

Spécifications techniques

Menu Maker - Qwenta

Version	Auteur	Date	Approbation
1.0	Atef, Webgencia	05/09/2024	John, Qwenta

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)
Faire comprendre l'utilité de l'application Menu Maker à l'utilisateur.	Attirer l'attention de l'utilisateur sur la description des outils que propose Menu Maker.	Landing page: HTML, CSS, React	Lorsque l'utilisateur utilisera le site pour la première fois, une brève description du site et de ses outils sera affichée sur la page d'accueil.	<ol style="list-style-type: none"> 1) La page d'accueil permet d'aiguiller l'utilisateur lors de sa première visite. 2) Réduire les besoins en support client en cas de mauvaise compréhension.
Donner la possibilité au restaurateur de pouvoir se connecter	Sécurité des données	Authentification par email / mot de passe + 2FA	Intégration d'un système d'authentification sécurisé avec option d'authentification à deux facteurs pour renforcer la sécurité.	<ol style="list-style-type: none"> 1) Protéger les données sensibles de l'utilisateur 2) Simple à mettre en place et facile à utiliser pour le restaurateur.
Créer une nouvelle catégorie	Interface intuitive et stockage des nouvelles catégories	Mongoose avec MongoDB pour la création et la gestion de catégories.	Mongoose permet de définir des schémas de données flexibles et d'enregistrer des catégories dans une base de données NoSQL, tout en assurant une interactivité rapide avec MongoDB.	<ol style="list-style-type: none"> 1) Mongoose offre une gestion flexible des données, permettant des modifications dynamiques. 2) MongoDB supporte facilement l'ajout de nouvelles données sans altérer les performances.

Ajouter un plat	Gestion des détails (prix, ingrédients, etc..).	React	React gère efficacement les formulaires interactifs tandis que Node.js	<ol style="list-style-type: none"> 1) React garantit une interface fluide pour l'utilisateur. 2) React permet une gestion efficace du "State" de l'application
Personnaliser le style du menu	Flexibilité des options de personnalisation et utilisation intuitive.	HTML / CSS / React JS	Utilisation d'écouteurs d'événements pour modifier le style en fonction de la couleur sélectionnée	<ol style="list-style-type: none"> 1) Permet de créer un menu unique adapté à l'identité du restaurant. 2) L'utilisateur peut voir les modifications en temps réel sans rechargement de la page.
Exportation PDF	Le fichier PDF doit refléter fidèlement le design personnalisé du menu, avec un rendu correct des polices, images, et mises en page.	react-pdf	react-pdf permet de créer des documents PDF à partir de composants React. Cela permet d'intégrer facilement les styles et de garantir que le PDF final correspond exactement à l'interface utilisateur.	<ol style="list-style-type: none"> 1) react-pdf est spécialement conçu pour React, ce qui facilite son intégration dans l'application sans ajouter de complexité. 2) Il offre un contrôle total sur le rendu des documents, permettant de maintenir la cohérence visuelle entre l'interface utilisateur et le fichier PDF exporté.

Imprimer un menu	Le format d'impression doit être correctement ajusté pour des résultats professionnels.	react-to-print	Cette bibliothèque React permet de créer des vues imprimables de manière simplifiée, avec la possibilité de gérer les styles spécifiques à l'impression.	<ol style="list-style-type: none"> 1) "React to Print" est facile à implémenter et maintient la qualité visuelle du menu. 2) Permet de définir des styles distincts pour l'impression, garantissant un rendu optimal.
Visualisation des anciens menus	Gestion de l'historique des menus avec accès rapide aux versions précédentes.	MongoDB avec Mongoose pour stocker et récupérer les menus archivés.	MongoDB permet de stocker les anciens menus sous forme de documents avec des métadonnées, facilitant leur récupération rapide.	<ol style="list-style-type: none"> 1) MongoDB gère efficacement des ensembles de données non structurées, idéal pour des historiques de menus. 2) La flexibilité de Mongoose permet une gestion intuitive des versions et des archives.

II. Liens avec le back-end

- Quel langage pour le serveur ?

NodeJS

Nous choisirons d'utiliser NodeJS pour le serveur pour la raison qu'il permet d'utiliser le même langage (Javascript) à la fois côté client et serveur, ce qui simplifiera le développement, réduira la courbe d'apprentissage, cela aidera pour la cohérence générale.

De plus, Node.js dispose d'un vaste écosystème de packages et de bibliothèques via npm, ce qui permettra de facilement intégrer des fonctionnalités comme l'authentification, la gestion des fichiers, etc.

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

Une API Restful combinée avec Express.js (pour sa facilité à configurer) nous permettra de séparer la logique du serveur de l'interface utilisateur.

Le frontend pourra utiliser le CRUD sans avoir à se soucier du fonctionnement interne du serveur.

- Base de données choisie

MongoDB (NO SQL)

Nous choisissons d'utiliser MongoDB pour ce projet dont les relations entre les différentes entités (catégories, plats, menus) sont assez simples.

III. Préconisations concernant le domaine et l'hébergement

- **Nom du domaine.**

Je recommande un nom simple et représentatif comme menumaker.com

- **Nom de l'hébergement.**

Heroku: Parfait pour déployer des applications Node.js. Il est facile à configurer et offre un plan gratuit pour les projets en développement.

- **Adresse e-mail.**

Google Workspace fournit une gestion facile des adresses e-mail avec le nom de domaine, ainsi que des outils supplémentaires comme Google Drive, Docs, etc.

contact@menumaker.com : Pour les questions générales des utilisateurs.

support@menumaker.com : Pour le support technique et l'assistance client.

admin@menumaker.com : Pour l'administration du site.

IV. Accessibilité

- **Compatibilité navigateur**

Le site devra être compatible avec les dernières versions de Chrome, Safari et Firefox.

- **Types d'appareils**

L'orientation du développement du site sera dans le but d'une utilisation Desktop.

- **Accessibilité**

Le site doit être accessible au minimum au clavier, et lisible par un lecteur d'écran.

V. Recommandations en termes de sécurité

Nous utiliserons des mécanismes d'authentification robustes pour protéger les comptes des utilisateurs.

- **Authentification à deux facteurs (2FA)** : Ajout d'une couche supplémentaire de sécurité en demandant un second facteur (comme un code envoyé par SMS ou une application d'authentification) lors de la connexion.
- **Gestion des sessions** : Assurer que les sessions utilisateurs expirent après une période d'inactivité et qu'elles sont correctement invalidées lors de la déconnexion.
- **Hashage des mots de passe** : Stockage des mots de passe de manière sécurisée en utilisant des algorithmes

de hashage sécurisés comme bcrypt.

VI. Maintenance du site et futures mises à jour

- **Objectif du contrat de maintenance:**

Définir clairement les objectifs du contrat, notamment la maintenance régulière, la gestion des mises à jour, et le support technique.

- **Maintenance:**

S'assurer de maintenir le bon fonctionnement du site Menu Maker par le biais d'actions visant à prévenir les problèmes avant qu'ils ne surviennent et les corriger si cela arrive.

Proposer l'ajout d'amélioration du site et des fonctionnalités au fil du temps.