

# Spécifications techniques

|        |                      |
|--------|----------------------|
| Projet | Menu Maker by Qwenta |
|--------|----------------------|

| Version | Auteur                 | Date       | Approbation  |
|---------|------------------------|------------|--------------|
| 1.0     | Soufiane,<br>Webgencia | 02/12/2023 | John, Qwenta |

Le but de ce document est de définir et justifier les spécifications techniques de Menu Maker by Qwenta.

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

| Besoin                   | Contraintes  | Solution  | Description de la solution  | Justification (2 arguments)   |
|--------------------------|--|---|---|---|
| Connexion Utilisateur    | Authentification sécurisée et récupération de mot de passe | Authentication Token                                    | Utilisation Authentication pour la gestion sécurisée des connexions et des récupérations de mot de passe. | 1) Firebase offre une gestion complète des identités, 2) Intégration fluide avec les applications web                                   |
| Confirmation de l'e-mail | Validité de l'adresse e-mail, Sécurité des données         | Envoi d'un e-mail de confirmation avec un lien sécurisé | Assure la validation des adresses e-mail des utilisateurs.  | 1) Protège contre les inscriptions frauduleuses. 2) Garantit que les utilisateurs fournissent des adresses e-mail valides.              |
| Catégorie de plat        | Doit pouvoir être créée dynamiquement                      | React-Router  | Utilisation de React-Router pour la navigation entre différentes catégories.                              | 1) Facilite la gestion dynamique des catégories. 2) Intégration naturelle avec React.   |
| Création de plat         | Besoin d'une interface intuitive et réactive               | React-Formik pour la gestion des formulaires            | Utilisation de Formik pour simplifier la gestion des formulaires de création de plats.                    | 1) Améliore l'expérience utilisateur en facilitant la saisie de données. 2) Réduit le code boilerplate pour la gestion des formulaires. |

|  |  |   |   |   |
|--|--|---|---|---|
| Style de menu                          | Personnalisation du style par le restaurateur                  | Styled Components                                 | Utilisation de Styled Components pour une personnalisation efficace du style du menu. | 1) Offre une approche basée sur les composants pour une stylisation modulaire. 2) Facilite la maintenance du code CSS.                            |
| Personnaliser votre menu               | Sélection d'options cohérentes, Gestion des variations de prix | Fonctionnalités de glisser-déposer avec React-DnD | Personnalisation intuitive de l'ordre des plats dans le menu.                         | 1) React-DnD facilite l'implémentation du glisser-déposer. 2) Amélioration de l'expérience utilisateur en permettant la personnalisation du menu. |
| Exportez et diffusez (Exporter en PDF) | Format PDF standardisé, Gestion des droits d'auteur            | React-PDF   | Génération de fichiers PDF pour les menus exportés.                                   | 1) React-PDF offre des fonctionnalités de génération de PDF en React. 2) Intégration transparente avec l'écosystème React.                        |
| Commander l'impression d'un menu       | Intégration avec un service d'impression                       | API d'impression en ligne                         | Intégration d'une API d'impression en ligne pour commander l'impression du menu.      | 1) Facilite la gestion des commandes d'impression. 2) Permet une solution flexible en fonction des besoins du restaurateur.                       |

|                        |   |                               |   |  |
|------------------------|---|-------------------------------|---|--|
| Menus précédents       | Historique des versions de menus        | Base de données               | Utilisation de Firebase pour stocker et gérer les versions antérieures des menus.       | 1) Intégration transparente avec d'autres services Firebase. 2) Possibilité de restaurer des versions antérieures en cas d'erreur.       |
| Informations légales   | Doit être accessible depuis chaque page | Footer global                 | Inclusion d'un composant Footer global pour l'accès aux informations légales.           | 1) Simplifie la maintenance en centralisant les éléments du footer. 2) Garantit l'accessibilité depuis n'importe quelle page.            |
| Tarifs                 | Mise à jour régulière des tarifs        | Gestion dynamique des données | Utilisation d'un système de gestion dynamique pour mettre à jour facilement les tarifs. | 1) Permet une mise à jour en temps réel des tarifs. 2) Réduit la dépendance aux mises à jour logicielles pour les changements de tarifs. |
| Diffuser sur Deliveroo | Conformité aux API de Deliveroo         | API de livraison Deliveroo    | Intégration avec les services de livraison Deliveroo.                                   | 1) Respect des normes API de Deliveroo. 2) Augmentation de la visibilité en se connectant à une plateforme de livraison populaire.       |
|                        |   |                               |   | 1) Utilisation des API officielles   |

|                        |  |                                       |  |  |
|------------------------|--|---------------------------------------|--|--|
| Partager sur Instagram | Conformité aux API d'Instagram                   | API d'Instagram                       | Partage direct d'images de plats sur Instagram                                   | d'Instagram. 2) Amélioration de la présence sur les réseaux sociaux.   |
| Déconnexion            | Sécurité des sessions, Révocation des tokens     | Gestion sécurisée de la déconnexion   | Fermeture de session, révocation des tokens, redirection vers la page d'accueil. | 1) Renforce la sécurité en fermant correctement la session. 2) Améliore l'expérience utilisateur avec une redirection appropriée.            |
| Infos utilisateur      | Affichage des informations utilisateur           | Context API                           | Utilisation de la Context API de React pour gérer les informations utilisateur.  | 1) Évite les problèmes de levée d'état entre les composants. 2) Offre une solution efficace pour le partage d'informations globales.         |
| Dashboard              | Accessibilité sécurisée aux données, Intuitivité | Tableau de bord personnalisé en React | Interface centralisée pour gérer tous les aspects de l'application.              | 1) React offre une construction modulaire pour une évolutivité facile. 2) Le tableau de bord centralisé améliore l'efficacité de la gestion. |
| Branding               | Personnalisation du branding                     | Gestion dynamique des styles          | Utilisation d'un système de gestion dynamique des styles pour permettre la       | 1) Facilite la personnalisation du branding sans toucher au code source. 2) Offre une solution   |

|              |  |  |                               |   |
|--------------|--|--|-------------------------------|---|
| restaurateur |  |  | personnalisation du branding. | flexible pour différents besoins de branding. |
|--------------|--|--|-------------------------------|---|

- **Liens avec le back-end**

- Quel langage pour le serveur ?

**Node JS** C'est le langage utilisé pour le développement du côté serveur. NodeJS est choisi ici en raison de sa performance et de sa scalabilité, ce qui peut être crucial pour une application comme MenuMaker.

- A-t-on besoin d'une API ? Si oui laquelle ? **Oui**

NodeJS a été choisi comme langage serveur pour "Menu Maker by Qwenta" en raison de son unification avec JavaScript, de son efficacité dans la gestion asynchrone des opérations en temps réel, et de son vaste écosystème de modules facilitant l'intégration et l'extension du projet. Sa simplicité, ses performances et sa capacité à évoluer horizontalement en font un choix optimal pour répondre aux besoins spécifiques de cette application.

- Base de données choisie : **NoSQL(MongoDB)**

MongoDB est choisi comme système de gestion de base de données NoSQL en raison de sa flexibilité et de sa capacité à gérer des données non structurées, ce qui pourrait être avantageux pour les besoins de MenuMaker.

- **Préconisations concernant le domaine et l'hébergement**

- Nom du domaine: **www.menu-maker-qwenta.com**
- Nom de l'hébergement: **menu-maker-qwenta.vercel.app**
- Adresses e-mail: **contact@menu-maker-qwenta.com**

- **Accessibilité**

- Compatibilité navigateur: **Chrome,Firefox** et **Safari**
- Types d'appareils: **Desktop**

- **Services tiers**

**Google Analytics** : Google Analytics permet de suivre et d'analyser le comportement des utilisateurs sur le site. on pourra obtenir des informations précieuses sur l'utilisation de l'application, les pages les plus populaires, les parcours d'utilisateur, etc.

**Stripe** :joue un rôle essentiel dans la facilitation des transactions financières en ligne, offrant aux entreprises une solution sécurisée, flexible et conviviale pour accepter les paiements électroniques.

- **Recommandations en termes de sécurité**

**Mises à jour régulières** : Assure que tous les plugins, ainsi que le système d'exploitation, sont régulièrement mis à jour. Les mises à jour contiennent souvent des correctifs de sécurité importants.

**Authentification à deux facteurs (2FA)** : Active l'authentification à deux facteurs pour les comptes critiques. Cela ajoute une couche supplémentaire de sécurité en exigeant une confirmation supplémentaire au-delà du mot de passe.

**Utilisation de mots de passe forts** : Assure que tous les comptes, en particulier les comptes administratifs, utilisent des mots de passe forts. Cela inclut une combinaison de lettres majuscules et minuscules, de chiffres et de caractères spéciaux.