Con un equipo de 4 personas y 80 horas por persona (320 horas en total), es fundamental definir un **alcance realista, modular y priorizado** para maximizar el impacto y asegurar la entrega de un prototipo funcional. Aquí tienes una propuesta adaptada a vuestras circunstancias:

**Alcance Realista y Modular del Proyecto**

**Objetivo para las Prácticas**

Desarrollar un prototipo funcional de una aplicación centralizada para el análisis de vulnerabilidades en páginas web y entornos cloud, integrando un sistema básico de “toquen de antesala” para el análisis preventivo de archivos.

**Funcionalidades Prioritarias**

**1. Módulo de Análisis de Vulnerabilidades Web**

* **Funcionalidad:** Integrar un escáner open source (por ejemplo, [OWASP ZAP](https://www.zaproxy.org/)) para analizar vulnerabilidades básicas en una URL proporcionada por el usuario.
* **Resultado:** Informe sencillo con vulnerabilidades detectadas y recomendaciones básicas de mitigación.

**2. Sistema de “Toquen de Antesala” (Versión Básica)**

* **Funcionalidad:** Crear un módulo que reciba archivos antes de almacenarlos, identifique su tipo (por extensión/MIME) y realice un análisis básico (por ejemplo, escaneo antivirus usando [ClamAV](https://www.clamav.net/)).
* **Resultado:** Solo los archivos “limpios” se almacenan; los sospechosos se rechazan o aíslan.

**3. Interfaz Web Sencilla**

* **Funcionalidad:** Dashboard para subir archivos, lanzar análisis de vulnerabilidades y consultar resultados.
* **Resultado:** Interfaz clara y usable para demostrar el flujo de trabajo.

**4. Documentación y Manual de Usuario**

* **Funcionalidad:** Documentar la arquitectura, instalación, uso y posibles ampliaciones futuras.
* **Resultado:** Facilita la entrega y evaluación del proyecto.

**Distribución del Trabajo (Sugerencia)**

| **Persona** | **Responsabilidad Principal** | **Horas Aproximadas** |
| --- | --- | --- |
| 1 | Integración y configuración de OWASP ZAP | 60 |
| 2 | Desarrollo del “toquen de antesala” y análisis de archivos | 60 |
| 3 | Desarrollo de la interfaz web (frontend y backend básico) | 60 |
| 4 | Integración de módulos, pruebas, documentación y presentación | 60 |
| Todos | Revisión cruzada, testing, puesta en común y ajustes finales | 80 (20 c/u) |

**Sugerencias para el Trabajo Colaborativo**

* **Repositorios compartidos (GitHub/GitLab):** Control de versiones y trabajo paralelo.
* **Reuniones breves y frecuentes:** Sincronización diaria o cada dos días para resolver bloqueos.
* **Tareas bien definidas y entregables intermedios:** Facilita la integración y reduce riesgos.
* **Uso de herramientas de gestión (Trello, Jira, Notion):** Para asignar tareas y visualizar el progreso.

**Roadmap de Entregas**

1. **Semana 1:**
   * Definición de requisitos, arquitectura y setup del entorno.
   * Primeros avances en el escáner de vulnerabilidades y el toquen de antesala.
2. **Semana 2:**
   * Desarrollo de la interfaz web y conexión con los módulos.
   * Pruebas unitarias de cada módulo.
3. **Semana 3:**
   * Integración de todos los módulos.
   * Pruebas integradas y ajustes.
4. **Semana 4:**
   * Documentación, manual de usuario y preparación de la presentación final.

**Futuras Ampliaciones (para dejar constancia en la memoria)**

* Integrar más escáneres (cloud, APIs).
* Ampliar el sistema de tokens para distintos tipos de archivos.
* Añadir autenticación y gestión de usuarios.
* Mejorar la interfaz y añadir visualización de datos.

**Conclusión:**  
Con este enfoque modular y priorizado, el equipo podrá entregar un prototipo funcional, bien documentado y con potencial de crecimiento, demostrando tanto habilidades técnicas como capacidad de trabajo colaborativo y gestión de proyectos.