Applications web transactionnelles

420-4GW-BB

**Proposition de travail**

Mamma’s Restaurant Pizzeria

Travail présenté à

Amal El Kharraz

Par

Équipe Elie

Collège de Bois-de-Boulogne

23 janvier 2023

Table des matières

[Description du projet 3](#_Toc126511052)

[Diagramme de contexte 3](#_Toc126511053)

[Portée 3](#_Toc126511054)

[Fonctionnement 4](#_Toc126511055)

[Modèle de bases de données 6](#_Toc126511056)

[Interfaces 6](#_Toc126511057)

# Description du projet

Mamma Mia, propriétaire du restaurant Mamma’s Restaurant Pizzeria, est venu nous consulter afin d’agrandir la portée de sa clientèle en digitalisant les commandes, la réservation des tables et simplifier la livraison par la création d’une application web transactionnelle. À la suite de la rencontre avec notre client, nous avons établi un document qui contient tous les renseignements sur le site web demandé.

# Diagramme de contexte

# Portée

Au cours du développement de l’application, nous pensons qu’il serait possible créer un système d’inscription d’utilisateur et de connexion d’utilisateur.

Ensuite, il faudrait pouvoir connecter une base de données avec ce système, afin de pouvoir entamer à la création du système de commandes.

Si la connexion fonctionne, il serait désormais possible de pouvoir créer le système de commandes, de réservation de tables et de livraison.

Si possible, on pourrait envoyer une facture des achats dans le courriel électronique du client.

Nous ne sommes pas sûrs de vouloir implémenter un API pour gérer les cartes de crédit et un API pour les livraisons de nourriture puisqu’il notre site web ne traitera pas des vraies commandes.

Toutefois, il est possible de créer notre propre système de livraison en utilisant l’API de Google Places, afin de pouvoir simuler une vraie livraison. Le système de gestion de carte de crédit pourrait être un prototype créé qui va simuler une vraie carte de crédit.

Finalement, après l’implémentation de tous les systèmes, il va falloir sécuriser l’application en cryptant et en hachant des données dans la base de données. Si possible, nous pouvons aussi créer des tokens de session.

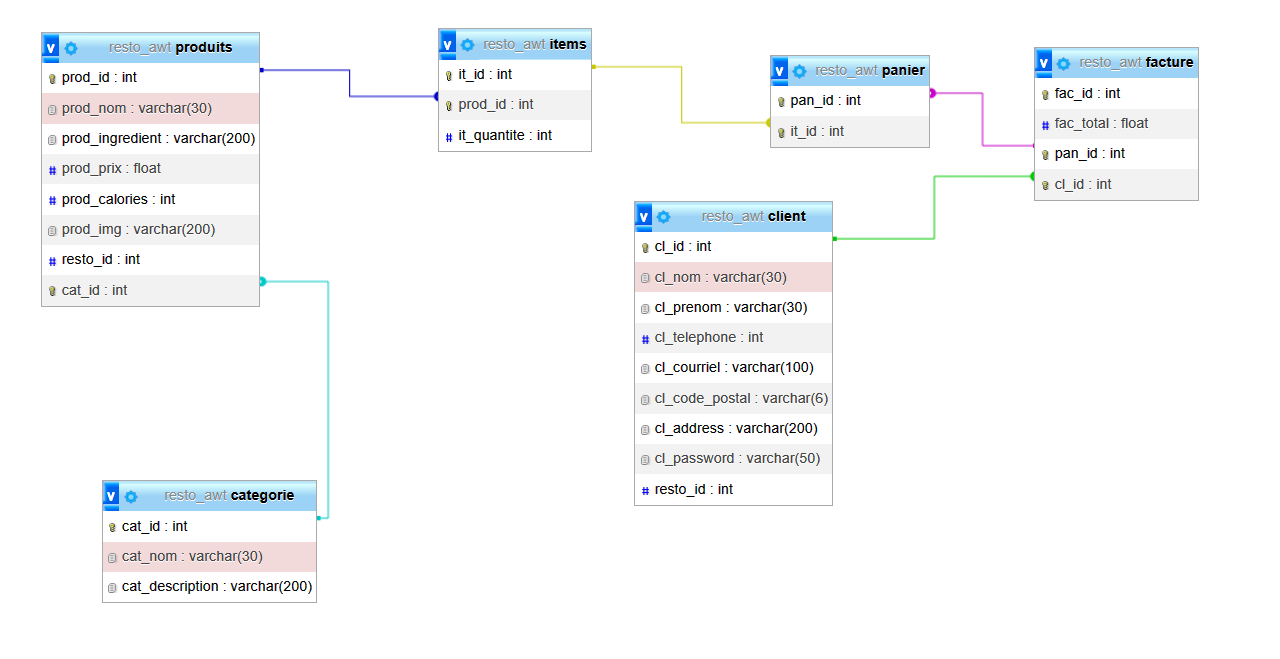
# Fonctionnement

La fonctionnalité principale de l’application web serait la commande en ligne de nourriture. Un utilisateur non connecté pourrait seulement accéder au menu, mais s’il s’inscrit ou se connecte à son compte, il pourrait ajouter des plats dans son panier. Si un ajout de items se fait à l’extérieur des heures d’affaire (8h à 20h), une alerte serait envoyée pour lui informer que cette fonctionnalité est indisponible. Ensuite, il passera à la caisse où il verra la liste de tous ses produits dans le panier. Si l’utilisateur décide de continuer l’achat, une fenêtre s’ouvrira pour qu’il rentre les informations de sa carte de paiement. Ensuite, il pourrait choisir de recevoir sa nourriture par livraison ou en allant sur place. En effet, la livraison est une fonctionnalité secondaire. Si l’utilisateur choisit de se faire livrer, il doit remplir un formulaire qui demande son code postal et son adresse. Avec l’API de Google Maps, il serait possible de calculer la distance entre le restaurant et le point de livraison. Si la distance est plus de 10 kilomètres, alors la livraison serait indisponible, sinon il serait possible d’estimer un coût pour la livraison selon la distance. Tel que soit l’option pour recevoir sa nourriture, l’utilisateur devrait confirmer sa commande en cliquant un bouton. Ensuite, une facture serait envoyée dans leur courriels. Un utilisateur employé / administrateur recevra la commande du client et pourrait supprimer sa commande quand elle serait effectuée.

La réservation des tables est une fonctionnalité secondaire qui est accessible dans la barre de navigation. L’utilisateur doit en effet être connecter à son compte sinon il va être rediriger à la page d’inscription. Quand l’utilisateur clique sur le bouton de réservation, il serait redirigé sur une autre page où il pourrait choisir de réserver une liste de table qui sont disponibles. Ensuite, une liste d’heure où la table n’est pas réservée serait affiché et le client aurait le droit de choisir seulement une option. Chaque option est une durée de 30 minutes. Le restaurant ouvre à 8h et ferme à 20h, mais la réservation pourrait s’effectuer dès 6h jusqu’à 19h30. Un client pourrait seulement réserver une table par jour. Il pourrait aussi annuler sa réservation quand il le souhaite. À la fermeture du restaurant, toutes les réservations vont être supprimés. L’utilisateur doit ensuite confirmer sa réservation en cliquant le bouton « confirmer ». Une note lui serait envoyée dans son adresse courriel pour confirmer sa réservation. L’utilisateur employé recevra ensuite une notification de réservation et pourrait s’organiser pour dégager la table.

En ce qui concerne les interfaces graphiques, il y aurait la page principale où les informations du restaurant seront présentées. Une barre de navigation pour accéder au menu, à la réservation ou à la connexion. Un utilisateur non connecté pourrait seulement accéder à la page principale et le menu. S’il décide de cliquer sur « Ajouter au panier » dans le menu, le bouton pour réserver une table ou pour se connecter, une interface graphique apparaitra sur la même page qui demanderait son courriel et son mot de passe. Si tous les champs rentrés sont corrects, il serait connecté à son compte, sinon un message en rouge lui demanderait de réessayer. Après 5 tentatives d’essais, il serait bloqué pendant 5 minutes et s’il échoue encore 5 essais, il serait bloqué pendant 30 minutes. S’il échoue encore une fois 5 essais, il serait constamment bloqué à 30 minutes. Un bouton « Mot de passe oublié? » serait disponible en dessous de l’interface. Si l’utilisateur clique ce bouton, un champ demandant son adresse courriel apparaitra. Ensuite, un code de validation de 6 chiffres lui serait envoyé dans son courriel et il aurait 5 minutes pour copier ce code et le rentrer dans le champ. S’il échoue de rentrer le bon code ou de le rentrer à temps, il pourrait regénérer un autre code après 5 minutes, mais l’ancienne ne serait plus valide. S’il rentre le bon code, il pourrait insérer son nouveau mot de passe. Un mot de passe serait seulement valide s’il respecte certain critère qui vont être mentionnées un peu plus bas. Si l’utilisateur n’a pas de compte, il pourrait créer son compte en cliquant le bouton en dessous de l’interface graphique pour se connecter. Un autre pop-up remplacerait l’interface de connexion. L’utilisateur devrait insérer son nom (moins de 30 caractères), son prénom (moins de 30 caractères), son numéro de téléphone (pas obligatoire), son adresse courriel (moins de 100 caractères) et son mot de passe. Le mot de passe serait seulement valide s’il contient au moins 8 caractères, au moins une lettre majuscule, au moins un chiffre, au moins un symbole et ne doit pas dépasser 50 caractères. Si l’utilisateur se connecte ou crée son compte, il pourrait ensuite accéder à la l’ajout des items dans le panier et la réservation des tables.

# Modèle de bases de données



Pour commencer, chaque client possède un nom, un prénom, un numéro de téléphone, une adresse courriel et un mot de passe lors de la création de son compte. Dépendamment du restaurant où il commande, le resto\_id changera. Son code postal et son adresse serait demandé s’il choisit de recevoir sa nourriture par livraison.

Dans le menu, les produits seront divisés dans des catégories distincts. Chaque catégorie contient un nom et une description des aliments qu’elle contient.

Chaque plat alimentaire contient un nom, les ingrédients qui la compose, le prix unitaire, le nombre de calories, une image et un resto\_id pour définir le produit vient de quel Mamma’s Restaurant Pizzeria.

Quand un produit est sélectionné, celui-ci deviendra un item où le client pourrait choisir la quantité à ajouter au panier.

Ensuite, cet item sera mis au panier. Celui-ci contient les id de chaque item.

Après le paiement, le prix total sera calculé et la facture sera attaché au client ciblé.

# Interfaces