

# SECURITY SCAN IOT

SISTEMA DE BÚSQUEDA Y  
ANÁLISIS DE SEGURIDAD DE  
DISPOSITIVOS IOT



Mike Encalada

**01**

# **PROBLEMÁTICA**

# PROBLEMÁTICA

01

## **Crecimiento Exponencial**

Cámaras de seguridad  
Luces inteligentes

02

## **Funcionalidades específicas**

Recursos de hardware y software  
limitados.

03

## **Pocas medidas de seguridad**

Sin prioridad en las medidas de  
seguridad durante su desarrollo.

04

## **Riesgos de seguridad**

Ataques cibernéticos  
Exposición de datos sensibles

**02**

# **OBJETIVOS Y ALCANCE**

# OBJETIVOS



## Desarrollo

Desarrollar el componente encargado de realizar la **recolección, gestión y procesamiento** de la información recopilada durante la búsqueda de los dispositivos IoT y el **análisis** de sus potenciales **vulnerabilidades**.



## Seguridad

Implementar **medidas de seguridad** con el fin de proteger información sensible y prevenir posibles amenazas que puedan afectar a la privacidad del usuario, garantizando la integridad, consistencia y seguridad de los datos.



## Comunicación

Establecer una **comunicación efectiva** con el componente Frontend para presentar de manera más organizada y legible los datos procesados dentro de un dashboard web.

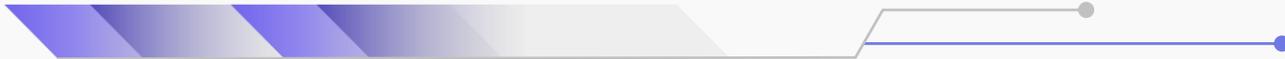
# ALCANCE

El desarrollo del componente Backend abarcará desde la recopilación y análisis de requisitos hasta la implementación final, garantizando escalabilidad y robustez. Se utilizarán

**herramientas modernas** como IDEs, control de versiones y pruebas automatizadas.

El diseño incluirá **el modelo C4** para representar de forma clara la arquitectura del sistema y facilitar la comunicación con los interesados.

Al concluir, se entregará un **sistema funcional** de búsqueda y análisis de seguridad para dispositivos IoT, junto con la **documentación** completa del ciclo de desarrollo.

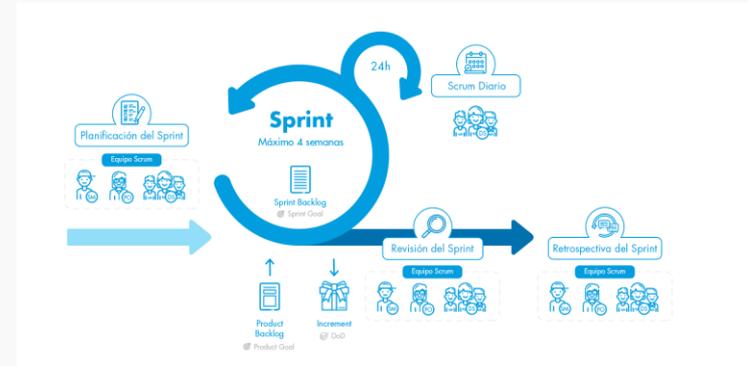


**03**

# **METODOLOGÍA**

# MARCO DE TRABAJO SCRUM

- Familiaridad -> Experiencia
- Entregas Continuas -> Feedback
- Control de trabajo -> Reuniones



## ROLES

- Scrum Master
- Product Owner
- Equipo Scrum

## ARTEFACTOS

- Product Backlog
- Sprint Backlog
- Incremento

## EVENTOS

- Sprints





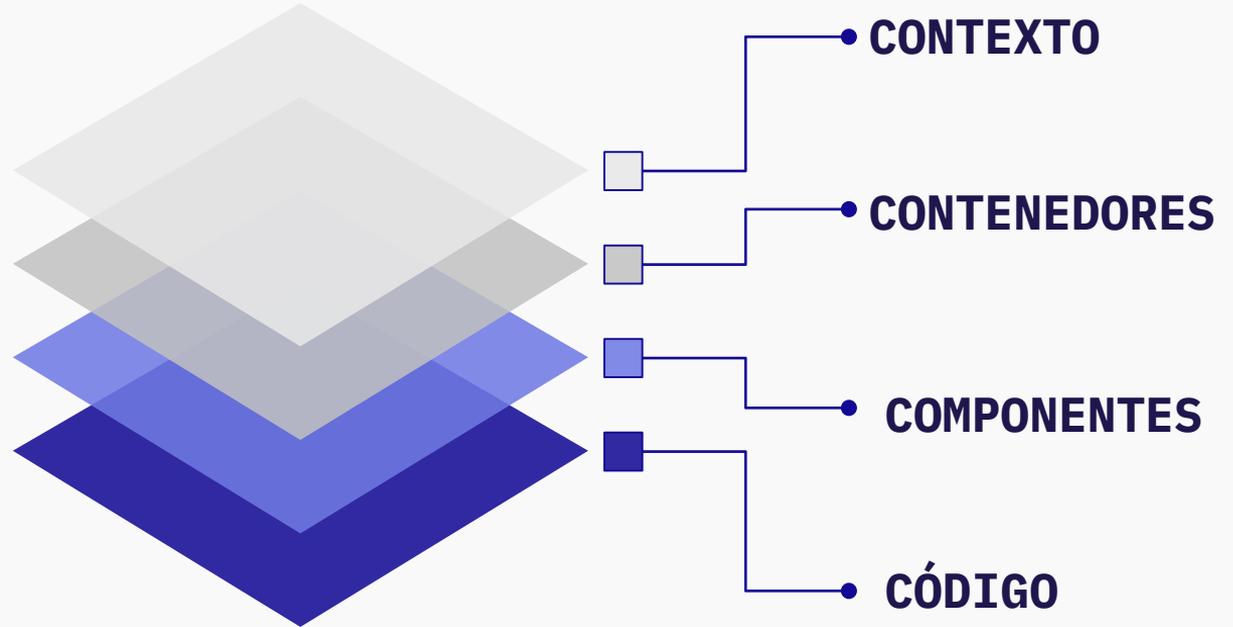
**MODELO C4**

# MODELO C4

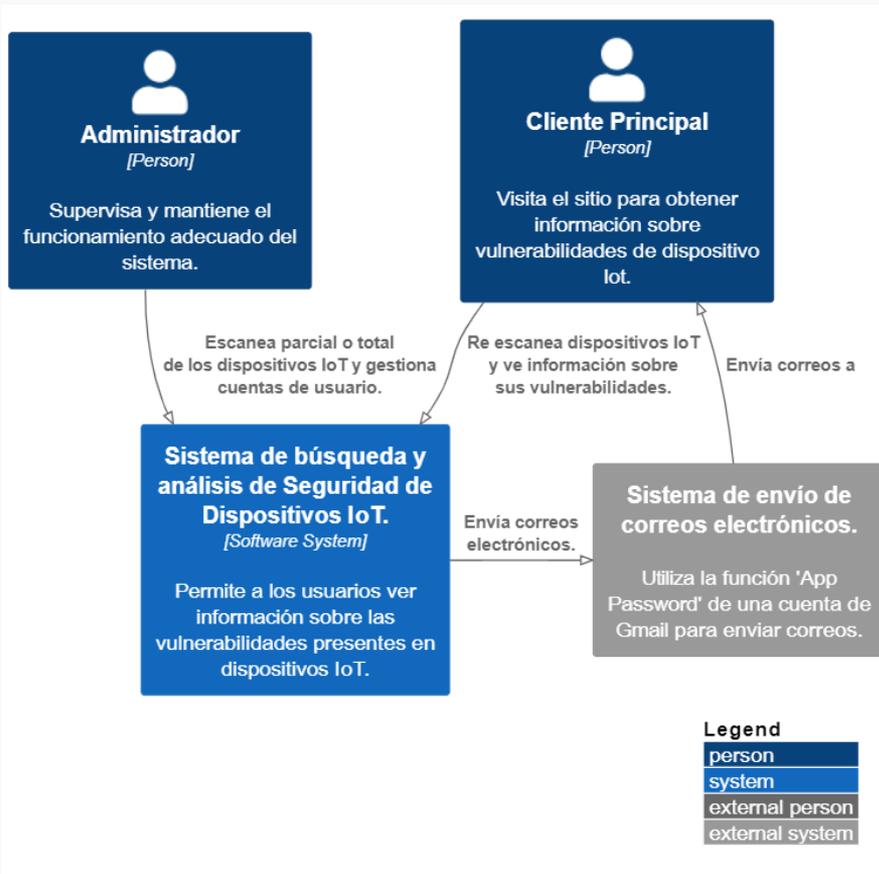
MÁS GENERAL



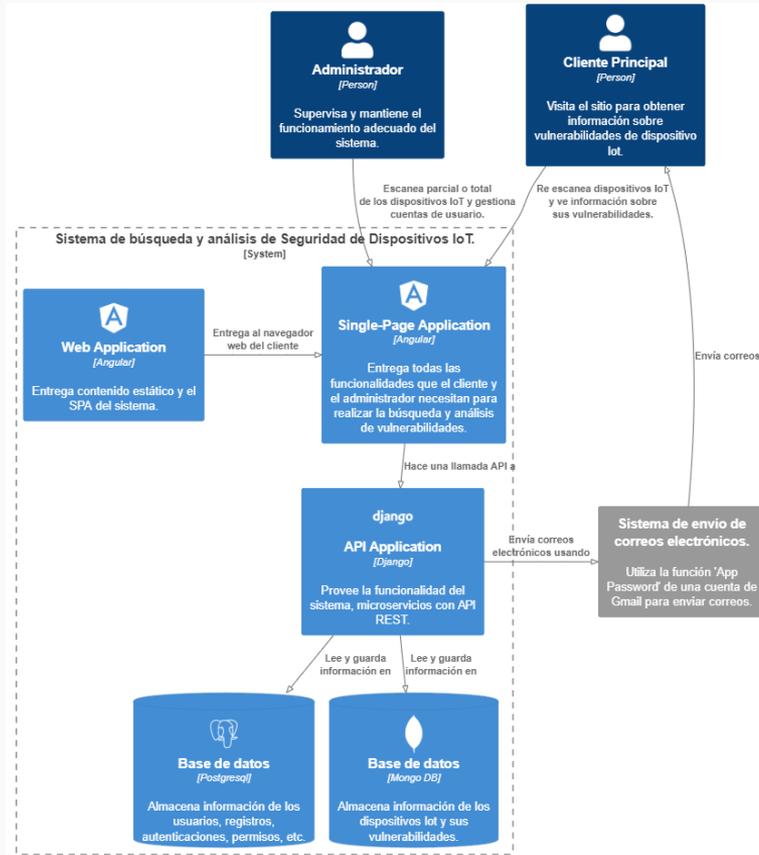
MÁS PARTICULAR



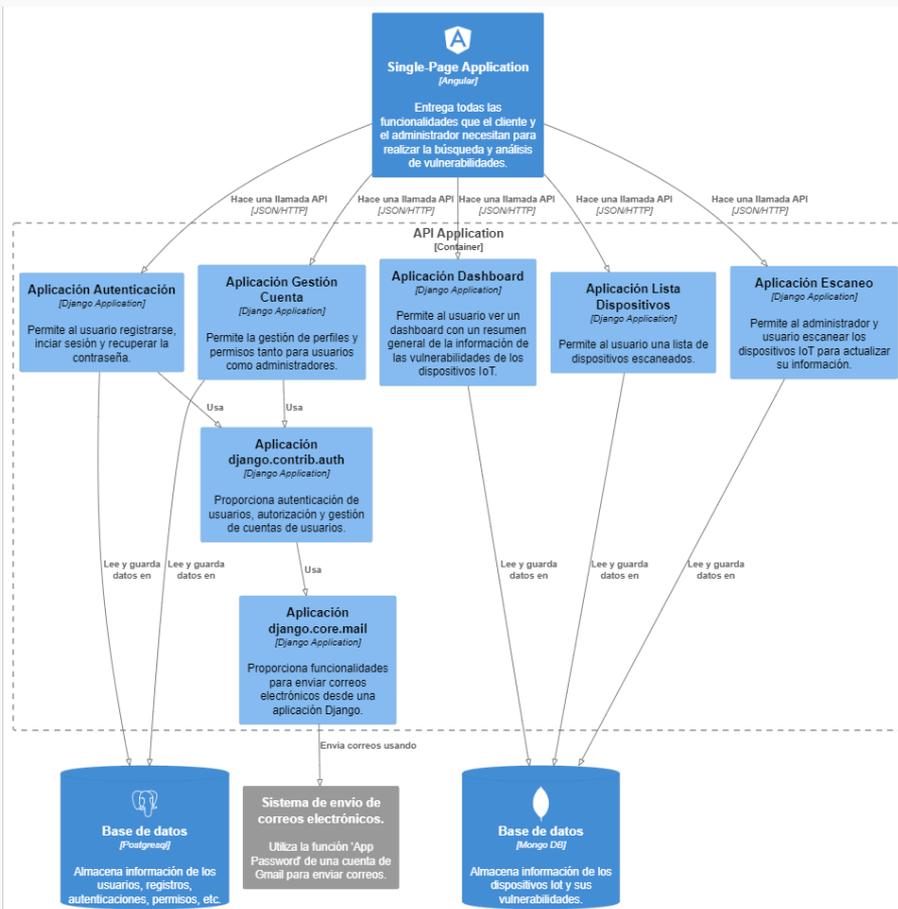
# DIAGRAMA DE CONTEXTO



# DIAGRAMA DE CONTENEDORES

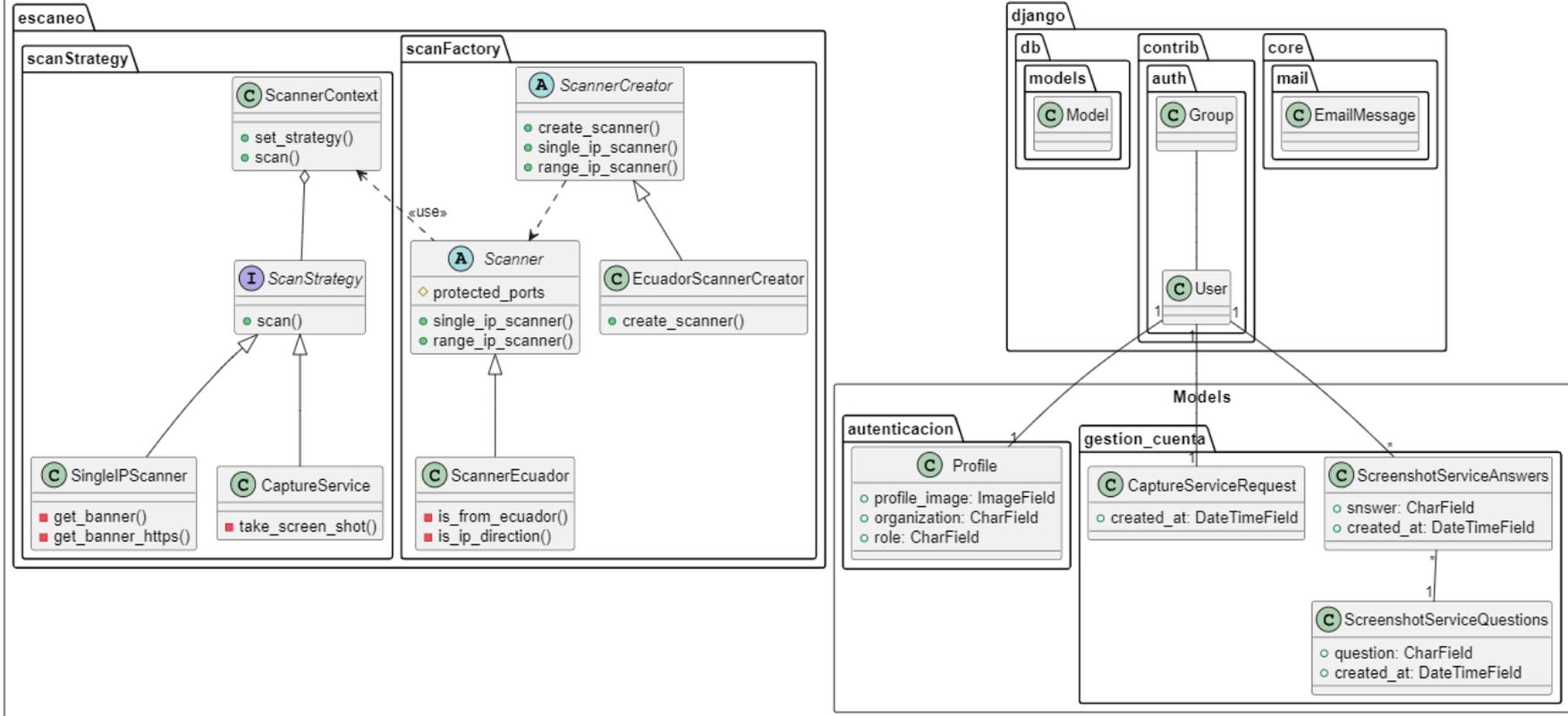


# DIAGRAMA DE COMPONENTES



# DIAGRAMA DE CÓDIGO

API Application



**04**

# **PROCESO**

# PROCESO

JIRA SOFTWARE

PLANNING  
POKER

MOCKUPS

## Product Backlog

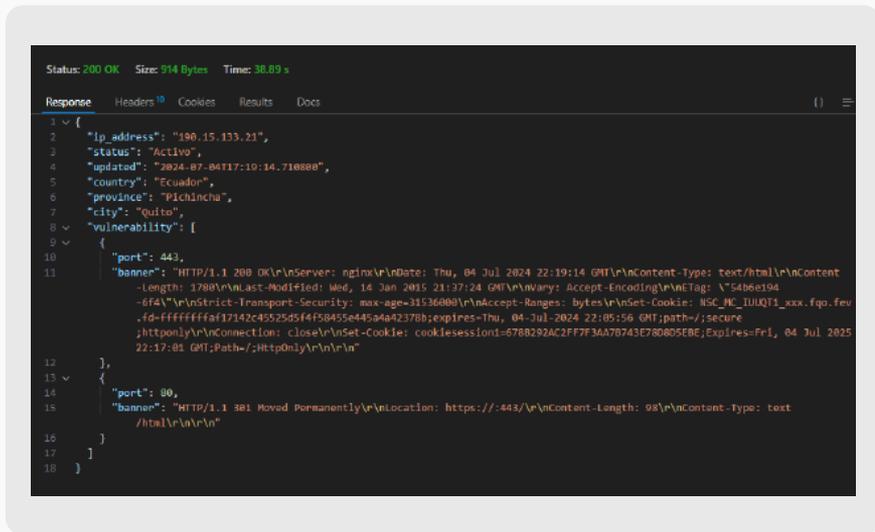
<u>Código</u>	<u>Nombre</u>	<u>Sprint</u>	<u>Estado</u>	<u>Prioridad</u>
HU001	Registrar e iniciar sesión	4	DONE	MEDIA
HU002	Visualizar mi perfil	5	DONE	BAJA
HU003	Gestionar cuentas	5	DONE	BAJA
HU004	Visualizar mapa de calor de direcciones IP y puertos	2	DONE	ALTA
HU005	Escanear completa o parcialmente las direcciones IP	1	DONE	ALTA
HU006	Re escaneo de un dispositivo	3	DONE	MEDIA
HU007	Mostrar historial de escaneos	3	DONE	MEDIA
HU008	Visualizar dashboard	2	DONE	ALTA
HU009	Mostrar captura de servicios web	4	DONE	MEDIA
HU0010	Mostrar información detallada de vulnerabilidades	1	DONE	ALTA

**05**

# **RESULTADOS**

# RESULTADOS

Se crearon en total 24 servicios API RESTful.



# PRUEBAS



## FUNCIONALIDAD

### Postman

Los endpoints de los servicios API RESTful creados funcionan correctamente.



## RENDIMIENTO

### Jmeter

Varias solicitudes manejadas sin errores.  
Desviación estándar alta.

**06**

# **CONCLUSIONES**

# CONCLUSIONES



## DESARROLLO

El componente back end del sistema ha sido desarrollado con éxito y con los resultados esperados.



## COMUNICACIÓN EFECTIVA

La comunicación efectiva con el front end se garantizó mediante rigurosas pruebas de funcionalidad y rendimiento.

# CONCLUSIONES



## PRUEBAS

Endpoint funcionan correctamente.

Manejo de varias solicitudes sin errores.

Puede existir mejoras en cuestión de rendimiento.



## SEGURIDAD

La elección de Django como framework principal ha contribuido significativamente a este objetivo.



**GRACIAS !**