

Spécifications techniques

Projet Menu Maker pour Qwenta

| | Auteur | Date | Approbation |
|-----|--------------|------------|------------------|
| 1.0 | Moret Jordan | 23/04/2025 | Soufiane, Qwenta |

I. 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'une page de connexion/d'inscription | Si l'utilisateur n'a pas de compte connu par le site, il peut créer un compte automatiquement avec son adresse mail. Un mail est ensuite envoyé pour confirmer son adresse mail pour accéder à l'application pour la première fois. Une fenêtre modale s'ouvre pour la connexion / l'inscription. Enfin, un lien « Besoin d'aide » permet de faire le lien avec nos équipes si le visiteur a une question. | Côté front-end : utilisation de React.js et plus précisément de react-modal puisque des fenêtres d'authentification et/ou de création de compte vont s'ouvrir. Côté back-end : Node.js avec Express, utilisation de tokens (JWT), bcrypt. Au niveau de la base de données, utiliser MongoDB. | React.js est un framework moderne pour des interfaces dynamiques. Au niveau du backend, il est important d'utiliser des tokens pour l'authentification ainsi que de hasher les mots de passe avec bcrypt. Enfin, OAuth2 permet aux visiteurs de s'authentifier avec Google, Facebook... | 1) L'utilisation de react-modal est ici nécessaire pour avoir un dynamisme et une fluidité au niveau du changement de page lors de la connexion/inscription. 2) Les éléments du back-end permettent d'avoir une sécurité robuste mais en même temps assez de ressources pour que les utilisateurs puissent s'inscrire ou se connecter facilement de manière efficace. |
| Landing non connectée | L'internaute doit pouvoir avoir accès aux différentes sections de | Utilisation de React (react-router) et de CSS/SASS : on va alors | La combinaison des deux permet de gérer une interface dynamique et efficace, en faisant le lien | 1) Bibliothèque react très populaire chez les développeurs web. 2) Utilisation de react-router |

| | | | | |
|---|--|--|---|--|
| | la landing page non connectée : Bannière Personnalisez votre menu Explications étape par étape | obtenir un affichage dynamique et efficace. Développer l'interface visuelle complète en donnant la possibilité aux visiteurs de visualiser le site de manière globale. Déterminer le contenu qui est accessible sans connexion, faire attention aux routes protégées ou non. | entre les différentes sections. | pour gérer de manière dynamique les différentes pages présentes sur le site. |
| Création d'une catégorie de menu | L'ajout d'une catégorie doit pouvoir se faire directement sur l'écran de création de menu depuis une modale. | Utilisation de React et notamment de react-modal puisqu'une modale s'ouvre lorsque l'utilisateur clique sur le bouton d'ajout d'une catégorie. Sauvegarder la catégorie | Cette librairie React permet de créer simplement des modales performantes, accessibles avec un minimum de code. | 1) Nous avons choisi de développer en React, la librairie est cohérente avec ce choix. 2) Il s'agit de la librairie la plus utilisée. |

| | | | | |
|-------------------------|---|---|---|--|
| | | dans la base de données. | | |
| Création de plat | <p>Le restaurateur doit pouvoir :</p> <p>Sélectionner une catégorie</p> <p>Modifier la catégorie sélectionnée</p> <p>Entrer les informations de son plat</p> <p>Une modale s'ouvre pour saisir les différents plats de la catégorie</p> <p>Chaque plat peut avoir :</p> <p>Une photo associée</p> <p>Un nom</p> <p>Un prix</p> <p>Une description</p> <p>Il est possible de créer autant de plats que l'on souhaite</p> | <p>React-modal avec choix interactifs, affiche dynamique.</p> <p>Ouverture d'une modale avec un formulaire composé de plusieurs types d'input (file, texte...).</p> <p>Créer un schema avec mongoose.Schema.</p> <p>Gestion des routes (notamment la route POST puisque c'est un ajout de données) et de l'obligation d'authentification.</p> <p>Gérer l'upload de l'image ajoutée avec "multer".</p> <p>Affichage dynamique des données suite à la validation du visiteur.</p> | <p>On utilisera ici react-modal puisqu'une modale s'ouvre lorsque l'utilisateur veut créer son plat, une page d'information s'ouvre avec différentes infos sur le plat.</p> | <p>1) L'utilisation de react-modal permet ici au restaurateur de créer un plat avec toutes les infos nécessaires, de manière simple et dynamique.</p> <p>2) La modal est une interface supplémentaire qui permet une différenciation par rapport à la page précédente, ce qui permet de démarquer la vue globale de l'édition et donc l'ajout d'un plat.</p> |

| | | | | |
|---|---|---|---|---|
| Styliser le menu restaurateur | <p>Le restaurateur doit pouvoir :</p> <p>Visualiser le menu actuellement créé</p> <p>Sélectionner une typographie</p> <p>Choisir une couleur de texte</p> | Utilisation de React, CSS3/SASS et une bibliothèque de polices type Google Fonts. | L'utilisation combinée de ces éléments permettra une visualisation en temps réelle sur les éléments changés (couleur et typographie). | <p>1) Une identité visuelle peut être trouvée par le restaurateur à travers l'utilisation de CSS pour la couleur et de bibliothèque de police pour trouver la police qui correspond à ses goûts.</p> <p>2) Le fait d'effectuer une visualisation en temps réel permet à l'utilisateur de prendre des décisions instantanément et a tous les outils en main pour prendre ses décisions de manière rapide et intuitive.</p> |
| Exportation du menu en PDF | Le restaurateur doit pouvoir en un clic télécharger le fichier PDF correspondant à son menu | Bibliothèque JavaScript jsPDF. | Création de fichier PDF côté client. | <p>1) Utilisation facile et supporte les éléments présents dans le menu créé auparavant (images, texte et style).</p> <p>2) Compatible avec React, un framework moderne.</p> |
| Commander l'impression d'un menu | Il s'agit d'un lien qui s'ouvre dans un nouvel onglet | Ajout d'un lien vers le backoffice de Qwenta. Ce lien doit être | L'utilisation de la balise HTML « target= « _blank » » permet d'insérer directement le lien vers le | 1) Lien facile d'accès et cliquable directement sur la page d'accueil. |

| | | | | |
|-----------------------------|--|---|--|---|
| | Le lien doit être fait vers le back-office de Qwenta | visible et facile d'accès. | back office de Qwenta. | 2) Le lien back-office permet de maintenir une sécurité et de proposer des fonctions spécifiques aux utilisateurs concernés. |
| Menus précédents | <p>Au clic sur "Mes menus", le restaurateur doit avoir accès aux menus créés précédemment</p> <p>La date de création s'affiche</p> <p>Il est possible de modifier un menu précédent</p> <p>Il est possible de supprimer un menu précédent</p> <p>Sur la même vue, le restaurateur doit pouvoir créer un nouveau menu</p> | Interface en React permettant d'avoir une vue globale sur les menus précédents. Stocker les menus précédents dans MongoDB et gestion des opérations CRUD. | La page affichée regroupe tous les menus précédemment créés. On aura donc besoin de conserver les données des précédents menus et de les modifier si besoin, l'utilisation d'une base de données est alors requise. Pour naviguer sur cette page, react-router est à nouveau utiliser pour faire la connexion entre les pages. | <p>1) React pour un affichage fluide et dynamique des menus précédemment créés.</p> <p>2) La base de données permet de sauvegarder tous les menus créés précédemment.</p> |
| Informations légales | L'internaute doit pouvoir cliquer sur "Mentions légales" depuis les | React-modal puisqu'une modale s'ouvre avec les | Le lien « Mentions légales » renvoie sur une modale qui affiche les informations légales. | 1) Les liens « mentions légales » présents sur toutes les pages du site montrent au |

| | | | | |
|---------------|---|---|--|---|
| | <p>pages connectées et déconnectées</p> <p>La modale s'ouvre alors</p> <p>La mention "Tous droits réservés" doit figurer sur la page d'accueil et sur toutes les autres pages</p> | <p>informations légales du site, implanter cette modale sur toutes les pages.</p> | | <p>visiteur du site la transparence de Qwenta sur la légalité de leur travail. Une preuve de confiance que l'utilisateur peut saisir.</p> <p>2) La modale permet une nouvelle fois au visiteur de différencier la page sur laquelle il était avec ce qui a été cliqué, une façon cohérente d'afficher les choses.</p> |
| Tarifs | <p>L'internaute doit pouvoir cliquer sur l'onglet Tarifs, et ouvrir un nouvel onglet</p> | <p>Ajout du lien : https://URL DE QWENTA/tarifs/enumaker avec l'attribut HTML « target= _blank ».</p> | <p>Le lien renverra sur l'onglet des tarifs, ce qui permettra à l'utilisateur de garder sa page précédente ouverte, tout en accédant au tarif sur une page différente.</p> | <p>1) Le lien apporte de la cohérence et du dynamisme au site.</p> <p>2) Le visiteur aura toutes les informations sur les tarifs sur une page spécialement dédiée à cela, pour ne pas être dans l'incompréhension d'informations affichées sur une même page.</p> |

| | | | | |
|------------------------------|---|--|--|--|
| Exportation Deliveroo | <p>L'encart "Diffuser sur Deliveroo" doit s'afficher dans la catégorie "Exportez et diffusez"</p> <p>Au clic sur l'encart, l'utilisateur doit être redirigé sur l'application Deliveroo</p> | <p>Ajout d'un bouton qui redirige automatiquement vers l'application Deliveroo. Interaction avec l'API Deliveroo pour synchroniser les données.</p> | <p>Au niveau de l'exportation, un bouton « Diffuser sur Deliveroo » permettra de faire le lien avec l'application Deliveroo et donc de réaliser l'exportation.</p> | <p>1) La redirection immédiate vers l'application « Deliveroo » permet au visiteur d'avoir une exportation fluide de la page internet vers l'application, ce qui améliore l'expérience utilisateur.</p> <p>2) Le bouton ajouté au niveau de l'exportation permet au restaurateur de voir directement les choix d'exportation qui lui sont offerts.</p> |
| Partage sur Instagram | <p>L'encart "Partager sur Instagram" doit s'afficher dans la catégorie "Exportez et diffusez"</p> <p>Au clic sur l'encart, des images du menu au format carré (pour les mettre sur Instagram) sont générées</p> <p>Le restaurateur est redirigé vers son compte</p> | <p>Le menu est généré en images carrées pour faciliter les posts sur Instagram.</p> <p>Possibilité de donner des paramètres aux images de manière dynamique (html2canvas). Interaction avec l'API d'Instagram.</p> | <p>Au niveau de l'exportation, un bouton « Partager sur Instagram » sera ajouté pour qu'au clic de ce dernier, le restaurateur verra son menu généré en images au format carré puis se connectera à son compte Instagram pour pouvoir partager ses menus sur le réseau social.</p> | <p>1) La génération d'images carrées permet encore une fois d'améliorer l'expérience utilisateur pour que tout soit optimisé au partage souhaité.</p> <p>2) Le bouton d'exportation « Partager sur Instagram » est encore une fois visible et facile d'accès et permet au restaurateur de partager facilement son menu.</p> |

| | | | | |
|---------------------------------|--|--|--|--|
| | Instagram avec les photos carrées des menus | | | |
| Déconnexion | Le restaurateur doit pouvoir se déconnecter depuis n'importe quelle page connectée | Un bouton « Déconnexion » doit apparaître sur les pages lorsque le restaurateur est connecté. Une fois ce bouton cliqué, la session de l'utilisateur doit être supprimée (via token par exemple). Un message de confirmation de déconnexion doit être affiché. | La déconnexion du compte du restaurateur sera effective une fois le bouton « Se déconnecter » cliqué. Ce bouton sera facile d'accès. | 1) Le bouton doit être facile d'accès pour que le visiteur puisse se déconnecter n'importe où et à n'importe quel moment. 2) L'expérience utilisateur est renforcée, le bouton est lisible et dans l'en-tête, ce qui nous donne une mise en page cohérente. |
| Informations utilisateur | Lier plusieurs adresses e-mail à son compte Modifier son adresse e-mail de base | Stockage et gestion sécurisée des données utilisateur à l'aide d'une base de données (interface visuelle à effectuer avec possibilité de | Une interface de gestion des adresses mails utilisées doit être mise en place (suppression, ajout, modification des adresses mail). | 1) La possibilité de modifier les adresses mails renseignées, dont l'adresse mail principale permet au restaurateur de se connecter sur le long terme, même si son adresse mail a été modifiée. |

| | | | | |
|------------------|--|---|---|---|
| | | modification). Penser à une sécurité forte de ces données sensibles. | | 2) Le fait d'accepter plusieurs adresses mails permet à une équipe d'un restaurant par exemple de pouvoir avoir accès au compte et donc d'apporter les modifications nécessaires sur ce qu'ils veulent. |
| Dashboard | <p>À l'encart "Créer un menu"</p> <p>À l'encart "Diffuser un menu"</p> <p>À l'encart "Imprimer un menu"</p> <p>À la section "Pour aller plus loin"</p> <p>Aux 3 derniers articles de blog qui parlent de MenuMaker</p> <p>Titre</p> <p>Photo de couverture</p> <p>Lien</p> | <p>Développement d'une interface de vue globale sur les menus créés ainsi que sur les trois derniers articles du blog de Qwenta.</p> <p>Ces données seront récupérées grâce à un appel au back-end.</p> | <p>Le tableau de bord se composera des fonctions de création d'un menu, de diffusion d'un menu, d'imprimer un menu, d'une section pour aller plus loin et les trois derniers articles de blog qui parlent de MenuMaker.</p> | <p>1) Le tableau de bord répertorie les fonctionnalités principales pour améliorer l'expérience utilisateur.</p> <p>2) Le fait d'avoir les 3 derniers articles de blog permet à l'utilisateur d'avoir un retour sur le site et d'effectuer une veille informationnelle sur le site.</p> |

| | | | | |
|------------------------------|--|--|---|---|
| Branding restaurateur | <p>Le restaurateur doit pouvoir ajouter / modifier / supprimer les éléments suivants :</p> <p>Logo</p> <p>Couleurs de base</p> | <p>Interface visuelle et fonctionnelle au niveau de l'identité du restaurant (logo et couleurs), proposer une prévisualisation du style. Utilisation de react-color pour le choix de couleurs et react-dropzone pour le logo. Une fois les données modifiées, les gérer et les stockées.</p> | <p>Depuis cette interface, le restaurateur aura le pouvoir de jouer avec le logo et les couleurs rattachés à son restaurant, une fonction dynamique et en temps réel.</p> | <p>1) L'identité visuelle de tous les restaurants pourra être réussie à travers cette possibilité de modification.</p> <p>2) Le site pourra alors toucher tous les restaurateurs, qui pourront avoir un menu qui se différencie des autres avec leur logo et les couleurs souhaitées (en plus des plats, de la typographie...).</p> |
|------------------------------|--|--|---|---|

II. Liens avec le back-end

Au niveau du choix côté serveur et de l'API, nous avons opté pour l'utilisation de Node.js avec Express, ils se complémentent bien avec un framework récent comme React. De plus, ils sont très complets : ils proposent une forte sécurité au niveau de l'authentification utilisateur et permettent d'avoir un affichage dynamique et fluide à travers les fonctionnalités du site. Nous aurons aussi besoin de l'API Instagram et Deliveroo.

La base de données choisie est MongoDB, une base de données NoSQL. Cette dernière propose une flexibilité forte en termes de schéma de données, se marie bien avec les frameworks modernes types React et permet de manipuler des données complexes en JSON.

III. Préconisations concernant le domaine et l'hébergement

Le nom de domaine sera très probablement un sous-domaine de Qwenta, cela est en cours de validation :

www.qwenta.com/menu-maker.com

Choix de l'hébergeur : J'ai opté pour Heroku, un hébergeur qui offre un déploiement simple et adapté à des projets de petite ou moyenne envergure. Étant donné que nous ne disposons pas d'experts spécialisés en DevOps, cette solution nous permettra de gérer facilement le déploiement sans problème technique.

Adresses e-mail : contact@easy-menu-maker.com, help@easy-menu-maker.com, contact@qwenta-menu-maker.com

IV. Accessibilité

Compatibilité navigateur : En reprenant le fichier des spécifications fonctionnelles, il est stipulé que le site doit être compatible sur les dernières versions de Chrome, Safari et Firefox.

Type d'appareil : le site devra être en version desktop, il n'y a 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. Il est également important de penser aux personnes en situation de handicap : mettre en place de bons contrastes de couleurs, utiliser des indicateurs de focus visibles, mettre des attributs « alt » aux images...

SEO : la stratégie d'acquisition ne prévoit pas d'exigences particulières en termes de SEO.

V. Recommandations en termes de sécurité

- Consulter les normes OWASP pour rester informé des dernières menaces.
- Utiliser la bibliothèque DOMPurify pour React, cela va nettoyer les données dynamiques affichées (protection contre les attaques XSS).
- Protéger les formulaires avec des tokens CSRF.
- Configurer les règles CORS pour restreindre les domaines autorisés à interagir avec l'API.
- Utiliser des tokens JWT pour l'authentification afin de configurer les autorisations.
- Hashage des mots de passe avec bcrypt.
- Chiffrement AES des données sensibles (informations bancaires ou autre).

- Effectuer des tests de sécurité réguliers.
- Utiliser des en-têtes http.
- Surveiller et limiter la taille des fichiers téléchargés.
- Limiter les tentatives de connexion sur une même adresse IP.
- Surveiller les données d'entrée sur le serveur.
- Utilisation de la bibliothèque express-rate-limit pour limiter le nombre de requêtes sur une API.

VI. Maintenance du site et futures mises à jour

- Mise à jour des dépendances (npm updates).
- Surveiller les performances à travers différents outils comme par exemple New Relic.
- Sauvegarder de manière régulière les données issues de la base de données.
- Nettoyer les données de manière régulière afin de ne pas surcharger le serveur avec le temps.
- S'assurer que tout soit à jour, si ce n'est pas le cas, remédier à cela en faisant des updates sur tout ce qui touche au site.
- S'assurer que le site reste compatible sur tous les navigateurs, suivre les mises à jour de ces derniers et ajuster s'il y a des modifications à effectuer. Si cela n'est pas fait, le site ne sera plus accessible pour un certain nombre de personnes ce qui pénalisera la visibilité.
- Communiquer un canal sur lequel les utilisateurs peuvent signaler des problèmes.
- Mettre en place des test automatisés.
- Inspecter le temps de chargement des données front-end pour conserver un site rapide et agréable d'utilisation.