

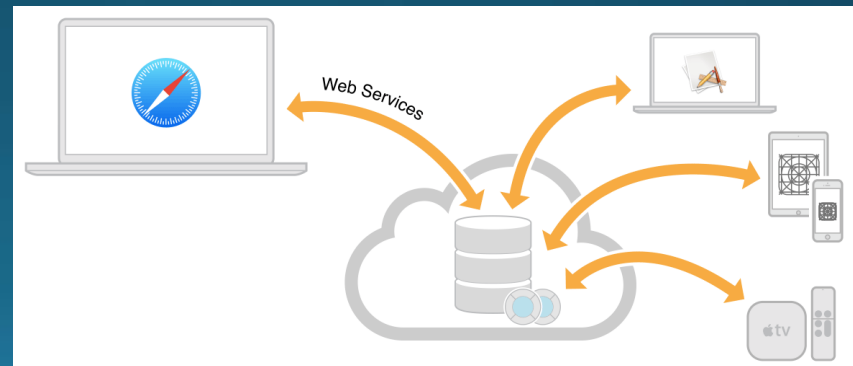
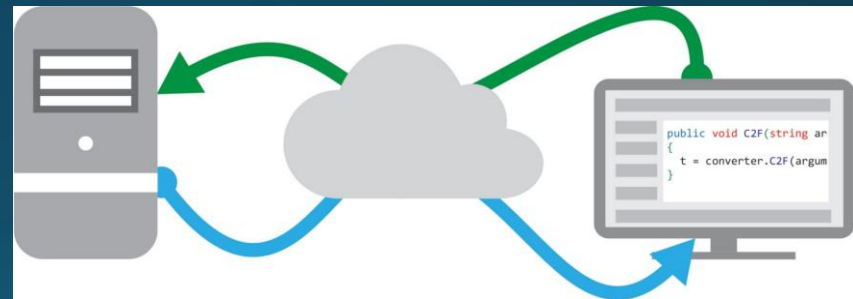
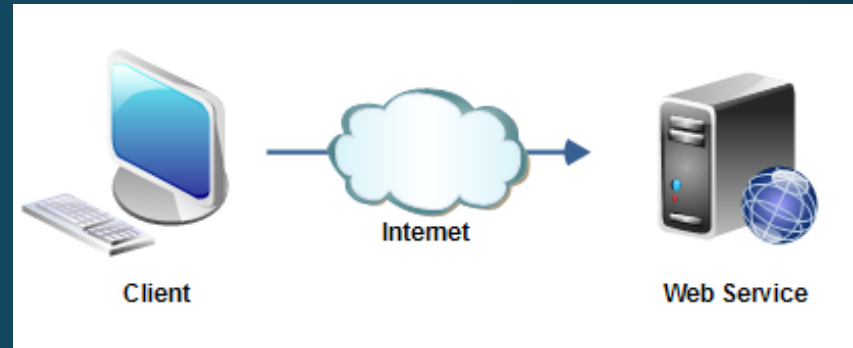
Professora Luciana P. de Araújo Kohler (lpa@furb.br)

Back-end – Parte 1

Tópicos

- Contextualização
- Web Services
- Tipos de Serviços
- Tipos de Erros

Contextualização



Web Services

↳ forma como os dados são transferidos (início - comando - fim)

SOAP (Simple Object Access Protocol)

- É um protocolo
- Não pode utilizar REST, pois é um protocolo
- Permite somente XML (wsdl)
- Suporte a SMTP e HTTP
- É mais confiável que REST
- Em alguns casos é mais rápido que REST
- Define sua própria segurança

REST (Representational State Transfer)

- É uma arquitetura
- Pode utilizar SOAP (como é um conceito, pode utilizar qualquer protocolo – HTTP, SOAP, etc)
- Permite diferentes formas de arquivos: text, HTML, JSON, XML, ...
- Necessita de menos banda e menos recursos
- Necessita somente do uso de HTTP
- É menos seguro que SOAP
- Em alguns casos é mais lento que o SOAP
- Serviços de segurança são herdados de outros serviços e recursos

Tipos de Serviços REST

- POST – utilizado para inserir dados
- PUT – utilizado para alterar dados
- GET – utilizado para selecionar dados
- DELETE – utilizado para excluir dados

Tipos de Serviços REST

- POST – utilizado para inserir dados
- PUT – utilizado para alterar dados
- GET – utilizado para selecionar dados
- DELETE – utilizado para excluir dados

E se eu usar o GET para tudo?

Nada acontece.... Mas semanticamente está incorreto!

Para que serve um serviço?

Interface de comunicação entre cliente e servidor;

Para disponibilizar funcionalidades de modo que uma aplicação cliente possa acessar uma aplicação servidora sem ter conhecimento de sua arquitetura;

É independente de uma linguagem de programação;

Permite integrar sistemas de arquiteturas e linguagens diferentes.

Serviço é método

Tipos de Erros

- As mensagens HTTP iniciam com números que especificam seus significados, por exemplo:
 - 1... = informação
 - 2... = sucesso
 - 3... = redirecionamento
 - 4... = erro de cliente
 - 5... = erros externos (desconhecidos)

Principais Erros

	Status	Descrição
SUCESSO	200, OK	Quando o serviço é executado pelo cliente com sucesso (mais comum).
	201, Created	Utilizado quando um recurso é criado com sucesso.
	204, No Content	Utilizado quando não há o que retornar.
ERRO NO LADO DO CLIENTE	400, Bad Request	Acessou o serviço, mas ocorreu algum erro de validação em alguma regra de negócio da aplicação. Chamada de forma errada.
	401, Unauthorized	Necessita autenticação ou ocorreu erro.
	403, Forbidden	Autenticação bem sucedida, mas usuário não tem permissão.
	404, Not Found	Serviço solicitado não existe.
ERRO NO LADO DO SERVIDOR	500, Internal Server Error	Erro inesperado na aplicação servidora.

Alguns conceitos de serviços

Rotas ou endpoints

URL onde o serviço pode ser acessado
por uma aplicação cliente

Tipo de conteúdo (content-type)

Forma como o cliente envia os dados para a aplicação servidora.

Exemplos:

application/json

application/xml

text/plain

text/html

....

Passagem de parâmetros

Tipo	Definição	Exemplo
Path	Apenas um parâmetro e é obrigatório para o serviço. Passado na URL.	/pessoa/ 1 <small>1 = path</small>
Query	Vários parâmetros, podendo ser opcionais. Passado na URL.	/pessoa?nome=joao &idade=10
Header	Envia metadados e atributos ao servidor (token, credenciais de autenticação, etc)	token-id:AnXDWj103
Body	Usado para passar conteúdos para os serviços PUT e POST	{"nome": "joao", "idade": 10}

APIs REST desenvolvidores

- <http://dummy.restapiexample.com/> (teste)
- <https://any-api.com/>
- <https://rapidapi.com/>

Cientes REST para testar

- Postman
- ARC (Advanced Rest Client – Google Chrome)