



Cahier des Charges

Projet : **SaaS d'Optimisation de l'Eau pour l'Agriculture (Smart Irrigation)**

Réalisé par : **Abdeljabbar MOUDIRI– YouCode Safi**

Encadré par : **Latifa AMOUGUAY– YouCode Safi**

1. **Présentation du projet**

1.1. Contexte

Au Maroc et dans la région MENA, la **rareté de l'eau** constitue un défi majeur pour l'agriculture, principale source d'emploi rural.

Les pratiques d'irrigation non optimisées entraînent un **gaspillage important de l'eau** et une **baisse de la productivité**.

Ce projet vise à créer une **plateforme SaaS intelligente** qui aide les agriculteurs et coopératives à **optimiser la consommation d'eau** à travers des outils numériques basés sur les **données météo, du sol et des capteurs IoT**.

2. **Objectifs du projet**

- Fournir un **outil numérique d'aide à la décision** pour l'irrigation.
 - Réduire la consommation d'eau et améliorer les rendements agricoles.
 - Offrir une **interface claire et simple** pour les agriculteurs, ONG et coopératives.
 - Permettre l'accès à des **statistiques et alertes en temps réel**.
 - Promouvoir la **transition vers une agriculture durable** au Maroc.
-

3. **Cibles du projet**

- Coopératives agricoles
- Exploitations familiales

- ONG et associations rurales
- Ministères et programmes d'irrigation durable

4. 💡 **Fonctionnalités principales (MVP)**

Module	Fonctionnalités principales
Authentification	- Inscription et connexion avec JWT - Gestion des rôles (Admin, Agriculteur, ONG)
Gestion des exploitations	- Ajout/modification/suppression d'une ferme - Association à un utilisateur
Module météo	- Intégration avec API OpenWeatherMap pour récupérer les données météo locales
Planification d'irrigation	- Recommandations automatiques selon météo et type de culture - Calendrier d'arrosage
Suivi de consommation	- Historique d'irrigation - Statistiques d'eau utilisée - Graphiques (par jour, semaine, mois)
Alertes intelligentes	- Notification si excès ou manque d'eau détecté - Alerte selon seuils personnalisés
Tableau de bord	- Vue synthétique de l'exploitation, météo, capteurs et consommation
Module financement/subventions	- Informations sur les aides et programmes d'économie d'eau (optionnel)

5. 🚀 **Fonctionnalités avancées (V2 - Bonus)**

- Intégration de **capteurs IoT** (ESP32, capteur humidité, DHT22)
- Intégration avec **InfluxDB** pour stockage de données capteurs temps réel
- Visualisation via **Grafana**
- Suggestions automatiques d'irrigation basées sur l'IA (future extension)

6. 🧱 **Architecture technique**

6.1. Stack technique

Couche	Technologie
Frontend	Angular 18 , HTML5, CSS3, TypeScript, Tailwind
Backend	Spring Boot 3, Spring Security, Spring Data MongoDB, Spring Cloud
Base de données	MongoDB (principale) (optionnel : InfluxDB pour capteurs)
Authentification	JWT (JSON Web Token)
API externe	OpenWeatherMap API
Conteneurisation	Docker + Docker Compose
Déploiement	Render / AWS / Railway
Monitoring	Grafana (optionnel)
Outils Dev	Postman, Git/GitHub, VS Code, IntelliJ, MongoDB Compass

7. 🧠 Architecture logicielle (modèle MVC / microservices)

- **Service Utilisateur** : gestion des comptes et rôles
- **Service Ferme** : gestion des exploitations agricoles
- **Service Météo** : intégration OpenWeather API
- **Service Irrigation** : planification et historique
- **Service Alertes** : génération et envoi d'alertes
- **Service IoT (optionnel)** : réception des données capteurs via API REST

🔗 Communication : REST + JSON

🔒 Sécurité : Spring Security + JWT

🔍 Découverte des services : Spring Cloud Eureka

☁️ Configuration centralisée : Spring Cloud Config

8. 💰 Matériel (prototype IoT facultatif)

Matériel	Description	Prix (MAD)
ESP32	Microcontrôleur WiFi	~100
Capteur humidité sol	Mesure l'humidité	~30

Matériel	Description	Prix (MAD)
DHT22	Température et humidité	~50
Pompe à eau miniature	Test irrigation	~40
Breadboard + câbles	Connexion	~40
Total prototype		~260 MAD

9. 🎨 Maquettes et interface utilisateur (prévision)

- Page d'accueil explicative du SaaS
- Tableau de bord agriculteur : météo, consommation, alertes
- Page de gestion des exploitations
- Historique d'irrigation (graphiques)
- Section des subventions et conseils

(Les maquettes seront conçues avec Figma ou Canva.)

10. 📈 Plan de réalisation (phases)

Phase	Description	Durée estimée
Phase 1	Étude et conception UML + cahier des charges	1 semaine
Phase 2	Développement backend (Spring Boot + MongoDB)	2 semaines
Phase 3	Développement frontend (Angular)	2 semaines
Phase 4	Intégration API météo et alertes	1 semaine
Phase 5	Tests, Dockerisation et documentation	1 semaine
Phase 6 (optionnelle)	Module IoT et InfluxDB	1 semaine
Total estimé		6 à 8 semaines

11. 📄 Livrables

- Cahier des charges
- Diagrammes UML (cas d'utilisation, séquence, classes, déploiement)
- Code source (GitHub)

- Base de données MongoDB exportée
 - Documentation technique et utilisateur
 - Rapport final + présentation (pitch YouCode)
-

12. **Impact attendu**

- Diminution du gaspillage d'eau agricole
- Sensibilisation à l'agriculture durable
- Valorisation des compétences technologiques dans le domaine AgriTech
- Potentiel de déploiement réel dans les coopératives marocaines