux dans son panier.
 - **Supprimer du Panier** : Le client peut retirer des jeux de son panier.
4. **Gérer Commandes (Client)** : Le client peut consulter l'historique de ses commandes et suivre l'état de ses achats.
- **Consulter Commandes** : Le client peut visualiser ses commandes passées et en cours.
 - **Annuler Commande** : Le client peut annuler une commande avant son traitement final.
5. **Authentification (Tous les utilisateurs)** : Les utilisateurs doivent s'authentifier pour accéder à leurs comptes et gérer leurs informations.
- **Se Connecter** : Les utilisateurs peuvent se connecter à leurs comptes en fournissant leurs identifiants.
 - **S'inscrire** : Les nouveaux utilisateurs peuvent créer un compte en fournissant les informations requises.

2.2. Diagramme de Classes

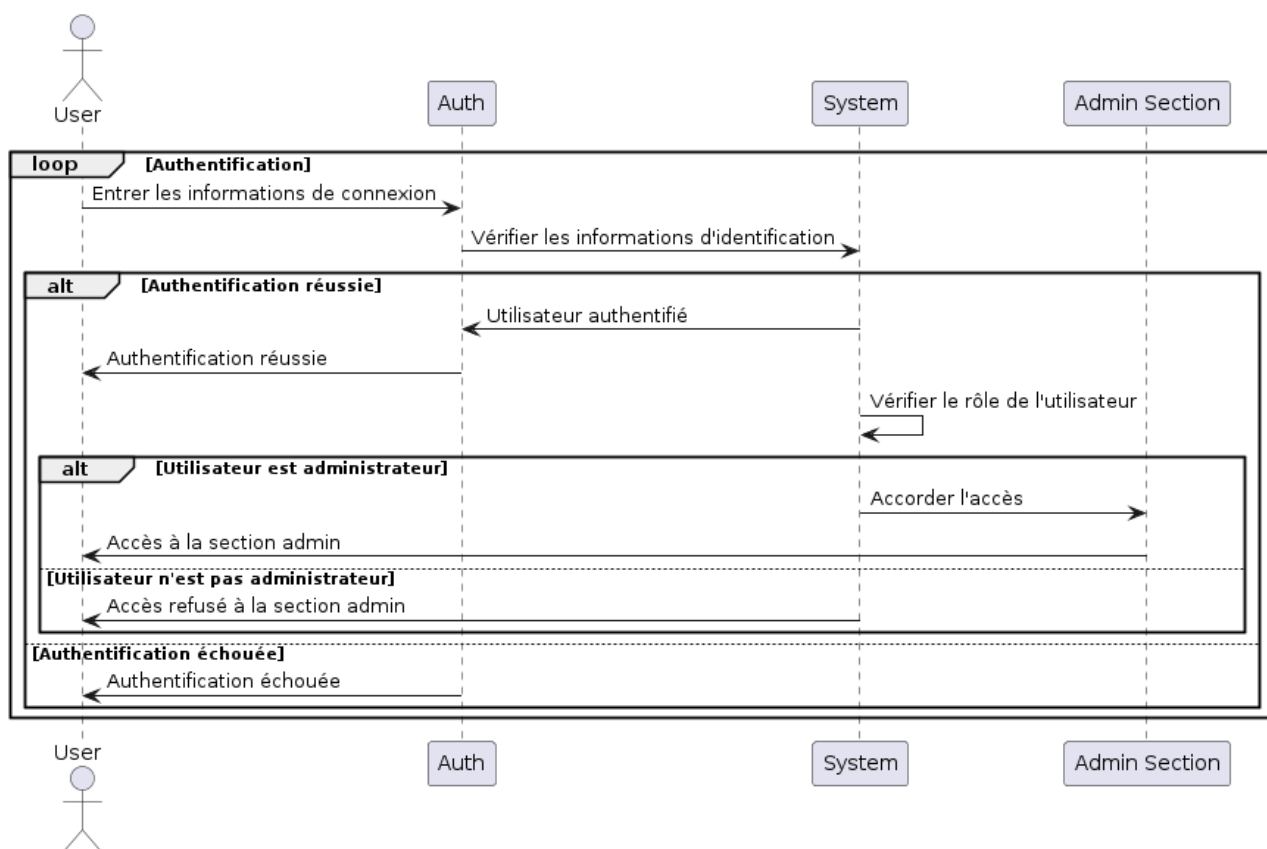


Explication du diagramme :

- **User** : Gère les informations des utilisateurs et leurs opérations.
- **Profile** : Contient les informations de profil utilisateur.
- **Address** : Gère les adresses des utilisateurs.
- **Cart** : Gère les paniers d'achat des utilisateurs.
- **CartItem** : Représente les articles dans un panier.
- **Product** : Représente les jeux vidéo disponibles.
- **Admin** : Représente les administrateurs avec des opérations spécifiques.

2.3. Diagrammes de Séquence

- **Séquence d'Authentification :**

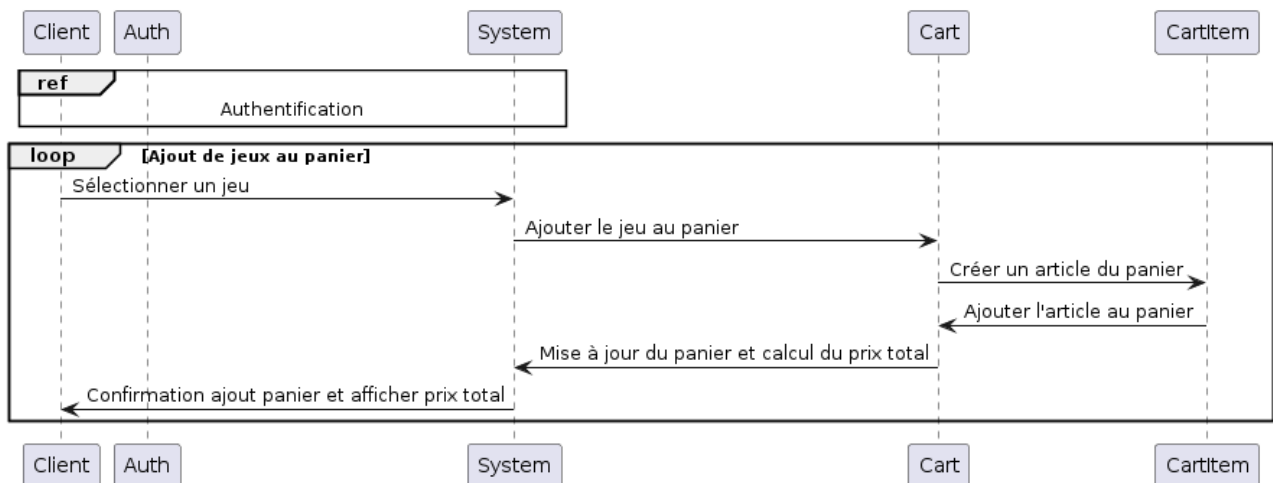


Explication du diagramme :

Ce diagramme montre le processus d'authentification des utilisateurs. Il inclut les étapes suivantes :

- Entrer les informations de connexion
- Vérifier les informations d'identification
- Authentification réussie ou échouée
- Accès à la section admin si l'utilisateur est administrateur

- **Séquence d'Ajout de Jeu au Panier :**

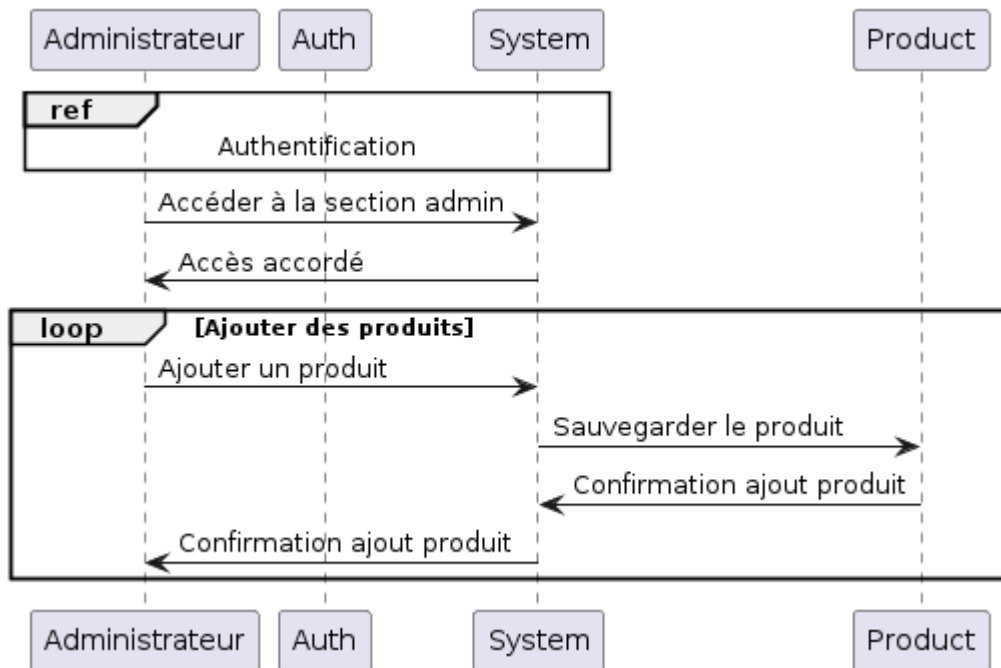


Explication du diagramme :

Ce diagramme montre le processus d'ajout d'un jeu au panier. Il inclut les étapes suivantes :

- Sélectionner un jeu
- Ajouter le jeu au panier
- Créer un article du panier
- Mettre à jour le panier et calculer le prix total
- Confirmation de l'ajout au panier et affichage du prix total

- Séquence d'Ajout de Produit par l'Administrateur :

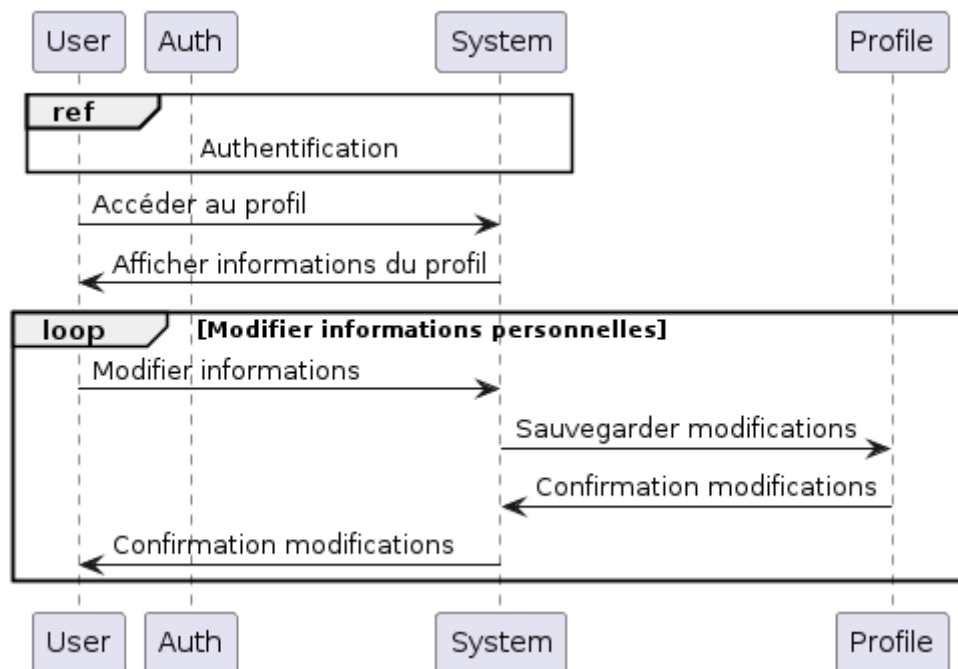


Explication du diagramme :

Ce diagramme montre le processus d'ajout d'un produit par un administrateur. Il inclut les étapes suivantes :

- Accéder à la section admin après authentification
- Ajouter un produit
- Sauvegarder le produit
- Confirmation de l'ajout du produit

- **Séquence de Modification des Informations Personnelles :**



Explication du diagramme :

Ce diagramme montre le processus de modification des informations personnelles d'un utilisateur. Il inclut les étapes suivantes :

- Accéder au profil après authentification
- Modifier les informations
- Sauvegarder les modifications
- Confirmation des modifications

Conclusion

Ce chapitre a détaillé le processus de modélisation et de conception de l'application E-Games Shop. L'analyse des besoins a permis d'identifier les fonctionnalités et les critères de performance essentiels. Les diagrammes UML ont fourni une vue claire de l'architecture et des interactions du système. Dans le chapitre suivant, nous aborderons la mise en œuvre technique, en décrivant les technologies et outils utilisés ainsi que les interfaces graphiques développées.

Chapitre 3 :

Réalisation du Système

Introduction :

Ce chapitre décrit la réalisation technique du système E-Games Shop. Nous commencerons par présenter l'environnement de développement, en couvrant à la fois l'environnement matériel et logiciel. Ensuite, nous examinerons les principales interfaces graphiques développées pour l'application, avant de conclure avec un résumé des réalisations et des défis rencontrés.

1. Outils et technologies utilisés

1.1 Django



Figure 3.1.1 – Logo django

Django (1 (2)) est un Framework web open-source en Python qui permet de développer des applications web rapidement et proprement. Il suit le modèle MVT (Modèle-Vue-Template) et inclut une multitude de fonctionnalités intégrées telles que l'ORM (Object-Relational Mapping), un système d'administration automatique, une gestion des formulaires, et plus encore. Django est connu pour sa sécurité et sa capacité à aider les développeurs à éviter les erreurs de sécurité courantes.

1.2 SQLite



Figure 3.1.3 – Logo SQLite

SQLite (2) est une bibliothèque en langage C qui fournit une base de données SQL légère et intégrée. Elle est particulièrement adaptée pour le développement et les tests en raison de sa simplicité et de sa configuration minimale. SQLite est embarquée dans l'application et ne nécessite pas de serveur de base de données séparé.

1.3 Bootstrap



Figure - Logo Bootstrap

Bootstrap (3) est un Framework CSS open-source développé par Twitter. Il permet de créer des interfaces utilisateur modernes et responsives avec un minimum d'effort. Bootstrap inclut des modèles de conception prêts à l'emploi pour les boutons, les formulaires, les barres de navigation, et bien plus encore.

1.4 Javascript



Figure - Logo JavaScript

JavaScript (4) est un langage de programmation léger, interprété et orienté objet. Il est utilisé pour rendre les pages web interactives et dynamiques. Avec l'évolution de JavaScript et l'ajout de frameworks et de bibliothèques modernes, il est devenu possible de développer des applications web complexes.

1.5 Environnement de développement : Visual Studio Code



Figure - Logo Visual studio Code

Visual Studio Code (5) est un éditeur de code source développé par Microsoft. Il est léger mais puissant, avec des fonctionnalités telles que la complétion de code IntelliSense, le débogage intégré, et le support pour Git. VS Code est hautement extensible grâce à une grande variété d'extensions disponibles.

1.6 Git



Figure - Logo Git

Git (6) est un système de contrôle de version distribué. Il permet aux développeurs de suivre les modifications du code source au fil du temps, de collaborer avec d'autres développeurs et de gérer les versions du code. Git est essentiel pour le travail en équipe et le maintien de l'intégrité du code.

1.7 GitHub



Figure - Logo Github

GitHub (7) est une plateforme de gestion de code source basée sur Git. Elle offre des fonctionnalités supplémentaires telles que les pull requests, les issues, et la gestion des projets. GitHub facilite la collaboration entre les développeurs et le partage de code.

1.8 Jira

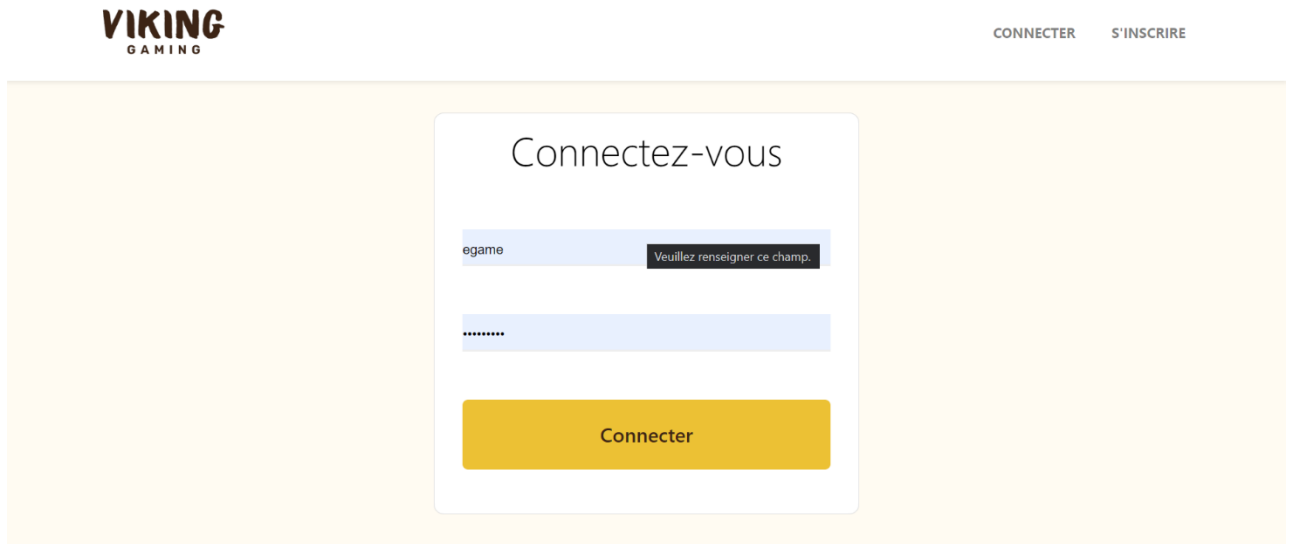


Figure - Logo Jira

Jira (8) est un outil de gestion de projet agile développé par Atlassian. Il est utilisé pour suivre les tâches, planifier les sprints et gérer les problèmes tout au long du développement. Jira permet de maintenir une organisation et une communication claires au sein de l'équipe de développement

2. Interfaces graphiques

— Interface Authentification :



The screenshot displays the Viking Gaming authentication interface. At the top left is the 'VIKING GAMING' logo. At the top right are the links 'CONNECTER' and 'S'INSCRIRE'. The main content area features a white card with the title 'Connectez-vous'. Below the title are two input fields: the first contains the text 'egame' and has a dark grey error message 'Veuillez renseigner ce champ.'; the second is a password field with masked characters '*****'. A yellow 'Connecter' button is positioned at the bottom of the card.

Description :

L'interface d'authentification permet aux utilisateurs de se connecter à leur compte. Elle inclut des champs pour entrer le nom d'utilisateur et le mot de passe. Il y a aussi des liens vers les pages d'inscription et de réinitialisation du mot de passe en cas d'oubli.

— Interface d'inscription:

VIKING
GAMING

CONNECTER S'INSCRIRE

Create your account

Nom d'utilisateur

Email

Mot de pass

Reécrire le mot de pass

Register Now

Description :


L'interface d'inscription permet aux nouveaux utilisateurs de créer un compte. Elle inclut des champs pour entrer le nom d'utilisateur, l'email, le mot de passe et la confirmation du mot de passe. Un bouton "Register Now" permet de soumettre le formulaire d'inscription.

— Interface de gestion des produits :

VIKING
GAMING

SHOP SERVICES A PROPOS

PANIER COMPTE DÉCONNECTER




FIFA 23

Avec EA SPORTS™ FIFA 23, le sport mondial est à l'honneur avec la FIFA World Cup™ des hommes et des femmes, l'ajout des clubs féminins et de nouvelles manières de profiter de vos modes préférés.

650.00 DH

Add to cart




Cup Head

Cuphead est un jeu d'action 2D où vous incarnez Cuphead ou Mugman. Évoluez dans un univers inspiré par les cartoons des années 1930

200.00 DH

Add to cart




Hollow knight

Hollow Knight est un jeu d'aventure et d'exploration qui vous plonge dans un vaste royaume en ruine. Dans ce jeu, résolument 2D

150.00 DH

Add to cart



Hogwarts Legacy

L'Héritage de Poudlard est un RPG d'action-aventure immersif en monde ouvert qui se déroule dans l'univers des livres Harry Potter.

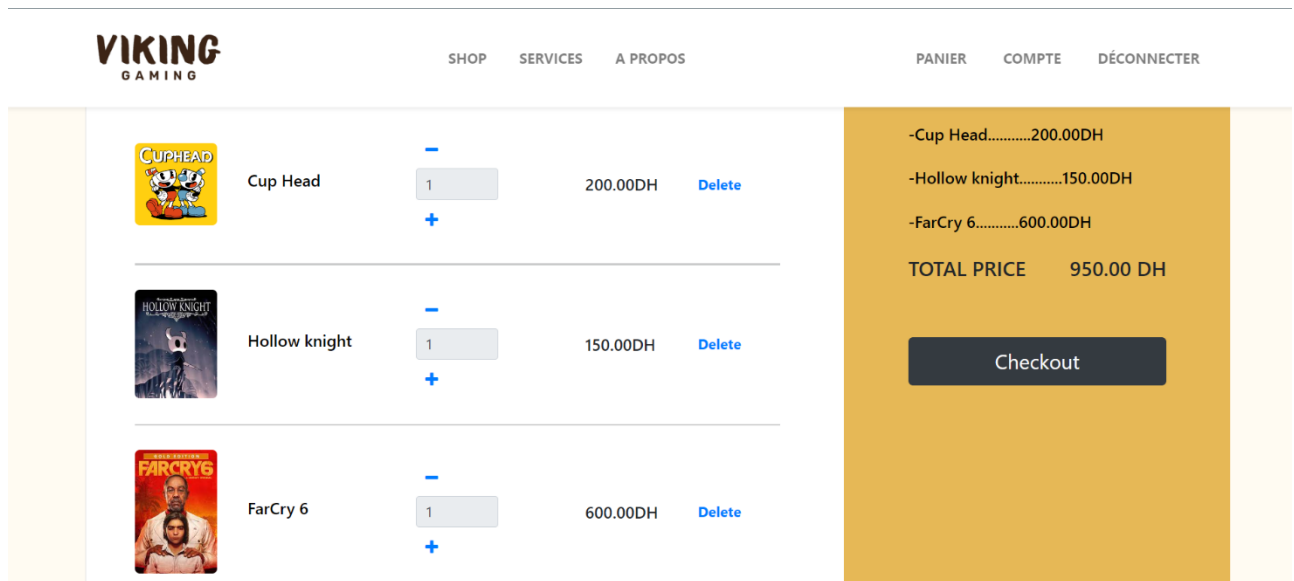
600.00 DH

Add to cart

Description :

Cette interface permet aux utilisateurs de parcourir et de sélectionner les jeux disponibles à l'achat. Chaque jeu est présenté avec une image, une description, et un bouton "Add to cart" pour l'ajouter au panier.

— Interface de panier d'achat:



Description :

L'interface de panier d'achat permet aux utilisateurs de visualiser les jeux ajoutés à leur panier, de modifier les quantités, et de finaliser leur commande. Le total des achats est calculé et affiché en temps réel. Un bouton "Checkout" permet de passer à l'étape de paiement.

— Interface de modification des informations personnelles:

The screenshot shows a web interface for 'VIKING GAMING'. The header includes the logo and navigation links: SHOP, SERVICES, A PROPOS, PANIER, COMPTE, and DÉCONNECTER. The main content area is divided into a left sidebar and a right main section. The sidebar displays a user profile for 'amine' with the email 'yassine@gmail.com' and a 'Choose file' button. The main section is titled 'Personal Details' and contains form fields for 'First Name' (amine), 'Last Name' (Bouazza), 'Email' (yassine@gmail.com), and 'Phone' (Enter phone number). Below this is an 'Address' section with fields for 'Street' (Enter Street), 'City' (Enter City), 'State' (Enter State), and 'Zip Code' (Zip Code). At the bottom of the main section, there are two buttons: 'Supprimer mon Compte' (red) and 'Update' (yellow).

Description :

Cette interface permet aux utilisateurs de consulter et de modifier leurs informations personnelles. Ils peuvent mettre à jour leur nom, email, numéro de téléphone, et adresse. Il y a aussi une option pour supprimer le compte.

Conclusion :

Ce chapitre est consacré à la réalisation du projet. La première partie est consacrée aux outils et technologies utilisés pour la mise en œuvre de la solution. La partie suivante est consacrée à la présentation des interfaces graphiques réalisées.

CONCLUSION GÉNÉRALE

Le projet E-Games Shop a permis de développer une plateforme de commerce électronique dédiée à la vente de jeux vidéo en ligne. Tout au long de ce projet, nous avons mis en œuvre diverses technologies et outils modernes pour créer une application performante, sécurisée et conviviale.

Nous avons commencé par une phase de planification détaillée, comprenant l'analyse des besoins, la modélisation du système, et la conception de l'architecture. L'utilisation de la méthodologie agile Scrum nous a permis de gérer efficacement le développement et d'assurer une livraison itérative et incrémentale des fonctionnalités.

Pour le développement backend, nous avons choisi le framework Django pour sa robustesse, sa sécurité et ses fonctionnalités intégrées. La base de données SQLite a été utilisée pour le développement et les tests en raison de sa simplicité et de sa légèreté. Pour le frontend, nous avons utilisé HTML5, CSS3, JavaScript et Bootstrap pour créer des interfaces utilisateur modernes et responsives.

L'intégration d'outils de développement tels que Visual Studio Code, Git, GitHub et Jira a facilité la gestion du code source, la collaboration entre les membres de l'équipe et la gestion de projet. Ces outils ont été essentiels pour maintenir une organisation et une communication claires tout au long du projet.

Les principales interfaces graphiques développées incluent l'authentification des utilisateurs, la gestion des produits, le panier d'achat, et la modification des informations personnelles. Chaque interface a été conçue pour offrir une expérience utilisateur intuitive et fluide, avec une attention particulière portée à la sécurité et à la performance.

Malgré les défis rencontrés, tels que la sécurité des données, la performance et la scalabilité, nous avons mis en place des solutions efficaces pour les surmonter. L'utilisation de HTTPS, le cryptage des mots de passe et l'optimisation des requêtes à la base de données sont quelques-unes des mesures que nous avons prises pour garantir un fonctionnement sécurisé et rapide de l'application.

En conclusion, le projet E-Games Shop a été un succès, offrant une plateforme de commerce électronique complète et fonctionnelle pour la vente de jeux vidéo. Les résultats obtenus démontrent la faisabilité et l'efficacité des choix technologiques et méthodologiques adoptés. Pour l'avenir, nous envisageons d'améliorer encore la plateforme en ajoutant de nouvelles fonctionnalités, en optimisant davantage la performance et en explorant des opportunités d'intégration avec d'autres services en ligne. Ce projet a non seulement permis de renforcer nos compétences techniques, mais a également démontré l'importance de la collaboration, de la planification et de la gestion de projet dans le développement de solutions technologiques innovantes.

BIBLIOGRAPHIE

- (1) Django Project. (n.d.). Django: The web framework for perfectionists with deadlines. Consulté à l'adresse <https://www.djangoproject.com/>
- (2) SQLite. (n.d.). SQLite Home Page. Consulté à l'adresse <https://www.sqlite.org/index.html>
- (3) Bootstrap. (n.d.). Bootstrap - The most popular HTML, CSS, and JS library in the world. Consulté à l'adresse <https://getbootstrap.com/>
- (4) Mozilla Developer Network. (n.d.). JavaScript. Consulté à l'adresse <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- (5) Visual Studio Code. (n.d.). Visual Studio Code - Code editing. Redefined. Consulté à l'adresse <https://code.visualstudio.com/>
- (6) Git. (n.d.). Git - Distributed version control system. Consulté à l'adresse <https://git-scm.com/>
- (7) GitHub. (n.d.). GitHub: Where the world builds software. Consulté à l'adresse <https://github.com/>
- (8) Atlassian. (n.d.). Jira | Issue & Project Tracking Software. Consulté à l'adresse <https://www.atlassian.com/software/jira>