

Ergonomie & développement d'une SPA animée



Auteur 1	Vanya CAYLANADIA
Auteur 2	Melda GÖNEN
Auteur 3	Nour REKAYA
Auteur 4	Youssef OUFTALLAH
Auteur 5	Imane TEMSAMANI
Date	17.12.2023
Référence	WEB2-2023-PROJECT-GROUP-12
Version	1.0

Contents



1	Consignes et évaluations.....	3
1.1	Consignes générales	3
1.1.1	Création des groupes sur le site du cours	3
1.1.2	Création d'un groupe sur GitHub Classroom et du web repo associé	4
1.1.3	Projet.....	4
1.2	JavaScript & Node.js : consignes techniques, timing et évaluations.....	6
1.3	Ergonomie : consignes techniques, timing et évaluations.....	11
2	Objectif du projet.....	12
3	Mind map du projet	12
4	Persona	12
5	Axiomes de Morville.....	12
6	Planning des tâches et cas d'utilisation	13
7	Besoins techniques.....	13
7.1	Système.....	13
7.2	Frontend	13
7.3	API.....	14
8	Choix technologiques.....	14
8.1	Frontend	15
8.2	RESTful API.....	15
8.3	Wireframe	15
9	Conception & Implémentation.....	15
9.1	Code repositories.....	15
9.2	Secrets éventuels pour vos API ou base de données.....	15
9.3	Documentation de votre API	15
9.4	Déploiement de vos applications	16
9.5	Code réutilisé.....	16
10	Analyse des résultats par le groupe	17
10.1	Évaluation du résultat par rapport au planning des tâches et des cas d'utilisation	17
10.2	Audit ergonomique de votre projet.....	17

10.3	Difficultés techniques rencontrées.....	18
10.4	Conseils pour appliquer cette technologie	18
10.5	Quels sont les points positifs à la manière dont s'est déroulée la collaboration au sein du groupe ?	19
10.6	Quels sont les points qui seraient à améliorer pour de futures collaborations ?	19
11	Présentation vidéo	19

1 Consignes et évaluations


1.1 Consignes générales

1.1.1 Création des groupes sur le site du cours

Veuillez former un groupe de 5 étudiants sur le site associé au cours : <https://e-vinci.github.io/web2>. Pour ce faire, veuillez-vous authentifier en cliquant sur l'icône . Rendez-vous sur l'onglet **Projets** (<https://e-vinci.github.io/web2/project-page>). Il est recommandé que l'attribution des **groupes** se fasse par **discussions** entre les **étudiants**. Lorsque 5 étudiants ont un **intérêt commun** pour un **projet**, ils s'inscrivent au sein d'un groupe en cliquant sur l'icône .

Pour aider à la création de groupes, il est aussi possible de vous inscrire :

- à un **groupe vide**. Cela permettra à tous d'identifier les partenaires potentiels.
- à un **groupe où il y a déjà un ou plusieurs étudiants**. Dans ce cas, veuillez-vous entretenir avec ces potentiels partenaires sur le **sujet de votre projet**.

Si nécessaire, vous pouvez vous désinscrire d'un groupe où vous n'avez pas trouvé de sujet commun dans le but de rejoindre un autre groupe. Il suffit de cliquer sur l'icône .

A la date ultime de création de groupe (**Séance 9 du cours de JS, 16/10 ou 17/10 selon la série**), pour les étudiants toujours en recherche de partenaires, nous faciliterons (ou imposerons si nécessaire) la création des groupes, mais pas des sujets de projet.

Une fois tous les groupes de 5 étudiants remplis, il restera maximum 4 étudiants non liés à un projet. Si nécessaire un ou plusieurs groupes de 4 étudiants seront créés.

1.1.2 Création d'un groupe sur GitHub Classroom et du web repo associé

Pour chaque groupe de projet, vous allez hériter d'un web repository contenant un boilerplate via GitHub classroom.

Veillez passer à cette étape qu'une fois votre groupe déjà finalisé sur le site du cours.

1.1.2.1 Création de l'équipe associée à un projet

Veillez identifier le membre qui créera votre équipe sur GitHub.

Ce membre accédera à l'assignement via : <https://classroom.github.com/a/zJz7A4kY>

Ce membre devra créer une équipe reprenant le numéro de projet donné sur <https://e-vinci.github.io/web2/project-page> : si le nom de projet indiqué est **Projet N°4** : ... , il créera une équipe portant le nom **group-04** puis cliquera sur **Create team**.

Ce membre devra encore cliquer par la suite sur **Accept this assignment**.

Un web repository aura été créé pour votre équipe.

1.1.2.2 Joindre une équipe existante

Une fois l'équipe d'un projet créée, les autres membres accèderont aussi à l'assignement via : <https://classroom.github.com/a/zJz7A4kY>.

Ces membres joindront l'équipe existante en cliquant sur **Join** au sein de la bonne équipe. Par exemple, pour les membres du **Projet N°4**, ils cliqueront sur **Join** dans l'équipe **group-04**.

Si vous le souhaitez, vous pouvez visualiser cette vidéo qui montre [comment Joindre un GitHub Classroom Group Assignment](#).

1.1.3 Projet

Vous allez créer une SPA mettant en œuvre :

- Des sujets et technologies qui vous tiennent à cœur ;
- Une RESTful API tournant sous Node.js & Express ;
- Un frontend animé ;
- Un frontend consommant votre RESTful API et éventuellement des APIs tierces ;
- Au moins une librairie JS non vue en cours pour le frontend (anime.js ou phaser.io sont autorisées) ainsi qu'une librairie non vue pour l'API.

Pour votre frontend animé, l'animation peut être 2D, 3D, sous forme de jeux ou de simples effets visuels...

Au niveau de la présentation de votre projet, veillez à :

- Prendre en compte l'expérience utilisateur dès le début
- Optimiser le choix de vos technologies en fonction de l'expérience utilisateur
- Appliquez un maximum de théorème psycho-marketing
- Respectez les règles de Usability et auditez votre projet
- Respectez le GDPR

1.2 JavaScript & Node.js : consignes techniques, timing et évaluations

Tâche	Compétences	Critères	Deadline	Pt	Consignes
Objectif du projet	C6) Documenter et présenter en vidéo le développement d'une SPA		Séance 9 : 16/10 ou 17/10		<p>Donnez un nom à votre projet et décrire l'objectif de votre projet au §2 de ce document ainsi que sur https://e-vinci.github.io/web2/project-page, complétez :</p> <ul style="list-style-type: none">- Le nom du projet : Projet N°X : Nom de votre projet- le champs « Description ». <p>Discuter de votre objectif avec un enseignant et assurer vous que cet objectif soit validé avant d'aller plus loin dans votre projet.</p>
Planning des tâches et cas d'utilisation	C6)		Séance 12 : 24/10 ou 27/10		<p>Décrire le planning des tâches et cas d'utilisation selon les instructions données au §6.</p> <p>Présenter votre planning à un enseignant, afin qu'il puisse vous aider à bien prioriser les tâches.</p>
Indiquer l'URL de votre code repository	C6)		Séance 12 : 24/10 ou 27/10		<p>Votre code doit être accessible par tout le monde via un web repository public qui vous sera assigné par GitHub Classroom. Cela permettra notamment aux enseignants de suivre vos avancées tout au long de votre projet. Veuillez indiquer votre URL sur https://e-vinci.github.io/web2/project-page.</p> <p>Plus d'information aux §1.1.2 et §9.1.</p>

Tâche	Compétences	Critères	Deadline	Pt	Consignes
Choix technologiques	C6)		Séance 15 : 13/11 ou 14/11		Compléter le §8. Discuter de vos choix technologiques avec un enseignant.
Rapports individuels d'activités	C6)	Rapports de qualité <i>Indicateurs : formulation de qualité, analyse de qualité, respect des consignes</i>	12/11 19/11 26/11 3/12 10/12 17/12	1 solo	Des sessions individuelles de feedback sont organisées via TEAMMATES permettant à chacun de répondre à des questions dont les réponses sont confidentielles ou anonymisées au sein d'un groupe. Des e-mails seront envoyés vous invitant à compléter un formulaire hebdomadaire, à compléter pendant le WE. À partir de la 2 ^{ème} soumission, tout formulaire hebdomadaire non complété amènera à une pénalité individuelle de 0.5 point. Si vous manquez deux soumissions, vous aurez l'obligation de montrer que vous êtes actif sur le projet sous risque d'être écarté du projet.
Soumission du rapport de groupe	C6)	Idem	17/12	1	Compléter le §10 ainsi que tous les paragraphes qui n'auraient pas été finalisés de ce document. Soumettre ce document, via Moodle (un devoir sera créé) ainsi que dans le répertoire /report de votre repo.

Tâche	Compétences	Critères	Deadline	Pt	Consignes
					Effacer toutes les consignes mises <i>en grisé</i> dans ce document avant de soumettre ce rapport sur Moodle.
Soumission de la vidéo	C6)	Vidéo de qualité <i>Indicateurs : présentation du projet de qualité, analyse de qualité, respect des consignes</i>	17/12	2	Présenter votre projet selon les exigences du §11.
Soumission du code du frontend	C2) Création d'IHM pour SPA inclus : C5 : Intégrer au développement d'une SPA des technologie non vues en cours inclus si nécessaire : C3) Sécurisation de SPA	Qualité de l'IHM produite <i>Indicateurs : esthétique, fonctionnel, codage de qualité, respect des consignes, ambitieux & original, utilisation d'une librairie pour l'IHM non vue en cours</i>	17/12	8	Réaliser un frontend et un backend de Qualité : Code bien structuré, UI et UX de qualité, API bien documentée (documentation des opérations de votre API, requêtes permettant de tester votre API...) Être ambitieux et original. Démontrer une appropriation personnelle du code (via commentaires dans le code, discussion lors des cours...).
Soumission du code du backend	C1 : Création de services web inclus : C5 : Intégrer au développement d'une SPA des technologie non vues en cours inclus si nécessaire : C3 : Sécurisation de SPA	Qualité du web service produit <i>Indicateurs : fonctionnel, codage de qualité, respect des consignes, ambitieux & original, utilisation d'une librairie pour le service web non vue en cours</i>	17/12	5	Respecter les spécifications techniques décrites dans ce document. Déployer votre frontend et votre backend chez un provider gratuit. NB : votre RESTful API doit être un minimum différente des APIs fournies dans les démos du cours de JS.

Tâche	Compétences	Critères	Deadline	Pt	Consignes
Déploiement tant de votre frontend que backend	C4) Déploiement d'applications web	Déploiement de la SPA sur le cloud <i>Indicateurs : fonctionnel, performances de chargement acceptables</i>	17/12	2	
Réaliser un minimum de 5 revues sur le site web	C7) Analyser le développement de SPA faites par des pairs	Revue de projets compréhensibles & constructives <i>Indicateur : présence d'un minimum de 5 revues</i>	Avant examen de 1 ^{ère} session	1 solo	Via https://e-vinci.github.io/web2/my-reviews-page , vous devez revoir les vidéos de présentation de 5 groupes (sauf le vôtre), exécuter leurs applications, et fournir votre critique de chacun de ces projets. Vous pourrez fournir la critique d'autant de projets que vous le souhaitez. Plus d'info sur la revue de projet au § Erreur ! Source du renvoi introuvable.
	TOTAL POINTS			20	<p>Il est à noter que des membres d'un même groupe pourront être cotés différemment en fonction de leur engagement sur le projet. L'engagement d'un étudiant est visible via les rapports individuels d'activités (outil TEAMMATES) via GitHub (GitHub Project, Issues, Milestones, commits...) et lors des sessions de cours.</p> <p>Les étudiants non actifs risquent d'être écarté du projet, spécialement s'ils ne soumettent pas leurs rapports individuels.</p> <p>Les étudiants n'ayant pas réalisé au moins un use case significatif seront considérés inactifs.</p>

Tâche	Compétences	Critères	Deadline	Pt	Consignes
					Les étudiants n'ayant pas participé significativement au projet recevront d'office une lourde pénalité au niveau de leurs points, voire un 0/20.

1.3 Ergonomie : consignes techniques, timing et évaluations

Les « deadlines » données ci-dessous sont les dates où au plus tard l'avancement des tâches doivent être présentables à un enseignant pendant le cours.

Compétence	Tâches	Deadline	Points	Consigne
Reporting & présentation	Objectif du projet	17/10		Décrire l'objectif de votre projet au §2 de ce document. Discuter de votre objectif avec un enseignant et assurer vous que cet objectif soit validé avant d'aller plus loin dans votre projet.
Conception	Définir la vision marketing	5/11	6	Décrire le Mind map du projet. Créer le persona de (s) l'utilisateur (s) ciblé (s) par le projet. Répondre aux axiomes de Morville.
Analyse d'applications web	Architecture UX	5/11	4	Construire les wireframes détaillés de votre application.
	Analyse des résultats et rapport associé	17/12	2	Auditez votre projet et vérifiez le respect des règles GDPR.
	Présentation vidéo	17/12	8	Présenter votre projet en intégrant l'expérience utilisateur.
	TOTAL		20	Il est à noter que différents membres d'un groupe pourront être cotés différemment en fonction de leur engagement sur le projet visible lors des sessions de cours.

2 Objectif du projet

Space Lover est un jeu de "combat" en 2D qui plonge les joueurs dans un univers spatial et futuriste. Dans cette expérience de jeu unique, les joueurs incarnent des robots au design à la fois inoffensif et futuriste, renforçant ainsi l'aspect spatial et avant-gardiste de l'aventure.

Au commencement de la partie, la jauge d'affection des joueurs est totalement vide. **Space Lover** se démarque des jeux de combat traditionnels en mettant de côté les attaques agressives, et en favorisant les câlins et les bisous pour remplir la jauge d'affection de l'adversaire. Le joueur perd la partie lorsque sa propre jauge atteint son maximum. Cette approche unique mélange l'émotion et l'action dans un environnement spatial et futuriste captivant.

Space Lover tire son inspiration du célèbre jeu "Street Fighter" mais se distingue par le style de combat, basé sur l'affection plutôt que la violence. Notre objectif est simple : sensibiliser les joueurs à la non-violence, tout en offrant une expérience ludique et en favorisant des interactions positives. En bref, **Space Lover** prône la paix plutôt que la violence, tout en garantissant une expérience divertissante.

Notre jeu, vise un large public, allant des passionnés de jeux vidéo aux amateurs de divertissement. De plus, il convient à un public de tout âge grâce à l'absence totale d'éléments violents ou traumatisants. **Space Lover** offre une expérience adaptée à tous, favorisant un environnement sûr et divertissant.

3 Mind map du projet

Lien vers la mind map : <https://github.com/e-vinci/web2-2023-project-group-12/blob/main/ergonomics/marketing-view/MindMap.pdf>

4 Persona

Lien vers le fichier des personas : <https://github.com/e-vinci/web2-2023-project-group-12/blob/main/ergonomics/marketing-view/Personas.pdf>

5 Axiomes de Morville

Lien vers le fichier axiomes de Morville : [https://github.com/e-vinci/web2-2023-project-group-12/blob/main/ergonomics/marketing-view/Axiomes de Morville.pdf](https://github.com/e-vinci/web2-2023-project-group-12/blob/main/ergonomics/marketing-view/Axiomes%20de%20Morville.pdf)

6 Planning des tâches et cas d'utilisation

- URL vers votre GitHub Project public : <https://github.com/orgs/e-vinci/projects/61>

7 Besoins techniques

7.1 Système

TRS01 : Vous devez développer une Single Page Application (SPA) à l'aide de JS et Node.js.

TRS02 : Votre RESTful API doit être indépendant de votre frontend ; vous aurez donc deux applications distinctes, une pour le frontend et l'autre pour la RESTful API.

TRS03 : Vous devez utiliser GitHub sur votre projet afin de gérer le développement de chacun des membres d'une équipe.

Nous vous recommandons d'appliquer un workflow vu dans votre cours de DevOps : pour chaque cas d'utilisation / feature que vous développez, essayez de créer une branche correspondante. De plus, il serait intéressant que vous mettiez en œuvre des revues de code au sein de votre projet via des Pull Request sur Github.

7.2 Frontend

TRF01 : Votre frontend doit utiliser Webpack en tant que package bundler.

TRF02 : Le frontend, développé en HTML / CSS (bootstrap ou autre) / JavaScript, doit consommer au moins une de vos RESTful API.

Votre frontend peut consommer des API externes, des APIs que vous n'avez pas développées vous-même (e.g. API de youtube, de google maps...)

TRF03 : Votre frontend doit mettre en œuvre une librairie JS externe, ou l'API Canvas, afin de réaliser une animation.

L'animation peut prendre la forme d'une animation 2D, 3D ou d'un jeu vidéo. Attention à ne pas juste offrir une minuscule animation à l'aide d'une librairie ne demandant aucun code JS, comme certaines librairies mettant tout en œuvre à l'aide de CSS.

TRF04 : Votre frontend doit mettre en œuvre au minimum une librairie JS non vue en cours. *Anime.js est autorisé pour votre animation.*

TRF05 : Votre frontend doit respecter les droits d'auteurs, que ça soit pour les éventuels sons, images, vidéos, librairies et morceaux de codes utilisés. Cela est de votre responsabilité et non pas de celle de vos enseignants.

TRF06 : Vous devez déployer votre frontend sur GitHub Pages ou d'autres providers gratuits supportant votre application.

7.3 API

TRA01 : Vous devez créer une RESTful API afin d'offrir des opérations sur des ressources utiles à votre projet.

La RESTful API ne peut pas être uniquement un « copier/coller » de ressources offertes dans le cours (notamment les ressources users et auths). Vous pouvez utiliser les ressources offertes dans le cours, mais vous devez y apporter des ajouts significatifs.

TRA02 : Votre RESTful API doit mettre en œuvre au minimum un package non vu en cours.

TRA03 : Vous devez documenter les opérations de votre API conformément aux conventions REST. *Vous pouvez documenter votre API soit sous forme de tableau, comme vu dans le cours, soit à l'aide d'outils tel que Swagger.*

TRA04 : Les tests de votre API, les requêtes HTTP, doivent être données au sein de votre projet. Pour chaque opération de votre API, il doit exister au minimum une requête HTTP associée.

TRA05 : Votre API doit respecter les droits d'auteurs, que ça soit pour les éventuelles librairies utilisées, les morceaux de code, les sons, images, vidéos... Cela est de votre responsabilité et non pas de celle de vos enseignants.

TRA06 : Vous devez déployer votre backend sur Azure ou d'autres providers gratuits supportant votre application.

8 Choix technologiques

En créant "Space Lover", nous avons soigneusement sélectionné nos outils pour garantir une expérience utilisateur agréable, tout en restant fidèle à notre message principal qui est de prôner la paix. Ces choix ont été faits en tenant compte des objectifs spécifiques que nous avons identifiés dans la section 6. Dans cette partie, nous allons parler des outils que nous utilisons pour la partie visible de notre application (le frontend) et pour la partie cachée qui gère les données (le backend).

8.1 Frontend

Pour le développement du frontend de notre jeu, nous avons opté pour Phaser (<https://phaser.io>), un framework de jeu en 2D construit en JavaScript. Cette dernière propose une gamme variée de fonctionnalités puissantes pour la réalisation de jeux interactifs. Sa documentation complète en fait un choix idéal pour notre projet.

8.2 RESTful API

Quant à notre backend, nous avons choisi d'intégrer l'API Advice Slip (<https://api.adviceslip.com/>), qui génère des conseils aléatoires, ajoutant une dimension instructive et divertissante à notre application. Le choix de cette API s'aligne parfaitement avec notre objectif de promouvoir des interactions positives et d'offrir des conseils inspirants à nos utilisateurs.

8.3 Wireframe

Lien vers le fichier wireframe : <https://github.com/e-vinci/web2-2023-project-group-12/blob/main/ergonomics/wireframes/Wireframes.pptx>

9 Conception & Implémentation

9.1 Code repositories

- URL pour le web repository public associé à votre projet : <https://github.com/e-vinci/web2-2023-project-group-12>

9.2 Secrets éventuels pour vos API ou base de données

Nous n'avons pas de secrets éventuels.

9.3 Documentation de votre API

Une ressource de type **users** contient les propriétés suivantes :

- **id** : un entier,
- **username** : nom du user (String) et il doit être unique,
- **password** : mot de passe du user ; il doit avoir au moins 5 caractères et il est hashé (String),
- **gamesPlayed** : nombre de parties totales jouées (un entier),
- **gamesWon** : nombre de parties gagnées (un entier),
- **gamesLost** : nombre de parties perdues (un entier).

- Tableaux représentant les opérations de votre API ou lien vers la documentation de votre API :

URI	Méthode HTTP	Opération
/auths/login	POST	Connecter un utilisateur si le username et le mot de passe correspondent
/auths/register	POST	Créer un utilisateur (avec un username et mot de passe) et l'enregistrer dans la base de données
/auths/loginSecondPlayer	POST	Connecter le deuxième joueur si le username et le mot de passe correspondent ainsi que si le premier joueur est déjà connecté
/auths/logout	GET	Déconnecter un utilisateur si celui-ci est connecté
/ranks	GET	Lister les 4 premiers utilisateurs ayant le nombre de parties gagnées le plus élevée
/users/{id}	PATCH	Mettre à jour le score des utilisateurs en fonction de la fin de la partie du jeu
/users/{username}	GET	Lire un user selon son username

- Requêtes HTTP se trouvent dans : **/api/REST Client**

9.4 Déploiement de vos applications

- URL de votre frontend déployé : <https://youssef-ouftallah-vinci.github.io/SpaceLover/>
- URL de votre RESTful API déployée : <https://spacelover.azurewebsites.net/>

9.5 Code réutilisé

Mis à part le boilerplate du projet, nous n'avons pas réutiliser de code.

10 Analyse des résultats par le groupe

10.1 Évaluation du résultat par rapport au planning des tâches et des cas d'utilisation

Nous avons pleinement réalisé tous les objectifs fonctionnels que nous nous étions fixés initialement, comme regroupé au point 6. Notre site web **Space Lover** est prêt à être utilisé et propose une expérience captivante comme nous le souhaitions.

Aucune déviation par rapport à nos objectifs initiaux établis n'a été remarqué. C'est donc le fruit résultant d'une planification soignée ainsi qu'une communication structurée et régulière au sein de l'équipe. Nous avons utilisé toutes les ressources disponibles de manière efficace, et les compétences diverses de chaque membre du groupe ont favorisé une progression harmonieuse du projet.

Pour conclure, notre application répond aux attentes que le groupe s'est fixé initialement. Des choix technologiques bien pensés et une collaboration efficace ont été primordial dans notre succès pour la réalisation de notre site.

10.2 **Audit ergonomique de votre projet**

En termes d'ergonomie, notre site offre une expérience attrayante avec des couleurs harmonieuses et une police d'écriture claire, garantissant une lisibilité optimale. La conception est soignée, limitant les pop-ups pour faciliter la navigation. Chaque élément est explicitement présenté, rendant l'ensemble du site intuitif et facilement compréhensible.

En ce qui concerne le respect des règles GDPR, nous avons intégré des mesures de sécurité et de confidentialité, garantissant la protection des données de nos utilisateurs en ajoutant un bouton "Accept our policy" (accepter notre politique), assurant ainsi une transparence totale et permettant aux utilisateurs de donner leur consentement en toute connaissance de cause. Notre approche centrée sur l'utilisateur s'aligne sur les principes de l'ergonomie et du respect de la vie privée, assurant une expérience en ligne positive et sécurisée.

10.3 Difficultés techniques rencontrées

Au cours de notre projet, nous avons été confrontés à certaines difficultés techniques qui ont nécessité une attention particulière. L'une de ces difficultés s'est manifestée lors de la mise en œuvre de la page de scores, plus spécifiquement dans la mise à jour des scores des utilisateurs. Le mécanisme d'affichage des scores a posé un défi et nous avons dû consacrer du temps supplémentaire à résoudre ce problème pour garantir une précision et une lisibilité optimales des résultats.

Par ailleurs, le déploiement de l'API a également présenté des complications. La mise en place de l'infrastructure nécessaire pour héberger et rendre accessible notre API a demandé une compréhension approfondie des outils de déploiement et a nécessité des ajustements pour garantir une intégration fluide avec le reste de notre application.

Bien que nous ayons rencontré des défis techniques, notre équipe a travaillé ensemble de manière étroite et a réalisé des ajustements réguliers pour surmonter ces obstacles. Ces efforts ont abouti à une application stable et fonctionnelle.

10.4 Conseils pour appliquer cette technologie

Dans la perspective d'appliquer une technologie similaire, il aurait été utile de disposer de connaissances approfondies sur la procédure de déploiement de l'API du projet. Cela aurait contribué à éviter une perte de temps et d'énergie en plus que ce qu'il ne fallait.

La mise en place d'une communication régulière entre les membres de notre équipe a joué un rôle crucial dans l'évitement de conflits majeurs entre les divers codes. En particulier, l'utilisation efficace du système de branches sur GitHub a permis une gestion plus fluide du développement. Cela a également favorisé un développement parallèle, où les fonctionnalités pouvaient être développées indépendamment.

Pour faciliter le processus d'élaboration d'un site web, des ressources en ligne se sont avérées précieuses, telles que des intelligences artificielles pour obtenir des conseils (<https://www.chatgpt.com/>), des tutoriels vidéo (<https://www.youtube.com/>), et des codes partagés

par d'autres développeurs sur GitHub (<https://github.com/>). Ces ressources ont enrichi notre compréhension et ont été des sources d'aide précieuses.

10.5 Quels sont les points positifs à la manière dont s'est déroulée la collaboration au sein du groupe ?

La collaboration au sein de notre groupe a été très positive, principalement grâce à une répartition claire des tâches dès le début du projet. Chaque membre a pu se concentrer sur des aspects spécifiques du développement, exploitant ainsi au mieux ses compétences. De plus, nous avons établi des rendez-vous réguliers sur Discord pour discuter de l'avancement du projet, partager nos idées et résoudre rapidement d'éventuels problèmes. Ces réunions ont également été l'occasion de travailler sur les parties communes du projet.

Dans l'ensemble, l'engagement de chaque membre, la communication fréquente et la répartition judicieuse des responsabilités ont contribué positivement à la réussite de notre collaboration.

10.6 Quels sont les points qui seraient à améliorer pour de futures collaborations ?

Malgré une collaboration généralement efficace, il y a toujours des points que nous pourrions améliorer pour de futures collaborations.

Tout d'abord, nous pourrions optimiser davantage notre gestion du temps en fixant des délais plus spécifiques pour certaines étapes du projet.

Par ailleurs, une communication encore plus transparente sur les difficultés rencontrées par chacun pourrait également améliorer notre capacité à résoudre rapidement les problèmes.

Enfin l'utilisation d'outils collaboratifs supplémentaires pourrait renforcer notre efficacité globale.

En somme, en identifiant ces domaines d'amélioration potentiels, nous pourrions rendre nos futures collaborations encore plus fructueuses.

11 Présentation vidéo

Lien vers la vidéo youtube : <https://www.youtube.com/watch?v=AiPDu00hJBQ>