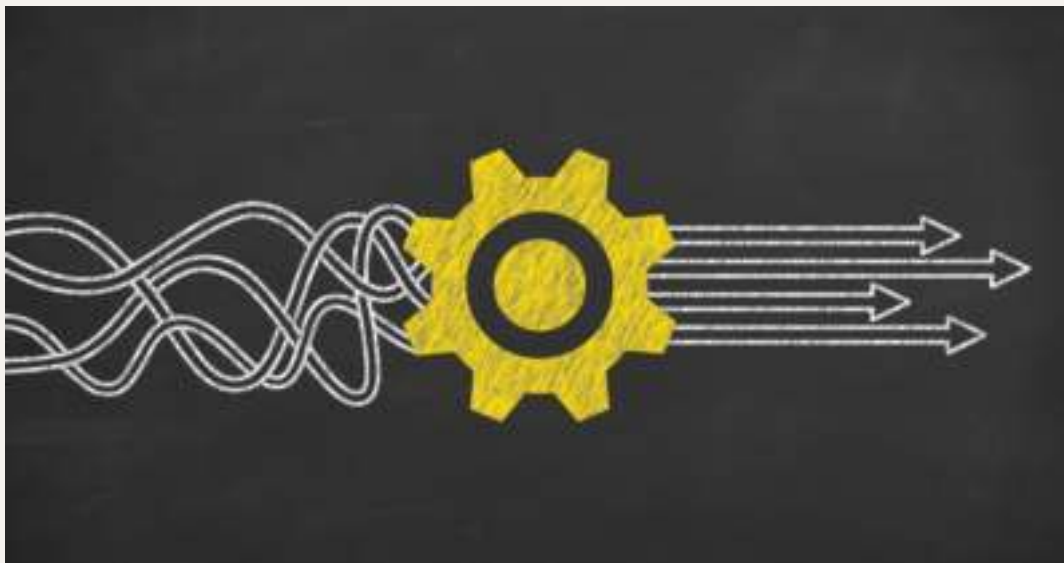


Monitoramento de aplicações em Python

Álvaro Ferreira Pires de Paiva



Alinhamento de expectativas



Quem sou eu na
fila do pão?



O que vem antes?

O que vem antes?

- Linter;
- Testes unitários;
- Garantir que não ocorra vazamento de credenciais;
- Etc.





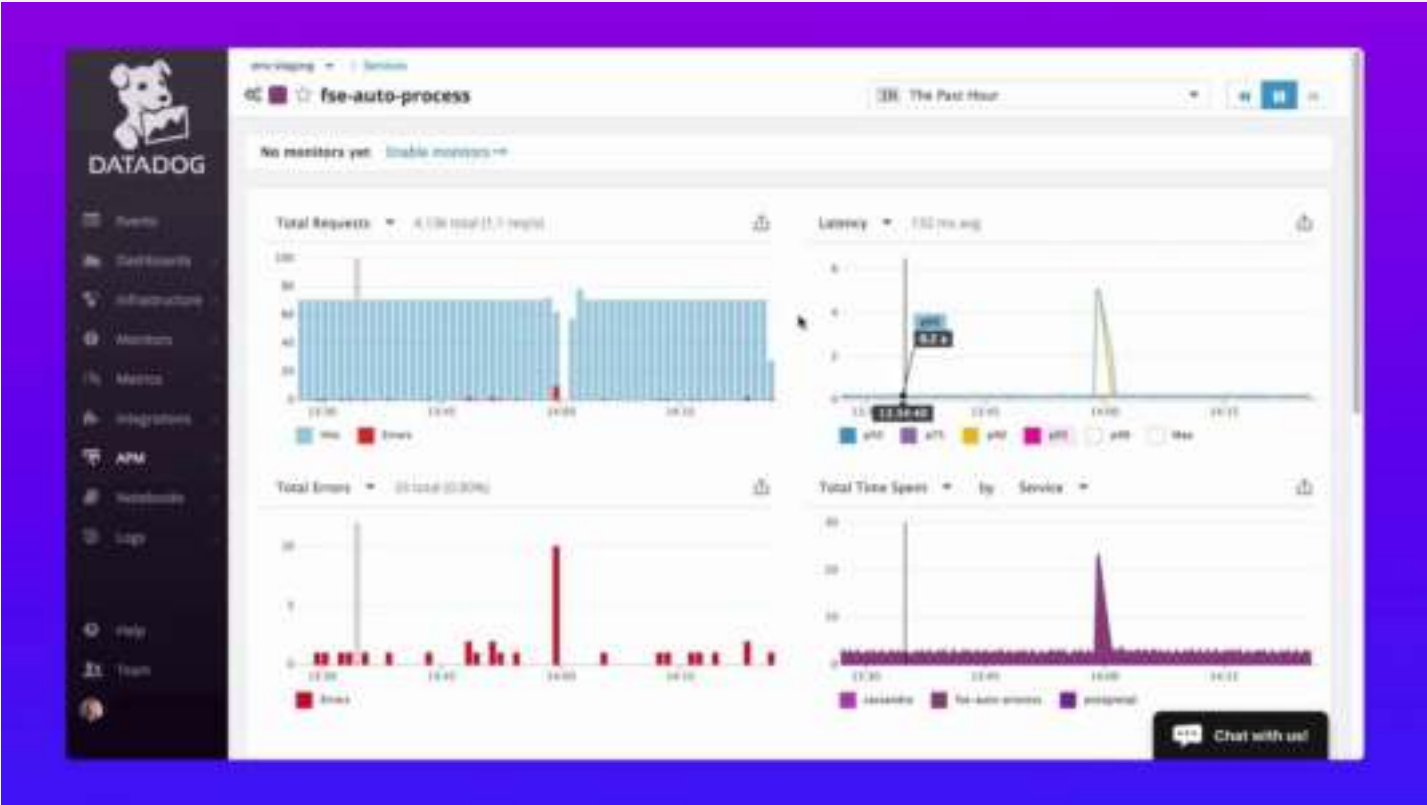
Monitoramento e Observabilidade

Logging

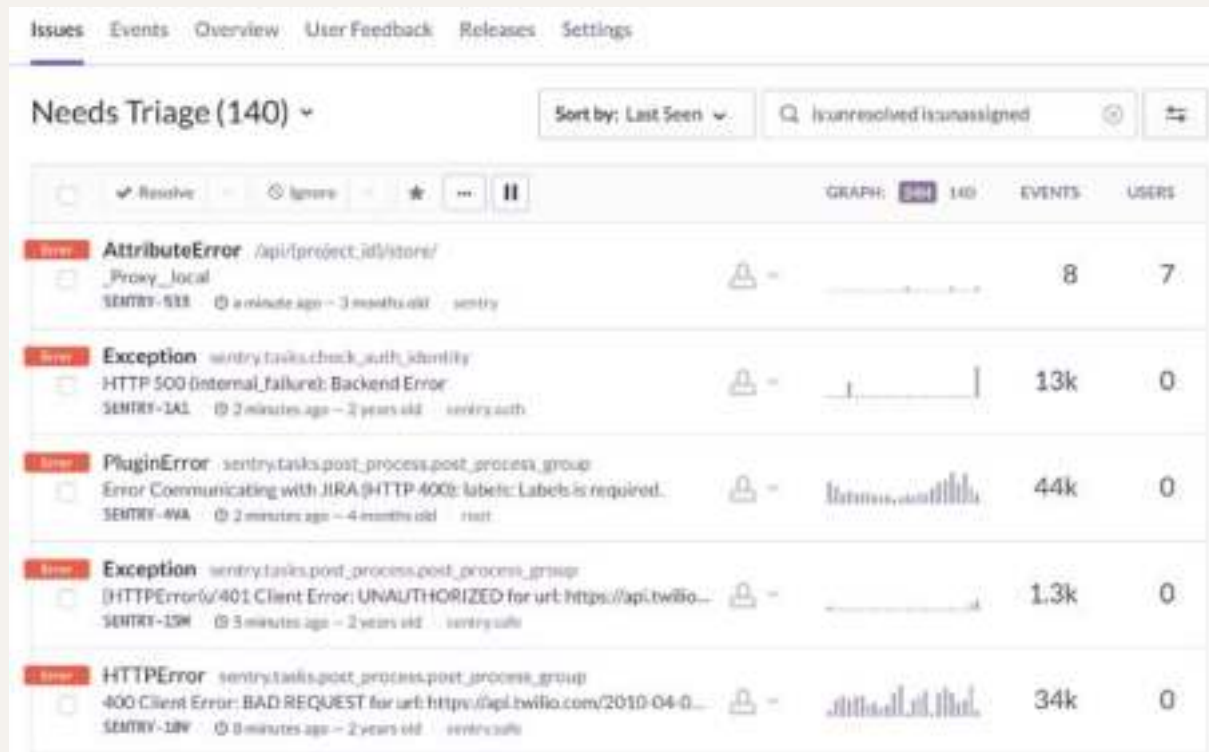
```
=> checking if the databases are recently dropped
/usr/bin/psql -U postgres -p 5432 -d db_info -t -c "
    SELECT max(drop_dt) FROM _dropped_timestamp
    WHERE db = 'lorem_ipsum'
    "
database lorem_ipsum is not dropped recently, so dropping
```

```
{
  "event": "FinancialTransaction",
  "type": "SUCCESS",
  "process_id": 1234,
  "timestamp": 1641438772,
  "data": {
    "sender": {
      "name": "michael"
    },
    "recipient": {
      "name": "andrea"
    },
    "transaction": {
      "value": "R$ 12.34"
    }
  }
}
```

Métricas e alertas



Rastreamento de erros



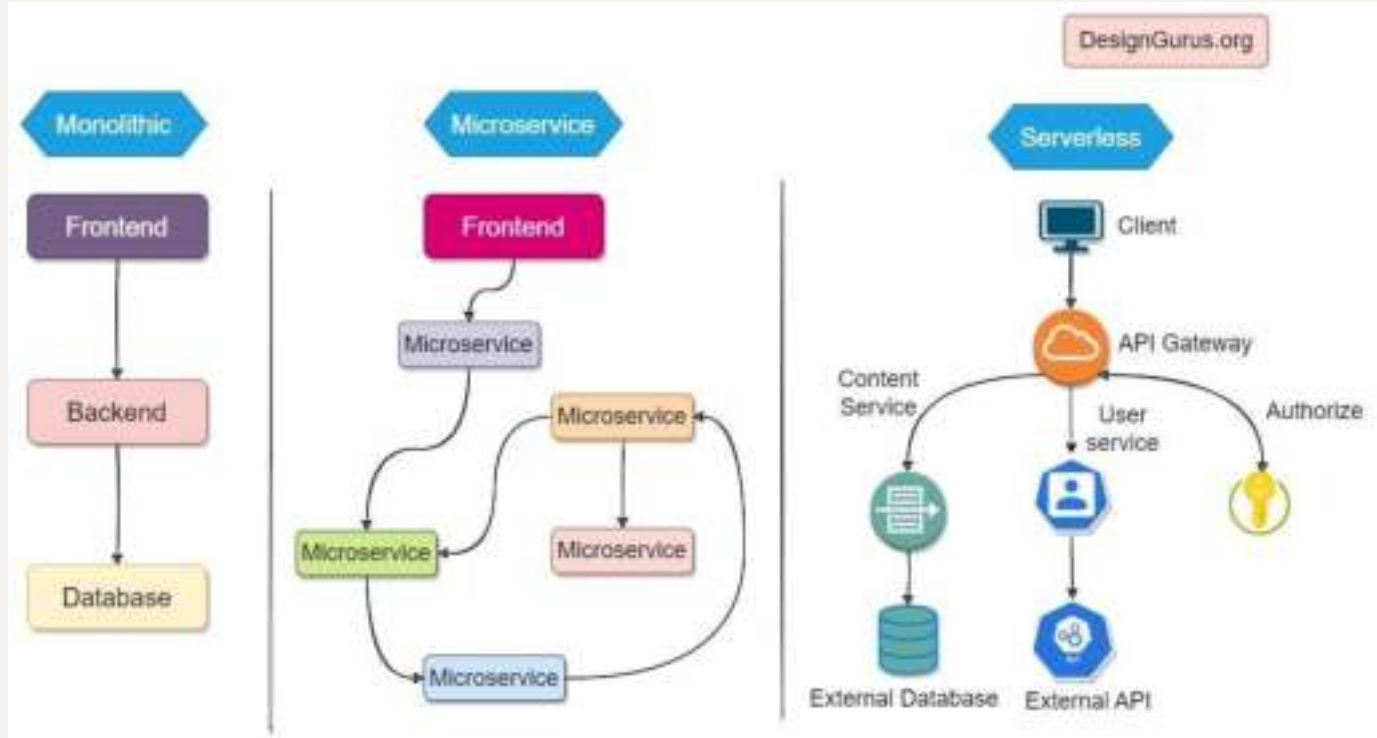
Performance





Implantação

Características da aplicação



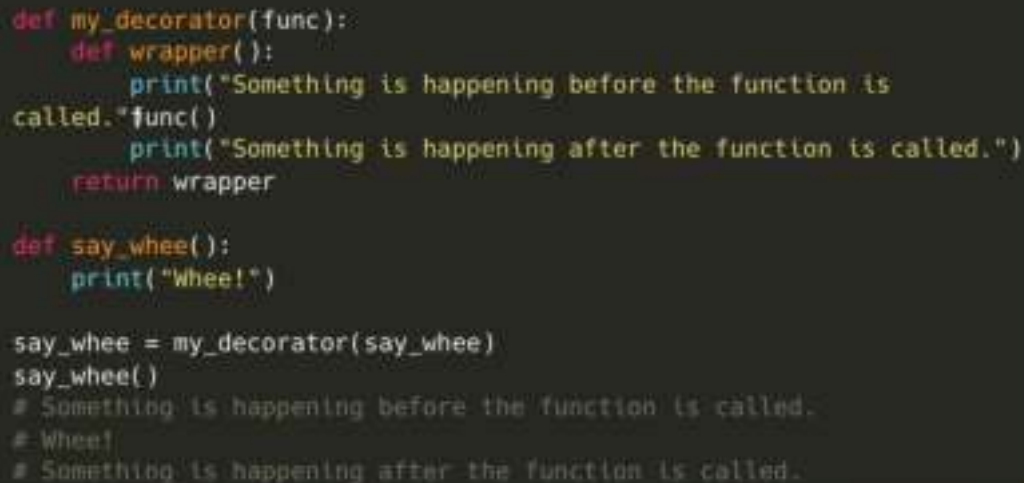
Do básico ao ideal

- Qualidade de código;
- Logging;
- Rastreamento de erros;
- Métricas;
- Alertas;
- Performance.



Metaprogramação

Códigos que criam, modificam ou instrumentam outros códigos.



```
def my_decorator(func):  
    def wrapper():  
        print("Something is happening before the function is  
called."func())  
        print("Something is happening after the function is called.")  
    return wrapper  
  
def say_whee():  
    print("Whee!")  
  
say_whee = my_decorator(say_whee)  
say_whee()  
# Something is happening before the function is called.  
# Whee!  
# Something is happening after the function is called.
```

Pacotes do Python

```
import logging

logging.warning('Watch out!')
# WARNING:root:Watch out!
```

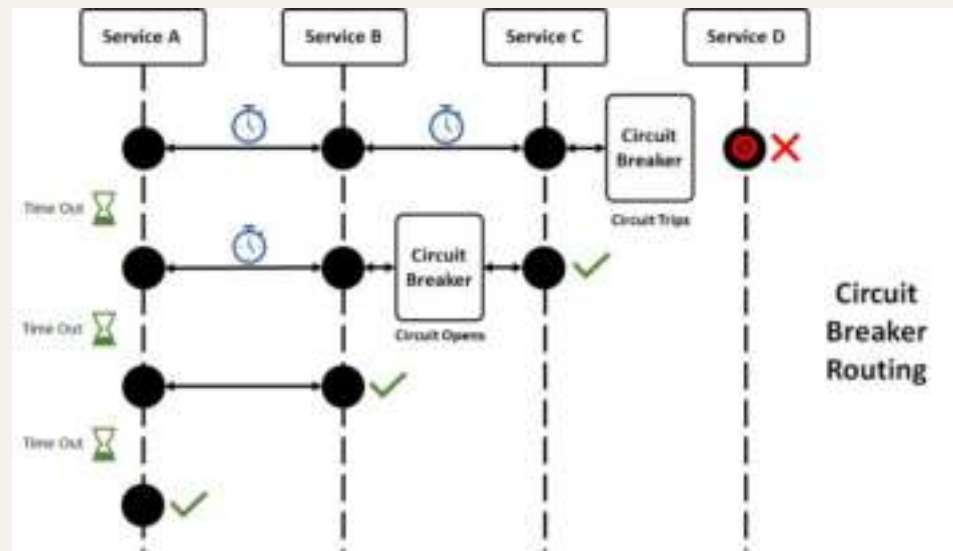
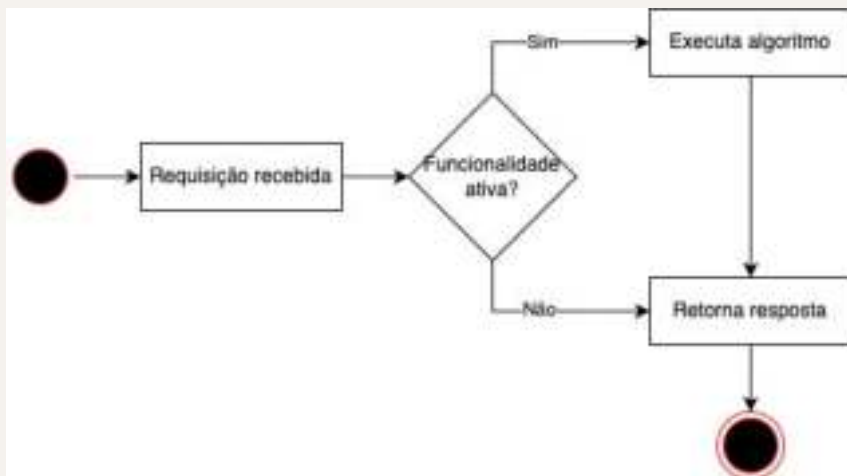
```
import traceback

def another_function():
    lumberstack()

def lumberstack():
    traceback.print_stack()
    print(repr(traceback.extract_stack()))
    print(repr(traceback.format_stack()))

another_function()
# File "<doctest>", line 11, in <module>
#   another_function()
# File "<doctest>", line 4, in another_function
#   lumberstack()
# File "<doctest>", line 7, in lumberstack
#   traceback.print_stack()
#[(' <doctest>', 11, '<module>', 'another_function()'),
# (' <doctest>', 4, 'another_function', 'lumberstack()'),
# (' <doctest>', 9, 'lumberstack', 'print(repr(traceback.extract_stack()))')]
#[' File "<doctest>", line 11, in <module>\n   another_function()\n',
# ' File "<doctest>", line 4, in another_function\n   lumberstack()\n',
# ' File "<doctest>", line 9, in lumberstack\n   print(repr(traceback.format_stack()))\n']
```

Padrões que podem ajudar



Ferramentas





Financeiro

Quanto custa?

- Permutação de métricas;
- Armazenamento de logs;
- Tempo de retenção de logs e métricas;
- Quantidade de alertas;
- Solução on-premise ou service;
- Agente dentro ou fora da aplicação;
- Etc.



Como convencer?



Dúvidas?

