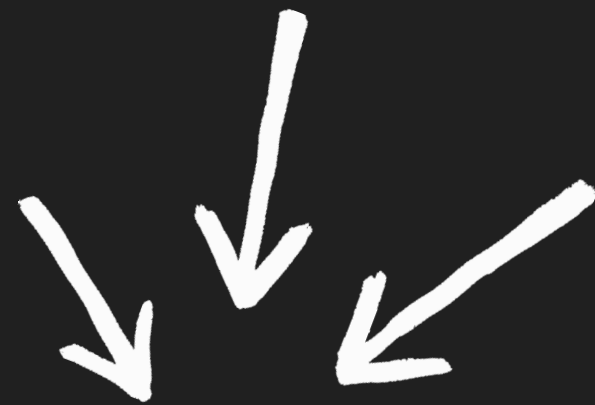


# Présentation de l'outil de gestion



Cliquez ici pour le consulter

Outil de gestion de projet

Visible par l'espace de travail

Tableau

Power-ups

Automatisation

Filtre

Backlog

Backlog

User story 9 : Informations légales

User story 10 : Tarifs

User story 11 : Exportation Deliveroo

User story 12 : Partage sur Instagram

User story 13 : Déconnexion

User story 14 : Infos utilisateur

User story 15 : Dashboard

User story : Branding restaurateur

+ Ajouter une carte

À faire

À faire

User Story 1 : Landing non connectée

User Story 2 : Page login

User Story 3 : Catégorie de plat

User Story 4 : Création de plat

User Story 5 : Style de menu

User Story 6 : Exportation PDF

User Story 7 : Commander l'impression d'un menu

User Stoty 8 : Menus précédents

+ Ajouter une carte

En cours

En cours

+ Ajouter une carte

Bloqué

Bloqué

+ Ajouter une carte

Révision du code

Révision du code

+ Ajouter une carte

Test

Test

+ Ajouter une carte

Terminé

Terminé

5 juin

1

+ Ajouter une carte

# Organisation du tableau Trello

Le Backlog	A faire	En cours
<ul style="list-style-type: none"><li>• user story avec <b>fonctionnalités intéressantes mais non prioritaire</b></li><li>• User story <b>pas encore prêtes</b> pour le développement</li></ul>	<ul style="list-style-type: none"><li>• user story de <b>priorité 1 (primordial de réaliser en premier)</b></li><li>• des tâches concrètes et précises sont définies ainsi que les spécificités techniques</li></ul>	<ul style="list-style-type: none"><li>• Cartes avec <b>tâches en cours de réalisation</b> par une ou plusieurs personnes</li></ul>
<ul style="list-style-type: none"><li>• Cette colonne offre une <b>vue d'ensemble des éléments à traiter prochainement</b> pour définir en <b>réunion de type Backlog</b> :<ul style="list-style-type: none"><li>◦ les tâches à faire</li><li>◦ niveau de difficulté (<b>poker planning</b>)</li><li>◦ planning de réalisation des cartes (<b>sprint planning</b>)</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Cette colonne offre une <b>vision claire de ce qui doit être traité immédiatement</b> pour éviter retard dans le processus de travail pour cause de tâches non essentielles</li></ul>	<ul style="list-style-type: none"><li>• Cette colonne fournit une <b>visibilité sur la réalisation des tâches et des cartes en cours</b></li><li>• Facilite la <b>communication</b> entre les membres de l'équipe</li><li>• Permet une <b>gestion plus efficace du flux</b> de travail</li></ul>
<ul style="list-style-type: none"><li>• De là, les cartes peuvent ensuite être déplacée vers la colonne à faire si celles de priorités 1 sont terminées</li></ul>	<ul style="list-style-type: none"><li>• Via la difficulté renseignée, la carte pourra être <b>assigné</b> à une personne ou à un binôme de développeurs. <b>suivant leurs compétences et expertises.</b></li></ul>	

# Organisation du tableau Trello

Bloqué	Révision du code	Test	Terminé
<ul style="list-style-type: none"><li>• Cartes dont une ou plusieurs tâches <b>empêchent les développeurs de progresser car représente un obstacle ou tout autre problème</b> (ressources manquantes outils ou connaissances...)</li></ul>	<ul style="list-style-type: none"><li>• <b>cartes avec tâches développées mais qui nécessite inspection de la qualité</b> du code avant prochaine étape (respect des normes et critères qualité définis)</li><li>• cette vérification est <b>faite par un.e développeur.se n'ayant pas participé</b> dans le code de cet user story</li></ul>	<b>cartes avec tâches développées, révisées et maintenant prêtes à être testées</b> pour vérifier si elles fonctionnent correctement	<ul style="list-style-type: none"><li>• cartes avec <b>tâches entièrement développées, révisées et testées</b></li><li>• prêtes à être livrées ou déployées</li><li>• Les membres de l'équipe peuvent alors <b>passer à d'autres cartes</b></li></ul>
<p>L'équipe entière doit <b>prendre des mesures pour résoudre le ou les problèmes</b> de la carte :</p> <ul style="list-style-type: none"><li>• <b>identification</b> de blocage</li><li>• <b>recherche</b> de solutions</li><li>• <b>mise en place</b> des solutions</li><li>• <b>réflexion sur actions correctives</b> pour que le ou les problèmes ne se reproduisent plus ou moins</li></ul>	<ul style="list-style-type: none"><li>• cette colonne permet par la vérification de <b>réduire les erreurs et les problèmes qualités</b> qui seraient relevés plus tard et <b>ralentiraient le processus</b> de développement</li><li>• les cartes de cette colonne peuvent être déplacées à la colonne "en cours" si besoin de modifications importantes avant d'être validées</li></ul>	les cartes de cette colonne peuvent être déplacé de nouveau dans "révision du code" si besoin de modifications supplémentaires	<ul style="list-style-type: none"><li>• cette colonne fournit une <b>vue claire de toutes les cartes et tâches qui ont été achevées avec succès</b></li><li>• permet de mesurer la <b>performance globale</b> de l'équipe et d'éventuellement prendre des mesures correctives pour éviter les retards ou autres soucis</li></ul>

## Section 1 :

- titre de la carte
- nom de la colonne d'appartenance de la carte

## Section 2 :

- Ordre de propriété (priorité 1, 2 ou 3)
- Type de fonctionnalité (suivant sur quelle page l'utilisateur se trouve)
- Période réalisation estimée

## Section 3 :

- description de la user story

## Section 4 :

- description des spécifications techniques suite à la veille technologique
  - outils à utiliser pour réaliser les fonctionnalités de la user story

## Section 5 :

- Estimation de ma difficulté de cette carte
- argumentation pour justifier le niveau de difficulté sur 10 (richesse du contenu, utilisation de nouveaux outils, fonctionnalités complexes...)

User Story 1 : Landing non connectée

Dans la liste À faire

Étiquettes

Propriété 1

Landing

+

Notifications

Suivie

Dates

☐ 14 mars - 27 mars à 21:00

Description

Modifier

En tant qu'internaute ne connaissant pas forcément Menu Maker, je veux pouvoir comprendre l'utilité de cette application.

Spécification technique :

- HTML5 et CSS3 pour créer la structure et le style de la landing page.
  - Bootstrap : framework CSS qui permet de créer rapidement des sites web responsives avec un style pré-défini.
  - Google Font : Google Font : bibliothèque en ligne de typos par Google
- JavaScript pour ajouter des animations et des effets visuels.
  - React-spring : Bibliothèque React pour l'animation de composants

Niveau de difficulté estimé : 2/10

- Il s'agit ici d'une création de contenu plus ou moins statique.
- Il faudra cependant faire attention au respect esthétique de la page.
- Il faudra aussi créer une route permettant l'ouverture de la modale de connexion.

### Critères d'acceptation :

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

- Bannière
- Personnalisez votre menu
- Explications étape par étape

### Futur

Améliorations possibles

- Ajouter des animations sur la photo de la bannière + sur les formes géométriques des sections.

☒ Mettre en place l'architecture de la landing page à l'aide de HTML et de CSS

Supprimer

0%

- ☐ Créer la bannière avec le menu de navigation ("Exemples" "tarifs" et "se connecter") + inclure image et logo
- ☐ Créer la section de personnalisation du menu + bouton "je me lance"
- ☐ Créer la section d'explications étape par étape
- ☐ Ajouter des animations et des effets visuels pour rendre la landing page plus attrayante

Ajouter un élément

### Section 6 :

- critères d'acceptation définis dans la user story
  - critères qui permettent d'établir les tâches à réaliser pour considérer carte validée

### Section 7 :

- objectifs futurs
  - fonctionnalités qui pourrait être ajoutées dans l'avenir et qu'il faut considérer dans le développement des fonctionnalité prioritaire

### Section 8 :

- tâches à réaliser pour le développement des fonctionnalités souhaitées et prioritaires
  - tâches concrètes et réalisable



# Description carte

## — User story I : Landing non connectée

- Niveau de difficulté = 2/10
  - Il s'agit ici d'une création de contenu plus ou moins statique.
  - Il faudra cependant faire attention au respect esthétique de la page.
  - Il faudra aussi créer une route permettant l'ouverture de la modale de connexion.
- Mettre en place l'architecture de la landing page à l'aide de HTML et de CSS :



Créer la bannière avec le menu de navigation ("Exemples" "tarifs" et "se connecter") + inclure image et logo



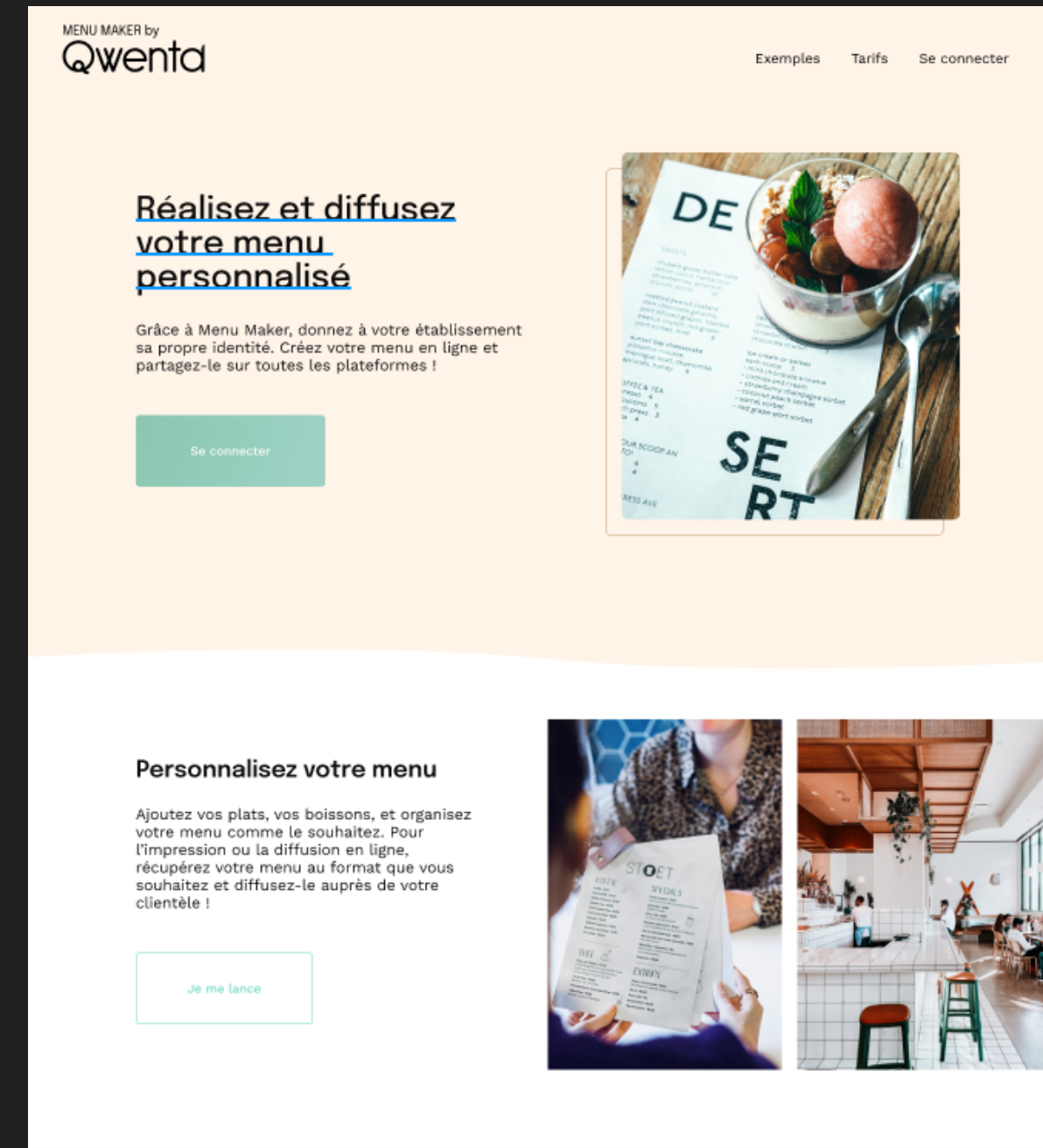
Créer la section de personnalisation du menu + bouton "je me lance"



Créer la section d'explications étape par étape



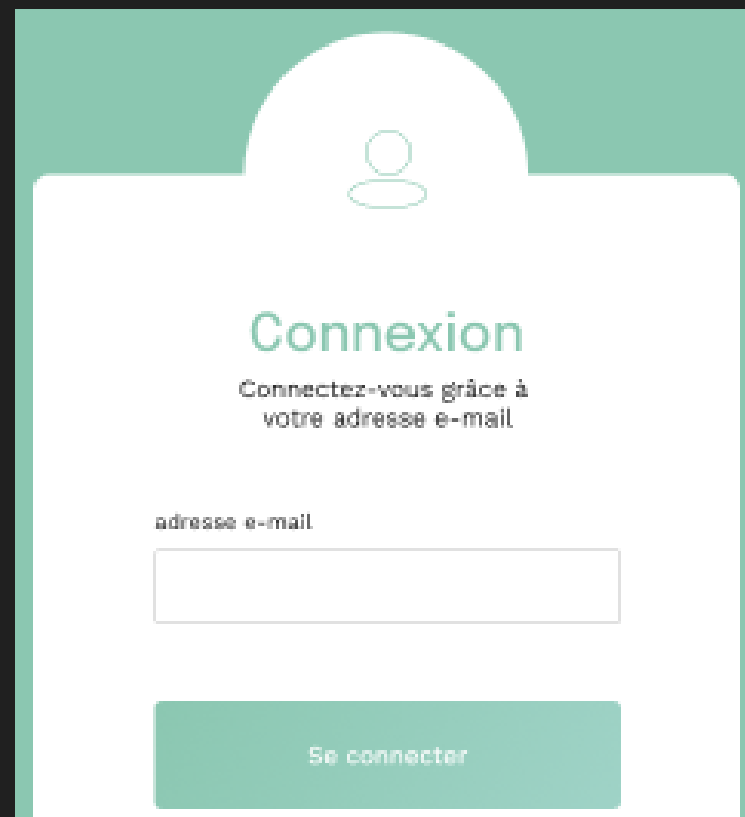
Ajouter des animations et des effets visuels pour rendre la landing page plus attrayante



# Description carte

## — User story 2 : Page login

- Niveau de difficulté = 7/10
  - Il s'agit ici d'utiliser de nombreux nouveaux outils pour gérer :
    - Firebase Auth (authentification utilisateur)
    - l'API de Cloudmersive (vérification e-mail)
  - Json Web Token et BCrypt (cryptage et stockage infos user)



- Création de la modale de connexion :



Créer une fenêtre de dialogue qui s'affiche lorsqu'un utilisateur clique sur les trois boutons spécifiques disponible sur le landing page ("se connecter" dans la barre de navigation, "se connecter" dans la bannière, "je me lance")



Créer un formulaire avec un input pour l'e-mail et un bouton de soumission grâce à la librairie "react-hook-form"



Créer un lien "Besoin d'aide" dans la fenêtre modale qui envoie un e-mail directement à notre équipe avec l'API de Cloudmersive



Gérer l'aspect de la modale avec icône de connexion, d'e-mail, de confirmation de connexion ainsi que l'animation d'attente de connexion



Envoyer un e-mail de confirmation en utilisant Firebase Auth pour l'authentification ou la confirmation de l'e-mail renseigné. Le mail contiendra un lien pour l'authentification ou la confirmation.



MISSION BACKEND : création de la base de donnée relationnelle et gestion de l'interaction avec le front



MISSION BACKEND : Stocker et sécuriser les données utilisateurs dans la base de données



Rediriger l'utilisateur lorsque confirmation de connexion



# Description carte

## — User story 3 : Catégorie de plat

- Niveau de difficulté = 5/10
  - La difficulté ici est qu'il y a un grand nombre de chose à faire, certaine plus complexes que d'autre :
  - création de l'interface, de la modale et du formulaire
  - gestion en basse de donnée de la création de menu, de catégorie et de plat
  - affichage dynamique de la catégorie

**Créer un menu**

Laissez-vous guider étape par étape dans la création de votre menu, vous pourrez le sauvegarder pour reprendre la création plus tard !

1 Ajoutez vos plats

+ Catégorie (ex. : entrées, plats...)

### • Interface de création de catégorie :

- ☐ On vérifie que l'utilisateur est bien connecté avant toute action, grâce aux infos user qui doivent être présentes dans la session
- ☐ Créer l'interface de création de menu avec différente section pour l'ajout de catégorie et de plat en premier lieu.
- ☐ Créer la section dans laquelle l'utilisateur pourra voir la prévisualisation du menu
- ☐ Ajouter un bouton "+ catégorie (...)" qui ouvre une modale au clic
- ☐ Créer un formulaire s'affichant dans la modale et demandant les informations suivantes : nom de la catégorie + bouton de soumission dans le formulaire pour permettre à l'utilisateur de valider la création de la nouvelle catégorie et l'enregistrer en basse de données
- ☐ Implémenter la basse de données avec les informations données par l'utilisateur sur la catégorie ajoutée
- ☐ Afficher la catégorie ajoutée sur l'interface de création de menu
- ☐ Afficher la catégorie ajoutée dynamiquement dans la prévisualisation de menu

# Description carte

## — User story 4 : Création de plat

- Niveau de difficulté = 5/10
  - Idem que la carte précédente, il y a un grand nombre de chose à faire, notamment la gestion du stockage des données et de l'affichage dynamique des nouveaux plats. Ici, on doit aussi gérer la récupération d'un fichier dans le formulaire.

Ajoutez vos : Souvlakis

Plat 1

  
+Ajouter photo  
max 2mo

Nom du plat

Prix

### • Interface de création de plat :

- ☐ On vérifie que l'utilisateur est bien connecté avant toute action, grâce aux infos user qui doivent être présentes dans la session
- ☐ Ajouter un bouton "modifier" à côté de chaque catégorie pour les modifier. En cliquant sur ce bouton, la modale de création s'ouvre de nouveau avec les informations de la catégorie. L'utilisateur peut modifier le titre et la description en cliquant de nouveau sur valider.
- ☐ Sous chaque catégorie existante, ajouter un bouton "+plats (...)" qui ouvre une modale avec un formulaire de création d'un nouveau plat pour la catégorie
- ☐ Créer un formulaire s'affichant dans la modale et demandant à l'utilisateur les informations suivante : photo du plat, nom, prix et description.
- ☐ Ajouter un bouton "+plat" qui rajoute un formulaire de création de plat dans la modale. La modale peut être scroller et l'utilisateur peut ajouter autant de plat que souhaité dans la modale de création de plat de la catégorie.
- ☐ Ajouter un bouton "Valider" dans le formulaire pour permettre à l'utilisateur de valider la création d'un ou plusieurs nouveaux plats et les enregistrer en base de données.
- ☐ Implémenter la base de données avec les informations données par l'utilisateur sur le ou les plats ajouté(s)
- ☐ Afficher le(s) plat(s) ajoutée(s) sous la bonne catégorie dans l'interface utilisateur ainsi que dans la prévisualisation du menu.

# Description carte

## — User story 5 : Style de menu

- Niveau de difficulté = 4/10
  - L'utilisation des bibliothèque react devrait nous simplifier la tâche, il faudra surtout gérer l'affichage dynamique ainsi que la sauvegarde du style

• Permettre la modification et sauvegarde de style de menu :

- ☐ On vérifie que l'utilisateur est bien connecté avant toute action, grâce aux infos user qui doivent être présentes dans la session
- ☐ Créer une bouton "Etape suivante" disponible une fois qu'au moins un plat a été ajouté. L'utilisateur saura qu'il est possible de cliquer dessus car la couleur sera passé du gris au vert.
- ☐ Changer l'aspect de l'interface utilisateur pour afficher la section "personnalisation du menu" : afficher trois boutons correspondant à trois types différents de typographie et un bouton permettant de sélectionner une couleur
- ☐ Lorsque l'utilisateur clique sur un style de typographie ou choisi une couleur, le menu prévisualisé doit changer dynamiquement.
- ☐ Les boutons de typographie change de couleur suivant qu'ils sont sélectionné ou non par l'utilisateur
- ☐ Une fois la typographie et la couleur choisie, le bouton "étape suivante" est de nouveau disponible et sauvegarde le style choisi en basse de donnée.

## — User story 6 : Exportation PDF

- Niveau de difficulté = 4/10
  - L'utilisation d'outils pour convertir et télécharger le PDF devrait simplifier la tâche, il s'agira de bien consulter la documentation et le code source

### • Exportation PDF :

- ☐ On vérifie que l'utilisateur est bien connecté avant toute action, grâce aux infos user qui doivent être présentes dans la session
- ☐ Changer la modale afin d'afficher la section "exporter et diffuser". Trois boutons sont disponibles : "exporter en PDF", "diffuser sur deliveroo" et "partager sur instagram".
- ☐ Au clic sur le bouton "exporter en PDF", générer le contenu du fichier PDF à partir des données du menu stockées dans la base de données
- ☐ Configurer le téléchargement du fichier PDF

# Description carte

## — User story 7 : Commander l'impression d'un menu

- Niveau de difficulté = 4/10
  - Il y a un certains nombre de chose à faire entre la création html et css de l'interface d'affichage des menus et la génération + redirection de l'utilisateur vers son pdf dans le navigateur mais ce sont des notions qui auront été vues en amont dans le projet
- Permettre l'impression d'un ou plusieurs menu depuis le back office :



Dans le Dashboard de l'utilisateur, on doit avoir un bouton "imprimer menu" qui renvoi l'utilisateur dans un nouvel onglet



On doit créer une interface back office qui permettrait à l'utilisateur de visualiser les différents menus qu'il a créée.



De cette liste qui est générée depuis le back end, on peut ajouter une option de filtre qui permettrait éventuellement de retrouver un menu avec son nom ou le nom de catégorie ou de plats



L'utilisateur peut cocher ou décocher des menus et grâce à un bouton qui envoi vers l'impression, imprimer les menus sélectionnés.

## — User story 8 : Menus précédents

- Niveau de difficulté = 4/10
  - Il s'agit principalement de la création html et css de l'interface ainsi que de requêtes HTTP pour modifier et supprimer, cela reste donc assez simple.

- Créer une interface pour visualiser les menus :



Depuis le menu du Dashboard, créer un onglet "mes menus" qui renvoi l'utilisateur vers une interface où il peut voir tous les menus déjà créés



Ajouter en dessous la date de création de ce menu que l'on récupère depuis la base de données



Ajouter un bouton "modifier" qui renvoi l'utilisateur vers l'interface de création de menu avec les informations de catégories de plats, des plats et du style sauvegardés et apparent dans l'interface, dans les champs des formulaires et dans la prévisualisation du menu.



Lorsque l'utilisateur valide le nouveau menu, une requête http est faite afin de modifier les anciennes données avec les nouvelles dans la base de données



Un bouton "supprimer" est ajouter qui, au clic, exécute une requête http delete afin de récupérer l'id du menu et de le supprimer dans la base de donnée



A côté de l'affichage des menus, un bouton "+ajouter un menu" permet de renvoyer l'utilisateur vers l'interface de création de menu, cette fois-ci vierge