

**Menu Maker**

**by**

**Qwenta**

Version	Auteur	Date	Approbation
1.0	Kadija, Soufiane, Webgencia	28/08/2024	John, Qwenta

<a href="#">I. Choix technologiques</a>	2
<a href="#">II. Liens avec le back-end</a>	3
<a href="#">III. Préconisations concernant le domaine et l'hébergement</a>	3
<a href="#">IV. Accessibilité</a>	4
<a href="#">V. Recommandations en termes de sécurité</a>	4
<a href="#">VI. Maintenance du site et futures mises à jour</a>	4
<a href="#">VII. Technologies utilisées</a>	5

## I. Choix technologiques

- État des lieux des besoins fonctionnels et de leurs solutions techniques :

Besoin	Contraintes	Solution	Description de la solution	Justification (2 arguments)
<b>Création de menu dynamique</b>	Les restaurateurs doivent pouvoir ajouter/modifier des plats et les organiser par catégorie.	React + Redux	React pour la création d'interfaces dynamiques, Redux pour gérer l'état global de l'application.	1) React permet de créer des interfaces utilisateur interactives et réactives. 2) Redux centralise l'état de l'application pour une gestion optimale.
<b>Personnalisation du branding</b>	L'utilisateur doit pouvoir personnaliser la police, les couleurs, et ajouter un logo.	Styled-Components	Utilisation de Styled-Components pour des styles modulaires et réutilisables, et intégration facile des préférences utilisateurs.	1) Permet de styliser les composants React de manière encapsulée. 2) Facilite la réutilisation et la gestion des thèmes.
<b>Export en PDF</b>	Le menu doit pouvoir être exporté en PDF.	React-PDF	Génération de PDF directement à partir des composants React.	1) Intégration fluide avec React. 2) Permet un rendu précis des composants dans le format PDF.
<b>Connexion sécurisée des utilisateurs</b>	Les utilisateurs doivent pouvoir se connecter et gérer leurs menus en toute sécurité.	JWT	JSON Web Tokens (JWT) pour une gestion sécurisée des authentifications.	1) JWT est une solution sécurisée pour la gestion des sessions utilisateur. 2) Compatible avec l'API REST de Node.js.
<b>Stockage et gestion des données</b>	Les données des menus et utilisateurs doivent être stockées et accessibles de manière rapide et sécurisée.	MongoDB	Base de données NoSQL permettant de stocker des documents JSON.	1) Flexibilité pour gérer des données variées. 2) Performance et scalabilité pour les applications web modernes.
<b>API pour gestion côté serveur</b>	Communication avec Deliveroo et Instagram	Node.js/Express	Node.js avec Express pour consommer les API externes et communiquer efficacement avec des services tiers.	1) Communiquer facilement avec les prestataires 2) Node.js permet une exécution rapide et efficace du code JavaScript côté serveur.



## II. Liens avec le back-end

- Quel langage pour le serveur ? Node.js est un environnement d'exécution JavaScript qui permet de gérer un grand nombre de requêtes simultanées grâce à son modèle asynchrone. Il permet d'utiliser le même langage, JavaScript, pour le développement côté serveur et côté client, simplifiant ainsi le processus de développement.
- A-t-on besoin d'une API ? Si oui laquelle ? Pour l'API Deliveroo, nous avons besoin de la consommer pour récupérer des données comme les menus, les restaurants disponibles, et suivre les commandes en temps réel. De même, nous consommons l'API Instagram pour accéder aux contenus des utilisateurs (photos, vidéos) et permettre des interactions telles que les likes et commentaires, sans avoir à développer ces fonctionnalités nous-mêmes.
- Base de données choisie : MongoDB, une base de données NoSQL flexible qui permet de stocker des documents JSON, elle est idéale pour des données non structurées. Elle offre une scalabilité horizontale et des performances élevées pour gérer des requêtes en temps réel et de grandes quantités de données.

## III. Préconisations concernant le domaine et l'hébergement

- Nom du domaine : Les noms du domaine seront des sous-domaines de Qwenta ; menu-maker.qwenta.com , ainsi que menu-maker.qwenta.fr.
- Nom de l'hébergement. Heroku
- Adresses e-mail :

- **Support Technique** : support@menu-maker.qwenta.com

- **Informations Générales** : info@menu-maker.qwenta.com

- **Service Client** : contact@menu-maker.qwenta.com

#### **IV. Accessibilité**

- Compatibilité navigateur : Le site doit être compatible avec les dernières versions des navigateurs suivants ; Google Chrome, Safari, Firefox.
- Types d'appareils : Desktop uniquement, le site est conçu pour une utilisation sur ordinateurs de bureau et ne prévoit pas de version mobile ou tablette pour le moment.

#### **V. Recommandations en termes de sécurité**

##### **Accès aux comptes :**

Mise en place de mécanismes d'authentification sécurisés pour protéger les comptes des utilisateurs.

Utilisation de tokens sécurisés pour la gestion des sessions avec JWT - JSON Web Tokens.

##### **Plugins :**

S'assurer que tous les plugins et bibliothèques utilisés sont régulièrement mis à jour pour éviter les vulnérabilités.

Utiliser uniquement des plugins provenant de sources fiables et bien maintenues.

##### **Autres recommandations de sécurité :**

Utiliser HTTPS pour sécuriser les communications entre le client et le serveur.

Valider et assainir toutes les entrées utilisateurs pour éviter les injections SQL et XSS.

Mettre en place des sauvegardes régulières des données pour prévenir les pertes en cas d'incidents.

#### **VI. Maintenance du site et futures mises à jour**

##### **Grandes lignes du contrat de maintenance :**

Fournir un support technique pour résoudre les problèmes et les bugs signalés par les utilisateurs.

Assurer les mises à jour régulières du logiciel pour ajouter de nouvelles fonctionnalités, améliorer la performance et garantir la sécurité.

Réaliser des sauvegardes périodiques des données et du système pour minimiser les risques de perte de données.

Mise en place d'outils de monitoring pour surveiller la performance du site et détecter les anomalies.

Évaluer et mettre en œuvre des évolutions fonctionnelles et techniques en fonction des retours utilisateurs et des nouvelles exigences.



## VII. Technologies utilisées :

- **Frontend :**

Pour le frontend, nous avons choisi **React** et **Redux** pour développer des interfaces utilisateur dynamiques et réactives, facilitant la gestion des menus par les restaurateurs. **Styled-Components** est employé pour personnaliser le branding avec des styles modulaires et réutilisables. Pour l'export des menus en PDF, nous utilisons **React-PDF**, offrant une intégration fluide et un rendu précis.

- **Backend :**

Pour le backend, nous utilisons **Node.js** avec **Express** pour consommer les API externes telles que Deliveroo et Instagram. Cette approche nous permet de récupérer des données comme les menus des restaurants et le contenu des utilisateurs sans développer ces fonctionnalités en interne. **MongoDB** sert de base de données flexible et performante pour stocker les données, tandis que la sécurité des sessions et des connexions est assurée par **JSON Web Tokens (JWT)**.