

## 1. Présentation du Projet

Le site Gameverse consiste en la création d'une plateforme dédiée aux jeux vidéo, permettant aux utilisateurs beaucoup de possibilités comme : rechercher, découvrir et organiser des jeux en fonction de différents critères comme le genre, la plateforme, les thèmes, etc... Le site est récupérable sur GitLab et propose une recherche dynamique avec filtres et recherche textuelle. A chaque fois qu'on sélectionne un jeu on obtient les informations détaillées sur chaque jeu. Il permet également de créer un compte qui peut servir à gérer une liste de jeux favoris.

Les technologies utilisées incluent React et CSS pour le frontend, Express js, pour le backend (MongoDB pour la gestion des données).

Utilisateurs cibles : Les utilisateurs visés sont principalement les joueurs passionnés de jeux vidéo qui souhaitent découvrir de nouveaux titres en fonction de leurs préférences (genre, plateforme, développeur, etc.).

Contexte d'utilisation : Le projet s'adresse aux utilisateurs qui cherchent une plateforme facile d'accès pour trouver des jeux vidéo facilement, et filtrer selon des critères spécifiques pour faciliter cela. L'objectif est d'offrir une recherche fluide et rapide.

## 2. Analyse Technique

### Qualité du Code

Le code du projet est structuré, avec une séparation claire entre le frontend et le backend comme le montre la structure du fichier dans le Readme. Cependant, on pourrait souligner quelques points d'amélioration :

Le code est bien rédigé, mais peut être que certains fichiers pourraient bénéficier de commentaires plus détaillés pour expliquer des parties complexes, comme les scripts d'importation des données ou autre.

**Modularité** : La séparation entre frontend et backend est bien définie. Cependant certaines fonctions comme celles pour la gestion des filtres dans la recherche, peuvent être découpées davantage pour faciliter la maintenance si ça doit être repris par d'autres personnes comme dans le cadre de la SAE BUT2 de l'an dernier.

### Performance

Les performances du projet sont très bonnes car le temps de réponse est très courts pour les recherches, entre 20 et 60 millisecondes en moyenne, à chaque recherche on nous renvoie le temps de réponse (voir img)

```
[1] 🕒 Temps de recherche MongoDB : 20 ms
[1] 🕒 Temps de traitement côté serveur : 1 ms
[1] 🕒 Temps total de la requête : 21 ms
[1] 🕒 Temps de recherche MongoDB : 18 ms
[1] 🕒 Temps de traitement côté serveur : 1 ms
[1] 🕒 Temps total de la requête : 19 ms
[1] 🕒 Temps de recherche MongoDB : 55 ms
[1] 🕒 Temps de traitement côté serveur : 1 ms
[1] 🕒 Temps total de la requête : 56 ms
```

L'algorithme de recherche repose sur l'utilisation de MongoDB et de l'opérateur \$regex pour les correspondances textuelles, ce qui permet des recherches efficaces.

### Sécurité

Le projet semble avoir des mesures de sécurité mises en place, étant donné qu'on peut voir qu'il y a une gestion des comptes utilisateurs.

## **3. Évaluation Fonctionnelle**

### Objectif Initial

Le projet remplit son objectif, nous avons un sélecteur de recherche multimodal de jeux vidéo avec des filtres dynamiques, une recherche textuelle. Les utilisateurs peuvent facilement naviguer à travers les différentes catégories de jeux et en fonction de leurs préférences.

### Facilité d'Utilisation et Ergonomie

L'interface utilisateur est claire et bien conçue. Je trouve que l'utilisation des couleurs sombres, contribuent à une expérience agréable.

Les filtres sont intuitifs et les utilisateurs peuvent rapidement affiner leurs recherches même si il y en a beaucoup.

### Bugs et Dysfonctionnements

Dans l'utilisation du site en lui-même je ne crois pas avoir rencontré de bugs.

## **4. Documentation et Maintenabilité**

### Code Commenté

Le code est bien commenté dans l'ensemble, mais comme dit avant certains blocs de code comme ceux à la gestion des données ou à l'importation dans MongoDB, pourraient être expliqués plus en détail pour faciliter la compréhension.

## Documentation Fournie

Il ya bien la présence d'un fichier README dans GitLab qui dans l'ensemble fonctionne bien, mais elle pourrait être améliorée, il s'agit peut être d'un cas rare mais personnellement j'utilise Docker pour faire fonctionner mongoDB ce qui fait que certaine commande sont nécessaire pour faire fonctionner le site. J'avais du faire :

```
docker run -d --name mongo -p 27017:27017 mongo
```

et non pas un simple docker start mongoddb comme je fais d'habitude.

Donc peut être prévoir des précision pour ce genre de personnes peuvent être intéressant

## Facilité d'Évolution

L'architecture permet d'ajouter facilement de nouvelles fonctionnalités, comme l'ajout de nouveaux filtres ou de nouveaux critères de recherche. La gestion des API REST et la séparation des rôles entre frontend et backend facilitent la maintenance.

## **5. Recommandations et Conclusions**

### Points Forts

Bonne Architecture et séparation bien définie entre frontend et backend pour permettre à d'autres personnes de s'y retrouver facilement

Bonnes performances, avec des temps de réponse rapides et des recherches fluides.

Expérience utilisateur agréable, avec un design épuré et une interface intuitive.

### Points Faibles

Documentation d'installation insuffisante, en particulier pour Docker, et manque de détails dans certaines parties du code.

### Propositions d'Amélioration

Si ce n'est pas déjà fait, améliorer la sécurité : protéger les mots de passe en les cryptant.

Documenter l'installation : Détaille les étapes d'installation et les commandes spécifiques à Docker ou autre dans le fichier README

En conclusion, "Gameverse" est un projet qui remplit ses objectifs et offre une expérience utilisateur fluide et agréable.