

# CAHIER DES CHARGES



**Sujet : Gestion de la Relation Client**

**Société d'accueil : DOPAWIN**

**Encadrant ISET**

**CHAABANI MARWA**

**Encadrant professionnel**

**RMILI ACHREF**

**Membres du binôme**

**KHADHRI RAMI  
ZARRAD SARRA**

**#PFE\_DSI\_2025**

# Table des matières

I.	Introduction.....	3
II.	Contexte du Projet .....	3
III.	Définition du Projet.....	3
IV.	Objectifs du Projet .....	4
V.	Choix de la Méthodologie : Scrum.....	5
VI.	Analyse des Besoins .....	7
VII.	Architecture et Technologies .....	9
VIII.	Spécifications Fonctionnelles Détaillées .....	11
IX.	Sécurité et Performances.....	14
X.	Planning et Calendrier.....	14
XI.	Ressources et Interface.....	17
XII.	Validation et Acceptation.....	21
XIII.	Conclusion .....	23

# I. Introduction

Le projet de développement d'un CRM interne vise à centraliser les données clients, améliorer la gestion des interactions et automatiser les processus de vente.

Il s'inscrit dans une démarche de digitalisation pour optimiser la communication et la collaboration entre les équipes. L'approche Agile garantira flexibilité et adaptabilité tout au long du projet, qui devrait durer quatre (4) mois.

## II. Contexte du Projet

La gestion efficace des relations clients est un enjeu majeur pour notre organisation, nécessitant des outils adaptés pour centraliser les informations, optimiser les interactions et améliorer la collaboration entre les équipes.

Dans un contexte où la digitalisation devient essentielle, le développement d'un CRM interne s'impose comme une solution stratégique. Ce projet s'appuiera sur des technologies modernes afin de garantir une performance optimale et une évolutivité à long terme.

## III. Définition du Projet

Le projet consiste à concevoir et déployer un CRM interne sur mesure, basé sur une architecture moderne et flexible. Le back-end sera développé avec **Spring Boot**, un framework Java robuste et évolutif, tandis que le front-end utilisera **React**, une bibliothèque JavaScript reconnue pour sa réactivité et son dynamisme.

La gestion des données sera assurée par **PostgreSQL**, un système de gestion de bases de données relationnelles performant et fiable.



## IV. Objectifs du Projet

### 1. Objectifs fonctionnels :

- Centraliser les données clients et améliorer le suivi des interactions.
- Automatiser les processus de vente et les tâches récurrentes.
- Faciliter la collaboration entre les équipes commerciales et support.

### 2. Objectifs techniques :

- Mettre en place une architecture modulaire favorisant la maintenabilité et l'évolutivité.
- Assurer une sécurité renforcée avec une gestion des accès et des permissions.

### 3. Objectifs méthodologiques :

- Adopter une approche **Agile** pour garantir flexibilité et adaptation aux besoins.
- Intégrer une phase de formation et de support technique après le déploiement.

La durée estimée du projet est de **4 mois**, incluant le développement, la mise en production et l'accompagnement des utilisateurs.



## V. Choix de la Méthodologie : Scrum

Pour mener à bien ce projet de CRM interne, nous avons choisi d'adopter **Scrum**, une méthodologie Agile qui permet une gestion itérative et collaborative du développement.

### 1. Pourquoi Scrum ?

Scrum est particulièrement adapté aux projets complexes nécessitant une flexibilité et une réactivité aux changements. Son approche itérative et incrémentale offre plusieurs avantages :

- **Amélioration continue** : Chaque itération (Sprint) permet d'ajuster le produit en fonction des retours des utilisateurs.
- **Transparence et collaboration** : L'ensemble de l'équipe est impliqué dans les prises de décision via des cérémonies Scrum régulières.

- **Livraison rapide et progressive** : Des versions fonctionnelles du CRM seront disponibles à chaque Sprint, permettant des tests précoces.

## 2. Organisation du Projet selon Scrum

- **Sprint Planning** : Définition des objectifs et des tâches à accomplir pour chaque Sprint (durée de 2 semaines).
- **Daily Scrum** : Courte réunion quotidienne (15 minutes) pour synchroniser l'équipe et identifier les obstacles.
- **Sprint Review** : Présentation des fonctionnalités développées à la fin de chaque Sprint pour obtenir des feedbacks.
- **Sprint Retrospective** : Analyse des améliorations possibles pour optimiser l'organisation et les processus.

## 3. Rôles Clés dans le Projet

- **Product Owner** : Définit et priorise les fonctionnalités en fonction des besoins métiers.
- **Scrum Master** : Facilite le bon déroulement des Sprints et s'assure du respect de la méthodologie.
- **Équipe de développement** : Composée de développeurs back-end, front-end.

## VI. Analyse des Besoins

Pour le développement de notre CRM interne, il est crucial d'identifier et de détailler les besoins fonctionnels et non fonctionnels qui guideront la conception et l'implémentation du système.

### 1. Besoins Fonctionnels

Les besoins fonctionnels principaux incluent :

1. **Gestion des Contacts** : Le système doit permettre l'ajout, la modification et la suppression de contacts, ainsi que la possibilité de segmenter ces contacts en fonction de divers critères (par exemple, secteur d'activité, localisation géographique).
  2. **Opportunités Commerciales** : Il est essentiel de pouvoir suivre les opportunités commerciales depuis leur création jusqu'à leur clôture. Cela inclut la gestion des étapes du processus de vente, des prévisions de revenus, et des rapports personnalisés sur les performances des ventes.
  3. **Gestion des Tâches et tickets**: Le CRM doit intégrer un module de gestion des tâches pour les utilisateurs, permettant de créer, assigner et suivre des tâches liées aux contacts et aux opportunités. Cela facilitera la collaboration entre les membres de l'équipe. Création d'un ticket de support par un utilisateur ou un administrateur. Et assigner un ticket à un agent pour traitement.
- 
1. **Authentification** : Une gestion des accès sécurisée est primordiale. Le système doit permettre une authentification par mot de passe, avec des rôles et des permissions définis pour chaque utilisateur, garantissant ainsi que seules les personnes autorisées puissent accéder à certaines informations.

2. **Reporting et tableau de board** : Des fonctionnalités de reporting avancées doivent être intégrées, permettant la génération de rapports sur les performances commerciales et visualisation des statistiques.
3. **Gestion des emails** : Enregistrement et suivi des échanges avec les contacts via emails avec notifications et rappels.
4. **Import/Export des Données** : Importation et exportation des contacts, tâches et opportunités au format CSV ou Excel, avec validation des données.

## 2. Besoins Non Fonctionnels

En ce qui concerne les besoins non fonctionnels, plusieurs aspects doivent être pris en compte :

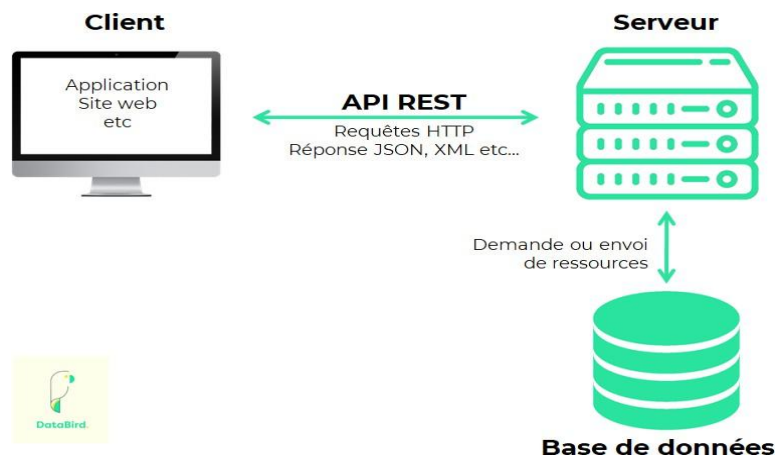
1. **Performance** : Le système doit être capable de gérer un volume élevé de données et d'utilisateurs simultanément sans dégradation des performances.
2. **Sécurité** : La protection des données sensibles est essentielle. Des mesures de sécurité telles que le cryptage des données, des sauvegardes régulières et une surveillance constante doivent être mises en place.
3. **Scalabilité** : Le CRM doit être conçu pour pouvoir évoluer facilement en fonction de l'augmentation du nombre d'utilisateurs et de données.
4. **Ergonomie** : L'interface utilisateur doit être intuitive et facile à naviguer pour garantir une adoption rapide par les utilisateurs.
5. **Compatibilité** : Le système doit être compatible avec d'autres outils et systèmes déjà en place au sein de l'organisation, facilitant ainsi l'intégration et l'interopérabilité.



## VII. Architecture et Technologies

L'architecture technique proposée pour notre projet de CRM interne repose sur une approche modulaire, intégrant une API REST comme point central de communication entre les différents composants du système.

Cette architecture se décline en trois grandes parties : le front-end, le back-end et la base de données, tout en intégrant des mesures de sécurité robustes.



### 1. Schéma Global

#### 1. Front-end :

- Le front-end sera développé en utilisant **React**, une bibliothèque JavaScript qui permet de créer des interfaces utilisateur dynamiques. React offre une expérience utilisateur fluide et réactive grâce à son système de composants réutilisables.
- Pour la gestion de l'état de l'application, nous utiliserons **Redux**, qui facilitera la gestion des données au sein de l'application frontale.

## 2. Back-end

- sera construit avec **Spring Boot**, un framework Java qui simplifie le développement de microservices. Spring Boot permettra de créer des API RESTful, facilitant ainsi la communication entre le front-end et la base de données.
- Pour l'authentification et la sécurité, nous intégrerons **Spring Security**, qui offre des mécanismes de protection contre les accès non

## 3. Base de données :

- Nous utiliserons **PostgreSQL** comme système de gestion de base de données relationnelle. PostgreSQL est connu pour sa robustesse et sa capacité à gérer de grandes quantités de données. Il nous permettra de stocker et de récupérer efficacement les informations des clients et des opportunités commerciales.

## 2. Mesures de Sécurité

La sécurité des données est une priorité absolue dans notre architecture. Nous mettrons en place plusieurs mesures, notamment :

- **Cryptage des données sensibles** en transit et au repos pour protéger les informations personnelles des utilisateurs.(Hashage de mot de passe)
- **Authentification multifactorielle (MFA)** pour renforcer la sécurité des comptes utilisateurs (avec Google Authenticator App)
- **Sauvegardes régulières** de la base de données pour garantir la continuité des opérations en cas de défaillance système.
- **Surveillance et audit des accès** pour détecter toute activité suspecte ou non autorisée. (journal de connexion )

### 3. Technologies Spécifiques

En résumé, les technologies spécifiques utilisées dans chaque composant de notre architecture sont :

- **Front-end** : React, Redux
- **Back-end** : Spring Boot, Spring Security
- **Base de données** : PostgreSQL

Cette architecture technique est conçue pour offrir une solution évolutive, sécurisée et performante, répondant aux besoins fonctionnels et non fonctionnels identifiés dans notre analyse.

## VIII. Spécifications Fonctionnelles Détaillées

### 1. Gestion des Utilisateurs et des Rôles

1. Création, modification et suppression d'un utilisateur par le Super Admin.
2. Définition des rôles et permissions : Super Admin, Admin, Utilisateur.
3. Activation/Désactivation d'un compte utilisateur.
4. Gestion du profil utilisateur (nom, email, téléphone, photo, préférences).
5. Historique des connexions et activité utilisateur.

### 2. Gestion des Contacts

1. Ajout de contacts avec des champs personnalisables (Nom, Prénom, Email, Téléphone, Entreprise, Notes).
2. Modification et suppression sécurisées des contacts avec confirmation.
3. Segmentation et recherche avancée par critères (secteur, localisation, type de contact).

4. Association des contacts avec des opportunités, tâches et historiques d'interactions.
5. Importation de contacts depuis un fichier CSV/Excel.
6. Exportation des contacts vers différents formats (CSV, Excel, PDF).

### **3. Gestion des Opportunités**

1. Création, modification et suppression d'une opportunité commerciale.
2. Suivi des étapes du cycle de vente (Prospection, Négociation, Gagné/Perdu).
3. Association d'une opportunité à un contact et à un responsable commercial.
4. Prédiction des revenus en fonction du pipeline et des opportunités ouvertes.
5. Génération de rapports sur les performances des ventes.
6. Suivi des interactions liées aux opportunités (Emails, Appels, Réunions).

### **4. Gestion des Tâches et des Tickets**

#### **4.1. Tâches**

1. Création et assignation d'une tâche à un membre de l'équipe.
2. Définition des délais, des priorités et des statuts (À faire, En cours, Terminé).
3. Suivi de l'avancement des tâches via un tableau de bord.
4. Notifications automatiques des échéances à venir.
5. Stockage de l'historique des tâches terminées.

#### **4.2. Tickets de Support**

1. Création d'un ticket de support par un utilisateur ou un administrateur.
2. Assignation d'un ticket à un agent pour traitement.
3. Suivi des statuts des tickets (Ouvert, En cours, Résolu, Fermé).
4. Historique des tickets et exportation des rapports d'intervention.

## **5. Authentification et Sécurité**

1. Authentification par mot de passe avec exigences de complexité.
2. Authentification à deux facteurs (MFA) avec Google Authenticator, valable 7 jours.
3. Gestion des sessions et déconnexion automatique après inactivité.
4. Récupération sécurisée du mot de passe via un lien temporaire envoyé par email.
5. Historique des connexions et alertes en cas d'accès suspect.

### **6. Gestion des Emails**

1. Gestion et suivi des emails envoyés et reçus.
2. Association des emails aux contacts et opportunités.
3. Notifications et rappels pour les suivis à effectuer.

## **7. Reporting et Tableau de Bord**

1. Génération de rapports dynamiques basés sur des filtres personnalisables.
2. Création de tableaux de bord interactifs pour visualiser les KPI (Chiffre d'affaires, Opportunités, Tickets en cours, etc.).
3. Analyses historiques et tendances sur les ventes, tâches et interactions clients.
4. Exportation des rapports en format PDF, Excel.

## **8. Import/Export des Données**

1. Importation de contacts, opportunités et tâches depuis des fichiers CSV ou Excel.

2. Exportation des données vers différents formats (CSV, Excel, PDF).
3. Validation des données avant importation pour éviter les erreurs.
4. Mécanisme de logs et d'historique des opérations d'import/export.

## **IX. Sécurité et Performances**

La sécurité du CRM sera assurée par le cryptage des données, l'authentification multifactorielle (MFA) et des mécanismes de surveillance continue. L'optimisation des requêtes et l'utilisation d'index dans PostgreSQL garantiront des performances élevées. L'architecture modulaire favorisera la scalabilité et l'évolution du système sans impact majeur. Enfin, des outils d'audit et de journalisation permettront de détecter toute activité suspecte et d'assurer la conformité

## **X. Planning et Calendrier**

Le calendrier de notre projet de développement d'un CRM interne se divise en plusieurs phases clés, chacune ayant une durée et des objectifs spécifiques. Voici un aperçu des étapes principales :

### **1. Analyse et spécification (Durée : 2 semaines)**

Cette phase initiale consiste à définir les objectifs du projet, identifier les parties prenantes et établir un cahier des charges préliminaire. Les activités comprennent des réunions de lancement, des sessions de brainstorming pour recueillir les besoins des utilisateurs, et la création d'un plan de communication. L'objectif est d'assurer une compréhension claire des attentes et des contraintes du projet

## **2. Conception (Durée : 3 semaines)**

La phase de conception se concentre sur la création d'architectures fonctionnelles et techniques détaillées. Cela inclut la modélisation de la base de données, la conception de l'interface utilisateur, et la définition des API. Des prototypes peuvent être réalisés pour valider les choix esthétiques et fonctionnels auprès des utilisateurs. Cette phase vise à établir un document de conception qui servira de base pour le développement.

## **3. Développement (Durée : 8 semaines)**

Le développement est la phase la plus intensive, où les équipes de développement travailleront sur la création des modules du CRM. Cette étape sera divisée en sprints Agile, permettant des itérations rapides et une adaptation aux retours des utilisateurs. Les tâches incluront l'implémentation des fonctionnalités de gestion des contacts, d'opportunités, de tâches, et de reporting etc... Un suivi régulier sera effectué pour s'assurer que les délais et les critères de qualité sont respectés.

## **4. Test et validation (Durée : 1 semaine)**

Une fois le développement achevé, des tests rigoureux seront menés pour garantir la qualité du produit. Cela comprendra des tests unitaires, des tests d'intégration et des tests d'acceptation utilisateur. L'objectif est d'identifier et de corriger les erreurs avant le déploiement. Des sessions de formation peuvent également être organisées pour préparer les utilisateurs finaux à utiliser le nouveau système.

## 5. Déploiement et documentation (Durée : 2 semaines)

La phase de déploiement du CRM comprend l'installation et la configuration du système sur l'environnement de production. Elle inclut la migration des données, les tests finaux de validation et la mise en ligne officielle. Une surveillance continue sera assurée afin d'identifier et corriger les éventuels dysfonctionnements.

Parallèlement, une documentation complète sera fournie, couvrant l'installation, l'utilisation et la maintenance du CRM. Cette documentation inclura un guide utilisateur détaillé ainsi qu'un manuel technique destiné aux administrateurs et développeurs pour faciliter l'évolution future du système.



## XI. Ressources et Interface

Pour mener à bien le développement de notre CRM interne, il est essentiel d'identifier les ressources humaines nécessaires ainsi que les ressources matérielles

### 1. Ressources Humaines

Les ressources humaines requises pour ce projet comprennent plusieurs types de développeurs et de designers. Nous aurons besoin de :

**Développeurs Full stack** : Deux développeurs spécialisés en Java et en Spring Boot pour construire l'API et gérer la logique métier. Ces développeurs devront également avoir une bonne compréhension des bases de données, en particulier PostgreSQL.

Maîtrisant React et Redux pour créer une interface utilisateur fluide et réactive. Leur rôle sera crucial pour garantir une expérience utilisateur optimale.

### 2. Ressources Matérielles

Concernant les ressources matérielles, nous aurons besoin de serveurs pour héberger l'application et la base de données, ainsi que des postes de travail pour les développeurs et le designer. De plus, l'utilisation de logiciel de gestion de projet **Jira** et d'outils de collaboration **Slack** sera nécessaire pour faciliter la communication et le suivi des tâches.

### 3. Public Cible

Le CRM interne est destiné aux entreprises souhaitant optimiser leur gestion des relations clients et améliorer leur efficacité commerciale. Les principaux utilisateurs incluent :

- **Utilisateur (User)** : Gère ses propres contacts, suit ses opportunités, crée et suit des tâches. Il peut consulter les informations nécessaires à son activité sans accès aux fonctionnalités avancées.
- **Administrateur (Admin)** : Supervision des performances commerciales et analyse des données.
- **Super Administrateur (Super Admin)** : Dispose de tous les droits, y compris la gestion des utilisateurs et des permissions, la configuration du CRM, l'import/export des données et l'accès à toutes les statistiques et rapport



## 4. Choix des Couleurs

Les couleurs du CRM doivent refléter une identité professionnelle, moderne et intuitive. Voici les choix recommandés :

- **Bleu (#0056B3)** : Transmet la confiance, la fiabilité et la sérénité. Principalement utilisé pour les éléments interactifs et les barres de navigation.

**Blanc (#FFFFFF)** : Favorise la lisibilité et apporte une sensation de clarté et d'espace.

- **Gris clair et foncé (#B6B6B6,)** : Utilisé pour les arrière-plans secondaires et les textes afin d'améliorer le contraste et l'ergonomie.
- **Vert (#28A745)** : Pour les validations et succès, symbolisant la croissance et la productivité.
- **Orange (#FFA500)** : Accent utilisé pour les rappels et notifications, attirant l'attention sans être agressif.

## 5. Logo



## 1. Défis d'intégration technique

**Risque** : Les systèmes existants peuvent ne pas être facilement intégrables avec le nouveau CRM, ce qui pourrait entraîner des retards et des coûts supplémentaires.

**Plan de Mitigation** : Effectuer une analyse approfondie des systèmes existants avant le début du développement pour identifier les problèmes potentiels d'intégration. Prévoir une phase de prototypage pour tester les intégrations clés en amont, et planifier des ressources dédiées à la résolution des problèmes d'intégration.

## 2. Délais de livraison

**Risque** : Les délais de développement pourraient être dépassés en raison de l'ampleur du projet ou de changements dans les exigences.

**Plan de Mitigation** : Adopter une méthodologie Agile, permettant des itérations fréquentes et des ajustements en cours de projet. Planifier des revues régulières avec les parties prenantes pour s'assurer que le projet reste aligné avec les attentes et les délais.

## 3. Sécurité des données

**Risque** : Les données sensibles des clients pourraient être compromises lors du développement ou après le déploiement.

**Plan de Mitigation** : Mettre en place des mesures de sécurité robustes dès le début, y compris le cryptage des données, des protocoles d'authentification forte, et des audits réguliers de sécurité. Former les équipes sur les meilleures pratiques en matière de sécurité des données.

## 4. Adoption utilisateur

**Risque** : Les utilisateurs finaux peuvent être réticents à adopter le nouveau système, ce qui pourrait réduire son efficacité.

**Plan de Mitigation** : Organiser des sessions de formation et de sensibilisation pour les utilisateurs, en impliquant des utilisateurs clés dès le début du projet pour recueillir leurs retours. Créer une documentation claire et accessible pour aider les utilisateurs à s'adapter au nouveau système.

## 5. Performance du système

**Risque** : Le système pourrait ne pas répondre aux performances attendues, entraînant une mauvaise expérience utilisateur.

**Plan de Mitigation** : Effectuer des tests de performance tout au long du développement pour identifier et corriger les goulets d'étranglement. Prévoir une capacité de scalabilité dès la conception pour assurer une performance optimale avec l'augmentation des utilisateurs et des données.

## XII. Validation et Acceptation

La validation et l'acceptation du projet CRM interne sont des étapes cruciales pour garantir que le système développé répond aux exigences établies et satisfait les attentes des utilisateurs. Ces étapes permettent de s'assurer que le produit final est conforme aux spécifications définies et qu'il est prêt à être déployé au sein de l'organisation.

## 1. Critères de Validation

Les critères de validation sont des normes spécifiques que le CRM doit respecter avant d'être accepté. Ces critères incluent :

1. **Fonctionnalité** : Toutes les fonctionnalités spécifiées dans le cahier des charges doivent être implémentées et fonctionner correctement. Cela comprend la gestion des contacts, le suivi des opportunités, et la génération de rapports.
2. **Performance** : Le système doit répondre à des exigences de performance définies, telles que le temps de réponse lors des requêtes ou la capacité à gérer un certain nombre d'utilisateurs simultanément.
3. **Sécurité** : Les mesures de sécurité mises en place doivent être vérifiées pour garantir la protection des données sensibles. Cela inclut le cryptage des données et la mise en œuvre d'une authentification sécurisée.
4. **Ergonomie** : L'interface utilisateur doit être intuitive et facile à naviguer, garantissant ainsi une adoption rapide par les utilisateurs.

## 2. Processus d'Acceptation

Le processus d'acceptation implique plusieurs étapes clés :

1. **Réunions de Validation** : Des réunions régulières seront organisées avec les parties prenantes pour examiner les avancées du projet, discuter des résultats des tests et recueillir des retours. Ces réunions permettront de s'assurer que le développement est en adéquation avec les attentes.

2. **Tests d'Acceptation Utilisateur (UAT)** : Avant l'acceptation finale, des tests d'acceptation utilisateur seront réalisés. Ces tests permettent aux utilisateurs finaux de tester le système dans un environnement proche de la production, afin de valider que toutes les fonctionnalités répondent à leurs besoins.
3. **Gestion des Corrections** : Si des défauts ou des problèmes sont identifiés lors des tests, un processus de gestion des corrections sera mis en place. Cela inclut la documentation des problèmes, l'allocation des ressources nécessaires pour les corriger et la planification de tests de validation pour s'assurer que les corrections sont efficaces.
4. **Acceptation Finale** : Une fois que toutes les corrections ont été effectuées et que le système répond aux critères de validation, une réunion d'acceptation finale sera organisée. Lors de cette réunion, les parties prenantes signeront un document d'acceptation, confirmant que le projet est prêt à être déployé.

## XIII. Conclusion

En conclusion, ce projet de CRM interne permettra d'améliorer significativement la gestion des relations clients au sein de l'organisation, tout en optimisant l'efficacité des équipes et la qualité des interactions. Grâce à une approche Agile et à une architecture modulaire, il assurera une évolution continue et un soutien adapté aux besoins futurs de l'entreprise.