

# Spécifications techniques

[Menu Maker - Qwenta]

Version	Auteur	Date	Approbation
1.0	Combaud François	20/08/2025	Soufiane

I. Choix technologiques	2
II. Liens avec le back-end	3
III. Préconisations concernant le domaine et l'hébergement	3
IV. Accessibilité	3
V. Recommandations en termes de sécurité	3
VI. Maintenance du site et futures mises à jour	4

## I. Choix technologiques

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

Besoin	Contraintes	Solution	Description de la solution	Justification (2 arguments)
Landing Page	Compréhension rapide, sections claires. L'intérêt est de créer un menu.	HTML & CSS	Création de la page web avec des langages simples	Langage simple qui permet d'avoir une landing page facile et rapide à charger pour l'utilisateur.
Gestion du compte utilisateur (connexion, déconnexion, modification email)	Connexion rapide et sécurisé Déconnexion accessible sur toutes les pages de connectés Gestion de session/token Sécuriser l'accès les modifications : réauthentification requise avant les changements sensible	Authentification avec MongoDB + JWT, API sécurisée pour modifications	Création d'un système d'authentification par mail/mdp L'utilisateur se connecte via un formulaire, l'API génère un JWT pour la session Déconnexion = suppression du token côté client. Mise en place de vérifications par email pour toute modification.	Centralisation de toute la logique d'authentification et de gestion des profils. Gain de temps, meilleure sécurité et cohérence globale.
Création catégories	La création doit être	React Modale	Ajout / sélection des	Flexible, UX fluide, base de

	intuitive, rapide et sécurisée. Sans rechargement de page.		catégories via modale + liste déroulante	données adaptée aux multiples entrées.
Ajout des plats	L'utilisateur doit pouvoir simplement créer son menu	React Modale	React permet de créer des interfaces utilisateurs	React permet de créer des composants réutilisables.
Personnalisation menu	Prévisualisation en temps réel Choix de typo et couleur Stockage des préférences utilisateur	React Google Font	Interface avec menu en direct L'utilisateur choisi une police et défini la couleur du texte	Live preview améliore l'expérience utilisateur. Google font garantit des polices variées et accessibles.
Export PDF	Mise en page lisible Génération rapide front et back	jsPDF	jsPDF pour générer un PDF directement via le navigateur.	jsPDF est simple à implémenter pour un export direct
Impression menu	Accès rapide depuis la page d'accueil Lien externe sécurisé	Lien externe vers backOffice Qwenta	Ajouter un bouton/encart visible en page d'accueil. Cliquer dessus ouvre un nouvel onglet pointant vers le back Office Qwenta	Facile à implémenter, améliorer l'expérience utilisateur et garantit une navigation claire et sécurisée
Gestion des menus	Vue claire et	React	Interface affichant tous	Permet une gestion

	organisé CRUD complet Affichage des dates	CRUD API	les menus sous forme de liste ou carte avec option Modifier/Supprimer/Crée r.	intuitive et complète des menus
Mention légales	Accessible depuis toutes les pages Modale statique Texte fixe 'tous droits réservés'	React CSS	Créer un composant modale affichant un texte statique. La modale s'ouvre au clic sur le lien "mentions légales" et peut être fermer facilement	Permet un accès rapide aux informations légales sur toutes les pages. Modale réutilisable et simple à maintenir
Tarifications	Lien accessible depuis toutes les pages Ouverture dans un nouvel onglet URL externe	Lien HTML / React Router externe	Ajouter un bouton/onglet "tarifs" qui ouvre l'url externe dans un nouvel onglet. L'URL sera paramétrable pour s'adapter au futur lien Qwenta	Simple à implémenter, navigation claire pour l'utilisateur, permet de pointer vers la page officiel Qwenta sans dupliquer les données.
Export Deliveroo	Lien visible dans la section "exportez et diffusez" Ouverture dans un nouvel onglet Redirection vers l'application Deliveroo	Lien HTML / React	Ajouter un encart "diffuser sur deliveroo" qui ouvre l'application Deliveroo dans un nouvel onglet ou redirige vers leur site	Simple et rapide pour l'utilisateur, permet centraliser la diffusion des menus sur les plateformes partenaires sans développer d'intégration complexe
Partage Insta	Encart visible dans "Exportez et diffusez" Image format carré	React pour l'UI Lien vers Insta	Encart "Partager sur Insta" qui génère des images carrés du menu	Permet un partage rapide et direct du menu sur Instagram, améliore la

	Redirection vers Insta		depuis le front-end. L'utilisateur peut ensuite ouvrir Instagram et publier les images générées	visibilité des restaurateurs, sans intégration API complexe
Dashboard	Interface claire : contenu dynamique	Front React (SPA) ; API pour récupérer articles Qwenta	Créer une page dashboard regroupant les encarts Créer / Diffuser / Imprimer menu + section "Pour aller plus loin" qui récupère via API les 3 dernier Qwenta	Simplifie la navigation, centralise les actions clés et rend l'app plus attractive grâce au contenu dynamique
Branding	Interface intuitive ; images stockées ; cohérence design	React + Mongo	Permettre d'uploader un logo, choisir/modifier/supprimer des couleurs de base via une interface simple. Les données sont stockées dans MongoDB.	Identité visuelle personnalisée, essentielle pour la communication d'une entreprise.

## II. Liens avec le back-end

- Quel langage pour le serveur ?
  - 🌕 [Node.js](#) et express
- A-t-on besoin d'une API ? Si oui laquelle ?
  - 🌕 API interne : CRUD (utilisateurs, menus, plats), authentification via JWT
  - 🌕 API externe : Qwenta (back-office impression menus) / Deliveroo (redirection simple, pas besoin d'intégration complexe) / Instagram (redirection ou génération image locale)
- Base de données choisie :
  - 🌕 MongoDB (données flexible, facile à scaler, intégration simple avec Node.js)

## III. Préconisations concernant le domaine et l'hébergement

- Nom du domaine.
  - 🌕 Qwenta-menu-maker.fr
- Nom de l'hébergement.
  - 🌕 Vercel (front) + Render (backend [Node.js](#)) + MongoDB Atlas
- Adresse e-mail.
  - 🌕 contact@menu-maker.fr

## IV. Accessibilité

- Compatibilité navigateur.
  - 🌐 Compatible pour toutes les dernières versions des navigateurs les plus utilisés.
- Types d'appareils.
  - 🌐 Dispo sur ordinateur mais version mobile dans le futur

## V. Recommandations en termes de sécurité

- Accès aux comptes, plugin :
  - 🌐 Authentification sécurisée avec JWT pour gérer les sessions utilisateurs
  - 🌐 Mot de passe hashés avec bcrypt avant stockage dans MongoDB
  - 🌐 Validation des emails lors de l'inscription
  - 🌐 Gestion des rôles (restaurateurs / admin avec autorisations distinctes)
- Plugin
  - 🌐 Audit des dépendances NPM pour vérifier les failles connues
  - 🌐 Mise à jour régulières des librairies front et back
  - 🌐 Limiter les plugins tiers, n'installer que ceux nécessaires
  - 🌐 Vérifications des licences open-sources

## VI. Maintenance du site et futures mises à jour

- Grandes lignes du contrat de maintenance.
  - Surveillance du site (performance, bon fonctionnement des services)
  - Mises à jour techniques (MAJ React, Nodejs, Express, MongoDB)
  - Sauvegardes (sauvegarde automatique quotidienne de la BDD)
  - Support utilisateurs (assistance en cas de bug ou incident signalé par les utilisateurs)
  - Correctifs de bugs (correction des anomalies dans délai rapide)
  - Evolution futures (ajout de nouvelles fonctionnalités (paiement, animations, réseaux sociaux))
  - Surveillances sécurité (scan régulier des vulnérabilités et test de sécurités)
  - Rapport de suivi (rapport mensuel sur les interventions, incidents et recommandations)