



# Spécifications techniques

|        |                       |
|--------|-----------------------|
| Projet | Menu Marker by Qwenta |
|--------|-----------------------|

| Version | Auteur           | Date               | Approbation  |
|---------|------------------|--------------------|--------------|
| 1.0     | Thierry DE SOUSA | Le 14 février 2024 | John, Qwenta |

Le but de ce document est de définir et justifier les spécifications techniques de « Menu Marker by 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)   |
|---------------------------|--|---|---|---|
| <i>Création d'un menu</i> | <i>Le restaurateur doit pouvoir ajouter un plat à son menu de manière intuitive et conviviale.</i> | <i>Utilisation d'un formulaire de saisie pour ajouter une entrée, un plat ou dessert.</i> | <i>Créer un formulaire de saisie dans l'interface utilisateur où le restaurateur peut renseigner le nom des plats qui</i> | <i>Nous avons choisi de développer en React, une librairie JavaScript populaire et cohérente pour le développement. React offre une</i> |



Webgencia

|                                 |  |  |  |  |
|---------------------------------|--|--|--|--|
|                                 |  |  | <i>le composent par catégorie (entrée, plat ou dessert), son prix, et sa description. Lorsque le formulaire est soumis, les informations du plat sont traitées et ajoutées au menu.</i>  | <i>structure claire pour gérer les composants et les états, ce qui facilite la création et la gestion du formulaire de saisie.</i>   |
| <i>Personnalisation du menu</i> | <i>Les restaurateurs doivent pouvoir personnaliser leur menu de manière intuitive et dynamique avec le logo du restaurant, choix de police et de couleur</i> | <i>Utilisation d'une interface utilisateur conviviale pour enregistrer le logo du restaurant et choisir la police et sa couleur, avec des mécanismes CSS dynamiques.</i> | <i>Créer une section dans l'interface utilisateur où les restaurateurs peuvent enregistrer leur logo, sélectionner la police et choisir sa couleur. Utiliser des outils CSS dynamiques pour appliquer ces choix au menu en temps réel.</i> | <i>Nous avons choisi de développer en React, une librairie JavaScript populaire et cohérente pour le développement. React offre une structure claire pour gérer les composants et les états, ce qui facilite la mise en place d'une interface utilisateur réactive et intuitive. Pour la gestion des choix de police et de couleur, nous optons pour des mécanismes CSS dynamiques. Cette approche permet une personnalisation en temps réel de l'apparence du menu sans avoir besoin de recharger la page, offrant ainsi une expérience utilisateur fluide et agréable. De plus, cela réduit la</i> |



|  |  |   |  |   |
|--|--|---|--|---|
|  |  |   |  | complexité du code JavaScript nécessaire pour gérer ces fonctionnalités.  |
| <i>Diffusion du menu sur différentes plateformes</i> | <i>Les restaurateurs doivent pouvoir diffuser le menu sur Deliveroo et le partager sur Instagram de manière simple et intuitive.</i> | <i>Utilisation de différentes technologies et services pour chaque plateforme de diffusion.</i> | <p><b>Diffusion sur Deliveroo :</b> Intégrer l'API de Deliveroo dans votre application web pour permettre aux restaurateurs de publier leur menu sur la plateforme Deliveroo. Utilisez les outils fournis par Deliveroo pour cette intégration.</p> <p><b>Partage sur Instagram :</b> Créez une fonctionnalité qui permet aux restaurateurs de partager leur menu sur Instagram en utilisant l'API d'Instagram ou un service tiers autorisé par Instagram.</p> | <i>L'intégration de l'API de Deliveroo nous permet d'offrir aux restaurateurs la possibilité de publier leur menu directement sur cette plateforme populaire de livraison de repas. Pour le partage sur Instagram, nous choisissons une approche qui respecte les directives d'Instagram et utilise des services autorisés. Cela garantit que le partage du menu sur Instagram se fait de manière légale et conforme aux politiques de la plateforme.</i> |
| <i>Exportation du menu en PDF</i>                    | <i>Les restaurateurs doivent pouvoir exporter leur menu en PDF</i>   | <i>Utilisation d'une technologie adaptée.</i>   | <i>Utiliser une bibliothèque JavaScript comme jsPDF pour générer un fichier PDF à partir du contenu HTML du menu.</i>  | <i>En utilisant jsPDF pour l'exportation en PDF, nous offrons une solution simple et efficace pour convertir le contenu HTML du menu en un format PDF standard.</i>   |



|   |   |  |  |  |
|---|---|--|--|--|
| <i>Impression du menu</i>               | <i>Les utilisateurs doivent pouvoir imprimer leur menu directement depuis l'application.</i>  | Utilisation d'un bouton d'impression et de gestion de l'impression en JavaScript.                            | <i>Intégrez un bouton "Imprimer" dans l'interface utilisateur du menu pour permettre aux utilisateurs de déclencher l'impression. Utilisez JavaScript pour détecter l'événement de clic sur le bouton "Imprimer" et déclencher la fonction d'impression du navigateur en utilisant window.print().</i> | <i>L'intégration d'un bouton d'impression et la gestion de l'impression en JavaScript offrent une interface conviviale et intuitive pour les utilisateurs, leur permettant d'imprimer leur menu en quelques clics, ce qui améliore l'utilisation de l'application.</i> |
| <i>Connexion au compte restaurateur</i> | <i>Les restaurateurs doivent pouvoir se connecter à leur compte de manière sécurisée pour accéder aux fonctionnalités de création et de personnalisation du menu.</i> | Utilisation d'un système d'authentification sécurisé avec une interface utilisateur conviviale de connexion. | <i>Mettez en place un système d'authentification robuste, comme JWT (JSON Web Tokens), pour sécuriser les connexions des restaurateurs à leur compte.</i>  | <i>L'utilisation d'un système d'authentification sécurisé comme JWT garantit que les informations d'identification des restaurateurs sont protégées lors de la connexion à leur compte.</i>  |

## II. Liens avec le back-end

---

- Quel langage pour le serveur ? *Express / React.js / NodeJS*



- A-t-on besoin d'une API ? Oui : API Swagger pour le login, API Deliveroo pour la diffusion du menu sur Deliveroo, l'API d'Instagram pour la diffusion du menu sur Instagram.
- Base de données choisie : MongoDB.

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

---

- Nom du domaine : Un sous-domaine de Qwenta en cours de validation.
- Nom de l'hébergement : Bluehost.
- Adresses e-mail : contact@webgencia.com

### IV. Accessibilité

---

- Compatibilité navigateur : compatibilité avec les dernières versions de Chrome, Safari et Firefox
- Types d'appareils : Navigable depuis le clavier et lisible par un lecteur d'écran. Uniquement en version desktop.

### V. Services tiers

---

- **Sentry** : C'est un service de surveillance des performances qui vous aide à détecter les problèmes de performances et les bugs en temps réel. Ils collectent et signalent automatiquement les erreurs JavaScript et les exceptions côté serveur.



- **Git et GitHub** : Utilisation de Git comme système de contrôle de version pour suivre les modifications apportées à votre code source. GitHub, quant à lui, est une plateforme d'hébergement de code qui facilite la collaboration et le partage de projet Git.
- **Bootstrap** : Utilisation de Bootstrap via un gestionnaire de packages avec npm. Cette bibliothèque CSS est une des plus populaires pour le développement web. Il fournit une multitude de composants prêt à l'emploi, tels que des boutons, des formulaires, des barres de navigations, etc. Il utilise également un système de grille pour faciliter la création de mises en page réactives.
- **Material -Ui** : Utilisation de cette bibliothèque d'interfaces utilisateur pour React qui implémente les principes du design Material de Google.
- **Visual Studio code** : VS code est l'IDE utilisé pour le projet. Il constitue notre environnement de développement intégré principal pour travailler sur le code source du projet. On l'utilise pour écrire, éditer et déboguer du code, ainsi que pour bénéficier de ses fonctionnalités avancées telles que la collaboration syntaxique, l'achèvement automatique, les raccourcis claviers personnalisables, etc. Il permet une flexibilité, une polyvalence, une extensibilité grâce à une vaste bibliothèque d'extension, une compatibilité avec de nombreux langages de programmation, ainsi qu'une facilité d'utilisation et une interface utilisateur intuitive.

## VI. Recommandations en termes de sécurité

---

- Description de l'API Swagger : L'utilisation de l'Api Swagger a pour rôle de gérer les connexions des comptes utilisateurs, comme la gestion des authentifications et des autorisations.
- Endpoint pour les connexions des comptes utilisateurs via leur adresse mail : `/users/login`
- Paramètres requis dans les requêtes en JavaScript, indiquer le chemin, le verbe « POST », les paramètres du header et la charge utile au format JSON (qui contient les adresses mail des utilisateurs).
- Les réponses attendues : 200 pour « Connected », 401 pour « Not Authorized » et 404 pour « User not found »
- Sécurité : Les mesures de sécurité mises en place par l'API Swagger pour protéger les connexions aux comptes incluent l'utilisation de protocoles d'authentification tels que JWT (JSON Web Tokens).



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

---

- Contrat de maintenance :
  - La prise en charge de la maintenance du site se fera par l'équipe de développement interne.
  - La fréquence de maintenance aura lieu mensuellement.
  - L'équipe couvrira les problèmes liés aux bugs, aux failles de sécurité, aux mises à jour logicielles.
  - Les canaux de communications à utiliser en cas de problèmes ou de demandes de support seront un système de tickets et une adresse e-mail dédiée.
  - La durée du contrat de maintenance est établie pour une durée de 3 ans renouvellement automatique sauf en cas de résiliation à la date anniversaire de souscription du contrat avec un préavis d'un mois.
- Mises à jour futures :
  - La fréquence des mises à jour majeures ou mineures seront déployées en fonction des besoins de l'entreprise et des utilisateurs.

Pour chaque déploiement, les nouvelles fonctionnalités ou mises à jour du site passeront des tests par l'équipe interne et Qwenta, suivi d'une validation de l'ensemble des partenaires avant la mise en production.