

Ergonomie & développement d'une SPA animée

Remember Or Die

Auteur 1	BUDAK Melik
Auteur 2	CIBOROWSKI Dawid
Auteur 3	DEMIR Ahmet Kusay
Auteur 4	GARCIA Alexandre
Auteur 5	KOUBAI Omar
Date	17.12.2023
Référence	WEB2-2023-PROJECT-GROUP-29
Version	1.2

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.....	5
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.....	13
4.1	Joueuse occasionnelle	13
4.2	Joueur compétitif	14
4.3	Joueuse sociale.....	14
4.4	Collectionneur de personnages	15
4.5	Novice	15
4.6	Accro au défi.....	16
5	Axiomes de Morville	16
6	Planning des tâches et cas d'utilisation.....	18
7	Besoins techniques	18
7.1	Système	18
7.2	Frontend	18
7.3	API	19
8	Choix technologiques	19
8.1	Frontend	20
8.2	RESTful API.....	20
8.3	Wireframe.....	21
9	Conception & Implémentation	21
9.1	Code repositories	21
9.2	Secrets éventuels pour vos API ou base de données.....	21

9.3	Documentation de votre API.....	22
9.4	Déploiement de vos applications	24
9.5	Code réutilisé	24
10	Analyse des résultats par le groupe	25
10.1	Évaluation du résultat par rapport au planning des tâches et des cas d'utilisation	25
10.2	Audit ergonomique de votre projet.....	25
10.3	Difficultés techniques rencontrées	26
10.4	Conseils pour appliquer cette technologie	26
10.5	Quels sont les points positifs à la manière dont s'est déroulée la collaboration au sein du groupe ?	27
10.6	Quels sont les points qui seraient à améliorer pour de futures collaborations ?.....	27
11	Analyses individuelles des résultats	27
12	Présentation vidéo	28
13	Revue de projets par les pairs	28

1 Consignes et évaluations


1.1 Consignes générales

1.1.1 Création des groupes sur le site du cours

Veillez 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 accéderont 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 §1.16.</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 §12.
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 §13.
	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.

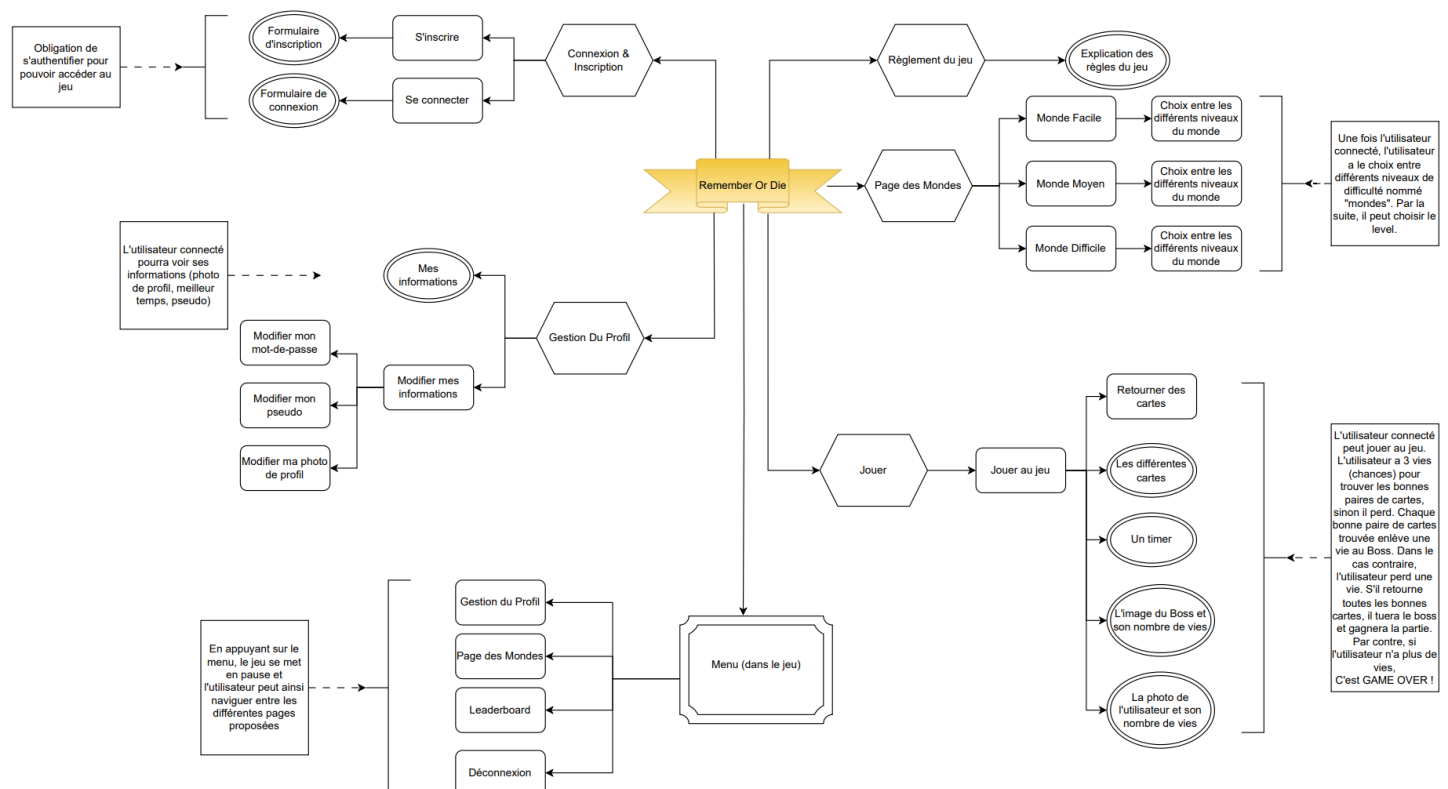
Compétence	Tâches	Deadline	Points	Consigne
Reporting & présentation	Objectif du projet	17/10		Décrire l'objectif de votre projet au §2 de ce document. Discuter de votre objectif avec un enseignant et assurer vous que cet objectif soit validé avant d'aller plus loin dans votre projet.
Conception	Définir 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.

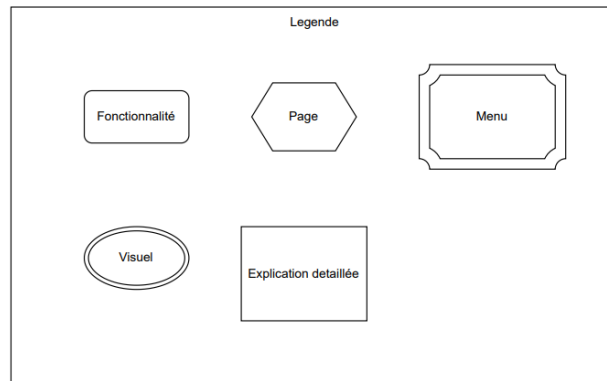
2 Objectif du projet

Le groupe 29 a décidé de réaliser un jeu de Memory avec des fonctionnalités avancées pour offrir une expérience ludique et engageante aux utilisateurs. Cette application est axée sur la compétition individuelle, et les joueurs ne coopèrent pas directement les uns avec les autres. Les joueurs peuvent rivaliser les uns contre les autres, que ce soit entre amis, en famille, ou avec des joueurs du monde entier.

Ce projet nous tient à cœur en tant que groupe 29, car il combine notre passion pour le développement informatique, notre amour des jeux et notre désir de créer des expériences de jeu captivantes. Il vise à offrir un divertissement de qualité tout en stimulant la mémoire et les compétences de résolution de problèmes des joueurs. La progression graduelle, les options de personnalisation de profil et les défis du jeu ajoutent une dimension motivante, encourageant les joueurs à revenir et à s'améliorer continuellement. De plus, l'aspect compétitif des classements mondiaux et des classements d'amis ajoute une couche de dynamisme au projet, incitant les joueurs à se dépasser et à interagir avec d'autres passionnés de jeux de mémoire.

3 Mind map du projet





4 Persona

1.4 Joueuse occasionnelle



<u>Nom</u> : Lisa
<u>Âge</u> : 25 ans
<u>Expérience de jeu</u> : Joue occasionnellement à des jeux en ligne.
<u>Objectif</u> : Rechercher un jeu décontracté pour passer le temps et se détendre.
<u>Frustrations</u> : Lisa a un emploi à temps plein et une vie sociale bien remplie. Elle peut être frustrée par des jeux qui demandent trop de temps et d'engagement.
<u>Contexte</u> : Lisa travaille comme infirmière et a une vie familiale active avec un partenaire et deux jeunes enfants.

1.5 Joueur compétitif



<u>Nom</u> :	Max
<u>Âge</u> :	30 ans
<u>Expérience de jeu</u> :	Un joueur passionné avec une histoire de jeu compétitif.
<u>Objectif</u> :	Chercher des défis, grimper les classements et devenir le meilleur.
<u>Frustrations</u> :	Max peut être frustré par les jeux qui manquent de contenu compétitif ou qui ne récompensent pas les joueurs talentueux.
<u>Contexte</u> :	Max travaille comme développeur de logiciels et consacre une grande partie de son temps libre aux jeux.

1.6 Joueuse sociale



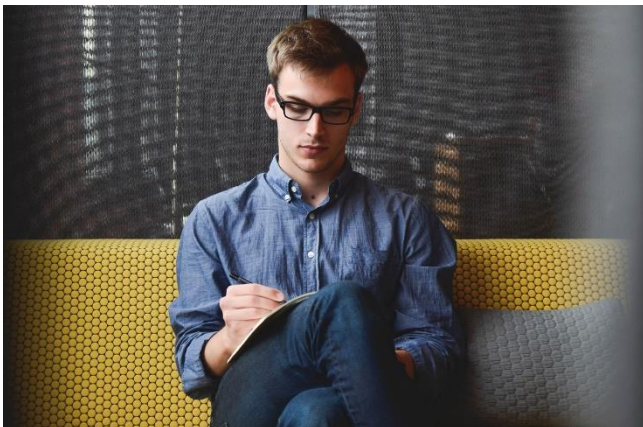
<u>Nom</u> :	Emma
<u>Âge</u> :	22 ans
<u>Expérience de jeu</u> :	Aime les jeux sociaux et jouer avec des amis en ligne.
<u>Objectif</u> :	Jouer en ligne avec des amis et rivaliser dans un environnement amical.
<u>Frustrations</u> :	Emma peut être frustrée si le jeu ne permet pas de connecter facilement avec des amis ou si la fonctionnalité de leaderboard d'amis est limitée.
<u>Contexte</u> :	Emma est étudiante en psychologie et joue en ligne avec un groupe d'amis proches.

1.7 Collectionneur de personnages



<u>Nom</u> : Alex
<u>Âge</u> : 22 ans
<u>Expérience de jeu</u> : Collectionneur de jeux et d'objets en jeu.
<u>Objectif</u> : Acquérir des éléments pour personnaliser son profil de joueur.
<u>Frustrations</u> : Alex peut être frustré si le processus de déblocage d'éléments est trop lent ou complexe.
<u>Contexte</u> : Alex est designer graphique et adore personnaliser son espace de jeu.

1.8 Novice



<u>Nom</u> : David
<u>Âge</u> : 18 ans
<u>Expérience de jeu</u> : Débutant sans expérience de jeu antérieure.
<u>Objectif</u> : Apprendre à jouer à des jeux en ligne et s'amuser.
<u>Frustrations</u> : David peut être frustré par des jeux qui sont difficiles à comprendre ou à jouer pour un débutant.
<u>Contexte</u> : David est étudiant en première année à l'université.

1.9 Accro au défi



Nom : Sandrine

Âge : 40 ans

Expérience de jeu : Recherche des jeux qui mettent sa mémoire et sa rapidité à l'épreuve.

Objectif : Améliorer sa mémoire et sa concentration tout en s'amusant.

Frustrations : Sandrine peut être frustrée par des jeux qui manquent de complexité et de défis.

Contexte : Sandrine travaille comme professeur d'école primaire et est mère de deux enfants.

5 Axiomes de Morville

1. **Utile** :

- Le jeu de memory permet aux utilisateurs de jouer à un jeu amusant de correspondance de cartes.
- Les utilisateurs peuvent choisir différents niveaux de difficulté pour adapter le jeu à leurs préférences.
- Le jeu intègre un système de timer, des monstres et un leaderboard pour ajouter un élément de défi.
- Le jeu intègre un système d'XP qui permet de monter en niveau et d'avoir des récompenses.

2. **Désirable** :

- Le jeu présente des graphismes attrayants grâce aux différentes animations.
- Les cartes et les éléments du jeu sont conçus pour plaire visuellement aux joueurs.

- Le jeu comporte un système de progression pour motiver les joueurs en leur offrant des récompenses et des objectifs à atteindre, les incitant ainsi à continuer à jouer et à s'investir davantage.

3. **Accessible :**

- Le jeu repose sur la correspondance visuelle de cartes, il ne sera donc pas accessible aux utilisateurs malvoyants.
- Le jeu est développé en français, il sera accessible à un public francophone.

4. **Crédible :**

- Le jeu inspire confiance grâce à un système d'inscription sécurisé.
- Les classements (leaderboard) sont fiables et reflètent les vrais temps des joueurs.
- Les récompenses sont obtenues en fonction du niveau de l'utilisateur dans le jeu.
- Le jeu permet une amélioration des capacités cognitives, une adaptabilité à différents âges et la familiarité qui en fait un choix solide et fiable pour de nombreux types de publics.

5. **Trouvable :**

- Le jeu est facilement accessible via un site web et son URL.
- Les classements sont facilement accessibles, il suffit juste de s'identifier dans le jeu.

6. **Valable :**

- Le jeu offre une valeur ludique en proposant un défi de mémoire amusant avec des monstres à battre.
- Les classements permettent de suivre les performances et de créer une compétition amicale.

7. **Utilisable :**

- L'interface utilisateur est simple et conviviale, offrant une expérience fluide et agréable pour les joueurs.
 - ➔ Partie login/register : Pour pouvoir jouer, il faut être inscrit. Il suffit juste d'introduire un nom d'utilisateur et un mot de passe. Cela permet de personnaliser l'expérience et de sauvegarder la progression.
 - ➔ Page principale : Choix du niveau de difficulté, accès à son profil + récompenses, accès aux leaderboards, accès à ses contacts (amis), favorisant ainsi une aventure progressive et sociale.
 - ➔ Page du jeu : Les cartes s'étendent sur une grande partie de l'écran pour une meilleure visibilité, affichage du monstre et de ses points de vie, optimisé pour le jeu et pour l'expérience de l'utilisateur.
- Les interactions, comme le retournement de cartes, sont intuitives facilitant ainsi le déroulement du jeu.
- La connexion obligatoire ne doit pas être perçue comme une contrainte, mais comme une opportunité en raison des fonctionnalités avancées et d'une progression enregistrée.

6 Planning des tâches et cas d'utilisation

- URL vers votre GitHub Project public : <https://github.com/orgs/e-vinci/projects/122>

7 Besoins techniques

1.10 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.

1.11 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.

1.12 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

Ayant un projet ambitieux qui nécessite un système de gestion de données complexe, nous avons opter pour l'utilisation de la librairie 'node-postgres' (pg) afin de stocker nos données dans une base de données dédiée à notre projet. Cette librairie nous permet de réaliser des associations complexes entre différentes relations, ce qui, sans système de gestion de base de données, serait difficile d'implémenter. Prenons l'exemple d'une alliance : c'est la relation entre un joueur et un autre. Cependant, il faut ajouter à cela, leurs scores, le renvoi des meilleurs selon un monde, selon

les alliances, etc. De ce qui pourrait paraître anodin, naît un ensemble de relations de données complexe, où ces dernières sont intrinsèquement liées les unes avec les autres.

L'infrastructure de notre base de données est hébergée à Paris grâce à ElephantSQL (<https://www.elephantsql.com>), offrant ainsi une performance et une accessibilité optimales pour notre système relationnel. En exploitant cette plateforme, nous bénéficions d'une solution robuste pour gérer ces relations intriquées les unes dans les autres.

L'application des concepts vus lors du cours de Gestion de Bases de données s'est révélée cruciale dans la conception et la mise en œuvre efficace de notre système. Ces fondements nous ont fourni une compréhension approfondie des mécanismes sous-jacents, facilitant ainsi la modélisation et l'optimisation de notre base de données pour répondre à nos exigences.

Quant à l'affichage, nous nous sommes penché vers anime.js afin de nous fournir un système de retournement de carte efficace et plaisant.

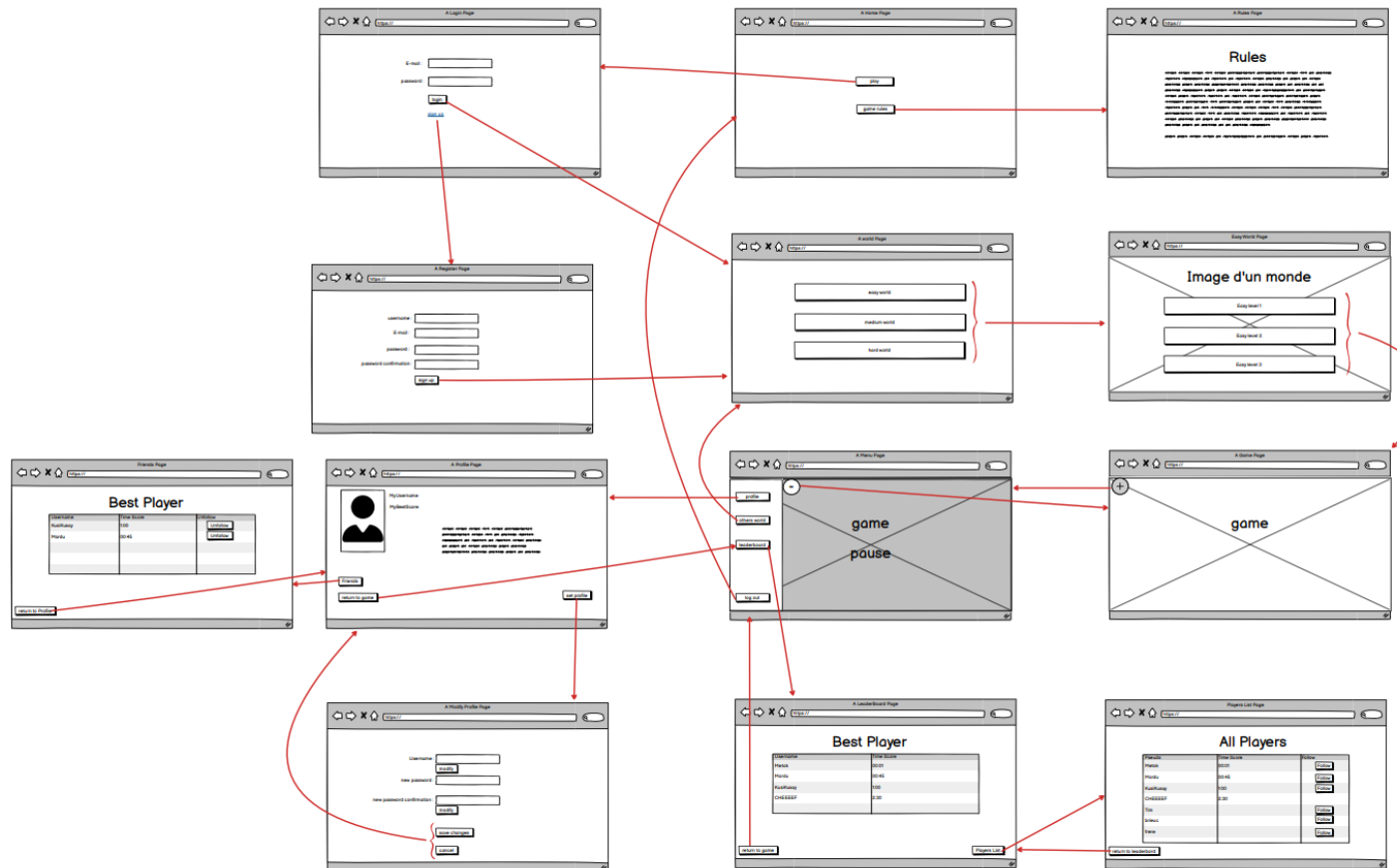
1.13 Frontend

Pour le frontend, vu que nous voulions faire un Memory Game nous avons utilisé animejs qui nous permet d'animer le retournement des cartes. De la même manière, nous avons animé la diminution de points de vie au niveau des boss. *Animejs* => <https://animejs.com>

1.14 RESTful API

Pour le backend, nous avons utilisé node-postgres => <https://node-postgres.com>

1.15 Wireframe



9 Conception & Implémentation

1.16 Code repositories

- URL pour le web repository public associé à votre projet : <https://github.com/e-vinci/web2-2023-project-group-29>

1.17 Secrets éventuels pour vos API ou base de données

Nous avons utilisé des secrets afin d'utiliser une database hébergée. Le fichier .env contenant les secrets et la console qui nous a permis de créer la base de données. Tout est remis sur moodle.

1.18 Documentation de votre API

- Tableaux représentant les opérations de notre API :

Objectifs	Routes	Router	Action	Méthode
alliances				
Obtenir la liste des alliés d'un joueur	/alliances	alliances.js	/allies/:id	GET
Obtenir la liste des invitations d'alliance pour un joueur	/alliances	alliances.js	/invitations/:id	GET
Ajouter un joueur en tant qu'allié	/alliances	alliances.js	/:id	POST
Retirer un joueur de la liste des alliés	/alliances	alliances.js	/:id/:ally	DELETE
Accepter ou rejeter une invitation d'alliance	/alliances	alliances.js	/:id/:sender	PUT
levels				
Obtenir la liste de tous les niveaux	/levels	levels.js	/	GET
Obtenir un niveau spécifique en fonction de son identifiant	/levels	levels.js	/:id	GET
Obtenir l'identifiant d'un niveau en fonction de son monde et de son numéro de niveau	/levels	levels.js	/:world/:level	GET
players				
Obtenir tous les joueurs	/players	players.js	/	GET

Enregistrer un nouveau joueur	/players	players.js	/register	POST
Authentifier et connecter un joueur	/players	players.js	/login	POST
Mettre à jour le mot de passe du joueur	/players	players.js	/updatePassword	PUT
Mettre à jour l'avatar du joueur	/players	players.js	/updateAvatar	PUT
scores				
Obtenir le dernier niveau atteint par un joueur en fonction de son identifiant	/scores	scores.js	/lastLevel/:id	GET
Obtenir les meilleurs scores pour un monde spécifique en fonction de son identifiant	/scores	scores.js	/bestScores/:id	GET
Obtenir les meilleurs scores des amis d'un joueur pour un monde spécifique en fonction de son identifiant et de l'identifiant du monde	/scores	scores.js	/friendsBestScores/:id/:worldId	GET
Ajouter un score pour un joueur à un niveau spécifique	/scores	scores.js	/addScore	POST

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

1.19 Déploiement de vos applications

- URL de votre frontend déployé : <https://e-vinci.github.io/remember-or-die>
- URL de votre RESTful API déployée : <https://remember-or-die-api.azurewebsites.net>

1.20 Code réutilisé

Chemin du fichier où se trouve le code réutilisé	Auteur du code source réutilisé	URL où le code réutilisé est disponible	Raison de la réutilisation du code
/frontend/src/Components/GamePage.js	Joshua McFarland	https://codesandbox.io/s/e7ou1	Code montrant comment retourner une carte avec AnimeJs
/frontend/src/Components/GamePage.js	ChatGpt	https://chat.openai.com/	Code permettant la création d'une fonction qui mélange un tableau de cartes en Javascript
/frontend/src/Components/GamePage.js	ChatGpt	https://chat.openai.com/	Code qui permet de créer un chronomètre et le formater en Javascript
/frontend et /api	ChatGpt	https://chat.openai.com/	Parfois nous avons eu besoin d'un accès rapide

Chemin du fichier où se trouve le code réutilisé	Auteur du code source réutilisé	URL où le code réutilisé est disponible	Raison de la réutilisation du code
			à la documentation Et aussi afin de nous rappeler certain concept vu en cours.

10 Analyse des résultats par le groupe

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

La répartition des tâches s'est déroulée de manière excellente, attribuant à chaque membre du groupe un nombre égal de cas d'utilisation. Cependant, en tant que chef de groupe, Dawid a assumé davantage de responsabilités, ce qui l'a mis en situation de surcharge. De la création des cas d'utilisation à leur attribution, il a pris en charge ces étapes. Cette approche a contribué à la structuration efficace de notre projet. Malgré le non-respect fréquent des délais, les cas d'utilisation ont été finalisés avec succès.

Évidemment, des changements de tâches ont eu lieu entre les membres tout au long du projet. Ces ajustements ont été effectués suite à un accord entre les membres, soit pour résoudre des problèmes de compréhension, soit pour pallier des retards dans la réalisation des tâches. Bien que certains cas d'utilisation aient été attribués à un seul membre, chacun a bénéficié de conseils et d'assistance pour mener à bien ses tâches.

1.22 **Audit ergonomique de votre projet**

La politique de confidentialité de Remember or Die expose clairement la collecte et l'utilisation des données personnelles des utilisateurs. Elle requiert des informations telles que l'adresse e-mail, le mot de passe, le pseudo et l'avatar pour personnaliser l'expérience sur le site, tout en permettant l'utilisation des fonctionnalités du système d'amis et du leaderboard.

Elle met l'accent sur la sécurité des données, déclarant la mise en place de mesures de protection appropriées contre tout accès non autorisé. Les utilisateurs bénéficient d'un certain contrôle en pouvant modifier leur mot de passe, leur avatar et gérer leur liste d'amis directement sur la plateforme. La durée de conservation des données est déterminée par leur nécessité pour l'utilisation du service et la gestion des amis, en plus du respect des obligations légales.

Cependant, l'adresse e-mail de contact pour des questions relatives à la politique de confidentialité n'est pas mentionnée, ce qui pourrait être considéré comme un manque de transparence. En conclusion, la politique aborde globalement les principes du RGPD concernant la collecte, l'utilisation et la sécurité des données, mais pourrait bénéficier d'une précision accrue pour assurer une conformité totale avec les normes de protection des données.

1.23 Difficultés techniques rencontrées

Chaque membre du groupe a rencontré des difficultés techniques n'est-ce pas le b.a.ba des informaticiens. Cependant les difficultés à retenir sont la partie register et login qui n'était pas évidente à réaliser car c'était de la nouvelle matière. Pour la réaliser on a dû apprendre tout ce qui était en lien avec la sécurisation des SPA c'est à dire les notions comme l'authentification et autorisation d'accès aux opérations d'une RESTful API via JWT, la sécurisation d'une RESTful API, la mise en place du localStorage pour sauvegarder les données de session,

Le création de la base de données a été assez complexe étant donné que nous avons décidé d'avoir une fonctionnalité d'amis et de leaderbord. La réalisation des tables était complexe. L'utilisation des concepts avancés vus au cours était obligatoire (triggers, procédures, ...).

Nous avons rencontré aussi quelques difficultés au niveau du squelette du jeu car nous avons insisté sur le fait de créer le jeu de A à Z en y ajoutant quelques fonctionnalités. Résultant a des problèmes logiques du jeu comme par exemple le fait de restreindre le joueur à pouvoir cliquer sur 2 cartes à la fois ou bien encore empêcher le joueur de cliquer sur les cartes au début du jeu quand le temps de mémorisation est en cours. Tous les détails de logique de jeu devenaient des problèmes à résoudre.

1.24 Conseils pour appliquer cette technologie

Voici les conseils que nous souhaitons partager, en commençant par cette question : qu'aurions-nous aimé savoir avant de débiter le projet ? En réalité, il est essentiel de comprendre que les problèmes surgissent souvent vers la fin. Bien que notre projet se déroulait sans encombre, à quelques jours de la date limite, plusieurs erreurs inattendues sont apparues. Ces problèmes ont

émergé lorsque nous avons établi des liens entre les différentes pages. Il est crucial de rester vigilant durant les derniers jours avant la date limite.

Notre deuxième conseil consiste en une question à se poser lors de la réflexion sur le projet : peut-on achever le projet à temps ? Si votre projet demande plus de temps que prévu, vous risquez de ne pas pouvoir le finaliser.

Un autre conseil important pour éviter certains problèmes est de surveiller le moment où vous déclarez vos variables.

De plus, voici quelques liens qui pourraient s'avérer utiles pour l'application de cette technologie.

Lien numéro 1 : l'endroit où trouver des solutions à vos problèmes :

- Cours de JS & Node.js (e-vinci.github.io) (le cours proposé par les professeurs, bien évidemment)

Et voici le deuxième lien :

- [anime.js • JavaScript animation engine \(animejs.com\)](https://anime.js.com/)

Ce lien propose des animations pour la 2D, ce qui est crucial pour éviter que le projet ne paraisseterne.

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

Voici une reformulation du texte pour améliorer la clarté et la fluidité :

Chaque membre du groupe a fourni un excellent travail. La cohésion présente au sein du groupe a été essentielle pour la réalisation de ce merveilleux projet. Les erreurs survenues ont été accompagnées d'encouragements et de sourires motivants. Ainsi, aucune baisse de motivation n'a affecté l'équipe. Comme mentionné précédemment, le chef de groupe, Dawid, a été un pilier et un soutien moral pour tous les membres de l'équipe. L'entraide a été le pilier central de cette cohésion.

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

Notre collaboration est excellente, ce qui signifie que nous ne pourrions pas répondre correctement à cette question.

11 Analyses individuelles des résultats

Voir TEAMMATES.

12 Présentation vidéo

Lien vers la vidéo YouTube : <https://www.youtube.com/watch?v=RtcaMU8Ew0>

13 Revues de projets par les pairs