Conception, développement et réalisation d’un système d’information de recherche, présentation et gestion de restaurants.

Cycle d’abstraction :

1. Système d’information manuel

Le client doit posséder les contacts du restaurant ou se présenter physiquement dans un restaurant pour obtenir : le menu du jour et le prix, la réservation, etc.

Le restaurant quant à lui doit utiliser des supports physiques tels des tableaux, des coupons, etc. pour afficher et présenter les informations importantes telles que les menus et prix, les contacts, la localisation géographique, jour et heures d’ouverture et de fermeture du restaurant.

1. Expression des besoins
2. Informations requises

Pour mener à bien notre travail, nous aurons besoin des informations suivantes :

* Restaurants : nom, lieu, numéro de téléphone, adresse email, jours et heures d’ouverture et de fermeture, une adresse URI du site web si le restaurant en possède, image profil, coordonnées géographiques du restaurant (longitude, latitude), etc. ;
* Les menus proposés par le restaurant ;
* Client (nom, numéro téléphone, mot de passe, ville résidence).

1. Fonctionnalités

* Permettre une recherche rapide et efficace de toutes les informations concernant une restaurant (emplacement, menu, etc.);
* Un utilisateur doit pouvoir donner son avis sur un restaurant ;
* Un utilisateur doit pouvoir obtenir les notifications d’un restaurant ;
* Un utilisateur doit pouvoir passer des commandes et faire des réservations dans un restaurant ;
* Présenter l’utilisateur et un restaurant dur une carte géographique, la distance les séparant et guider ainsi l’utilisateur vers le restaurant;
* Un utilisateur doit pouvoir inviter une personne dans un restaurant via une ***Notification***;
* Un utilisateur doit pouvoir discuter avec un autre utilisateur ;

1. Modèle conceptuel

L’application sera une PWA (Progressive Web App) ; 

Le système permettra une inscription pour restaurants en plusieurs étapes ou pas;

1. Modèle Conceptuel de données

Liste des classes d’entité :

Avis : id\_avis, message, date publication;

User : id\_user, nom, prénom, téléphone, ville résidence ;

Commande : id\_commande, régler, livrer, date commande, date règlement;

Réservation : id\_réservation, nombre place, date réservation, infos supplémentaires, date consommation ;

Notification : id\_notification, envoyé, reçu, lu, date envoi, date réception, date lecture, date\_accept\_reject ;

Message : id\_message, message, envoyé, reçu, lu, date envoi, date réception, date lecture ;

Restaurant : id\_restaurant, nom, téléphone, email, lieu, site web, latitude, longitude, accept\_command, accept\_resa, accept\_livreur;

Menu : id\_menu, référence, date ajout ;

Plat : id\_plat, nom, prix ;

Jour : jour ;

Menu : id\_menu, référence, date ajout.

1. Diagramme de cas d’utilisation
2. Diagramme de classe

**Reference : D:\Allresteau\diagramme\_classe.mlo**

1. Modèle logique
2. Modèle logique de données

Client (id\_client, nom, téléphone)

Commande (#id\_client, #id\_restaurant, id\_commande, date\_commande, description, état, coords)

Réservation (#id\_client, #id\_restaurant, id\_reservation, nombre\_place, date\_prevu\_resa, infos\_sup, état)

Menu\_restaurant (#id\_restaurant, #id\_menu)

Avis (id\_avis, #id\_restaurant, pseudo, message, date\_publication)

Menu (id\_menu, référence, prix)

Coordonnées (id\_coords, latitude, longitude)

Restaurant (id\_restaurant, #id\_coords, nom, téléphone, email, lieu, url)

**Reference : D:\Allresteau\allrestau.mlr**

1. Système automatisé
2. Formes normales
3. 1FN

Toutes les relations sont en première forme normale parce que tous les domaines des attributs sont des valeurs atomiques.

1. 2FN

Toutes les relations sont en 2FN parce qu’elles sont en 1FN et de plus tout attribut non clé dépend totalement de la clé.

Exemple :

id\_client  nom ;

id\_restaurant  email ;

1. 3FN
2. BCNF