SECUSCAN

Outil d’analyse de code

Date de début : 23 juin 2025

Date de fin : 31 Août 2025

Chef de projet : Aurore Kouakou

Équipe : Léo Guerizec, Samuel Gauthier, Guillaume Parisel, Germain Rideau, Alexandre Benoist, Arthur Bartczak, Jaures Azandossessi

Cahier des Charges — Projet SecuScan

1. Contexte du projet

Aujourd'hui, de nombreuses entreprises et développeurs utilisent des outils open source ou des projets complexes comportant des dépendances multiples. Cependant, peu de solutions simples et accessibles existent pour analyser automatiquement la sécurité de ces projets, en particulier depuis une interface web ou en ligne de commande.

La cybersécurité étant un enjeu majeur, il est essentiel de disposer d'un outil capable de scanner du code source ou des fichiers de configuration pour détecter des vulnérabilités (CVE), de mauvaises pratiques ou encore des secrets exposés.

2. Objectifs

- Proposer une solution simple pour auditer la sécurité d'un projet (dépôt ou fichiers).
- Identifier les vulnérabilités et les mauvaises pratiques dans le code source et les dépendances.
- Fournir des explications techniques et des suggestions via une IA.
- Permettre à l'utilisateur de personnaliser certaines règles d'analyse.

3. Présentation du projet

SecuScan est une application web destinée à analyser le contenu de projets de développement afin de détecter des failles de sécurité, des secrets exposés, des dépendances vulnérables ou de mauvaises pratiques de codage.

Le projet s'organise autour de plusieurs modules :

- Frontend : une interface web ergonomique pour déclencher des scans, consulter les résultats, personnaliser les règles.
- Backend (FastAPI) : une API centralisant les appels vers les services d'analyse, d'intelligence artificielle et de base de données.
- Analyseur : un module qui traite les fichiers et génère des rapports de vulnérabilités.
- Intelligence artificielle (Ollama) : pour formuler des suggestions ou corriger les erreurs détectées.
- CLI (en option) : un outil pour les développeurs souhaitant interagir avec SecuScan depuis le terminal.
- L'ensemble repose sur une architecture moderne et modulaire, intégrant MongoDB, Docker et des outils de sécurité comme Semgrep, Bandit ou Trivy.

4. Fonctionnalités principales

- Scan d'un dépôt Git complet ou de fichiers choisis.

- Analyse statique du code source (peu importe le langage de programmation).
- Détection de dépendances, vulnérabilités, mauvaises pratiques de codage.
- IA intégrée pour assister dans l'analyse des failles.
- Règles personnalisables (nom, description, tags, paramètres).
- Export des résultats dans une base MongoDB exportable en json puis pdf au besoin.
- Interface CLI simple et claire.

5. Outils techniques

- **Backend :** Python (Fast API)
- **Frontend :** React js, Tailwind css
- **Analyser :** Java
- **Base de données :** MongoDB Atlas
- **Sécurité :** CVE, Kics, Trivy, GitLeaks, Bandit
- **CI/CD :** GitHub Actions
- **Déploiement :** Docker
- **Outils IA :** Ollama, intégration basique dans l'analyse
- **Hébergement :** Render, Vercel, Railway

6. Public cible

- Développeurs souhaitant vérifier la sécurité de leur projet.
- Étudiants ou enseignants dans un cadre pédagogique.
- Entreprises ayant besoin d'un outil simple et extensible pour scanner des dépôts.

7. Livrables attendus

- Code source sur GitHub
- Documentation technique
- Fichiers de configuration de la base de données
- README complet
- Diagramme de Gantt, Kanban et charte graphique
- Rapport final / présentation orale

8. Répartitions des équipes

- Développement Frontend : Arthur, Alexandre, Jaures
- Développement Backend : Léo, Guillaume, Jaures
- Développement CLI / Analyseur : Samuel, Guillaume, Germain
- DevOps / QA : Germain
- IA : Léo, Aurore
- Base de données : Aurore