

Grimoirebots

Sistema SaaS para el análisis de proyectos software

Sergio Merino Hernández



Universidad
Rey Juan Carlos

Máster Universitario en Ingeniería de Telecomunicación

28 de junio de 2023

Presentation Overview

1 Introducción

- Motivación
- Objetivos

2 Desarrollo del Proyecto

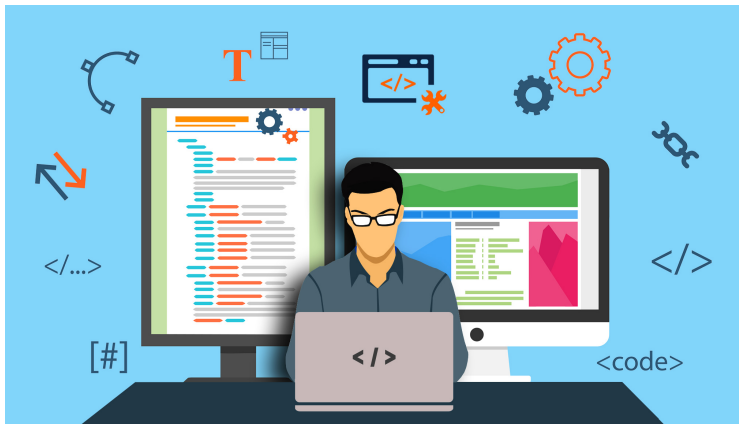
- Análisis de Cauldron
- Arquitectura del software

3 Ejemplo de uso

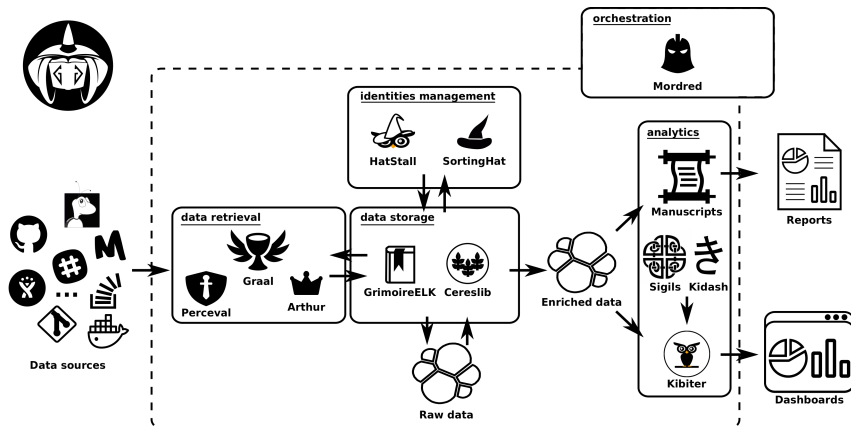
4 Conclusiones y trabajos futuros

- Conclusiones
- Líneas de desarrollo futuras

Introducción



GrimoireLab



Motivación



Objetivos

Desarrollo de un **servicio web** similar a Cauldron, que permita el análisis de ecosistemas de desarrollo de *software*, modificando algunos de sus principios básicos.

Subobjetivos:

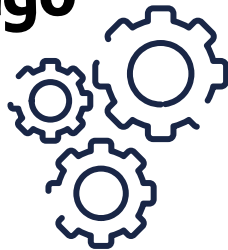
- Desarrollar un sistema que genere informes inalterables
- Desarrollar múltiples componentes *software* independientes que colaboren entre ellos para dar un servicio al usuario
- Simplificar el uso de GrimoireLab mediante la ejecución de su contenedor Docker original
- Realizar varias mejoras menores respecto a Cauldron

Análisis de Cauldron

Características

- Facilitar el uso de GrimoireLab
- Django + Elasticsearch
- Git, GitHub, Meetup, etc
- Informes alterables
- Sistema de *workers*
- Métricas y gráficas

django



Open Distro

Análisis de Cauldron

Puntos de mejora

- Múltiples iteraciones en los modelos
- *Cauldron Pool Scheduler*
- Manejo de gran cantidad de datos
- Difícil modificación de gráficas



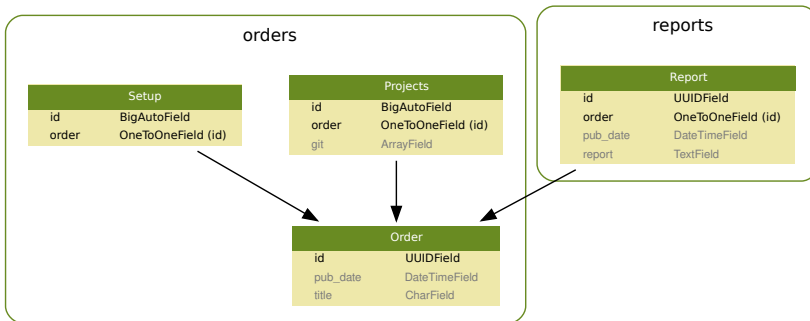
Definición y funcionalidades

Grimoirebots es un complejo sistema de análisis de repositorios git que, de forma automática, obtiene y genera datos sobre el desarrollo de *software*.

Principales características

- API
- OpenSearch Dashboards
- Cliente
- Herramientas de despliegue

Modelos de datos



Descripción de la API

```
{  
  "title": "string",  
  "projects": {  
    "git": [  
      "string"  
    ]  
  },  
  "setup": {}  
}
```

orders

GET	/orders/	▼
POST	/orders/	▼
GET	/orders/pending/	▼
GET	/orders/{id}/	▼
PUT	/orders/{id}/	▼
PATCH	/orders/{id}/	▼
DELETE	/orders/{id}/	▼
GET	/orders/{id}/projects.json/	▼
GET	/orders/{id}/setup.cfg/	▼

Descripción de la API

```
{  
  "order": "string",  
  "report": "string"  
}
```

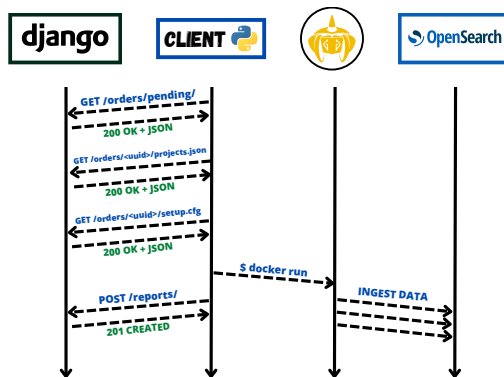
reports

GET	/reports/	▼
POST	/reports/	▼
GET	/reports/{id}/	▼
PUT	/reports/{id}/	▼
PATCH	/reports/{id}/	▼
DELETE	/reports/{id}/	▼

Flujo de ejecución del usuario



Flujo de ejecución del cliente



Ejemplo de uso

Conclusiones

- Sistema de análisis para repositorios *software*
- Componentes independientes
- Simplificación del uso de GrimoireLab
- **No logrado**: mejorar las prestaciones de Cauldron

Líneas de desarrollo futuras

Trabajos futuros

- Sistema de colas
- Creación de usuarios
- Plataformas Cloud
- Interfaz de usuario
- CI / CD

Código utilizado

- `github.com/merinhunter/grimoirebots`
- `github.com/merinhunter/grimoirebots-client`
- `github.com/merinhunter/grimoirebots-deployment`

The End

¿Alguna pregunta?