Dossier de Spécifications Techniques - Application de Veille des Marchés Publics

# Objectif du projet

* Développer une application web permettant aux enterprises de:
* - S'inscrire et définir leurs domaines d'intervention.
* - Recevoir automatiquement les publications pertinentes de la revue des marchés publics.
* - Être alertées en cas de publication ou de résultat les concernant.

# Fonctionnalités principales

## Gestion des entreprises

* - Inscription / Connexion sécurisée.
* - Interface de définition des domaines d'intervention (multi-sélection).
* - Dashboard pour consulter les publications reçues et les alertes.

## Extraction automatique

* - Scraping ou API des publications officielles de marchés publics.
* - Téléchargement automatique des fichiers PDF.

## Analyse intelligente

* - Analyse du contenu PDF pour identification de mots-clés.
* - Correspondance entre les mots-clés et les domaines des entreprises.
* - Extraction des noms d'entreprises dans les résultats de marchés.

## Notification et alertes

* - Envoi d'e-mails ou alertes in-app pour chaque publication pertinente.
* - Notification spéciale si une entreprise est identifiée comme attributaire d'un marché.

## Interface administrateur

* - Gestion des entreprises inscrites.
* - Suivi des publications analysées.
* - Logs des alertes envoyées.

# Architecture technique proposée

## Frontend

* - Framework : React.js
* - UI : Bootstrap / Tailwind CSS
* - Responsive : Oui (mobile, tablette, desktop)

## Backend

* - Framework : Django
* - API REST pour communication avec le frontend
* - Gestion de l'authentification (JWT / session-based)

## Automatisation & NLP

* - Programmation de tâches périodiques (cron / Celery)
* - Librairies : PyMuPDF / spaCy / regex avancé

## Base de données

* - PostgreSQL
* - Tables : Utilisateurs, Entreprises, Domaines, Publications, Alertes, Logs

## Infrastructure

* - Conteneurisation : Docker
* - CI/CD : GitHub Actions / GitLab CI
* - Hébergement : VPS ou Cloud (DigitalOcean, AWS, etc.)

# Modèle de données simplifié

* - Entreprise (nom, domaine, email, etc.)
* - Domaine (libellé)
* - Publication (date, titre, source, contenu extrait)
* - Alerte (entreprise, publication, type, date)
* - Utilisateur (admin ou entreprise)

# Exigences non fonctionnelles

* - Sécurité des données (HTTPS, chiffrement, RGPD)
* - Montée en charge (scalabilité backend)
* - Monitoring (logs, erreurs, succès)

# Tests prévus

* - Tests unitaires sur les composants critiques (analyse PDF, matching)
* - Tests d'intégration (du scraping à l'envoi d'alerte)
* - Recette fonctionnelle avec un jeu de données réel