

Cahier des charges techniques

Thomas Kany, Felipe Alvariza

15 octobre 2025

Sommaire



Présentation du projet



Architecture technique



Base de données



Environnement applicatif



Cadriels et bibliothèques



Fonctionnalités clés



Contraintes et évolutions

Présentation du projet

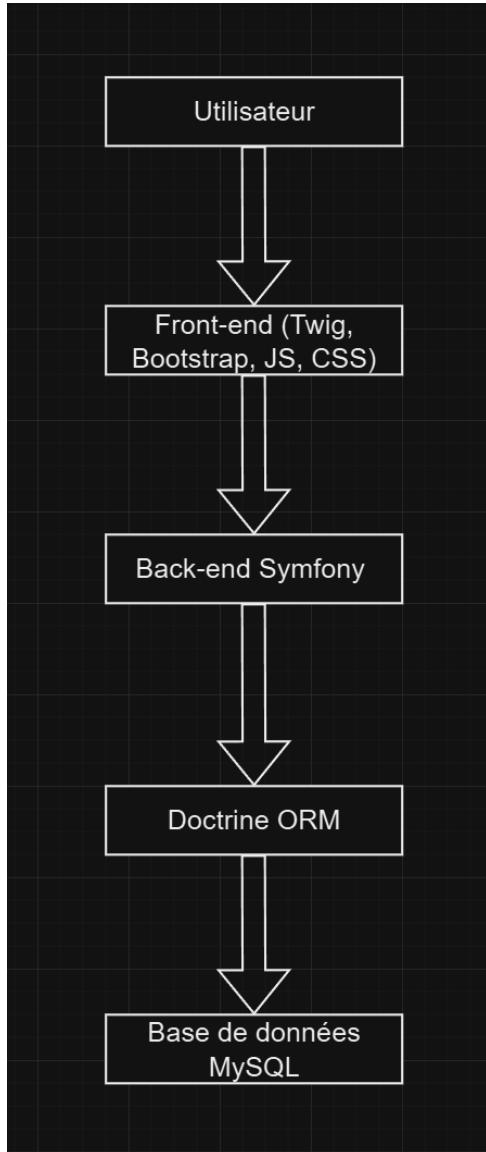
- Software Architecture Évolutions (SAÉ) : entreprise de développement logiciel
- Rôle : développeur junior en binôme
- Client : réseau Cocagne, application pour les adhérents
- Méthode : démarche itérative et incrémentale avec intégration continue
- Enjeux : qualité logicielle, ergonomie, amélioration continue
- Contraintes : respect des délais, sérieux, règles internes
- Technologies : frameworks, API complexes, base de données, serveur



Qui sommes nous

- Thomas kany, Felipe Alvariza
(20 ans)
- Développeurs juniors chez
Software Architecture
Évolutions (SAÉ)
- Connaissances de plusieurs
langages de programmation
(Java, C, Flutter, Dart, CSS,
PHP...)





Architecture technique

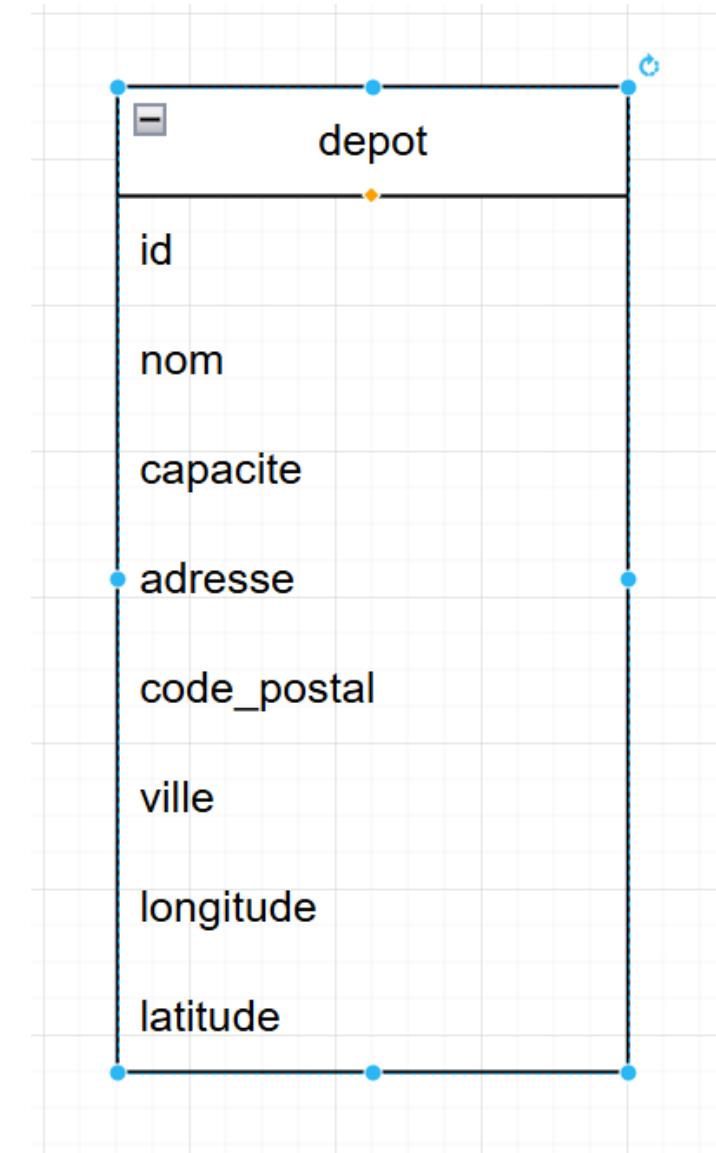
- Frontend : Twig et CSS (ainsi que Bootstrap)
- Backend : Symfony (PHP 8.2)
- Base de donnée : MySQL via Doctrine ORM
- Environnement d'exécution : conteneurs Docker (PHP, Nginx, MySQL)
- Communication : HTTP entre client et Nginx

Base de données



Modèle de données

- Système de gestion : MySQL
- Mapping objet-relationnel : Doctrine ORM
- Structure actuelle : 3 tables (depot, doctrine_migration_versions, messenger_messages)

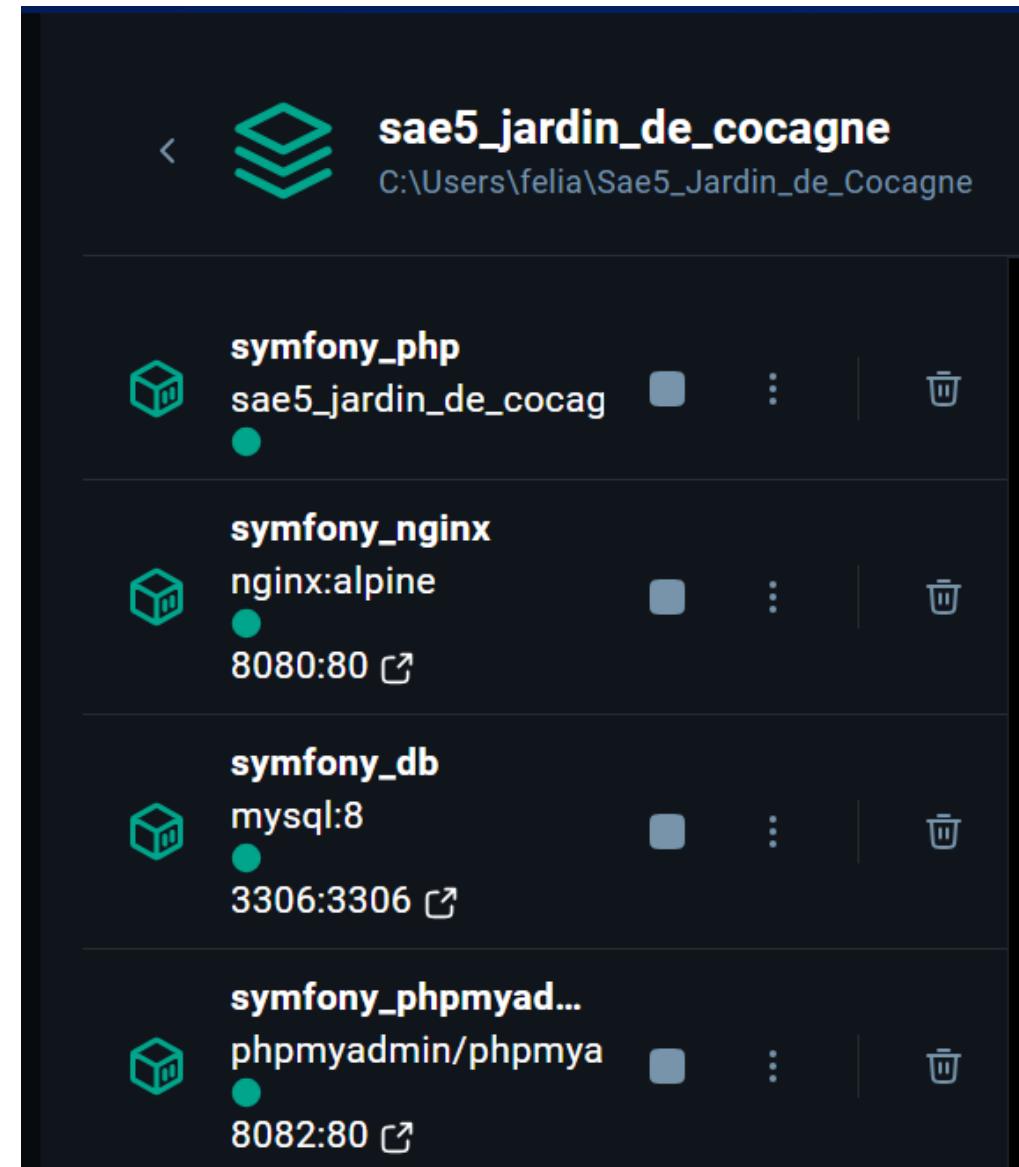


Environnement applatif



Environnement de déploiement

- Serveur web : Nginx
- Serveur applicatif : PHP 8 (version 8.2 via Docker)
- Versionnement : Github
- Application : symfony_php
- Base MySQL : symfony_db



Cadriels et bibliothèques

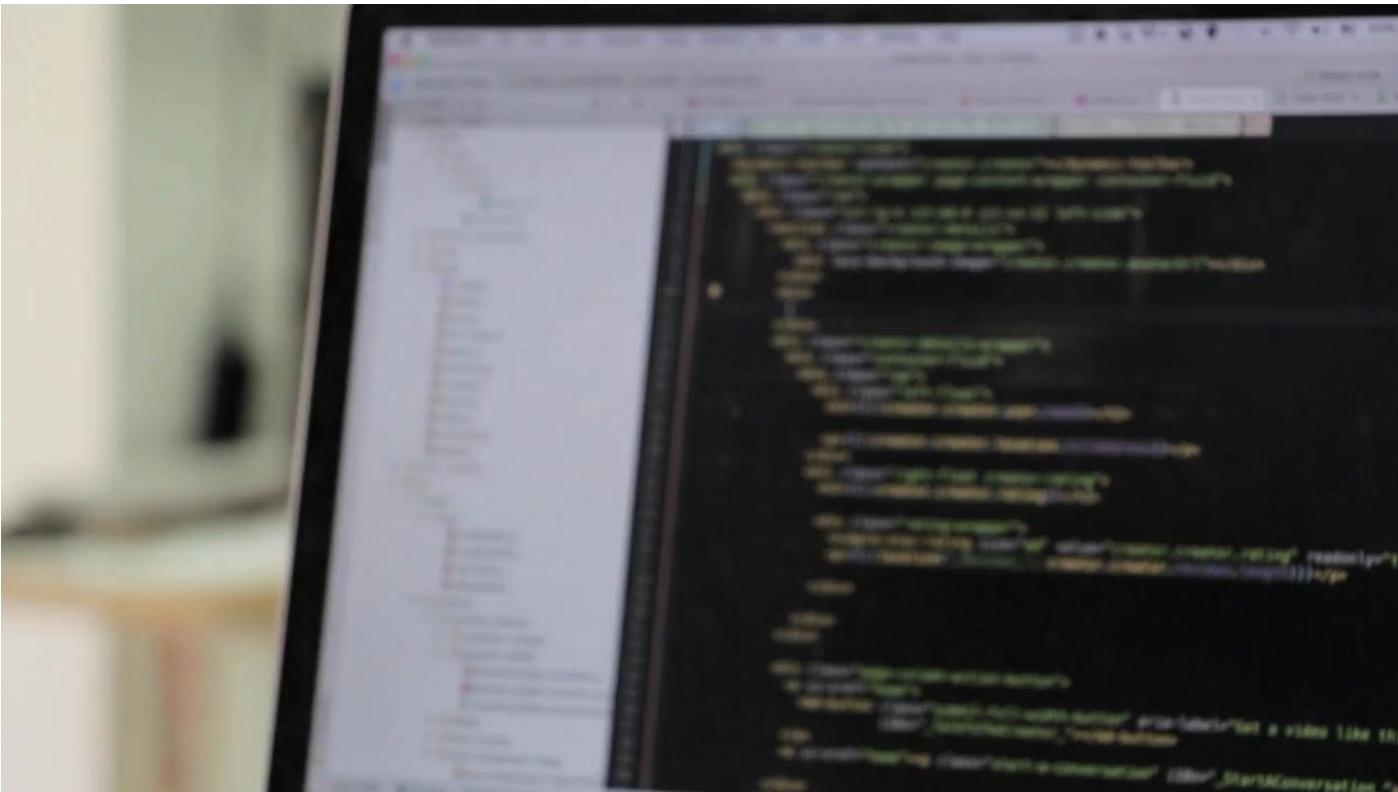


Outils et technologies

- **Symfony (PHP)** – Framework principal suivant l'architecture MVC
- **Twig** – Génération des pages dynamiques
- **Bootstrap** – Style
- **Doctrine ORM** – Accès et gestion des données
- **Docker** – Environnement de développement
- **Git** – Suivi et collaboration au code



Fonctionnalités principales



- Inscription & abonnement adhérents
- Gestion des dépôts (CRUD)
- Carte interactive des dépôts
- Itinéraires de livraison
- Calendrier des tournées
- Répartition des légumes dans les paniers
- Suivi & modification des tournées

Contraintes

- Déploiement dans Docker pour portabilité
- Sécurité des données : authentification et droits utilisateurs
- Compatibilité multi-navigateurs et responsive
- Respect des délais du projet
- Documentation claire pour les utilisateurs et mainteneurs
- Accessibilité et ergonomie de l'interface utilisateur



Possibles évolutions

- Optimisation des performances front-end et back-end
- Mise en place d'un cache pour accélérer les requêtes fréquentes
- Développement d'une application mobile
- Ajout de modules pour la gestion d'abonnements ou de stocks avancés
- Internationalisation (multi-langues)

