Documentation technique

Vite & Gourmand – Application de gestion traiteur

I. Réflexions technologiques initiales

1. Analyse du besoin

L'objectif du projet était de concevoir une application web complète permettant la gestion d'une entreprise traiteur avec plusieurs niveaux d'accès :

- Visiteur
- Client
- Employé
- Administrateur

L'application devait permettre :

- La gestion des menus et plats
- La gestion des commandes
- La gestion des utilisateurs
- La validation des avis
- Le suivi des statuts de commandes
- L'affichage de statistiques administratives
- Un déploiement en ligne accessible au jury

Il était donc nécessaire de choisir :

- Une architecture claire
- Une technologie robuste et simple à déployer
- Une base de données relationnelle adaptée aux dépendances fortes entre entités

2. Choix technologiques

🔹 **Front-end : HTML5 / CSS3**

Le front-end a été développé en HTML5 et CSS3 afin de :

- Garantir la compatibilité navigateur

- Respecter les standards web
- Assurer une structure sémantique propre
- Permettre une mise en page responsive
- Faciliter l'accessibilité

🔹 **Back-end : PHP (architecture MVC)**

PHP a été choisi pour :

- Sa simplicité d'intégration avec MySQL
- Sa compatibilité avec l'hébergement mutualisé
- Sa rapidité de mise en œuvre

L'architecture MVC (Model – View – Controller) a été adoptée afin de :

- Séparer la logique métier des vues
- Améliorer la maintenabilité du code
- Faciliter les évolutions futures
- Structurer clairement le projet

🔹 **Accès base de données : PDO**

PDO a été utilisé pour :

- Sécuriser les requêtes via requêtes préparées
- Éviter les injections SQL
- Uniformiser l'accès aux données

🔹 **Base de données : MySQL**

Une base relationnelle a été choisie pour :

- Garantir l'intégrité référentielle
- Gérer les relations entre entités (utilisateur, commande, menu…)
- Assurer la cohérence des données

II. Configuration de l'environnement de travail

1. Environnement de développement

- Système d'exploitation : Windows
- Serveur local : XAMPP (Apache + MySQL)
- Base de données : phpMyAdmin
- IDE : Visual Studio Code
- Gestion de version : Git
- Hébergement distant : InfinityFree

2. Gestion des variables sensibles

Un fichier .env a été utilisé pour stocker :

- DB_HOST
- DB_NAME
- DB_USER
- DB_PASS

Ce fichier est exclu du dépôt via .gitignore.

3. Gestion des branches Git

Le projet respecte les bonnes pratiques Git :

- Branche principale : main
- Branche de développement : develop
- Chaque fonctionnalité développée sur une branche dédiée
- Merge vers develop
- Puis validation finale vers main

4. Outil de gestion de projet

Un tableau Trello a été utilisé afin de :

- Organiser les fonctionnalités
- Suivre l'avancement
- Prioriser les tâches

- Structurer le développement par itérations

Le projet a été découpé en fonctionnalités indépendantes, développées progressivement.

III. Modèle conceptuel de données (MCD)

Le modèle conceptuel de données est présenté en annexe (diagramme joint).

1. Entités principales

- utilisateur
- role
- commande
- commande_statut
- menu
- plat
- allergene
- avis
- menu_plat
- plat_allergene

2. Relations importantes

- Un utilisateur possède un rôle
- Un utilisateur peut passer plusieurs commandes
- Une commande possède un statut
- Un menu contient plusieurs plats
- Un plat peut contenir plusieurs allergènes
- Un client peut déposer plusieurs avis

La base utilise des clés étrangères afin d'assurer :

- La cohérence des relations
- La suppression contrôlée
- La prévention des incohérences de données

IV. Diagramme de classe

Le diagramme de classe est joint en annexe.

1. Acteurs

Visiteur

- Consulter les menus
- Consulter les plats
- Consulter les avis validés
- S’inscrire
- Se connecter

Client

- Créer une commande
- Consulter ses commandes
- Annuler une commande
- Déposer un avis

Employé

- Modifier le statut d’une commande
- Valider ou refuser un avis
- Créer un menu
- Créer un plat

Administrateur

- Gérer les utilisateurs
- Gérer les commandes
- Gérer les avis
- Créer menus et plats
- Consulter les statistiques

V. Diagramme de séquence

Le diagramme de séquence est fourni en annexe.

Exemple : Création d'une commande

1. L'utilisateur se connecte.
2. Il sélectionne un menu.
3. Il renseigne le nombre de personnes.
4. Le système calcule automatiquement le prix total.
5. La commande est enregistrée en base.
6. Un statut initial est attribué.
7. Une confirmation est affichée.

Le diagramme montre l'interaction entre :

- Utilisateur
- Controller
- Model
- Base de données

VI. Sécurité mise en place

1. Authentification

- Système de session PHP
- Vérification du rôle utilisateur
- Accès restreint selon le niveau (Client / Employé / Admin)

2. Protection des données

- Requêtes préparées (PDO)
- Validation des données formulaire
- Protection contre injections SQL

3. Protection des accès

- Fonctions requireLogin()

- requireMinRole()
- Vérification des droits avant chaque action sensible

VII. Documentation du déploiement

1. Déploiement sur InfinityFree

Étapes réalisées :

1. Création du dépôt GitHub public
2. Organisation des branches
3. Création du compte InfinityFree
4. Création d'une base de données distante
5. Import du fichier database.sql
6. Modification du fichier config/database.php
7. Upload des fichiers via File Manager
8. Correction des chemins relatifs (__DIR__)
9. Tests en production
10. Mise en ligne finale

2. Difficultés rencontrées

- Problèmes de chemins relatifs (../)
- Restriction open_basedir
- Erreurs de connexion base de données
- Adaptation des variables d'environnement

Ces problèmes ont été résolus en :

- Ajustant les chemins absolus
- Vérifiant la configuration PDO
- Corrigeant les paramètres de connexion

VIII. Conclusion technique

L’application respecte :

- Une architecture MVC claire
- Une séparation logique métier / présentation
- Une gestion propre des rôles
- Une base relationnelle cohérente
- Un déploiement fonctionnel en production

Le projet est maintenable, évolutif et conforme aux attentes du sujet.

ANNEXES.

Modèle conceptuel de données :

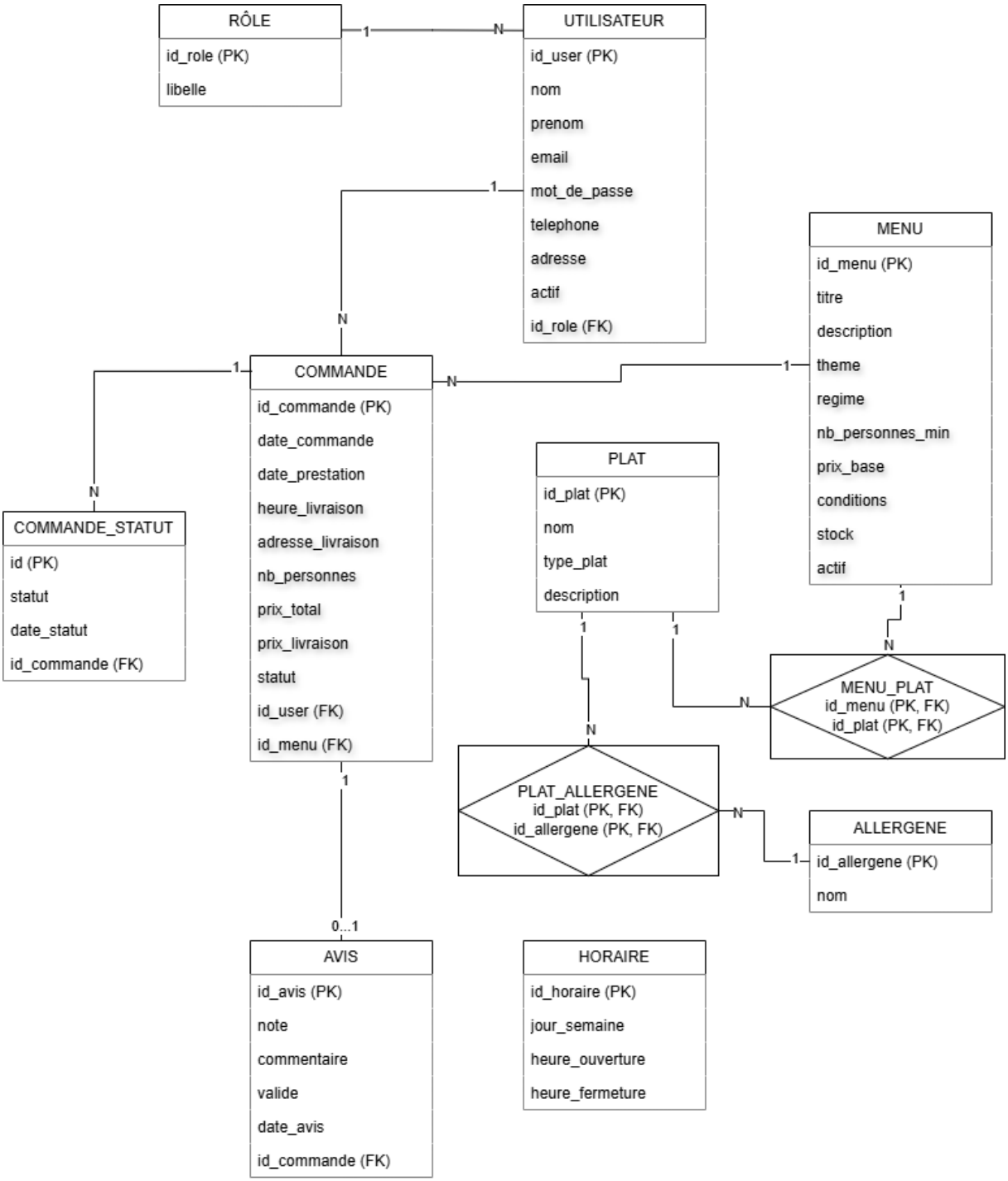


Diagramme de classe :

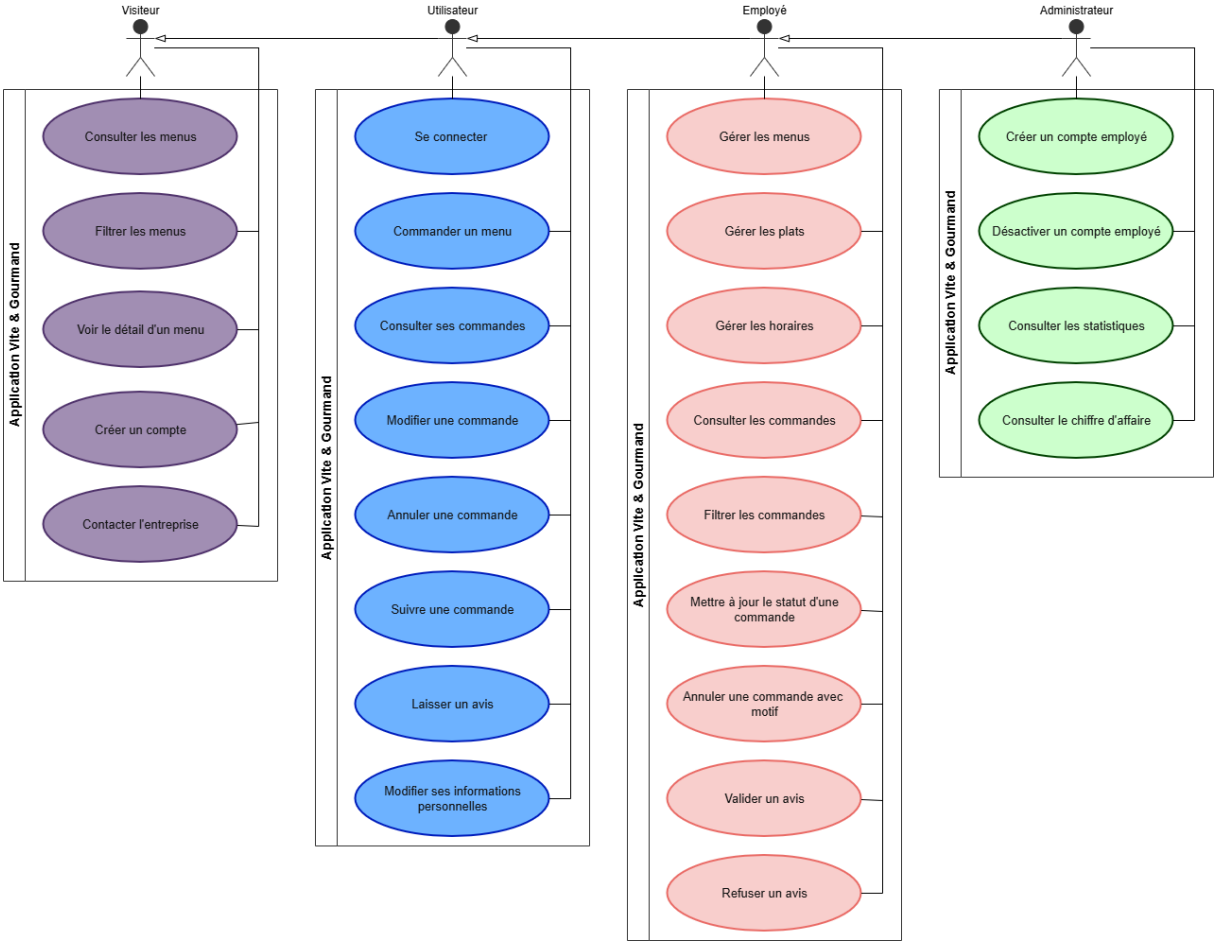
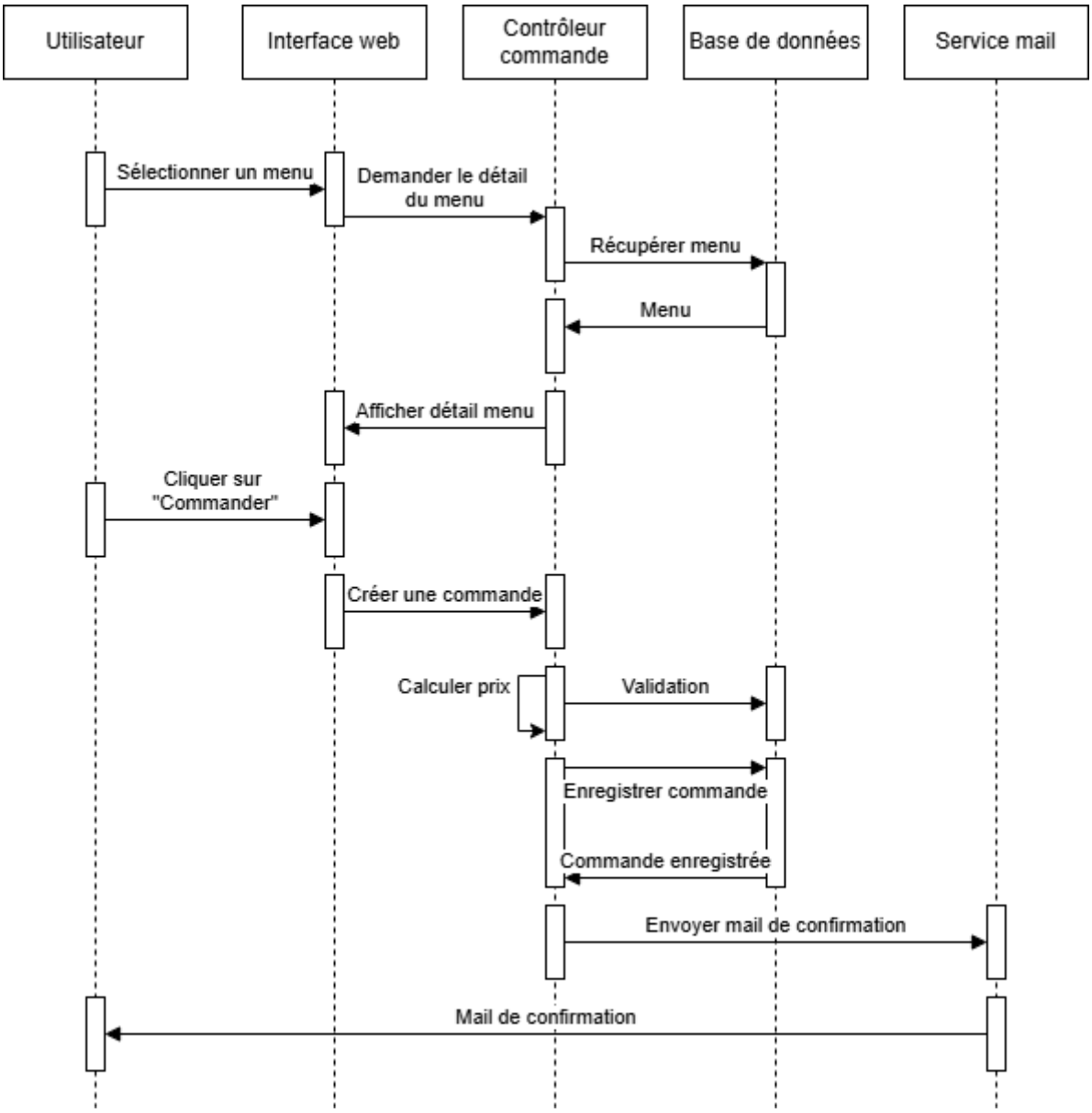


Diagramme de séquence :


Utilisateur
Interface web
Contrôleur commande
Base de données
Service mail
Sélectionner un menu
Demander le détail du menu
Récupérer menu
Menu
Afficher détail menu
Cliquer sur "Commander"
Créer une commande
Calculer prix
Validation
Enregistrer commande
Commande enregistrée
Envoyer mail de confirmation
Mail de confirmation