

# CAHIER DES CHARGES TECHNIQUE

*Plateforme de gestion commerciales des Jardins  
Cocagne*

**Réalisé par:**

NGUYEN Quang-Liêm  
RALLI Clément

**Formation:**

BUT Informatique 3

**Établissement:**

IUT de Saint-Dié des Vosges – Université de Lorraine

**Année:** 2025-2026





# Notre équipe

Nous sommes une équipe de deux passionnés, engagés à développer une solution technique adaptée aux enjeux sociaux et logistiques du Réseau Cocagne.



NGUYEN Quang-Liêm  
Co-fondateur

Contact: 07.XX.XX.XX.XX  
Courriel: [quang-liem.nguyen7](mailto:quang-liem.nguyen7)

Spécialisations:

- Back-end
- Design
- Base de données



RALLI Clément  
Co-fondateur

Contact: 06.XX.XX.XX.XX  
Courriel: [clement.ralli9](mailto:clement.ralli9)

Spécialisations:

- Front-end
- Design
- Pilotage de Projet



# Les Jardins de Cocagne

1. Réseau de plus de 100 jardins de Cocagne en France
2. Associations d'insertion sociale via le maraîchage biologique
3. Vente de paniers de légumes bio sous forme d'abonnement à des adhérents-consommateurs
4. Fonctionnement associatif: Adhésion obligatoire, gestion locale par chaque Jardin





# Pourquoi changer d'outil de gestion ?



Depuis plus de 20 ans, les Jardins de Cocagne utilisent un logiciel local basé sur Microsoft Access.

Si cet outil a longtemps rendu service, il ne correspond plus aux exigences actuelles du réseau



# Pourquoi changer d'outil de gestion ?



## Un outil existant devenu obsolète

Il était initialement conçu pour quelques Jardins et une utilisation locale, mais aujourd'hui :

- Il n'est pas accessible en ligne, ce qui empêche le travail collaboratif à distance.
- Il nécessite des installations manuelles sur chaque poste, rendant les mises à jour lourdes et complexes.
- L'interface est vieillissante et peu intuitive, surtout pour des utilisateurs peu à l'aise avec l'informatique.
- Les possibilités de personnalisation sont limitées, alors que chaque Jardin fonctionne différemment.



# Pourquoi changer d'outil de gestion ?



## Une absence d'automatisation et de mutualisation

Les Jardins de Cocagne souhaitent aujourd'hui un outil :

- Connecté et mutualisé, utilisable par tous les Jardins via une plateforme web unique.
- Automatisé, pour réduire les tâches administratives répétitives (calcul des abonnements, gestion des factures, suivi des livraisons...).
- Capable de gérer plusieurs modules activables à la carte selon les besoins de chaque Jardin (gestion des adhérents, paniers, livraisons, facturation...).



# Pourquoi changer d'outil de gestion ?



## Des besoins de modernisation et d'accessibilité

Le contexte a évolué:

- Les Jardins doivent gérer plus d'adhérents, de produits et de tournées qu'il y a 20 ans.
- Les utilisateurs du logiciel sont souvent salariés en insertion, peu familiarisés avec l'informatique : l'outil doit donc être simple, visuel et didactique.
- Le réseau national souhaite un outil moderne et évolutif, en open source, pour favoriser la contribution de développeurs bénévoles.
- L'hébergement en mode SaaS (en ligne) permettra de centraliser les données et de simplifier la maintenance.



# 4 Objectifs principaux



Gérer efficacement  
les adhérents,  
abonnements,  
livraisons et dépôts



Simplifier le travail  
administratif des  
équipes



Offrir une interface  
autonome pour les  
adhérents



Mutualiser les outils  
entre les Jardins du  
réseau Cocagne





# Objectif 1 : Gérer efficacement les adhérents, abonnements, livraisons et dépôts



## Centraliser la gestion des données

- Regrouper toutes les fonctions dans un seul outil : adhérents, abonnements, produits, dépôts et livraisons
- Faciliter le suivi et la mise à jour des données
- Réduire les erreurs liées aux saisies manuelles
- Offrir une vision claire de l'activité du Jardin



# Objectif 2: Simplifier le travail administratif des équipes



## Alléger la charge administrative

- Automatiser les tâches répétitives (renouvellement d'adhésions, factures, feuilles de route)
- Réduire le temps de saisie et les risques d'erreur
- Permettre aux salariés d'insertion de se concentrer sur le cœur de leur activité
- Offrir une interface simple et intuitive adaptée à tous les profils



# Objectif 3: Offrir une interface autonome pour les adhérents

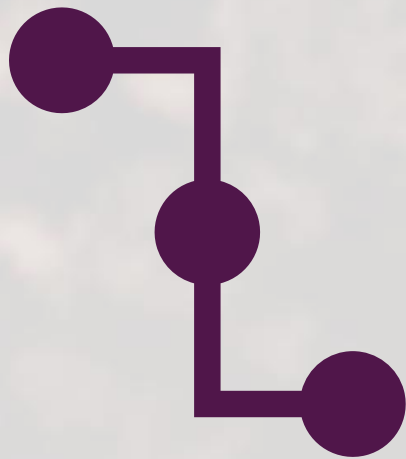


## Donner plus d'autonomie aux adhérents

- Espace personnel pour suivre ses abonnements et factures
- Choix du point de dépôt et gestion des absences
- Consultation du calendrier de livraisons
- Communication directe avec le Jardin



# Objectif 4: Mutualiser les outils entre les Jardins du réseau Cocagne



## Mutualiser et harmoniser les outils

- Créer une base commune utilisée par tous les Jardins
- Permettre à chaque structure de personnaliser ses paramètres
- Partager les évolutions et les bonnes pratiques au sein du réseau
- Favoriser la collaboration et la montée en compétence collective



# Une plateforme web pour la gestion des Jardins de Cocagne

Le projet consiste à développer une application web mutualisée destinée aux Jardins de Cocagne, structures d'insertion en maraîchage biologique

Elle permettra de:

- Gérer les adhérents et leurs adhésions
- Suivre les abonnements et paniers livrés
- Organiser les tournées de livraison et points de dépôt
- Offrir aux adhérents un espace personnel en ligne

L'application sera hébergée en ligne (mode SaaS), open source, et personnalisable par chaque Jardin du réseau.



Adhérent



Application Web



Équipe du Jardin



# Architecture générale

Niveau	Choix Technique	Justification
Interface (Front-end)	React.js / Vue.js	UX Didactique pour les utilisateurs peu qualifiés. Application Responsive.
Logique (Back-end)	Node.js (Express) / Django	Simplicité du langage pour l'évolution par des développeurs bénévoles.
Données (DB)	PostgreSQL / MySQL	Open Source et robustesse pour l'architecture Multi-Sociétés.
Communication (API)	REST (JSON) + JWT	Performances (< 2s) et Sécurité (HTTPS, Chiffrement bcrypt).
Hébergement	Serveur mutualisé / Cloud	Facilite l'administration et les mises à jour de l'outil mutualisé centralisé.



# Modules fonctionnels: Le périmètre et les enjeux

Le système est conçu autour de 8 modules principaux qui structurent l'intégralité du cycle de vie commercial des Jardins.





# Modules fonctionnels: Le périmètre et les enjeux



Module	Description	Interface
Structures	Paramétrage des informations de chaque Jardin	Back-office
Saisons	Gestion des saisons agricoles et fermetures	Back-office
Dépôts	Gestion des points de dépôt (catégories, capacités, horaires)	Front/Back
Tournées de livraison	Planification des tournées et ordres de livraison	Back-office
Adhérents	Gestion des fiches, adhésions et abonnements	Front/Back
Produit	Catalogue de produits et paniers	Back-office
Abonnements	Liaison produit-fréquence-dépôt	Front/Back
Calendrier	Gestion du calendrier et jours fériés	Back-office





# Modèle de données cible

Le modèle s'articule autour des 8 entités principales listées dans le CDC. Il est conçu pour gérer deux complexités majeures : la **mutualisation** et la **logistique des abonnements**.

## Domaine 1

*Gestion des Client et Mutualisation*

### Entités clés:

- Structure
- Adhérent
- Adhésion



## Domaine 2

*Vente et offre*

### Entités clés:

- Saison
- Produit
- Abonnement



## Domaine 3

*Logistique*

### Entités clés:

- Dépôt
- Tournée
- Abonnement





# Modèle de données cible

Entité	Clé (ID)	Description	Relations Clés (Règles Métier)
Structure	ID_Structure	Représente l'unité administrative et légale de chaque Jardin de Cocagne.	C'est la clé de partitionnement pour la gestion multi-sociétés.
Saison	ID_Saison	Définit la période de validité des abonnements et les contraintes temporelles du Jardin.	Liée à 1 Structure. Contient les dates de début/fin, les Périodes de Fermeture et les Jours Fériés.
Adhérent	ID_Adhérent	Fiche client (personne physique ou morale) qui achète des abonnements et/ou soutient l'association.	Appartient à 1 Structure. Peut avoir N Abonnements et N Adhésions (au sens associatif).
Produit	ID_Produit	Catalogue des articles vendus (légumes, paniers types, lots, services, etc.) par la structure.	Paramétré par 1 Structure
Abonnement	ID_Abonnement	Le cœur du système : représente le contrat commercial pour un produit livré selon une fréquence donnée.	Basé sur 1 Produit. Lié à 1 Adhérent et rattaché à 1 Dépôt.
Dépôt	ID_Dépôt	Le Point de Dépôt physique où l'adhérent vient retirer son panier/produit.	Lié à 1 Structure. Peut faire partie de N Tournées de livraison.
Tournée	ID_Tournée	Définit la séquence de livraison logistique pour un jour donné.	Contient N Dépôts selon un ordre de passage défini (relation ordonnée).
Adhésion (Assoc)	ID_Ahsésion	Documente l'adhésion annuelle de soutien à l'association (distincte de l'achat d'abonnement).	Relie 1 Adhérent à 1 Type d'Adhésion sur une période donnée.



# Interface Utilisateur (UI/UX) : Simplicité & Accessibilité

## Objectif 1 : L'Intuitivité pour le Back-Office (Focus)

L'interface doit être **Didactique**, **Conviviale** et **Extrêmement Intuitive** pour compenser le **niveau de maîtrise informatique variable** des utilisateurs et le fort turn-over des salariés en insertion.

## Objectif 2 : Accès Universel (Responsive Design)

L'application doit être consultable et utilisable de manière ergonomique sur tous supports : **PC**, **Tablette**, et **Mobile**.

## Objectif 3 : Conformité et Accessibilité

Garantir une **Accessibilité de niveau AA** minimum. Respecter le **RGPD** et les normes de sécurité (HTTPS).



# Interface Utilisateur (UI/UX) : Simplicité & Accessibilité

## Back-office (Gestion interne)

**Public Cible** : Salariés permanents et en insertion (fort turn-over et niveau variable).

**Rôles** : Tableau de bord de gestion, gestion des adhésions, produits, livraisons, et exports.

**Exigence UX** : Interface **Intuitive, Conviviale et Didactique** (nécessité de minimiser les erreurs de manipulation et de réduire les temps de formation).



# Interface Utilisateur (UI/UX) : Simplicité & Accessibilité

## Back-office (Gestion interne)

Public Cible : Adhérents-consommateurs.

Rôles : Consultation et gestion de l'abonnement, accès aux factures, consultation du **calendrier des livraisons** et des **points de dépôt**.

Exigence UX : **Autonomie** maximale du client, réduisant la charge administrative du Jardin.



# Contraintes techniques



## Contraintes d'environnement

Ces contraintes découlent directement des besoins du Réseau.

### Technologie/Développement

L'outil doit être Open Source et développé avec un langage simple (ex: PHP/Python).

→ **Faciliter l'évolution** et la maintenance par des développeurs bénévoles ou locaux.

### Architecture

Doit supporter l'architecture Multi-Utilisateurs et Multi-Sociétés avec un hébergement centralisé.

→ Assurer l'autonomie et la propriété des données pour chaque Jardin, tout en mutualisant le code.



# Contraintes techniques

## Exigences de qualité (ENF)

Ces exigences définissent la robustesse du système.

### Performance

Le temps de chargement des pages ne doit idéalement pas dépasser **2 secondes**.

→ Assurer une bonne expérience utilisateur malgré l'architecture centralisée et la gestion de multiples structures.

### Accessibilité/adaptabilité

Application web entièrement responsive (PC, tablette, mobile). Atteindre le niveau **d'Accessibilité AA minimum**.

→ Garantir l'utilisabilité par le plus grand nombre et sur tous les supports (notamment pour les livraisons sur tablette).

### Sécurité/réglementation

Respecter les exigences **RGPD** et utiliser les protocoles de sécurité standard (HTTPS, Chiffrement bcrypt des mots de passe).

→ Protéger les données personnelles des adhérents et assurer la confiance du Réseau.



# Contraintes techniques



## Contraintes fonctionnelles clés

Ces contraintes techniques valident la faisabilité des modules métier.

### Logistique

Capacité à gérer un **calendrier complexe** (saisons, fermetures, jours fériés) et l'**ordre de passage des dépôts** au sein d'une tournée.

→ Permet la génération automatique des feuilles de route pour les équipes de livraison.

### Architecture

Le modèle de données doit gérer la distinction claire entre l'**Adhésion** (associative) et l'**Abonnement** (commercial).

→ Assurer la conformité avec le statut d'association loi 1901 des Jardins.





# Méthodologie et livrables

## Méthodologie de travail

- Méthodologie Agile / Scrum → Travail itératif en sprints
- Découpage du projet en phases:
  - Conception, dev, test, déploiement
- Réunions régulières (revues, points d'avancement, feedbacks)
- Livraison progressive des fonctionnalités





# Méthodologie et livrables

## Outils et environnement de travail

Domaine	Outil
IDE / Éditeur	VS Code / IntelliJ
Gestion de projet	Trello / Notion / Gantt
Contrôle de version	Git / GitHub
Conception & maquettes	Figma
Documentation	Markdown / Google Docs
Communication	Discord / Teams





# Méthodologie et livrables

## Livrables attendus

- Cahier des charges technique – Document de référence du projet
- Maquette UI / UX – Interfaces du front et back office
- Code source documenté – disponible sur GitHub
- Cahier de tests – Scénarios et validation des fonctionnalités
- Documentation utilisateur – mode d'emploi pour les gestionnaires et les adhérents
- Présentation finale / Soutenance





# Planning prévisionnel

III – Développement front-end  
Durée estimée: 2 semaines

II – Développement back-end  
Durée estimée: 2 semaines

IV – Intégration et tests  
Durée estimée: 1 semaine

I – Analyse et modélisation  
Durée estimée: 2 semaines



V – Recette et déploiement  
Durée estimée: 1 semaine