**Spécifications techniques**

Nom du projet: Menu Maker   
Nom du client : Qwenta

| **Version** | **Auteur** | **Date** | **Approbation** |
| --- | --- | --- | --- |
| 1.0 | Quentin Della mattia | 28/07/2024 | Soufiane, Webgencia |

[I. Choix technologiques](#_1fob9te) 2

[II. Liens avec le back-end](#_3znysh7) 3

[III. Préconisations concernant le domaine et l’hébergement](#_2et92p0) 3

[IV. Accessibilité](#_tyjcwt) 3

[V. Recommandations en termes de sécurité](#_3dy6vkm) 3

[VI. Maintenance du site et futures mises à jour](#_1t3h5sf) 4

## Choix technologiques

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

### 1. Interface Utilisateur

| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| --- | --- | --- | --- | --- |
| **Landing Page** | L’utilisateur non connecté doit pouvoir comprendre l'utilité de cette application. | HTML, CSS, REACT | Utilisation de HTML et CSS pour structurer et styliser la page, et React pour une gestion dynamique du contenu. | 1) HTML et CSS assurent une structure claire et des styles personnalisés.  2) React permet des interactions dynamiques et une flexibilité pour les futures améliorations. |
| **Informations Légales** | Pouvoir accéder au contenu "Mentions légales" dans une modale, et l'information “Tous droits réservés” doit être affichée. | React-modal | Utilisation de React-modal pour afficher le contenu légal dans une modale. | 1) React-modal est cohérente avec l'utilisation de React.  2) Elle est largement adoptée pour la création de modales simples et performants. |

### 2. Navigation et Contenu

| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| --- | --- | --- | --- | --- |
| **Tarifs** | Pouvoir avoir accès aux tarifs de MenuMaker, depuis une URL. | React Router | Utilisation de React Router pour gérer la navigation vers les pages de tarifs. | 1) Permet une gestion dynamique des routes.  2) Largement adopté dans la communauté React pour la gestion des routes. |
| **Dashboard** | Pouvoir avoir accès à un dashboard regroupant : La création de menu, La diffusion de menu, L'impression de menu, Les 3 derniers articles du blog de Qwenta sur MenuMaker. | React, Node.js avec Express.js | Création d'un composant Dashboard en React et mise en place de routes API pour récupérer les articles de blog et les données nécessaires. | 1) Permet une gestion centralisée des fonctionnalités.  2) Offrir une interface utilisateur unifiée pour les actions de gestion. |

### 3. Gestion des Menus, Partage et Export

| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| --- | --- | --- | --- | --- |
| **Création et Modification de Menu** | Pouvoir créer et modifier des catégories de menu depuis une modale  Pouvoir créer et modifier de nouveau plats  Pouvoir modifier et supprimer des menus | React, Node.js avec Express.js, React-Modal, Mongoose | Utilisation de React pour les composants d'interface utilisateur, Express.js pour les opérations CRUD, React-Modal pour la gestion des modales,  Mongoose pour la création de modèle | 1) React permet des interfaces interactives et réactives.  2) Express.js offre un cadre solide pour les opérations API.  3) Mongoose facilite la gestion des données avec MongoDB |
| **Exportation Deliveroo** | Exportation du menu en un clic vers l'application Deliveroo. | API Deliveroo | **Une API** qui permet d’envoyer la demande utilisateur vers la plateforme Deliveroo | 1) Permet une intégration fluide et direct  2) Données transmise directement pour Deliveroo |
| **Exportation PDF** | Exportation du menu doit se faire en PDF | pdfmake | Solution efficace pour générer et télécharger leur menu en PDF | 1) Permet de créer des documents PDF à partir de données dynamiques  2) Intégration facile dans une application React |
| **Partage sur instagram** | Instagram ne permet pas d’intégrer directement le partage d’images via une API publique pour les posts en raison de restrictions sur l'automatisation du contenu. | API Instagram Graph, mais elle nécessite une configuration spécifique | **Une API** qui permet de lier la demande utilisateur vers la plateforme instagram | 1) Permet une intégration fluide et direct  2) Données transmise directement instagram |

### 

### 

### 4. Personnalisation

| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| --- | --- | --- | --- | --- |
| **Personnalisation de Menu** | Le restaurateur connecté doit pouvoir personnaliser le style de son menu avec un choix prédéfini. | Chakra UI | Utilisation de Chakra UI pour offrir une personnalisation flexible des styles et des thèmes. | 1) Chakra UI offre une grande flexibilité pour la personnalisation.  2) Adopte des composants modulaires pour des interfaces accessibles. |

### 5. Notifications et Communication

| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| --- | --- | --- | --- | --- |
| **Envoi et Réception de Mails** | L’utilisateur doit pouvoir recevoir des mails de confirmation et envoyer des e-mails. | Nodemailer | Utilisation de Nodemailer pour envoyer et recevoir des e-mails. | 1) Nodemailer est conçu pour une utilisation simple et efficace.  2) Offre une riche collection de fonctionnalités pour les emails. |
| **Connexion et Déconnexion** | Pouvoir partager le menu sur Instagram, et gérer les connexions/déconnexions des utilisateurs. | Passport.js | Passport.js pour l'authentification des utilisateurs. | 1)  2) Passport.js s'intègre bien avec Express et supporte diverses stratégies d'authentification. |

### 

### 6. Impression

| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| --- | --- | --- | --- | --- |
| **Impression du Menu** | On doit pouvoir commander l'impression du menu en un clic. | React-to-print | Utilisation de React-to-print pour convertir les composants React en documents imprimables. | 1) Intégration facile des fonctionnalités d'impression.  2) Minimise la configuration et facilite la gestion des mises en page imprimées. |

## Liens avec le back-end

* Quel langage pour le serveur ? *NodeJS.*
* A-t-on besoin d’une API ? Si oui laquelle ?
  + Nous aurons besoins de :
    - l’API deliveroo
    - L’API instagram
  + Puis nous aurons besoin interagir avec les données :
    - API RESTful
* Base de données choisie :
  + Pour ce projet, **MongoDB** (une base de données NoSQL) pourrait être un choix judicieux en raison de sa flexibilité avec les schémas de données. Déploiement sur le cloud provider ***AWS***.

## Préconisations concernant le domaine et l’hébergement

* **Nom du domaine :** MenuToMaker.Qwenta.com
* **Nom de l’hébergement :** 
  + Il me faudra un peu plus de détails sur l'hébergeur, l’agence qui comptabilise 3 collaborations donc 3 projets possiblement hébergés sur un même serveur. A voir si vous voulez continuer à héberger sur ce même serveur, ou alors trouver un nouvelle hebergement auquel cas je vous propose une solution :
    - Hébergement du site web sur un serveur DigitalOcean, qui est un bon support pour MongoDb.
* **Adresses e-mail :** [contact@qwenta.com](mailto:contact@menu-to-maker.com)

## Accessibilité

* **Compatibilité navigateur :** Pour le moment, on se contente de la compatibilité avec les dernières versions de Chrome, Safari et Firefox.
* **Types d’appareils :** Le site pour l’instant doit être en version desktop. Une version mobile n'est pas prévue pour le moment.

## Recommandations en termes de sécurité

* Assurez-vous que votre site utilise HTTPS en obtenant et en configurant un certificat SSL. Cela chiffre les données transmises entre le navigateur de l'utilisateur et votre serveur.
* vérification de l'adresse IP implique de s'assurer que les requêtes provenant d'une adresse IP différente sont correctement identifiées.
* Email de confirmation pour vérifier l'identité de l'utilisateur en envoyant un lien ou un code de vérification à son adresse email.
* Double Authentification (2FA) : ajoute une couche supplémentaire de sécurité en demandant à l'utilisateur de fournir deux formes distinctes d'identification avant d'accéder à un compte.
* Gardez tous les packages, bibliothèques, et dépendances à jour pour éviter les vulnérabilités connues.

## **Maintenance du site et futures mises à jour** Grandes lignes du contrat de maintenance.

* Maintenance :
  + Maintenance Corrective : Correction des anomalies et dysfonctionnements signalés par le Client
  + Maintenance Préventive : Actions régulières pour prévenir l’apparition de pannes ou de dysfonctionnements
  + Sauvegardes : Sauvegarde régulière des données du site pour assurer la restauration en cas de perte de données.
  + Maintenance Évolutive : Intégration de nouvelles fonctionnalités ou modifications fonctionnelles demandées par le Client.
* .Mise à jour :
  + Sauvegardes : Sauvegarde régulière des données du site pour assurer la restauration en cas de perte de données.
  + Mise à jour de sécurité : Application des mises à jour de sécurité pour les plugins et autres composants du site.