# Contexte du projet

Application Odoo 16 déployée en local via Docker pour gérer l'enrôlement et le suivi des établissements pharmaceutiques. Deux services Docker : PostgreSQL (base 'pharma\_registry\_db') et Odoo (image reconstruite avec les modules personnalisés.

# Fichiers principaux créés ou modifiés

* docker-compose.yml – orchestre les services 'db' (PostgreSQL 14) et 'odoo' (image personnalisée exposée sur le port 8069).
* Dockerfile – part de odoo:16, installe openpyxl/xlsxwriter pour les imports/exports Excel, copie les modules 'pharma\_enrollment' et 'pharma\_registry', applique odoo.conf.
* odoo.conf – configure l'instance : base 'pharma\_registry\_db', chemin des addons incluant /mnt/extra-addons, paramètres admin/password.
* pharma\_registry/\_\_manifest\_\_.py – déclare le module, ses dépendances, la liste des vues et les assets backend (pharma\_forms.scss).
* pharma\_registry/views/officine\_views.xml – refonte du formulaire Officine : notebook avec onglets profil/coordonnées/équipe/indicateurs, cartes en grille, commentaires XML pour guider les futures évolutions.
* pharma\_registry/views/assets.xml – enregistre l'asset backend 'pharma\_registry\_backend\_assets' qui pointe vers le fichier SCSS.
* pharma\_registry/static/src/scss/pharma\_forms.scss – stylesheet commentée : variables de couleur, style des onglets, grille responsive, cartes vertes.
* docs/guide\_projet\_pharma\_registry.docx – ce guide résumant l’architecture et le workflow.

# Workflow de développement

1. Modifier le code dans pharma\_registry (vues, wizards, styles).

2. Recréer l'image et le volume /mnt/extra-addons :

* docker-compose stop odoo
* docker rm <id\_conteneur\_odoo>
* docker volume rm <id\_volume\_extra\_addons> # volume anonyme monté sur /mnt/extra-addons
* docker-compose build --no-cache odoo
* docker-compose up -d odoo

3. Mettre à jour le module : docker-compose exec odoo odoo -c /etc/odoo/odoo.conf -d pharma\_registry\_db -u pharma\_registry --stop-after-init

4. Rafraîchir le navigateur en vidant le cache (Ctrl+F5) ou utiliser l'URL ?debug=assets pour invalider les bundles.

# Personnalisations livrées

* Vues formulaire restructurées pour tous les types d'établissement (Officine, Dépôt, Grossiste, Agence, Fabrication) avec onglets thématiques et cartes.
* Menus 'Registre Pharmaceutique' → établissements, imports, référentiels géographiques.
* Feuille de style dédiée aux formulaires : onglet actif vert, cartes vertes, bandeau de section.
* Assistants d'import CSV/XLSX pour établissements et référentiels (wizards/import\_\*).

# Commandes utiles

- Lancer/relancer les services : docker-compose up -d

- Voir les logs : docker-compose logs -f odoo

- Ouvrir un shell dans Odoo : docker-compose exec odoo bash

- Arrêter l'environnement : docker-compose down

# Continuer le projet

* Ajouter des champs : modifier les modèles Python, mettre à jour les vues XML, exécuter -u pharma\_registry.
* Nouvelles vues/onglets : réutiliser les classes CSS existantes (pharma-card, pharma-grid).
* Exports / imports supplémentaires : s'inspirer des wizards existants, openpyxl pour lire, XlsxWriter pour écrire.
* Rapports : créer de nouvelles actions et intégrer des vues pivot/graph/board.
* Documenter chaque évolution (README, doc supplémentaire) pour garder la trace des choix techniques.