A continuación, se presenta una lista completa de las herramientas necesarias para ejecutar el proyecto \*\*PlatSentinel\*\*, basada en los documentos proporcionados (`Arbol\_Per.docx`, `Arbol\_GPT.docx`, y las consultas previas). Cada herramienta incluye una descripción general de su función y su propósito específico en el proyecto. La lista abarca herramientas de desarrollo, despliegue, pruebas, integración, y documentación, alineadas con la arquitectura modular, contenerizada (Docker), y orientada a DevSecOps de PlatSentinel. La respuesta está estructurada en un `<xaiArtifact>` con un UUID único, en español, y organizada para claridad y profesionalismo, incluyendo todas las herramientas necesarias para un despliegue exitoso en un entorno cloud con integración de WhatsApp y soporte para PYMEs y servicios técnicos.

**Herramientas Necesarias para el Proyecto PlatSentinel**

Esta lista detalla todas las herramientas requeridas para desarrollar, desplegar, probar y mantener **PlatSentinel**, una plataforma SaaS modular de ciberseguridad para PYMEs y servicios técnicos. Cada herramienta incluye su función general y su rol específico en el proyecto, alineado con la arquitectura Dockerizada, la integración con WhatsApp, y la gestión de servicios externos.

**1. Herramientas de Desarrollo**

**Python 3.8+**

* **Función General**: Lenguaje de programación versátil, utilizado para backend, scripts y automatización debido a su simplicidad y amplia comunidad.
* **Función en PlatSentinel**:
  + Base para el backend (API REST con FastAPI en codigo\_fuente/api/).
  + Implementación de toquens (toquen\_antesala/escaneo.py, toquen\_pdf/sanitizer.py).
  + Lógica de integración con WhatsApp (bot\_whatsapp/main.py) y servicios externos (integracion/mapeo\_servicios.py).
  + Generación de informes (informes/generador\_informes.py).

**Node.js (16+)**

* **Función General**: Entorno de ejecución para JavaScript, ideal para aplicaciones web modernas y escalables.
* **Función en PlatSentinel**:
  + Soporte para el frontend basado en Vue.js (codigo\_fuente/dashboard/).
  + Ejecución de scripts de construcción y despliegue del dashboard (npm run build en dashboard/Dockerfile).

**Vue.js 3**

* **Función General**: Framework JavaScript progresivo para construir interfaces de usuario reactivas y modulares.
* **Función en PlatSentinel**:
  + Construcción del dashboard web (dashboard/src/main.js, vistas/PanelPrincipal.vue).
  + Visualización de resultados de escaneos, gestión de servicios, y módulo de aprendizaje interactivo.
  + Componentes reutilizables (componentes/BarraNavegacion.vue, TablaResultados.vue).

**Tailwind CSS**

* **Función General**: Framework CSS basado en clases utilitarias para diseñar interfaces modernas y responsive rápidamente.
* **Función en PlatSentinel**:
  + Estilización del dashboard web (dashboard/tailwind.config.js).
  + Diseño responsive para pantallas móviles y de escritorio, asegurando accesibilidad para usuarios no técnicos.

**FastAPI**

* **Función General**: Framework Python de alto rendimiento para construir APIs REST rápidas, con soporte para validación automática y documentación OpenAPI.
* **Función en PlatSentinel**:
  + Implementación de la API central (codigo\_fuente/api/app.py, rutas.py).
  + Gestión de endpoints para escaneos, informes, y servicios externos (/scans, /reports, /services).
  + Integración con autenticación JWT (autenticacion.py).

**Axios**

* **Función General**: Cliente HTTP basado en promesas para realizar solicitudes a APIs desde el frontend.
* **Función en PlatSentinel**:
  + Conexión del frontend con la API REST (dashboard/src/utilidades/api.js).
  + Envío de solicitudes para iniciar escaneos, obtener informes, y gestionar servicios.

**2. Herramientas de Contenerización y Despliegue**

**Docker**

* **Función General**: Plataforma para contenerizar aplicaciones, garantizando aislamiento, portabilidad y escalabilidad.
* **Función en PlatSentinel**:
  + Contenerización de todos los módulos: API (api/Dockerfile), toquens (tokens/toquen\_antesala/Dockerfile), dashboard (dashboard/Dockerfile), y bot de WhatsApp (bot\_whatsapp/Dockerfile).
  + Orquestación de servicios mediante docker-compose.yml para desarrollo y pruebas locales.
  + Aislamiento de toquens especializados (toquen\_pdf/, toquen\_web/) para análisis seguro.

**Docker Compose**

* **Función General**: Herramienta para definir y gestionar aplicaciones multi-contenedor con configuraciones YAML.
* **Función en PlatSentinel**:
  + Orquestación de servicios (API, toquens, dashboard, base de datos) en entornos locales (docker/docker-compose.yml).
  + Definición de dependencias entre módulos (e.g., API depende de toquens y base de datos).

**Kubernetes (k3s)**

* **Función General**: Sistema de orquestación para gestionar contenedores en entornos de producción, con escalabilidad y tolerancia a fallos.
* **Función en PlatSentinel**:
  + Despliegue en producción de la plataforma (infraestructura/kubernetes/).
  + Uso de Helm charts (k3s\_valores\_helm/) para configurar entornos (desarrollo, pruebas, producción).
  + Gestión de certificados y gateways (manifests\_basicos/).

**Terraform**

* **Función General**: Herramienta de infraestructura como código (IaC) para provisionar recursos cloud de forma automatizada.
* **Función en PlatSentinel**:
  + Configuración de infraestructura cloud (AWS, Azure) en infraestructura/terraform/.
  + Creación de recursos como buckets S3 (S3\_BUCKET=sentinel-artefactos) y clústeres Kubernetes.

**3. Herramientas de Base de Datos y Mensajería**

**PostgreSQL**

* **Función General**: Sistema de gestión de bases de datos relacional robusto, escalable y de código abierto.
* **Función en PlatSentinel**:
  + Almacenamiento de resultados de escaneos, configuraciones, y datos de informes (CONFIGURACION\_BD en variables\_globales.py).
  + Soporte para consultas estructuradas en el módulo de base de datos (codigo\_fuente/api/modelos.py).

**Redis**

* **Función General**: Base de datos en memoria para colas de mensajes y almacenamiento de datos de alta velocidad.
* **Función en PlatSentinel**:
  + Gestión de colas para comunicación entre módulos (CONFIGURACION\_MENSAJERIA.redis\_host).
  + Coordinación asíncrona de toquens (tokens/toquen\_antesala/escaneo.py).

**RabbitMQ**

* **Función General**: Sistema de mensajería para colas de mensajes, ideal para aplicaciones distribuidas.
* **Función en PlatSentinel**:
  + Orquestación de tareas entre toquens y módulos (CONFIGURACION\_MENSAJERIA.rabbit\_url).
  + Envío de tareas de escaneo y notificaciones (orquestador.py).

**4. Herramientas de Seguridad y Escaneo**

**OWASP ZAP**

* **Función General**: Herramienta de código abierto para pruebas de seguridad de aplicaciones web, detectando vulnerabilidades como XSS y SQLi.
* **Función en PlatSentinel**:
  + Escaneo de aplicaciones web en el módulo toquen\_web/escaneo\_web.py.
  + Integración vía API (CLAVES\_API\_EXTERNAS.owasp\_zap) para resultados automatizados.

**Wapiti**

* **Función General**: Escáner de vulnerabilidades web ligero, enfocado en pruebas de seguridad de aplicaciones.
* **Función en PlatSentinel**:
  + Complementa OWASP ZAP en el módulo toquen\_web/escaneo\_web.py para análisis adicionales.
  + Procesa URLs enviadas desde el dashboard o WhatsApp.

**ClamAV**

* **Función General**: Antivirus de código abierto para detectar malware en archivos.
* **Función en PlatSentinel**:
  + Sanitización de archivos en toquen\_pdf/sanitizer.py y toquen\_antesala/escaneo.py.
  + Integración vía API (CLAVES\_API\_EXTERNAS.clamav) para análisis en contenedores Docker.

**VirusTotal**

* **Función General**: Plataforma para análisis de archivos y URLs mediante múltiples motores antivirus.
* **Función en PlatSentinel**:
  + Escaneo de archivos y URLs en tokens/partners/virustotal/ como servicio gestionado.
  + Resultados integrados en el dashboard (mapeo\_servicios.py).

**Trivy**

* **Función General**: Escáner de seguridad para imágenes Docker y dependencias, detectando vulnerabilidades conocidas.
* **Función en PlatSentinel**:
  + Escaneo de imágenes Docker en el pipeline CI/CD (.github/workflows/escaneo-seguridad.yml).
  + Garantiza que los contenedores (api/Dockerfile, tokens/\*/Dockerfile) sean seguros.

**Semgrep**

* **Función General**: Herramienta de análisis estático para identificar problemas de seguridad en el código.
* **Función en PlatSentinel**:
  + Análisis de código Python (api/, tokens/) en el pipeline CI/CD (.github/workflows/escaneo-seguridad.yml).
  + Detección de vulnerabilidades en desarrollo temprano.

**5. Herramientas de Integración y Comunicación**

**WhatsApp Business API (Twilio)**

* **Función General**: Plataforma para enviar y recibir mensajes de WhatsApp, con soporte para bots y automatización.
* **Función en PlatSentinel**:
  + Notificaciones de vulnerabilidades y comandos (bot\_whatsapp/main.py, API\_WHATSAPP\_CREDENCIALES).
  + Interacción con usuarios y técnicos (/start\_scan, /generate\_report) desde el dashboard o móvil.

**MinIO**

* **Función General**: Almacenamiento de objetos compatible con S3, ideal para gestionar archivos en la nube.
* **Función en PlatSentinel**:
  + Almacenamiento de informes generados (S3\_BUCKET=sentinel-artefactos en variables\_globales.py).
  + Gestión de archivos escaneados en toquen\_antesala/.

**6. Herramientas de Pruebas**

**pytest**

* **Función General**: Framework de pruebas para Python, usado para pruebas unitarias e integración.
* **Función en PlatSentinel**:
  + Pruebas unitarias (pruebas/test\_api.py, test\_tokens.py).
  + Pruebas de integración para módulos (api/, tokens/, integracion/).

**Postman**

* **Función General**: Plataforma para probar APIs, diseñando y ejecutando colecciones de pruebas.
* **Función en PlatSentinel**:
  + Pruebas de endpoints REST (pruebas/postman/api\_tests.json).
  + Validación de respuestas de la API (/scans, /reports).

**Selenium**

* **Función General**: Herramienta para automatización de pruebas de interfaces web.
* **Función en PlatSentinel**:
  + Pruebas de usabilidad del dashboard (pruebas/selenium/).
  + Simulación de interacciones de usuario (iniciar escaneos, generar informes).

**k6**

* **Función General**: Herramienta para pruebas de carga y rendimiento en aplicaciones web y APIs.
* **Función en PlatSentinel**:
  + Pruebas de carga en la API y dashboard (pruebas/carga\_k6/).
  + Evaluación de escalabilidad bajo alta demanda.

**7. Herramientas de CI/CD y Gestión de Código**

**Git**

* **Función General**: Sistema de control de versiones para gestionar código fuente.
* **Función en PlatSentinel**:
  + Repositorio en GitHub para control de versiones (PlatSentinel/.git).
  + Gestión de ramas y pull requests para colaboración del equipo.

**GitHub Actions**

* **Función General**: Plataforma de CI/CD para automatizar flujos de trabajo (build, test, deploy).
* **Función en PlatSentinel**:
  + Pipelines para construcción, pruebas, y escaneos de seguridad (.github/workflows/ci-cd.yml, escaneo-seguridad.yml).
  + Firma de imágenes Docker con Cosign.

**Dependabot**

* **Función General**: Herramienta para actualizar dependencias automáticamente, detectando vulnerabilidades.
* **Función en PlatSentinel**:
  + Actualización de dependencias Python (requirements.txt) y Node.js (package.json).
  + Integrado en el pipeline CI/CD (.github/workflows/).

**8. Herramientas de Documentación**

**Markdown**

* **Función General**: Lenguaje ligero para documentación estructurada y legible.
* **Función en PlatSentinel**:
  + Creación de documentación técnica y de usuario (documentacion/manual\_usuario.md, arquitectura\_modular.md).
  + Registro de decisiones técnicas (AADR/adr-0001-unificacion-ui.md).

**OpenAPI/Swagger**

* **Función General**: Estándar para documentar APIs REST, generando especificaciones interactivas.
* **Función en PlatSentinel**:
  + Documentación automática de la API (documentacion/api\_specs/).
  + Facilita la integración de desarrolladores y partners.

**9. Herramientas de Infraestructura Adicional**

**Helm**

* **Función General**: Gestor de paquetes para Kubernetes, simplificando despliegues complejos.
* **Función en PlatSentinel**:
  + Configuración de entornos Kubernetes (infraestructura/kubernetes/k3s\_valores\_helm/).
  + Despliegue de módulos con valores personalizados por entorno.

**Chaos Mesh**

* **Función General**: Herramienta para pruebas de caos en entornos Kubernetes, simulando fallos.
* **Función en PlatSentinel**:
  + Pruebas de resiliencia (infraestructura/scripts/pruebas\_chaosmesh.sh).
  + Validación de tolerancia a fallos en producción.

**10. Herramientas de Monitoreo (Futuras)**

**Prometheus (Opcional)**

* **Función General**: Sistema de monitoreo y alertas para métricas en tiempo real.
* **Función en PlatSentinel**:
  + Futura integración para monitorear rendimiento de la API y toquens (infraestructura/observabilidad/).
  + Seguimiento de tiempos de respuesta y uso de recursos.

**Grafana (Opcional)**

* **Función General**: Plataforma para visualización de métricas y dashboards interactivos.
* **Función en PlatSentinel**:
  + Visualización futura de métricas de escaneos y servicios (infraestructura/observabilidad/).
  + Paneles personalizados para técnicos y PYMEs.

**Resumen de Herramientas**

| **Categoría** | **Herramientas** |
| --- | --- |
| **Desarrollo** | Python, Node.js, Vue.js, Tailwind CSS, FastAPI, Axios |
| **Contenerización/Despliegue** | Docker, Docker Compose, Kubernetes (k3s), Terraform |
| **Base de Datos/Mensajería** | PostgreSQL, Redis, RabbitMQ |
| **Seguridad/Escaneo** | OWASP ZAP, Wapiti, ClamAV, VirusTotal, Trivy, Semgrep |
| **Integración/Comunicación** | WhatsApp Business API (Twilio), MinIO |
| **Pruebas** | pytest, Postman, Selenium, k6 |
| **CI/CD** | Git, GitHub Actions, Dependabot |
| **Documentación** | Markdown, OpenAPI/Swagger |
| **Infraestructura Adicional** | Helm, Chaos Mesh |
| **Monitoreo (Futuro)** | Prometheus, Grafana |

**Notas Adicionales**

* **Modularidad**: Las herramientas de seguridad (OWASP ZAP, ClamAV) y toquens están contenerizadas para aislamiento, asegurando que un fallo en un toquen no afecte al sistema.
* **Escalabilidad**: Docker, Kubernetes, y Terraform permiten escalar la plataforma en entornos cloud (AWS, Azure).
* **Accesibilidad**: La integración con WhatsApp (Twilio) y el dashboard Vue.js facilitan el uso por PYMEs y técnicos no especializados.
* **Evolución**: La estructura está preparada para añadir herramientas de observabilidad (Prometheus, Grafana) y nuevos toquens (tokens/partners/).
* **Seguridad**: Trivy, Semgrep, y autenticación JWT (PARAMETROS\_SEGURIDAD) garantizan un desarrollo seguro.

Esta lista asegura que PlatSentinel pueda desarrollarse, desplegarse y mantenerse de manera eficiente, cumpliendo con los objetivos de ser una solución SaaS modular, proactiva y accesible.