Cahier des Charges

Code insight

2025

[**1.** **Contexte Et Objectives** 2](#_Toc194242636)

[**2.** **Description du système** 2](#_Toc194242637)

[**3.** **Fonctions Principales** 2](#_Toc194242638)

[**3.1** **Gestion des utilisateurs** 3](#_Toc194242639)

[**3.2** **Analyse et génération des rapports** 3](#_Toc194242640)

[**3.3** **Personnalisation et optimisation des explications** 4](#_Toc194242641)

[**3.4** **Génération et export des documents** 4](#_Toc194242642)

[**3.5** **Visualisation du code et des explications (Perspective)** 5](#_Toc194242643)

[**3.6** **Sécurité et confidentialité** 5](#_Toc194242644)

[**4. Spécifications Techniques du Frontend** 6](#_Toc194242645)

[**4.1 Technologies Utilisées** 6](#_Toc194242646)

[**4.2 Pages et Fonctionnalités** 6](#_Toc194242647)

[**5. Spécifications Techniques du Backend** 7](#_Toc194242648)

[**5.1 Technologies Utilisées** 7](#_Toc194242649)

[**5.2 Endpoints API** 7](#_Toc194242650)

[**5.3 Gestion de la Base de Données** 7](#_Toc194242651)

[**6. Spécifications Générales** 8](#_Toc194242652)

[**6.1 Fonctionnalités Générales** 8](#_Toc194242653)

[**6.2 Limitations & Contraintes** 8](#_Toc194242654)

# **Contexte Et Objectives**

* **Problématique** : Dans le domaine du développement logiciel, la compréhension du code source représente un défi majeur pour plusieurs catégories d’utilisateurs :
* Les étudiants qui découvrent la programmation et peinent à assimiler la logique et la structure des langages.
* Les développeurs juniors qui manquent d'expérience dans la lecture et l’analyse de code, surtout lorsque celui-ci est peu ou pas documenté.
* Les professionnels qui doivent maintenir ou améliorer des logiciels développés par d’autres sans disposer de documentation claire, ce qui entraîne une perte de temps et des erreurs potentielles.
* L’absence de commentaires explicites, de documentation et de bonnes pratiques de structuration rend l’interprétation du code difficile, augmentant ainsi la courbe d’apprentissage et réduisant l’efficacité du travail collaboratif.
* **Objectif global** : L’objectif principal de ce projet est de concevoir un outil basé sur l’IA capable de générer automatiquement des rapports explicatifs à partir du code fourni. Cet outil permettra de :
  + Analyser le code source et en extraire les fonctionnalités clés.
  + Fournir des explications claires et adaptées en fonction du niveau de compétence de l’utilisateur (débutant, intermédiaire, avancé).
  + Détecter et signaler d’éventuelles incohérences ou mauvaises pratiques dans le code.
  + Générer des rapports interactifs facilitant l’apprentissage et l’intégration de nouveaux développeurs dans un projet.

# **Description du système**

* Type d’application : Web
* Langues : Multilingue, avec une priorité pour le français, l’anglais et l’arabe.
* Technologies proposées :
* Frontend: React.js pour l’interface web.
* Backend & Base de données : Django pour le traitement, le stockage des rapports et l’analyse du code.
* Génération des rapports : LaTeX pour la mise en forme et conversion en PDF.
* Intégration : API REST pour faciliter l’intégration avec des plateformes d’apprentissage et outils collaboratifs.

# **Fonctions Principales**

Ce projet vise à fournir une plateforme permettant d’analyser du code source et de générer automatiquement des rapports détaillés sous format PDF. Le système repose sur l’utilisation de l’IA pour produire des explications adaptées à différents niveaux d’utilisateurs (débutants, avancés, experts).

Les fonctionnalités principales s’articulent autour de plusieurs axes :

* **Gestion des utilisateurs (si un Frontend est prévu ou si l’outil a des niveaux d'accès différents)**
* **Analyse et génération des rapports**
* **Personnalisation et optimisation des explications**
* **Génération et export des documents**
* **Visualisation du code et des explications**

## **Gestion des utilisateurs**

Le système peut inclure une gestion des utilisateurs permettant de sauvegarder des préférences et un historique d’analyses. (On peut ignorer cela si on utilise une API sans authentification)

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Inscription / Connexion | Un utilisateur peut créer un compte et se connecter. |
| Gestion des préférences | Sauvegarde des préférences de génération de rapports. |
| Historique des analyses | La possibilité d’avoir une liste avec les noms des rapports. Utile pour les rechercher dans le répertoire de sa machine (User) |
| Abonnement (perspective) | Différents rôles peuvent être définis selon l’usage. |
| Suppression des données utilisateur | Un utilisateur peut demander la suppression de ses données. |

## **Analyse et génération des rapports**

Cette section décrit la manière dont le système prend en charge l’analyse du code et la production du rapport.

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Soumission de code | L’utilisateur soumet un ou plusieurs extraits de code. |
| Sélection du type d’explication | Choix entre plusieurs types d’explication (basique, détaillée, optimisée, avec exemple). |
| Analyse syntaxique et extraction de fonctions | Détection automatique des fonctions et structures de code. |
| Appel à l’API IA pour explication | Envoi du code à une IA pour génération de texte explicatif. |

## **Personnalisation et optimisation des explications**

L’application permet une personnalisation poussée des rapports générés afin de répondre aux besoins spécifiques des utilisateurs. C’est le fait de mener l’utilisateur d’utiliser les prompts d’une façon indirecte, par exemple :

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Filtrage du niveau de détail | Option pour obtenir une explication simple ou avancée. |
| Mode "développeur avancé" | Génération d’explications techniques approfondies. |
| Mode "pédagogique" | Explication adaptée aux débutants en programmation. |
| Génération Auto | Génération automatique de rapports en PDF avec un résumé des analyses. |

## **Génération et export des documents**

Le système génère un rapport LaTeX transformé en PDF, contenant le code soumis et les explications correspondantes

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Génération automatique du rapport en LaTeX | Création d’un fichier LaTeX contenant code et explications. |
| Compilation et export en PDF | Transformation du fichier LaTeX en PDF téléchargeable. |
| Envoi par email (perspective) | Option pour recevoir le rapport par email. |

## **Visualisation du code et des explications (Perspective)**

L'application peut intégrer des outils de visualisation pour mieux comprendre le fonctionnement du code.

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Affichage du code colorisé | Présentation du code soumis avec mise en forme syntaxique. |
| Génération de diagrammes de flux | Utilisation de Mermaid.js pour représenter la logique du code. |

## **Sécurité et confidentialité**

Le système prend en charge des mesures de sécurité pour protéger le code soumis et les données des utilisateurs

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Politique de suppression automatique des fichiers | Les rapports sont supprimés après un certain délai. |
| Logs et suivi d’utilisation | Suivi des requêtes pour améliorer le service tout en respectant la confidentialité. |

# **4. Spécifications Techniques du Frontend**

## **4.1 Technologies Utilisées**

* **Framework** : React.js
* **State Management** : Redux / Context API (selon besoin)
* **Styling** : Tailwind CSS / Bootstrap
* **Routing** : React Router
* **Form Handling** : React Hook Form
* **API Communication** : Axios (pour appeler l'API Django)

## **4.2 Pages et Fonctionnalités**

**POUR LE MOMENT C’EST PREVISIONNEL. LORS D’ATTRIBUTION DES OBJECTIVES LE NOMBRE DE PAGES PEUT ETRE REDUIT/MODIFIE.**

### **4.2.1 Page de Soumission de Code**

**Objectif :** Permettre à l'utilisateur de soumettre un ou plusieurs extraits de code et personnaliser l'explication.

* Zone de texte pour entrer le code
* Bouton "Ajouter un extrait" pour soumettre plusieurs snippets
* Options de personnalisation du prompt (déroulant : Basique, Détaillé, Optimisé, Personnalisé)
* Bouton "Générer Rapport"

### **4.2.2 Page de Visualisation des Rapports**

**Objectif :** Voir la liste des rapports générés avec des options de téléchargement et de suppression.

* Liste des rapports générés avec dates
* Bouton "Télécharger PDF"
* Bouton "Supprimer"
* Option de filtrage par date / type d'analyse

### **4.2.3 Page de Tutoriel**

**Objectif :** Guider l'utilisateur sur comment utiliser l'application.

* Explications avec captures d'écran
* Vidéo de démonstration
* FAQ

### **4.2.4 Page Profil Utilisateur**

**Objectif :** Permettre à l'utilisateur de gérer son compte et voir son historique.

* Historique des rapports générés
* Bouton pour supprimer son compte
* Option pour gérer ses préférences

# **5. Spécifications Techniques du Backend**

## **5.1 Technologies Utilisées**

* **Framework**: Django Rest Framework (DRF)
* **Base de données** : PostgreSQL / SQLite
* **Authentification** : Django Auth (JWT ou Session-Based)
* **Communication avec IA** : Requêtes API vers LLaMA-3
* **Gestion des fichiers** : Stockage temporaire des PDFs
* **Tâches asynchrones** : Celery + Redis (optionnel pour envoi email)

## **5.2 Endpoints API**

### **5.2.1 Endpoint : Soumission de Code**

**URL :** /api/analyze/

### **5.2.2 Endpoint : Récupérer un Rapport**

**URL :** /api/reports/{report\_id} /

### **5.2.3 Endpoint : Historique des Rapports**

**URL :** /api/user/reports/

### **5.2.4 Endpoint : Suppression de Rapport**

**URL :** /api/reports/{report\_id} /

## **5.3 Gestion de la Base de Données**

**Modèles Django**

**User**

* Username
* Email
* Password

**Report**

* User (FK vers User)
* Created\_at
* Status (completed, processing)
* File\_path

# **6. Spécifications Générales**

## **6.1 Fonctionnalités Générales**

|  |  |
| --- | --- |
| Fonctionnalité | Description |
| Authentification | Connexion via email/mot de passe |
| Soumission de code | Support de plusieurs snippets à la fois |
| Personnalisation | Choix de format d'explication, style du PDF |
| Historique | Liste des rapports générés par l'utilisateur |
| Suppression automatique | Les fichiers PDF sont supprimés après X jours |

## **6.2 Limitations & Contraintes**

* **Limite de requêtes par utilisateur** (ex : 5 rapports par jour)
* **Stockage temporaire** (les PDFs ne sont pas conservés indéfiniment)
* **Support IA** : Limiter l’utilisation du tokens en cas d’une prompt très personnalisée