

Engenharia de Software 2 - Trabalho Prático

Valor total: 20 pontos - Grupos de dois alunos

O objetivo deste trabalho é avançar o máximo possível na construção de uma aplicação baseada em uma arquitetura de microservices.

Para mais detalhes sobre microservices:

<https://martinfowler.com/articles/microservices.html>

O trabalho será realizado usando Heroku. Heroku (heroku.com) é um sistema para hospedar aplicações na Web ("cloud"). Por exemplo, ele é similar o Amazon Web Service (AWS). Porém, ele possui uma versão gratuita, com diversas limitações, que, no entanto, não devem impactar o nosso trabalho.

Parte 1: Desenvolver uma aplicação Heroku.

Tarefa: implementar uma aplicação bem simples e fazer o deploy no Heroku. Pode ser um Hello, World; como explicado neste tutorial:

<https://purplepalmdash.github.io/2014/05/09/write-python-weather-app-on-heroku/>

Sugestão: implementar em Python/Flask (fortemente sugerido; não obrigatório). Claro, existem na Web diversos tutoriais sobre Heroku com Python etc; bem como vídeos no YouTube.

Objetivo: dominar deploy no Heroku, integração com git, configuração de Python/Flask etc.

===

Parte 2: Implementar uma segunda aplicação, que faça uso de um banco de dados. Com isso, ela deve permitir queries simples. Exemplo: myapp.herokuapp.com/states/minasgerais (essa query retorna como resposta a capital de Minas Gerais).

Formato de entrega: mail para o professor, contendo URL da sua aplicação (parte 1) e URL com algumas queries (parte 2).

Data de entrega: 06/07/2018 - Valor: 3 pontos

Parte 3: Módulos principais de uma rede social simples - Twitter Light

Semelhante às primeiras versões do Twitter; com mensagens apenas textuais, limitadas a 140 caracteres.

Módulos/microserviços a serem implementados:

(1) Users:

- login, internal_id, full name, followers, following, short bio
- cadastrar
- remover (sair da rede social)
- seguir um dado usuário (dado o seu ID)
- parar de seguir um dado usuário (dado o seu ID)

(2) Mensagens:

- internal_id (da mensagem), ID do usuário que postou, texto
- postar mensagem
- recuperar msg dado seu ID

(3) Timelines:

- Posts: dado um ID de um usuário, retorna uma lista com suas msg
- Home: dado um ID de um usuário, retorna uma lista com suas msgs e de todos os usuários que ele segue

Documente um conjunto de URLs que podem ser chamadas em um browser para teste do sistema (será feita uma demonstração em sala de aula).

O código deverá estar no GitHub (um diretório para cada microserviço).

Data de entrega: 18/09/2018 - Valor: 6 pontos

Grupos de até 3 alunos (de forma que cada um pode ficar responsável por um microserviço)