

# SUPA SUSHI

COMPANY PRESENTATION

# INTRODUCTION

01 Présentation Générale

02 Public cible

03 Objectifs et besoins

04 Fonctionnalités attendues

05 Spécifications techniques

06 Design et UX

07 Contraintes

08 Référencement SEO

09 Accessibilité

10 Eco-conception

11 Équipe du projet

12 Planning et budget

13 Outils collaboratifs

14 Sécurité

15 Veille technique

16 Améliorations futures

17 Annexes

# 1. Présentation Générale

Le projet consiste à développer un site web pour un restaurant de sushi. Ce site aura pour but de présenter le restaurant, son menu, ses spécialités, et de permettre aux clients de passer des commandes en ligne, et d'interagir avec le restaurant via un blog ou une newsletter. Le site doit refléter l'image de marque du restaurant, en mettant en avant la qualité des produits, l'authenticité de la cuisine japonaise, et une expérience utilisateur agréable.

## **Nom et objectif :**

Projet "Supa Sushi" – Développer un site web pour présenter le restaurant, son menu, ses spécialités, et permettre aux clients de commander en ligne, et interagir via un blog ou une newsletter.

## **Origine et contexte :**

Né de la nécessité de s'adapter aux habitudes des consommateurs (commandes et réservations en ligne), dans un marché concurrentiel où la digitalisation est essentielle.

## **Valeur ajoutée :**

Design épuré inspiré de la culture japonaise, mettant en avant la qualité et l'authenticité des produits. Fonctionnalités avancées (réservation en temps réel, blog interactif, newsletter) et optimisation SEO/accessibilité pour une meilleure visibilité et expérience utilisateur.

## 2. Public Cible

Tranche d'âge et profils d'utilisateurs :

- Jeunes adultes (18-35 ans) : Habités aux commandes en ligne et aux réseaux sociaux.
- Adultes (35-55 ans) : Recherche de qualité et d'authenticité, souvent en famille ou entre amis.
- Touristes : En quête de découvertes culinaires locales.
- Professionnels : Recherche de lieux pour déjeuners d'affaires ou événements.

Préférences ou besoins spécifiques :

- Technologie : Accès via mobile, tablette ou ordinateur.
- Attente d'une interface intuitive et rapide.
- Accessibilité : Site adapté aux personnes handicapées (normes WCAG 2.1).
- Besoins : Commandes faciles, réservations rapides, et informations claires (menu, horaires, localisation).



# 3. Objectifs et Besoins

## Objectifs à court terme :

Augmenter la visibilité du restaurant en ligne.  
Faciliter les commandes et réservations via une plateforme intuitive.  
Améliorer l'engagement client grâce à un blog et une newsletter.

## Objectifs à long terme :

Fidéliser la clientèle grâce à des fonctionnalités interactives et des promotions ciblées.  
Devenir une référence locale pour les amateurs de sushi.  
Augmenter le chiffre d'affaires via les commandes en ligne et les réservations.

## Problèmes à résoudre et solutions proposées :

Problème : Manque de visibilité en ligne.  
Solution : Site optimisé SEO et présence sur les annuaires locaux.

Problème : Processus de commande et réservation peu pratique.  
Solution : Interface utilisateur simplifiée et système de réservation en temps réel.

Problème : Difficulté à fidéliser les clients.  
Solution : Blog interactif, newsletter personnalisée et programme de fidélité.

## Indicateurs de succès :

Augmentation du trafic sur le site (mesuré via Google Analytics).  
Nombre de commandes et réservations en ligne.  
Taux de rétention des clients (abonnés à la newsletter, participants au programme de fidélité).



## 4. Fonctionnalités Attendues

Utilisateurs	Fonctionnalités
Anonymes	Consulter le menu, créer un compte, voir les promotions.
Inscrits	Commander, ajouter au panier ou modifier le panier, laisser des avis.
Administrateurs	Gérer les stocks, mettre à jour le menu, générer des rapports de ventes.
Fonctionnalités clés	Paiement en ligne (Stripe/PayPal)

# 5. Spécifications Techniques



## BACK-END

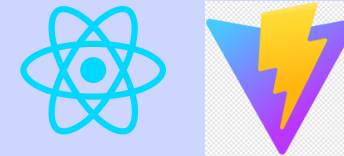
Environnement d'exécution JavaScript côté serveur.

### Avantages :

- Non-bloquant et asynchrone (idéal pour les I/O lourds).
- Partage de code entre front-end et back-end (ex : validation).
- Écosystème NPM (modules prêts à l'emploi).

### Choix :

Permet un stack full JavaScript, contrairement à Python (Django) ou Ruby (Rails).  
Meilleure performance que PHP pour les apps temps réel.



## FRONT-END

React : Librairie JavaScript pour construire des interfaces utilisateur composables.  
Vite : Outil de build ultra-rapide pour le développement moderne.

### Avantages :

- Composants réutilisables et état managé efficacement (ex : hooks).
- Virtual DOM pour des performances optimisées.

### Choix :

- React : Plus flexible et populaire que Vue/Angular, idéal pour les projets évolutifs.
- Vite : Alternative plus performante à Create React App (CRA) ou Webpack.



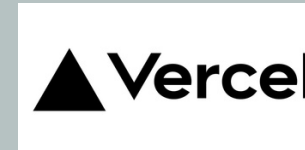
Système de gestion de base de données relationnelle (SGBDR).

### Avantages :

- Structure claire avec schémas et relations (tables, clés étrangères).
- Transactions ACID pour la fiabilité des données.
- Scalable et largement supporté (ex : AWS RDS).

### Choix :

Préféré à MongoDB (NoSQL) quand les données sont structurées et nécessitent des jointures complexes.



Plateforme de déploiement et d'hébergement pour applications front-end.

### Avantages :

- Intégration Git continue (déploiement automatique).
- Serverless Functions pour des APIs légères.
- CDN global et optimisations de performance automatiques.

### Choix :

Meilleure expérience pour les projets React/Next.js que Netlify ou Heroku. Gratuit pour les petits projets.

## Pourquoi ce stack ?

- Cohérence JavaScript : Un seul langage (JS) pour le front-end, back-end et les outils.
- Productivité : React + Vite accélèrent le développement, Node.js simplifie le full-stack.
- Scalabilité : MySQL et Node.js gèrent bien les montées en charge.
- Coût/Performance : Vercel et MySQL (via des solutions cloud) sont économiques pour les MVP.

## Compatibilité

### **Compatibilité navigateurs :**

Assurer la compatibilité avec Chrome, Firefox, Safari et Edge est essentiel pour toucher une audience diversifiée, garantir un rendu cohérent des fonctionnalités, éviter les bugs spécifiques à certains navigateurs, et respecter les exigences SEO et d'accessibilité, sous peine d'exclure des utilisateurs ou de dégrader l'image de marque.

### **Responsive design :**

Adapter l'interface aux mobiles, tablettes et desktop permet de répondre à l'usage majoritaire du trafic mobile, d'optimiser l'expérience utilisateur (lisibilité, interactions tactiles), de respecter le mobile-first indexing de Google, et d'éviter des coûts supplémentaires liés à des versions séparées.

### **Performance :**

Atteindre un chargement en moins de 3 secondes est crucial pour retenir l'attention des utilisateurs (évitant jusqu'à 53% d'abandons), améliorer le référencement (SEO) via les Core Web Vitals, et réduire les coûts d'infrastructure grâce à des techniques comme le code splitting et la mise en cache.

### **Optimisation des images et ressources:**

Compresser les images (en WebP/AVIF), appliquer le lazy loading et adapter les médias aux écrans permet de réduire jusqu'à 70% du poids des pages, diminuer le taux de rebond, et économiser des données pour les utilisateurs mobiles, tout en respectant les bonnes pratiques écologiques.

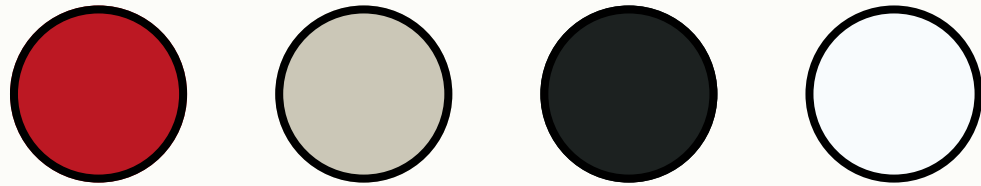


# 6. Design et UX

## Palette de couleurs et typographie :

- Palette de couleurs :

Inspirée de la culture japonaise : noir, blanc, rouge, et touches de beige.



- Typographie :

- Polices modernes et épurées.
- Hiérarchie claire (titres, sous-titres, texte) pour une lecture aisée.

## Wireframes et maquettes :

- Wireframes :

- Structure simplifiée des pages principales (accueil, menu, contact).
- Mise en place des zones clés (en-tête, pied de page, call-to-action).

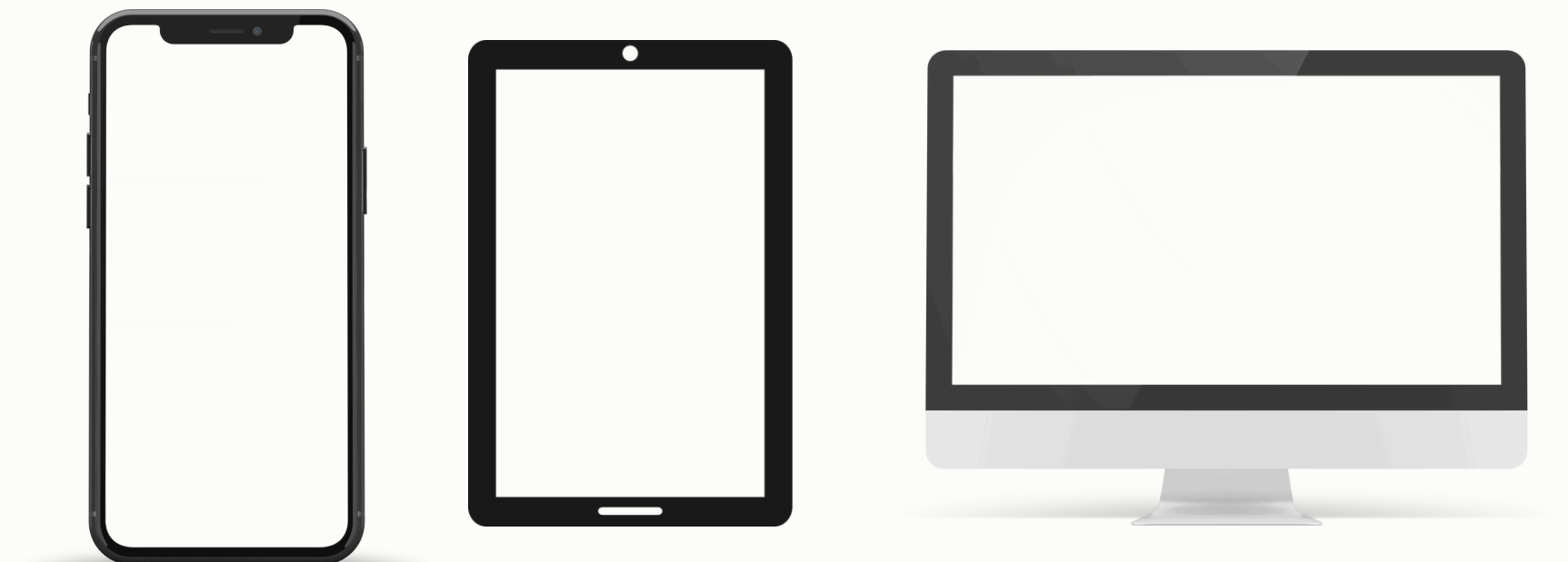
## Design responsive et accessibilité :

- Responsive design :

- Adaptation du site à tous les appareils (mobile, tablette, desktop).
- Menus et boutons optimisés pour les écrans tactiles.

- Accessibilité :

- Respect des normes WCAG 2.1 (niveau AA).
- Contraste élevé, alternatives textuelles pour les images, navigation au clavier.



# 7. Contraintes

## **Contraintes Techniques :**

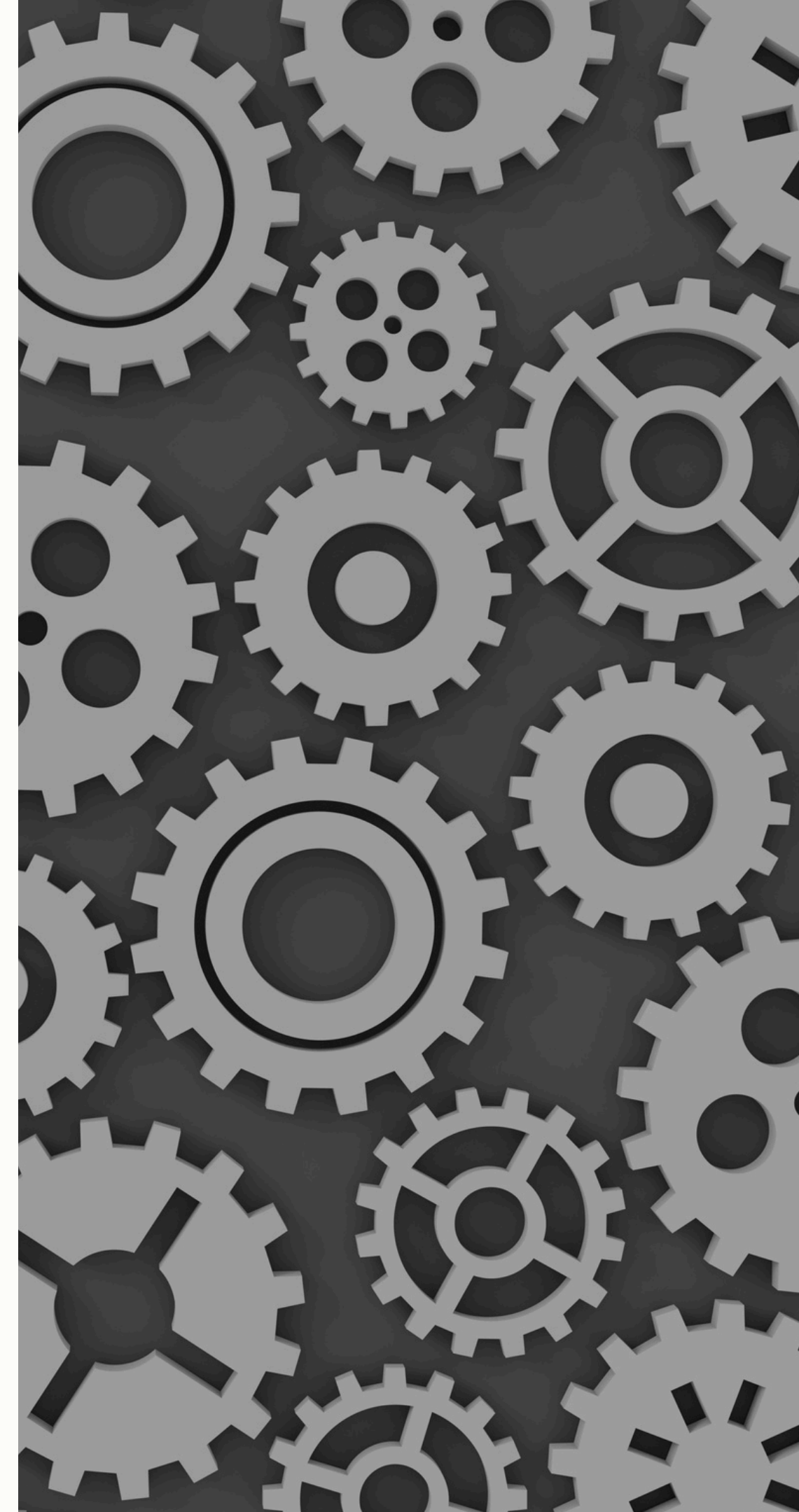
- Respect des normes W3C pour le code HTML/CSS.
- Compatibilité avec les appareils mobiles.

## **Contraintes de Temps :**

- Livraison du site dans un délai de 6 semaines.

## **Contraintes Budgétaires :**

- Budget total du projet : 10 000 €.



# 8. Référencement SEO

## Définition :

Le SEO est l'ensemble des techniques visant à améliorer la visibilité d'un site web dans les résultats organiques (non payants) des moteurs de recherche (Google, Bing, etc.). L'objectif est d'attirer un trafic qualifié en répondant aux requêtes des utilisateurs et aux critères d'algorithmes.

- Stratégie SEO :
  - Optimisation des balises meta (title, description).
  - Utilisation de mots-clés pertinents (ex : "sushi Oyonnax", "commande de sushi en ligne").
  - Création de contenu régulier sur le blog pour améliorer le référencement naturel.
  - Soumission du site aux annuaires locaux (Google My Business, TripAdvisor).
- Performance SEO :
  - Audit technique SEO avant la mise en ligne.
  - Suivi des performances via Google Analytics et Search Console.

## Pourquoi ces pratiques ?

SEO Local : Ciblez une audience géographique précise.

Contenu frais : Google privilégie les sites mis à jour régulièrement.

Technique : Un site rapide et sans erreurs est mieux indexé.



# 9. Accessibilité



L'accessibilité web vise à rendre un site utilisable par tous les utilisateurs, y compris ceux en situation de handicap (visuel, auditif, moteur, cognitif). Cela inclut le respect des normes internationales (WCAG) et l'adaptation aux technologies d'assistance (lecteurs d'écran, claviers adaptés, etc.).

- Respect des normes :
  - Conformité aux normes WCAG 2.1 (niveau AA) et RGAA pour garantir l'accessibilité des contenus.
- Typographie :
  - Taille de police adaptable (zoom sans perte de fonctionnalité).
- Contraste des couleurs :
  - Contrastes suffisants entre le texte et l'arrière-plan pour une lisibilité optimale.
- Descriptions alternatives :
  - Ajout d'attributs "alt" pour toutes les images, permettant une compréhension par les lecteurs d'écran.
- Navigation intuitive :
  - Structure logique des balises HTML (titres H1, H2, etc.) pour une navigation claire.
  - Possibilité de naviguer au clavier (sans souris).
- Mode clair/sombre :
  - Intégration d'un mode clair/sombre pour adapter l'affichage aux préférences des utilisateurs.
- Autres fonctionnalités :
  - Boutons et liens descriptifs pour une compréhension immédiate.

# 10. Eco-conception



L'éco-conception en développement (ou éco-conception logicielle) est une approche qui vise à réduire l'impact environnemental des produits ou services numériques tout au long de leur cycle de vie, depuis leur conception jusqu'à leur fin de vie. Elle intègre des principes de durabilité et d'efficacité dans le processus de développement logiciel, en tenant compte des aspects énergétiques, matériels et fonctionnels.

## Éléments clés :

- Architecture en MVC pour charger uniquement les éléments nécessaires.
- Optimisation des médias (images au format WebP, réduction de taille sans perte de qualité).
- Limitation des polices, bibliothèques, et API externes inutiles.
- Combinaison et minimisation des fichiers pour réduire les requêtes HTTP.
- Réduction de la charge serveur en favorisant des requêtes optimisées et des jointures en base de données.



# 11. Équipe du projet

Principaux rôles au sein d'une équipe de projet web :

1. **Chef de projet** : Responsable de la coordination globale, de la planification et de la communication entre les membres de l'équipe et les parties prenantes.
2. **Développeur Front-End** : S'occupe de l'interface utilisateur (UI) en codant les aspects visuels et interactifs à l'aide de technologies comme HTML, CSS, et JavaScript.
3. **Développeur Back-End** : Gère la logique serveur, les bases de données et l'intégration des fonctionnalités du site web.
4. **Désigner UI/UX** : Conçoit l'aspect visuel et l'expérience utilisateur pour garantir un site ergonomique, esthétique et fonctionnel.
5. **Responsable SEO/Marketing Digital** : Optimise le site pour les moteurs de recherche et met en place des stratégies pour améliorer sa visibilité et ses performances.
6. **Testeur QA (Quality Assurance)** : Vérifie que le site est fonctionnel, exempt de bugs, et conforme aux spécifications.
7. **DevOps/Administrateur Systèmes** : Assure le déploiement, la gestion des serveurs, et la disponibilité du site en production.

# 12. Planning et budget



ROLE	Allocation budgétaire	Etapas	Détails des coûts
Chef de projet	1 000 €	Semaine 1-2	Chef de projet : 1 000 € pour la gestion globale du projet.
Designer UI/UX	1 500 €	Semaine 1-2	Designer UI/UX : 1 500 € pour la conception des maquettes et l'optimisation de l'expérience utilisateur.
Développeur Front-End	2 000 €	Semaine 3-4	Développeur Front-End : 2 000 € pour l'intégration des interfaces et la responsivité.
Développeur Back-End	2 500 €	Semaine 3-4	Développeur Back-End : 2 500 € pour la logique métier, les API et la base de données.
Responsable SEO/Marketing	1 000 €	Semaine 5	Responsable SEO/Marketing : 1 000 € pour l'optimisation du référencement et la stratégie de visibilité.
Testeur QA	1 000 €	Semaine 5	Testeur QA : 1 000 € pour les tests et la garantie de la qualité.
DevOps/Admin Systèmes	1 000 €	Semaine 6	DevOps/Admin Systèmes : 1 000 € pour le déploiement et la maintenance.

# Retro-Planning

# SupaSushi

# PLANNING D'ÉQUIPE

## MOIS : FEVRIER/MARS

# PÔLE

# Semaine

03/02/2025 - 07/02/25

## Semaine 2

10/02/2025 - 14/02/25

## Semaine 3

17/02/2025 - 21/02/25

## Semaine 4

24/02/2025 - 28/02/25

## Semaine 5

03/03/2025 - 07/03/25

## Semaine 6

10/03/2025 - 14/03/25

## CHEF DE PROJE

## DESIGNER UI/UX

# DÉVELOPPEUR FRONT-END

## DÉVELOPPEUR BACK-END

RESPONSABLE

## SEO/MARKETING

## TESTEUR QA

# DEVOPS/ADMIN SYSTÈMES

## LÉGENDE

OFF

## PRODUCTION

# 13. Outils collaboratifs

Les outils collaboratifs dans le développement sont des logiciels ou plateformes qui facilitent le travail d'équipe en permettant aux développeurs, designers, chefs de projet et autres acteurs de collaborer efficacement sur un projet. Ces outils aident à améliorer la communication, la coordination, la gestion des tâches et le suivi des progrès, tout en centralisant les ressources et en optimisant les workflows.

- Gestion de projet : Utilisation d'outils comme Trello pour organiser les tâches, suivre leur progression et attribuer des responsabilités.
- Communication : Plateformes comme Discord pour des échanges rapides et Zoom pour des réunions régulières.
- Partage de fichiers : Services comme Google Drive et Dropbox pour stocker et échanger des documents et médias en toute sécurité.
- Versioning: GIT pour gérer les versions du code et collaborer efficacement sur les modifications.

# 14. Sécurité

La sécurité est un concept qui désigne l'ensemble des mesures, pratiques, techniques et processus mis en place pour protéger un système, une application, un réseau, des données ou des individus contre des menaces, des attaques, des dommages ou des accès non autorisés. Elle vise à garantir la confidentialité, l'intégrité et la disponibilité des ressources et des informations.

## Outils de sécurité :

- CORS (Cross-Origin Resource Sharing) : Contrôle des requêtes inter-domaines.
- express-validator : Validation et assainissement des données entrantes (Express.js).
- dotenv : Gestion sécurisée des variables d'environnement.
- Sécurité des cookies : Attributs HttpOnly, Secure, SameSite pour protéger contre XSS et CSRF.
- Validation des schémas (JOI) : Validation des données selon des schémas prédéfinis.
- Protection des routes (React-router-dom) : Restriction d'accès basée sur les autorisations.
- Sécurité des API (jsonwebtoken) : Authentification via JWT pour sécuriser les échanges client-serveur.

## Conclusion :

La sécurité est essentielle pour protéger les données, les utilisateurs et les systèmes. Une approche proactive et une mise à jour régulière des mesures sont nécessaires pour contrer les menaces.



# 15. Veille technique

La veille technique dans le domaine du web désigne l'ensemble des activités de surveillance, de recherche et d'analyse des nouvelles technologies, outils, frameworks, langages de programmation, tendances et bonnes pratiques liés au développement web.

## Recherche d'outils et solutions innovantes

- Exploration de nouvelles fonctionnalités : mode clair/sombre, animations interactives, accessibilité améliorée.
- Suivi des tendances via des plateformes comme Medium, W3C, GitHub, Smashing Magazine, et Dev.to.
- Participation à des conférences et événements technologiques : Google I/O, WWDC, DevFest, Web Summit, Meetup.com.
- Expérimentation avec de nouvelles bibliothèques et frameworks (ex. : Next.js, Svelte, Astro).

## Prévention des vulnérabilités

- Surveillance des menaces et mise à jour régulière des connaissances en cybersécurité (SQL Injection).
- Utilisation des ressources de l'OWASP (Top 10, Cheat Sheets) et d'outils spécialisés :
  - Pare-feux WAF (Web Application Firewall).
  - Content Security Policy (CSP) pour limiter les scripts malveillants.
  - Scanners de vulnérabilités (ex. : OWASP ZAP, Burp Suite, Snyk).
- Tests de sécurité réguliers : audits de code, tests de pénétration, validation des entrées utilisateur pour éviter les failles.
- Adoption des meilleures pratiques en cryptographie : HTTPS, gestion des tokens (JWT), hachage sécurisé des mots de passe (bcrypt).

# 16. Améliorations futures

Implémentation d'une fonctionnalité de commentaires pour les utilisateurs.

## **Objectif :**

Permettre aux utilisateurs de commenter des articles pour favoriser l'interaction, le partage d'idées et l'engagement sur la plateforme.

## **Fonctionnalités prévues :**

1. Système de commentaires :
  - Ajout d'un formulaire de commentaire sous chaque article.
2. Modération des commentaires :
  - Outil de modération pour éviter les contenus inappropriés.
3. Personnalisation :
  - Possibilité pour les utilisateurs de modifier ou supprimer leurs commentaires.
  - Affichage du nom de profil de l'utilisateur.

## **Avantages :**

- Engagement : Encourage les utilisateurs à interagir et à partager leurs opinions.
- Communauté : Favorise la création d'une communauté active et participative.
- Feedback : Permet aux auteurs de recueillir des retours constructifs sur leurs articles.