



Grimoires Guardians

Nom de l'Étudiant: Lehenerz Geoffrey

Date: 23/04/2024

Dernière modification: 10/06/2024

Cours/Institution: Projet Web, EAFC Fleron Charlemagne

- Grimoires Guardians
 - 1. Définition des Parties
 - 2. Contexte du Projet
 - Objectifs Principaux
 - Objectifs Secondaires
 - 3. Présentation de l'Application
 - Accessibilité
 - Fonctionnalités
 - Pour la première phase (projet) :
 - En conclusion :
 - Pour la seconde phase (TFE) :
 - 4. Technologies Utilisées
 - Front-end
 - Back-end
 - Hébergement
 - Base de Données
 - 5. Déploiement et Maintenance
 - 6. Phases de Développement
 - Phase 1 : Conception et Planification
 - Phase 2 : Développement et Implémentation
 - 7. Prix
 - 8. Mentions Légales

1. Définition des Parties

Client :

Le client pour ce projet est l'université qui supervise le travail de fin d'études. Le représentant désigné pour ce projet est le professeur chargé du cours de développement web, qui évaluera les aspects techniques et la réalisation du projet.

Développeur :

Lehenerz Geoffrey, étudiant en développement web, responsable de la conception, du développement, de l'implémentation, et de la maintenance de l'application web "Grimoires Guardians". Ce projet constitue une partie significative de son évaluation finale.

2. Contexte du Projet

"Grimoires Guardians" est un projet ambitieux visant à créer un site de jeu de rôle interactif qui permettra aux utilisateurs de s'immerger dans des univers fantastiques à travers des jeux de rôle en ligne. Ce projet émerge de la passion pour les jeux de rôle et de l'observation d'un besoin de plateformes intégrées permettant une gestion fluide et complète des éléments clés de ces jeux. Notre but est de développer une plateforme qui non seulement héberge des sessions de jeu captivantes mais facilite également la gestion des campagnes par les maîtres de jeu (MJ) et l'engagement des joueurs.

Le projet utilisera des technologies modernes telles qu'AdonisJS pour le backend et React pour le frontend, choisies pour leur robustesse, leur flexibilité, et leur large communauté de support. Ces technologies permettront de mettre en place une architecture solide, adaptative et capable de supporter des extensions futures, comme l'ajout de nouveaux systèmes de jeux.

Objectifs Principaux

1. **Création d'une Plateforme Multi-utilisateurs** : Permettre à plusieurs groupes de joueurs de créer, gérer et participer à des jeux de rôle. Les utilisateurs pourront s'inscrire en tant que MJ ou joueurs, avec des niveaux d'accès et des fonctionnalités spécifiques à chaque rôle.
2. **Gestion Intégrée des Parties** : Offrir des outils pour la création et la maintenance de fiches de personnage, de jets de dés automatisés, et de gestion des éléments de jeu tels que les cartes, les sortilèges, les objets et les créatures.
3. **Interface Interactive et Immersive** : Développer une interface utilisateur conviviale et esthétiquement agréable qui engage les utilisateurs et enrichit leur expérience de jeu.
4. **Tchat en Direct pour les Joueurs** : Implémenter un système de tchat en direct où les joueurs peuvent non seulement effectuer des jets de dés visibles par tous mais aussi écrire et communiquer librement pendant le jeu, renforçant ainsi l'immersion et l'interaction.

Objectifs Secondaires

1. **Scalabilité** : Construire une base solide qui permette l'intégration de nouveaux jeux et systèmes à l'avenir sans nécessiter de refonte majeure.

3. Présentation de l'Application

Accessibilité

L'application sera accessible via un navigateur web, sans nécessité de télécharger ou installer de logiciel additionnel. Optimisée pour les ordinateurs de bureau, elle garantira une accessibilité maximale pour tous les utilisateurs intéressés par les jeux de rôle.

Fonctionnalités

Pour la première phase (projet) :

- **Connexion** : Les utilisateurs arriveront sur une page de connexion avec possibilité de création de compte.
- **Page d'accueil interactive** : Après connexion, l'utilisateur arrivera sur une page de bienvenue, munie d'une barre de navigation permettant de créer une partie, de rejoindre une partie, et d'accéder à la gestion de son compte, de ses parties, et à la déconnexion du site.
- **Page de création de partie** : Un simple formulaire permettant de donner un nom (nom de la campagne par exemple) et une description à la partie. Une fois la partie créée, l'utilisateur sera redirigé vers la page pour rejoindre une partie.
- **Page rejoindre une partie** : Une page permettant de voir toutes les parties où l'utilisateur est inscrit, que ce soit ses propres tables sous l'indication "Parties en tant que Maître de Jeu", ou en tant que joueur sous "Parties en tant que Joueur".
- **Le menu dropdown** :
 - **L'onglet gérer le compte** : Permet à l'utilisateur de modifier son pseudo, son adresse mail, ainsi que son mot de passe.
 - **L'onglet gérer les parties** : Permet à l'utilisateur de modifier ou supprimer ses parties créées.
 - En cliquant sur modifier une partie, un élément apparaîtra dans le DOM où l'utilisateur pourra modifier le nom et la description de la partie.
 - **Le bouton de déconnexion** : Permet de se déconnecter du site.
- **L'écran d'une partie** : Contient un écran principal où est affichée la carte ainsi qu'un composant aside sur le côté droit de l'écran. Celui-ci est composé de plusieurs onglets : chat, personnages, objets, notes, gestion.
 - Le chat permet l'affichage de messages ainsi que de jets de dés. Pour effectuer un jet de dés depuis le chat, il faut commencer le message par un /r puis préciser le nombre et la valeur des dés que l'on veut jeter. Par exemple : /r 1d20 jettera 1 dé à 20 faces tandis que /r 4d6 jettera 4 dés à 6 faces. Il existe d'autres jets que l'on dit à avantage ou à désavantage. Dans ce cas-là, on jette 2 dés et garde soit le meilleur ou le pire des 2 dés jetés. Par exemple, /r 1d20a pour un jet à avantage jettera 2 dés mais ne gardera que le plus haut score, tandis que /r 1d20d gardera le plus petit score.
 - L'onglet personnage permet la création de fiches de personnage, ainsi que la création de dossiers pour permettre de classer les fiches et ainsi améliorer l'expérience utilisateur. Tout sera classé via de simples glisser-déposer (actuellement, les limites techniques ne permettent qu'une profondeur de dossier).
 - L'onglet Objets permet la création de nombreux types d'objets différents : Arme, Armure, Accessoire, Autre, Peuple, Profil, Voie, Capacité. Le même système de gestion de fichiers/dossiers est identique à celui des personnages.
 - L'onglet Notes permet aux joueurs ou MJ de prendre facilement des notes sur ce qui se passe dans la partie. Elles peuvent être publiques ou privées selon le bon vouloir de leur créateur, sachant que le MJ aura toujours accès à tout.

- L'onglet gestion des parties permet l'ajout de joueurs dans la partie via l'onglet inviter des membres ou de simplement quitter la partie, ce qui ramène sur la page d'accueil du site.
- En cas de vidéo utilisée comme carte, il existe un raccourci clavier permettant d'activer ou de désactiver le son en faisant la combinaison de touches ALT+S.

En conclusion :

La première phase du projet permet de placer le squelette de l'application. Tout ce qui est gestion d'images, de carte et également de toute la partie multi-joueurs n'est pas encore fonctionnel. Mais elle permet déjà de voir beaucoup du fonctionnement de celle-ci.

Pour la seconde phase (TFE) :

- Système de carte où l'on pourra déplacer nos tokens (symbolisant les créatures et différents personnages) pour imager les phases de combat.
- Mise à jour du chat de façon automatisée.
- Gestion des accès aux fichiers (pour éviter aux joueurs d'avoir accès aux fiches des créatures par exemple, ou aux notes privées de chacun).
- Ajout de la gestion des images que ce soit pour les personnages, les objets et autres.
- Amélioration des notes pour pouvoir faire des notes ne contenant que des images.

4. Technologies Utilisées

Front-end

- **React** : Pour construire une interface utilisateur dynamique avec des composants réactifs.
- **React Router** : Pour gérer la navigation dans l'application.
- **Axios** : Pour faire des requêtes HTTP au serveur back-end, facilitant l'intégration avec AdonisJS.
- **Tailwind CSS** : Utilisé pour le stylisme des composants avec une approche utilitaire.
- **Material-UI ou Bootstrap React** : Pour utiliser des composants UI prêts à l'emploi, accélérant ainsi le développement de l'interface.
- **Socket.IO Client** : Pour la communication en temps réel, nécessaire pour le chat et la mise à jour des positions des tokens.

Back-end

- **AdonisJS** : Framework backend robuste qui fournit une architecture MVC, facilitant l'organisation et la maintenance du code.
- **Lucid ORM** : Inclus dans AdonisJS, permettant de manipuler facilement les données relationnelles.
- **AdonisJS Auth** : Pour une implémentation facile de l'authentification, incluant les sessions et les tokens.
- **Edge Template Engine** : Utilisé pour générer des vues côté serveur, si nécessaire.
- **Socket.IO** : Intégré pour la gestion des communications en temps réel avec les clients.
- **Envoy** : Outil de AdonisJS utilisé pour la gestion des tâches d'administration système et de déploiement.

Hébergement

- **OVH** : Hébergement de fichiers statiques pour le frontend.

- **Heroku** : Plateforme de cloud computing utilisée pour héberger l'application backend, ainsi que la base de données en PostgreSQL.

Base de Données

- **PostgreSQL** : Base de données relationnelle utilisée pour stocker toutes les données structurées, telles que les informations des utilisateurs, les fiches de personnage, les détails des parties, etc.
- **Migrations et Seeds** : Utilisez les outils de migration d'AdonisJS pour définir et maintenir la structure de la base de données, facilitant les mises à jour et les déploiements.

5. Déploiement et Maintenance

- **GitHub** : Il a été décidé d'utiliser GitHub pour le déploiement ainsi que la maintenance de l'application.

6. Phases de Développement

Phase 1 : Conception et Planification

1. Définition des exigences :

- Identification des besoins des utilisateurs (MJ et joueurs).
- Analyse des fonctionnalités essentielles et secondaires.

2. Étude de faisabilité :

- Évaluation des technologies à utiliser (AdonisJS, React, PostgreSQL, etc.).
- Analyse des risques et planification des solutions.

3. Wireframes et maquettes :

- Création des wireframes pour l'interface utilisateur.
- Développement de prototypes et maquettes graphiques.

4. Planification du projet :

- Création d'un calendrier de développement.
- Répartition des tâches et définition des jalons.

Phase 2 : Développement et Implémentation

1. Développement de l'Architecture Backend :

- Mise en place de l'environnement de développement.
- Création des schémas de base de données.
- Développement des API et implémentation de l'authentification.
- Intégration de Socket.IO pour la communication en temps réel.

2. Développement de l'Interface Frontend :

- Configuration de l'environnement frontend.
- Création des composants UI et intégration des composants Material-UI ou Bootstrap React.
- Implémentation de Redux ou Context API pour la gestion de l'état de l'application.

- Connexion des composants frontend aux API backend via Axios.

3. Développement des Fonctionnalités Utilisateurs :

- Création et gestion des fiches de personnage pour les joueurs.
- Développement du chat et des jets de dés.
- Création et gestion des fiches de PNJ et de créatures pour les MJ.
- Outils de gestion de campagne pour les MJ.

4. Tests et Assurance Qualité :

- Écriture et exécution de tests unitaires et d'intégration.
- Réalisation de tests utilisateurs et ajustements basés sur les retours.

5. Déploiement :

- Préparation pour le déploiement et configuration de l'environnement de production.
- Déploiement initial sur le serveur de production et vérification du bon fonctionnement.

6. Maintenance et Support :

- Surveillance des performances de l'application et correction des bugs.
- Mise en place d'un système de support pour les utilisateurs et documentation.

7. Prix

- Les services proposés par l'application sont fournis gracieusement. La main-d'œuvre et le coût de l'investissement moral du prestataire ne sont pas facturés au client.

8. Mentions Légales

- **Droits et responsabilités** : Tous les utilisateurs auront accès aux fonctionnalités de base sans restriction, mais seront soumis aux termes d'utilisation qui incluent la non-responsabilité en cas de perte de données ou d'interruption de service.