

Cahier des charges

Community-Services

Steven Durand

25.05.2024

Community-ServicesTous les servicesProfil


Tous les services

Page 1 / 3

Suivant

Catégorie : Informatique

Tuto ViteJS



Comment démarrer rapidement avec ViteJS


Durée : 2 heure(s)
Coût : 200 CommuniTokens

Commander Contacteur

Ajouté par : bricoleur@brico.fr qui a déjà rendu 3 service(s).
Le : 26/05/2024 à 18:10:44

Catégorie : Informatique

Full-stack dev



Devenir full stack en 1h par jour :)

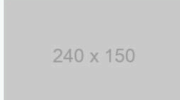
Durée : 2 heure(s)
Coût : 200 CommuniTokens

Commander Contacteur

Ajouté par : test@test.fr qui a déjà rendu 2 service(s).
Le : 26/05/2024 à 17:39:19

Catégorie : Jardinage

Tondre gazon



Je vous propose de tondre votre gazon

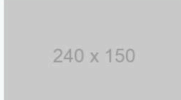
Durée : 1 heure(s)
Coût : 150 CommuniTokens

Commander Contacteur

Ajouté par : jardinier@plantes.fr qui a déjà rendu 0 service(s).
Le : 25/05/2024 à 14:44:07

Catégorie : Jardinage

Planter arbre



Je vous propose de planter l'arbre que vous voulez

Durée : 1 heure(s)
Coût : 150 CommuniTokens

Commander Contacteur

Ajouté par : jardinier@plantes.fr qui a déjà rendu 0 service(s).
Le : 25/05/2024 à 14:43:52

PRÉSENTATION

- PRÉSENTATION
- LE PUBLIC CIBLE
- LES BESOINS ET OBJECTIFS
- LES FONCTIONNALITÉS ATTENDUES
- CONTRAINTES ET LIMITES DU PROJET
- LE RÉFÉRENCEMENT SEO
- LES RESSOURCES HUMAINES NÉCESSAIRES
- RETRO-PLANNING
- LE BUDGET
- **RÉGLEMENTATION GÉNÉRALE PROTECTION DES DONNÉES**
- LA SÉCURITÉ
- ACCESSIBILITÉ
- L'ADAPTABILITÉ DU SITE
- ECO-CONCEPTION
- LES SPÉCIFICATIONS TECHNIQUES
- VEILLES TECHNIQUES
- LES OUTILS COLLABORATIFS
- LE MODEL CONCEPTUEL DE DONNÉES
- L'ARBORESCENCE DU SITE
- AMÉLIORATIONS FUTURES

PRÉSENTATION

Le présent cahier des charges a pour objectif de définir les besoins et les exigences pour le développement d'une plateforme en ligne dédiée à l'échange de services entre particuliers, basée sur un système de points.

Ce projet vise à promouvoir l'entraide locale et à valoriser les compétences de chaque individu au sein de la communauté. En permettant aux utilisateurs de proposer et de recevoir divers services sans recours à une transaction monétaire traditionnelle, nous espérons faciliter les échanges sociaux et renforcer les liens entre les membres.

La plateforme permettra aux utilisateurs de créer des profils, de lister les services qu'ils offrent et recherchent, et de gérer leur solde de points de manière transparente.

En détaillant les fonctionnalités, les performances attendues, les interfaces utilisateur, et les critères de qualité, ce document assure une compréhension claire et partagée des attentes du projet entre toutes les parties prenantes.

L'objectif ultime est de créer une plateforme conviviale et sécurisée qui encourage l'entraide et renforce les liens sociaux entre les membres de la communauté.

PUBLIC CIBLE

Le public cible de notre plateforme d'échange de services est très large. Cette plateforme s'adresse à des personnes de tous âges (majeures), ayant des compétences variées et souhaitant offrir ou recevoir des services dans divers domaines tels que le bricolage, la garde d'enfants, les cours particuliers, ou encore l'aide informatique.

En particulier, nous visons :

Les jeunes adultes : Étudiants ou jeunes professionnels, souvent en quête de services abordables ou d'opportunités pour mettre à profit leurs compétences.

Les familles : Cherchant des solutions pratiques et économiques pour répondre à leurs besoins quotidiens tout en tissant des liens sociaux dans leur quartier.

Les retraités : Désireux de rester actifs, de partager leur expérience et leurs compétences tout en bénéficiant de services dont ils pourraient avoir besoin.

Les nouveaux arrivants : Souhaitant s'intégrer dans leur nouvelle communauté et trouver des moyens d'échanger des services localement.

En ciblant ces groupes spécifiques, la plateforme vise à créer un réseau dynamique et diversifié, capable de répondre à une large gamme de besoins tout en favorisant la coopération et l'entraide au sein de la communauté.

LES BESOINS ET OBJECTIFS

La création de la plateforme d'échange de services repose sur l'identification de besoins spécifiques et sur des objectifs clairs visant à offrir une solution pratique et efficace à notre public cible.

Les principaux besoins identifiés incluent :

1. **Facilitation des échanges de services** : Besoin d'une plateforme intuitive et facile à utiliser permettant aux utilisateurs de proposer et de rechercher des services diversifiés.
2. **Renforcement des liens communautaires** : Importance de créer des interactions sociales positives et de renforcer le tissu social au sein des communautés urbaines et périurbaines.
3. **Accessibilité et inclusivité** : Besoin d'une plateforme accessible à tous, indépendamment de l'âge, des compétences technologiques, des handicaps ou des ressources économiques.

Les objectifs de la plateforme sont les suivants :

1. **Créer une communauté active et engagée** : Développer une base d'utilisateurs diversifiée et active, participant régulièrement aux échanges de services.
2. **Assurer la sécurité et la confiance** : Mettre en place des mesures de sécurité et de vérification pour garantir des échanges sûrs et fiables entre les utilisateurs.
3. **Offrir une expérience utilisateur optimale** : Développer une interface utilisateur intuitive, facilitant la navigation et l'utilisation de la plateforme.
4. **Promouvoir l'entraide et l'échange équitable** : Encourager une culture de solidarité et de coopération en valorisant les contributions de chaque membre de la communauté par un système basé sur la volonté et la confiance.

FONCTIONNALITÉS ATTENDUES

Tout le monde :

- Lister tous les services disponibles
- Rechercher un service par mot clef
- S'inscrire

Utilisateurs inscrits :

- Se connecter, se déconnecter
- Supprimer son compte, changer son mot de passe
- Ajouter et supprimer un ou des service(s) avec une image
- Commander un ou des service(s) si son solde est suffisant
- Contacter les autres utilisateurs et être contacté
- Finaliser une commande

Administrateurs :

- Modérer les services
- Bannir un utilisateur

CONTRAINTES ET LIMITES DU PROJET

Date de livraison : 19/07/2024

1. Contraintes budgétaires :

- Investissement personnel de 40 000 euros.
- L'application ne propose aucun moyen de monétisation et ne générera donc aucun revenu. Cependant, le coût de maintenance reste faible et nous restons ouverts à d'éventuels sponsors.

2. Contraintes immatérielles :

- La communication et l'obtention des autorisations entre les différentes organisations de l'État peuvent prendre du temps et être difficiles.
- Un avocat devra valider le dossier juridique.

3. Contraintes de sécurité :

- Protection des informations utilisateur : le site doit être protégé contre tout type d'attaque (injection SQL, XSS, phishing).
- Conformité aux réglementations RGPD et CNIL.

4. Contraintes temporelles :

- Temps d'obtention des autorisations officielles.
- Livraison impérative le 15 juillet 2024 au plus tard.

LE RÉFÉRENCEMENT SEO

Optimisation technique :

- Légèreté des pages
- Rapidité de chargement et d'affichage des pages
- Pratique du Server-Side Rendering au maximum
- Une structure du site facilitant la navigation. (le fil d'Ariane)
- Utilisation des balises sémantiques.
- Utilisation de mots-clés
- Optimisation des requêtes (HTTP et SQL)
- Optimisation des images

Promotion du site web :

- Sur les réseaux sociaux (Facebook, LinkedIn ...)
- Bouche à oreille
- Possible publicité

LES RESSOURCES HUMAINES NÉCESSAIRES

Lead Dev :

- Responsable de la direction technique du projet, du choix des technologies et des procédures à suivre.
- Tarif journalier moyen : 600 euros.

Développeur Full Stack :

- Chargé du développement back-end et front-end, maîtrisant Express-JS, React et Sass.
- Tarif journalier moyen : 400 euros.

Designer/Intégrateur :

- Création de la maquette.
- Responsable de l'intégration des contenus dans le site internet conformément aux maquettes.
- Tarif journalier moyen : 500 euros.

Administrateur Système et Serveur :

- Gère l'infrastructure technique, assure le bon fonctionnement des serveurs et la sécurité.
- Tarif journalier moyen : 400 euros.

Testeur :

- Responsable des tests techniques du projet et de la rédaction des rapports de bugs.
- Tarif journalier moyen : 150 euros.

Avocat :

- Chargé des aspects juridiques et des autorisations.
- Tarif journalier moyen : 400 euros.

Documentaliste :

- Rédige la documentation du produit final.
- Tarif journalier moyen : 100 euros.

RÉTRO-PLANNING

		Semaine 1	Semaine 2	Semaine 3	Semaine 4	Semaine 5	Semaine 6	Semaine 8	Semaine 9
		27 au 31 mai	3 au 7 juin	10 au 14 juin	17 au 21 juin	24 au 28 juin	1 au 5 juillet	8 au 12 juillet	15 au 19 juillet
Ressource	Mission								
Tout le monde	Préparation du projet								
Lead dev	Gestion du projet								
Admin système/server	Supervision technique								
Intégrateur/Designer	Création de la maquette								
Full stack dev	Création MCD et base de données								
Full stack dev	Création de l'architecture								
Full stack dev	Création des routes API								
Full stack dev	Sécurisation authentification								
Intégrateur/Designer	Reproduction de la maquette (CSS)								
Full stack dev	Réalisation du front								
Intégrateur/Designer	Réalisation du front								
Testeur	Premiers tests								
Full stack dev	Corrections de bugs								
Documentaliste	création de la documentation								
Testeur	Tests finaux								
Full stack dev	Corrections de bugs								
Full stack dev	Déploiement de l'application								
Admin système/server	Déploiement de l'application								
Avocat	Gestion juridique								
Tout le monde	Livraison								

LE BUDGET

Acteur	Jours de travail	TJM	Total
--------	------------------	-----	-------

Lead dev	15	600,00 €	9000,00 €
Full stack dev	35	400,00 €	14 000,00 €
Designer/Intég.	15	500,00 €	7500,00 €
Admin sys/server	10	400,00 €	4000,00 €
Testeur	10	150,00 €	1500,00 €
Avocat	15	400,00 €	6000,00 €
Frais + Imprévus			7000,00 €

			40 000,00 €
--	--	--	-------------

RÉGLEMENTATION GÉNÉRALE DE LA PROTECTION DES DONNÉES

Dans le cadre de notre conformité au RGPD, nous nous engageons à :

- Informer les utilisateurs de manière claire et concise sur la manière dont leurs données personnelles seront collectées, traitées et utilisées.

- Obtenir le consentement explicite des utilisateurs avant de collecter leurs données personnelles et leur fournir la possibilité de retirer ce consentement à tout moment.
- Ne collecter que les données personnelles nécessaires aux fins spécifiques pour lesquelles elles sont traitées et garantir qu'elles sont utilisées uniquement à ces fins.
- Ne pas conserver les informations personnelles après la désinscription ou la suppression du compte.
- Mettre en place des mesures de sécurité appropriées pour protéger les données personnelles contre toute perte, accès non autorisé, divulgation ou altération.
- Ne pas transférer les données personnelles des utilisateurs à des fins commerciales

LA SÉCURITÉ

Les validations des données doivent se faire dans le front et dans le backend.

En ce qui concerne les mots de passe, il est essentiel d'imposer des exigences de complexité afin de renforcer la sécurité des comptes utilisateurs. Cela inclut la demande d'un minimum de caractères (au moins 8 caractères), avec une combinaison de lettres majuscules, minuscules, chiffres et caractères spéciaux.

Pour protéger contre les attaques telles que les injections SQL et les attaques XSS, il est crucial de mettre en œuvre des mesures de sécurité au niveau de l'application.

Les injections SQL sont des attaques où un attaquant insère des commandes SQL malveillantes dans les champs de saisie d'un formulaire ou d'une URL, dans le but de manipuler ou de compromettre la base de données de l'application.

Pour se protéger contre les injections SQL, il est important de mettre en place certaines pratiques comme limiter les privilèges d'accès à la base de données pour que les utilisateurs n'aient que les permissions nécessaires à leurs tâches et utiliser des outils de sécurité web tels que les pare-feux d'application web (WAF) pour détecter et bloquer les tentatives d'injection SQL.

Les attaques XSS sont des attaques où un attaquant injecte du code JavaScript malveillant dans les pages web d'une application, généralement en exploitant des failles de sécurité dans les entrées utilisateurs non validés.

Pour se protéger contre les attaques XSS, voici quelques mesures à prendre :
Valider et échapper correctement toutes les données d'entrée utilisateur pour empêcher l'exécution de scripts non autorisés;

Utiliser des bibliothèques ou des frameworks sécurisés pour la génération dynamique de contenu HTML, tels que ReactJS ou AngularJS, qui offrent des mécanismes de protection intégrés contre les attaques XSS.

Mettre en œuvre des stratégies de sécurité au niveau du navigateur, telles que l'utilisation de l'en-tête HTTP **Content-Security-Policy (CSP)** pour limiter les sources autorisées des ressources chargées par la page.

Pour prévenir les attaques CSRF, l'utilisation de tokens CSRF est essentielle.

Ces tokens sont des valeurs uniques générées par le serveur et incluses dans les formulaires web. Lorsqu'un utilisateur soumet un formulaire, le token CSRF est

également envoyé. Le serveur peut alors vérifier si ce token correspond à celui qui lui a été initialement attribué, empêchant ainsi les attaques CSRF.

Pour sécuriser le processus **d'upload de fichiers** via un formulaire, les pratiques suivantes doivent être appliquées :

- **Vérification de l'extension du fichier** : Les extensions autorisées (jpg, png, jpeg, pdf, etc.) doivent être vérifiées avant l'upload. Tout fichier avec une extension non autorisée doit être rejeté.
- **Contrôle du poids du fichier** : Limiter le poids maximum des fichiers uploadés à une taille raisonnable, par exemple, 2 Mo. Les fichiers dépassant cette limite doivent être rejetés.
- **Renommage des fichiers** : Les fichiers uploadés doivent être renommés de manière aléatoire ou unique pour éviter les collisions de noms de fichiers et prévenir les attaques potentielles basées sur les noms de fichiers prévisibles.
- **Vérification du contenu** : Outre l'extension, le contenu réel des fichiers doit être vérifié pour s'assurer qu'il correspond au type de fichier attendu. Par exemple, un fichier d'extension "jpg" doit contenir des données image valides.

Il est recommandé de réaliser des sauvegardes régulières de la base de données et des fichiers du site dans le cloud. Les méthodes de sauvegarde doivent être robustes et fiables, assurant la disponibilité des données en cas de problème.

L'ACCESSIBILITÉ

Il est essentiel de s'assurer que les contrastes de couleurs respectent les normes d'accessibilité. Cela peut être vérifié à l'aide d'extensions Chrome telles que WAVE, AXE DevTools ou Accessibility Insights for the Web.

Il est également important de définir une palette de couleurs qui offre un contraste suffisant pour assurer une lisibilité optimale, tout en prenant en compte les besoins visuels de tous les utilisateurs.

Les couleurs choisis dans le cadre de notre projet seront :

- Pour le thème clair :

- background : #F7FDFB;
- text: #04100E;
- primary: #4CCDB1;
- secondary: #A294E1;
- accent: #B370D7;
- borders: rgba(0, 0, 0, 0.2);
- toast-info: #c08800;
- toast-error: #fc6060;
- toast-success: #12a112;
- danger: #cb0707;
- success: #12a112;

- Pour le thème sombre :

- background: #040500;
- texte: #FCFFF5;
- primary: #32B397;
- secondary: #2C1E6B;
- accent: #6B288F;
- borders: rgba(255, 255, 255, 0.2);

En ce qui concerne les éléments d'accessibilité, il est crucial de s'assurer que les attributs alt, aria-label et rôle sont correctement utilisés pour fournir des descriptions alternatives aux éléments visuels et faciliter la navigation pour les utilisateurs qui dépendent des technologies d'assistance. Cela garantit que le contenu du site est accessible à tous, y compris aux personnes malvoyantes.

Par ailleurs, **les titres doivent être clairs et bien organisés** pour faciliter la navigation et la compréhension du contenu, notamment pour les utilisateurs qui utilisent des lecteurs d'écran. Nous devons donc respecter l'ordre des balises titre (H1 suivi de H2 puis H3 ...). Une bonne utilisation des balises est primordiale pour que tout le monde puisse utiliser notre site Web, mais aussi pour le référencement de celui-ci.

Pour les polices de caractères, il est recommandé d'en utiliser des lisibles et facilement accessibles tels que ARIAL, HELVETICA, SANS SERIF, LUCIOLE ou autres sans empattement.

Dans le cadre de notre projet nous utiliserons :

- **Logo** : Jomhuria

<https://fonts.googleapis.com/css?family=Jomhuria:700|Jomhuria:400>

- **Texte** : Inter

(<https://fonts.googleapis.com/css?family=Inter:700|Inter:400>)

Pour les icônes :

- Lucid Dev - <https://lucide.dev/>
- FontAwesome - <https://fontawesome.com/>

L'ADAPTABILITÉ DU SITE

Responsive Design :

- Le site sera conçu en utilisant des **techniques de responsive design** pour garantir une expérience utilisateur optimale sur une variété d'appareils, y compris les ordinateurs de bureau, les tablettes et les smartphones.
- **Les éléments de la page, tels que la mise en page, les images et les menus, seront redimensionnés et réorganisés automatiquement** en fonction de la taille de l'écran de l'appareil utilisé, afin d'assurer une lisibilité et une utilisation efficace du site.
- **Points de rupture :**
 - Smartphone: max-width 350px;
 - Tablette : max-width 900px;
 - Bureau : min-width 901px;

Compatibilité avec les Navigateurs :

- Le site sera **testé et optimisé pour assurer une compatibilité avec les principaux navigateurs web**, tels que Google Chrome, Mozilla Firefox, Safari et Microsoft Edge.
- **Des tests seront effectués pour garantir que toutes les fonctionnalités du site fonctionnent correctement** et que l'affichage est cohérent sur chaque navigateur, afin d'offrir une expérience utilisateur homogène quel que soit le navigateur utilisé.
- **Le W3C Validator** est un outil essentiel pour vérifier la conformité de notre code HTML et CSS aux normes du World Wide Web Consortium (W3C). En utilisant ce validateur, nous nous assurons que notre site respecte les standards du web, ce qui améliore l'accessibilité, l'interopérabilité et la performance de notre application. Cela permet également de détecter et de corriger les erreurs de code qui pourraient affecter le rendu et la fonctionnalité du site sur différents navigateurs et appareils. <https://validator.w3.org/>
- Utilisation du site <https://caniuse.com/> afin de s'assurer de la compatibilité des navigateurs.

ECO-CONCEPTION

Dans le cadre de notre projet, **nous nous engageons à adopter des pratiques éco-responsables** pour réduire l'empreinte environnementale de notre application. Voici comment nous envisageons d'intégrer des principes d'éco-conception :

1. Choix de l'hébergement :

- Nous privilégierons des services d'hébergement qui utilisent des **sources d'énergie renouvelable** et qui sont certifiés pour leur **faible consommation énergétique**.

2. Optimisation du code :

- Nous développerons le code de manière à **minimiser la consommation de ressources système**, en utilisant des pratiques de programmation efficaces et en évitant le surdimensionnement inutile des fonctionnalités.
- **Valider les données en amont** si possible afin d'éviter des requêtes inutiles au serveur.
- **Système de mise en cache** afin de limiter les appels au serveur.

3. Gestion des médias :

- Nous limiterons l'utilisation de médias lourds tels que les images haute résolution et les vidéos, en favorisant des formats plus légers lorsque cela est possible. De plus, nous envisageons des techniques de chargement paresseux pour optimiser le chargement des médias.

4. Utilisation de technologies éco-responsables :

- Nous choisirons des technologies et des bibliothèques de développement qui sont connues pour leur efficacité énergétique et leur faible impact sur l'environnement.

En intégrant ces pratiques d'éco-conception dans notre processus de développement, nous visons à réduire l'empreinte carbone de notre application et à contribuer à un environnement plus durable.

LES SPÉCIFICATIONS TECHNIQUES

Langages de Programmation :

- **HTML** : Nous utiliserons des balises sémantiques et la page devra être bien structurée.
- **CSS** : Nous utiliserons Sass comme préprocesseur CSS pour faciliter le développement et la maintenance des feuilles de style. Sass nous permettra d'utiliser des fonctionnalités avancées telles que les variables, les mixins et les fonctions, ce qui nous aidera à créer des styles modulaires et réutilisables.
- **Backend** : Nous utiliserons **Node.js** avec **ExpressJS** comme framework pour le développement du backend. ExpressJS offre une grande flexibilité et une gestion efficace des requêtes HTTP, ce qui en fait un choix idéal pour la construction d'APIs robustes et performantes.
- **Frontend** : Pour le développement du frontend, nous utiliserons Vite et React. Vite nous permettra d'ouvrir un serveur pour faire tourner React qui nous permettra de construire une interface utilisateur dynamique et réactive, en utilisant des composants réutilisables pour organiser et structurer notre application.
- **Modules complémentaires** : Nodemon, Cors, Dotenv, Express-session, Express-mysql-session, mysql2/promise, React-Router

Base de Données :

- Nous utiliserons MySQL comme système de gestion de base de données relationnelle pour stocker et gérer les données de notre application. MySQL est largement utilisé, stable et bien documenté, ce qui en fait un choix fiable pour nos besoins de stockage de données.
- Nous ferons nos requêtes sans ORM afin d'alléger notre projet.

Sécurité :

- Nous utiliserons également l'**algorithme de hachage bcrypt** pour stocker de manière sécurisée les mots de passe des utilisateurs dans la base de données.
- **Utilisation de middlewares** pour la sécurisation de certaines routes.

- **Validation des données côté client ET côté serveur à l'aide d'expressions régulières.**

Déploiement et Gestion :

- **Pour la gestion du code source**, nous utiliserons **Git**, un système de contrôle de version, avec un dépôt hébergé sur la plateforme **GitHub**.

En suivant ces spécifications techniques, nous serons en mesure de développer une application **robuste, sécurisée et performante**, répondant aux besoins de nos utilisateurs de manière efficace et fiable.

VEILLES TECHNIQUES

Pour garantir que notre application reste à la pointe de la technologie et respecte les meilleures pratiques en matière de développement web, nous nous engageons à effectuer une veille technique régulière. Voici les outils et les ressources que nous utiliserons dans ce cadre :

Réseaux professionnels et plateformes de publication :

- **LinkedIn** : LinkedIn est une ressource précieuse pour rester informé des dernières tendances et innovations dans le domaine du développement web. Nous suivrons des leaders d'opinion, des groupes de discussion et des pages de technologies pour accéder à des articles, des études de cas, et des mises à jour sur les nouvelles technologies et pratiques de développement.
- **Medium** : Medium est une plateforme de publication où des experts et des développeurs partagent leurs connaissances et expériences. En suivant des publications et des auteurs influents dans le domaine du développement web, nous pourrons nous tenir au courant des nouvelles techniques, des frameworks émergents et des meilleures pratiques de l'industrie.

Sécurité des applications web :

- **OWASP (Open Web Application Security Project)** : OWASP est une communauté en ligne dédiée à la sécurité des applications web. Nous consultons régulièrement les ressources proposées par OWASP, telles que l'OWASP Top Ten, qui liste les dix principales failles de sécurité des applications web. En intégrant ces recommandations dans notre processus de développement, nous renforcerons la sécurité de notre application en identifiant et en corrigeant les vulnérabilités potentielles, telles que les injections SQL, les scripts inter sites (XSS) et les problèmes de gestion des sessions.

En intégrant ces outils et ressources dans notre veille technique, nous nous assurons de développer une application conforme aux standards du web, à la pointe des dernières tendances technologiques et sécurisée contre les menaces courantes. Cette approche proactive nous permettra d'offrir à nos utilisateurs une expérience optimale et sécurisée.

LES OUTILS COLLABORATIFS

Pour assurer une collaboration efficace au sein de notre équipe de développement, nous utiliserons une variété d'outils collaboratifs. Ces outils nous aideront à gérer le code source, à concevoir l'interface utilisateur, à communiquer et à organiser notre travail en groupe.

Voici les principaux outils que nous utiliserons :

Gestion du code source :

- **Git et GitHub** :
 - **Git** est un système de contrôle de version décentralisé qui nous permettra de suivre les modifications apportées au code source de notre application. Il facilitera la collaboration entre les développeurs en permettant des branches et des fusions sans conflit.

- **GitHub** est une plateforme de gestion de projets basée sur Git. Elle offre des fonctionnalités supplémentaires telles que les pull requests, les revues de code, les issues et les wikis. GitHub nous permettra de collaborer efficacement, de suivre les bugs et les fonctionnalités à développer, et de documenter notre projet.

Conception de l'interface utilisateur :

- **Figma** :
 - Figma est un outil de conception d'interface utilisateur en ligne qui permet de créer des maquettes et des prototypes interactifs. Il supporte la collaboration en temps réel, permettant aux designers et aux développeurs de travailler ensemble sur les mêmes fichiers. Figma facilite également la communication des idées de design et la réception de feedback instantané.

Communication et réunions :

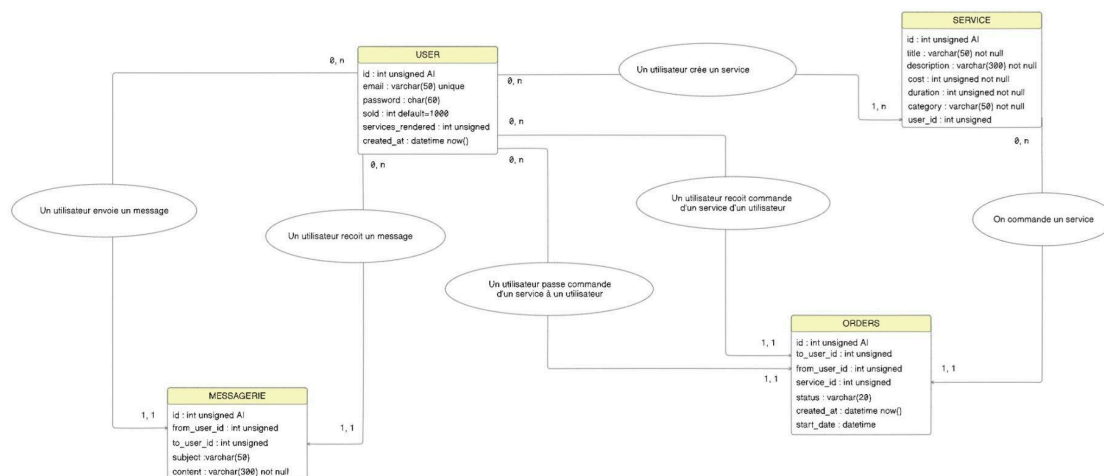
- **Discord** :
 - **Discord** est une plateforme de communication qui offre des canaux textuels et vocaux. Elle sera utilisée pour les discussions quotidiennes, les mises à jour de projet et la résolution rapide des problèmes. Les canaux thématiques permettront de structurer les discussions par sujet ou par équipe.
- **Zoom** :
 - **Zoom** est un outil de visioconférence qui sera utilisé pour les réunions d'équipe, les démonstrations de projet et les sessions de travail en direct. **Zoom** offre des fonctionnalités de partage d'écran, d'enregistrement des réunions et de collaboration sur un tableau blanc virtuel.

Gestion de projet :

- **Trello :**
 - **Trello** est un outil de gestion de projet basé sur des tableaux **Kanban**. Il permet de créer des cartes pour les tâches, de les organiser en listes et de les déplacer entre différentes étapes du flux de travail. **Trello** facilite la visualisation de l'état d'avancement des tâches, la répartition du travail entre les membres de l'équipe et la gestion des priorités. Avec des fonctionnalités telles que les dates d'échéance, les étiquettes et les checklists, **Trello** nous aidera à rester organisés et à respecter les délais du projet.

En utilisant ces outils collaboratifs, nous assurons une communication fluide, une gestion efficace des tâches et une collaboration optimale entre les membres de l'équipe. Cela nous permettra de développer notre application de manière cohérente et organisée, en respectant les délais et les objectifs fixés.

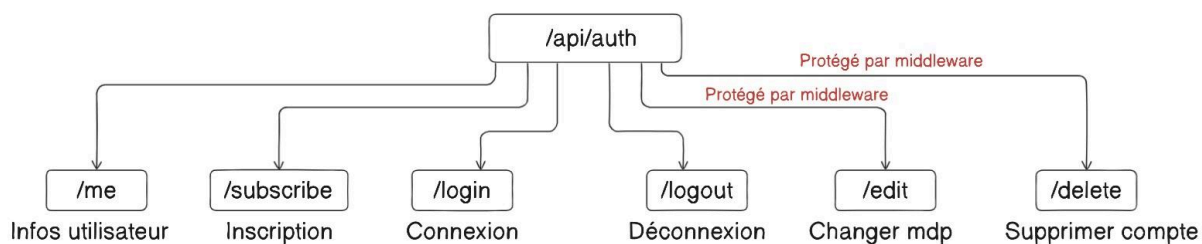
LE MODÈLE CONCEPTUEL DE DONNÉES



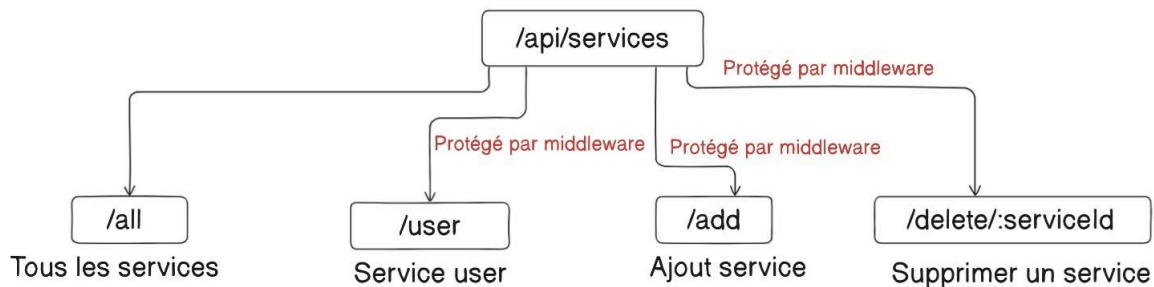
L'ARBORESCENCE DU SITE

Back-end (API Express)

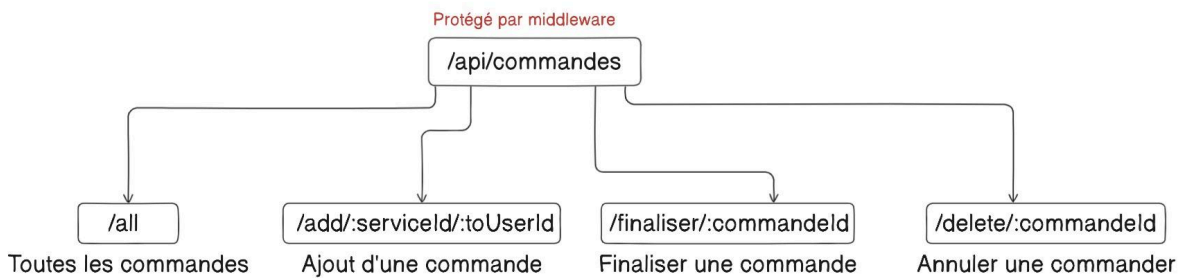
Authentification



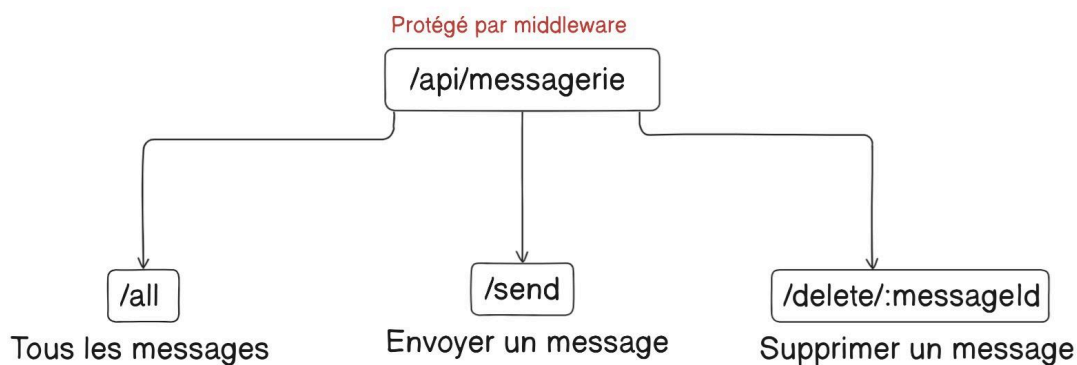
Services



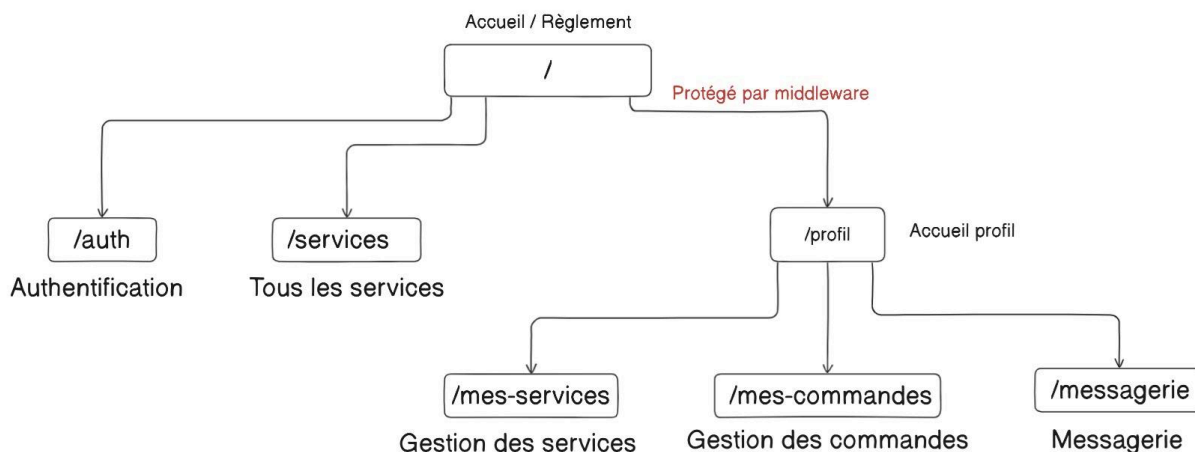
Commandes



Messagerie



Front-end (React Router)



AMELIORATIONS FUTURES

Pour assurer la croissance et l'amélioration continue de notre site d'échange de services, nous envisageons d'ajouter plusieurs fonctionnalités futures. Ces fonctionnalités visent à améliorer l'expérience utilisateur, à encourager l'engagement et à diversifier les offres de services disponibles.

Voici quelques-unes des fonctionnalités que nous prévoyons d'intégrer :

- **Bonus pour les meilleurs utilisateurs :**
 - Mettre en place un système de récompenses pour les utilisateurs les plus actifs et les mieux notés. Les utilisateurs pourront gagner des bonus ou des badges spéciaux pour leurs contributions et leur engagement sur la plateforme. Ces récompenses pourront être échangées contre des services gratuits ou des avantages exclusifs.
- **Abonnements à un ou plusieurs services :**
 - Offrir des options d'abonnement pour accéder à des services premium ou à des packages de services. Les utilisateurs pourront s'abonner à des services spécifiques pour une durée déterminée, bénéficiant ainsi d'avantages supplémentaires tels que des réductions, un accès prioritaire ou des fonctionnalités exclusives.
- **Évaluations et avis :**
 - Ajouter un système de notation et de commentaires pour les services rendus. Les utilisateurs pourront évaluer la qualité des services reçus et laisser des avis détaillés. Cela aidera à établir la réputation des prestataires de services et à guider les choix des futurs utilisateurs.
- **Recherche avancée et filtres :**
 - Améliorer la fonctionnalité de recherche en permettant aux utilisateurs de filtrer les résultats par catégorie, localisation, disponibilité et autres critères pertinents. Cela facilitera la recherche de services spécifiques et l'identification rapide des prestataires disponibles.
- **Notifications et alertes :**
 - Mettre en place un système de notifications pour informer les utilisateurs des nouvelles demandes de services, des réponses à leurs offres, des messages reçus et des mises à jour importantes. Les utilisateurs pourront personnaliser leurs préférences de notification pour rester informés en temps réel.
- **Intégration avec les réseaux sociaux :**
 - Permettre aux utilisateurs de se connecter et de partager leurs activités sur les réseaux sociaux tels que Facebook, LinkedIn et Twitter. Cette fonctionnalité augmentera la visibilité de la plateforme et encouragera davantage d'utilisateurs à rejoindre la communauté.

En intégrant ces fonctionnalités futures, nous visons à créer une plateforme d'échange de services dynamique, intuitive et enrichissante pour tous les utilisateurs. Ces améliorations continueront de faire évoluer notre service en répondant aux besoins et aux attentes de notre communauté.