

Death Runners



Jeu de Survie dans le Dark Web

- Cahier des Charges -

Date : 16 Février 2025

Auteur : - BAZAZ ABKENAR Elina
- WENINGER MOZOMBITE Jonathan

Enseignant : SUCAL Virgile

Sommaire

1.Introduction

2.Objectifs du projet

3.Présentation du concept et fonctionnalités

4.Architecture du site et technologies

5.Modélisation des données

a.5.1 Modèle UML des classes

b.5.2 Schéma de la base de données

6.Déroulement du projet et phases de développement

7.Conclusion

1. Introduction

1.1 Contexte

Death Runners est un jeu de survie immersif inspiré de notre imagination sur l'univers du Dark Web. Le joueur est plongé dans une course contre la montre où chaque décision compte. Un mystérieux parieur anonyme l'a sélectionné pour un jeu mortel : résoudre des énigmes complexes en un temps limité pour survivre. L'ambiance du jeu mêle **hacking, pannes simulées et interactions avec une IA suspecte.**

1.2 Problématique et Enjeux

L'objectif est de créer une **expérience interactive et immersive**, en combinant **narration évolutive, énigmes et prise de décision morale.**

Ce projet doit répondre aux enjeux suivants :

- Développer un jeu web innovant avec une mécanique de **choix influençant le scénario.**
- Intégrer des **effets de piratage simulés** (faux glitches, messages cryptés, etc.).
- Assurer une **expérience utilisateur fluide et engageante.**

2. Objectifs du projet

2.1 Objectifs Généraux

- Concevoir un jeu web interactif avec une **narration dynamique**.
- Développer une **interface immersive** qui simule un environnement hacké.
- Proposer une **intelligence artificielle** capable d'envoyer des messages mystérieux.

2.2 Objectifs Techniques

- Implémenter un **chatbot IA** simulant un hacker (si possible).
- Créer une **base de données structurée** pour la progression des joueurs.
- Développer des **effets de pannes et piratage simulé** via JavaScript.
- Mettre en place une **interface sécurisée** pour l'administration.

3. Présentation du Concept et Fonctionnalités

3.1 Fonctionnalités Principales

◇ Joueur

- Inscription / Connexion sécurisée.
- Interaction avec un **chatbot IA** qui envoie des indices cryptiques.
- Résolution d'**énigmes** pour avancer dans le jeu.
- Prise de **décisions morales** influençant la progression.

◇ Administrateur

- Gestion des **joueurs et de leur progression**.
- Ajout / suppression / modification des **énigmes**.
- **Surveillance des activités** des joueurs en temps réel.

◇ Expérience immersive

- **Simulation de piratage** (glitches visuels, faux crashes...).
- **Fichiers corrompus** à restaurer pour obtenir des indices.
- **Bruitages et animations** renforçant le stress du joueur.

4. Arborescence et Structure du Site

4.1 Structure du Site

```

└─ monsite_exemple/
  └─ admin/ (Interface administrateur – gestion du jeu et des joueurs)
    └─ control/ (Gestion des utilisateurs et des énigmes)
      └─ admin_control.php (Gestion des comptes admins, bans...)
      └─ enigme_control.php (Ajout/Suppression/Modification des énigmes)
    └─ model/ (Gestion de la base de données admin)
      └─ admin_db.php (Connexion et requêtes BDD admin)
      └─ AdminDB.php (Class pour obtenir un PDO coter admin)
      └─ EnigmeManager.php (Class-DAO, la manipulation des enigmes en
admin)
      └─ UserManager.php (Class-DAO, la manipulation des users en admin)
    └─ templates/back/ (Interface admin)
      └─ dashboard.php (Vue de gestion des joueurs)
      └─ footer.php (Bas de page du BackOffice)
      └─ gestion_enigmes.php (Gestion des énigmes)
      └─ header.php (Haut de page/ menu du BackOffice)
      └─ main.php (???)
      └─ users.php (Gérer les utilisateurs en admin)
    └─ index.php (Page d'accueil du BackOffice)
  └─ class/
    └─ Enigme.php (Class entité représentant une énigme)
    └─ User.php (Class entité représentant un user)
  └─ control/ (Gestion des énigmes, chatbot, progression du jeu)
```

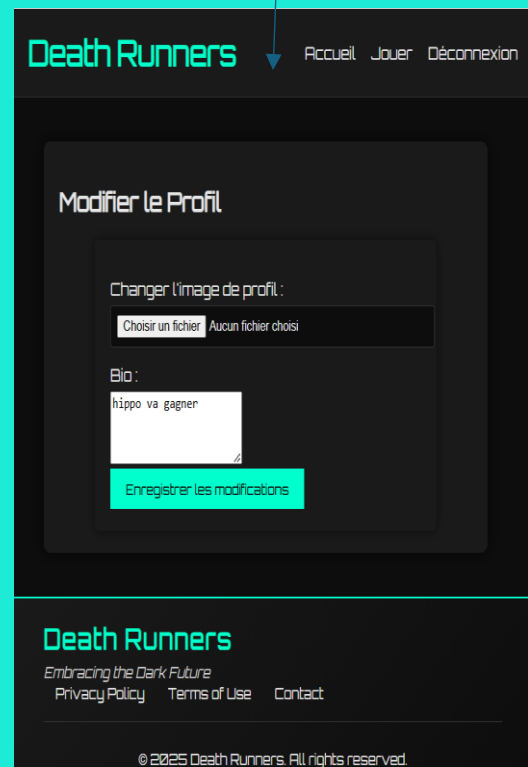
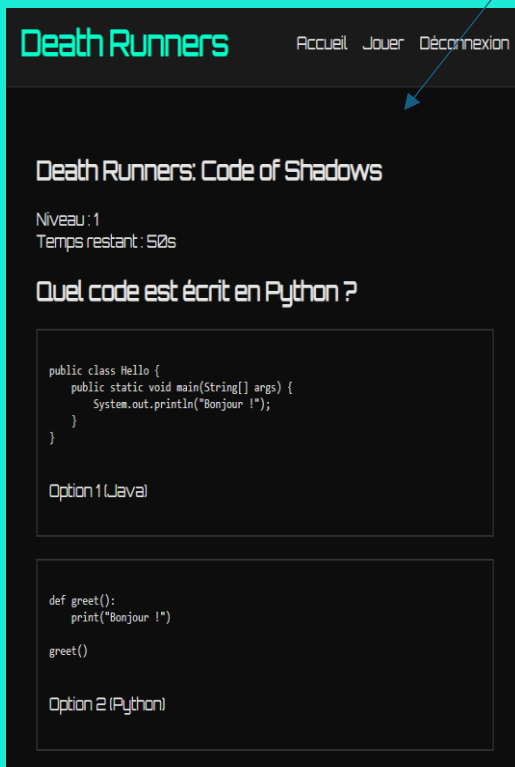
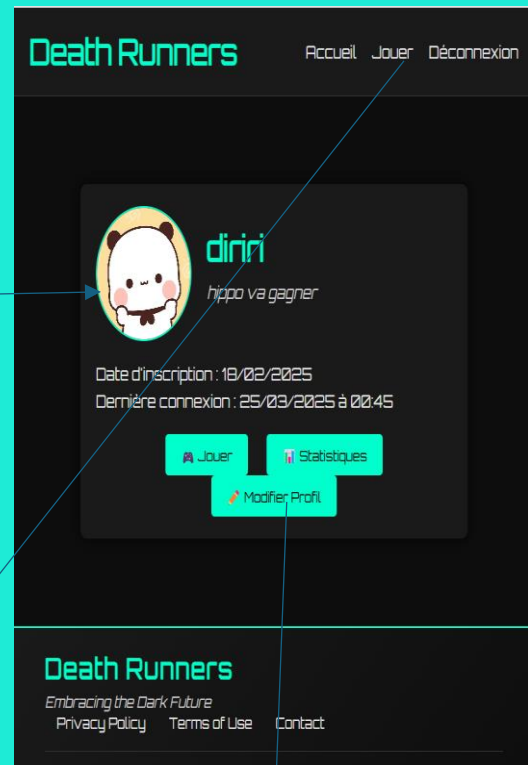
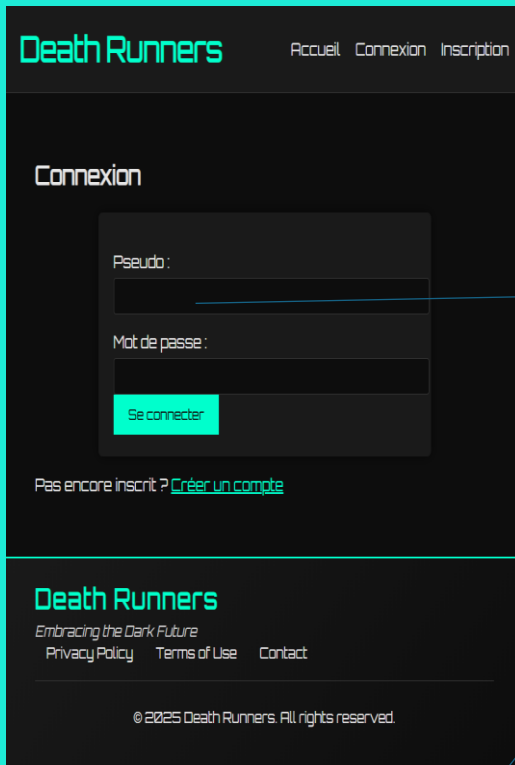
- | — 📄 **game_control.php** (Gère la progression du joueur)
- | — 📄 **chatbot_control.php** (Génère les messages et réponses de l'IA)
- | — 📄 **enigme_control.php** (Gère la logique liées aux énigmes côté front)
- | — 📄 **glitch_control.php** (Effets de piratage simulés sur le site)
- | — 📄 **logout.php** (Déconnexion du joueur)

- | — 📁 **model/** (Gestion de la base de données des joueurs et du jeu)
 - | — 📄 **db.php** (Connexion globale à la base de données)
 - | — 📄 **enigme_model.php** (Requêtes SQL pour les énigmes)
 - | — 📄 **UserManager.php** (Class DAO pour manipuler les users côté Front)

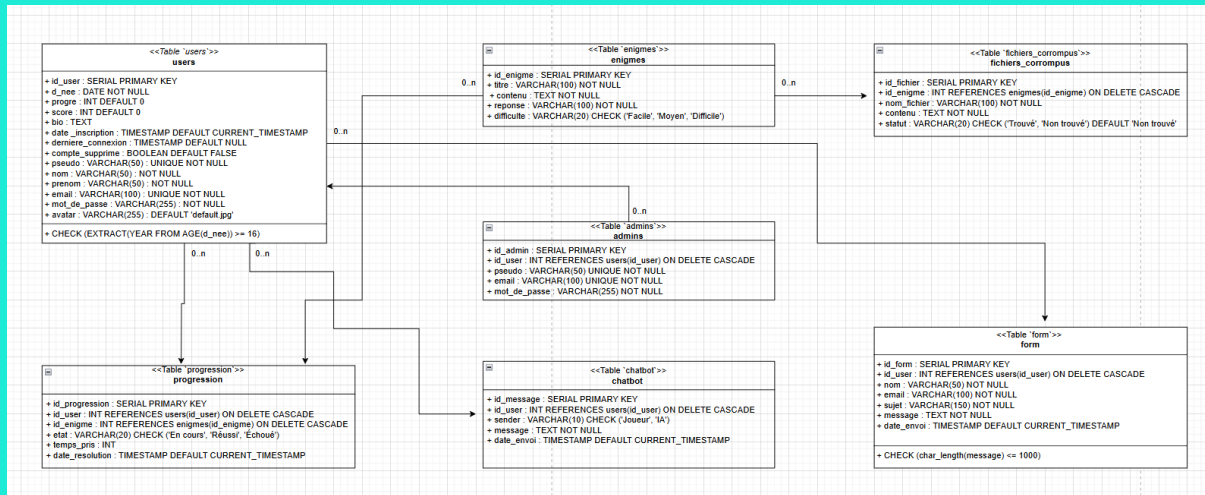
- | — 📁 **templates/front/** (Interface utilisateur - Pages HTML/CSS du joueur)
 - | — 📄 **header.php** (Navigation globale)
 - | — 📄 **footer.php** (Pied de page)
 - | — 📄 **process_contact** (Traite l'envoi du formulaire de contact)
 - | — 📄 **game.php** (Interface principale du jeu)
 - | — 📄 **login.php** (Connexion du joueur)
 - | — 📄 **register.php** (Inscription du joueur)
 - | — 📄 **edit_profile.php** (Modification du profil)
 - | — 📄 **form.php** (Formulaire de contact)
 - | — 📄 **stats.php** (Affiche les statistiques du joueur)
 - | — 📄 **home.php** (Page d'accueil du profil de l'user)
 - | — 📄 **main.php** (Gabarit qui inject la variable \$main)

- | — 📄 **index.php** (Page d'accueil du jeu, introduction à l'histoire)

4.2 Schéma du Site



4.3 UML



4.4 Langage Utilisées

Backend : PHP, MySQL

Frontend : HTML, CSS, JavaScript

Effets dynamiques : JavaScript (animations, IA Chatbot)

5. Modélisation des Données

5.1 Modèle UML des Classes

Classes principales : Joueur, Admin, Chatbot, Énigme, Progression.

5.2 Base de Données

Table users (*Joueurs*)

id_user	pseudo	nom	prenom	email	mdp	d_nee	progre	score
1	ElinaBZ	BAZZAZ	Elina	ElinaBZ@gmail.com	hashé	30/10/03	2	500

Table enigmes (*Défis*)

id_enigme	titre	contenu	reponse	difficuler
1	Corruption système	Trouve le bon checksum	45A7C3	Difficile

Table progression (*Suivi des joueurs*)

id_progre	id_user	id_enigme	etat	temps
1	1	3	réussi	5 min

Table chat_messages (*Stocke les messages du chatbot et du joueur*)

id_c hat	id_user	sender	message	timestamp
1	1	chatbot	"Regarde dans les logs système..."	12:30:15

6. Déroulement du Projet et Phases de Développement

Phase 1 : Initialisation

- Définition des classes et architecture de la base de données.
- Développement du système **d'inscription et connexion**.

Phase 2 : Développement des Fonctionnalités

- Implémentation du **chatbot IA** et du système d'énigmes.
- Développement des **effets de piratage** et immersion.

Phase 3 : Tests et Optimisation

- Correction des bugs et optimisation des performances.
- Ajout de nouveaux scénarios et améliorations UX.

7. Conclusion

Le projet **Death Runners** vise à offrir une **expérience immersive unique**, en combinant **énigmes, interactions IA et simulations de piratage**. Ce jeu est un mélange entre **réflexion, tension et innovation**. On l'a conçu pour plonger le ou les joueur(s) dans un univers inquiétant et captivant où chaque décision compte.

Bonne partie !