



Webgencia

Spécifications techniques

Projet	Menu Maker by Qwenta		
Version	Auteur	Date	Approbation
1.0	Amandine, Webgencia	05/08/2023	Soufiane, Webgencia

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

I. Choix technologiques

- État des lieux des besoins fonctionnels et de leurs solutions techniques

Besoin	Contraintes	Solutions	Description de la solution	Justification
Affichage d’un descriptif de l’application Menu Maker.	Un internaute (utilisateur non connecté) doit pouvoir avoir accès aux différentes sections de la page d’accueil : la bannière, la section « Personnalisez votre menu » et la section « Explications étape par étape ». Améliorations possibles : Animations sur la photo de la bannière et sur les	HTML / CSS	Utilisation des langages standards HTML et CSS.	Langages les plus adaptés et validés par le W3C.
		JS	Langage JS pour la création de pages dynamiques.	Nécessaire pour la création de pages Web dynamiques.
		React	Librairie React proposant de nombreux	React permet un développement



Webgencia

	formes géométriques des sections.		composants responsives et personnalisables.	efficace et performant.
Création d'un compte et connexion.	<p>Un internaute doit pouvoir créer un compte depuis la page d'accueil en cliquant sur un des liens « Se connecter ».</p> <p>Un restaurateur inscrit doit pouvoir se connecter à son compte en cliquant sur l'un des mêmes liens.</p> <p>Dans les deux cas, une même modale doit s'ouvrir afin de proposer à l'utilisateur de saisir une adresse mail.</p> <p>Lorsque l'utilisateur clique sur le bouton « Se connecter » de la modale, un e-mail doit lui être envoyé.</p> <p>A partir de cet e-mail, l'internaute doit pouvoir confirmer son adresse mail, et ainsi automatiquement créer un compte et se connecter ; tandis que le restaurateur inscrit doit pouvoir s'authentifier et se connecter.</p> <p>La modale doit proposer un lien « Besoin d'aide ? » permettant à l'utilisateur d'envoyer directement un e-mail à nos équipes.</p>	react-modal	Librairie permettant de créer des modales personnalisables avec un minimum de code.	Librairie très utilisée, performante et compatible avec notre choix de développer en React.
		API Web Cryptography	API permettant la gestion du cryptage des données.	API recommandée par le W3C.
		MangoBD	Utilisation d'une base de données MangoDB pour la sauvegarde des données des restaurateurs.	Base de données NoSQL, recommandée pour les modèles de données simples et les applications web en temps réel.
		Jsonwebtoken	Package permettant de créer et vérifier les tokens d'authentification.	Compatible avec le type de serveur que nous avons choisi (NodeJS).
		react- fontawesome	Composant React permettant l'utilisation d'une librairie d'icônes.	Composant qui permet d'afficher les icônes de la librairie Font Awesome, très utilisée en web design.



Webgencia

Création d'une catégorie de plat.	L'ajout d'une catégorie de plat doit pouvoir se faire depuis la page de création de menu via une modale spécifique. Cette modale doit contenir un champ de saisi et un bouton de validation.	react-modal	Librairie permettant de créer des modales avec un minimum de code.	Librairie très utilisée, performante et compatible avec notre choix de développer en React.
Ajout d'un plat / des plats dans un menu.	L'ajout d'un plat / des plats dans un menu doit se faire depuis la page de création de menu. Après avoir préalablement sélectionné une catégorie de plat, le restaurateur doit pouvoir ouvrir une modale pour ajouter des plats. Dans cette modale, le restaurateur doit pouvoir ajouter pour chaque plat : une photo, un nom, un prix et une description. L'utilisateur doit avoir la possibilité de scroller dans la modale pour voir toutes les informations affichées.	react-modal	Librairie permettant de créer des modales avec un minimum de code.	Librairie très utilisée, performante et compatible avec notre choix de développer en React.
		File Upload	Composant React permettant de gérer le chargement d'un fichier et notamment de le prévisualiser.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.
		react-scroll	Librairie permettant le défilement du contenu d'un composant React.	Librairie facile à utiliser et compatible avec notre choix de développer en React.
Personnalisation du style du menu.	La personnalisation du menu doit pouvoir se faire depuis la page de création de menu, après avoir ajouté des plats. Le restaurateur doit pouvoir visualiser son menu actuellement créé, sélectionner une typographie et choisir une couleur de texte.	Google Fonts	Librairie gratuite de polices de caractères.	Librairie très populaire proposant de nombreuses polices.
		Color Picker	Composant React permettant à l'utilisateur de sélectionner une couleur sur une palette.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.



Webgencia

Exportation du menu en PDF.	Le restaurateur doit pouvoir exporter le fichier PDF correspondant à son menu en cliquant sur l'encart « Exporter en PDF » situé dans la catégorie « Exportez & diffusez ».	Wkhtmltopdf	Outil permettant de convertir du HTML en PDF.	Facile à utiliser avec le type de serveur que nous avons choisi (NodeJS).
Commande d'impression d'un menu.	Le restaurateur doit pouvoir commander l'impression d'un menu depuis l'encart "Imprimer un menu" visible depuis la page d'accueil. Au clic, le lien doit ouvrir un nouvel onglet vers le back-office de Qwenta.	Lien vers une url de Qwenta	Hébergement de la page HTML sur un serveur Qwenta.	Maintenance de la page directement par Qwenta.
Accès à une vue regroupant les menus précédemment créés.	Le restaurateur doit pouvoir accéder à une vue regroupant les menus précédemment créés depuis le lien « Mes menus ». Chaque menu est affiché avec sa date de création. Cette vue doit permettre au restaurateur de créer, modifier et supprimer un menu.	Dashboard Layout	Composant React permettant de facilement créer un dashboard.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.
		Button	Composant React permettant de créer et personnaliser un bouton.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.
Affichage des mentions légales et de l'information « Tous droits réservés ».	L'internaute (non connecté) et le restaurateur connecté doivent pouvoir afficher les mentions légales dans une modale en cliquant sur le lien « Mentions légales » visible, respectivement, sur la page d'accueil et sur le dashboard. La mention « Tous droits réservés » doit figurer sur la page d'accueil et sur toutes les autres pages.	react-modal	Librairie permettant de créer des modales personnalisables avec un minimum de code.	Librairie très utilisée, performante et compatible avec notre choix de développer en React.



Webgencia

Accès aux tarifs.	L'internaute doit pouvoir accéder aux tarifs de l'outil Menu Maker en cliquant sur le lien « Tarifs » visible sur la page d'accueil. Au clic, un nouvel onglet doit s'ouvrir en utilisant une url de Qwenta.	Lien vers une url de Qwenta	Hébergement de la page HTML sur un serveur Qwenta.	Maintenance de la page et mise à jour des tarifs directement par Qwenta.
Exportation d'un menu vers l'application Deliveroo.	Le restaurateur doit pouvoir exporter un menu vers l'application Deliveroo en cliquant sur l'encart « Diffuser sur Deliveroo » situé dans la catégorie « Exportez & diffusez ». Au clic, le restaurateur doit être redirigé sur l'application Deliveroo.	API Menu	Utilisation de l'API Menu pour publier un menu sur le compte professionnel Deliveroo d'un restaurateur.	Seule API dédiée à la publication et mise à jour de menus sur Deliveroo.
Partage d'un menu sur Instagram.	Le restaurateur doit pouvoir partager un menu sur Instagram en cliquant sur l'encart « Partager sur Instagram » situé dans la catégorie « Exportez et diffusez ». Au clic, le restaurateur doit être redirigé vers son compte Instagram avec les photos du menu reformattées au format compatible, carré.	API Graph	Utilisation de l'API Graph pour publier une image (au format JPEG) sur le compte professionnel Instagram d'un restaurateur.	Seule API dédiée à la publication de media sur Instagram.
Déconnexion.	Le restaurateur doit pouvoir se déconnecter depuis n'importe quelle page connectée en cliquant sur le lien « Se déconnecter ».	Jsonwebtoken	Package permettant de créer et vérifier les tokens d'authentification.	Compatible avec le type de serveur que nous avons choisi (NodeJS).
Modification des informations utilisateur.	Le restaurateur doit pouvoir modifier ses informations utilisateur en cliquant le lien « Mon compte »	API Web Cryptography	API permettant la gestion du cryptage des données.	API recommandée par le W3C.



Webgencia

	<p>visible depuis la page d'accueil de son compte. Il doit pouvoir modifier son adresse mail de base et lier plusieurs adresses mails à son compte.</p> <p><u>Améliorations possibles</u> : Possibilité de modifier le moyen de paiement.</p>	StripeJS	Package permettant de facilement vérifier des coordonnées bancaires.	Outil permettant une gestion simple, rapide et sécurisé des paiements en ligne.
Affichage d'un dashboard.	<p>Lors de la connexion, le restaurateur doit être dirigé vers la page d'accueil de son compte (dashboard). Depuis cette page, le restaurateur doit pouvoir avoir accès aux encarts « Créer un menu », « Diffuser un menu » et « Imprimer un menu », ainsi qu'à une section « Pour aller plus loin » contenant les trois derniers articles du blog de Qwenta qui parlent de Menu Maker. Chaque article doit être affiché avec un titre, une photo de couverture et un lien.</p> <p><u>Améliorations possibles</u> : Création d'un blog interne à Menu Maker.</p>	Dashboard Layout	Composant React permettant de facilement créer un dashboard.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.
Création d'un Branding.	<p>Le restaurateur doit pouvoir créer le branding de son restaurant. En cliquant sur le lien « Mon restaurant » de son dashboard, le restaurateur doit pouvoir ajouter,</p>	NextJS	Framework React qui permet de faire de la génération de pages statiques, idéal pour faire un blog.	Framework facile à utiliser et cohérent avec la librairie React que nous avons choisi d'utiliser.
		Image Editor	Composant React permettant de facilement modifier une image.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.



Webgencia

	modifier et supprimer son logo et ses couleurs de base.	Color Picker	Composant React permettant à l'utilisateur de sélectionner une couleur sur une palette.	Composant disponible dans la librairie React que nous avons choisi d'utiliser.
--	---	--------------	---	--

II. Liens avec le back-end

- Serveur

- Langage **NodeJS** (<https://nodejs.org/fr>).
- Utilisation du framework **Express** (<https://expressjs.com/fr>) pour faciliter la création et la gestion du serveur.
- Installation du package **Nodemon** (<https://nodemon.io>) qui permettra de surveiller plus facilement les modifications des fichiers et de redémarrer le serveur lorsqu'il aura besoin d'être mis à jour.

- API

Création d'une **API REST** (<https://docs.github.com/en/res>) pour le transfert de données, basé sur le protocole HTTP.



- **Base de données**

Utilisation de **MongoDB**, une base de données NoSQL et open source, plus évolutive et flexible qu'une base de données relationnelle SQL. Les données sont stockées sous forme de document clé-valeur au format JSON (<https://www.mongodb.com/docs>).

III. Préconisations concernant le domaine et l'hébergement

- **Nom du domaine**

Le nom de domaine sera un sous-domaine de Qwenta tel que : www.MenuMakerByQwenta.fr (en cours de validation).

- **Nom de l'hébergement**

L'outil Menu Maker sera hébergé chez Hostinger, un hébergeur Web très populaire grâce à ses multiples offres d'hébergement et son prix très attractif (<https://www.hostinger.fr>).

- **Adresses e-mail**

- de contact : contact@MenuMakerByQwenta.fr
- de maintenance : maintenance@MenuMakerByQwenta.fr

IV. Accessibilité

- **Compatibilité navigateur**

Dans un premier temps, l'outil Menu Maker devra être compatible avec les dernières versions de Chrome, Safari et Firefox.

- **Types d'appareils**

Le site devra être en version desktop. Pas de version mobile à développer ni à prévoir.
L'application devra être navigable depuis le clavier et lisible par un lecteur d'écran.

V. Services tiers

- **APIs**

- **API Web Cryptography** : qui permet l'accès aux fonctionnalités cryptographiques côté client (<https://www.w3.org/TR/WebCryptoAPI>).
- **API Menu** : dédiée à la publication et la mise à jour de menus (incluant images, prix, points de vente, informations sur les allergènes et stocks), disponible sur le portail développeur Deliveroo (<https://api-docs.deliveroo.com/docs/menu-api-overview>).
- **API Graph** : dédiée à la publication de media sur des comptes professionnels Instagram, disponible sur la plateforme Instagram (<https://developers.facebook.com/docs/instagram-api>).



Webgencia

- **Librairies**

- **ReactJS** : Librairie qui facilite la création d'interfaces utilisateurs interactives en proposant de nombreux composants responsives et personnalisables (<https://fr.legacy.reactjs.org>).
- **React-modal** : Librairie permettant de créer des modales personnalisables avec un minimum de code (<https://reactcommunity.org/react-modal>).
- **React-scroll** : Librairie permettant le défilement du contenu d'un composant React (<https://www.npmjs.com/package/react-scroll>).
- **Google Fonts** : Librairie gratuite de polices de caractères (<https://fonts.google.com>).

- **Packages**

- **Jsonwebtoken** : Package permettant de créer et vérifier les *tokens* d'authentification (<https://jwt.io>).
- **StripeJS** : Package permettant une gestion simple, rapide et sécurisé des paiements en ligne (<https://stripe.com/fr>).

- **Autres outils**

- **Wkhtmltopdf** : Outil open source (LGPLv3) pour convertir du HTML en PDF à l'aide du moteur de rendu Qt WebKit (<https://www.npmjs.com/package/wkhtmltopdf>).
- **NextJS** : Framework React qui permet de faire de la génération de pages statiques, idéal pour faire un blog (<https://nextjs.org>).



Webgencia

VI. Recommandations en termes de sécurité

- L'application respectera le *Règlement Général sur la Protection des Données* (RGPD) qui veille à garantir la sécurité des données personnelles stockées et échangées.
- L'application sera conforme à la norme *Payment Card Industry Data Security Standard* (PCI DSS) veillant à la sécurité lors de la transmission, du traitement et du stockage des données bancaires.
- Les vulnérabilités de l'application seront testées selon le Top Ten de l'OWASP (*Open Web Application Security Project*).
- L'application sera testée selon l'ASVS (*Application Security Verification Standard*) et devra remplir tous les critères de sécurité pour se placer au niveau deux : niveau jugé comme standard, atteint lorsque l'application se défend contre la plupart des risques associés aux logiciels actuels.
- HTTPS, qui utilise les protocoles *Secure Sockets Layer/Transport Layer Security* (SSL/TLS), sera utiliser pour chiffrer les transmissions de données.
- Le code de l'application sera sécurisé contre les attaques par injection grâce à une validation systématique des entrées et des variables paramétrées.
- Afin de sécuriser, l'accès aux comptes utilisateurs, il sera mis en place une authentification à double facteur (2FA ou MFA). Un mail sera envoyé à chaque tentative de connexion afin que l'utilisateur s'authentifie correctement avant de se connecter.
- Le mot de passe utilisateur devra respecter les mesures de sécurité préconisées par le CNIL (<https://www.cnil.fr/fr/mot-de-passe>) : 8 caractères minimum avec au moins une majuscule, une minuscule, un caractère spécial et un chiffre.
- Nous utiliserons l'API Web Cryptography pour sécuriser et cryptographier les données côté client (<https://www.w3.org/TR/WebCryptoAPI>).
- Nous utiliserons le package `jsonwebtoken` afin de créer et vérifier les *tokens* d'authentification (<https://jwt.io>).



Webgencia

VII. Maintenance du site et futures mises à jour

- Webgencia assurera la maintenance de l'application Menu Maker.
- Webgencia s'engage à :
 - faire en sorte que l'application soit constamment en bon état de fonctionnement.
 - remédier à toute anomalie de fonctionnement.
 - apporter son assistance à Qwenta en cas d'accident résultant du dysfonctionnement de l'application.
 - signaler à Qwenta toute modification apportée au logiciel dans le cadre de son entretien et remettre les documents relatifs à cette intervention.
 - effectuer une révision de l'application qui serait obligatoire en raison d'un changement des dispositions légales ou réglementaires. Si cette révision est impossible en raison des caractéristiques du système informatique de Qwenta, Webgencia s'engage à le lui signaler.
- Qwenta ne pourra pas exiger de Webgencia les tâches suivantes :
 - reconstitution de tout fichier détruit accidentellement.
 - sauvegarde d'un fichier quelconque.
 - entretien d'une application autre que Menu Maker.
- Webgencia s'engage à respecter la confidentialité la plus totale en ce qui concerne le contenu de ce projet.
- La responsabilité de Webgencia pourra être engagée s'il est établi qu'il a manqué à son obligation contractuelle. En revanche, elle ne pourra pas être engagée en cas de retard résultant d'une cause indépendante de sa volonté ou si Qwenta omet de lui transmettre une information nécessaire pour le projet.