

1) Utilizando [www.draw.io](http://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.

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

- Signals é um forma de manipulação do python, onde ocorre implementações assíncronas reagindo a eventos.

Exemplo: Um serviço utiliza um a **request\_started** para abrir um arquivo em tela, porem ocorre um erro e demora mais que o tempo determinado, é disparado um alarme (**got\_request\_exception**) ou um evento informado o caso e no caso é aberto com sucesso obtemos a (**request\_finished**) e mantem a continuidade ao serviço.

5) Descreva o funcionamento do pacote **flask\_restful**

- É uma extensão do Flask. Facilidades para construir um REST APIs facilitando a criação dos métodos HTTP, ele atua juntamente com a ORM, facilitando a conversão de dados e a montagem das responses. Possui os benefícios de ser leve, fácil integração, documentação.

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

- São os métodos de retorno do serviço, faz o mapeando a partir do mesmo nome do controller com interação direta ao banco, uma exibição e tempo real. O metodo praticamente é adicionado dentro da requisição (url).

Exemplo: Site da g1 ele possui diversas rotas, porém a inicial fica em uma view e todas vez que ocorre uma mudança é chamada essa view, assim não precisando alterar a estrutura de código.

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

Referencia: <https://flask.palletsprojects.com/en/1.1.x/signals/>