**1** [**Q2108867**](https://www.qconcursos.com/questoes-de-concursos/questoes/14d48080-d2) [Sistemas Distribuídos](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/sistemas-distribuidos)

**Ano:**2022 **Banca:**[FCC](https://www.qconcursos.com/questoes-de-concursos/bancas/fcc) **Órgão:**[TRT - 19ª Região (AL)](https://www.qconcursos.com/questoes-de-concursos/institutos/trt-19-regiao-al) **Prova:**[FCC - 2022 - TRT - 19ª Região (AL) - Analista Judiciário - Área Apoio Especializado Especialidade: Tecnologia da Informação](https://www.qconcursos.com/questoes-de-concursos/provas/fcc-2022-trt-19-regiao-al-analista-judiciario-area-apoio-especializado-especialidade-tecnologia-da-informacao)

Na classe principal de uma aplicação criada com Spring Boot, para fazer com que a aplicação atue como um cliente Eureka utiliza-se a anotação

**A** @EurekaClient

**B** @EnableEurekaClient

**C** @EnableDiscoveryEurekaClient

**D** @EurekaDiscoveryClient

**E** @IsEurekaClient

**2** [**Q1991616**](https://www.qconcursos.com/questoes-de-concursos/questoes/d1021979-73) [Arquitetura Orientada a Objetos](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/arquitetura-orientada-a-objetos)

**Ano:**2022 **Banca:**[FCC](https://www.qconcursos.com/questoes-de-concursos/bancas/fcc) **Órgão:**[TJ-CE](https://www.qconcursos.com/questoes-de-concursos/institutos/tj-ce) **Prova:**[FCC - 2022 - TJ-CE - Analista Judiciário - Ciência da Computação - Sistemas da Informação](https://www.qconcursos.com/questoes-de-concursos/provas/fcc-2022-tj-ce-analista-judiciario-ciencia-da-computacao-sistemas-da-informacao)

Considerando as características de uma arquitetura de objetos distribuídos,

**A** o modelo cliente-servidor é, por si só, um impeditivo para seu uso.

**B** um objeto pode ser cliente e servidor no mesmo momento.

**C** são necessários apenas protocolos de baixo nível para a interação entre os objetos.

**D** a referência ao objeto não necessita do endereço de rede.

**E** apenas a interface do objeto é visível.

Diego M.

13 de Dezembro de 2022 às 14:46

Um sistema de objetos distribuídos é aquele que permite a operação com **objetos remotos**. Dessa forma é possível, a partir de uma aplicação cliente orientada a objetos, obter uma referência para um objeto que oferece o serviço desejado e, através dessa referência, invocar métodos desse objeto -- mesmo que a instância desse objeto esteja em uma máquina diferente daquela do objeto cliente.

**3** **[Q1986136](https://www.qconcursos.com/questoes-de-concursos/questoes/59cd8d7f-69" \t "_blank)** [SOAP (Simple Object Access Protocol)](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/soap-simple-object-access-protocol)

**Ano:**2022 **Banca:**[FCC](https://www.qconcursos.com/questoes-de-concursos/bancas/fcc) **Órgão:**[TRT - 22ª Região (PI)](https://www.qconcursos.com/questoes-de-concursos/institutos/trt-22-regiao-pi) **Prova:**[FCC - 2022 - TRT - 22ª Região (PI) - Analista Judiciário - Tecnologia da Informação](https://www.qconcursos.com/questoes-de-concursos/provas/fcc-2022-trt-22-regiao-pi-analista-judiciario-tecnologia-da-informacao)

 Para indicar informações de erro em uma mensagem SOAP, um Analista utilizou o elemento Fault, que

**A** só pode aparecer uma vez em cada mensagem.

**B** utiliza o subelemento <faultname> para identificar a falha.

**C** é obrigatório em toda mensagem SOAP.

**D** é filho do elemento <header>.

**E** utiliza o subelemento <faultsource> para indicar a causa da falha.

Anderson Rodrigues

10 de Janeiro de 2023 às 15:45

**O Elemento Fault**

O elemento **opcional** Fault é usado para indicar mensagem de erro. Se este elemento estiver presente, ele aparecera como **filho do elemento Body**. O elemento Fault só pode aparecer **uma vez** em cada mensagem SOAP.

https://www.adonai.eti.br/2011/06/soap-simple-object-access-protocol/

* <Envelope> é o elemento raiz em cada mensagem SOAP (é obrigatório);
* <Header> é um subelemento do envelope SOAP (é opcional)
* <Body> é um subelemento do envelope SOAP (é obrigatório)
* <Fault> é um subelemento do corpo SOAP (é opcional)

Pedro Oliveira da Silva Rosado

09 de Janeiro de 2023 às 19:56

https://www.adonai.eti.br/2011/06/soap-simple-object-access-protocol/

**4** [**Q1962704**](https://www.qconcursos.com/questoes-de-concursos/questoes/ba5857ba-34) [Sistemas Distribuídos](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/sistemas-distribuidos)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[SEFAZ-AM](https://www.qconcursos.com/questoes-de-concursos/institutos/sefaz-am) **Prova:**[FGV - 2022 - SEFAZ-AM - Analista de Tecnologia da Informação da Fazenda Estadual - Tarde](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-sefaz-am-analista-de-tecnologia-da-informacao-da-fazenda-estadual-tarde)

Com relação ao papel dos **middleware** nos sistemas distribuídos, analise as afirmativas a seguir.

I. São uma camada de *software* que fornece uma abstração de programação, bem como o mascaramento de heterogeneidade de redes de computadores, sistemas operacionais e linguagens de programação dos *hosts*.

II. Fornecem um modelo computacional uniforme para ser usado pelos programadores de serviços e de aplicativos distribuídos. Os modelos possíveis incluem a invocação remota de objetos, a notificação remota de eventos, o acesso remoto a banco de dados e o processamento de transação distribuído.

III. Atuam como contêineres que oferecem modos de gerar um código executável ou *bytecode* otimizado para uma ampla variedade de dispositivos hospedeiros: o compilador de uma linguagem em particular gera um pseudo-código para o contêiner de acordo com um processador e um sistema operacional específicos.

Está correto o que se afirma em

**A** I, apenas.

**B** II, apenas.

**C** III, apenas.

**D** I e II, apenas.

**E** II e III, apenas.

Asdrubal Trombonni

19 de Outubro de 2022 às 22:15

Da Microsoft (https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary/what-is-middleware/):

**Middleware**é um software que fica **entre um sistema operacional e os aplicativos executados nele**. Funcionando essencialmente como uma **camada de tradução oculta**, o middleware permite a **comunicação e o gerenciamento de dados para aplicativos distribuídos**. Às vezes, é chamado de pipe, pois conecta dois aplicativos para que dados e bancos de dados possam ser facilmente transmitidos entre o “pipe”. O uso de middleware permite que os usuários realizem solicitações como enviar formulários em um navegador da Web ou permitir que o servidor da Web retorne páginas da Web dinâmicas com base no perfil de um usuário.

Exemplos de middleware comuns incluem middleware de banco de dados, middleware de servidor de aplicativos, middleware orientado a mensagens, middleware da Web e monitores de processamento de transações. Cada programa normalmente fornece serviços de mensagens para que diferentes aplicativos possam se comunicar usando estruturas de mensagens como protocolo de acesso a objetos simples (SOAP), serviços da Web, transferência de estado representacional (REST) ​​e notação de objeto JavaScript (JSON). Embora todos os middlewares desempenhem funções de comunicação, o tipo que uma empresa escolhe usar dependerá de qual serviço está sendo usado e que tipo de informação precisa ser comunicada. Isso pode incluir autenticação de segurança, gerenciamento de transações, filas de mensagens, servidores de aplicativos, servidores da Web e diretórios. O middleware também pode ser usado para processamento distribuído com ações ocorrendo em tempo real, em vez de enviar e receber dados.

Rafael Catelan Do Nascimento

24 de Novembro de 2022 às 09:04

III seria mais um interpretador/compilador. Algo tipo a JVM e não middleware

**5** [**Q1962702**](https://www.qconcursos.com/questoes-de-concursos/questoes/ba50d5fd-34) [Sistemas Distribuídos](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/sistemas-distribuidos)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[SEFAZ-AM](https://www.qconcursos.com/questoes-de-concursos/institutos/sefaz-am) **Prova:**[FGV - 2022 - SEFAZ-AM - Analista de Tecnologia da Informação da Fazenda Estadual - Tarde](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-sefaz-am-analista-de-tecnologia-da-informacao-da-fazenda-estadual-tarde)

Considerando os relógios de computadores e os eventos de temporização em sistemas distribuídos, assinale a opção que indica o termo que descreve a taxa de desvio de um relógio e a quantidade relativa pela qual o relógio de computador difere de um relógio de referência perfeito.

**A** *jitter.*

**B** *drift.*

**C** latência.

**D** *timestamp.*

**E** decaimento.

Cleber godoy

13 de Novembro de 2022 às 16:19

- Jitter é a variação estatística do retardo (atraso) na entrega de dados em uma rede, a qual produz uma recepção não regular dos pacotes. [...] Jitter é a variação no tempo exigida para distribuir uma série de mensagens, muito crucial para dados multimídia.

Como temos vários recursos computacionais em sistemas distribuídos, os relógios internos dos computador são utilizados para obter o valor atual da hora pelos processos em computadores diferentes. Cada computador possui um relógio desses.

Cada processo pode associar carimbos de tempo, conhecidos como time stamping. Nesse caso, mesmo que os processos façam a leitura dos seus relógios locais ao mesmo tempo, podemos ter valores diferentes.

Isso acontece, pois os relogios se desviam de uma base de tempo e com isso suas taxas de desvio ficam diferentes entre si.

Para calcular essa diferença temos um termo conhecido como taxa de desvio do relógio (drift) que "se refere à quantidade relativa pela qual um relógio de computador difere de um relógio de referência perfeito".

Como estamos em um sistema distribuído, podemos acrescentar mais uma informação.

Mesmo que os relógios de todos os computadores de um sistema distribuído fossem inicialmente ajustados com o mesmo horário, com o passar do tempo eles variariam entre si significativamente, a menos que fossem reajustados.

- A latência é o atraso decorrido entre o início da transmissão de uma mensagem em um processo remetente e o início da recepção pelo processo destinatário.

- Time stamps são carimbos de tempo. Esse é um recurso que um processo utiliza para marcar o tempo dos eventos.

- Decaimento se refere a perda, ou diminuição, de intensidade.

**6** [**Q1962647**](https://www.qconcursos.com/questoes-de-concursos/questoes/b98712e3-34) [WebServices](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/webservices)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[SEFAZ-AM](https://www.qconcursos.com/questoes-de-concursos/institutos/sefaz-am) **Prova:**[FGV - 2022 - SEFAZ-AM - Analista de Tecnologia da Informação da Fazenda Estadual - Tarde](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-sefaz-am-analista-de-tecnologia-da-informacao-da-fazenda-estadual-tarde)

*Web Services* são frequentemente utilizados para a integração entre sistemas computacionais.

A notação XML padrão, utilizada para descrever *Web Services*, é a

**A** SOAP.

**B** UDDI.

**C** XPATH.

**D** XSLT.

**E** WSDL.

Arthur - Informática

03 de Dezembro de 2022 às 19:40

**GAB E**

A **Web Services Description Language** (**WSDL**) é uma linguagem baseada em XML utilizada para descrever Web Services funcionando como um contrato do serviço.

**7** [**Q1962645**](https://www.qconcursos.com/questoes-de-concursos/questoes/b9807a1f-34) [Padrões de projeto (Design Patterns) ,](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/padroes-de-projeto-design-patterns)[Arquitetura Orientada a Objetos](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/arquitetura-orientada-a-objetos)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[SEFAZ-AM](https://www.qconcursos.com/questoes-de-concursos/institutos/sefaz-am) **Prova:**[FGV - 2022 - SEFAZ-AM - Analista de Tecnologia da Informação da Fazenda Estadual - Tarde](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-sefaz-am-analista-de-tecnologia-da-informacao-da-fazenda-estadual-tarde)

Os padrões GRASP descrevem princípios fundamentais de atribuição de responsabilidades em projetos de *software* orientado a objeto.

De acordo com esses padrões, o *polimorfismo* é um princípio orientador para atribuir responsabilidades

**A** a classes artificiais que não representem nada no domínio do problema de maneira a obter uma coesão alta.

**B** a classes abstratas que representem o comportamento de classes concretas para permitir ao sistema lidar com vários tipos de maneira homogênea.

**C** a uma classe controladora global para centralizar o tratamento de todas as mensagens e eventos do sistema.

**D** a classes que têm a informação necessária para satisfazer à responsabilidade e colaborar para que o acoplamento permaneça alto.

**E** a classes estáticas, altamente coesas e que são utilizadas no sistema sem a necessidade de instanciar objetos.

Jerffeson Gomes

20 de Setembro de 2022 às 22:01

A -> **Pure Fabrication**(Pura Fabricação)  - a classes artificiais que não representem nada no domínio do problema de maneira a obter uma coesão alta.

B -> **Polymorphism**(Polimorfismo) — As responsabilidades devem ser atribuídas a abstrações e não a objetos concretos, permitindo que eles possam variar conforme a necessidade.

C -> **Controller**(Controlador) — Atribui a responsabilidade de lidar com os eventos do sistema para uma classe que representa a um cenário de caso de uso ou sistema todo.

D -> **Information Expert**(Especialista na Informação) — Determina quando devemos delegar a responsabilidade para um outro objeto que seja especialista naquele domínio.

E -> Essa eu não tenho certeza, mas acho q seja: **High Cohesion**(Alta Coesão)— este princípio determina que as classes devem se focar apenas na sua responsabilidade.

Matos Roberto

02 de Novembro de 2022 às 20:12

a classes abstratas que representem o comportamento de classes concretas para permitir ao sistema lidar com vários tipos de maneira homogênea.

**8** [**Q1962644**](https://www.qconcursos.com/questoes-de-concursos/questoes/b97d7fe6-34) [Padrões de projeto (Design Patterns)](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/padroes-de-projeto-design-patterns)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[SEFAZ-AM](https://www.qconcursos.com/questoes-de-concursos/institutos/sefaz-am) **Prova:**[FGV - 2022 - SEFAZ-AM - Analista de Tecnologia da Informação da Fazenda Estadual - Tarde](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-sefaz-am-analista-de-tecnologia-da-informacao-da-fazenda-estadual-tarde)

O padrão de projeto de *software* orientado a objetos, que pode ser utilizado para garantir que uma determinada classe tenha somente uma única instância, é

**A** *adapter.*

**B** *flyweight.*

**C** *memento.*

**D** *singleton.*

**E** *visitor.*

Silas Rodrigues do Rêgo Júnior

05 de Outubro de 2022 às 00:19

O **Singleton** é um padrão de projeto criacional, que garante que apenas um objeto desse tipo exista e forneça um único ponto de acesso a ele para qualquer outro código.

Fonte: https://refactoring.guru/pt-br/design-patterns/singleton/java/example

Matos Roberto

02 de Novembro de 2022 às 20:11

singleton.

Novaes

02 de Maio de 2023 às 15:44

**GABARITO: D)SINGLETON**

**O Adapter** é um padrão de projeto estrutural que permite objetos com interfaces incompatíveis colaborarem entre si.

**O Flyweight** é um padrão de projeto estrutural que permite a você colocar mais objetos na quantidade de RAM disponível ao compartilhar partes comuns de estado entre os múltiplos objetos ao invés de manter todos os dados em cada objeto.

**O Memento** é um padrão de projeto comportamental que permite que você salve e restaure o estado anterior de um objeto sem revelar os detalhes de sua implementação.

**O Singleton** é um padrão de projeto criacional que permite a você garantir que uma classe tenha apenas uma instância, enquanto provê um ponto de acesso global para essa instância.

**O Visitor** é um padrão de projeto comportamental que permite que você separe algoritmos dos objetos nos quais eles operam.

FONTE: https://refactoring.guru/pt-br/design-patterns/catalog

**9** **[Q1936791](https://www.qconcursos.com/questoes-de-concursos/questoes/328d94cb-fd" \t "_blank)** [WebServices ,](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/webservices)[SOAP (Simple Object Access Protocol)](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/soap-simple-object-access-protocol)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[TJ-DFT](https://www.qconcursos.com/questoes-de-concursos/institutos/tj-dft) **Prova:**[FGV - 2022 - TJ-DFT - Analista Judiciário - Análise de Dados](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-tj-dft-analista-judiciario-analise-de-dados)

No âmbito de Web services, analise as afirmativas a seguir sobre a abordagem REST e o uso de tecnologias baseadas em SOAP.

I. Uma característica dos serviços Web RESTful é a capacidade de transmitir dados diretamente via HTTP.

II. As mensagens SOAP precisam ser retornadas como documentos XML.

III. Um navegador não pode armazenar em cache uma solicitação concluída por uma API SOAP.

É correto o que se afirma em:

**A** somente II;

**B** somente I e II;

**C** somente I e III;

**D** somente II e III;

**E** I, II e III.

Elton Lagroteria

16 de Novembro de 2022 às 23:10

I. uma API REST tem como padrão a utilização de HTTP como protocolo

II. a resposta do protocolo SOAP é via XML

III. para API REST é possível armazenar dados em cache, para evitar novas requisições, porém no protocolo SOAP não.

https://www.redhat.com/pt-br/topics/integration/whats-the-difference-between-soap-rest

Gloomy Gulch

11 de Agosto de 2022 às 22:43

O Web Application Description Language (WADL) é um XML utilizado para descrever serviços RESTFul.- Web services baseados sobre a arquitetura REST são conhecidos como RESTFul Web services. RESTFul Web services utilizam o HTTP para transportar o dado e o JSON para representar o dado.

**10** [**Q1936747**](https://www.qconcursos.com/questoes-de-concursos/questoes/873ca656-fc) [WebServices](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/webservices)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[TJ-DFT](https://www.qconcursos.com/questoes-de-concursos/institutos/tj-dft) **Prova:**[FGV - 2022 - TJ-DFT - Analista Judiciário - Análise de Sistemas](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-tj-dft-analista-judiciario-analise-de-sistemas)

O analista de sistemas Pedro desenvolveu o *webservice* RService aplicando o estilo de arquitetura REST (*Representational State Transfer*). As aplicações clientes que utilizam o RService são desenvolvidas de forma desacoplada e dissociada de RService e manipulam os recursos de RService através de representações transferidas em mensagens autodescritivas.

Para habilitar a independência no desenvolvimento de aplicações clientes com o uso de representações em mensagens autodescritivas, Pedro aplicou em RService o princípio REST:

**A** arquitetura cliente-servidor;

**B** código sob demanda;

**C** interface uniforme;

**D** sistema em camadas;

**E** capacidade de cache.

Elionay Manuel

10 de Agosto de 2022 às 20:43

GABARITO - C) INTERFACE UNIFORME

Representational State Transfer, abreviado como REST, não é uma tecnologia, uma biblioteca, e nem tampouco uma arquitetura, mas sim um modelo a ser utilizado para se projetar arquiteturas de software distribuído, baseadas em comunicação via rede.

REST na verdade pode ser considerado como um conjunto de princípios:

1. todo recurso deve possuir uma identificação única. Essa identificação serve para que a aplicação consiga diferenciar qual dos recursos deve ser manipulado em uma determinada solicitação.
2. Ao definir uma URI, utilize nomes legíveis por humanos, que sejam de fácil dedução e que estejam relacionados com o domínio da aplicação.
3. evite definir URIs que contenham a operação a ser realizada em um recurso
4. evite definir URIs que contenham o formato desejado de um recurso
5. Após definir uma URI e disponibilizar a manipulação de um recurso por ela, evite ao máximo sua alteração.
6. https://www.alura.com.br/artigos/rest-principios-e-boas-praticas?gclid=Cj0KCQjwrs2XBhDjARIsAHVymmTO48GNdu\_KTN5HS-YPNwvpPfDECrOyFBENssfMbdF7OeIyAg6PtL4aAuTkEALw\_wcB

Ao construir uma API não importa qual cliente vá utiliza-la os recursos sempre serão acessados pelas URIs definidas, que são uma só, portanto uniformes

Tyrel Sonahara

13 de Janeiro de 2023 às 13:13

**INTERFACE UNIFORME**

Se divide em mais quatro restrições, portanto, não é realmente uma única restrição, mas apenas mais quatro restrições:

* O que REST retorna é uma representação do recurso, que pode ter um formato diferente do recurso no servidor. Exemplo: Dados armazenados como uma tabela no banco, mas representação de retorno pode ser JSON, XML ou HTML;
* O cliente com o nível de acesso correto pode controlar o que está armazenado no servidor;
* Cada representação deve descrever seu próprio formato de dados;
* **O serviço REST descreve seu próprio uso com cada recurso retornado, por meio dos hiperlinks fornecidos.﻿**

https://dev.to/dev\_jessi/o-basico-para-entender-apis-restful-parte-1-1apm

**11** **[Q1936746](https://www.qconcursos.com/questoes-de-concursos/questoes/873943ef-fc" \t "_blank)** [WebServices ,](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/webservices)[SOAP (Simple Object Access Protocol)](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/soap-simple-object-access-protocol)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[TJ-DFT](https://www.qconcursos.com/questoes-de-concursos/institutos/tj-dft) **Prova:**[FGV - 2022 - TJ-DFT - Analista Judiciário - Análise de Sistemas](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-tj-dft-analista-judiciario-analise-de-sistemas)

O analista de sistemas Bravo desenvolveu o *webservice* TJService para o TJDFT sem mecanismos de controle de segurança. A comunicação de TJService ocorre conforme especificações do protocolo SOAP (*Simple Object Access Protocol*). A fim de proteger a troca de mensagens de TJService, Bravo foi orientado pelo TJDFT a implementar o suporte à autenticação baseada em token.

Os mecanismos de proteção de mensagens SOAP baseados em token que Bravo deve implementar são definidos na especificação Web Services:

**A** Dynamic Discovery;

**B** Addressing;

**C** Reliable Messaging;

**D** Security;

**E** Make Connection.

Pablo Robert Pereira Alves

12 de Outubro de 2022 às 11:47

**gab. D**

**Web Services Security** (**WS-Security**, **WSS**) is an extension to to apply security to . It is a member of the and was published by .

The protocol specifies how integrity and confidentiality can be enforced on messages and allows the communication of various security token formats, such as (SAML), , and . Its main focus is the use of and to provide end-to-end security.

Fonte: Wikipedia: https://en.wikipedia.org/wiki/WS-Security

**12** [**Q1936742**](https://www.qconcursos.com/questoes-de-concursos/questoes/872d4dc9-fc) [Arquitetura da Informação](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/arquitetura-da-informacao)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[TJ-DFT](https://www.qconcursos.com/questoes-de-concursos/institutos/tj-dft) **Prova:**[FGV - 2022 - TJ-DFT - Analista Judiciário - Análise de Sistemas](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-tj-dft-analista-judiciario-analise-de-sistemas)

Em User Interface (UI), Formulário é um grupo de controles de entrada relacionados que permite que os usuários forneçam dados ou configurem opções. O Input está entre os componentes de entrada de dados mais utilizados e pode fazer uso de Placeholders.

O uso de Placeholders pode proporcionar mais clareza no design pois eles:

**A** agrupam elementos em formulários;

**B** dispensam a utilização de máscaras;

**C** geram rótulos dinâmicos tais como Legends;

**D** são acionáveis pelos usuários em caso de dúvidas;

**E** fornecem dicas e exemplos sobre o dado correto a ser inserido.

Leandro Henrique

16 de Julho de 2022 às 18:30

**GAB E**

**Exemplo em HTML:**

* <label>Conta corrente: </label>
* <input placeholder="000000-0">

**13** **[Q1935280](https://www.qconcursos.com/questoes-de-concursos/questoes/cde9a90a-fb" \t "_blank)** [SOAP (Simple Object Access Protocol)](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/soap-simple-object-access-protocol)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[MPE-SC](https://www.qconcursos.com/questoes-de-concursos/institutos/mpe-sc) **Prova:**[FGV - 2022 - MPE-SC - Analista em Tecnologia da Informação](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-mpe-sc-analista-em-tecnologia-da-informacao)

A analista em TI Isabel está implementando o web service DocService para o MPSC. A comunicação do DocService deve se basear em especificações derivadas do protocolo SOAP (*Simple Object Access Protocol*) e do framework WS-Security (*Web Services Securit*y).

Para que DocService possa utilizar contextos de segurança baseados em token entre múltiplas mensagens SOAP, Isabel deve observar diretamente a especificação WS (*Web Services*):

**A** BrokeredNotification;

**B** MakeConnection;

**C** SecureConversation;

**D** BaseNotification;

**E** MetadataExchange.

Luciano Silva

23 de Janeiro de 2023 às 20:12

**✅Gabarito(C)**

A especificação WS-SecureConversation introduz o conceito de um contexto de segurança e seu uso. O token do contexto de segurança é um novo tipo de token do WS-Security que representa o conceito abstrato do contexto de segurança. O token é identificado por um URI e consiste nas chaves negociadas, além de outras propriedades de segurança relacionadas. O modelo de autenticação de contexto autentica uma série de mensagens e, portanto, trata desses interesses. O modelo de autenticação do contexto aumenta o desempenho geral e a proteção de trocas subseqüentes, mas requer comunicações adicionais quando a autenticação acontece antes das trocas de aplicativo normais.

Fonte: https://www.ibm.com/docs/pt-br/was-nd/8.5.5?topic=conversation-web-services-secure-standard

**14** [**Q1935271**](https://www.qconcursos.com/questoes-de-concursos/questoes/cdcf6fff-fb) [WebServices](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/webservices)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[MPE-SC](https://www.qconcursos.com/questoes-de-concursos/institutos/mpe-sc) **Prova:**[FGV - 2022 - MPE-SC - Analista em Tecnologia da Informação](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-mpe-sc-analista-em-tecnologia-da-informacao)

A Equipe de Sistemas de Software (ESS) de um Ministério Público deseja criar um Web Service para verificação do andamento dos processos a ser reutilizável por diversos usuários e aplicações. Para que o Web Service seja utilizado, a ESS deve incluir na descrição do serviço a:

**A** definição dos n*amespaces* em WSDL (*Web Service Definition Language*);

**B** descrição da assinatura do serviço com as operações, parâmetros e seus tipos;

**C** especificação de suas características não funcionais, como desempenho e confiança;

**D** definição da interface de serviço incluindo informações sobre a semântica do serviço;

**E** especificação do padrão UDDI (*Universal Description, Discovery and Integration*) como protocolo para troca de mensagens.

Luciano Silva

23 de Janeiro de 2023 às 20:01

✅Gabarito(Certo)

Esta se referindo ao Web Service description Languagem - WSDL

* WSDL é utilizado pelo provedor de serviços para descrever seus serviços para o registro.
* A WSDL permite descrever web services em XML, tornando-os disponíveis para usuários e programas web.

Fonte: QC

**15** [**Q1891218**](https://www.qconcursos.com/questoes-de-concursos/questoes/40e69569-ae) [WebServices](https://www.qconcursos.com/questoes-de-concursos/disciplinas/tecnologia-da-informacao-arquitetura-de-software/webservices)

**Ano:**2022 **Banca:**[FGV](https://www.qconcursos.com/questoes-de-concursos/bancas/fgv) **Órgão:**[CGU](https://www.qconcursos.com/questoes-de-concursos/institutos/cgu) **Prova:**[FGV - 2022 - CGU - Auditor Federal de Finanças e Controle - Tecnologia da Informação](https://www.qconcursos.com/questoes-de-concursos/provas/fgv-2022-cgu-auditor-federal-de-financas-e-controle-tecnologia-da-informacao)

Marcos recebeu a tarefa de configurar as regras de entrada de três grupos de segurança (sg-1, sg-2 e sg-3) em uma nuvem do provedor AWS, a fim de atender as restrições descritas a seguir. Um serviço ECS, associado ao primeiro grupo de segurança (sg-1), somente deve receber requisições HTTP de um balanceador de carga, associado ao segundo grupo de segurança (sg-2). Adicionalmente, as tarefas desse serviço ECS devem ser os únicos recursos a acessar uma instância de banco de dados PostgreSQL no serviço RDS, associada ao terceiro grupo de segurança (sg-3). Portanto, as regras de entrada de cada grupo devem ser:

**A**

(sg-1) Intervalo de portas: 5432,Origem: sg-2

(sg-2) Intervalo de portas: 80, Origem: 0.0.0.0/0

(sg-3) Intervalo de portas: 80, Origem: sg-1

**B**

 (sg-1) Intervalo de portas: 80, Origem: sg-2

(sg-2) Indiferente

(sg-3) Intervalo de portas: 5432, Origem: sg-1

**C**

(sg-1) Indiferente

(sg-2) Intervalo de portas: 80, Origem: sg-1

(sg-3) Intervalo de portas: 5432, Origem: sg-1

**D**

(sg-1) Intervalo de portas: 5432, Origem: sg-3

(sg-2) Intervalo de portas: 80, Origem: 0.0.0.0/0

(sg-3) Indiferente

**E**

(sg-1) Indiferente

(sg-2) Intervalo de portas: 80, Origem: 0.0.0.0/0

(sg-3) Intervalo de portas: 5432, Origem: sg-1

André Nascimento

14 de Janeiro de 2023 às 08:59

SG-1 recebe requisição http. Se é http, é porta 80, só a alternativa b tem essa opção. O resto nem precisa conhecer, já mata a questão.