# Proyecto: SentinelCore

## 🎯 Objetivo del Proyecto

Desarrollar una aplicación modular que centralice herramientas de análisis de vulnerabilidades en sitios web y entornos cloud, ofreciendo información clara sobre cómo mitigar riesgos. El sistema incorporará un mecanismo innovador de “toquens de antesala” que interceptan archivos antes de ser almacenados, los analizan, y los redirigen a módulos especializados para su tratamiento e inoculación.

## 🔑 Funcionalidades esenciales del MVP

* Escaneo web básico mediante OWASP ZAP o Wapiti.
* Prototipo funcional de 'toquen de antesala'.
* Interfaz básica (CLI o web) para interacción con el sistema.
* Base de datos de resultados (SQLite u otra ligera).
* Sugerencias de mitigación por tipo de vulnerabilidad detectada.

## 🧩 Estructura Modular del Proyecto

SentinelCore/  
├── core/  
│ ├── main.py  
│ ├── dispatcher.py  
│ └── config/  
├── toquens/  
│ ├── antesala.py  
│ ├── static\_scan.py  
│ └── pdf\_sanitizer.py  
├── webscanner/  
│ ├── zap\_wrapper.py  
│ └── results\_parser.py  
├── gui/  
│ └── simple\_web\_gui.py  
├── database/  
│ └── models.py  
└── docs/

## 👥 Reparto del trabajo por roles

Dev A – WebScanner & Mitigación:  
- Integración de OWASP ZAP o Wapiti  
- Parseo de resultados  
- Recomendaciones OWASP  
  
Dev B – Toquen de antesala & Clasificador:  
- Desarrollo del sistema interceptador  
- Clasificación de archivos y reenvío  
  
Dev C – Toquens especializados:  
- Desarrollo de dos toquens (ClamAV/YARA y PDF Sanitizer)  
  
Dev D – Interfaz y orquestador:  
- Desarrollo de interfaz CLI o Web  
- Conexión entre módulos

## 🧠 Valor añadido frente al mercado

|  |  |
| --- | --- |
| Competencia (Nessus, OpenVAS, etc.) | SentinelCore MVP |
| Detectan vulnerabilidades | ✅ También |
| Mitigación automática real | ✅ (aunque básica) |
| Pre-almacenamiento de archivos | ✅ |
| Modularidad / Plugins | ✅ con toquens |
| Ideal para formación/laboratorio | ✅ educativo |

## 📅 Planificación Tentativa

Semana 1:  
- Diseño técnico, definición de módulos y repositorio  
  
Semana 2:  
- Desarrollo del escáner web y toquen de antesala básico  
  
Semana 3:  
- Desarrollo de interfaz, conexión entre módulos, integración base de datos  
  
Semana 4:  
- Testing, documentación, revisión, presentación y demo final

## ✅ Resultado Esperado del MVP

* Escaneo web funcional con vulnerabilidades detectadas.
* Sistema de interceptación de archivos funcional.
* Clasificación básica y redirección a toquens.
* Dos toquens funcionales (simulados o reales).
* Panel CLI o web funcional.
* Informe básico con sugerencias de mitigación.