**Plan de ejecución – MVP PlatSentinel (Demo en 8 semanas)**

**Equipo de 4 integrantes recién titulados en Seguridad Informática**

| **Rol propuesto\*** | **Foco principal** | **Competencias a reforzar durante el proyecto** |
| --- | --- | --- |
| **Líder técnico / Full‑stack** | Coordinación, frontend React, UX | Gestión ágil, UI/UX, comunicación con cliente |
| **Desarrollador API & Back‑end** | FastAPI, lógica de tokens, DB | Python avanzado, diseño REST, pruebas unitarias |
| **DevOps / Cloud & Security** | Docker‑K8s, pipelines CI/CD, hardening | IaC (Terraform), GitHub Actions, Zero‑Trust |
| **QA, Documentación & Integraciones** | Testing, bot WhatsApp, guías de usuario | Postman, Selenium, Twilio API, redacción técnica |

\* Los títulos pueden intercambiarse según la capacidad real; el plan asume 25–30 h/semana por persona.

**1 · Cronograma macro (8 semanas)**

| **Semana** | **Hitos expresos** |
| --- | --- |
| **1** | Kick‑off, reparto de roles, instalación del stack, definición de MVP |
| **2** | Estructura repo (API, UI, tokens, infra); diseño de base de datos |
| **3** | API v0 (endpoints CRUD clientes/tokens); token Nmap dockerizado |
| **4** | Dashboard básico (login, vista de tokens); pipeline CI/CD inicial |
| **5** | Bot WhatsApp «/scan» + resultados JSON; token Virustotal integrado |
| **6** | Módulo de alertas + aislamiento IP; hardening contenedores |
| **7** | Documentación GUI y API; pruebas end‑to‑end; reporte de métricas |
| **8** | Ensayo de demo comercial, cierre backlog, retro y plan post‑MVP |

**2 · Work‑packages y checklists detallados**

**WP‑A | Governance & Gestión ágil *(Líder técnico)***

| **Ítem** | **✔** | **Responsable** | **Artefacto / evidencia** |
| --- | --- | --- | --- |
| Creación board Kanban en GitHub Projects | ☐ | Líder | URL tablero |
| Definición de “Definition of Done” | ☐ | Equipo | Documento DoD |
| Reuniones daily (15 min) agendadas | ☐ | Líder | Calendario |
| Retros semanal + acta | ☐ | Líder | Acta en repo |

**WP‑B | Infraestructura & DevOps *(DevOps)***

| **Ítem** | **✔** | **Entorno** |
| --- | --- | --- |
| Dockerfile base (Python slim + best‑practices) | ☐ | /tokens/base/ |
| Compose/Kind para entorno local | ☐ | infra/docker-compose.yml |
| GitHub Actions con stages: lint → test → build → push registry | ☐ |  |
| Configuración de entorno demo (AWS Lightsail / Hetzner) | ☐ |  |
| TLS terminación vía Caddy / Traefik | ☐ |  |

**WP‑C | Back‑end & API *(Back‑end Dev)***

| **Ítem** | **✔** | **Endpoint** |
| --- | --- | --- |
| Esquema SQLAlchemy (usuarios, clientes, tokens, jobs) | ☐ |  |
| /auth – JWT con expiración y refresh | ☐ |  |
| /tokens/launch (POST) + job status | ☐ |  |
| /results/{job\_id} (GET) | ☐ |  |
| OpenAPI docs autogeneradas | ☐ |  |

**WP‑D | Tokens Docker *(Back‑end + DevOps)***

| **Ítem** | **✔** | **Observaciones** |
| --- | --- | --- |
| Token Nmap dockerizado, input JSON & output JSON | ☐ |  |
| Token Virustotal (usa API key en secret) | ☐ |  |
| Token Firewall‑Isolate (iptables o ufw) | ☐ |  |
| Manifest estándar (manifest.yml) validado por esquema | ☐ |  |
| Registro interno de imagen + digest firmado (cosign) | ☐ |  |

**WP‑E | Dashboard Web *(Líder técnico)***

| **Ítem** | **✔** | **Rutas React** |
| --- | --- | --- |
| Login + MFA (OTP email) | ☐ | /login |
| Listado de tokens + botón “Ejecutar” | ☐ | /tokens |
| Vista de resultados – tabla paginada | ☐ | /results |
| Indicadores en tiempo real (WebSocket) | ☐ | /status |

**WP‑F | Bot WhatsApp & Integraciones *(QA/Integraciones)***

| **Ítem** | **✔** | **Detalle** |
| --- | --- | --- |
| Configuración Twilio sandbox | ☐ |  |
| Comando /scan <IP> → API /tokens/launch | ☐ |  |
| Manejo de errores y help /help | ☐ |  |
| Firma de mensajes (HMAC) | ☐ |  |
| Pruebas Postman + captura de flujo | ☐ |  |

**WP‑G | QA, Seguridad y Documentación *(QA/Integraciones + Equipo)***

| **Ítem** | **✔** | **Herramienta** |
| --- | --- | --- |
| Unit tests ≥ 70 % coverage | ☐ | Pytest |
| Escaneo SAST (Semgrep) en pipeline | ☐ |  |
| Pentest básico con OWASP ZAP (API y UI) | ☐ |  |
| Manual de instalación rápida (README) | ☐ |  |
| Guía de uso demo + vídeo 3 min | ☐ |  |

**3 · Recomendaciones de mejora temprana**

1. **Template de token**: generar cookiecutter para acelerar nuevos contenedores.
2. **Secrets Manager**: usar HashiCorp Vault dev‑mode para API keys (Virustotal).
3. **Log centralizado**: Filebeat + Elastic‑Lite; facilita troubleshooting.
4. **Panel Dark‑mode**: mejor para demos en conferencias.
5. **Script de datos ficticios**: poblar la demo con alertas y resultados realistas.
6. **Feedback loop**: crear encuesta corta para pilotos y definir backlog “Fase 2”.

**4 · Indicadores de seguimiento (KPIs demo)**

| **Métrica** | **Objetivo** | **Frecuencia de revisión** |
| --- | --- | --- |
| Cobertura tests | ≥ 70 % | Semanal |
| Jobs de token exitosos | ≥ 95 % | CI diario |
| Tiempo de despliegue CI/CD | < 8 min | En PR |
| Latencia comando WhatsApp | < 5 s | Prueba fin de sprint |
| Disponibilidad demo cloud | 99 % | UptimeRobot |

**5 · Plan de aprendizaje**

| **Disciplina** | **Recurso sugerido** | **Sesiones internas** |
| --- | --- | --- |
| Docker & K8s | *Docker in Action*, Play With Kubernetes | Taller Semana 2 |
| FastAPI avanzado | Docs oficiales + Curso Test‑Driven FastAPI | Código compartido |
| React + Tailwind | FrontendMasters Intro | Pair‑coding UI |
| DevSecOps | eBook *Automating DevSecOps* | Kata SAST semana 5 |
| WhatsApp/Twilio | Quickstarts Twilio | Demo live semana 5 |

**Nota**: dedicar 4 h semanales a formación dirigida; registrar lecciones aprendidas en Wiki.

**✅ Entregable final (fin de Semana 8)**

* URL demo pública (dashboard + bot operativo).
* Repositorio Git con pipelines verdes.
* Manual PDF para clientes (5 pág.) y vídeo demostración.
* Informe de retrospectiva y backlog priorizado post‑MVP.

Con este plan, el equipo podrá **aprender mientras construye** y, en 2 meses, presentar a clientes una demo viable, segura y profesional de PlatSentinel.