

# Spécifications techniques

« Menu Maker » for **Qwenta**

Version	Auteur	Date	Approbation
1.0	Mickael	20 Décembre 2024	Soufiane

I. Choix technologiques .....	2
II. Liens avec le back-end.....	3
III. Préconisations concernant le domaine et l'hébergement .....	3
IV. Accessibilité .....	3
V. Recommandations en termes de sécurité .....	3
VI. Maintenance du site et futures mises à jour .....	4

## I. Choix technologiques

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

Besoin	Contraintes	Solution	Description de la solution	Justification (2 arguments)
Ex. : Création d'une catégorie de menu	Ex. : L'ajout d'une catégorie doit pouvoir se faire directement sur l'écran de création de menu depuis une modale.	Ex. : react-modal	Ex. : Cette librairie React permet de créer simplement des modales performantes, accessibles avec un minimum de code.	Ex. : 1) Nous avons choisi de développer en React, la librairie est cohérente avec ce choix. 2) Il s'agit de la librairie la plus utilisée.
Création d'un menu	Doit permettre l'ajout de plats dynamiquement	React	Framework pour créer des interfaces utilisateur dynamiques et réactives.	1) Adapté pour des projets front-end modernes. 2) Grande communauté et support constant.
Création de plat	Ajouter des plats avec nom, prix, description et image	Formik et react-dropzone	<b>Formik</b> Bibliothèque pour React qui simplifie la gestion des formulaires <b>Dropzone</b> Bibliothèque React qui simplifie la création de	1) Compatible avec React 2) elles se complètent bien pour gérer des formulaires

			zones de dépôt pour le téléchargement de fichiers.	
Gestion de données (plats, catégories).	Stockage et récupération de données complexes.	MongoDB	Base de données NoSQL.	1) Idéal pour les relations dynamiques entre catégories et plats. 2) Haute scalabilité.
Gestion des sessions	sécuriser les échanges de données entre le client et le serveur	JWT	souvent utilisé pour l'authentification et l'autorisation dans les applications web.	1) Le serveur n'a pas besoin de stocker les sessions. Le token contient toutes les informations nécessaires. 2) Les données sont signées, empêchant leur altération sans le secret ou la clé privée.
Authentification ou inscription	Doit permettre l'authentification ou l'inscription sécurisée à l'utilisateur	nodemailer	bibliothèque pour Node.js permettant d'envoyer des e-mails souvent utilisée pour la validation d'inscription, la réinitialisation de mot de passe, ou des notifications.	1) Nodemailer permet d'envoyer des e-mails rapidement grâce à une API simple et compatible avec de nombreux services (Gmail, Yahoo, etc.). 2) Prise en charge des e-mails HTML, des pièces jointes et des configurations

				spécifiques comme les authentifications sécurisées.
Export PDF	Génération rapide et fidèle des menus en fichier PDF.	jspdf	Librairie JS légère pour créer des documents PDF depuis le navigateur.	1) Flexible et performant. 2) Facile à intégrer avec React.
Diffusion sur Deliveroo et instagram.	API standardisée pour l'envoi des menus.	API REST	Création d'une API pour transmettre les données de menu vers des plateformes tierces.	1) Norme couramment utilisée. 2) Extensible pour d'autres intégrations futures.
Personnalisation des menus	Gestion des polices et couleurs en temps réel et ajout d'un logo.	Styled-components et Color Picker Library	<b>Color Picker Library</b> permet de sélectionner facilement des couleurs dans une interface utilisateur. <b>Styled-components</b> est une bibliothèque pour React qui permet d'écrire du CSS directement dans le JavaScript	1) Compatible avec React. 2) Simplifie les modifications visuelles.

## II. Liens avec le back-end

- Quel langage pour le serveur ? **Node.js**
  - *Justification : Compatible avec JavaScript côté front-end, simplifie la communication.*
- A-t-on besoin d'une API ? Si oui laquelle ? Oui, **API REST** pour gérer les données de menus (CRUD : Create, Read, Upload, Delete).
- Base de données choisie : **MongoDB** (NoSQL)
  - *Justification :*  
*Flexible pour des données structurées (plats, catégories) et non structurées (descriptions, options).*  
*Facile à scaler si l'application se développe.*

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

- Nom du domaine : menu.qwenta.com (à valider)
- Nom de l'hébergement : **AWS** ou **Vercel** pour des performances optimales.
- Adresses e-mail : Créer une adresse spécifique pour le support, par exemple support@qwenta.com.

## IV. Accessibilité

- Compatibilité navigateur.
  - Chrome, Firefox, Safari (dernières versions).

- Types d'appareils.
  - Desktop uniquement (pas de version mobile prévue pour ce projet)
- Extension Wave
  - Vérifier la conformité du site

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

- Accès utilisateur :
  - Authentification par mot de passe sécurisé (hashage avec bcrypt).
  - Utilisation de JSON Web Tokens (JWT) pour la gestion des sessions.
- Gestion des plugins :
  - Vérification régulière de la fiabilité et des mises à jour des dépendances utilisées (React, Nodemailer, etc.).
- Communication sécurisée :
  - Mise en place de HTTPS pour toutes les connexions.
  - Sauvegardes régulières de la base de données

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

- Grandes lignes du contrat de maintenance.
  - Mise à jour des dépendances front-end et back-end tous les 6 mois pendant 2 ans (renouvelable).

- Correction des bugs critiques sous 48 heures après leur signalement.
- Support pour de nouvelles fonctionnalités ou évolutions, avec devis spécifique.