



Ciências  
ULisboa

Informática

# Introdução à framework Flask

Pedro Ferreira, Vinicius Cogo

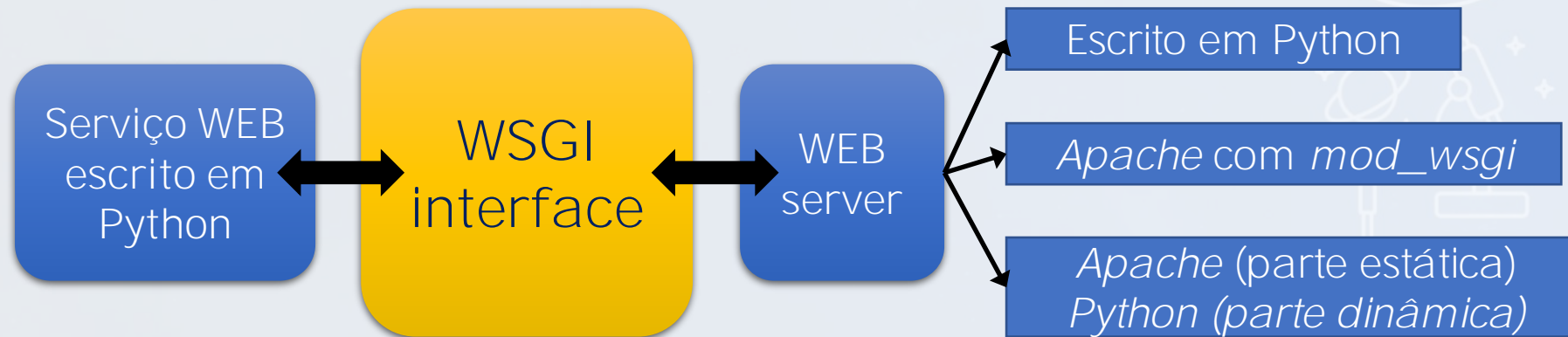
AD - TP06 | ©DI, Ciências, ULisboa



1. Servidores HTTP em Python
2. Framework Flask

# 1. Servidores de HTTP em Python

- Opções clássicas dependem de mecanismos diferentes de interface aplicação/servidor
- Forma padronizada introduzida no PEP<sup>1</sup> 3333:  
*Web Server Gateway Interface* (WSGI)



- <sup>1</sup>: *PEP: Python Enhancement Proposal*

## 2. Framework Flask para serviços WEB

- Flask é uma micro *web framework*
  - Baseada no *toolkit* WSGI *Werkzeug* e no motor de modelos (*template engine*) Jinja2;
  - Extensível para que possam adicionar funcionalidades existentes noutras bibliotecas;
  - Podem-se derivar classes da classe Flask;
  - Tem servidor e debugger incorporado;
  - Baseado em Unicode;
  - Fácil implementação de serviços RESTful;
  - Boa documentação.



## 2. Instalação do Flask na máquina virtual do DI

- Descarregar a máquina virtual equivalente aos labs do DI nas páginas da admin do DI ([admin.di.fc.ul.pt](http://admin.di.fc.ul.pt))
- Iniciar a máquina virtual
- Num terminal:

```
$ sudo apt-get update  
$ pip3 install Flask  
$ exit
```



## 2. Framework Flask para serviços WEB

### Hello world em Flask

```
from flask import Flask
app = Flask(__name__)

@app.route('/') # Python decorator: que URL usa a função!
def hello_world():
    return 'Hello World!'

if __name__ == '__main__':
    app.run()
    # ou app.run(debug = True), para usar o debugger
```

Aceder a:

<http://localhost:5000>

<http://localhost:5000/sub>

## 2. Framework Flask para serviços WEB

`route(regra, **kwargs)`

- Regista a função associada a uma determinada regra de URL.
  - regra: string com a regra da URL
  - kwargs: argumentos *keyword = value*, ...
    - methods = [lista de strings com métodos HTTP a que se vai responder]  
(Por omissão route responde apenas a 'GET')

## 2. Framework Flask para serviços WEB

### Métodos HTTP para cada URL

```
from flask import Flask, request
app = Flask(__name__)

@app.route('/', methods = ['GET', 'POST'])
def hello_world():
    return 'Hello World!'

@app.route('/recurso', methods = ['GET', 'PUT', 'POST'])
def recurso():
    return "%s %s" % (request.method, request.url)

if __name__ == '__main__':
    app.run()
```



## 2. Framework Flask para serviços WEB

Testar código anterior: módulo *requests* para código cliente

```
import requests

r = requests.get('http://localhost:5000/')
print r.status_code
print r.content
print '***'

r = requests.post('http://localhost:5000/recurso',
                  data = {'key': 'value'})

print r.status_code
print r.content
print '***'

r = requests.put('http://localhost:5000/recurso',
                 data = {'key': 'value'})

print r.status_code
print r.content
```

## 2. Framework Flask para serviços WEB

`route(regra, **kwargs)`

- Especificação de partes variáveis na regra de URL
  - regra: string com a regra da URL, pode incluir partes variáveis dentro de `<>`:
    - `/recurso/<id>`
    - `/recurso/<int:id>`
  - As partes variáveis são passadas à função como argumentos *keyword = value*

## 2. Framework Flask para serviços WEB

### Exemplo com regra de URL variável

```
from flask import Flask, request
app = Flask(__name__)

@app.route('/recurso/<int:id>')
def recurso(id):
    return "Acedeu com %s ao recurso com id=%d" %
        (request.method, id)

if __name__ == '__main__':
    app.run(debug = True)
```

## 2. Framework Flask para serviços WEB

Testar código anterior: URLs com parte variável

```
import requests

r = requests.get('http://localhost:5000/recurso/25')
print r.status_code
print r.content
print '***'

r = requests.get('http://localhost:5000/recurso/quatro')
print r.status_code
print r.content
```

- Brandon Rhodes and John Goerzen. Foundations of Python Network Programming, second edition, Apress.
- Flask User's guide:  
<https://flask.palletsprojects.com/en/1.1.x/>
- Flask API:  
<https://flask.palletsprojects.com/en/1.1.x/api/>
- Módulo requests:  
<https://requests.readthedocs.io/en/master/>