



## Aplicação 2 - serviços web (SOAP ou REST)



Ana Cristina Barreiras Kochem Vendramin • 9 de jun. Editado às 4 de ago.

30/30

Data de entrega: 11 de ago. 13:50



Adicionar comentário da turma

Vamos continuar com a aplicação de caronas e os mesmos métodos solicitados na aplicação 1. Porém, cliente e servidor precisam estar em linguagens de programação diferentes. O mais fácil é alterar a linguagem do cliente. Mas, vocês podem alterar a linguagem do cliente e do servidor, caso não queiram mais usar Java ou Python.

Para prover a comunicação entre os processos deve-se utilizar Web Services (SOAP ou REST) e não mais Java RMI ou Pyro.

A assinatura digital não precisa ser implementada.

Para o envio das notificações de eventos, vocês vão utilizar o SSE (Server-Sent Events).

Resumindo:

- Duas linguagens de programação diferente;
- Uso de SOAP ou REST para comunicação entre clientes e servidor;
- Uso do SSE para envio das notificações do servidor para os clientes.

Avaliação (valor 3,0):

- Consulta de caronas (valor 0,5);
- Registro de interesse em eventos (valor 1,0);
- Cancelamento de registro de interesse (valor 0,5);
- Envio de notificações com uso do SSE do servidor para o cliente (valor 1,0).

Obs.: como não precisamos mais enviar a chave pública, pois não precisamos mais assinar as mensagens, se vocês preferirem podem remover o método de cadastro de usuário. Nesse caso, informações para contato podem ser enviadas no registro de interesse. Porém, não há problema se quiserem manter a assinatura e o cadastro.

Seguem alguns links interessantes para implementar o SSE (Server-Sent Events) usando diferentes linguagens:

<https://www.baeldung.com/java-ee-jax-rs-sse>

<https://www.ibm.com/docs/en/was-liberty/core?topic=liberty-starting-server-sent-events>

<https://www.velotio.com/engineering-blog/how-to-implement-server-sent-events-using-python-flask-and-react>

Uso de canais: <https://flask-sse.readthedocs.io/en/latest/advanced.html>



<https://github.com/tserkov/vue-sse>



<https://medium.com/@chrisbautistaaa/server-sent-events-in-angular-node-908830cc29aa>

<https://docs.servicestack.net/csharp-server-events-client>

## Seus trabalhos

Com nota



GitHub - elderrenan/sistemasdistribuidosT2: Trabalho 2 da disciplina de ...  
<https://github.com/elderrenan/sistemasdistribuidosT2>



+ Adicionar ou criar

Enviar novamente



## 2 comentários particulares



**Elder Nogueira** 11 de ago.

Boa tarde professora. Provavelmente não vou conseguir entregar o trabalho a tempo hoje e nem defender. Me bati bastante com versões diferentes de GlassFish, NetBeans, eclipse, Maven, Tomcat, configurações de pom (o que é novo para mim) e por aí vai. Tentei várias abordagens diferentes e versões diferentes dessas ferramentas seguindo alguns tutoriais, mas sempre acabava travando em algum problema desse gênero. Cheguei a implementar um banco de dados usando o PostgreSQL e Python no Eclipse, e o REST em Java no NetBeans, mas descobri que não seria impossível implementar o SSE nessa abordagem. Na raiva deletei tudo e comecei do zero algumas vezes, portanto ainda não tenho nada a apresentar. Depois de pesquisar e entender um pouco melhor o escopo do trabalho, meu plano atual é usar o Tomcat como servidor de aplicação, chamando os mesmos métodos de cadastro e listagem do trabalho anterior através de uma página HTML com JavaScript. Após fazer isso funcionar, seguirei para a implementação das notificações utilizando o JAX-RS 2.1 implementation of SSE, conforme o link que você disponibilizou.



**Ana Cristina Barreiras Kochem Vendramin** 11 de ago.

Vou devolver, pois ainda não está finalizada, ok?



Adicionar comentário particular...

