Spécifications techniques

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Auteur** | **Date** | **Approbation** |
| 1.0 | Ilona, Webgencia | 25/01/2024 | John, Qwenta |

Le but de ce document est de définir et justifier les spécifications techniques de “Menu Maker by Qwenta”.

1. **Choix technologiques**

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| Authentification | -Sécurité des informations d'identification des utilisateurs.  -Intégration avec d'autres services Firebase. | Utilisation de Firebase Authentication pour gérer l'authentification des utilisateurs. | -Intégration d'une interface d'authentification avec Firebase Authentication dans l'application.  -Prise en charge de diverses méthodes d'authentification, telles que l'email/mot de passe, l'authentification via des fournisseurs tiers (Google, Facebook, etc.) et l'authentification anonyme si nécessaire.  -Firebase Authentication gère le stockage sécurisé des informations d'identification des utilisateurs et fournit des mécanismes de récupération de mot de passe. | -Firebase Authentication utilise des protocoles sécurisés pour garantir la confidentialité des informations d'identification des utilisateurs, y compris le stockage sécurisé des mots de passe.  - Les méthodes d'authentification tierces offrent une alternative sécurisée, réduisant le risque lié aux mots de passe faibles. |
| Création de menu | -Visualisation dynamique de création | React | 1. Mise en place d'un composant de formulaire de création de catégorie, intégré dans l'interface d'administration.  2.Le formulaire permet aux administrateurs de saisir le nom, la description et d'autres détails pertinents pour la nouvelle catégorie.  3.L'utilisation de composants React offre une gestion modulaire et réutilisable du formulaire. | -La structure modulaire de React permet la création de composants réutilisables, facilitant ainsi l'ajout de nouvelles fonctionnalités sans impact sur le reste du système.  -React permet une mise à jour rapide et dynamique de l'interface utilisateur sans nécessiter de rafraîchissement de la page. |
| Images upload | -Assurer la sécurité du stockage des images.  -Intégration avec l'authentification Firebase pour un accès sécurisé. | Utilisation de Firebase Storage, une composante de l'architecture Firebase, pour le stockage des fichiers, y compris les images. | Mise en place d'un formulaire d'upload d'images dans l'interface utilisateur.  Lorsqu'un utilisateur sélectionne et envoie une image, celle-ci est stockée dans Firebase Storage.  La sécurité est assurée par l'intégration avec l'authentification Firebase, garantissant que seuls les utilisateurs authentifiés peuvent uploader des images. | - L'authentification Firebase garantit que seuls les utilisateurs authentifiés ont accès à la fonction d'upload d'images, renforçant ainsi la sécurité du processus.  -Firebase, en tant que plateforme soutenue par Google, bénéficie d'un support continu, de mises à jour régulières et de l'intégration avec d'autres services Google, offrant ainsi une solution robuste et évolutive pour le stockage d'images. |
| Exportation PDF | Assurer la génération de fichiers PDF de manière efficace. | Utilisation de la bibliothèque jsPDF pour la génération de fichiers PDF dans l'application web. | -Mise en place d'une fonctionnalité d'exportation sur l'interface utilisateur, déclenchée par l'utilisateur lorsqu'il souhaite exporter des données spécifiques.  -Utilisation de jsPDF pour créer dynamiquement un fichier PDF contenant les informations désirées. | -La bibliothèque prend en charge la création de documents PDF à partir de zéro ou en ajoutant du contenu à des documents existants de manière simple.  -L'intégration de jsPDF peut être effectuée de manière transparente avec l'interface utilisateur existante, permettant aux utilisateurs de générer des PDF sans quitter l'application web. |
| Commander des impression | Utilisation d'une bibliothèque d'impression pour simplifier la mise en œuvre. | Utilisation de la bibliothèque Print.js pour gérer le processus d'impression côté client. | -Mise en place d'un bouton "Imprimer" à côté des fichiers PDF sur l'interface utilisateur, déclenchant l'impression lorsque sélectionné (à voir avec le designeur).  -Intégration de Print.js pour faciliter le processus d'impression, en permettant aux utilisateurs d'imprimer directement depuis leur navigateur sans quitter l'interface du site. | -L'utilisation de Print.js simplifie l'intégration de la fonctionnalité d'impression côté client, éliminant ainsi la nécessité de configurer manuellement des scripts d'impression complexes.  -Cette approche garantit une expérience utilisateur fluide et évite des étapes inutiles pour les utilisateurs. |
| Accès aux menus précédents | Minimiser la complexité côté serveur. | Firebase Realtime Database | Realtime Database fait partie  de la boite à outils Firebase. Il  s’agit d’une base de données  non relationnelle en JSON. | -L'utilisation de Firebase Realtime Database simplifie la persistance des données, éliminant la nécessité d'un serveur dédié.  -La gestion côté client minimise la complexité du côté serveur, améliorant ainsi l'efficacité de l'application. |
| Exportation Deliveroo | S'assurer que l'exportation respecte les normes et les formats requis par Deliveroo. | Utilisation des API ou des formats standard pris en charge par Deliveroo. | -Tableau de Bord Restaurateur : Ajout d'une section permettant aux restaurateurs de gérer l'exportation du menu.  -API Deliveroo : Utilisation des API officielles de Deliveroo pour synchroniser les menus de manière efficace.  -Formats Compatibles : Proposer des formats d'exportation compatibles avec les exigences de Deliveroo, comme JSON ou CSV. | L'utilisation des API officielles de Deliveroo garantit la conformité avec les normes et les processus établis par la plateforme.  La fonction d'exportation est conçue de manière conviviale, permettant aux restaurateurs de gérer facilement leurs menus sans avoir à effectuer des ajustements manuels. |
| Partage sur Instagram | Respect des directives et des capacités d'intégration d'Instagram. | -Intégration d'un bouton de partage Instagram dans la section du menu sur le site.  -Utilisation de l'API Instagram ou de liens de partage adaptés. | Génération de liens spécifiques pour le partage sur Instagram, permettant aux utilisateurs de partager directement le menu dans leurs stories ou sur leur profil. | -L'intégration d'une option de partage spécifique à Instagram améliore l'expérience utilisateur en offrant une méthode simple et directe pour partager le contenu.  - Les liens de partage générés facilitent le processus pour les utilisateurs sans nécessiter une compréhension approfondie de la plateforme Instagram. |

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

* **Langages de serveur :**

***Node.js***

Node.js est un environnement d'exécution côté serveur construit sur le moteur JavaScript V8 de Google Chrome.

Il utilise JavaScript, qui est largement utilisé côté client dans les navigateurs web. L'utilisation d'une seule langue (JavaScript) côté client et côté serveur simplifie le développement et la maintenance du code.

Node.js est basé sur un modèle asynchrone et événementiel, ce qui permet de gérer un grand nombre de connexions simultanées sans avoir à créer un thread pour chaque connexion. Cela améliore l'évolutivité et l'efficacité des applications.

Grâce à son moteur V8, Node.js offre des performances élevées en exécutant le code JavaScript rapidement. Il est particulièrement efficace pour les applications nécessitant une manipulation rapide des données en temps réel.

* **API**

***MongoDB Atlas Data API.***  
Par ailleurs, il sera nécessaire de consommer des APIs externes telles que Menu API de Deliveroo, l’API graph de Meta pour Instagram.

* **Base de données choisie :**

***MongoDB*** est un système de gestion de base de données (SGBD) de type NoSQL (Not Only SQL), qui se distingue par son approche orientée document et sa capacité à gérer de grandes quantités de données de manière flexible et évolutive.

Elle utilise un modèle de données orienté document, ce qui offre une flexibilité significative dans la structure des données.

MongoDB est conçu pour l'évolutivité horizontale, permettant d'ajouter de nouveaux serveurs au cluster pour gérer une augmentation de la charge de données. Cette architecture facilite la montée en puissance en ajoutant simplement des nœuds au système.

MongoDB offre des performances élevées, en particulier pour les opérations de lecture et d'écriture. Son modèle de stockage orienté document permet d'accéder rapidement aux données, et ses mécanismes d'indexation contribuent à optimiser les requêtes.

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

* **Nom du domaine :**

Le nom de domaine sera probablement un sous-domaine de Qwenta. Cette question est en cours de validation.

* **Adresses e-mail :**

[contact@qwenta.fr](mailto:contact@qwenta.fr)

[support@qwenta.fr](mailto:support@qwenta.fr)

1. **Accessibilité**

● **Compatibilité navigateur :**

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

**● Types d’appareils :**

Pour l’instant il n’y a pas de version mobile de prévue, seule une version desktop est à produire.

● **Accessibilité :**

Le site doit être accessible au minimum au clavier, et avec un lecteur d’écran.

1. **Services tiers**

* Noms et justification de ceux qui devront être implémentés pour le bon fonctionnement et le suivi du site (par exemple les plugins).

1. **Recommandations en termes de sécurité**

Le développement d'un site web nécessite une attention particulière à la sécurité. Voici un résumé des recommandations en termes de sécurité :

**Protocole HTTPS :**

Utilisez HTTPS pour chiffrer les communications entre le navigateur et le serveur.

**Protection contre les Injections du code :**

Il est crucial de vérifier toutes les données fournies par les utilisateurs afin de prévenir les attaques par injection de code. Il est essentiel de garantir que les utilisateurs ne peuvent pas insérer de code malveillant dans les champs de saisie de données.

**Politiques de Sécurité des Contenus (CSP) :**

Appliquez des politiques de sécurité des contenus pour restreindre les sources de scripts et de médias autorisées.

**Contrôle d'Accès Strict :**

Pour mettre en place un contrôle d'accès strict, définissez des règles qui limitent l'accès aux ressources autorisées en fonction des utilisateurs, des rôles ou d'autres critères. Utilisez des mécanismes d'authentification robustes, attribuez des autorisations spécifiques, et assurez-vous que seuls les utilisateurs autorisés peuvent accéder à certaines parties du système. Utilisez également des sessions sécurisées et des jetons d'accès (accès tokens) pour garantir une authentification fiable et une autorisation appropriée. La mise en œuvre précise dépend de la technologie et du contexte du système, consultez la documentation pertinente pour des détails spécifiques.

**Gestion des Mots de Passe :**

Stockez les mots de passe de manière sécurisée et encouragez l'utilisation de mots de passe forts par des critères obligatoires : 8 caractères minimum, 1 lettre majuscule, 1 lettre minuscule, 1 caractère spécial.

**Surveillance des Journaux :**

Activez la surveillance des journaux pour détecter toute activité suspecte et configurez des alertes en cas de comportement anormal.

**Mises à Jour Régulières :**

Assurez-vous que tous les composants du système sont mis à jour régulièrement pour corriger les vulnérabilités connues.

**Contrôle d'Accès aux Ressources :**

Appliquez des règles strictes de contrôle d'accès aux ressources pour éviter les fuites d'informations sensibles.

**Tests de Sécurité :**

Intégrez des tests de sécurité réguliers, tels que des tests de pénétration, pour identifier et corriger les vulnérabilités.

Ces recommandations contribuent à renforcer la sécurité du site web, protégeant ainsi les données et la confidentialité des utilisateurs contre les menaces potentielles. Il est important de personnaliser ces mesures en fonction des besoins spécifiques du projet et d'adopter une approche proactive en matière de sécurité.

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

**Services Inclus dans la Maintenance :**

* Correction des bugs et erreurs techniques.
* Mises à jour régulières du système et des plugins.
* Surveillance de la sécurité, avec application de correctifs si nécessaire.
* Optimisation des performances pour assurer une expérience utilisateur fluide.

**Fréquence des Mises à Jour :**

Une fois chaque deux mois.

**Nouvelles Fonctionnalités :**

Intégrer un blog interne. Intégrer des modes de paiement.

**Optimisation des Performances :**

Réviser et optimiser le code source pour améliorer les temps de chargement des pages. Mettre en œuvre des techniques de mise en cache pour accélérer l'accès aux ressources du site.

**Intégration des Médias Sociaux :**

Ajouter de nouvelles fonctionnalités d'intégration avec les médias sociaux pour encourager le partage et l'engagement. Mettre à jour les liens vers les profils sociaux.