Spécifications techniques

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Auteur** | **Date** | **Approbation** |
| 1.0 | Lejeune Corentin | 16/03/2024 | John, Qwenta |

Menu Maker by Qwenta est une plateforme web dédiée aux restaurateurs, offrant une solution intuitive et efficace pour la création et la gestion de leurs menus en ligne. L'objectif de ce document est de décrire les spécifications techniques de l'outil, en mettant en avant les choix technologiques, les liens avec le back-end et les recommandations relatives à l'hébergement.

Les choix technologiques ont été orientés par les besoins fonctionnels du projet ainsi que par les dernières tendances en matière de développement web. Le document débutera par une présentation des technologies utilisées pour chaque fonctionnalité clé de Menu Maker. Ensuite, il abordera les aspects liés au back-end, incluant la description du serveur, des APIs et de la base de données.

Enfin, des recommandations seront fournies concernant le domaine et l'hébergement de l'application. Ces informations visent à garantir la performance, la sécurité et la fiabilité de Menu Maker tout en assurant une expérience utilisateur optimale pour les restaurateurs.

1. **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.* |
| Authentification | Identification via E-mail  Assistance | Firebase Authentification | Firebase Authentication est un service de Google Firebase permettant d'ajouter facilement l'authentification sécurisée aux applications web et mobiles. Il offre différentes méthodes d'authentification, comme l'e-mail/mot de passe, les réseaux sociaux, et gère automatiquement les sessions utilisateur, les réinitialisations de mots de passe, et plus encore. | | | -Externaliser l'authentification offre un support accru en matière de sécurité et de maintenance.  -Le coût de cette solution est nul jusqu'à un certain volume. Au-delà, la facturation est échelonnée, permettant de gérer la croissance de l'activité sans contraintes. |
| Création de menus | Visualisation dynamique des modifications | React | La librairie React permet de créer des modules flexibles et réutilisables qui permet de gérer efficacement l’état de l’application | | | -Il facilite les mises à jour rapides sans rafraîchissement de la page, améliorant ainsi la fluidité et l'expérience utilisateur.  -La popularité de cette bibliothèque garantit un support continu et constant. |
| Upload des images | Gestion du téléchargement et stockage des fichiers | Cloud Storage for Firebase | Gestion du stockage des fichiers de l’architecture Firebase | | | -Sécurité renforcée grâce à l'authentification Firebase.  -Plateforme bénéficiant du soutien de Google. |
| Accéder aux menus précédents et enregistrer un branding | Persistance des données de l’application web | Firebase realtime database | La base de données en temps réel fait partie de l'écosystème Firebase. Il s'agit d'une base de données NoSQL qui utilise le format JSON. | | | -L'utilisation de cette base de données et des SDK Firebase ne requiert ni serveur ni développement d'une API spécifique.  -Les données sont stockées en temps réel et sont accessibles instantanément. |
| Exporter en PDF | En un click | PDFKit | PDFKit est une bibliothèque open-source pour Node.js qui facilite la création de documents au format PDF. | | | -PDFKit est une bibliothèque robuste et largement adoptée pour la génération de PDF en Node.js.  -Elle propose une gamme étendue de fonctionnalités, notamment la gestion des polices, des images et des couleurs. |
| Commander des Impressions de menu | Nouvel onglet vers le back-office de Qwenta | Il semble que davantage d'informations soient nécessaires pour étudier cette spécification. En fonction de ce qui est déjà en place au niveau du back-office de Qwenta, plusieurs éléments pourraient nécessiter une mise en place ou une étude approfondie. | | | | |
| Diffuser le menu sur instagram | Avant l'export vers Instagram, il doit être possible de découper le menu en carrés. | React-to-image | | React-to-image simplifie le processus de conversion des composants React en images en utilisant la bibliothèque html-to-image sous-jacente. En encapsulant les éléments React dans ce composant, il devient facile de les exporter sous forme d'images, ce qui peut être utile pour diverses applications, telles que la génération de captures d'écran ou la création de miniatures. | -React-to-image est compatible avec React et offre une utilisation simplifiée.  -Il permet la création d'images directement à partir de contenu HTML. | |

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

Sur cette application web il n’est pas nécessaire d’utiliser un serveur pour faire le lien entre la DB et l’application au vu du fait que nous utiliserons Firebase.

Pour les fonctions qui auront besoin de code supplémentaire coté Back-end, nous utiliserons Google Cloud Functions.

**API’s :**

Concernant les API. L’API REST de la DB de Firebase va permettre à l’application de communiqué avec la DB.

Cependant il sera nécessaire de consommer des APIs externes comme Menu API de deliveroo ou encore API graph pour instagram.

**Base de données :**

L'utilisation de la base de données Realtime de Firebase offre de nombreux avantages et s'avère particulièrement adaptée à divers types de projets, notamment ceux nécessitant une gestion en temps réel des données. Voici quelques-uns des principaux avantages et caractéristiques de cette base de données :

**NoSQL et JSON :** Firebase Realtime Database est une base de données NoSQL, ce qui signifie qu'elle n'est pas basée sur un schéma fixe comme les bases de données relationnelles traditionnelles. Au lieu de cela, elle utilise le format JSON (JavaScript Object Notation) pour stocker et organiser les données, offrant ainsi une grande flexibilité dans la structure des données.

**Facilité de mise en place :** Firebase Realtime Database ne nécessite pas de configuration complexe ou de gestion de serveur. En utilisant les SDK Firebase disponibles pour différentes plateformes (web, mobile, etc.), vous pouvez facilement intégrer la base de données dans votre application et commencer à stocker et récupérer des données en quelques lignes de code.

**API REST bien documentée :**

Bien que Firebase Realtime Database soit principalement utilisé avec les SDK Firebase, il offre également une API REST complète et bien documentée. Cela signifie que vous pouvez interagir avec la base de données à l'aide de requêtes HTTP standard, ce qui offre une grande souplesse dans le choix des technologies et des plateformes.

**Sécurité :**

Firebase Realtime Database garantit la sécurité des données en n'acceptant que le trafic chiffré via HTTPS. Cela signifie que toutes les données échangées entre votre application et la base de données sont cryptées, ce qui réduit considérablement les risques de violation de la confidentialité ou de l'intégrité des données.

Mise à l'échelle automatique : Firebase Realtime Database est conçu pour s'adapter automatiquement à la charge de votre application. Il peut gérer efficacement des millions d'utilisateurs simultanés tout en maintenant des performances élevées, ce qui en fait une solution idéale pour les applications à forte croissance.

En résumé, l'utilisation de la base de données Realtime de Firebase offre une combinaison unique de facilité d'utilisation, de flexibilité, de performances et de sécurité, ce qui en fait un choix attrayant pour de nombreux projets d'application modernes.

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

* **Nom du domaine :**

Le nom de domaine sera probablement un sous-domaine de Qwenta. Question en attente.

* **Nom de l’hébergement :**

Si aucun contrat plus avantageux et répondant aux besoins de ce projet n'est déjà en place chez Qwenta, la recommandation se tournera vers un hébergeur proposant également un service d'envoi de mails. Idéalement, il serait plus avantageux de choisir le même fournisseur pour la base de données, l'hébergement, l’envoi de mails et la gestion de l'authentification, permettant ainsi de bénéficier des outils de développement associés.

Pour garantir une bonne fiabilité en termes d'uptime et de sécurité, le choix devrait se porter vers une plateforme telle que Google Firebase pour l'hébergement. Une analyse détaillée des tarifs pourra être effectuée en fonction d'une estimation des volumes de Menu Maker, incluant le nombre de visiteurs mensuels, le nombre d'emails de connexion envoyés, l'accès à la base de données et l'espace de stockage de fichiers nécessaire.

Cependant, il convient de noter que Firebase propose une offre gratuite avec des plafonds très raisonnables. Au-delà de ces limites, les offres payantes se font à l'utilisation, ce qui permet de rester flexible en fonction du succès de Menu Maker. Une attention particulière doit être portée au stockage de fichiers, car les restaurateurs auront la possibilité d'uploader leur logo et des photographies des plats.

En optant pour Firebase pour l'hébergement, la base de données, l’envoi des mails et l’authentification de notre application web, nous bénéficions d’une plus grande facilité de développement, et donc d’une réduction du time to market, ainsi que des coûts de production et de maintenance. De plus, la communication entre ces différents services sera plus aisée grâce aux SDK Firebase.

* **Adresses e-mail :**

contact@qwenta.fr

assistance@qwenta.fr

1. **Accessibilité**

* **Compatibilité navigateur :**

L’application dois être compatible avec les toutes dernières versions de Chrome, Opera, Safari, Firefox…

* **Types d’appareils :**

Aucune version mobile prévue pour le moment. Une version desktop first est a produire.

1. **Services tiers**

* **Google cloud storage for Firebase** :

Google Cloud Storage for Firebase offre une solution robuste pour stocker les données, notamment les images destinées à l'impression. Comme mentionné précédemment, il est crucial de prendre en compte la taille des données à stocker. Les images haute résolution nécessaires à l'impression ont des exigences différentes de celles diffusées numériquement.

Il est important de noter que les images destinées à l'impression doivent avoir une résolution minimale de 300 dpi (dots per inch), tandis que les images diffusées en ligne sont généralement à 72 dpi. Pour éviter d'encombrer la base de données, il est préférable de stocker ces images sur un serveur dédié aux fichiers, comme Google Cloud Storage.

De plus, il peut être judicieux de compresser les fichiers avant leur stockage pour réduire leur taille, tout en évitant toute perte de qualité. Google Cloud Storage offre des fonctionnalités de compression et permet une gestion efficace des fichiers, ce qui en fait un choix idéal pour héberger des fichiers multimédias à grande échelle, tout en garantissant des performances optimales et une sécurité robuste.

* **Firebase authentification :**

Externaliser l'authentification vers un service comme Firebase Authentification offre plusieurs avantages significatifs. Tout d'abord, cela renforce la sécurité de l'identité des utilisateurs en confiant cette tâche à un service spécialisé, qui met en œuvre des protocoles de sécurité avancés pour garantir la confidentialité des données d'authentification.

En outre, externaliser l'authentification permet d'éviter les coûts de maintenance réguliers associés à la gestion de la sécurité des comptes utilisateur. Firebase Authentication gère les mises à jour de sécurité et les correctifs de manière transparente, ce qui allège la charge de travail de l'équipe de développement et réduit les risques d'exploitation des vulnérabilités.

De plus, en utilisant un service d'authentification externalisé tel que Firebase, le déploiement du site est accéléré, car il n'est pas nécessaire de développer et de tester des fonctionnalités d'authentification personnalisées. Cela permet de se concentrer davantage sur le développement des fonctionnalités principales de l'application, ce qui peut contribuer à un déploiement plus rapide et à une amélioration de l'expérience utilisateur.

En résumé, externaliser l'authentification vers Firebase Authentication offre une solution sécurisée, évolutive et économique, tout en permettant un déploiement plus rapide et une meilleure concentration sur le développement des fonctionnalités principales de l'application.

* **Firebase Cloud Messaging (FCM) :**

Firebase Cloud Messaging (FCM) est principalement un service de messagerie push, conçu pour envoyer des notifications en temps réel à des appareils mobiles et des applications Web. Bien que FCM ne soit pas spécifiquement destiné à l'envoi de mails pour l'authentification, Firebase propose un autre service appelé Firebase Authentication pour gérer l'authentification des utilisateurs.

Firebase Authentication permet d'intégrer facilement des fonctionnalités d'authentification sécurisée dans une application, en prenant en charge divers mécanismes d'authentification tels que l'authentification par e-mail/mot de passe, par numéro de téléphone, par fournisseur de réseaux sociaux, etc.

Lorsqu'un utilisateur s'authentifie avec succès dans une application utilisant Firebase Authentication, il est possible de déclencher des actions supplémentaires, telles que l'envoi de notifications push via FCM pour informer l'utilisateur de l'état de son authentification ou de toute autre action importante.

En résumé, bien que Firebase Cloud Messaging ne soit pas utilisé pour l'envoi de mails pour l'authentification, Firebase propose des services distincts, tels que Firebase Authentication, pour gérer l'authentification des utilisateurs, et Firebase Cloud Messaging pour envoyer des notifications push à ces utilisateurs une fois qu'ils sont authentifiés.

1. **Recommandations en termes de sécurité**

Il est crucial de définir des durées de vie appropriées pour les liens d'authentification et les tokens de connexion afin de garantir à la fois la sécurité et une expérience utilisateur fluide. Des durées trop courtes peuvent entraîner une augmentation du volume de mails envoyés, tandis que des durées trop longues peuvent présenter des risques de sécurité.

Une suggestion raisonnable serait de fixer une durée de vie d'environ 1 heure pour les liens d'authentification envoyés par mail, ce qui offre un compromis entre la sécurité et la praticité pour les utilisateurs. Quant à la durée de la session connectée, elle pourrait être établie entre 10 et 24 heures, avec une possibilité de discussion et d'ajustement ultérieurs.

Il est également essentiel de valider toutes les données saisies par les utilisateurs pour prévenir les attaques d'injection de code. En veillant à ce que les utilisateurs ne puissent pas saisir de code malveillant dans les champs de saisie de données, on renforce la sécurité de l'application.

La mise en place d'un système de journalisation pour enregistrer les événements système, les erreurs et les activités des utilisateurs est une pratique recommandée. Cela permet de détecter les tentatives d'attaque et de résoudre rapidement les problèmes de sécurité ou de fonctionnement de l'application.

Firebase propose également des outils de test et de monitoring pour les bugs et les crashs, ce qui facilite la détection et la résolution des problèmes techniques.

Enfin, il est important de souligner que l'utilisation du protocole HTTPS est indispensable pour pouvoir accéder à la base de données Firebase, garantissant ainsi la sécurité des échanges de données entre l'application et la base de données.

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

Il est judicieux d'envisager l'ajout de fonctionnalités supplémentaires à Menu Maker pour répondre aux besoins évolutifs des utilisateurs. Voici quelques-unes des fonctionnalités proposées :

-Blog interne à Menu Maker : Permet aux utilisateurs de partager des actualités, des conseils ou des recettes avec la communauté Menu Maker.

-Création de plusieurs profils de branding pour les utilisateurs propriétaires de plusieurs restaurants : Offre la possibilité aux propriétaires de personnaliser l'identité visuelle de chaque restaurant.

-Ajout d'un moyen de paiement : Permet aux clients de payer directement en ligne pour les commandes ou les réservations.

-Intégration des tarifs directement sur Menu Maker : Facilite la gestion et la mise à jour des tarifs des produits et des services proposés par les restaurants.

-Ajout d'animations sur la photo de la bannière et sur les formes géométriques des sections : Améliore l'attrait visuel et l'expérience utilisateur de Menu Maker en intégrant des animations interactives.

En outre, il est essentiel de proposer régulièrement de nouveaux templates de menus pour suivre les tendances et satisfaire les besoins des utilisateurs. Les mises à jour régulières en termes de couleurs, polices et mises en page contribuent à maintenir l'attrait visuel de l'application.

La transition entre le web et l'impression des documents doit également faire l'objet d'une surveillance constante, en tenant compte des mises à jour des normes PDF et en assurant la compatibilité avec les différents supports.

Enfin, il peut être nécessaire de vérifier la colorimétrie des couleurs proposées aux restaurateurs une fois passées du RVB au CMJN lors de l'impression. Une collaboration étroite avec les équipes de Qwenta peut s'avérer nécessaire pour résoudre tout problème potentiel lié à la reproduction des couleurs.