

# Menu Maker by Qwenta

***Présentation de la Solution Technique***

# ***Sommaire***

***I. Spécifications fonctionnelles***

***II. Veille technologique***

***III. Spécifications techniques***

***IV. Gestion de projet***

***V. Technologies utilisées***

***VI. Hébergement/Base de données***

***VII. API/Services tiers***

***VIII. Outils de développement***

***IX. Sécurité/Maintenance et surveillance***

***X. Equipe/Communication***


# Spécifications fonctionnelles

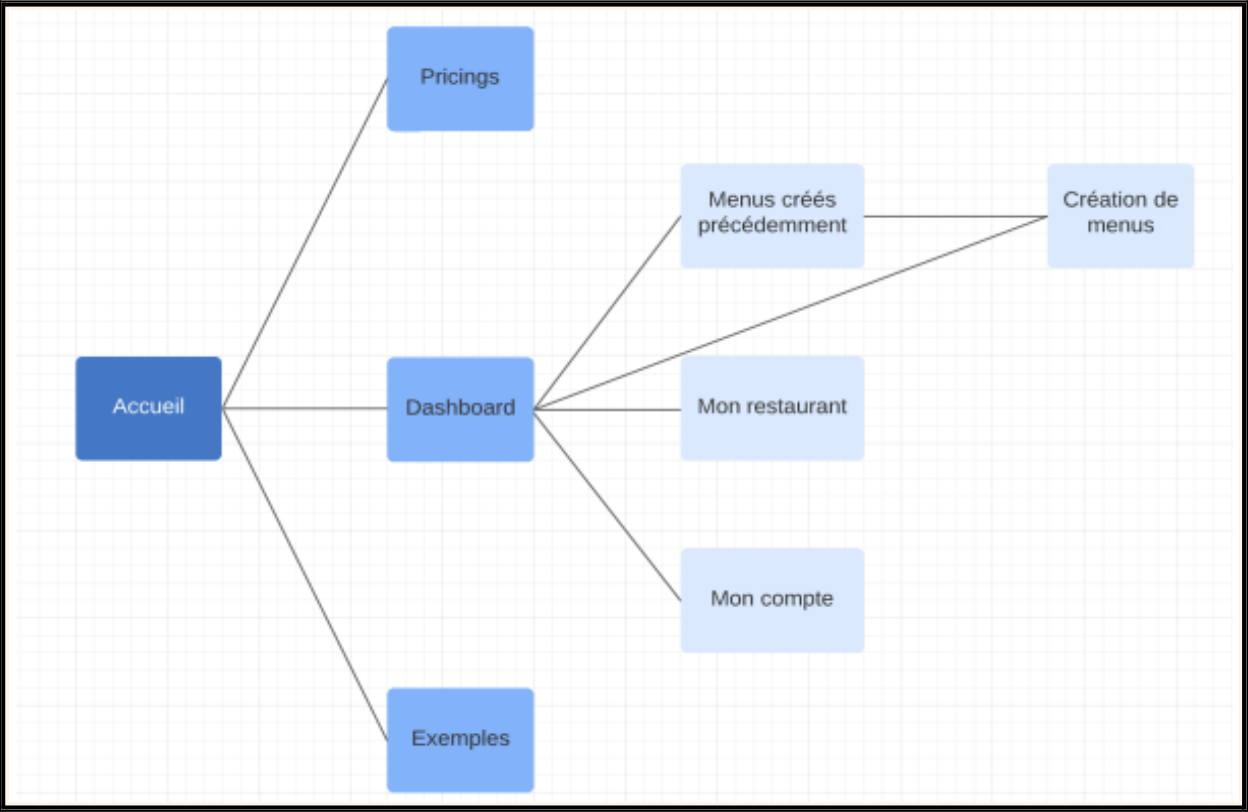
## Le restaurateur pourra

## Autres spécifications

## Arborescence du site

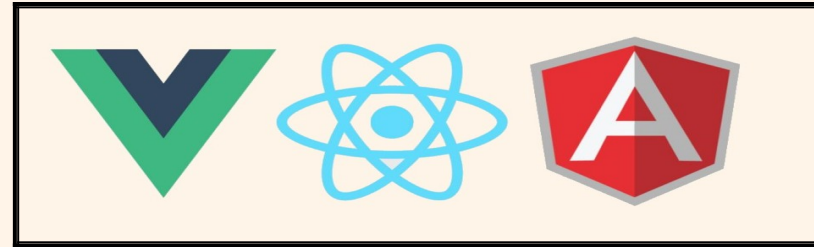
<b>Créer un menu</b>	Pour créer son menu, le restaurateur pourra ajouter : <ul style="list-style-type: none"><li>- le nom des plats qui le composent par catégorie (entrée, plat ou dessert) ;</li><li>- le prix du plat ;</li><li>- la description du plat.</li></ul>
<b>Personnaliser un menu</b>	Les restaurateurs pourront : <ul style="list-style-type: none"><li>- enregistrer leurs préférences de branding, notamment pour enregistrer le logo du restaurant ;</li><li>- choisir de façon dynamique la police et sa couleur.</li></ul>
<b>Diffuser un menu</b>	Une fois le menu prêt, il est possible de : <ul style="list-style-type: none"><li>- l'exporter en PDF ;</li><li>- le diffuser sur Deliveroo ;</li><li>- le partager sur Instagram.</li></ul>
<b>Imprimer un menu</b>	Une fois le menu prêt, il est également possible de l'imprimer.

<b>Versions</b>	Le site devra être en version desktop. Pas de version mobile à développer ni à prévoir.
<b>Couleurs du site</b>	Couleurs principales : <ul style="list-style-type: none"><li>- Beige : #FFF4E8</li><li>- Green : #8BC7B1</li><li>- Black : #000</li><li>- White : #FFF</li><li>- Brown : #C5A073</li></ul>
<b>Maquette</b>	La maquette présentant le résultat attendu pour le site est disponible <a href="#">ici</a> .
<b>Compatibilité navigateurs</b>	Pour le moment, on se contente de la compatibilité avec les dernières versions de Chrome, Safari et Firefox.
<b>Lien avec le back-end</b>	Il faudra déterminer la programmation côté serveur : choisir le langage, l'API et la base de données les mieux adaptés pour le projet.
<b>Nom de domaine</b>	Le nom de domaine sera très probablement un sous-domaine de Qwenta  en cours de validation.
<b>Accessibilité</b>	L'application devra être accessible au minimum : navigable depuis le clavier, et lisible par un lecteur d'écran.
<b>SEO</b>	La stratégie d'acquisition ne prévoit pas d'exigences particulières en termes de SEO.
<b>Analyse du comportement utilisateur</b>	Il n'y aura pas d'outil à intégrer pour capter le comportement des utilisateurs pour le moment.

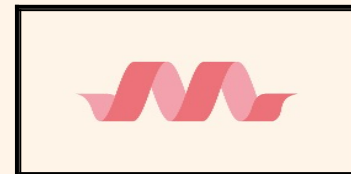
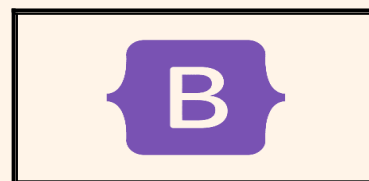
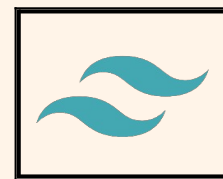


# *Veille technologique*

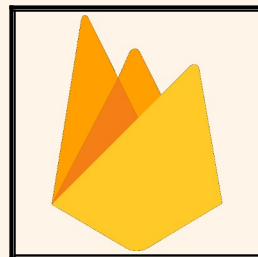
## Framework front-end



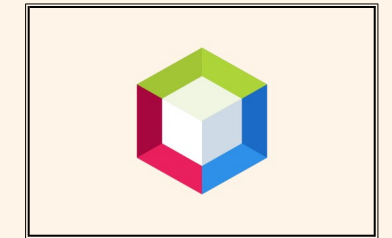
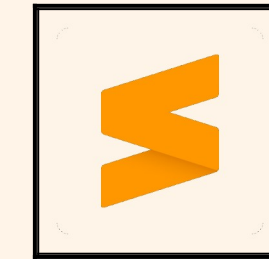
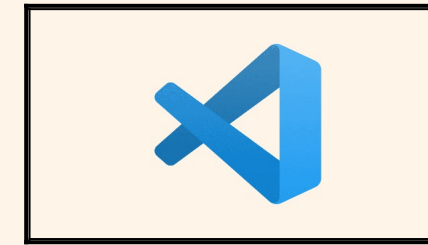
## Librairies conception web



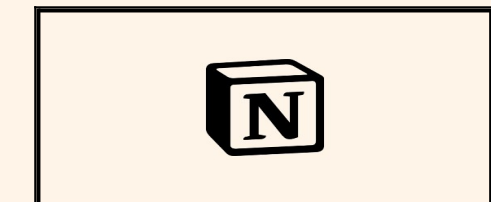
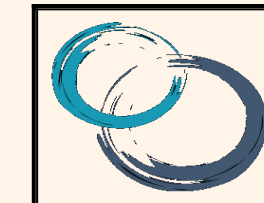
## Technologies back-end



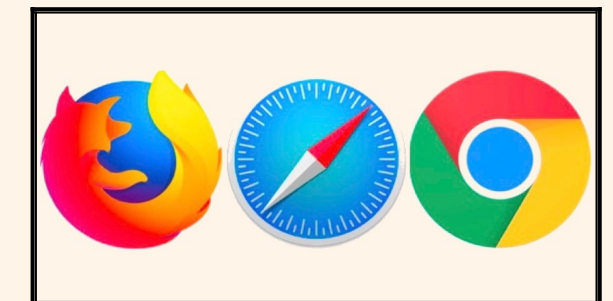
## Editeurs de code



## Gestion de projet/Agilité



## Compatibilité/Accessibilité



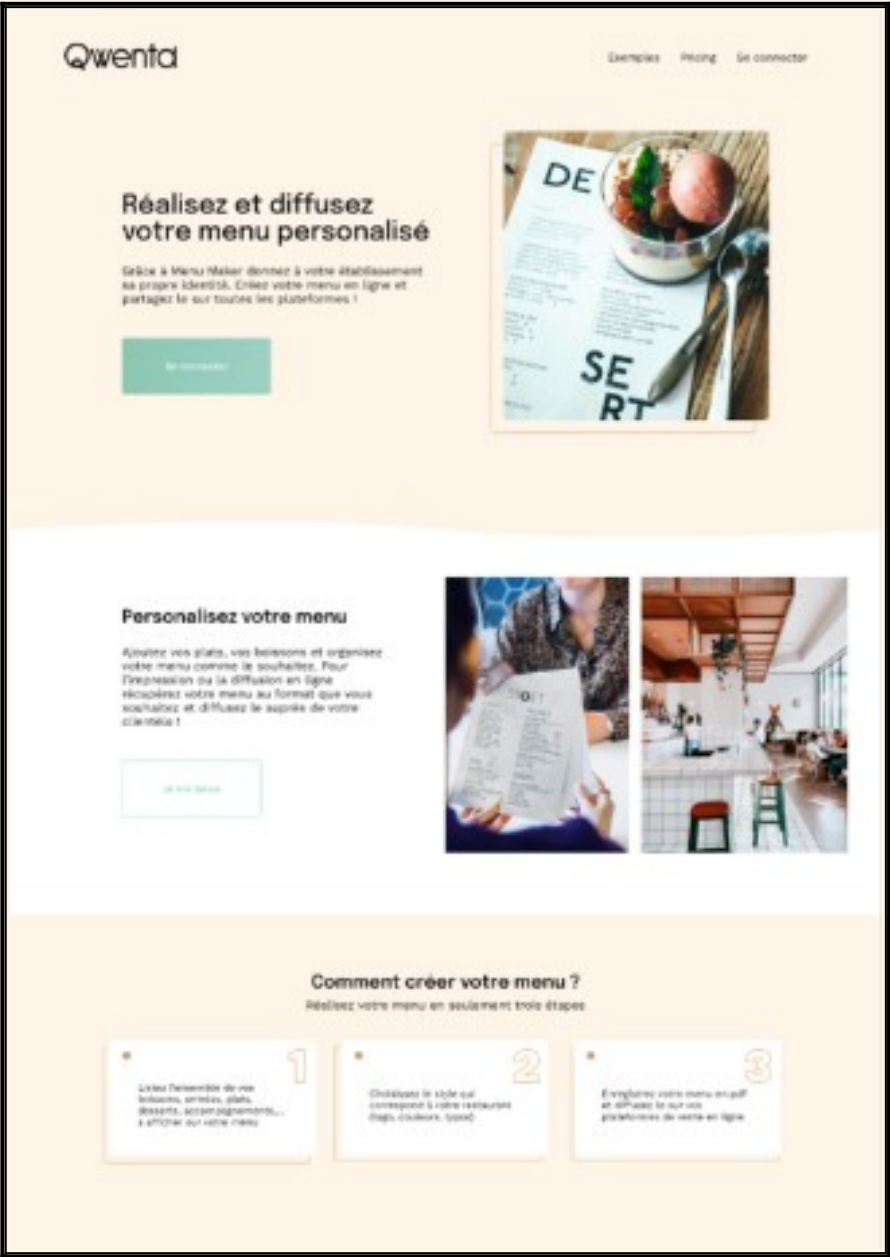
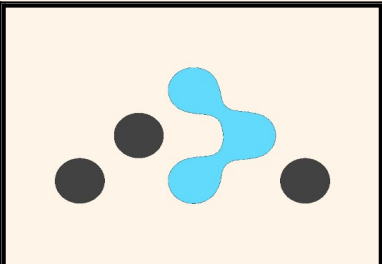
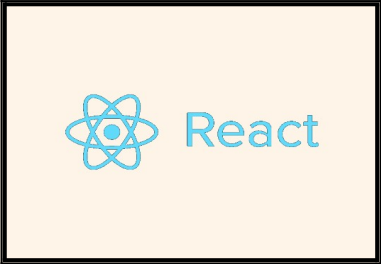
# Spécifications techniques

## Landing page non connectée:

L'internaute doit pouvoir avoir accès aux différentes sections de la landing page non connectée :

Bannière  
Personnalisez votre menu  
Explications étape par étape

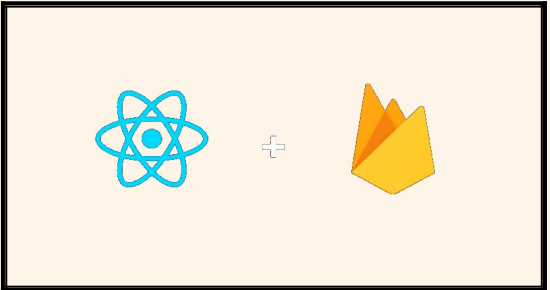
## Technologies utilisées



## Authentification:

Cette fenêtre s'ouvre sous forme de modale  
L'utilisateur peut entrer son adresse mail  
Qu'il se soit déjà connecté ou non, un mail lui est envoyé pour lui permettre de s'authentifier, ou au contraire de confirmer son mail pour accéder pour la première fois à l'application.  
Un lien "Besoin d'aide" permet d'envoyer directement un e-mail à nos équipes.

## Technologie utilisée



1.



2.



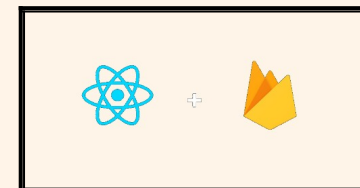
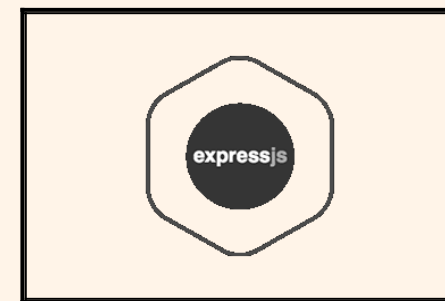
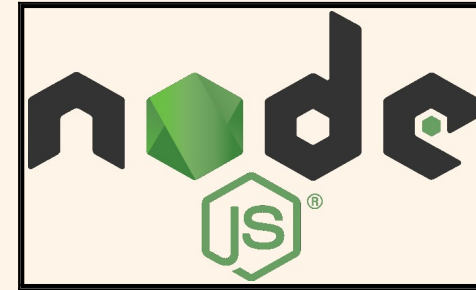
## Catégories de plats:

### L'utilisateur doit pouvoir :

### Technologies utilisées

- Créer une nouvelle catégorie:
  - La création de catégorie s'ouvre dans une modale spécifique, et doit pouvoir être validée.
- Sélectionner une catégorie créée précédemment .

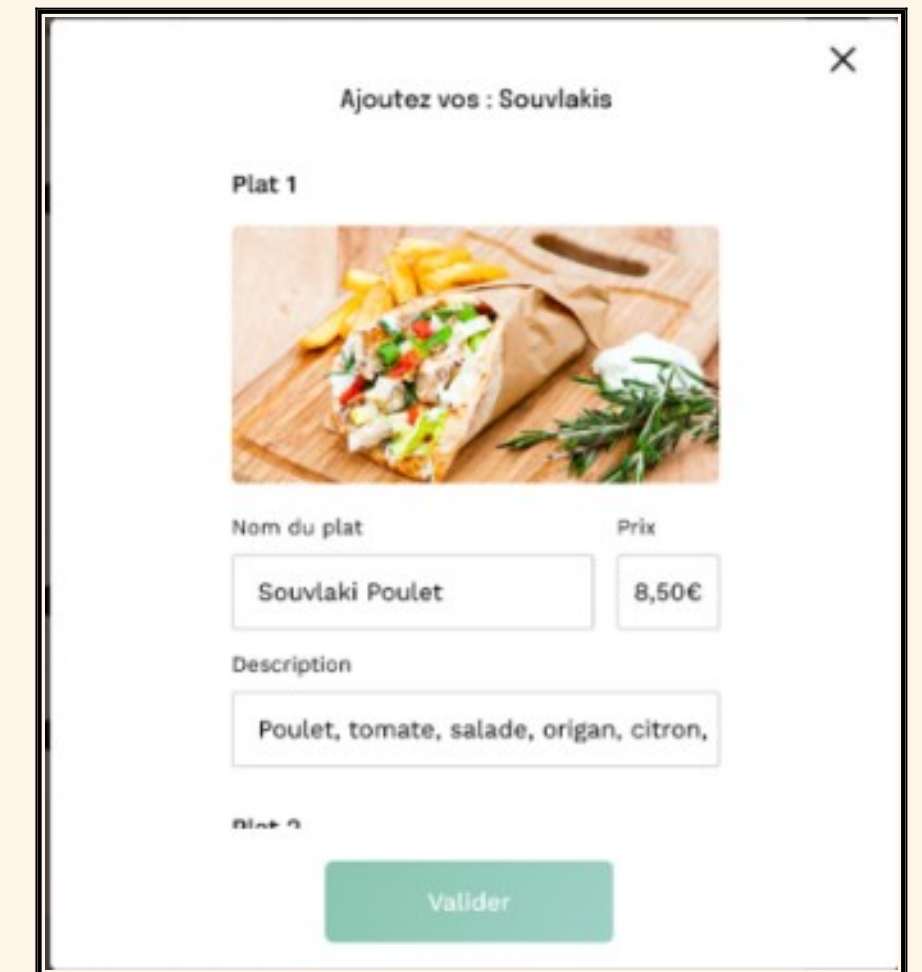
Modale d'ajout de catégorie:

A screenshot of a modal window titled "Ajouter une catégorie" with a close button (X) in the top right corner. Inside the modal, there is a label "Nom de la catégorie" above a text input field. Below the input field is a green button labeled "Valider".

## Créations de plats:

### Le restaurateur doit pouvoir :

- Sélectionner une catégorie:
  - . Modifier la catégorie sélectionnée.
  - . Entrer les informations de son plat
  - . Une modale s'ouvre pour saisir les différents plats de la catégorie.
- Chaque plat peut avoir :
  - . Une photo associée
  - . Un nom
  - . Un prix
  - . Une description
  - . Il est possible de créer autant de plats que l'on souhaite.

A screenshot of a modal window titled "Ajoutez vos : Souvlakis" with a close button (X) in the top right corner. The modal is divided into sections for "Plat 1", "Plat 2", and "Plat 3". The "Plat 1" section includes a photo of a souvlaki, a label "Nom du plat" with a text input containing "Souvlaki Poulet", a label "Prix" with a text input containing "8,50€", and a label "Description" with a text input containing "Poulet, tomate, salade, origan, citron,". A green "Valider" button is at the bottom.

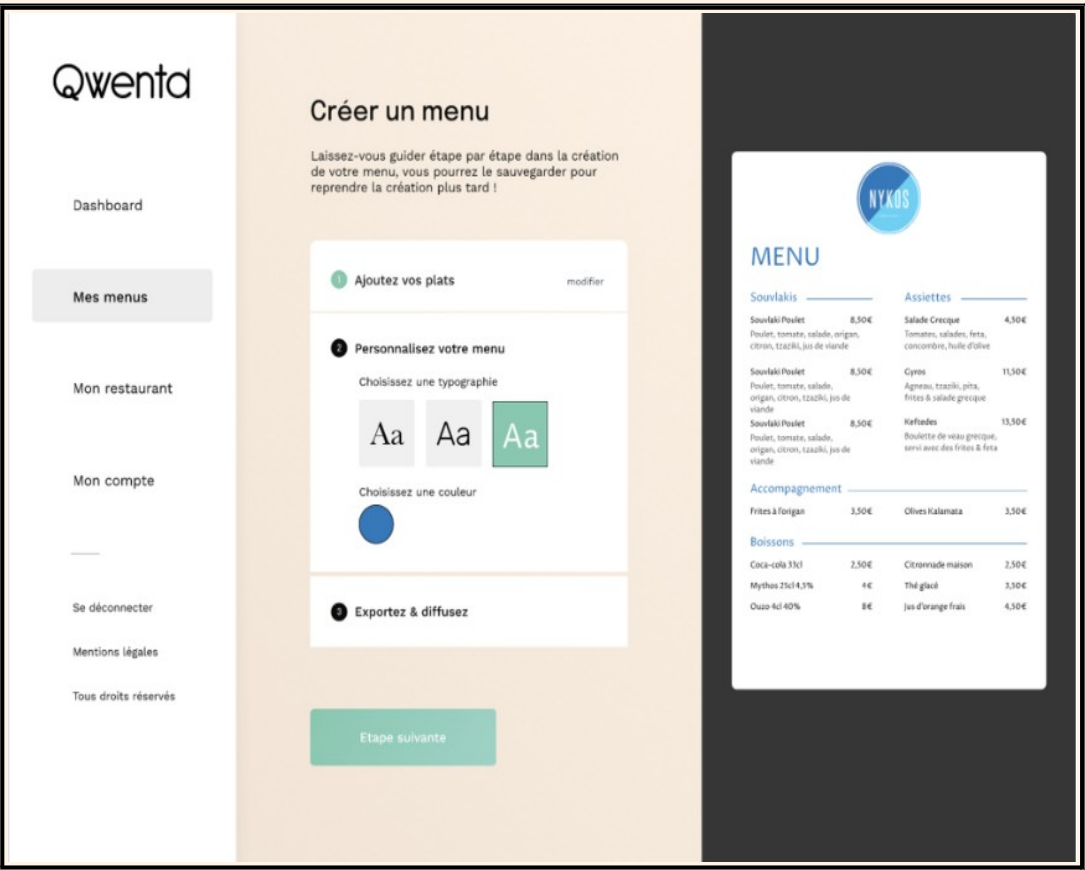
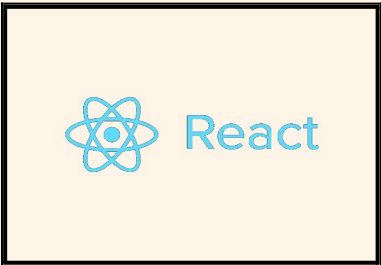


## Style de menu:

### Le restaurateur doit pouvoir :

- Visualiser le menu actuellement créé
- Sélectionné une typographie
- Choisir une couleur de texte

### Technologies utilisées

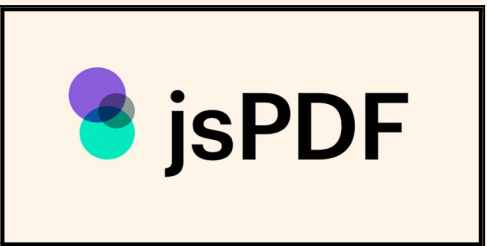


## Exportation en PDF:

### Le restaurateur doit pouvoir :

- En un clic télécharger le fichier PDF correspondant à son menu.

### Technologie utilisée



## Exportation Deliveroo:

L'encart «diffuser sur Deliveroo» doit s'afficher dans la catégorie «Exportez et diffusez».

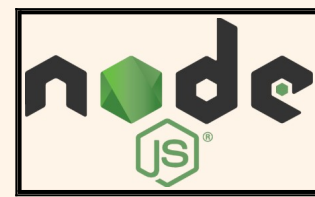
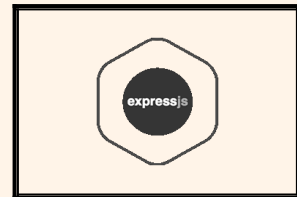
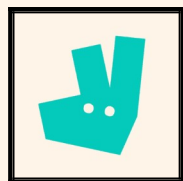
## Partage sur Instagram

L'encart "Partager sur Instagram" doit s'afficher dans la catégorie "Exportez et diffusez".

Au clic sur l'encart, des images du menu aux formats carrés, (pour les mettre sur Instagram) sont générées.

Le restaurateur est redirigé vers son compte Instagram avec les photos carrées des menus.

## Technologies utilisées



## Commander l'impression d'un menu:

L'encart "Imprimer un menu" doit être visible depuis la page d'accueil.  
Il s'agit d'un lien qui s'ouvre dans un nouvel onglet.  
Le lien doit être fait vers le back-office de Qwenta



## Technologies utilisées

Il n'est pas nécessaire de développer une fonction d'impression personnalisée, car les navigateurs modernes offrent des options d'impression avancées.

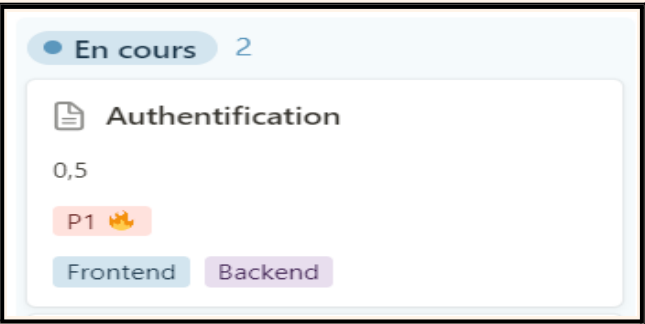


# Gestion de projet

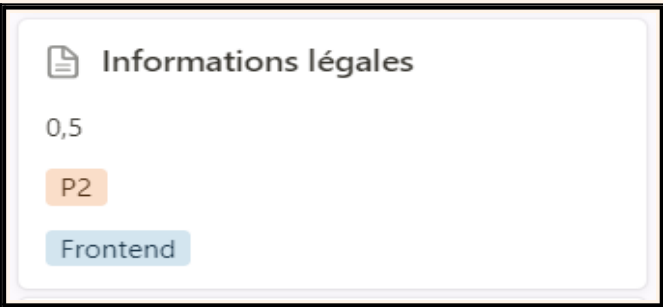
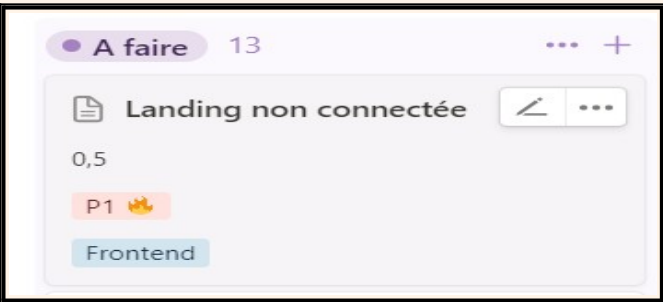
## L’outil Notion

La gestion de projet a été faite avec «Notion» sous forme de tableau Kanban afin de découper les tâches principales en sous-tâches et en plusieurs catégories (état,priorité,estimations...).

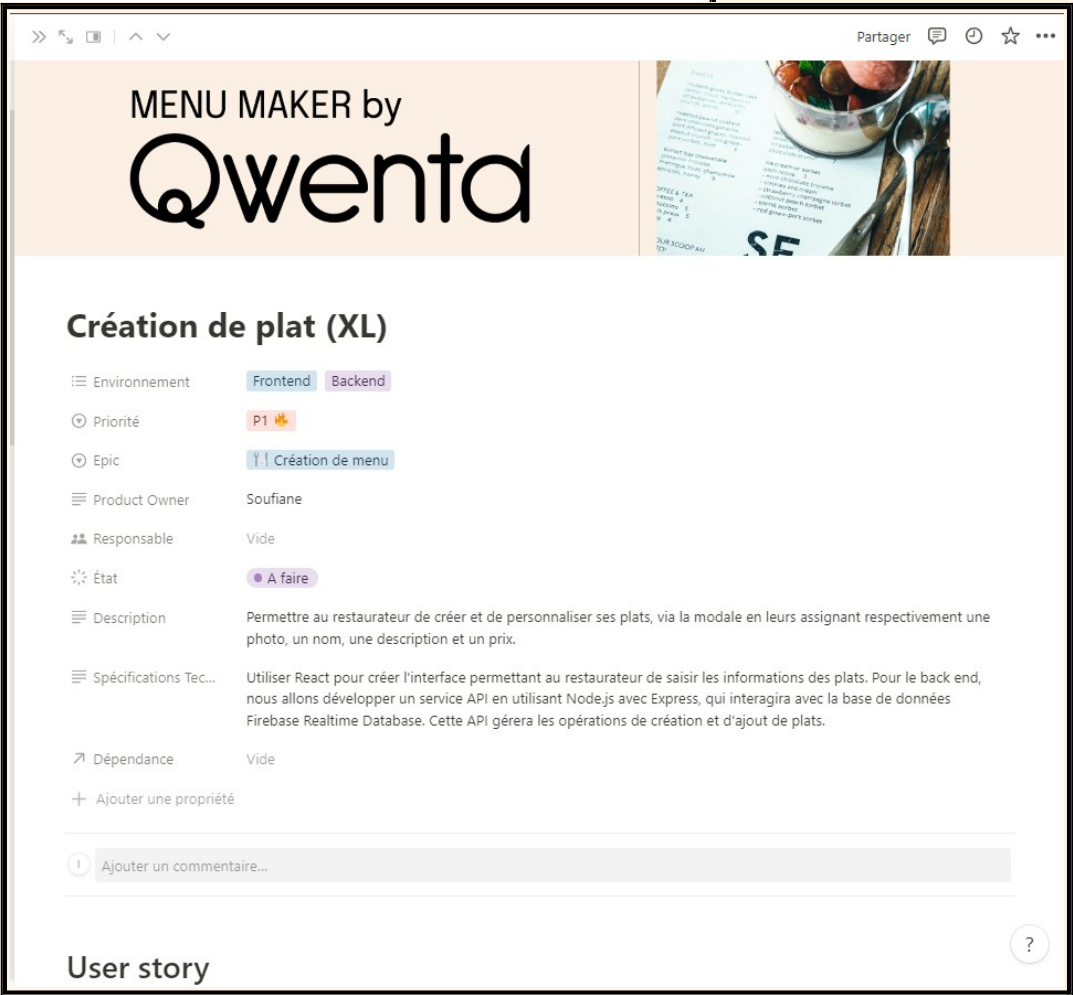
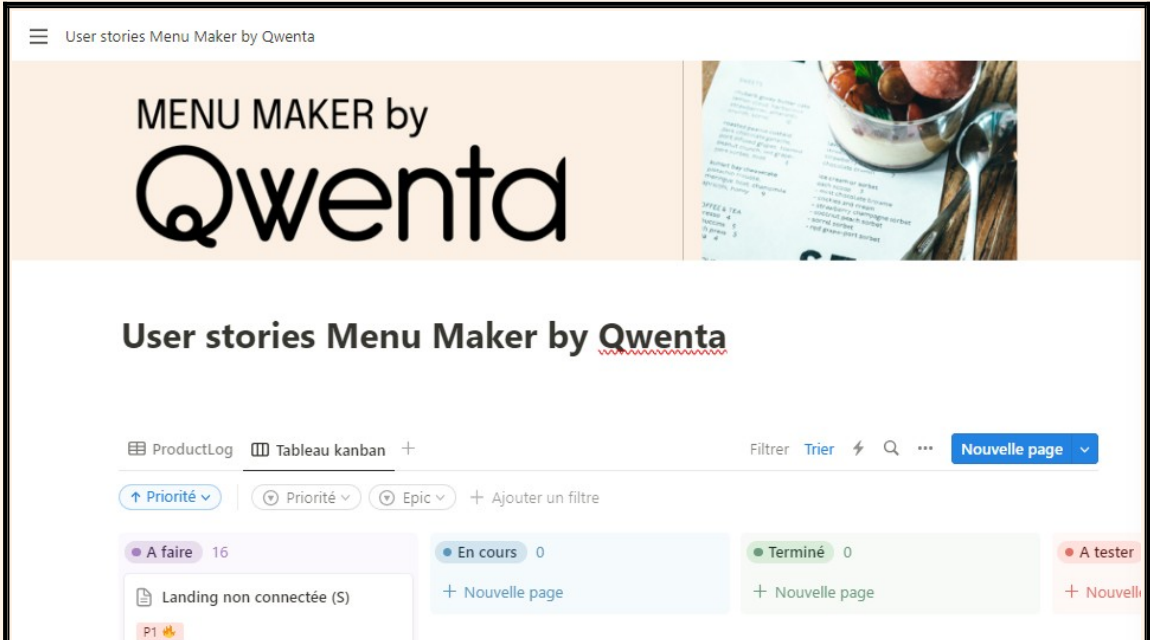
### Par état:



### Par priorité:



## Aperçu de l’accueil



# Technologies utilisées

## Technologies Frontend

**HTML/CSS :**  
Langages de base pour la structuration et la mise en forme des pages web.

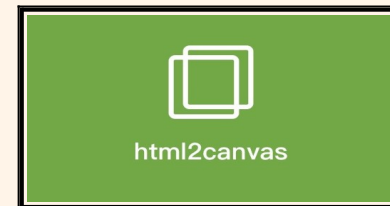
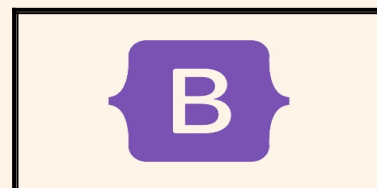
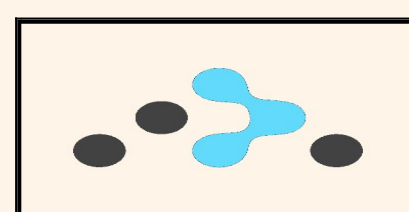
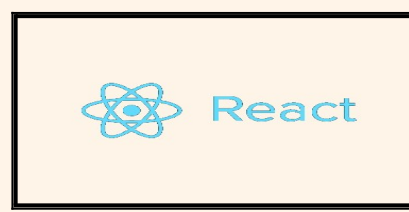
**React :**  
Bibliothèque JavaScript pour la construction de l'interface utilisateur.

**React router :**  
Pour la gestion de la navigation au sein de l'application.

**Bootstrap :**  
Frameworks CSS pour faciliter le développement de l'interface utilisateur .

**HTML2Canvas :**  
Bibliothèque JavaScript pour la capture d'écrans et la génération d'images à partir du contenu HTML .

**Redux (éventuellement) :**  
Pour la gestion de l'état global de l'application, en particulier si l'application devient complexe.



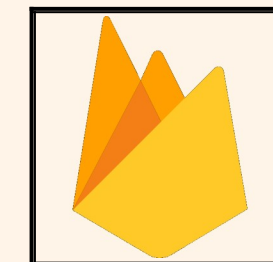
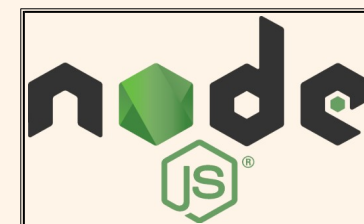
## Technologies Backend

**Node.js :**  
En tant que langage de serveur pour le back-end.

**Express.js :**  
Framework Node.js pour la création d'API RESTful et la gestion des routes.

**Firebase (Realtime Database, Authentication, Hosting, Cloud Functions) :**  
Plateforme cloud de Google pour gérer la base de données, l'authentification des utilisateurs, le déploiement d'application web, et l'exécution de fonctions côté serveur.

**Firebase Firestore :**  
Une base de données NoSQL en complément de la Firebase Realtime Database, pour le stockage de données structurées .



# *Hébergement / Base de données*

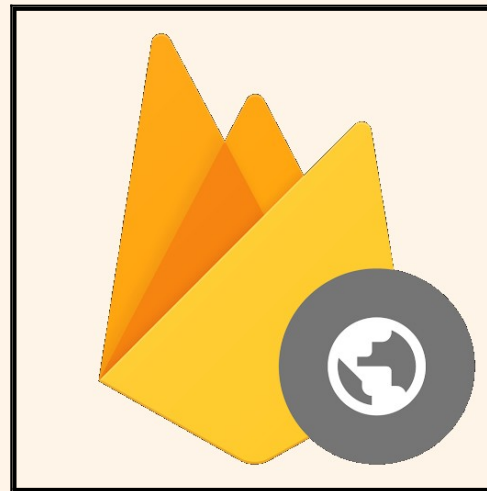
## **Hébergement**

### **Firebase hosting :**

**Pour héberger l'application web front-end et le back-end sur serveurs de Firebase.**

**Nom de domaine personnalisé ([www.menu-maker.com](http://www.menu-maker.com)) :**

**Pour rendre l'application accessible au public.**



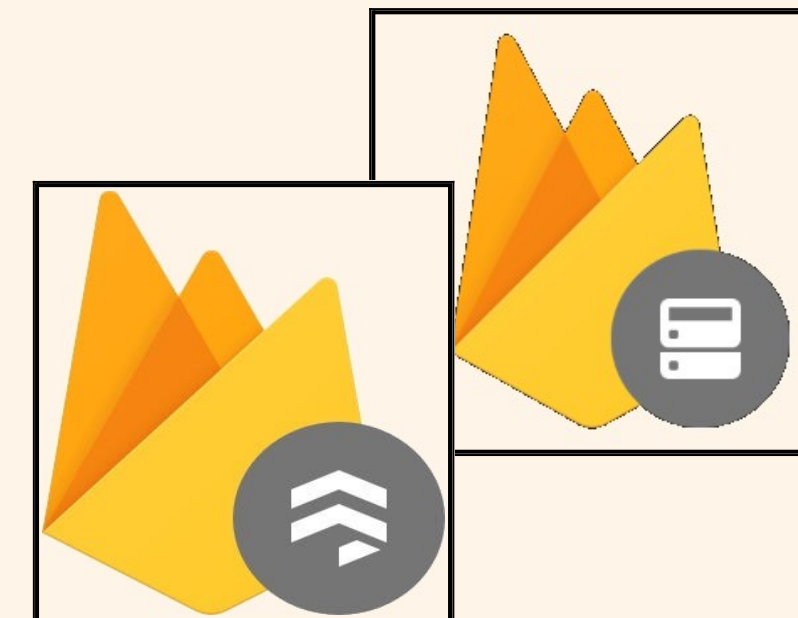
## **Base de données**

### **Firebase realtime database :**

**Pour le stockage de données en temps réel, comme les menus, les informations des restaurants, etc.**

**Firebase Firestore (éventuellement) :**

**Pour des besoins de stockage de données plus structurées ou complexes.**



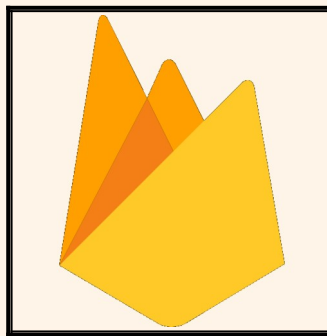
# API / Services tiers

## API

### A-t-on besoin d'une API ? Si oui laquelle ?

#### Firestore:

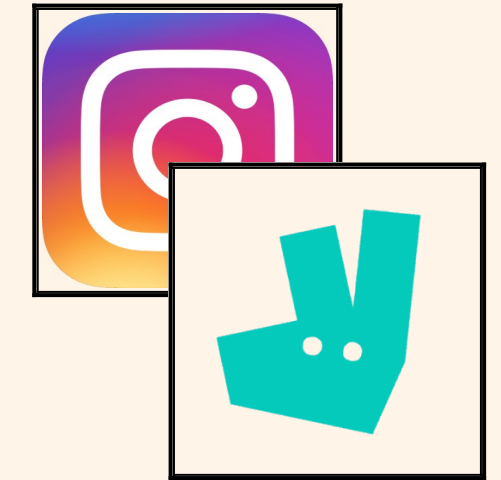
Firestore est l'API la plus appropriée pour notre projet de gestion de menu avec React, car elle offre une solution complète et intégrée pour la gestion des données, de l'authentification, du déploiement et des fonctions côté serveur, simplifiant ainsi le développement et la mise en œuvre de notre application.



## Services tiers

#### API Instagram et Deliveroo:

Les API d'instagram et Deliveroo seront utilisées pour le partage des menus.



#### Stripe / Paypal:

Pour la gestion des paiements en ligne si nécessaire pour des fonctionnalités premium.

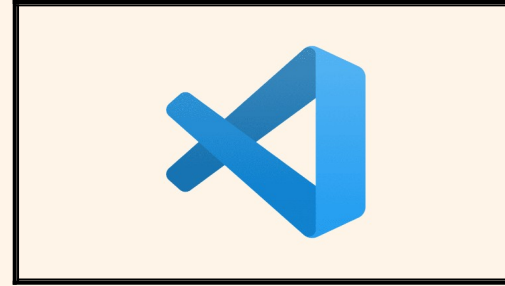


#### AddThis ou ShareThis :

Pour les boutons de partage social sur le site web.



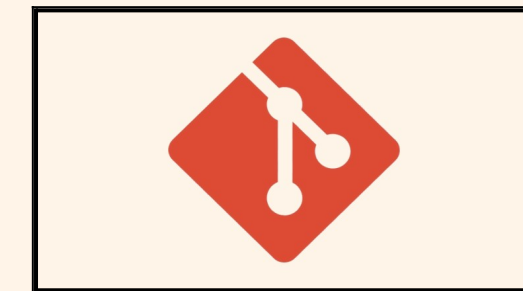
# *Outils de développement*



**Visual Studio Code :**  
**Environnement de développement intégré (IDE).**

**Git :**

**Pour la gestion de version et le suivi des modifications du code source.**



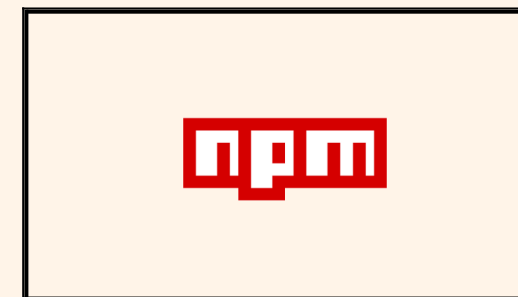
**GitHub :**

**Plateforme de gestion de code source pour le stockage du code,  
la collaboration et le suivi des problèmes.**



**Npm (Node Package Manager) :**

**Pour l'installation et la gestion des dépendances du projet.**



# *Sécurité / Maintenance et surveillance*

## **Sécurité**

### **Recommandations OWASP :**

Pour la sécurité de l'application web.

### **Firestore Authentication :**

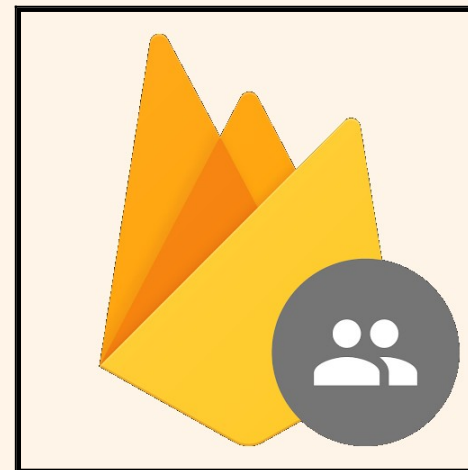
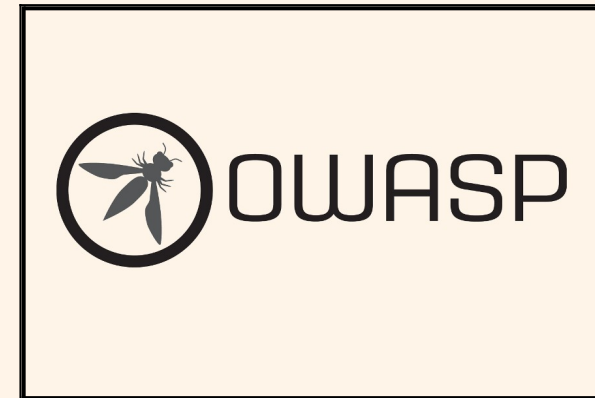
Pour l'authentification des utilisateurs.

### **Validation des données :**

Pour prévenir les attaques de type XSS, (Cross-SiteScripting) et les injections SQL.

### **Protection contre les attaques XSS et CSRF :**

Utilisation de mécanismes intégrés de protection



## **Maintenance et surveillance**

### **Outils de surveillance :**

Pour suivre les performances de l'application en temps réel.

### **Mises à jour de sécurité :**

Pour appliquer rapidement les correctifs de sécurité.

### **Mises à jour fonctionnelles et technologiques :**

Pour ajouter de nouvelles fonctionnalités et mettre à jour les technologies utilisées.

### **Sauvegardes régulières :**

Pour protéger les données contre la perte.

### **Documentation à jour :**

Pour garantir que la documentation de l'application est à jour.



# Équipe / Communication

## Équipe

**Product owner**

**Lead développeur**

**Développeur front-end**

**Développeur full-stack**

Le développeur full-stack exécutera les tâches qui nécessitent du front-end et du back-end.

Il pourra compter sur le renfort du développeur front-end s'il a finit les tâches qui lui sont attribuées (tâches front-end uniquement), afin que celui-ci puisse se concentrer uniquement sur le back-end du projet.

## Communication

### Communication en interne:

Organiser des sprints d'une à deux semaines, un sprint review aura alors lieu en présentiel à chaque fin de sprint afin de présenter le travail en cours, et de mettre à jour le product backlog.

### Format de la discussion:

Réunion une fois par semaine en visio-conférence pour présenter l'avancement du projet.

### Communication entre Webgencia et Qwenta:

Coopération régulières entre Soufiane le Product owner, et John le chef de projet.

***Merci de votre attention !!***