



# Documentation Technique

Site « Quai Antique »

Evaluation « ECF été 2023 »

Graduate developpeur web full stack

## Table des matières

|   |    |
|---|----|
| Technologies choisies : .....                       | 3  |
| Gestion de projets : .....                          | 3  |
| Maquettage : .....                                  | 4  |
| Stack de développement : .....                      | 4  |
| Configuration de l'environnement de travail : ..... | 5  |
| Diagramme de cas d'utilisation : .....              | 5  |
| Diagramme de séquence : .....                       | 6  |
| Séquence User : .....                               | 6  |
| Séquence Administrateur : .....                     | 10 |
| Diagramme de classe : .....                         | 13 |
| Diagramme conceptuel : .....                        | 13 |
| Diagramme logique : .....                           | 14 |
| Diagramme Physique : .....                          | 15 |
| Test de l'application : .....                       | 16 |
| Page de Contact : .....                             | 16 |
| Réservation de table au restaurant : .....          | 17 |
| Les utilisateurs du site : .....                    | 22 |
| Utilisateur connecté : .....                        | 22 |
| Administrateur : .....                              | 24 |

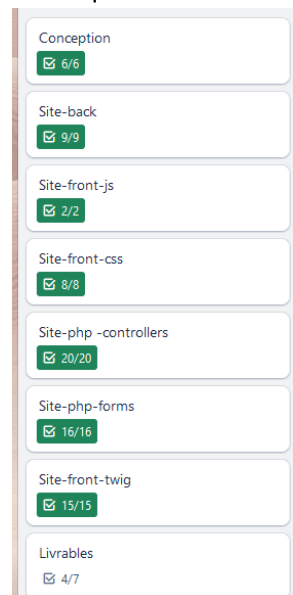
## Technologies choisies :

### Gestion de projets :

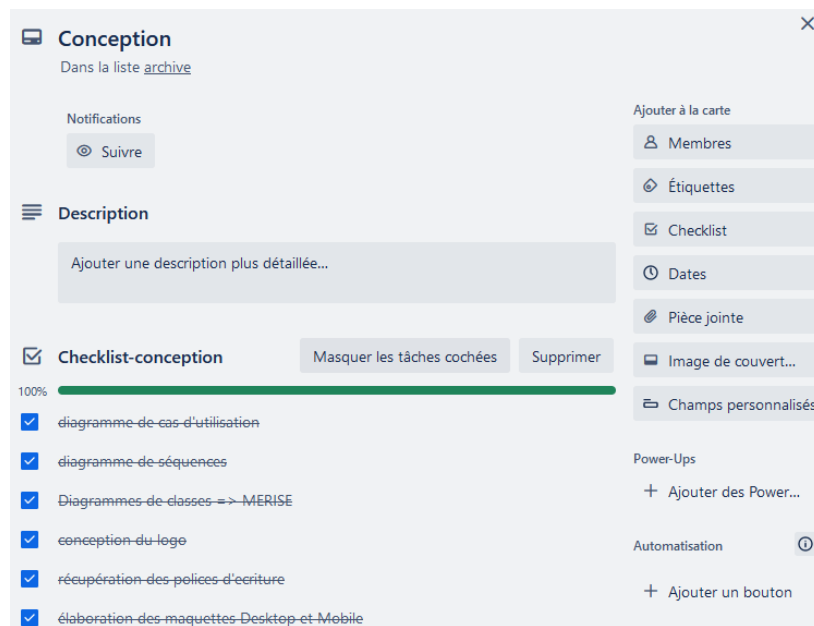
Concernant la gestion de projet, j'ai utilisé le logiciel en ligne Trello pour organiser les différentes tâches à effectuer selon la méthode des tableaux Kaban. La méthode Kanban consiste à créer 3 tableaux : « à faire », « en cours » et « fait ».



J'ai ensuite créé des cartes représentant chaque ensemble de tâches à effectuer.



Enfin à l'intérieur de chaque carte j'ai créé une « checklist » permettant de lister les sous-tâches à réaliser :



Dans la figure ci-dessus sont listées les sous-tâches pour la partie conception du travail que j'ai réalisé.

Ci-dessous le lien Trello pour le projet Quai Antique :

<https://trello.com/b/AJP4nOpS/quai-antique-ecf>

## Maquettage :

J'ai réalisé la maquette grâce à l'outil en ligne Figma.

J'ai tout d'abord réalisé le *zoning*, première étape, qui définit le découpage et l'organisation du contenu des pages grâce à des boîtes en nuances de gris.

J'ai ensuite réalisé les *wireframes* qui représentent le squelette du site.

Enfin j'ai réalisé les *mockup* qui représentent une prévisualisation haute-fidélité des pages telles qu'elles apparaîtront sur le site définitif.

La maquette du site se trouve dans le document « charte graphique site quai-antique.pdf », qui se trouve dans le dossier « /documents » du projet sur Github.

## Stack de développement :

Pour effectuer cet ECF j'ai utilisé les technologies suivantes :

Développements Front-End : HTML, CSS, JS, Bootstrap.

Développements Back-End : PHP 8.1 et le framework Symfony 6.1.

Développements bases de données : Migrations du framework Symfony, SQL et base de données MySQL.

## Configuration de l'environnement de travail :

J'ai réalisé ce projet sur un ordinateur ayant comme système d'exploitation Windows.

J'ai utilisé *PHPStorm* comme environnement de développement.

J'ai créé et développé le projet en local.

J'ai utilisé Xampp comme serveur web local, hébergement en local de la base de données.

J'ai utilisé le logiciel Beekeeper Studio afin d'écrire les requêtes d'ajout de données dans les différentes tables en SQL sur la base de données.

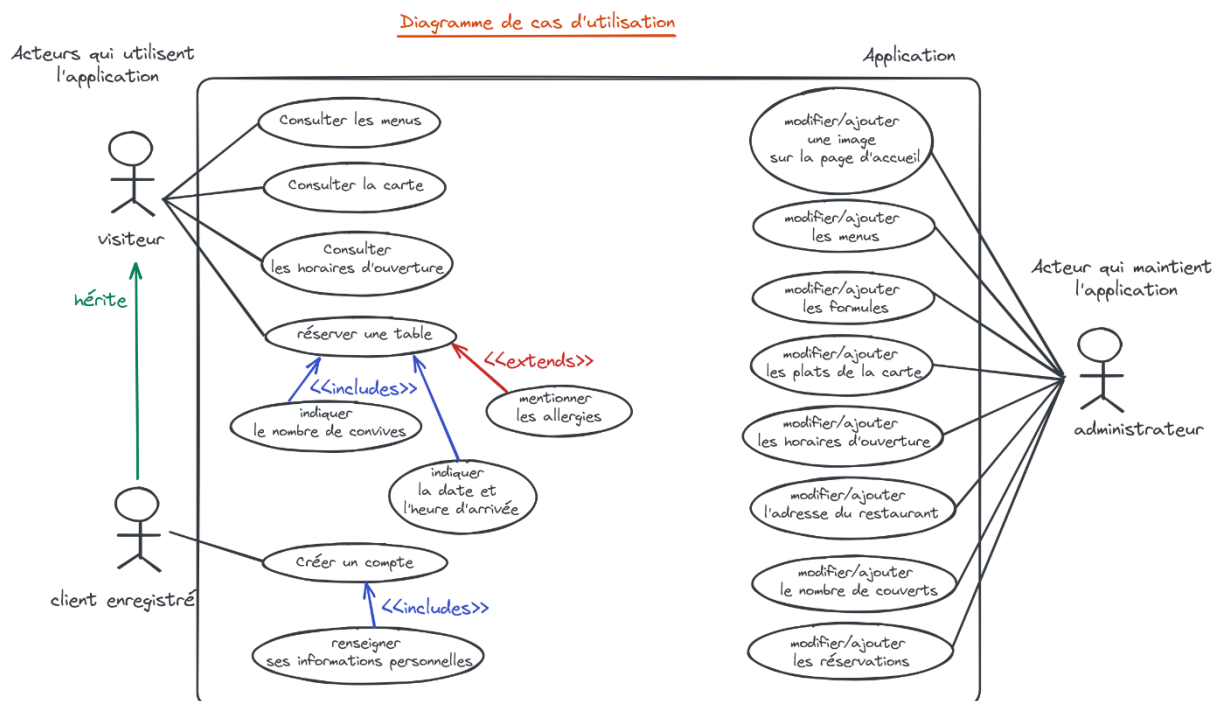
Afin de pouvoir développer un projet Symfony j'ai dû également installer Composer, un gestionnaire de dépendances pour PHP.

J'ai téléchargé et utilisé le CLI de Symfony afin de gagner en cohérence et en productivité par les commandes symfony.

J'ai publié en ligne le site chez l'hébergeur *PlanetHoster*.

## Diagramme de cas d'utilisation :

J'ai réalisé le diagramme de cas d'utilisation afin de résumer les actions d'un visiteur, un utilisateur connecté, ainsi qu'un administrateur sur le site.



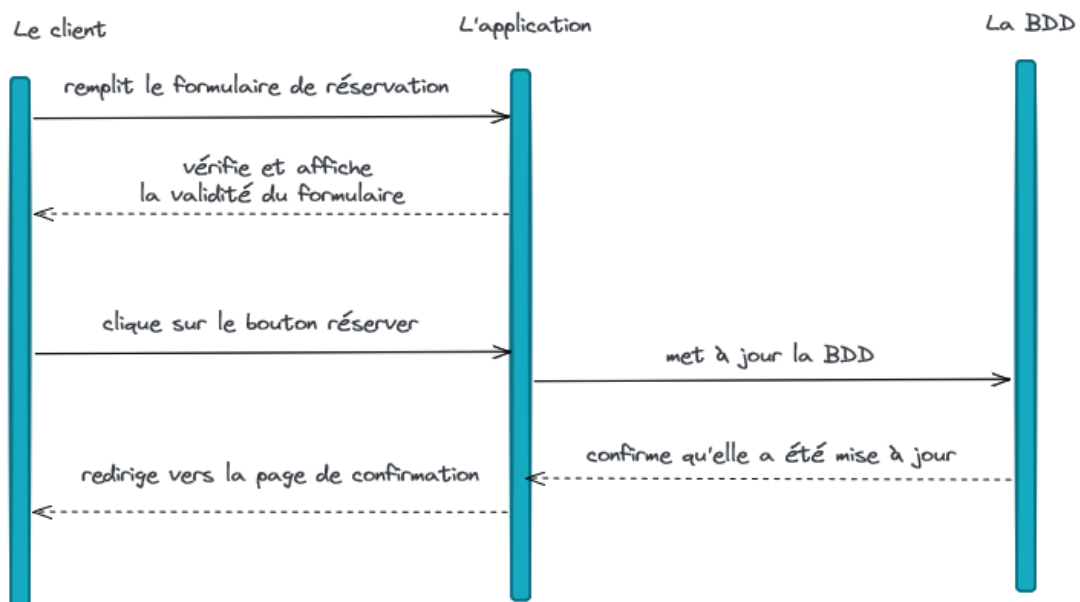
## Diagramme de séquence :

### Séquence User :

J'ai réalisé les diagrammes de séquence pour les utilisateurs qui visitent le site :

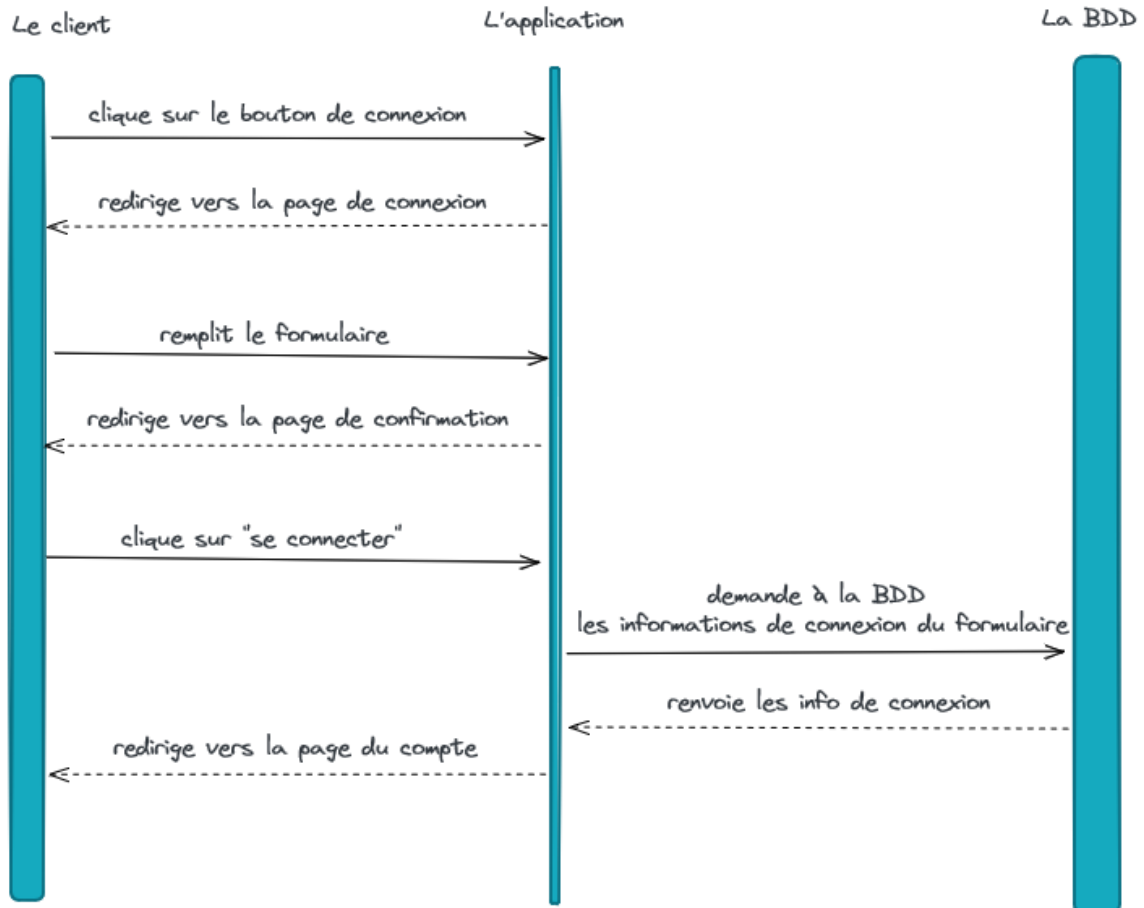
#### Réserver une table

- 1- Le client arrive sur le site web
- 2- Le client se dirige sur la page de réservation
- 3- Le client remplit le formulaire de réservation
- 4- L'application valide ou non le formulaire
- 5- Le client clique sur le bouton réserver si le formulaire est validé
- 6- L'application met à jour la BDD
- 7- La BDD confirme qu'elle a été mise à jour
- 8- L'application redirige le client vers la page de confirmation



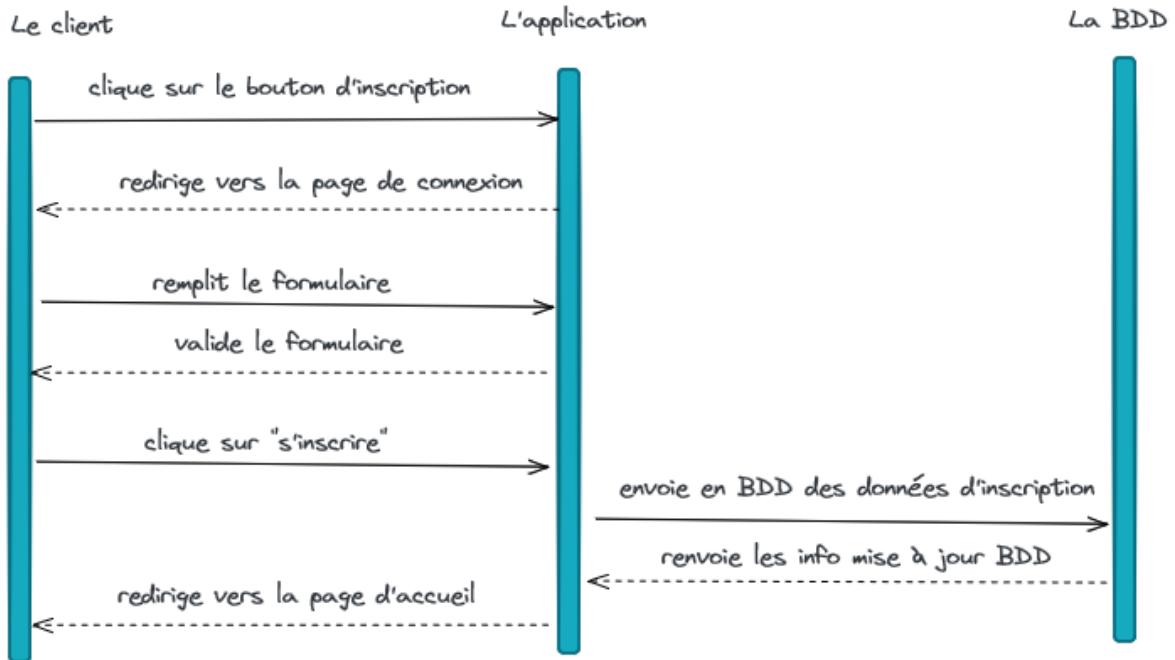
## Se connecter

- 1- Le client arrive sur le site web
  - 2- Le client clique sur le bouton de connexion
  - 3- L'application redirige le client vers la page de connexion
  - 4- Le client remplit le formulaire de connexion
  - 5- L'application valide ou non le formulaire
  - 6- Le client clique sur le bouton "se connecter" si le formulaire est validé
  - 7- L'application demande à la BDD les informations de connexion du formulaire
  - 8- L'application redirige le client vers la page d'accueil s'il est bien connecté
- L'application redirige le client vers la page de connexion si la connexion n'a pas pu se faire



### S'inscrire

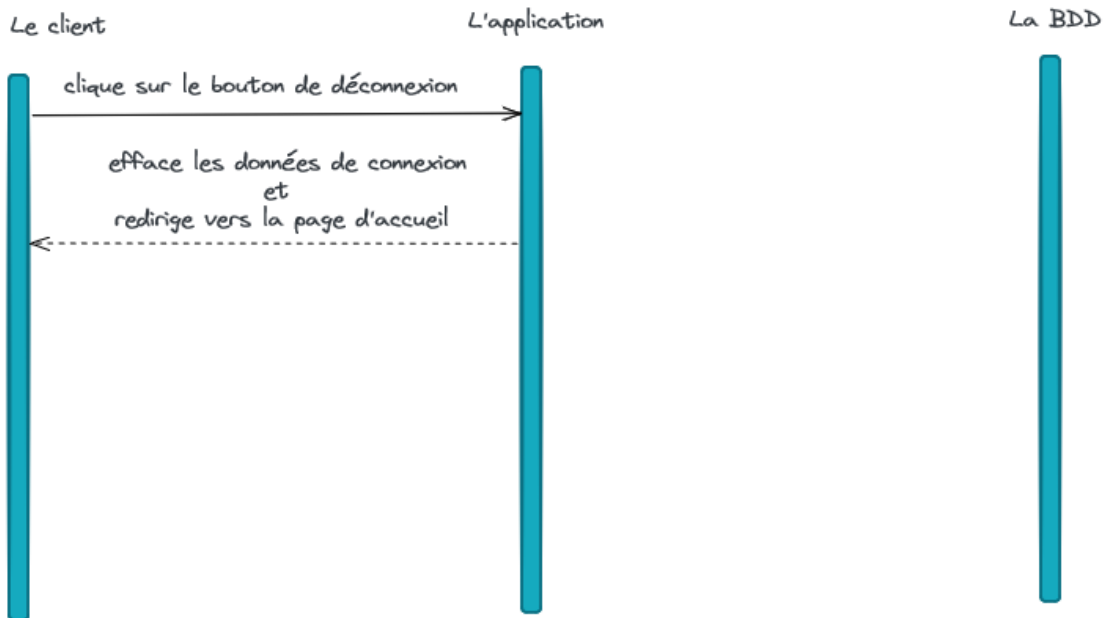
- 1- Le client arrive sur le site web
- 2- Le client clique sur le bouton d'inscription
- 3- L'application redirige le client vers la page d'inscription
- 4- Le client renseigne le formulaire d'inscription
- 5- L'application valide le formulaire
- 6- Le client clique sur le bouton "s'inscrire" si le formulaire est validé
- 7- L'application envoie en base de données les infos du formulaire d'inscription
- 8- L'application redirige le client vers la page d'accueil





### Se déconnecter

- 1- Le client clique sur le bouton de déconnexion sur n'importe quelle page du site web
- 2- L'application efface les données de connexion du client
- 5- L'application redirige le client vers la page d'accueil

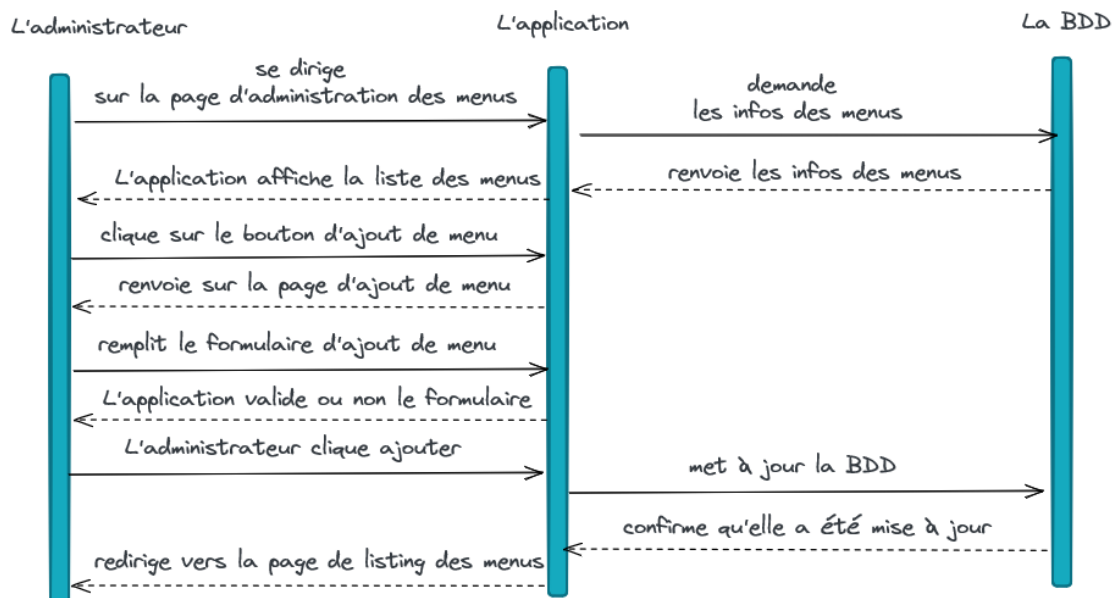


## Séquence Administrateur :

J'ai réalisé les diagrammes de séquence qui seront uniquement possibles pour les administrateurs du site :

### Ajouter un menu

- 1- L'administrateur se dirige sur la page d'administration des menus
- 2- L'application demande les infos des menus à la BDD
- 3- La BDD renvoie les infos des menus
- 4- L'application affiche la page de listing des menus
- 5- L'administrateur clique sur le bouton d'ajout de menu
- 6- L'application renvoie sur la page d'ajout de menu
- 7- L'administrateur remplit le formulaire d'ajout de menu
- 8- L'application valide ou non le formulaire
- 9- L'administrateur clique sur le bouton ajouter si le formulaire est validé
- 10- L'application met à jour la BDD
- 11- La BDD confirme qu'elle a été mise à jour
- 12- L'application redirige administrateur vers la page de listing des menus

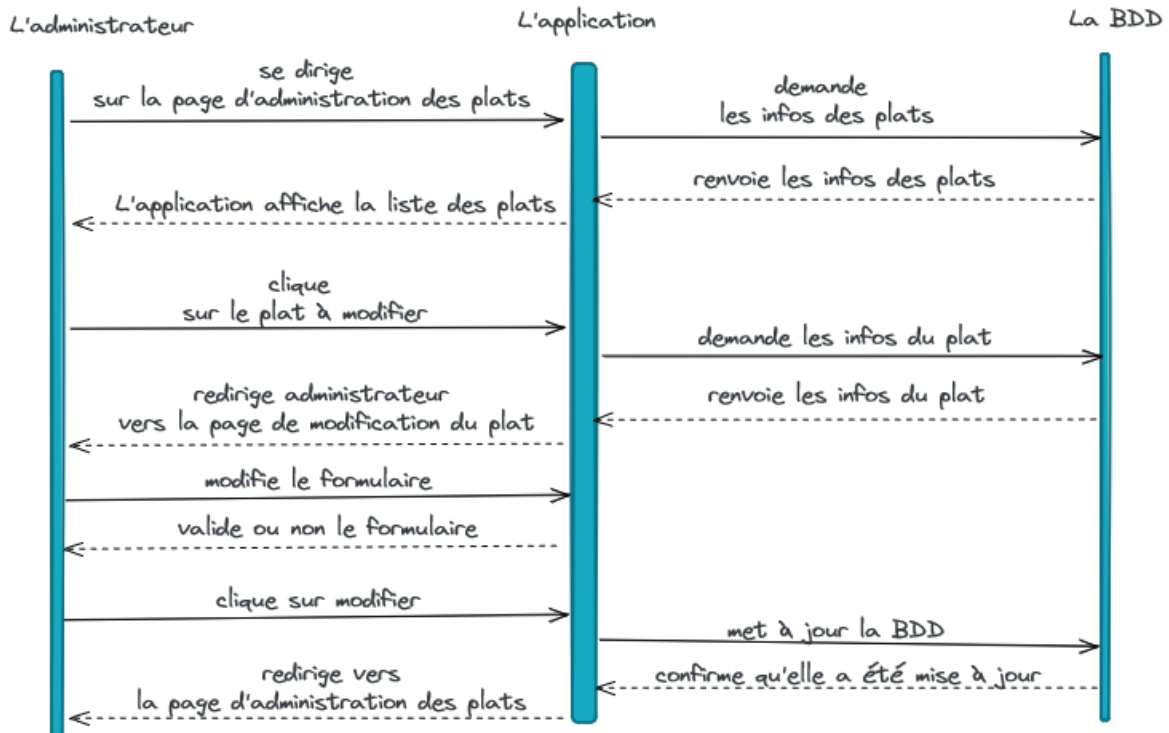


NB: Cette séquence est valable également pour les séquences administrateur:

- ajouter un menu
- ajouter une formule
- ajouter un plat
- ajouter un horaire d'ouverture

### modifier un plat

- 1- L'administrateur se dirige sur la page d'administration des plats
- 2- L'application demande les infos des plats à la BDD
- 3- La BDD renvoie les infos des plats
- 4- L'application affiche la liste des plats
- 5- L'administrateur clique sur le plat à modifier
- 6- L'application demande les infos du plat à modifier à la BDD
- 7- La BDD renvoie les infos du plat à modifier
- 8- L'application redirige administrateur vers la page de modification du plat
- 9- L'administrateur modifie le formulaire
- 10- L'application valide ou non le formulaire
- 11- L'administrateur clique sur le bouton modifier si le formulaire est validé
- 12- L'application met à jour la BDD
- 13- La BDD confirme qu'elle a été mise à jour
- 14- L'application redirige administrateur vers la page de liste des plats

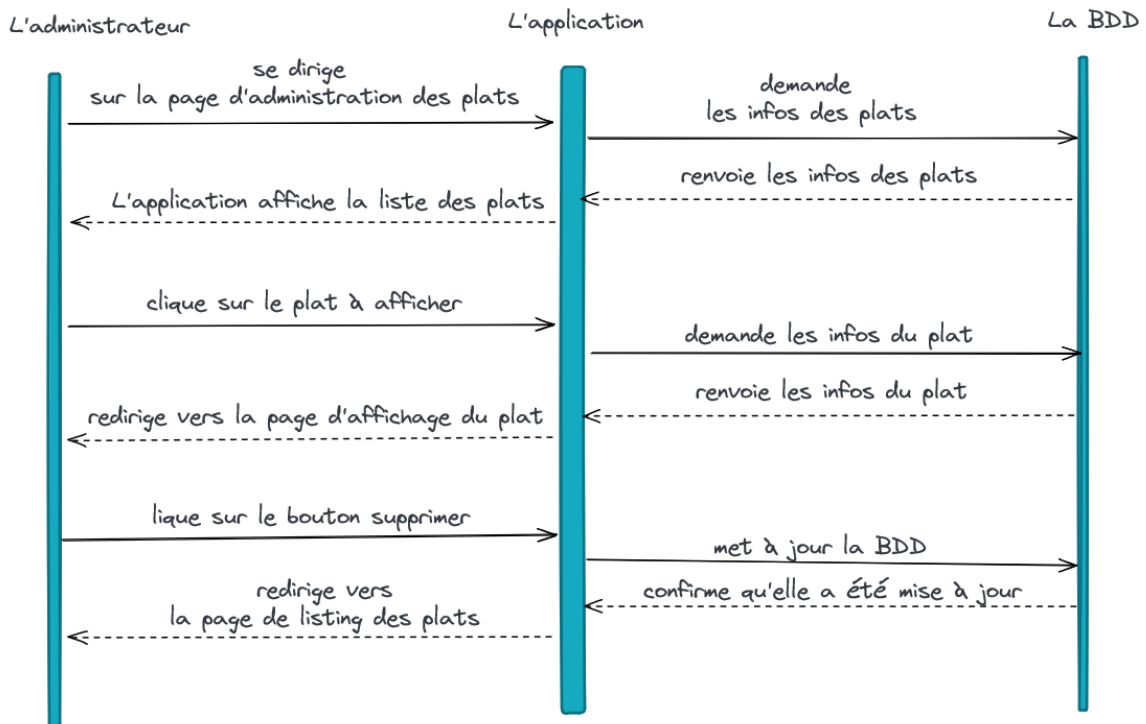


NB: Cette séquence est valable également pour les séquences administrateur:

- modifier une formule
- modifier un menu
- modifier un horaire d'ouverture
- modifier le nombre de couverts

### supprimer un plat

- 1- L'administrateur se dirige sur la page d'administration des plats
- 2- L'application demande les infos des plats à la BDD
- 3- La BDD renvoie les infos des plats
- 4- L'application affiche la liste des plats
- 5- L'administrateur clique sur le plat à afficher
- 6- L'application demande les infos du plat à afficher à la BDD
- 7- La BDD renvoie les infos du plat
- 8- L'application redirige vers la page d'affichage du plat
- 9- L'administrateur clique sur le bouton supprimer
- 10- L'application met à jour la BDD
- 11- L'application redirige administrateur vers la page de listing des plats



NB: Cette séquence est valable également pour les séquences administrateur:

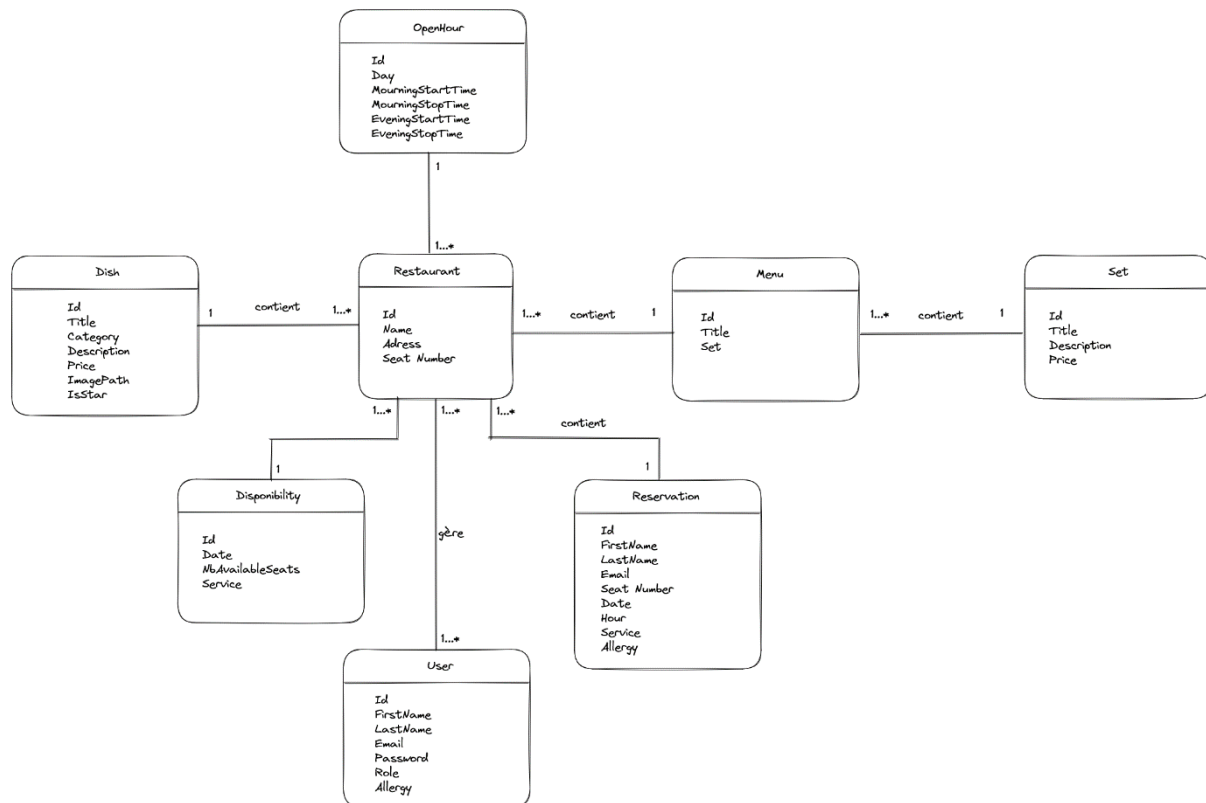
- supprimer une formule
- supprimer un menu
- supprimer un horaire d'ouverture
- supprimer le nombre de couverts

## Diagramme de classe :

### Diagramme conceptuel :

Afin de créer le diagramme de classe j'ai utilisé la méthode MERISE. Elle consiste à créer tout d'abord un diagramme conceptuel, dans lequel apparaît les différentes tables, leurs attributs et les relations entre les tables.

Ci-dessous le diagramme conceptuel pour le projet Quai Antique :



J'ai créé une entité *Restaurant*, dans lequel sera implémenté le nombre de couverts, utile pour la gestion des réservations.

J'ai créé une entité *User* qui comprend un nom, un prénom ; un email, un mot de passe, un rôle et un commentaire d'informations sur les allergies du convive.

J'ai créé une entité *Dish* pour les plats. Chaque plat aura un titre, une catégorie, une description, un prix et un chemin d'image représentant le plat. J'ai ajouté l'attribut *isStar* de manière à donner la possibilité à l'administrateur d'afficher le plat dans la page d'accueil ou non.

J'ai créé une entité *OpenHour* pour les différents horaires d'ouverture. Chaque horaire d'ouverture comprendre un jour, une heure d'ouverture et de fermeture pour le matin ainsi que pour le soir.

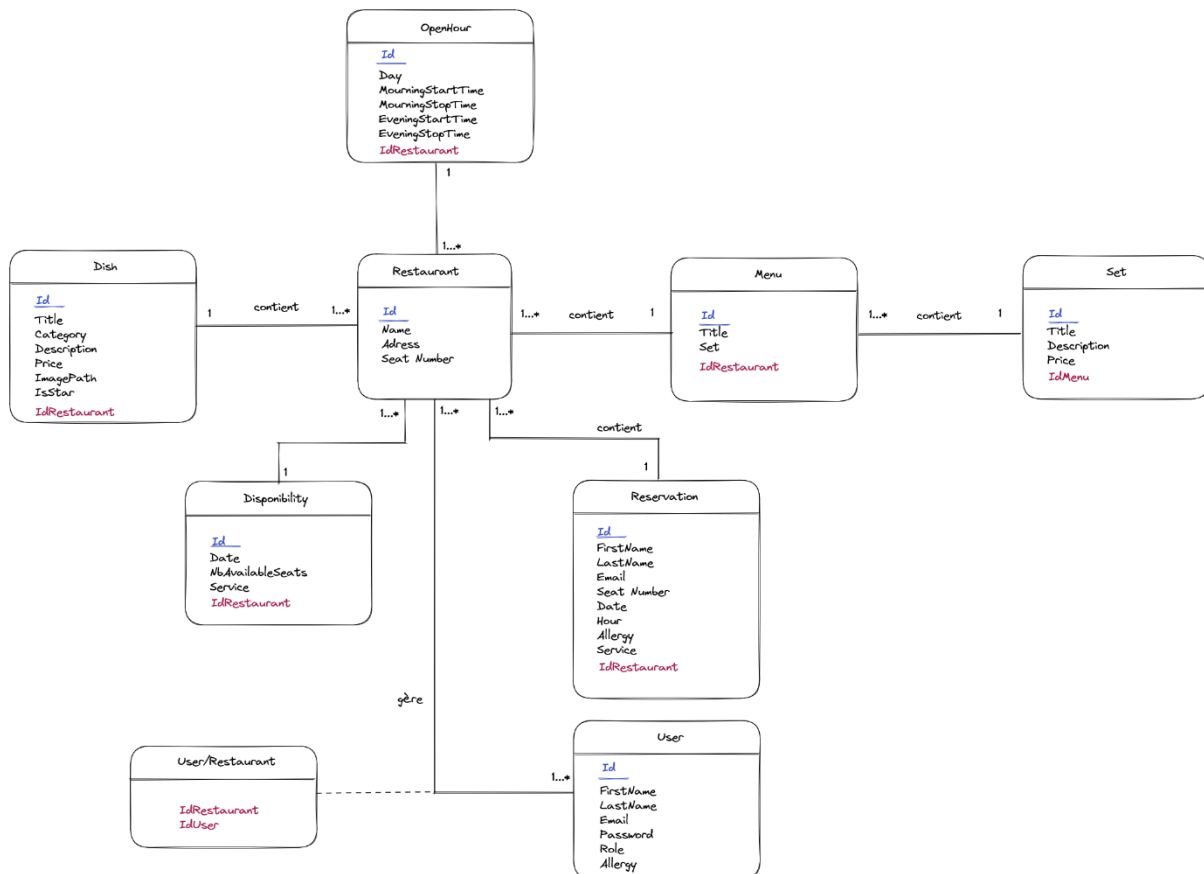
J'ai créé une entité *Menu* et une entité *Set* pour les menus et les formules.

J'ai créé une entité *Réservation* qui a comme attributs un prénom et un nom de la personne qui a réservé, un email, un nombre de places à réserver, une date, une heure d'arrivée, un service (matin ou soir) ainsi que des renseignements sur les allergies des convives.

J'ai créé une entité *Disponibilité* qui a comme attributs une date, un service, et un nombre de place restantes. Cette entité a été créée afin de mettre en relation les places du restaurant avec une date, un service, et la possibilité d'effectuer une réservation.

### Diagramme logique :

Le but du diagramme logique est de pondérer les relations entre les entités afin de pouvoir introduire les clés étrangères.

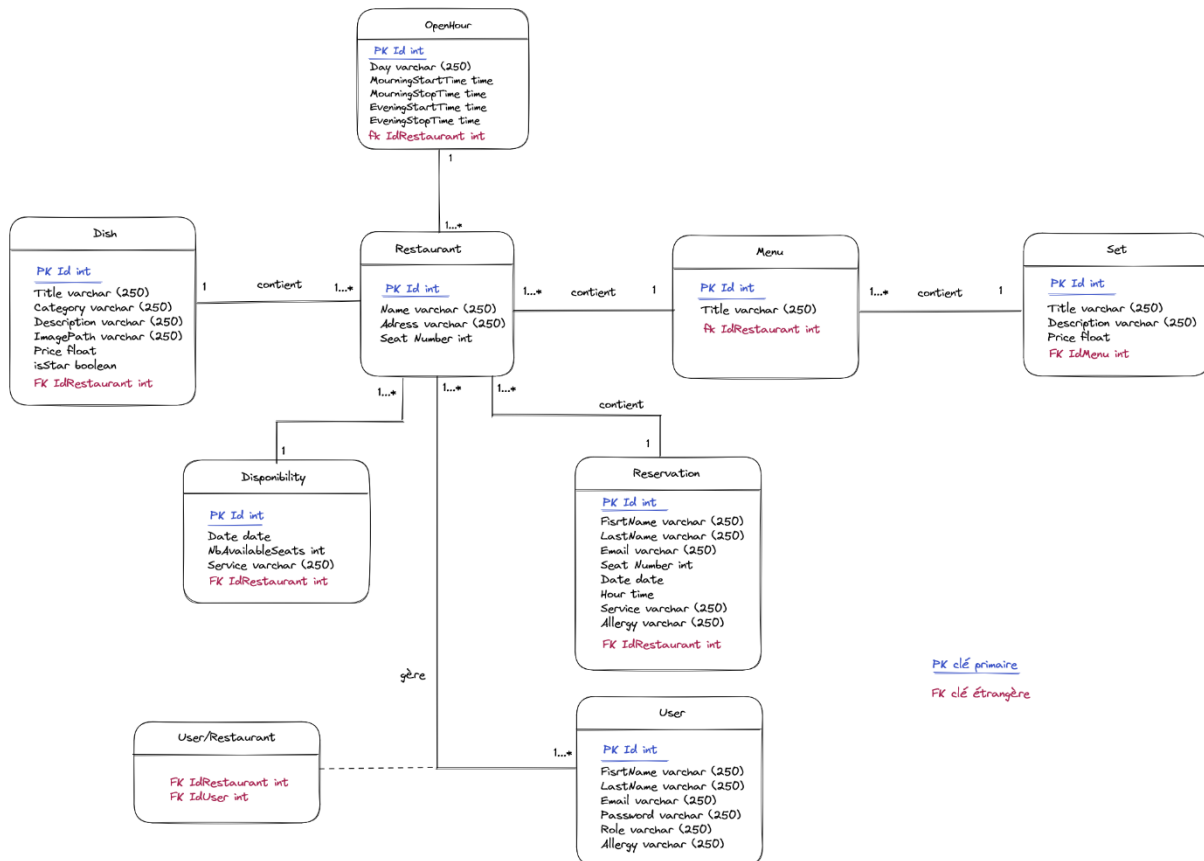


Dans ce modèle j'ai identifié les relations  $1 \Leftrightarrow 1..*$  afin d'adjoindre une clé étrangère dans la classe de poids le plus faible.

J'ai également ajouté une classe pour les relations  $1..* \Leftrightarrow 1..*$ .

## Diagramme Physique :

J'ai enfin créé le diagramme physique qui se rapproche le plus de ce que sera la base de données. Dans ce diagramme est placé les entités et leurs attributs. Leurs relations avec les éventuelles clés étrangères. Les clés primaires, ainsi que le type des différents attributs.



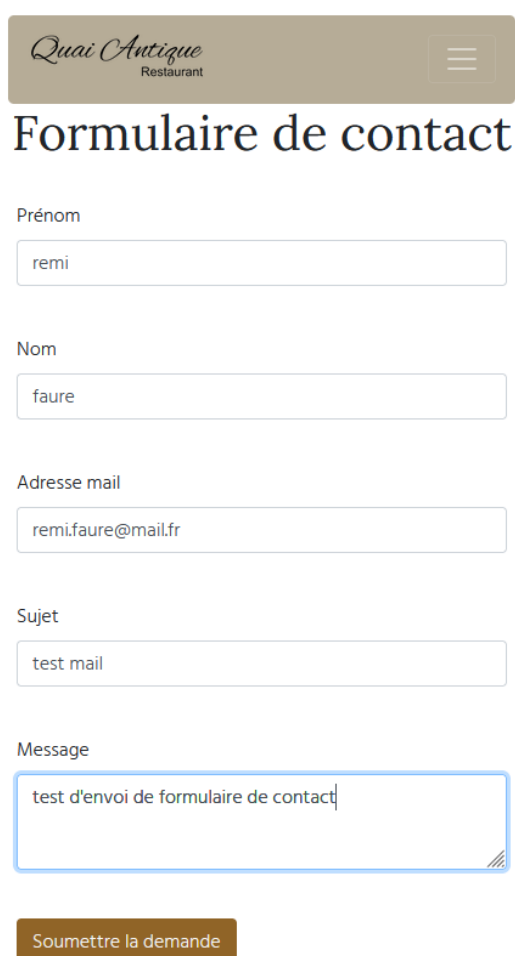
## Test de l'application :

Dans ce chapitre j'indique comment tester l'application telle qu'elle est déployée en ligne à l'adresse <https://quai-antique-studi-faure.fr>. J'évoquerai dans ce paragraphe seulement les pages demandant une interaction avec l'utilisateur. Les pages d'information (Accueil, Menu, Carte, Horaires) sont du point de vue de l'utilisateur statiques.

### Page de Contact :

Vous pouvez tester la page de contact indépendamment d'être connecté.

Vous pouvez donc remplir le formulaire et le soumettre :



Quai Antique  
Restaurant

## Formulaire de contact

Prénom

Nom

Adresse mail

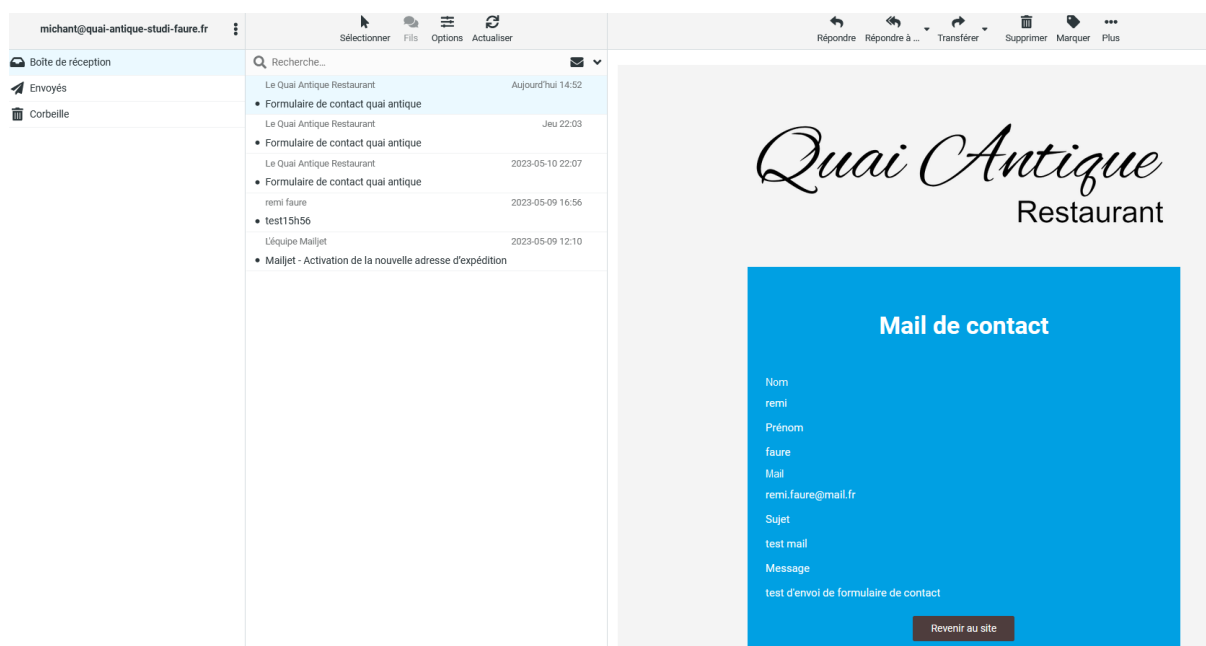
Sujet

Message

Soumettre la demande



Après avoir soumis le formulaire, un mail est envoyé à l'administrateur du site.



## Réservation de table au restaurant :

Comme pour la page de contact, il n'est pas nécessaire d'être connecté pour réserver une table sur le site.

Pour réserver une table, il faut renseigner le formulaire de la page de réservation :


The image shows a reservation form for 'Quai Antique Restaurant'. The form has a title 'Réservez une table' and several input fields: 'Prénom' (filled with 'rémi'), 'Nom' (filled with 'faure'), 'Email' (filled with 'djaroul@hotmail.fr'), 'Date de réservation' (filled with '01/06/2023'), 'Service' (dropdown menu with 'midi' selected), 'Heure de réservation' (dropdown menu with '12h15' selected), 'Nombre de couverts' (filled with '2'), and 'Commentaires éventuelles, allergies' (empty text area). A 'Réserver' button is at the bottom.

Si la réservation est acceptée, vous êtes renvoyés vers une page de confirmation.



Un mail est également envoyé conformément à ce qui est indiqué sur la page de confirmation.

**Réservation au quai antique**

 Le Quai Antique Restaurant <michant@quai-antique-studi-faure.fr>  
15:07  
À : Le Quai Antique Restaurant



Afin de pouvoir réserver, il faut choisir une date à laquelle l'administrateur a ouvert un service via l'entité *Disponibility*.

☐ Profilage
 [ Éditer en ligne ]
 [ Éditer ]
 [ Expliquer SQL ]
 [ Créer le code source PHP ]
 [ Actualiser ]

- Options

[illegible]

Si on sélectionne un jour/service qui n'a pas été ouvert par l'administrateur, un message d'erreur apparaît sur le site :

## Réservez une table

Vous ne pouvez pas réserver à cette date/horaire

Prénom

test

Nom

test

Email

test@mail.fr

Date de réservation

01 / 07 / 2023



Service

midi



Heure de réservation

12h30



Nombre de couverts

5

Commentaires éventuelles, allergies

-



Réserver

Une erreur est affichée également si l'utilisateur essaie de réserver à une date où le restaurant n'a plus de place :

Il n'y a plus assez de places pour cette date pour votre réservation

Prénom

test

Nom

test

Email

test@mail.fr

Date de réservation

01 / 07 / 2023

Service

soir

Heure de réservation

19h45

Nombre de couverts

5

Commentaires éventuelles, allergies

-

Réserver


Les utilisateurs du site :

J'ai créé 2 utilisateurs type afin de pouvoir tester ce site.

Un utilisateur connecté. Email : [user@quaiautique.fr](mailto:user@quaiautique.fr), mdp : user.

Un administrateur du site. Email : [admin@quaiautique.fr](mailto:admin@quaiautique.fr), mdp : admin.

Utilisateur connecté :



The screenshot shows the login page for 'Quai Antique Restaurant'. At the top, there is a header with the restaurant's logo and a hamburger menu icon. The main heading is 'Connexion'. Below it, there are two input fields: 'Email' with the value 'user@quaiautique.fr' and 'Mot de passe' (Password) which is masked with three dots. A 'Se connecter' (Login) button is positioned below the password field. At the bottom, a footer contains the copyright notice: '© 2023 Copyright: Le Quai Antique Restaurant'.

En tant qu'utilisateur connecté, vous pouvez accéder à votre compte, et ainsi modifier votre mot de passe ou vos informations complémentaires :



The screenshot shows the 'Mon compte' (My account) page for 'Quai Antique Restaurant'. The header is identical to the login page. The main heading is 'Mon compte'. Below it, the text reads 'Bienvenue dans votre compte Charles,' followed by 'Vous pouvez ici modifier vos données personnelles'. There are two links: 'Modifier mon mot de passe' and 'Modifier mes informations complémentaires (allergies, etc...)'. At the bottom, the footer contains the copyright notice: '© 2023 Copyright: Le Quai Antique Restaurant'.

# Changement de mot de passe

[Retour à mon compte](#)

Mon prénom

Charles

Mon nom

Michant

Mon adresse mail

user@quaiantique.fr

Mon mot de passe actuel

Veuillez saisir votre mot de passe actuel

Nouveau mot de passe

Le nouveau mot de passe

Confirmez votre nouveau mot de passe

Merci de confirmer le nouveau mot de passe

Mettre à jour

*Quai Antique*  
Restaurant

## Changement de mes informations complémentaires

[Retour à mon compte](#)

Mon prénom

Mon nom

Mon adresse mail

Commentaires | Allergies eventuelles...

Mettre à jour

Administrateur :

*Quai Antique*  
Restaurant

## Connexion

Email

Mot de passe

Se connecter

© 2023 Copyright: Le Quai Antique Restaurant

En tant qu'administrateur, vous pouvez modifier vos informations de compte comme tout utilisateur connecté.



Vous pouvez également administrer toutes les données du site en cliquant sur le lien « Administrer le site ».



A partir de cette interface, vous pouvez lister, créer, modifier ou supprimer chaque donnée clé du site :

- Les Informations du restaurant
- Les horaires d'ouverture
- Les menus
- Les formules
- Les plats
- Les disponibilités. L'administrateur peut lister chaque service. Il est en charge également d'ouvrir les places de restaurant. Il peut pour cela créer un service seul ou créer les services à partir d'une date de départ et une date de fin (illustration ci-dessous)
- Les réservations. L'administrateur peut lister les réservations par date, pouvant visualiser ainsi chaque réservation de la journée.

*Quai Antique*  
Restaurant



## Créer une nouvelle liste de disponibilités

[Revenir à la liste](#)

---

Date de début



Date de fin



Nombre de places disponibles

Enregistrer

© 2023 Copyright: Le Quai Antique Restaurant

NB : J'ai ouvert les disponibilités jusqu'au 31/07/2023. Il est donc impossible de réserver sur le site à une date ultérieure.