

ÉCOLE NATIONALE SUPÉRIEURE EN SCIENCES ET TECHNOLOGIES DE L'INFORMATIQUE
ET DU NUMÉRIQUE

Rapport du mini-projet : Gestion d'un restaurant

Réalisé par :

MOKHTARI Billal

OULD KACI Amine

Groupe: 1

Module:

Bases De Données

1 SOMMAIRE

2	INTRODUCTION.....	3
3.1	Contexte :	4
3.2	Objectifs :.....	4
5.1	Base de données :.....	5
4.2.	Les outils employés :.....	6
5.3	Design et fonctionnalités :	7
5.3.1	Client :.....	7
5.3.2	Administrateur :.....	9
6	CONCLUSION	11

2 INTRODUCTION

Notre projet consiste à la conception et la mise en œuvre d'un site Web dynamique pour la gestion d'un restaurant. En effet, le but de notre travail est d'offrir une application web générale qui facilite et automatise la gestion des commandes, des clients, des tables et des employés. Son organisation est comme suit : Dans le premier chapitre, nous allons détailler la conception du modèle Entité-Association et la création de la base de données. Dans le chapitre qui suit, nous présentons d'avantage le site Web développé, y compris la partie client et la partie administrateur, en réalisant des captures écran, avec des explications, sur les différentes pages développées de notre site Web. Nous terminons ce présent rapport par une conclusion générale.

3 PROJET ET OBJECTIFS :

3.1 CONTEXTE :

Les restaurants ont connu une situation difficile. Pour aider les restaurateurs, nous faisons le point dont la numérique peut accompagner leur reprise tant en matière d'efficacité, fluidité, organisation et du développement commercial. C'est-à-dire, remédier aux obstacles affrontés par les restaurants classiques en termes de technologies.

Au début le client doit s'inscrire à ce site Web. Ensuite, il aura la possibilité de lancer ses commandes, à travers un panier dans lequel il rajoute ses consommations, une fois l'administrateur lui réserve une table.

3.2 OBJECTIFS :

L'objectif principal de ce projet est de mettre en évidence toutes les connaissances acquises durant les séances du Cours, TD et TP, afin de renforcer encore loin les notions de bases de données relationnelles et le langage SQL.

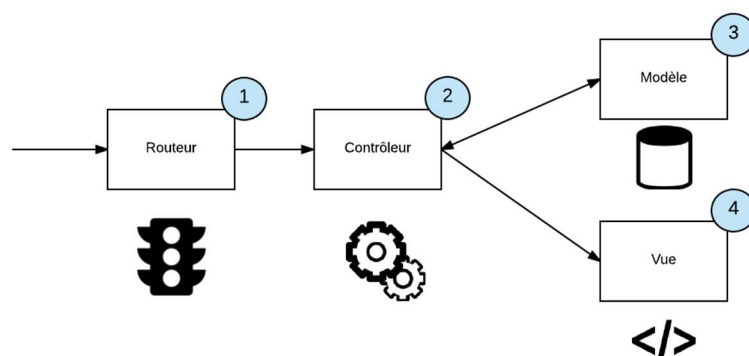
4 L'ARCHITECTURE GLOBALE DE L'APPLICATION :

Notre site Web est divisé en deux applications, autrement dit, deux parties ;

- Partie Client : c'est celle utilisée par le client pour interagir avec le système en s'inscrivant, se connectant et lançant ses commandes.
- Partie Administrateur : utilisé essentiellement par les administrateurs du site (du restaurant également) afin de communiquer avec les clients et les employés.

Nous avons opté pour une architecture trois-tiers MVC (Model View Controller) puisque c'est un site web dynamique, d'autres termes, il permet une interaction avec les données.

Nous expliquerons ces deux dernières plus en détails dans les prochaines sections.



Pour faciliter la maintenance, il est plus simple de passer par un contrôleur *frontal*, qui va jouer le rôle de *routeur*. Son objectif va être d'appeler le bon contrôleur (on dit qu'il *route* les requêtes).

5 DEVELOPPEMENT :

5.1 BASE DE DONNEES :

Modèle Entité-Association - Gestion d'un restaurant

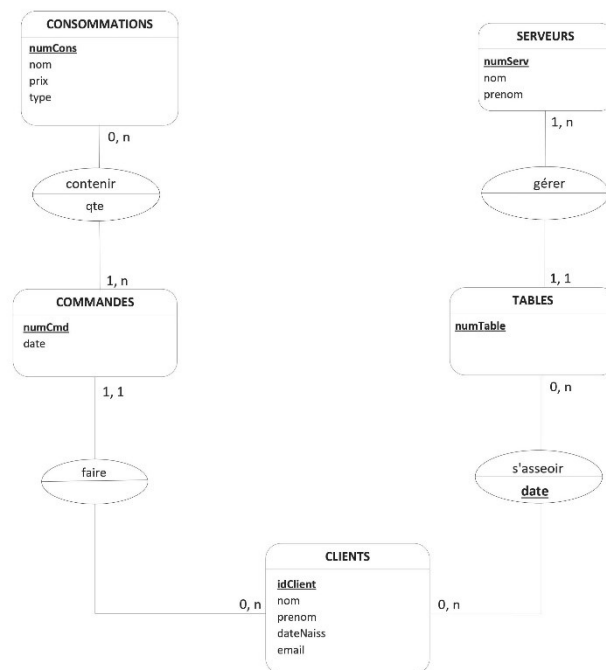


Figure 1

La figure ci-dessus, représente le modèle de données entité-association pour la gestion d'un restaurant (on ne s'intéresse qu'aux repas de midi).

- La table CONSOMMATIONS est identifiée par un numéro unique, et caractérisée par un nom unique, son type et son prix associé.
- La table CLIENTS est identifiée par un identifiant client, et caractérisée par un nom, un prénom, une date de naissance et un email. Un email ne peut être associé qu'à un et un seul client, par contre un client peut en disposer d'un ou plusieurs.
- La table SERVEURS est identifiée par un identifiant client, et caractérisée par un nom et un prénom.
- La table TABLES est identifiée par son numéro.

- La table COMMANDES est identifiée par un numéro, et caractérisée par une date.

Définition des contraintes d'intégration :

- Une commande contient une ou plusieurs consommations avec une quantité donnée. Une consommation peut être contenue dans aucune ou plusieurs commandes.
- Un client peut effectuer zéro ou plusieurs commandes. Une commande est associée à un et un seul client.
- Un client peut s'asseoir dans plusieurs tables à des dates différentes. Une table peut être occupée par zéro ou plusieurs clients en même temps.
- Un serveur gère une ou plusieurs tables. Mais une table ne peut être gérée que par un seul serveur.

Le langage utilisé en cette matière est le langage SQL pour des raisons pédagogiques.

Schéma relationnel :

CONSOMMATIONS(numCons, nom, prix, type, photo)

COMMANDES(numCmd, dateCmd, #client, etat)

CONTENIR(#consommation, #commande, qte)

CLIENTS(idClient, nom, prenom, dateNaiss, email, tel, mdp)

TABLES(numTable, #serveur)

ASSEOIR(#numTable, #client, dateServ, etat)

SERVEURS(numServ, nom, prenom)

4.2. LES OUTILS EMPLOYES :

	Front-end	Back-end
Application client	<ul style="list-style-type: none"> - React js : Framework de JavaScript - Material UI 	<ul style="list-style-type: none"> - Node js (Express js plus exactement)
Application Administrateur	<ul style="list-style-type: none"> - Bootstrap - AJAX - JavaScript 	<ul style="list-style-type: none"> - PHP

Le choix des langages à utiliser dépend principalement du besoin. Par exemple, nous avons choisi React js avec la librairie MUI pour implémenter l'interface de la partie client parce qu'il la rend attractive. Contrairement à l'interface de la partie administrateur qui est réalisée par AJAX, c'est dans l'objectif d'avoir une communication en temps réel avec les clients du fait qu'il permet de modifier partiellement la page affichée par le navigateur sans avoir besoin de recharger la page entière. Avec cette technique l'envoi des données se fait silencieusement et Javascript permet de manipuler l'interface et d'afficher les données de manière dynamique.

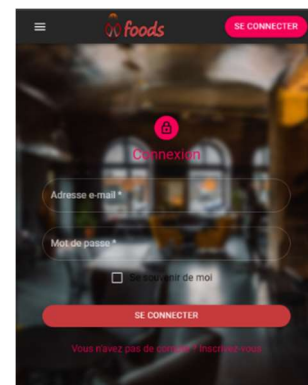
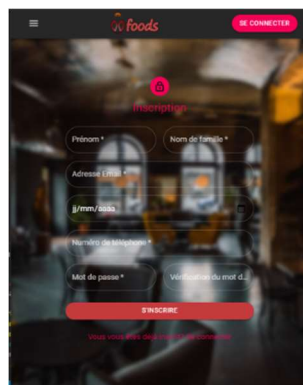
5.3 DESIGN ET FONCTIONNALITES :

5.3.1 Client :

Le client se retrouve face à une page d'accueil avec une liste de consommations affichée au-dessous de la page. C'est ce qui l'attire en première vue.

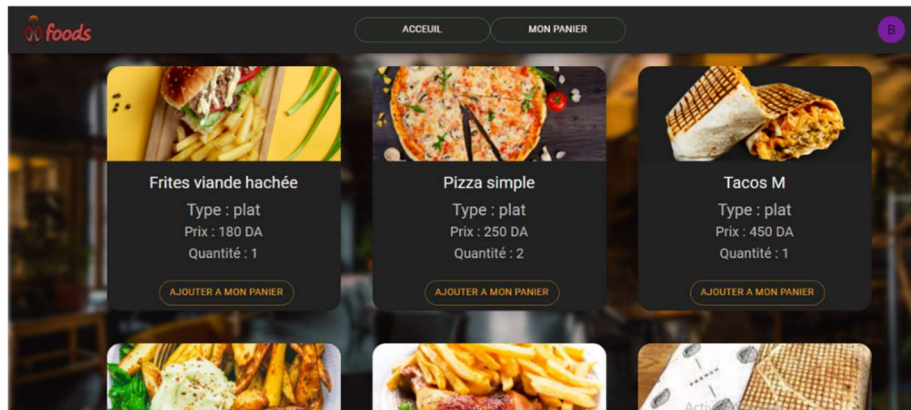


Ensuite, il aura la possibilité de se connecter ou de s'inscrire pour qu'il puisse accéder aux fonctionnalités proposées par le site.

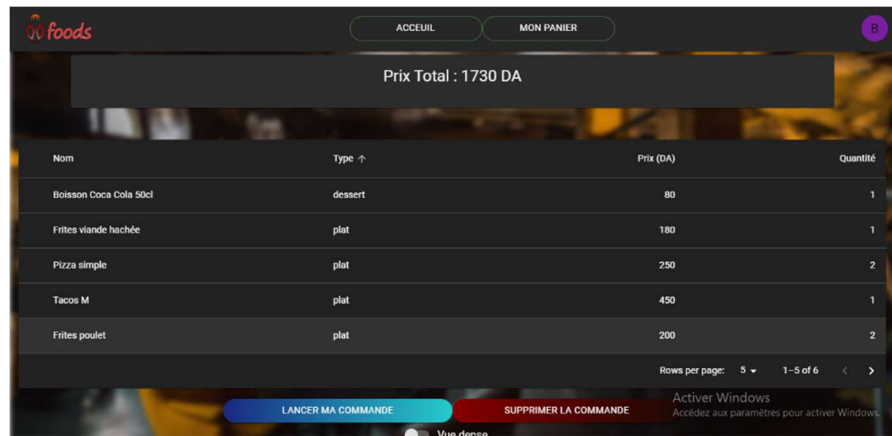


Après que l'utilisateur s'y connecte, le menu de consommations disponibles au restaurant s'affiche avec leur nom, type et prix. Il ajoute à son panier autant

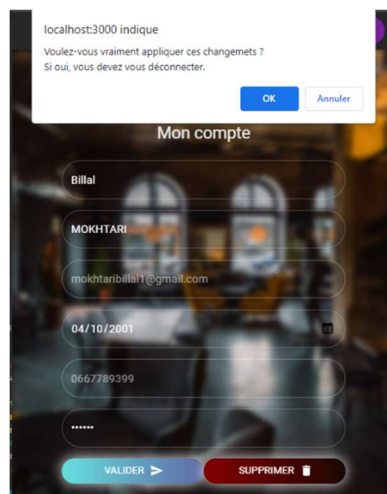
de consommations qu'il veut (avec affichage de la quantité pour chaque consommation).



Le client peut confirmer ou supprimer sa commande. Cependant, il ne pourra la confirmer qu'à après avoir occupé une table.



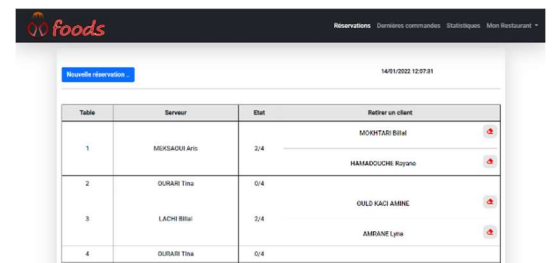
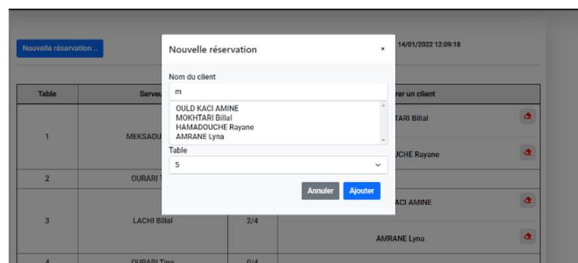
Chaque inscrit aura la possibilité de modifier ses coordonnées (sauf son email et son numéro de téléphone) ou supprimer carrément son compte.



5.3.2 Administrateur :

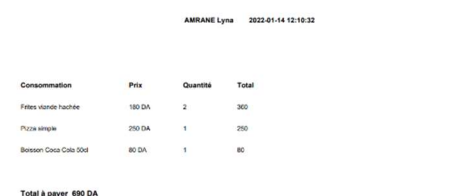
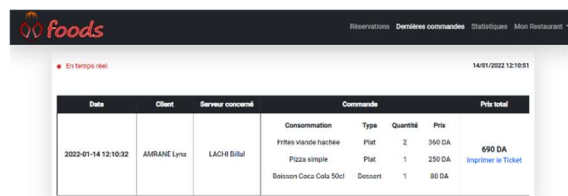
L'administrateur du restaurant dispose d'un tableau de bord permettant de :

- Réserver une table libre pour le client. On libère sa place lorsqu'il

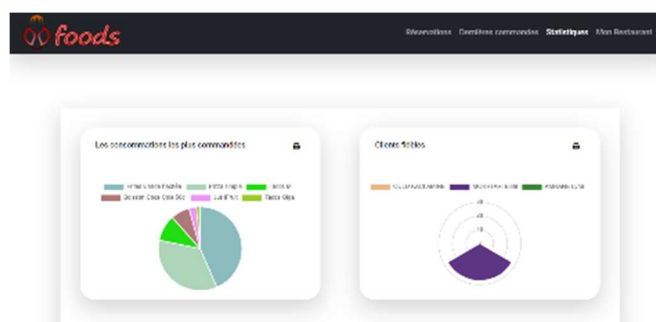


quitte le restaurant.

- Les commandes arrivent en temps réel à l'administrateur, un ticket est automatiquement attribué au client concerné.



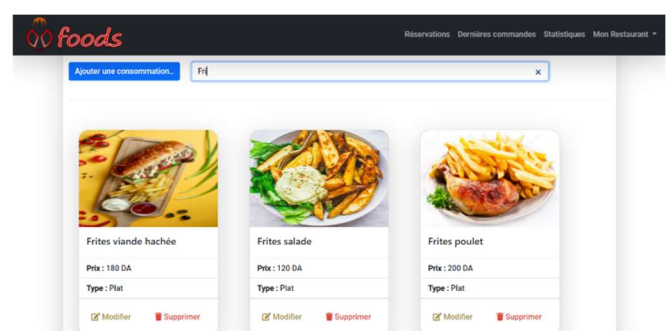
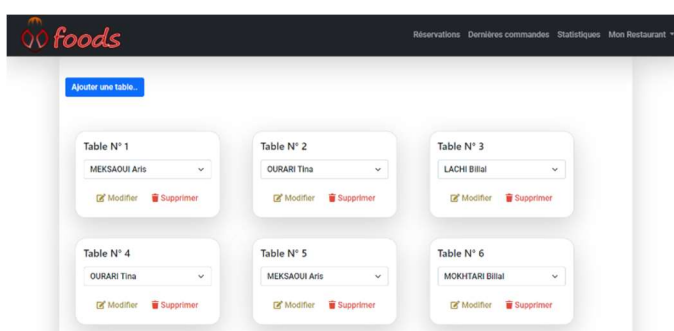
- L'administrateur peut également visualiser les consommations les plus commandées, les clients fidèles, le chiffre d'affaires, ... sous forme de graphes pouvant être imprimés.

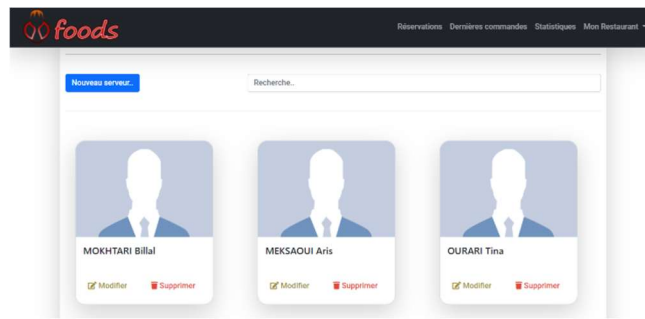


LISTE DES CONSOMMATIONS LES PLUS COMMANDEES

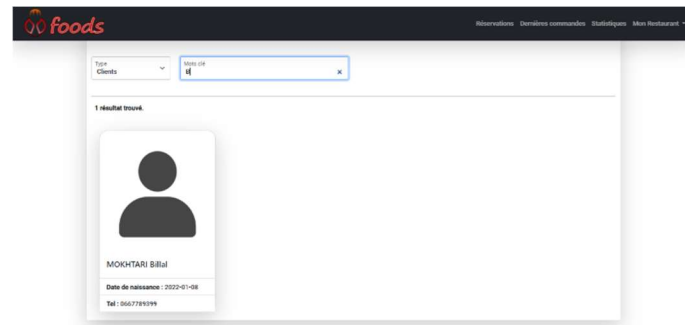
Consommation	Prix unitaire	Type	Quantité total vendue
Frites viande hachée	180	plat	18
Pizza simple	250	plat	74
Tacos M	250	plat	7
Boisson Coca Cola 50cl	80	boisson	1
Frites poulet	200	plat	7
Tacos Grande	280	plat	1

- L'application dispose d'une interface pour ajouter des consommations, des tables et des serveurs. La modification, la suppression et la recherche sont possibles.





- Bonus : L'administrateur (ou les administrateurs) du site peut effectuer des recherches avancées suivant un filtre particulier.



6 CONCLUSION

Nous sommes satisfaits du travail que nous avons fourni et son résultat. Cependant, toutes nos idées d'amélioration auraient pu être poussées plus loin, sauf que nous n'avons pas pu les réintégrer toutes illico par le manque du temps vu que c'est un mini projet.

En conclusion, ce mini-projet nous a permis de renforcer nos connaissances en bases de données, d'apprendre des nouveaux outils de développement et acquérir des notions en matière de conception d'architecture globale du logiciel. Ceci dit, ce projet nous a beaucoup plu et nous a donné envie de continuer à développer des applications web.