

Spécifications techniques

[Menu maker by Qwenta]

Version	Auteur	Date	Approbation
1.0	Clovis, Webgencia	09/08/2024	John, Qwenta

I.	Choix technologiques	2
II.	Liens avec le back-end.....	5
III.	Préconisations concernant le domaine et l'hébergement	6
IV.	Accessibilité.....	7
V.	Recommandations en termes de sécurité.....	7
VI.	Maintenance du site et futures mises à jour	8
	Services Inclus dans la Maintenance.....	8
	Niveaux de Service (SLA)	8
	Frais de Maintenance	8
	Confidentialité et Sécurité des Données	8

I. Choix technologiques

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

Besoin	Contraintes	Solution	Description de la solution	Justification (2 arguments)
Création / connexion de compte Modification information Utilisateur Gestion donnée Utilisateur	L'utilisateur doit pouvoir soit se connecter avec son compte ou créer un compte, les différentes données de l'utilisateur doivent être sauvegardé	Node.JS avec Express.JS, React-modal et comme base de données MongoDB	Node.js avec Express.js va nous permettre de traiter les requêtes de connexion/création de compte. MongoDB nous permet de stocker les informations utilisateurs. React-modal pour la modale de Login.	MongoDB utilise un format JSON pour stocker les données, ce qui permet de stocker des informations utilisateur variées sans avoir besoin de schémas fixes. MongoDB permet une scalabilité horizontale. Node.js avec Express.js est Facile à comprendre et à utiliser. React-modal est une des librairies les plus utilisées.
Ajout d'un plat et Création d'une catégorie de menu	L'ajout d'une catégorie ou d'un plat doit pouvoir se faire directement sur l'écran de création de menu depuis une modale.	React-modal	Cette librairie React permet de créer simplement des modales performantes, accessibles avec un minimum de code.	Nous avons choisi de développer en React, la librairie est cohérente avec ce choix. Il s'agit de la librairie la plus utilisée.

Visualisation du menu en temps réel, Export du menu en PDF	On doit pouvoir visualiser le menu pendant qu'on le crée/modifie , à la fin on peut exporter le menu en PDF	Bibliothèque jsPDF	jsPDF est une bibliothèque JavaScript permettant de générer des fichiers PDF directement dans le navigateur	<p>jsPDF est compatible avec React.</p> <p>Il s'agit d'une des bibliothèques les plus utilisées pour la génération de PDF en JavaScript.</p> <p>Elle offre de nombreuses fonctionnalités comme l'ajout de texte, d'images, et de styles, nécessaires pour la mise en forme d'un menu.</p>
Visualisation des informations légales	En tant qu'internaute, je dois pouvoir accéder au contenu "Mentions légales" dans une modale, et l'information "Tous droits réservés" doit être affichée.	React-modal	Cette librairie React permet de créer simplement des modales performantes, accessibles avec un minimum de code.	<p>Nous avons choisi de développer en React, la librairie est cohérente avec ce choix.</p> <p>Il s'agit de la librairie la plus utilisée.</p>

Diffuser un menu	L'utilisateur doit pouvoir diffuser son menu sur Instagram/deliveroo	API Deliveroo API Instagram	On utilisera les API correspondant à chacun des sites pour pouvoir faire la diffusion	Cela semble logique d'utiliser les API des sites correspondant car ce sont les plus adapté pour eux
Prévisualisation du menu	L'utilisateur doit pouvoir prévisualiser le menu en direct avec différents style	SCSS	On utilisera un sass pour modifier et la visualisation du menu	<p>SCSS permet de créer des styles modulaires avec des variables, des mixins et des fonctions, facilitant la maintenance et la réutilisation du code</p> <p>SCSS simplifie la lecture du code CSS et accélère le développement</p> <p>Il est cohérent avec le choix de développement en React</p>

II. Liens avec le back-end

- Quel langage pour le serveur ? *Node.js avec Express.js*

Performance et Scalabilité :

Asynchrone et non-bloquant : Node.js est particulièrement performant pour les applications nécessitant beaucoup d'entrées/sorties (I/O) comme les requêtes de base de données et les appels API.

Scalabilité : Facile à mettre à l'échelle horizontalement en utilisant des processus et des threads.

Écosystème :

npm : Un riche écosystème de modules et de packages pour presque toutes les fonctionnalités que vous pourriez imaginer.

Microservices : Node.js est bien adapté pour une architecture de microservices, ce qui permet de gérer facilement une grande base d'utilisateurs.

Intégration avec React :

Isomorphisme : Partage de code entre le client et le serveur.

API RESTful: Facile à implémenter pour servir les données React.

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

RESTful API Utilise les verbes HTTP (GET, POST, PUT, DELETE) pour les opérations CRUD (Create, Read, Update, Delete).

Avantages :

- Simplicité et standardisation.
- Bien supportée par de nombreux frameworks et bibliothèques.
- Facile à comprendre et à utiliser.

Base de données choisie : *Ex : SQL / NO SQL.*

Les bases de données NoSQL comme MongoDB sont optimisées pour des performances élevées dans des scénarios de lecture/écriture intensifs. Ce qui rentre parfaitement dans le cadre du projet car on doit pouvoir ajouter et modifier les menus facilement et MongoDB est très flexible et facilement scalable, bien adapté pour les applications modernes avec des structures de données dynamiques.

III. Préconisations concernant le domaine et l'hébergement

- Nom du domaine : QWENTA/MenuMaker.com
- Nom de l'hébergement : Google Cloud Platform (Cloud) :
 - Offre des solutions de cloud computing performantes et évolutives.
 - Intégration facile avec d'autres services Google, comme les bases de données et le stockage.
- Adresses e-mail : contact@menumakerqwenta.com

IV. Accessibilité

Compatibilité navigateur : on se contente de la compatibilité avec les dernières versions de Chrome, Safari et Firefox.

Types d'appareils : Le site devra être en version desktop. Pas de version mobile à développer ni à prévoir.

V. Recommandations en termes de sécurité

Pour la gestion des comptes dans la base de données on utilisera bcrypt pour le hashage des adresse e-mail dans la base de données.

Pour le middleware d'authentification on utilisera des tokens d'identification avec jsonwebtoken.

Le système de connexion n'utilise pas de mot de passe mais un système de vérification par e-mail.

VI. Maintenance du site et futures mises à jour

Services Inclus dans la Maintenance

- Mises à Jour du Système : Des plugins et des extensions pour garantir la sécurité et les performances.
- Corrections de Bugs : Identification et résolution des bugs et des dysfonctionnements signalés par le client ou découverts lors des vérifications régulières.
- Optimisation de la Performance : Amélioration continue des performances du site, y compris l'optimisation des temps de chargement et de la réponse du serveur.
- Sauvegardes Régulières : Mise en place de sauvegardes automatiques régulières et stockage sécurisé de celles-ci.
- Sécurité : Surveillance et protection contre les menaces de sécurité, y compris les mises à jour de sécurité, les audits et la mise en œuvre de mesures correctives.
- Support Technique : Assistance technique par e-mail, téléphone, ou chat pour résoudre les problèmes rencontrés par le client.

Niveaux de Service (SLA)

- Temps de Réponse : à prévoir.
- Résolution des Problèmes : à prévoir.
- Disponibilité du Service : à prévoir.

Frais de Maintenance

- Tarification : à prévoir.

Confidentialité et Sécurité des Données

- Confidentialité : Engagement à respecter la confidentialité des données du client.
- Sécurité : Mise en œuvre de mesures de sécurité pour protéger les données du client contre tout accès non autorisé.