Spécifications techniques

|  |  |
| --- | --- |
| **Projet** | Menu Maker, Qwenta |

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Auteur** | **Date** | **Approbation** |
| 1.0 | Webgencia | 29/01/2024 | Soufiane |

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)** |
| Landing |  |  |  |  |
| Landing page | Description courte + cta.  Description longue + cta. Explications pas à pas de l’outil | React | React permet de créer des pages et sites web rapidement. | Html + css, serait suffisant pour cette page, mais les autres éléments du site réclameront des unes librairies comme React. |
| Connexion |  |  |  |  |
| Se connecter au BO | Formulaire de connexion dans une modal | React Modal | Permet de créer des modal avec React simplement | 1. Librairie très utilisée 2. Bonne documentation en ligne |
| Mail d’authentification | Se fait par l’envoie d’un email contenant un lien d’authentification. | Nodemailer + React emails | Permet de composer et d’envoyer des mails | 1. Simple d’utilisation 2. Composant React  Compatible avec plusieurs plateformes |
| Contacter l’équipe par mail pour avoir de l’aide | Bouton sur la page de connexion | Nodemailer + React emails | Permet de composer et d’envoyer des mails | 1. Simple d’utilisation 2. Composant React  Compatible avec plusieurs plateformes |
| Déconnexion | Se déconnecter depuis n’importe quelle page | JS : LocalStorage API | Supprimer les informations de connexion. | 1. Native au navigateur internet  2. Simple à utiliser |
| Création de menu |  |  |  |  |
| Un formulaire de création de menu | S’adapte au fur et à mesure de l’Edition du mail | React | Permet de créer des formulaires réactifs | 1. Permet de partager l’état des composants. 2. Rafraichir l’affichage en prenant en compte les changements de l’user. |
| Formulaire catégorie de plat | C’est utilisateur qui doit la renseigner | React | Permet de créer des formulaires réactif | 1. Permet de partager l’état des composants. 2. Rafraichir l’affichage en prenant en compte les changements de l’user. |
| Enregistrement de la catégorie | La catégorie doit être réutilisable | NodeJS + MogoDB | Permet de persister des données | 1. Stack technique assez utiliser.  2. Répond au besoin |
| Formulaire de description de plats | Importer une image et permettre de décrire le plat | React | Permet de créer des formulaires réactif | 1. Permet de partager l’état des composants. 2. Rafraichir l’affichage en prenant en compte les changements de l’user. |
| Enregistrement du menu | Le menu doit être sauvegardé | NodeJS + MogoDB | Permet de persister des données | 1. Une stack très utiliser  2. Bien documenté |
| Menu Live Viewer | Voir en live le rendu du menu | React | Permet de créer des formulaire réactifs | 1. Permet de recevoir l’état d’un composant. 2. Rafraichir l’affichage en prenant en compte les changements de l’user. |
| Exportation PDF | Le menu doit être téléchargeable en PDF | NodeJS + React Pdf | Transforme un composant React en pdf | 1. Module React  2. Facilement configurable |
| Exportation Deliveroo | Convertir le menu en menu Deliveroo | NodeJS + Deliveroo Menu API | Le plus simple à l’heure actuelle | 1. API officiel 2. Offre une flexibilité |
| Partage sur Instagram | Convertir le menu en post Instagram | NodeJS + Instagram Graph API | Le plus simple à l’heure actuelle | 1. API officiel 2. Offre une flexibilité |
| Impression d’un menu |  |  |  |  |
| Commander un menu | Un bouton qui permet de passer l’information de commande au BO de Qwenta | React + NodeJS | Un bouton qui lors ce que le restaurateur clique dessus, envoie sa commande au service de Qwenta. | 1. Récupérer les informations nécessaires depuis la DB 2. Générer une URL avec les informations à envoyer au BO de Qwenta |
| Menus |  |  |  |  |
| Menus précédents | Liste tous les menu Permet de modifier/supprimer un menu Ajouter un menu | React | Une page web listant tous les menus créés précédemment par le restaurateur. | 1. Permet de partager l’état des composants. 2. Rafraichir l’affichage en prenant en compte les changements de l’user. |

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

* Quel langage pour le serveur ?
  + NodeJS.
* A-t-on besoin d’une API ?
  + Oui pour les services tel que Deliveroo Menu API, Instagram Graph API, NodeJs Express
* Base de données choisie :
  + *No SQL avec Mongo DB*.

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

* Nom de domaine  
  Proposition : menu.qwenta.com
* Hébergement  
  Proposition : le même domaine que le site principal,
* Adresse e-mail  
  Proposition : menu-maker@qwenta.com; hello@menumaker-qwenta.com

1. **Accessibilité**

* Pour le moment, on se contente de la compatibilité avec les dernières versions de Chrome, Safari et Firefox.
* Le site devra être en version desktop. Pas de version mobile à développer ni à prévoir.
* L'application devra être accessible au minimum : navigable depuis le clavier, et lisible par un lecteur d'écran.

1. **Services tiers**

* Deliveroo Menu API
  + Pour envoyer le menu sur le compte Deliveroo du client
* Instagram Graph API
  + Pour envoyer l’image généré du menu sur le compte Instagrame du client
* Mongo DB
  + Pour persister les données du compte utilisateur, ses préférences, les menus qu’il aura créés, etc…

1. **Recommandations en termes de sécurité**

* Accès aux comptes, plugins…
  + Encryptions des mots de passe avec Bcript
  + Connexions au services tiers via OAuth 2.0

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

* Grandes lignes du contrat de maintenance.

Il a été convenu ce qui suit :

1. **Objet :**

Le présent contrat a pour objet la maintenance applicative du site web développé par le Prestataire pour le compte du Client, selon les spécifications et les fonctionnalités définies dans le cahier des charges.

1. **Durée :**

La durée du contrat est de [durée], à compter de la date de signature des présentes. Il peut être renouvelé par accord mutuel des parties.

1. **Périmètre de la TMA :**

La Tierce Maintenance Applicative comprend :

La correction des anomalies détectées par le Client ou par le Prestataire.

L'évolution des fonctionnalités existantes du site web.

Le support technique pour l'utilisation du site web.

1. **Responsabilités du Client :**

Le Client s'engage à :

Signaler toute anomalie ou dysfonctionnement dans les meilleurs délais au Prestataire.

Fournir au Prestataire toutes les informations et ressources nécessaires à l'exécution de la TMA.

1. **Responsabilités du Prestataire :**

Le Prestataire s'engage à :

Répondre aux demandes de maintenance dans un délai raisonnable.

Proposer des solutions adaptées aux problèmes rencontrés.

Assurer la confidentialité des données du Client.

1. **Modalités de facturation :**

Les prestations de TMA seront facturées au Client selon les modalités suivantes :

* Vous souscrivez à un contrat incluant un certain nombre de tickets d’intervention, utilisables sur une période d’un an. (Renouvelable)
* Chaque ticket correspond à 15 min d’intervention, toutes catégories confondues. La comptabilisation s’effectue dès qu’un nouveau quart d’heures est entamé.
* Dans le cadre de la TMA corrective ou évolutive, les tickets d’intervention sont facturés 20€ l’unité.

Exemples d’intervention

* De 10 min = 1 ticket de TMA, soit 20€
* De 20 min = 2 tickets de TMA, soit 40€
* D’une heure = 4 tickets de TMA, soit 80€
* De 7 heures = 28 tickets de TMA, soit 560€