



Webgencia

Spécifications techniques

Projet	[Menu Maker by Qwenta + Qwenta]
---------------	---------------------------------

Version	Auteur	Date	Approbation
1.0	[Soufiane, Webgencia]	[26 Septembre 2023]	[John, Qwenta]

Le but de ce document est de définir et justifier les spécifications techniques de [nom du projet].



I. Choix technologiques

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

Besoin	Contraintes	Solution	Description de la solution	Justification (2 arguments)
<i>Création d'un outil en ligne dynamique permettant aux restaurateurs de publier, partager, mettre en forme et imprimer leurs menus. Page de login & déconnexion</i>	<i>Les restaurateurs doivent se connecter à un espace privé et sécurisé pour créer et modifier les menus. Nécessité de s'authentifier, ou au contraire de confirmer son mail pour accéder pour la première fois à l'application</i>	<i>Framework Front-end: REACT Framework Backend: EXPRESS JS (Language NodeJS). HTML, CSS et JavaScript pour la partie frontend, Node.js ou PHP, une base de données et des API pour la gestion de l'authentification côté serveur.</i>	<i>Elle est utilisée pour créer des applications web à petite ou grande échelle et est particulièrement utile pour créer des sites web interactifs. HTML est utilisé pour structurer la page web. CSS est utilisé pour la mise en page et la présentation visuelle de la page. JavaScript est un langage de programmation côté client qui vous permet d'ajouter des fonctionnalités interactives à la page web. React, Angular ou Vue.js pour simplifier le</i>	<i>1) Nous avons choisi de développer en React et Express JS 2) Il s'agit de la librairie très utilisée. (Facilite la compréhension) 3) Utilise les fonctionnalités de Full-stack JS (gain de temps et de l'argent)</i>



			<p>développement de la modale et l'interaction avec l'API du serveur.</p> <p>Serveur backend pour gérer l'authentification de l'utilisateur (Node.js, Python (avec Django ou Flask), Ruby (avec Ruby on Rails), PHP, etc.</p> <p>Base de données pour stocker ces données de manière sécurisée.</p> <p>Utilisation de protocoles de cryptage (HTTPS), la gestion sécurisée des mots de passe, la gestion sécurisée des mots de passe, la protection contre les attaques CSRF (Cross-Site Request Forgery) et XSS (Cross-Site Scripting), etc.</p>	
Création d'une catégorie de menu	L'ajout d'une catégorie doit pouvoir se faire directement sur l'écran de création de menu	react-modal	<p>Cette librairie React permet de créer simplement des modales performantes, accessibles avec un</p>	<p>1) Nous avons choisi de développer en React, la librairie est cohérente avec ce choix.</p> <p>2) Il s'agit de la librairie la</p>



	<i>depuis une modale.</i>		<i>minimum de code.</i>	<i>plus utilisée.</i>
Génération d'un pdf	<i>Le lien doit être fait vers le back-office de Qwenta</i>	<i>HTML, CSS</i>	<i>Intégration du doucement PDF en HTML</i>	<i>Langage simple d'utilisation</i>
Exportation deliveroo	<i>Accès à l'API deliveroo</i>	<i>API</i>	<i>Intégrer son utilisation dans le site web.</i>	
Landing, Dashbord	<i>Pouvoir accéder au contenu "Mentions légales" dans une modale, et l'information "Tous droits réservés" doit être affichée. Pouvoir comprendre l'utilité de cette application.</i>	<i>HTML, CSS</i>	<i>Utilisation des liens internes avec des ancrs vers les sections spécifiques de la page, les internautes pourront facilement accéder aux différentes parties de la landing page non connectée en cliquant sur les liens correspondants.</i>	
Infos utilisateur	<i>Lier plusieurs adresses e-mail à son compte. Modifier son adresse e-mail de base</i>	<i>Base de données MySQL</i>	<i>Création base de données ou un modèle qui stocke les adresses e-mail associées à chaque compte utilisateur.</i>	



Pourquoi choisir REACT ET EXPRESS ?

ReactJS est très **rapide**, simple à lire, Le Javascript plus **simple** à écrire.

Avec EXPRESS, grâce à l'utilisation d'un serveur NodeJS, le code va pouvoir être généré **côté client ET côté serveur en Javascript** et faire appel à un seul Développeur **Full-stack JS**.

II. Liens avec le back-end

- Quel langage pour le serveur ? *Ex. : NodeJS / PHP / Python, etc.*

Qu'est-ce que Node.js et pourquoi l'utiliser ?

Node.js est un environnement d'exécution single-thread, open-source et multi-plateforme permettant de créer des applications rapides et évolutives coté serveur et en réseau. Il fonctionne avec le moteur d'exécution JavaScript V8 et utilise une architecture non bloquante et pilotée par les évènements, ce qui le rend efficace et adapté aux applications en temps réel.

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



Pourquoi utiliser Node JS pour construire une API REST ?

Pour la construction d'une API Node JS est un choix qui est souvent pertinent pour les raisons suivantes :

1. Son traitement non bloquant des requêtes,
2. Sa performance et sa scalabilité,
3. L'écosystème JavaScript et les packages open sources disponibles.

- Base de données choisie :

MySQL sera utilisé.

III. Préconisations concernant le domaine et l'hébergement

- Nom du domaine.

Menu maker by qwenta.co

- Nom de l'hébergement.

Bluehost est un hébergeur web bien établi et populaire qui est souvent recommandé pour une variété de



projets en ligne, y compris les sites web personnels, les blogs, les sites d'entreprise, et même les applications web plus complexes.

IV. Accessibilité

- Compatibilité navigateur.

Il est d'une importance primordiale de vérifier si votre site ou application s'affiche correctement sur les différents supports (ordinateurs, tablettes, smartphones) et en fonction des différents navigateurs. Cela permet, en plus de répondre aux attentes des internautes, d'être mieux référencé dans les résultats de recherche.

- Types d'appareils.

Le site devra être en version desktop. Pas de version mobile à développer ni à prévoir.

V. Services tiers

1. **Services de Sécurité et de Protection contre la Fraude :** Pour protéger votre application et les transactions en ligne, nous pourrions utiliser des services de sécurité et de prévention de la fraude, tels que Cloudflare.
2. **Hébergement de Fichiers et de Médias :** l'application permet aux restaurateurs de télécharger des images



de leurs menus, nous aurions besoin d'un service d'hébergement de fichiers et de médias, comme Amazon S3.

3. **Services d'Authentification et d'Identité :** Pour gérer l'authentification des utilisateurs et des restaurateurs, vous pourriez envisager d'utiliser des services d'authentification tiers, tels que Auth0.
4. **Services de Gestion des Contenus :** Pour permettre aux restaurateurs de gérer et personnaliser leurs menus, nous pourrions envisager d'intégrer un système de gestion de contenu (CMS) tel que WordPress.

VI. Recommandations en termes de sécurité

La sécurité de l'application web est une préoccupation majeure pour Menu Maker, car elle traite des données sensibles des utilisateurs tels que leurs informations de connexion, leurs données de menu et leurs informations de paiement. Ainsi, pour assurer la sécurité de l'application web, plusieurs mesures de sécurité ont été mises en place :

- HTTPS :

Toutes les communications entre l'application et les navigateurs des utilisateurs sont sécurisées par le protocole HTTPS, qui chiffre les données en transit pour empêcher les attaquants d'intercepter et de manipuler les données.

- Authentification et autorisation :

L'application utilise un système d'authentification et d'autorisation robuste pour protéger les données des utilisateurs. Les utilisateurs doivent se connecter avec un nom d'utilisateur et un mot de passe pour accéder à leur compte, et l'authentification est gérée en utilisant des protocoles de sécurité tels que JWT (JSON Web Token). L'application met également en œuvre des contrôles d'accès basés sur les rôles pour s'assurer que seuls les utilisateurs autorisés peuvent accéder aux données sensibles.



- Cryptage des données :

Toutes les données sensibles sont stockées dans la base de données MongoDB, qui est protégée par des algorithmes de cryptage de niveau industriel tels que AES (Advanced Encryption Standard) pour empêcher les attaquants d'accéder aux données en cas de violation de la sécurité.

- Validation des entrées utilisateur :

L'application dispose de mécanismes de validation pour toutes les entrées utilisateur, afin d'empêcher les attaquants d'injecter du code malveillant ou des données non valides dans l'application.

VII. Maintenance du site et futures mises à jour

- Grandes lignes du contrat de maintenance.