

VITE & GOURMAND

Application Web de Traiteur Événementiel

Présenté par : Yves Mukuna

Formation : [Votre nom de formation]

Date : [Date de soutenance]

<https://vite-gourmand.fr>

SOMMAIRE

1. Contexte & Objectifs
2. Analyse des Besoins
3. Modélisation des Données (MCD)
4. Architecture Technique
5. Fonctionnalités Développées
6. Sécurité & RGPD
7. Tests Automatisés
8. Déploiement en Production
9. Démonstration Live
10. Bilan & Perspectives

CONTEXTE DU PROJET



Problématique

Un traiteur événementiel souhaite digitaliser son activité



Objectifs

- Permettre aux clients de commander des menus en ligne
- Faciliter la gestion des commandes pour l'équipe
- Offrir un outil d'administration complet
- Assurer la sécurité et conformité RGPD



Durée : 12 jours

ANALYSE DES BESOINS

ANALYSE DES BESOINS

Acteurs Identifiés

Visiteurs

Consultation menus, Inscription

Clients Utilisateurs

Commandes, Suivi, Avis

Employés

Gestion commandes, Validation avis

Administrateurs

Gestion complète (menus, plats, users)



Total 29 cas d'usage

MODÈLE CONCEPTUEL DE DONNÉES



13 Tables Principales

Users, Menus, Plats, Allergènes

Commandes, Avis, Contacts

Tables associatives (menu_plat, plat_allergene)

Relations Clés :

Users (1) —< Commandes (N)

Users (1) —< Avis (N)

Menus (1) —< Commandes (N)

Menus (N) >—< Plats (N)

Plats (N) >—< Allergènes (N)



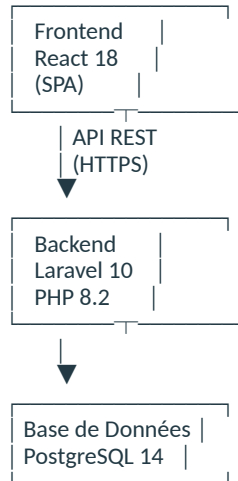
Normalisation 3NF



Intégrité référentielle

ARCHITECTURE TECHNIQUE

Architecture 3-Tiers



Serveur : Nginx (VPS OVH Ubuntu 22.04)

JUSTIFICATION DES CHOIX TECHNIQUES



React 18

→ Composants réutilisables, écosystème riche, expérience SPA fluide



Laravel 10

→ Framework PHP robuste, Eloquent ORM, sécurité intégrée, documentation complète



PostgreSQL 14

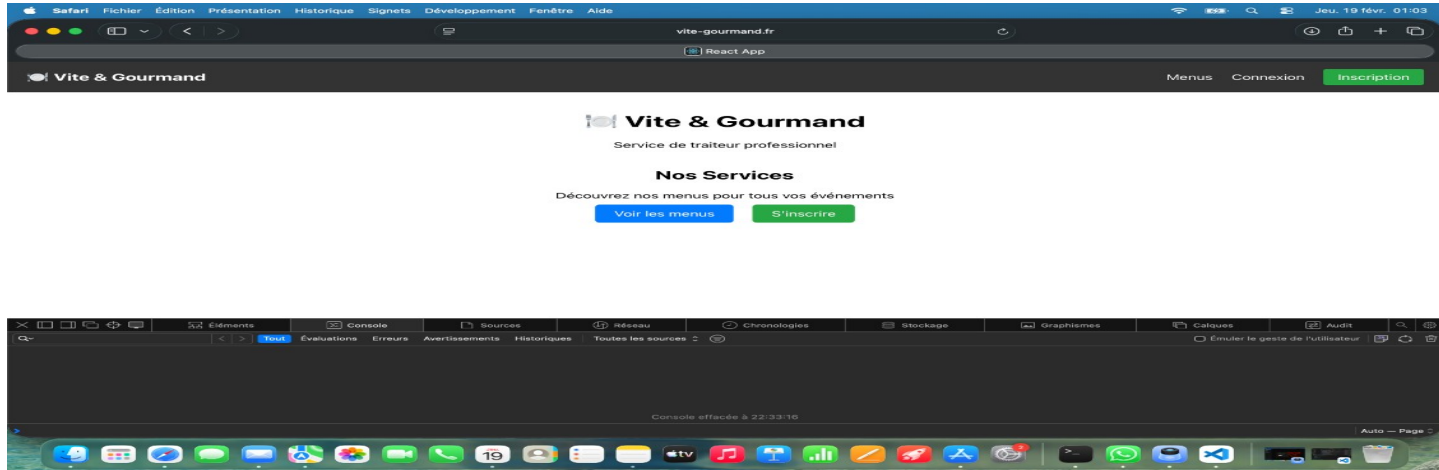
→ SGBD fiable, respect standards SQL, intégrité des données



Laravel Sanctum

→ Authentification API simple et sécurisée

FONCTIONNALITÉS CÔTÉ CLIENT

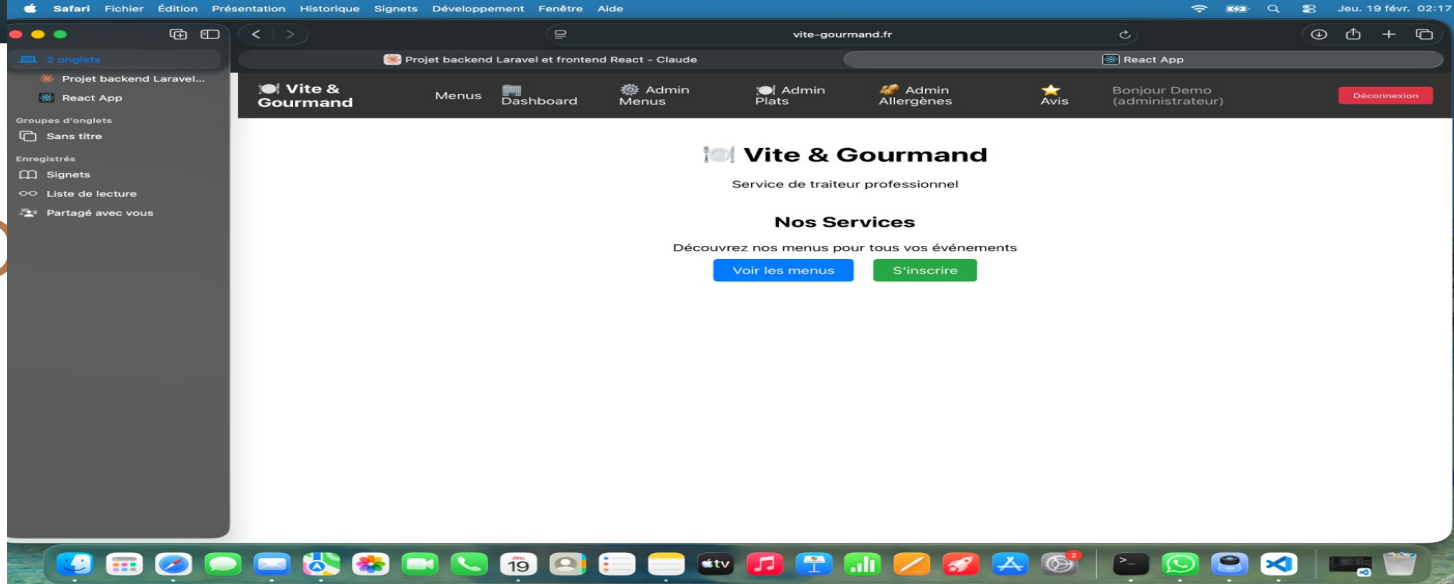


✓ Inscription / Connexion sécurisée

✓ Consultation catalogue menus

✓

FO



Employés

Gestion des commandes (statuts)

- Validation avis clients
- Création/modification menus

Administrateurs

- CRUD Menus / Plats / Allergènes
- Gestion utilisateurs

MESURES DE SÉCURITÉ



Authentication

- Tokens Sanctum (API)
- Mots de passe hashés (bcrypt)
- Complexité : 10+ caractères, majuscules, chiffres, symboles



Protection des Données

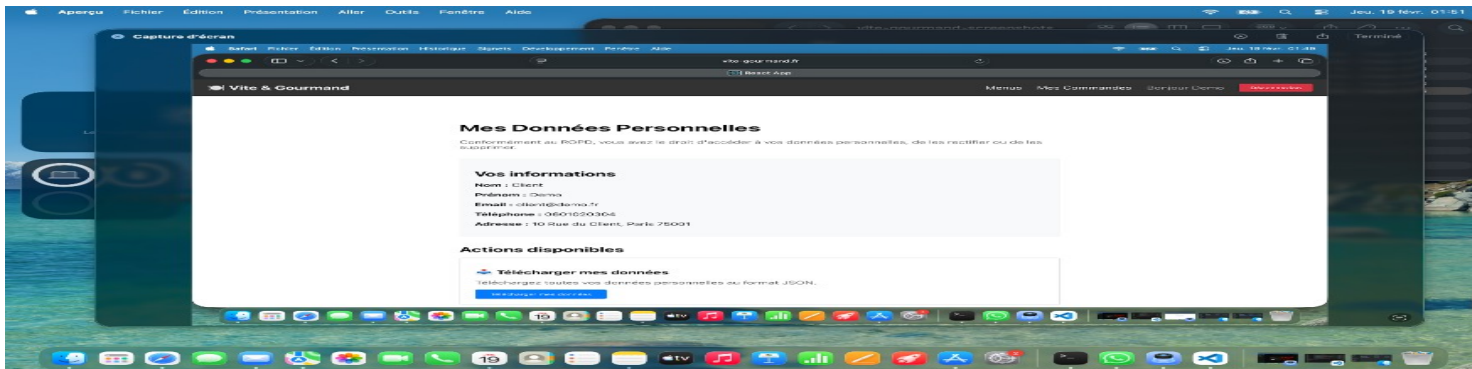
- HTTPS obligatoire (SSL)
- Headers HTTP sécurisés (CSP, HSTS, X-Frame-Options)
- CORS configuré strictement



Contrôle d'Accès

- Politiques (autorisations par rôle)
- FormRequests (validation entrées)
- Rate Limiting (5 login/min, 60 API/min)

CONFORMITÉ RGPD



Droits Implémentés



Transparence

Politique de confidentialité + Consentement obligatoire

TESTS AUTOMATISÉS



15 Tests Implémentés



Authentification (9 tests)

- Inscription (validation RGPD)
- Connexion (succès/échec)
- Déconnexion



Menus CRUD (3 tests)

- Liste publique
- Création admin
- Autorisations



RGPD (6 tests)

DÉPLOIEMENT EN PRODUCTION



Infrastructure

- VPS OVH (Ubuntu 22.04)
- Domaine : vite-gourmand.fr
- IP : 37.59.124.193



SSL/HTTPS

- certificat Let's Encrypt
- Renouvellement automatique
- HTTPS obligatoire



Stack Serveur

- nginx (proxy inverse)
- PHP-FPM 8.2

MÉTRIQUES DU PROJET

Quantitatif



12 jours



~8000 lignes de code



13 tables de base de données



46 routes API



15 tests automatisés



15+ pages/composants React

Qualitatif

DÉMONSTRATION EN DIRECT

Scénario :

- 1 Page d'accueil & Navigation
- 2 Consultation des menus
- 3 Connexion Client (commande)
- 4 Mes Commandes
- 5 Connexion Employé (gestion)
- 6 Connexion Admin (administration)
- 7 Export données RGPD



<https://vite-gourmand.fr>

DIFFICULTÉS RENCONTRÉES

Accès SSH au VPS

Problème : Mot de passe non accepté

Solution : Mode rescue + création utilisateur

Configuration API Frontend

Problème : URL hardcodée (localhost)

Solution : Variables d'environnement (.env.production)

Routing Nginx

Problème : Routes API 404

Solution : Configuration location blocks adaptée



PERSPECTIVES D'ÉVOLUTION

Fonctionnelles



Paiement en ligne (Stripe)



Notifications email



Application mobile (React Native)



Dashboard statistiques avancé

Techniques



Cache Redis



Monitoring (logs, métriques)



CI/CD automatisé



Internationalisation (i18n)

CONCLUSION

Objectifs Atteints

- ✓ Application complète et fonctionnelle
- ✓ Stack technique moderne
- ✓ Sécurité & RGPD respectés
- ✓ Déployée en production
- ✓ Testée et documentée

Bilan Personnel