



UNIVERSIDADE  
FEDERAL DO CEARÁ

## UNIVERSIDADE FEDERAL DO CEARÁ

Novembro de 2019

**Curso:** Bacharelado em Ciência da Computação

Semestre: 2018.2

**Disciplina:** Sistemas Distribuídos

**Professor:** Filipe Fernandes dos Santos Brasil de Matos

**Nome da dupla:** \_\_\_\_\_

### TRABALHO PRÁTICO COM WEBSERVICES E REST

O objetivo deste trabalho é implementar uma aplicação de Agenda simples utilizando algumas tecnologias de comunicação remota inter-processo. A aplicação **DEVE** usar uma arquitetura Cliente/Servidor, onde a parte Servidora deverá gerenciar a Agenda do usuário (Cliente). O Cliente deve interagir com o Servidor utilizando algum tipo de interface (Gráfica ou não). O Servidor **DEVE**, pelo menos, disponibilizar um CRUD dos Contatos da Agenda. Algumas outras características da aplicação:

1. A Agenda deve ser tratada como uma lista de Contatos;
2. A Agenda deve ser persistida em disco utilizando alguma ferramenta a ser escolhida pelos membros. Pode ser um banco de dados relacional, não-relacional, arquivo de texto...
3. Cada Contato **DEVE** ser modelado com os campos: Nome, Endereço Completo, Telefone(s) e E-mail. O campo Telefone(s) é multivalorado, ou seja, pode assumir de um a N valores. Desses campos, somente o Nome e Telefone(s) são obrigatórios (os demais são opcionais).

O sistema deverá implementar um CRUD simples dos contatos da Agenda. O servidor do Sistema deverá oferecer os seguintes serviços:

- Adicionar Contato (via método POST do protocolo HTTP)
  - Os dados de uma notícia deverão ser recebidos via formulário HTML e enviados **no corpo da mensagem HTTP**.
  - Utilize as mensagens de status do HTTP como resposta ao cliente.
    - Status do HTTP para informar que operação foi realizada com sucesso e a retorná-la em formato JSON ou;
    - Status do HTTP informando que houve algum tipo de problema.
- Editar Contato (via método PUT do protocolo HTTP)
  - O id do contato a ser atualizado deverá ser passado **parametrizado na URI**.
  - As novas informações deverão ser recebidas via formulário HTML e enviados **no corpo da mensagem HTTP**.
  - Utilize as mensagens de status do HTTP como resposta ao cliente.
    - Status do HTTP para informar que a operação foi realizada com sucesso, retornando o contato (os novos dados) em um JSON ou;

- Status do HTTP informando que o contato não foi encontrado.
- Remover Contato (via método DELETE do protocolo HTTP)
  - O id do contato a ser removido deverá ser passado parametrizado na URI.
  - Utilize as mensagens de status do HTTP como resposta ao cliente.
    - Status do HTTP para informar que a operação foi realizada com sucesso e retornar, no corpo da mensagem, a notícia removida no formato XML ou;
    - Status do HTTP informando que o contato não foi encontrado.
- Requisitar Contato (via método GET do protocolo HTTP);
  - O ID do contato desejado deverá ser passado parametrizado na URI.
  - Utilize as mensagens de status do HTTP como resposta ao cliente.
    - Status do HTTP para informar que a operação foi realizada com sucesso e retornar, no corpo da mensagem, a representação da notícia em HTML ou;
    - Status do HTTP informando que o contato não foi encontrado.

Também deve ser desenvolvido uma aplicação Cliente, com menu (não necessariamente com interface gráfica), para consumir o serviço REST descrito acima.

Todo o código-fonte produzido deverá ser enviado, via e-mail, até as 23:59:59 do dia 30 de Novembro de 2019 ao meu e-mail institucional ([filipe.fernandes@crateus.ufc.br](mailto:filipe.fernandes@crateus.ufc.br)). O e-mail deverá ter como título **Trabalho Prático de WebService REST** e o corpo do e-mail deverá conter: **I) os nomes dos integrantes da equipe, II) um minitutorial de como executar os programas criados. O trabalho deve ser feito em, no máximo, trios.**

Além disso, cada grupo deverá apresentar os seus dois projetos para mim no dia 02 de Dezembro de 2019 (Segunda-Feira). Nesse dia, todos os participantes do grupo deverão estar presentes (participantes ausentes ficarão sem a nota do trabalho) e deverão, junto comigo, colocar o sistema para rodar, fazer alguns testes. Os integrantes também deverão apresentar os códigos gerados e, eventualmente, responder alguns questionamentos meus sobre a implementação. A ordem das apresentações será definida por sorteio e os horários serão definidos diretamente com cada equipe.

Todos os trabalhos serão analisados e comparados. Caso seja identificada cópia de trabalhos, todos os trabalhos envolvidos receberão nota ZERO. Trabalhos com atraso serão aceitos mediante um desconto de 1.0 ponto na nota final por dia de aula de atraso. Este trabalho deverá ser feito em grupo de, no máximo, três componentes.

Qualquer dúvida me procurem no Gabinete 2 ou me contactem via e-mail.