

# 003 - domaines ( macro-périmètre )

## Projet : Refonte du SI DIGICHEESE – Gestion des cadeaux fidélité

### 3.1. Objectif de cette partie

Cette section décrit l'organisation fonctionnelle de l'application en la découplant en grands domaines, conformément au cahier des charges.

L'objectif est de structurer clairement :

- ce que fait le système,
- pour qui (rôles utilisateurs),
- et dans quel périmètre.

L'application est une application interne (Intranet) destinée aux salariés, et elle est centrée sur trois domaines principaux liés aux rôles utilisateurs :

- **Administrateur (rôle Admin)**
- **Gestion des colis (Op-Colis)**
- **Gestion des stocks (Op-Stock)**

Chaque domaine correspond à un ensemble cohérent de fonctionnalités, et sert de base pour construire ensuite les **Epics** et les **User Stories** dans Jira.

---

### 3.2. Domaine 1 — Administrateur (Admin)

#### 3.2.1. Finalité du domaine Administrateur

Le domaine Administrateur permet de paramétriser, sécuriser et maintenir le système.

Il est réservé aux utilisateurs ayant le rôle **Admin**.

Ce domaine sert à gérer :

- les utilisateurs du système,
- les paramètres métiers essentiels au fonctionnement de l'application (**Communes, Adresses, Poids, Poids-Vignettes, Objets, Conditionnements, Rôles**),
- certaines sorties administratives (**Impressions**).

Ce domaine est **obligatoire**, car il garantit que :

- les données de référence sont correctes,

- les autres modules (Colis & Stock) peuvent fonctionner avec des informations cohérentes.

### 3.2.2. Fonctionnalités détaillées du domaine Administration

#### Gestion des utilisateurs

L'administrateur doit pouvoir effectuer les opérations principales sur les comptes utilisateurs :

- création d'un utilisateur,
- suppression d'un utilisateur via son id,
- modification d'un utilisateur via son id,
- récupération d'un utilisateur via son id,
- récupération de la liste des utilisateurs,
- gestion du rôle et des droits (**Admin, Op-Colis, Op-Stock**).

L'objectif est de contrôler l'accès et de garantir que seuls les profils autorisés utilisent chaque espace.

#### Gestion des paramètres métiers (CRUD)

L'administrateur doit pouvoir gérer les paramètres métiers présents en base de données via des opérations CRUD (Créer, Lire, Mettre à jour, Supprimer) sur les tables suivantes :

- **t\_communes** : référentiel de communes (utilisé pour les adresses, clients ou livraisons),
- **t\_objet** : référentiel des objets/articles disponibles,
- **t\_conditionnement** : formats de conditionnement utilisés pour la préparation des colis,
- **t\_poids** : référentiel des poids (utile pour relier colis et poids),
- **t\_poidsV (Poids-Vignettes)** : référentiel complémentaire lié aux vignettes et au conditionnement.

Ces paramètres sont des données structurantes : sans eux, la gestion des colis et des stocks devient incohérente.

#### Génération d'impressions

Le cahier des charges prévoit la possibilité de générer une impression papier depuis l'espace Administrateur.

Même si l'implémentation n'est pas faite dans le projet Python, cette fonctionnalité doit :

- être identifiée dans l'organisation fonctionnelle,
- être prise en compte dans la planification Jira (au minimum sous forme de **Story** ou **tâche**).

## 3.3. Domaine 2 — Gestion des colis (Opérateur Colis / Op-Colis)

### 3.3.1. Finalité du domaine Gestion des colis

Le domaine Gestion des colis représente le cœur opérationnel du SI côté « fidélité et cadeaux ».

Il est destiné au rôle **Op-Colis** et regroupe les fonctionnalités liées :

- aux clients,
- aux commandes,
- à la préparation et au suivi des colis,
- aux statistiques et à la communication client.

**Note** : les fonctionnalités détaillées (clients, commandes, suivi, statistiques, mailing, impressions) seront décrites et déclinées en Epics et User Stories dans la suite de la documentation.

---

### 3.4. Domaine 3 — Gestion des stocks (Opérateur Stock / Op-Stock)

#### 3.4.1. Finalité du domaine Gestion des stocks

Le domaine Gestion des stocks vise à assurer le suivi des stocks physiques et la cohérence entre les données du système et la réalité terrain.

Il est destiné au rôle **Op-Stock**.

**Note** : les fonctionnalités détaillées (consultation, mise à jour, mise à jour annuelle, impressions) seront décrites et déclinées en Epics et User Stories dans la suite de la documentation.

---

### 3.5. Synthèse de l'organisation fonctionnelle

L'application est structurée en trois domaines alignés sur les rôles métiers :

- **Administration** : gestion des utilisateurs + paramétrage des référentiels métiers + impressions
- **Gestion des colis** : gestion opérationnelle « fidélité & cadeaux » (clients, commandes, colis, statistiques, mailing, impressions)
- **Gestion des stocks** : suivi et mise à jour des stocks + impressions

Cette organisation sert directement de base à la gestion de projet dans Jira :

- chaque domaine deviendra un ou plusieurs **Epics**,
- chaque fonctionnalité deviendra une **User Story**,
- avec des **critères d'acceptation**, des **priorités** et des **dépendances**.