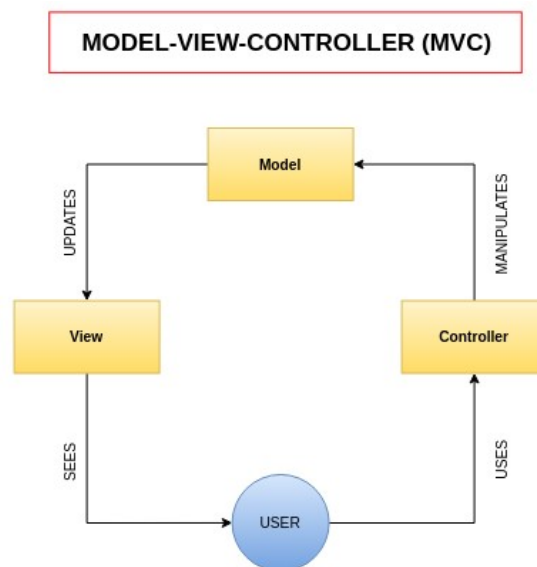


Exercícios Kabum

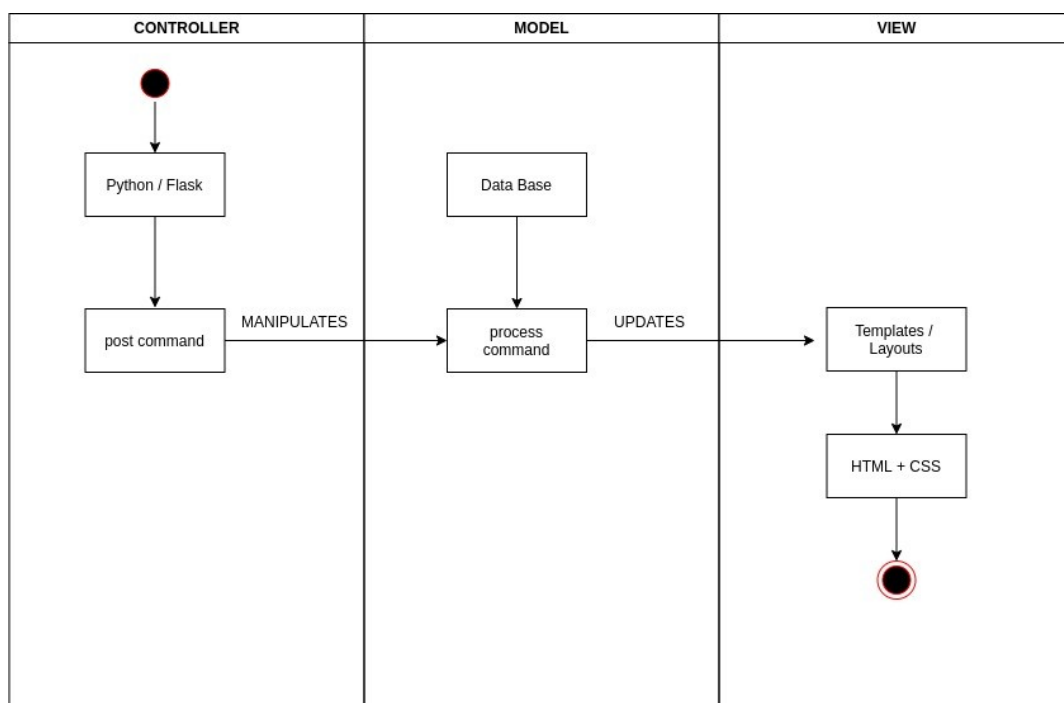
Nome: William Roberto de Araujo

27 de Julho de 2020

- 1) Utilizando www.draw.io, desenhe um esquemático em MVC (MODEL-VIEW-CONTROLLER) de uma aplicação flask básica.



- 2) Separar os níveis da aplicação em camadas arquiteturais.



3) Utilizar software design patterns.

Para desenvolvimento da aplicação foi utilizado a IDE Pycharm utilizando o PEP8.

4) Cite os principais eventos e signals durante um request, como acessá-los e um exemplo prático.

Sinais é uma maneira de interagir com a aplicação de maneira indireta, reagindo a eventos.

Os decoradores são recursos padrão da linguagem Python. Uma aplicação do uso dos decoradores é chamar uma função (callback) quando ocorrer um evento.

Por exemplo o decorador `@flask.route` responde a um requisição HTTP para uma VIEW.

5) Descreva o funcionamento do pacote **flask_restful**

Flask-RESTful é uma extensão do Flask para suporte a REST APIs. Possui a capacidade de gerar uma documentação das API utilizando Swagger.

6) Descreva o funcionamento do pacote **flask.views.MethodView**

O MethodView permite que cada método seja tratado por um método de classe diferente para separar responsabilidades:

```
from flask.views import MethodView

class UserView(MethodView):
    def get(self):
        ...
    def post(self):
        ...
    def put(self):
        ...
    def delete(self):
        ...
app.add_url_rule(
    '/user',
    view_func=UserView.as_view('user')
)
```

7) Utilizando os conceitos a cima, desenvolva uma aplicação básica em flask que extraia informações de um banco mocado (podes usar os JSONs dos produtos abaixo como mock):

https://github.com/WilliamAraujo/test_kabum/

O projeto está rodando em uma máquina virtual (EC2) da AWS utilizando um servidor HTTP Nginx.

Exemplo:

`http://18.223.44.44:5000/produto`

`http://18.223.44.44:5000/produto/109766`

Foram realizados testes no POSTMAN para validação.

GET	/produto
POST	/produto/{codigo}
PUT	/produto/{codigo}
GET	/produto/{codigo}
DEL	/produto/{codigo}