Spécifications techniques

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Auteur** | **Date** | **Approbation** |
| 1.0 | Mathieu, Webgencia | 17/01/2024 | John, Qwenta |

Le but de ce document est de définir et justifier les spécifications techniques de [nom du projet].

1. **Choix technologiques**

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Besoin** | **Contraintes** | **Solution** | **Description de la solution** | **Justification (2 arguments)** |
| *Création d’un outil permettant aux restaurateurs de créer, éditer, imprimer et partager des menus* | *Pour pouvoir utiliser le menu maker, les restaurateurs vont devoir se connecter avec un mail et un mdp.* | *Backend: Express.js*  *Frontend: react.js* |  | *1) Langage de base commun (javascript)*  *2) Deux des Framework les plus utilisés au monde*  *3) compréhension faciles et rapide de ces 2 frameworks* |
| *Pouvoir se créer un compte* | *Utilisation simple et rapide* | API REST | *Elle va permettre d’envoyer de nouvelles informations à la base de données et de créer un nouveau compte* | * *Création plutôt simple d’une API* * *Une API REST est multitâche* |
| *Pouvoir se connecter à son compte* | *Utilisation simple et rapide* | API REST | *Elle va permettre de vérifier si les informations donné* | * *Création plutôt simple d’une API* * *Une API REST est multitâche* |
| *Avoir accès à son Dashboard* | *Être connecter à son compte* | React.js | *Utilisation de react pour le frontend* | * *Framework frontend très utilisé* * *Conception rapide* |
| *Créer un menu* | *Ouverture simple en cliquant sur un bouton* | React-modal  + API REST | *La modal va s’ouvrir pour créer un nouveau menu qui sera stocké dans la base de données* | * *React-modal simplifie l’implantation de modal sur react.js* * *L’API va stocker le nouveau menu sur la BDD* |
| *Créer une catégorie de plat* | *Ouverture simple en cliquant sur un bouton* | React-modal  + API REST | *La modal va s’ouvrir pour créer une nouvelle catégorie et choisir son nom* | * *React-modal simplifie l’implantation de modal sur react.js* * *L’API va stocker la catégorie sur la BDD* |
| *Créer un plat* | *Ouverture simple en cliquant sur un bouton* | React-modal  + API REST | *La modal va s’ouvrir pour créer un plat avec l’ajout d’une photo, le nom du plat, une description et le prix.* | * *React-modal simplifie l’implantation de modal sur react.js* * *L’API va stocker le nouveau plat sur la BDD* |
| *Styliser le menu* | L’utilisateur doit pouvoir personnaliser son menu en choisissant la typographie ainsi que la couleur de son menu | React.js + API REST | Utilisation de React pour créer la stylisation + API REST pour envoyer les informations de style choisi à la base de données. | * On utilise React pour créer la stylisation + * API REST pour envoyer les informations de style choisi à la base de données. |
| Exporter son menu en pdf | L’utilisateur doit pouvoir exporter ses menus au format pdf. | React-pdf | React-pdf permet de générer des documents PDF directement dans un navigateur web | * Facile à intégrer (documentation en ligne) * Code très léger. |
| Exporter le menu sur Instagram | L’utilisateur doit pouvoir exporter son menu sur Instagram via le menumaker | API Graph Instagram | L'API Graph est idéale pour diffuser des contenus, les identifier et répondre aux commentaires. | * API officielle Instagram * Utilisation et configuration très simple |
| Exporter le menu sur Deliveroo | L’utilisateur doit pouvoir exporter son menu sur Deliveroo via le menumaker | MENU API Deliveroo | L'API Menu vous permet de gérer votre menu Deliveroo depuis un système tiers. | * API officielle Deliveroo * Update les menus des restaurants directement |

1. **Liens avec le backend**

* Quel langage pour le serveur ? Node.js pour utiliser le Framework Express.js coté serveur.
* A-t-on besoin d’une API ? Si oui laquelle ? API REST + API IG & Deliveroo
* Base de données choisie : MongoDB car il utilise javascript comme langue.
  + Solution globale : MERN (MongoDB, Express, React.js, Node.JS) qui fonctionne avec javascript autant en frontend qu’en backend.

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

* Nom du domaine : [www.menumaker.qwenta.com](http://www.menumaker.qwenta.com) (sous domaine pertinent)
* Adresses e-mail : [contact@menumaker.qwenta.com](mailto:contact@menumaker.qwenta.com) (permet de pas mélanger les autres activités de qwenta)
* Hébergeur web :  [Hostinger](https://www.hostinger.fr/hebergement-web) (10€ par mois avec full service), hébergeur français.

1. **Accessibilité**

* Compatibilité navigateur : Focus principal sur la compatibilité des navigateurs Chrome, Firefox et Safari (représente 9/10 des navigateurs desktop)
* Types d’appareils. : desktop, pas de version mobile à prévoir dans le dev)

1. **Services tiers**

* API Deliveroo
* API Instagram
* Google Analytics
* Pixel Meta
* Futur : Paiement via Stripe ou PayPal

1. **Recommandations en termes de sécurité**

* HTTPS : protocole http sécurisé entre le navigateur et le site web.
* Authentification client : demande de MDP complexe (avec Majuscule, chiffre et autres)
* Utilisation des SSL et d’un CDN
* Sécurité via l’hébergeur
* Respect de la RGPD

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

* Maintenance : Une demi-journée de maintenance par mois peut être nécessaire pour de la correction de bug futur, mise à niveau mensuel du SEO et maintenance générale du site.
* Futures MAJ :
  + Version responsive du menu maker
  + Ajout de nouveaux exports du menu (Uber Eats, Facebook)
  + Création d’un blog
  + Création de landing page pour de SEA