Spécifications techniques

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

| **Version** | **Auteur** | **Date** | **Approbation** |
| --- | --- | --- | --- |
| 1.0 | Atef Lachhab | 18/01/2024 | A définir |

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

1. **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. | Page d’introduction (landing page) | Lorsque l’utilisateur utilisera le site pour la première fois,  une brève description du site et des ses outils seront affichés sur la page d’accueil. | 1. La page d’accueil permet d’aiguiller l’utilisateur lors de sa première visite. 2. L’utilisateur aura une compréhension plus globale des différentes fonctionnalités. |
| Donner la possibilité au restaurateur de pouvoir se connecter | L’utilisateur doit être inscrit sur le site | Inscription | L’utilisateur pourra se connecter grâce à son adresse email. | 1. Il suffit seulement d’un adresse email pour pouvoir s’inscrire 2. Avoir un compte permet de ne pas perdre ses travaux en cas de déconnection |
| Créer une nouvelle catégorie | La nouvelle catégorie doit pouvoir se faire directement sur l’écran de création de menu depuis une modal | react-modal | Cette librairie react permet de créer simplement des modales performantes, accessibles, avec un minimum de code | 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 |
| Ajouter un plat | Le plat doit comporter une photo, avec son nom et sa description et l’ajout doit se faire dans une modal | react-modal | Cette librairie react permet de créer simplement des modales performantes, accessibles, avec un minimum de code | 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 |
| Personnaliser le style du menu | La typographie et les couleurs doivent être visualisables dynamiquement | javascript dynamique | On peut utiliser JavaScript pour modifier dynamiquement les styles de réponse à des événements utilisateur ou à d’autres conditions. | 1. Interaction en temps réel avec le DOM. 2. Adaptabilité et personnalisation |
| Exportation PDF |  |  |  |  |
| Imprimer un menu | L’utilisateur doit pouvoir imprimer son menu facilement | PDF | Le restaurateur pourra exporter son menu en PDF puis choisir d’imprimer le menu à partir de son logiciel de lecture PDF | 1. C’est la méthode la plus répandue |
| Visualisation des anciens menus | L’accès aux menus anciennement créés doit être accessible et doit afficher leur date de création. | API | L’envoi des données des anciens menus sur l’API permet à l’utilisateur d’y avoir accès en tout temps. | 1. En envoyant les données des anciens menus sur l’API, l’utilisateur peut y accéder à tout moment 2. L’envoi des données sur l’API centralise le stockage des anciens menus. |

1. **Liens avec le back-end**

* Quel langage pour le serveur ?

**NodeJS**

Nous choisirons d’utiliser NodeJS pour le serveur pour la raison qu’il fait partie de l'écosystème de Javascript et React, cela aidera pour la cohérence générale.   
De plus, la documentation de Node JS fait partie de celles qui sont le mieux documentées.

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

**Oui,** une API nous permettra d’authentifier l’utilisateur afin qu’il puisse faire ses menus,les enregistrer également. Évidemment il pourra par la suite récupérer ses anciens menus enregistrer pour continuer à travailler dessus par exemple.

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

**SQL,** ca SQL est bien adapté aux schémas fixes et aux relations. (par exemple, les plats, les catégories de plats, les ingrédients, etc.)

1. **Préconisations concernant le domaine et l’hébergement**

* **Nom du domaine.**

menumaker.com

* **Nom de l’hébergement.**

**Hostinger:**  Ce serveur possède des tarifs concurrentiels et figure comme un des hébergeurs les plus performants du marché.

* **Adresse e-mail.**

menumaker@contact.com

1. **Accessibilité**

* **Compatibilité navigateur.**

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

* **Types d’appareils.**

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

1. **Services tiers**

* **Firebase**: Service d’authentification pour simplifier la gestion des utilisateurs et garantir la sécurité des comptes.
* **MongoDB Atlas** : Service de sauvegarde automatique pour la base de données pour éviter la perte de données en cas de problème.
* **Github :** Système de contrôle de version pour le développement collaboratif et le suivi des modifications apportées au code source.
* **AWS WAF :**  Service de sécurité pour protéger contre les attaques par injection SQL, XSS, etc..
* **Datadog :** Pour surveiller les performances du serveur, identifier les goulets d’étranglement et optimiser l’efficacité du site.

1. **Recommandations en termes de sécurité**

* OAuth 2.0 : Méthode d’authentification sécurisée basée sur des protocoles modernes.
* Mettre en place des politiques de mots de passe robustes, (par exemple: au moins 8 caractères alphanumériques, qui doit contenir des symboles %\*$..) et qui n’inclut pas d’information personnelles (date de naissance, prénom, etc.).
* Mettre en place des mécanismes de verrouillage de compte en cas de multiples échecs pour limiter les tentatives de connexion infructueuses.
* Utilisation des tokens de session sécurisés.
* Mise en place d’un système de journalisation robuste pour enregistrer les activités du site pour détecter de potentiels comportements suspects.

1. **Maintenance du site et futures mises à jour**

* Les mises à jour de sécurité pour le serveur nodeJS, les dépendances du projet et les système de gestion de base de données SQL doivent être effectués régulièrement pour protéger le site contre les vulnérabilités connues.
* Effectuer des tests de sécurité réguliers, y compris des analyses de vulnérabilité?
* Maintenir une documentation complète et à jour décrivant l’architecture, les composants, les dépendances, les procédures de déploiement, et les bonnes pratiques de développement pour ainsi faciliter l’intégration des nouveaux membres de l'équipe.
* Se servir de la veille pour rester à l'affût des nouvelles versions des technologies utilisées (NodeJs, React, etc.) et des meilleures pratiques de développement. Planifier en conséquence les mises à niveau nécessaires pour bénéficier des dernières fonctionnalités et des correctifs de sécurité.