Excelente, Chema. Estos pasos son clave para convertir tu proyecto en algo sólido y bien estructurado, especialmente si el equipo está dividido y cada persona trabaja en un módulo. Vamos por partes, con profundidad, claridad y recomendaciones prácticas:

**🔧 1. Identificar y Documentar los Módulos Funcionales**

**🎯 Objetivo:**

Definir cada componente funcional del sistema, su propósito, entradas/salidas y cómo se relaciona con el resto.

**✅ Módulos sugeridos para SentinelCore:**

| **Módulo** | **Descripción** | **Ejemplo de tecnología** |
| --- | --- | --- |
| **Escáner Web** | Escanea sitios web para detectar vulnerabilidades comunes (XSS, SQLi, etc). | OWASP ZAP, Wapiti |
| **Toquen de Antesala** | Intercepta archivos antes de que lleguen al disco, los clasifica y los reenvía. | Python + Watchdog |
| **Toquens Especializados** | Analizan tipos de archivos específicos. Ej: PDFs, binarios, ejecutables. | ClamAV, YARA, PyMuPDF |
| **Dispatcher/Orquestador** | Gestiona el flujo entre módulos y controla la lógica de decisión. | Python con colas (Celery, etc.) |
| **Base de Datos** | Guarda resultados de escaneo, trazabilidad y decisiones de toquens. | SQLite o PostgreSQL |
| **Interfaz (CLI o GUI)** | Para lanzar análisis, revisar resultados y gestionar configuraciones. | Flask, GTK, curses |
| **Módulo de Mitigación** | Muestra recomendaciones automáticas personalizadas. | OWASP, MITRE DB local |
| **Generador de Informes** | Exporta resultados a HTML o PDF. | ReportLab, Pandas |

📝 Recomendación: Usa una tabla como la anterior para documentar los módulos en tu documento de arquitectura.

**🔌 2. Definir Interfaces y Protocolos de Comunicación**

**🎯 Objetivo:**

Asegurar que los módulos se comuniquen de forma clara, predecible y extensible.

**📐 Recomendaciones:**

* **Estándar común**: JSON como formato de intercambio (ligero, humano-legible).
* **Protocolo local**: comunicación por funciones internas o colas (como multiprocessing.Queue, Redis, o simplemente archivos temporales).
* **REST API (opcional)**: Si hay GUI web o si se quiere controlar remotamente.

**📦 Ejemplo:**

{

"filename": "documento.pdf",

"filetype": "pdf",

"from": "toquen\_antesala",

"to": "pdf\_sanitizer",

"hash": "sha256:abc123...",

"timestamp": "2025-06-29T10:45:00"

}

📚 Documenta:

* Qué datos recibe cada módulo
* Qué datos entrega
* Qué errores puede devolver
* Qué pasa si un módulo falla

**👤 3. Asignar Responsables de Desarrollo para Cada Módulo**

**🎯 Objetivo:**

Dividir el trabajo para maximizar productividad y autonomía.

**👥 Ejemplo de distribución para 4 personas:**

| **Persona** | **Módulos asignados** | **Justificación** |
| --- | --- | --- |
| **A** | Escáner Web + Mitigación | Familiaridad con herramientas como ZAP o Nmap |
| **B** | Toquen de Antesala + Clasificador | Responsable del núcleo de innovación |
| **C** | Toquens Especializados + Antivirus | Desarrollo de plugins específicos |
| **D** | Interfaz + Dispatcher + DB | Conecta todo, experiencia fullstack ligera |

💡 Cada uno puede crear su propio entorno y pruebas locales, y luego integrar.

**🏗️ 4. Redactar un Documento de Arquitectura Modular**

**🎯 Objetivo:**

Centralizar toda la información técnica de la arquitectura para el equipo, revisión y defensa.

**📄 Contenido mínimo sugerido:**

1. **Resumen de módulos** (tabla como la del punto 1)
2. **Diagrama de arquitectura**
   * (ej: flujo → escáner → toquen → mitigar → informe)
3. **Protocolos de comunicación**
4. **Flujos principales (escaneo, archivo sospechoso, generación de informes)**
5. **Responsables de cada módulo**
6. **Tecnologías por módulo**
7. **Posibles extensiones futuras**

🎁 [¿Quieres que te lo redacte todo en .docx?](../Arquitectura_Modular_SentinelCore.docx) Puedo preparártelo.

**✅ 5. Validar la Arquitectura con el Equipo**

**🎯 Objetivo:**

Alinear expectativas, detectar cuellos de botella y ajustar tareas.

**🧪 Proceso recomendado:**

1. **Reunión de validación inicial**
   * Revisar cada módulo y si el responsable está cómodo con él
   * Confirmar tiempos estimados por fase
2. **Checklist para revisión:**
   * ¿Está claro cómo se comunican los módulos?
   * ¿Hay dependencias que bloqueen a otros?
   * ¿Cada persona tiene autonomía para comenzar su parte?
   * ¿Está clara la prioridad de cada módulo?
3. **Resultado esperado:**
   * Documento validado por todos
   * Roles y entregables definidos
   * Primeros commits del proyecto iniciados