



APRENDIZAGEM EM FOCO

INTERFACE DE PROGRAMAÇÃO DE APLICAÇÕES (API) E WEB SERVICES



APRESENTAÇÃO DA DISCIPLINA

Autoria: Arthur Gonçalves Ferreira

Leitura crítica: Rennan Martini Rodrigues

Na disciplina *Interface de programação de aplicações (API) e Web Services*, você conhecerá sobre os conceitos, as características, funcionalidades e aplicabilidades de APIs e Web Services. Sendo assim, você irá estudar sobre o que é uma API, qual a sua estrutura e como ela funciona. Vamos aprender que a utilização de APIs são muito mais comuns do que você imagina, e que diariamente elas nos auxiliam a desenvolver desde tarefas simples as mais complexas.

Além dos conceitos e aplicabilidade das APIs, você terá a oportunidade de estudar sobre o que é uma Web Service, como ela funciona e qual é sua estrutura. Vamos verificar que apesar de parecidas, as Web Services se diferenciam das APIs, em que a primeira é classificada como uma interface de programação, um conjunto de rotinas e padrões de programação que possuem o objetivo de acessar aplicativos de software. Já a segunda, são tecnologias que realizam a comunicação entre máquinas.

Por fim, você vai conhecer na prática como realizar o desenvolvimento de APIs e Web Services com a utilização de ferramentas específicas. Dessa forma, você vai entender conceitos extremamente importantes, como o que é o consumo de APIs e Web Services, quais as tecnologias aplicadas nelas e quais os tipos de arquiteturas que são utilizadas em cada uma.

Ao final deste tema, você será capaz de colocar em prática todos os conhecimentos adquiridos, realizando análise e desenvolvimento de APIs e Web Services. Dessa forma, te convido a estudar os materiais dessa disciplina, para que você possa adquirir o melhor

conhecimento possível e se desenvolver profissionalmente. Tenha um ótimo estudo!

INTRODUÇÃO

Olá, aluno (a)! A *Aprendizagem em Foco* visa destacar, de maneira direta e assertiva, os principais conceitos inerentes à temática abordada na disciplina. Além disso, também pretende provocar reflexões que estimulem a aplicação da teoria na prática profissional. Vem conosco!



TEMA 1

Caracterizando as APIs dos Web Services

Autoria: Arthur Gonçalves Ferreira

Leitura crítica: Rennan Martini Rodrigues



DIRETO AO PONTO

As APIs são fundamentais na comunicação entre aplicações no mundo atual. Sem elas não seríamos capazes, por exemplo, de integrar nossa lista de contato do celular em aplicativos como é o caso do WhatsApp. O trabalho maior seria para o desenvolvedor, que teria que criar interfaces do zero sem poder utilizar padrões já existentes.

É verdade que as APIs vão muito além de auxiliar em ações simples do cotidiano, como é o caso do aplicativo de mensagens citado anteriormente. Podemos entender uma API como uma ferramenta, na realidade um conjunto de ferramentas, que proporciona funcionalidades novas a um determinado software, utilizando padrões já existentes. A Figura 1 ilustra a aplicabilidade de uma API.

Figura 1 – Aplicabilidade de uma API



Fonte: <https://becode.com.br/o-que-e-api-rest-e-restful/>. Acesso em: 25 out. 2021.

Dessa forma, podemos pensar: o que seria da compra on-line sem uma API? Além de poder encarar sérios problemas relacionados a interferências externas, como cobranças indevidas, basicamente

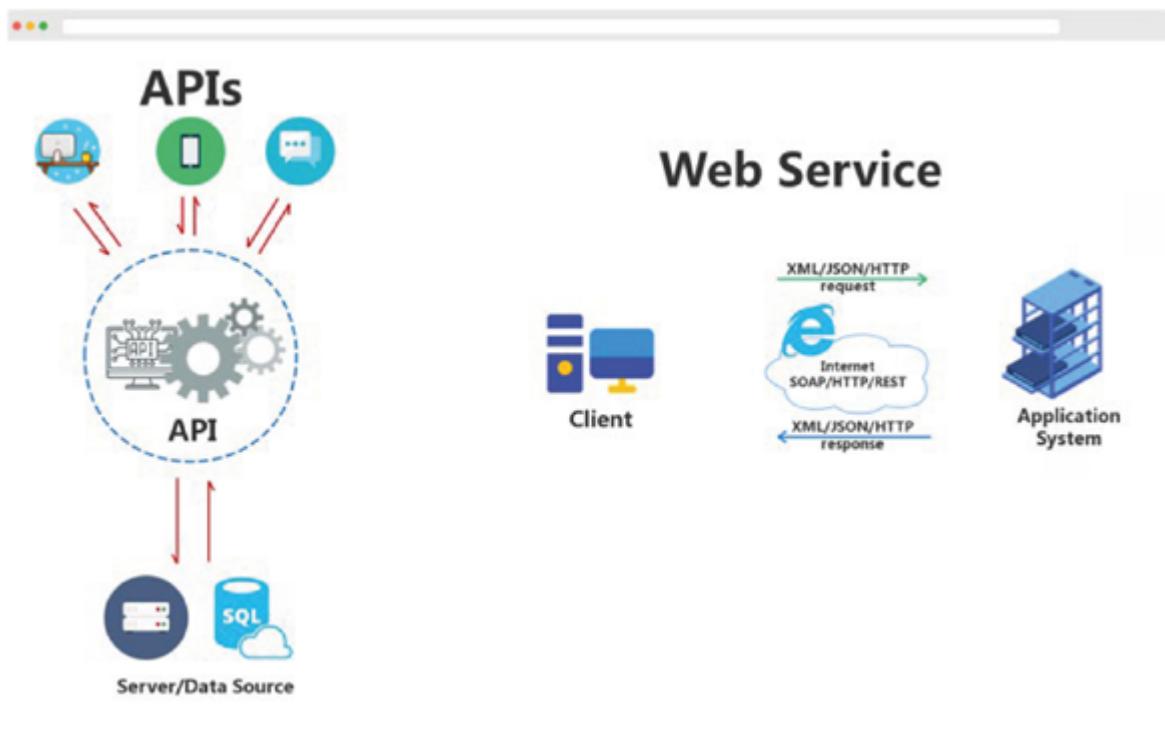
você teria muito trabalho para realizar compras, tendo que informar todas as vezes seus dados, além de se sentir inseguro ao realizar o pagamento e, posteriormente, esse pagamento não for reconhecido. Já com a API, a segurança é aliada da certeza de uma compra efetivada e sem medos de possíveis transtornos.

Sendo assim, uma API é capaz de comunicar diversas plataformas, aplicações e programas para que seja implementada uma determinada funcionalidade, um padrão já existente e, dessa forma, o usuário consegue realizar suas tarefas do cotidiano, por exemplo, pagar contas on-line e acessar redes sociais.

Quando conhecemos as APIs, chegamos a um ponto crucial: o que é Web Services? Provavelmente, você já deve lido ou ouvido falar a famosa frase “todo Web Services é uma API, mas nem toda API é um Web Services”. De fato, essa frase é verdadeira, ambas são ferramentas que auxiliam na comunicação entre aplicações. A diferença é que Web Services é uma aplicação, é o *back-end* de uma determinada aplicação, disponibilizada em um servidor web que poderá ser acessada por um usuário para realizar a comunicação como outro computador. Já a API é uma interface de aplicação que realiza comunicação entre aplicações.

O Web Services utiliza tecnologias, como o HTTP que é um protocolo de transferência de informações, e o WSDL para descrever interfaces de um Web Services. Além das tecnologias, Web Services utilizam arquiteturas para poder funcionar na rede. As arquiteturas mais utilizadas são: a SOAP e a REST. Portanto, o ponto forte de um Web Services é ter que atuar em uma rede, e isso não é obrigatório para uma API. A Figura 2 ilustra a estrutura básica da lógica API e Web Services.

Figura 2 – Web Services e APIs



Fonte: <https://developpaper.com/correctly-distinguish-the-difference-and-connection-between-api-rest-api-restful-api-web-service/>. Acesso em: 25 out. 2021.

PARA SABER MAIS

Já sabemos que um Web Services tem como principal objetivo oferecer diversos tipos de diferentes aplicações, que podem funcionar em vários ambientes de sistemas operacionais e que Web Services foram criados baseados em tecnologias, por exemplo, XML e o protocolo de transporte de dados HTTP, possuindo uma interface padronizada. O que provavelmente você ainda não sabe é a história sobre seu surgimento e evolução.

O Web Service foi criado, padronizado e desenvolvido por uma associação de várias empresas da área da tecnologia. Esse consórcio é conhecido como W3C. Com a criação e o desenvolvimento de um Web Service, a W3C tinha ambição de criar padrões comuns, que poderiam ser utilizados para conteúdo da web. Uma das

grandes vantagens que fortaleceu a W3C foi o apoio que recebeu de empresas poderosas da área da tecnologia. Essas empresas eram nada mais, nada menos que a Microsoft, IBM, HP, Apple entre outras.

Com todas as ferramentas em mãos e contando com um apoio e incentivo de dar inveja, a W3C começou a desenvolver seu primeiro projeto de Web Service. É importante você entender o contexto; por que criar um Web Service? Qual seria a necessidade? E por que o apoio das empresas Microsoft, IBM, HP, Apple entre outras era tão importante? Essas perguntas são fáceis de serem respondidas.

Estamos falando do ano de 1999, ano em que se observou a necessidade de criar padrões de comunicações que poderiam envolver diferentes tipos de computadores, por exemplo, os computadores pessoais, conhecidos como desktops e os *mainframes*, que são supercomputadores. E quem desenvolvia esses computadores? Microsoft, IBM, HP, a Apple e outras. Entendeu porque elas apoiavam a W3C? É fácil observar que elas seriam beneficiadas com a criação de uma ferramenta que auxiliaria na comunicação entre sistemas que eles desenvolviam.

Sendo assim, os sistemas operacionais classificados como softwares proprietários, Mac e Windows, e até mesmo o sistema operacional de software livre Linux eram utilizados em testes na criação de Web Services que estava sendo desenvolvido pela W3C. É importante ressaltar que antes do desenvolvimento e criação do Web Services da W3C, vários outros padrões foram desenvolvidos, como o DCOM e o CORBA, mas, nenhum deles teve sucesso devido a não independência de plataforma e pelo modelo proposto.

Alguns padrões criados possuíam problemas de comunicação e apresentavam riscos de segurança, o que impedia a utilização destes padrões em ambientes como a internet. Foi nesse contexto que

surgiu o Web Services da W3C, com o objetivo de implementar a comunicação entre vários tipos de diferentes ambientes.

TEORIA EM PRÁTICA

Imagine que você é responsável por um setor de TI e precisa auxiliar na análise da implantação de uma ferramenta na empresa, em que você teria que opinar na implantação de um Web Services ou de uma API. A empresa que você trabalha é do ramo de vendas on-line, e a ideia é fazer com que não existam falhas na tentativa de compra de um cliente, principalmente no que diz respeito a pagamentos não reconhecidos.

Diante desta situação, reflita sobre os conceitos, funcionalidades, aplicabilidades e características estudadas relacionadas a Web Services e APIs. Como você responderia à solicitação da empresa? E se você tivesse que montar uma apresentação com a estrutura que deve ser implantada?

Para conhecer a resolução comentada proposta pelo professor, acesse a videoaula deste *Teoria em Prática* no ambiente de aprendizagem.

LEITURA FUNDAMENTAL

Indicações de leitura

Prezado aluno, as indicações a seguir podem estar disponíveis em algum dos parceiros da nossa Biblioteca Virtual (faça o log in por meio do seu AVA), e outras podem estar disponíveis em sites acadêmicos (como o SciELO), repositórios de instituições

públicas, órgãos públicos, anais de eventos científicos ou periódicos científicos, todos acessíveis pela internet.

Isso não significa que o protagonismo da sua jornada de autodesenvolvimento deva mudar de foco. Reconhecemos que você é a autoridade máxima da sua própria vida e deve, portanto, assumir uma postura autônoma nos estudos e na construção da sua carreira profissional.

Por isso, nós o convidamos a explorar todas as possibilidades da nossa Biblioteca Virtual e além! Sucesso!

Indicação 1

O artigo ***Introduction to Web services architecture*** apresenta os principais componentes e padrões associados à arquitetura de Web Services, além de apresentar as diferentes funções associadas a essas arquiteturas. Você ainda verá como os elementos arquitetônicos dos serviços da web são relacionados a um cenário de negócios do mundo real para ilustrar como a abordagem dos Web Services ajudam a resolver problemas de negócios empresariais.

Para realizar a leitura, acesse a nossa plataforma Biblioteca Virtual e busque pelo título da obra no parceiro *ProRequest*.

GOTTSCHALK, K. et al. Introduction to Web services architecture. **IBM Systems Journal**, North Castle, 41.

Indicação 2

No livro *Integração de Aplicações*, você terá a oportunidade de estudar a estrutura e componentes de XML, o conceito, estrutura e

mensagem SOAP, segurança Web Services e os principais problemas relacionados a erros.

Para realizar a leitura, acesse a nossa plataforma Biblioteca Virtual e busque pelo título da obra no parceiro *Minha Biblioteca*.

RODRIGUES, T. N. et al., E. Integração de Aplicações. Porto Alegre: Grupo A, 2020.

QUIZ

Prezado aluno, as questões do Quiz têm como propósito a verificação de leitura dos itens *Direto ao Ponto, Para Saber Mais, Teoria em Prática e Leitura Fundamental*, presentes neste Aprendizagem em Foco.

Para as avaliações virtuais e presenciais, as questões serão elaboradas a partir de todos os itens do Aprendizagem em Foco e dos slides usados para a gravação das videoaulas, além de questões de interpretação com embasamento no cabeçalho da questão.

1. O Web Services utiliza tecnologias como o _____, que é um protocolo de transferência de informações e o WSDL para _____ interfaces um Web Services. Além das tecnologias, o Web Services utiliza _____ para poder funcionar na rede, as mais utilizadas são a _____ e a REST. Portanto, o ponto forte de um Web Services é ter que atuar em uma rede, e isso não é obrigatório para um API.

As lacunas são preenchidas respectivamente por:

- a. HTTP; descrever; arquiteturas e SOAP.
- b. SOAP; arquiteturas; descrever e HTTP.

- c. Arquiteturas; HTPP; descrever e SOAP.
 - d. Descrever; HTTP; arquiteturas e SOAP.
 - e. SOAP; HTTP; arquiteturas e descrever.
2. As APIs são fundamentais na comunicação entre aplicações no mundo atual, sem elas não seríamos capazes, por exemplo, de integrar nossa lista de contato do celular em aplicativos como é o caso do WhatsApp. O trabalho maior seria para o desenvolvedor, que teria que criar interfaces do zero sem poder utilizar padrões já existentes.

Dessa forma, as APIs:

- a. São aplicações.
- b. São sistemas operacionais.
- c. São Interfaces.
- d. É um WhatsApp.
- e. São sistemas criados do zero.



GABARITO

Questão 1 - Resposta A

Resolução: O HTTP é um protocolo de transferência de informações, o WSDL é utilizado para descrever o Web Services. O Web Services utiliza arquiteturas, como a arquitetura SOAP.

Questão 2 - Resposta C

Resolução: APIs são interfaces de aplicações web. As demais alternativas estão incorretas, pois APIs não são aplicações, não são sistemas operacionais, não é o WhatsApp não são sistemas criados do zero.



TEMA 2

Análise de projetos de arquiteturas de APIs e Web Services e seleção de tecnologias para implementação

Autoria: Arthur Gonçalves Ferreira

Leitura crítica: Rennan Martini Rodrigues



DIRETO AO PONTO

Analizar projetos de arquiteturas de aplicações API e de Web Services é um processo extremamente importante e que requer muito cuidado. O usuário deve verificar qual o propósito do seu projeto, para que tipo de ambiente sua API ou Web Services será implementado, dessa forma, não terá riscos de utilizar uma interface ou um programa que não esteja de acordo com a sua necessidade, com a ideia do projeto.

Antes de qualquer escolha, é um princípio básico saber as definições de API e Web Services. Por isso, sempre lembre que API é uma interface de aplicação que vai realizar a comunicação com outra aplicação, oferecendo implementações de atividades que não fazem parte daquela aplicação, por exemplo, implementar linhas de códigos em uma aplicação que realize troca de mensagens entre vários usuários de maneira instantânea.

Web Services, diferentemente de uma API, é um programa que será disponibilizado em um servidor web e que poderá ser acessado por um usuário, por meio de outras máquinas, utilizando um protocolo de comunicação como o protocolo HTTP. Este protocolo por sinal é o protocolo padrão da Web Services. O programa é disponibilizado no servidor web em forma de *back-end*, basicamente é a estrutura do programa, a forma como ele foi desenvolvida, sua codificação.

Entendendo a sua necessidade e lembrando dos conceitos de API e Web Services, você pode começar a escolher, sempre com muito cuidado. Caso o seu projeto seja um projeto que vai depender exclusivamente de uma aplicação que depende de uma rede, você provavelmente deve utilizar uma API web, ou seja, uma Web Services, já que toda API é uma Web Services.

As APIs web possuem características extremamente importantes que devem sempre ser levadas em consideração. Isso porque elas não se limitam somente à utilização do protocolo HTTP para realização da comunicação por meio do envio de mensagens. Em alguns casos, esse protocolo não consegue atingir o objetivo de comunicação de forma eficaz ou simplesmente não consegue realizar a comunicação.

Primeiramente, analise a necessidade do seu projeto. Em muitos casos, o protocolo HTTP pode ser trocado pelos protocolos Websocket, XMPP e MQTT. Todos eles são protocolos de transporte de dados, realizando a comunicação entre aplicações. Este protocolo é uma API muito utilizada para aplicações que possuem o objetivo de realizar uma comunicação on-line, que vai ocorrer em tempo real e que possui o objetivo de atingir vários usuários ao mesmo tempo, é o que ocorre em aplicações de jogos on-line, chat, links de conteúdos esportivos, reuniões e outros.

Já os protocolos XMPP também são uma API, sendo amplamente utilizados em aplicações de mensagens instantâneas, por exemplo, as aplicações já existentes WebEx, GotoMeeting, Facebook Messenger, WhatsApp e Telegram. O XMPP pode realizar o controle de status do usuário, permitindo que o próprio usuário gerencie seu status, oferece também recursos de chamadas de voz, de vídeo e envio de arquivos na comunicação.

O protocolo MQTT é uma API que realiza a troca de mensagens entre máquinas, sendo muito utilizado para construção de APIs web que realizam a troca de pequenas mensagens. É o que ocorre, por exemplo, em aplicações voltados para IoT (Internet das Coisas), Arduino (plataforma que armazena componentes elétricos e programas) e, também, na comunicação via Bluetooth.

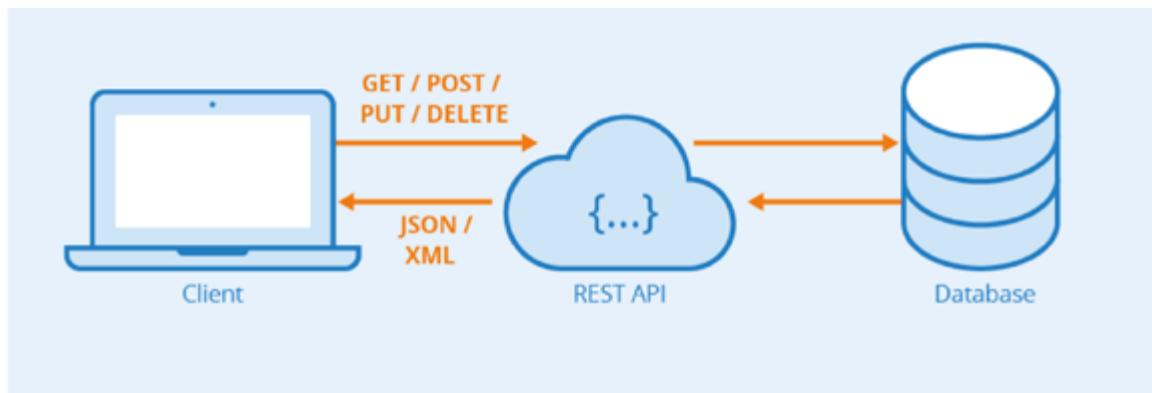
Dessa forma, é possível perceber a importância de conhecer qual a sua necessidade, qual tipo de projeto você está desenvolvendo? Cada projeto deve ser montado de acordo com uma arquitetura de API ou Web Services e muitos detalhes devem ser levados em consideração. Você não deve utilizar um protocolo MQTT, por exemplo, se seu projeto está voltado para a aplicação que vai realizar uma comunicação em tempo real e necessariamente possui o objetivo de alcançar muitos usuários.

Quando se fala em projetos que não vão necessariamente utilizar a rede, podemos pensar em interfaces de APIs RESTful. Lembre-se do conceito: o RESTful é uma arquitetura de API que utiliza como referência a arquitetura rest. Ela pode ser conceituada, também, como um ambiente, uma plataforma ou um sistema em que é utilizado a arquitetura REST.

O RESTful é uma interface disponibilizada em um banco de dados e que pode ser acessada por diferentes aplicações, em diferentes meios, por exemplo, aplicações web, aplicações mobile, aplicações *front end* e aplicações mobile disponibilizadas no meio web, conhecidas como *progressive web applications* (PWA), que em uma tradução livre significa aplicações web progressivas (metodologias de desenvolvimento de softwares híbridos entre aplicativos móveis e páginas web).

Sua análise deve passar por todas as tecnologias que envolvem a implementação de uma API e Web Services. Essas tecnologias estão relacionadas a hardware, software e linguagens de programação. Os hardwares utilizados podem variar entre equipamentos de informática, como computadores, que possivelmente vão possuir características de servidor web, com especificações de hardware avançada relacionada à memória, armazenamento e processamento.

Figura 1 – REST API



Fonte: <https://www.seobility.net/en/wiki/images/f/f1/Rest-API.png>. Acesso em: 25 out. 2021.

Os hardwares envolvem, ainda, equipamentos de rede que precisam estar funcionando para a disponibilização da API ou Web Services; essa rede, portanto, precisa de tecnologias de conexão de internet que podem ser cabeadas ou via Wi-fi. Pode-se citar como tecnologias as linguagens de programação que fazem parte da estrutura do *back-end* de um Web Services e da estrutura de um API. A utilização de linguagens varia muito, sendo as mais utilizadas as linguagens Java, JavaScript, Python, C, C++, C#, PHP e NodeJS.

Resumidamente, as tecnologias utilizadas na implementação de API e Web Services, são conceitos de tecnologias já conhecidas, tais como arquiteturas e protocolos são considerados tecnologias, portanto, temos como REST, SOAP, HTTP, WebSocket, XMPP, MQTT, WSDL e UDDI.



PARA SABER MAIS

É muito importante que você conheça boas práticas para a implementação de determinado estilo de arquitetura, dessa forma, ao final da implementação, terá como resultado um software seguro, escalável e com poder de manutenção sem complicações,

isso significa que seu software poderá receber manutenção sem burocracia, de forma simples. Vamos verificar um pouco quais seriam as boas práticas da implementação de um projeto de arquitetura de API RESTful e SOAP.

A primeira boa prática, independentemente do tipo de implementação de projeto de arquitetura, seria conhecer os conceitos, características e aplicabilidades que envolvem a arquitetura que será implementada. Quando se fala em API RESTful, você deve entender o funcionamento da sua estrutura cliente-servidor. Lembre-se que o cliente-servidor armazena estados transitórios entre as mensagens de pedido e resposta realizados entre as aplicações. Sendo assim, o cliente fica isolado no servidor, existindo maior segurança entre os processos envolvidos.

Já em relação à arquitetura SOAP, lembre-se dos conceitos relacionados, suas características estão ligadas à implementação de uma Web Services, possui o padrão saída de informação no formato XML, utiliza o protocolo HTTP como principal meio de transmissão de dados, mas pode utilizar outros, por exemplo, Websockets e MQTT.

Uma das melhores boas práticas é entender quando devo e posso utilizar o RESTful ou o SOAP, sendo assim, utilize o RESTful quando:

- Tem aplicações com serviços web sem estado.
- O desempenho da aplicação pode ser aproveitado pelo armazenamento em cache.
- Quando se sabe o contexto, o ambiente e a função do que vai ser disponibilizado, desenvolvido.
- Quando existir limitação de sinal de internet, onde a largura da banda não é muito grande.

Já a arquitetura SOAP, deve ser utilizada quando:

- Um *back-end* é disponibilizado em um servidor web.
- Quando quer disponibilizar segurança entre serviços web, com padronização de troca de mensagens, podendo ser protegidas por tecnologias como um token.
- Quando queremos empacotar metadados em cabeçalhos de mensagens SOAP.

Entendendo as arquiteturas e suas características, você será capaz de executar as boas práticas e realizar o desenvolvimento de um projeto API ou Web Services de maneira eficiente, onde a aplicação funcionará de acordo com o objetivo que foi projetada, satisfazendo o cliente.

TEORIA EM PRÁTICA

Realizar a análise de um projeto de arquitetura de API ou Web Service requer muito cuidado para que você possa desenvolver uma aplicação eficiente. Escolher a API, lembrando que toda API é uma Web Services, que não está de acordo com a aplicação que está sendo desenvolvida, coloca em risco a funcionalidade do projeto, bem como sua segurança.

Imagine que você foi contratado para desenvolver um projeto para uma instituição de ensino, onde existem tutores on-line, disponíveis em tempo real, para tirar dúvidas de alunos, dar suporte relacionado ao material didático, dúvidas de aulas etc. Neste é necessária uma plataforma, um ambiente e um programa que realize a troca de mensagens em tempo real, em que vários alunos podem interagir ao mesmo tempo com um mesmo tutor.

Reflita sobre a situação colocada no texto e analise que tipo de projeto de arquitetura você propõe para o desenvolvimento desta plataforma. É necessário escolher entre implementar uma API web ou API, escolher uma arquitetura SOAP ou API RESTful. Qual seria sua posição? Apresente um esquema da estrutura da arquitetura escolhida.

Para conhecer a resolução comentada proposta pelo professor, acesse a videoaula deste *Teoria em Prática* no ambiente de aprendizagem.

LEITURA FUNDAMENTAL

Indicações de leitura

Prezado aluno, as indicações a seguir podem estar disponíveis em algum dos parceiros da nossa Biblioteca Virtual (faça o log in por meio do seu AVA), e outras podem estar disponíveis em sites acadêmicos (como o SciELO), repositórios de instituições públicas, órgãos públicos, anais de eventos científicos ou periódicos científicos, todos acessíveis pela internet.

Isso não significa que o protagonismo da sua jornada de autodesenvolvimento deva mudar de foco. Reconhecemos que você é a autoridade máxima da sua própria vida e deve, portanto, assumir uma postura autônoma nos estudos e na construção da sua carreira profissional.

Por isso, nós o convidamos a explorar todas as possibilidades da nossa Biblioteca Virtual e além! Sucesso!

Indicação 1

O artigo Web Services REST Conceitos, análise e implementação como objetivo compreender questões relacionadas à uma análise e conceituação do estilo arquitetural REST proposto por Roy Thomas Fielding.

Para realizar a leitura, pesquise o título da dissertação no buscador de sua preferência e acesse o diretório da instituição.

BERKENBROCK, C. D. M. **Investigação e Implementação de Estratégias de Notificação de Invalidação para Coerência de Cache em Ambientes de Computação Móvel sem Fio.** 2005. 87 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Santa Catarina, Florianópolis, 2005.

Indicação 2

O capítulo indicado tem como foco o estudo relacionado à integração de aplicações. Nesta leitura, é possível entender mais sobre as características de implementação de arquitetura SOAP e REST.

Para realizar a leitura, acesse a nossa plataforma Biblioteca Virtual e busque pelo título Integração de aplicações no parceiro Minha Biblioteca.

RODRIGUES, T. N. et al. **Integração de Aplicações.** Porto Alegre: Grupo A, 2020.

QUIZ

Prezado aluno, as questões do Quiz têm como propósito a verificação de leitura dos itens *Direto ao Ponto, Para Saber Mais, Teoria em Prática e Leitura Fundamental*, presentes neste Aprendizagem em Foco.

Para as avaliações virtuais e presenciais, as questões serão elaboradas a partir de todos os itens do Aprendizagem em Foco e dos slides usados para a gravação das videoaulas, além de questões de interpretação com embasamento no cabeçalho da questão.

1. Analisar arquiteturas de aplicações API e de Web Services é um processo extremamente importante e que requer muito cuidado. O usuário deve verificar qual o _____ da implementação da arquitetura e para que tipo de _____ sua API ou Web Services será implementado, dessa forma não terá riscos de utilizar uma interface ou um programa que não esteja de acordo com a sua _____ e com a ideia do _____.
 - a. Projeto; necessidade; ambiente; propósito.
 - b. Propósito; ambiente; necessidade; projeto.
 - c. Necessidade; ambiente; projeto; propósito.
 - d. Ambiente; projeto; propósito; necessidade.
 - e. Necessidade; projeto; propósito; ambiente.

2. Para realizar o desenvolvimento de um projeto API ou Web Services de maneira eficiente, onde a aplicação funcionará de acordo com o objetivo que foi projetada, satisfazendo o cliente, é necessário realizar uma análise completa de muitas características que devem ser levadas em consideração. A análise realiza o levantamento de aspectos que podem fazer com que seu projeto possa ser desenvolvido da maneira correta, sem que você possa ter prejuízo em ter que realizar várias alterações no meio do caminho. Sobre a análise que deve ser feita para o desenvolvimento de um projeto API ou Web Services, assinale a alternativa correta:

- a. Leva em consideração somente aspectos tecnológicos.
- b. Não inclui o fator internet ou banda larga.
- c. Considera apenas as linguagens de programação que podem ser usadas.
- d. Analisa tecnologias que envolvem hardwares e softwares.
- e. Realiza a análise de quantos programadores serão necessários para o desenvolvimento do projeto.



GABARITO

Questão 1 - Resposta B

Resolução: Na implementação de aplicações API e de Web Services, o usuário deve verificar o propósito e o ambiente para implementação para que o tudo possa ser desenvolvido de acordo com a necessidade do cliente e com a ideia do projeto.

Questão 2 - Resposta D

Resolução: A análise para a implementação de uma API ou Web Services deve ser completa e relaciona aspectos de hardwares e softwares, todas as tecnologias que estão envolvidas no projeto, desde computadores, servidores, a serviços de rede e internet que vão disponibilizar a API e Web Service.



TEMA 3

Consumo das APIs e Web Services por soluções do *front end*

Autoria: Arthur Gonçalves Ferreira

Leitura crítica: Rennan Martini Rodrigues



DIRETO AO PONTO

O consumo de APIs e Web Services é o processo final da construção de um projeto. Ao contrário do que muita gente pensa, este processo não é o mais difícil. Se você tem um projeto bem definido e estruturado, foi idealizado, planejado e bem executado, o consumo torna-se essencialmente simples e tranquilo.

As APIs e Web Services podem ser consumidas por um navegador web, o que necessariamente torna o processo um pouco mais complicado, isso porque será preciso utilizar uma linguagem de programação para montar uma estrutura, geralmente uma página HTML e outra JavaScript. É possível, também, consumir uma API e Web Services a partir de uma plataforma, um framework, o que torna o processo bem mais simples e fácil.

Independentemente do tipo de consumo que você optar para sua API ou Web Services, você deve entender todos os processos que envolvem esse consumo, por exemplo, se optar por um consumo via plataforma, você deve conhecer qual a plataforma mais indicada. Uma dica seria pesquisar pela plataforma mais utilizada no momento e claro, você deve estudar a plataforma para poder utilizá-la para o consumo de sua API e Web Services, verificando como realizar requisições e outros processos.

Para consumir uma API ou uma Web Services, você deve entender o funcionamento das requisições que podem ser realizadas. Essas requisições estão diretamente ligadas ao que conhecemos como verbos de HTTP. É muito importante ressaltar que, dependendo do consumo que você esteja utilizando, as requisições podem ser um serviço menos braçal, isso vai depender se você vai consumir por uma aplicação ou pelo navegador. A Figura 1 mostra as requisições que podem ser realizadas no consumo de uma API ou Web services.

Figura 1 – Requisições em APIs e Web Services



The screenshot shows a Windows Notepad window titled "Exemplo de requisições - Bloco de Notas". The window contains the following Java code:

```
});  
.post((req,res) -> {  
});  
});  
});  
.get((req,res) -> {  
});  
});  
.put((req,res) -> {  
});  
});  
.delete((req,res) -> {  
});
```

The code demonstrates the use of Java's `RequestBuilder` methods (`.post`, `.get`, `.put`, `.delete`) to handle HTTP requests. The Notepad window also shows standard status bar information: Ln 5, Col 3, 100%, Windows (CRLF), and UTF-8.

Fonte: captura de Boco de Notas.

Dessa forma, torna-se essencial entender todas as requisições que podem ser utilizadas no consumo de uma API e Web Services, porque elas auxiliam muito no processo do consumo.

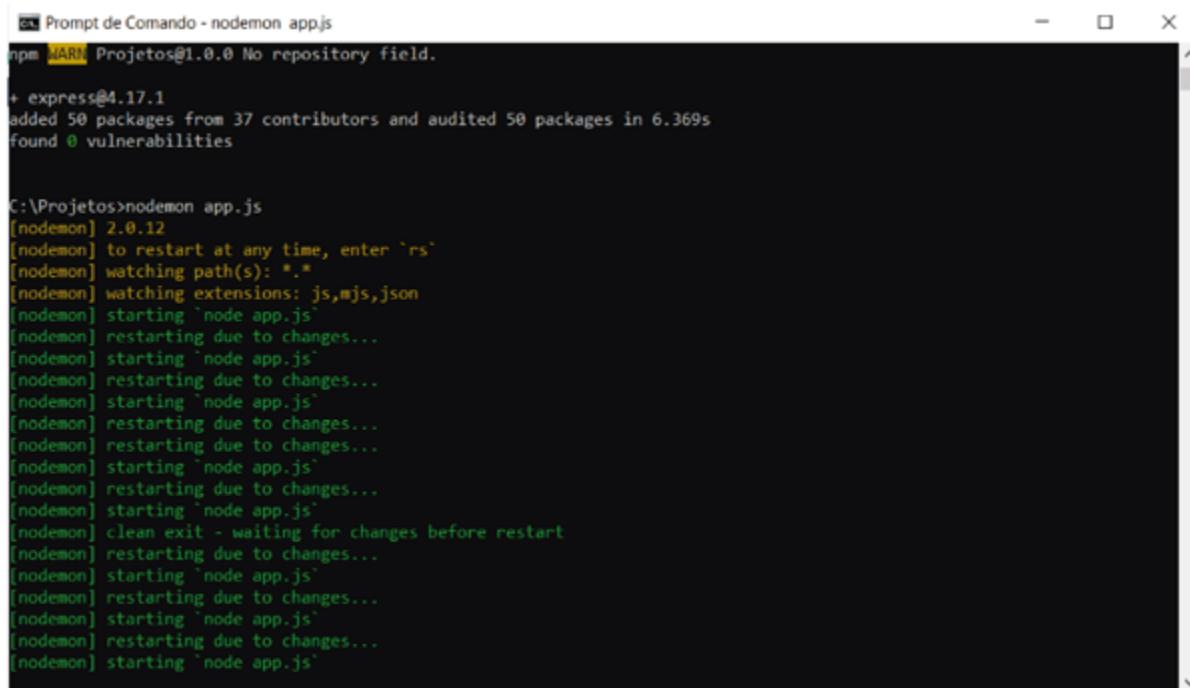


PARA SABER MAIS

Existe um termo muito importante e interessante no contexto de consumo de API e Web Services. Na realidade, este termo está muito ligado à ferramenta Node.js e auxilia os programadores para que eles não possam ter muito trabalho no desenvolvimento do projeto de APIs e Web Services, bem como os seus consumos. O termo em questão é o **nodemon**.

O nodemon é um pacote que deve ser instalado dentro da pasta do projeto que você está desenvolvendo. Este não é um pacote qualquer, é o pacote que trabalha diretamente com a atualização automática do servidor quando você realiza a alteração em algum arquivo do seu projeto. Neste caso, ele evita que você precise o tempo todo, a cada modificação, que você tenha que parar o servidor e reiniciá-lo. A Figura 2 ilustra a ação do nodemon.

Figura 2 – Nodemon



```
C:\ Prompt de Comando - nodemon app.js
npm WARN Projetos@1.0.0 No repository field.

+ express@4.17.1
added 50 packages from 37 contributors and audited 50 packages in 6.369s
found 0 vulnerabilities

C:\Projetos>nodemon app.js
[nodemon] 2.0.12
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
[nodemon] clean exit - waiting for changes before restart
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
[nodemon] restarting due to changes...
[nodemon] starting `node app.js`
```

Fonte: captura de tela de nodemon

Desta forma você pode entender a importância do pacote nodemon para a construção de um projeto de API e Web Services, bem como a sua importância para o consumo das mesmas.

TEORIA EM PRÁTICA

Uma empresa trabalha com o desenvolvimento de APIs e Web Services e está com uma demanda muito grande, precisando de desenvolvedores para trabalhar em projetos. Você foi contratado por essa empresa e começou seu trabalho desenvolvendo um projeto de API utilizando o Node.js, mas tem tido muitas dificuldades, pois nunca tinha utilizado esta plataforma para desenvolver APIs, mas, é uma exigência da empresa que você utilize.

Reflita sobre o texto acima e analise de que forma você consegue vencer as barreiras e as dificuldades da utilização do Node.js. Verifique o passo a passo para desenvolver um bom projeto com esta ferramenta. De que forma você se organizaria? Quais pacotes você precisa instalar para poder desenvolver sem dificuldades?

Para conhecer a resolução comentada proposta pelo professor, acesse a videoaula deste *Teoria em Prática* no ambiente de aprendizagem.

LEITURA FUNDAMENTAL

Indicações de leitura

Prezado aluno, as indicações a seguir podem estar disponíveis em algum dos parceiros da nossa Biblioteca Virtual (faça o log

in por meio do seu AVA), e outras podem estar disponíveis em sites acadêmicos (como o SciELO), repositórios de instituições públicas, órgãos públicos, anais de eventos científicos ou periódicos científicos, todos acessíveis pela internet.

Isso não significa que o protagonismo da sua jornada de autodesenvolvimento deva mudar de foco. Reconhecemos que você é a autoridade máxima da sua própria vida e deve, portanto, assumir uma postura autônoma nos estudos e na construção da sua carreira profissional.

Por isso, nós o convidamos a explorar todas as possibilidades da nossa Biblioteca Virtual e além! Sucesso!

Indicação 1

O livro *Beginning API Development with Node.js : Build Highly Scalable, Developer-Friendly APIs for the Modern Web with JavaScript and Node.js* ajudará você a entender os principais conceitos relacionados ao desenvolvimento e construção de projetos de APIs com Node.js.

Para realizar a leitura, acesse a nossa plataforma Biblioteca Virtual e busque pelo título da obra no parceiro *Proquest*.

NANDAA, A. **Beginning API Development with Node.js:** Build Highly Scalable, Developer-Friendly APIs for the Modern Web with JavaScript and Node. js. Birmingham: Packt Publishing, 2018.

Indicação 2

A monografia *Construção de aplicações distribuídas utilizando-se de APIs e Rest*, aborda conceitos básicos e importantes sobre APIs e Rest, além de consumo de APIs, que o ajudarão no entendimento sobre o assunto.

Para realizar a leitura, pesquise pelo título da monografia no seu buscador de preferência e acesse o repositório da instituição.

OLIVEIRA, Á. G. G. D. **Construção de aplicações distribuídas utilizando-se de APIs REST**. 2018. 64 f. Monografia (Bacharelado em Ciência da Computação) – Universidade do Estado do Rio Grande do Norte, Mossoró, 2018.

QUIZ

Prezado aluno, as questões do Quiz têm como propósito a verificação de leitura dos itens *Direto ao Ponto, Para Saber Mais, Teoria em Prática e Leitura Fundamental*, presentes neste Aprendizagem em Foco.

Para as avaliações virtuais e presenciais, as questões serão elaboradas a partir de todos os itens do Aprendizagem em Foco e dos slides usados para a gravação das videoaulas, além de questões de interpretação com embasamento no cabeçalho da questão.

1. Leia o texto abaixo:

“Em comparação com os servidores comuns, em cada conexão, é alocado um espaço de memória. As requisições de cada usuário são enfileiradas e processadas na devida ordem, o que gera certo atraso na resposta da requisição, porque as requisições não são processadas em paralelo, ou seja, o servidor por sua vez é sobrecarregado e a eficiência do serviço ou sistema é comprometida, isto no caso de um aumento de tráfego e normalmente este problema é contornado adicionando mais servidores ou adicionando mais recursos de hardware para aumento da memória e maior capacidade de processamento. Para solucionar este tipo de problema, foi

criado o Node.js, uma plataforma de software criada em 2009 por Ryan Dahl através do uso da *engine V8 JavaScript*".

CARRENH, R. M. Criando serviço de microblog com Node.js.
DevMedia, [s.l.], 2014.

Sobre a análise do conceito de Node.js, assinale a alternativa correta:

- a. O Node.js é uma linguagem de programação.
 - b. Com o node.js não é necessário um servidor local.
 - c. O Node.js utiliza a linguagem Java.
 - d. Utilizamos o pacote nodemon com o Node.js.
 - e. É a evolução da linguagem de programação JavaScript.
-
2. O _____ é uma plataforma em que é desenvolver diversas aplicações com a linguagem _____. E por que você deve utilizar essa plataforma? Para poder consumir _____ ou _____. A linguagem utilizada nessa plataforma mantém o projeto simples, com a possibilidade de realizar possíveis manutenções de uma maneira mais fácil.
 - a. JavaScript; Web Services; APIs; node.js.
 - b. Node.js; JavaScript; APIs; Web Services.
 - c. APIs; node.js; Web Services; JavaScript.
 - d. Web Services; APIs; Node.js; JavaScript.
 - e. JavaScript; APIs; Node.js; Web Services.



GABARITO

Questão 1 - Resposta D

Resolução: O pacote nodemon é utilizado no Node.js para realizar a atualização automática do servidor toda vez que um arquivo do projeto for alterado. O Node.js é uma plataforma que utiliza um servidor local e não uma linguagem de programação.

Questão 2 - Resposta B

Resolução: O node.js é uma plataforma que utiliza a linguagem JavaScript para desenvolver aplicações. Ela auxilia no consumo de APIs ou Web Services.



TEMA 4

Decisões tecnológicas para o desenvolvimento e testes de APIs e Web Services

Autoria: Arthur Gonçalves Ferreira

Leitura crítica: Rennan Martini Rodrigues



DIRETO AO PONTO

As APIs e Web Services são tecnologias extremamente importantes e utilizadas mundialmente. Essas tecnologias podem ser desenvolvidas por meio de ferramentas, embora um Web Service possa ser desenvolvido sem a utilização de ferramentas. Entretanto, este não é um procedimento padrão e nem é o recomendável, já que tornaria o desenvolvimento muito mais difícil e que poderia comprometer a implementação da web services.

Existem diversas ferramentas que para realizar o desenvolvimento de Web Services, cada uma delas possui suas funcionalidades e características. Cada desenvolvedor deve analisar qual ferramenta adequa-se ao seu projeto; neste caso, não existe uma ferramenta padrão. O que deve ser levado em consideração são as funções da Web Service que será desenvolvida e pequenos detalhes, como escolher uma ferramenta que utilize comunicação HTTP e que trabalhe com o tipo de documento XML.

As ferramentas mais utilizadas para o desenvolvimento de Web Services são de grandes empresas renomadas e conhecidas no mercado, embora as ferramentas produzidas por elas não desqualifiquem em nada as menos conhecidas. Podem ser citadas as ferramentas de desenvolvimento e teste de web services estão a framework .net e as ferramentas Java Apache Axis, JAXRPC, IBM Web service Toolkit e Java Web Service Developer Pack. O Quadro 1 apresenta as ferramentas utilizadas para o desenvolvimento e teste de Web Services com suas características.

Quadro 1 - Ferramentas para o desenvolvimento e testes de Web Services

Ferramenta	Característica
Framework .Net.	Oferece automatização de tarefas realizando a geração de classes para o desenvolvimento e teste de Web Services.
Apache Axis.	Possui uma estrutura de serviço da web de código aberto baseada em XML. Ele consiste em uma implementação em Java e C ++ do servidor SOAP, além de vários utilitários e APIs para gerar e implantar aplicativos de serviço da web.
JAXRPC.	Faz com que uma aplicação Java chame por um Web Service estruturado em Java, consistente e com sua descrição no WSDL.
IBM Web service Toolkit.	Possui os componentes UDDI Java API (UDDI4J), Java2WSDL, Apache AXIS, Xerces e WebSphere Application Server.
Java Web Service Developer Pack.	Auxilia o desenvolvedor realizando a geração de documentos WSDL, usa o RPC para realizar chamadas de métodos em serviços disponíveis, usa o SOAP para envio e recebimento de mensagens e publica um serviço no <i>service providers</i> .

Fonte: elaborado pelo autor.

As APIs também possuem ferramentas para o seu desenvolvimento e teste. A escolha dessas ferramentas segue a mesma linha de raciocínio das web services, não existe um padrão. Para desenvolver uma API são necessárias a utilização de ferramentas que trabalhem com o protocolo HTTP e suas requisições, por exemplo, o GET, POST, PUT ou DELETE.

Também é necessário escolher ferramentas de desenvolvimento de APIs relacionadas à linguagem de programação, ao desenvolvimento estrutural da API, isso porque podem ser utilizadas várias linguagens de programa em um desenvolvimento, por exemplo, PHP, Java, JavaScript, Python e outras.

No que diz respeito ao teste de APIs, podemos citar algumas ferramentas essenciais, como SoapUI, Postman, Katalon Studio e Rest-Assured. Cada ferramenta deve ser analisada para que possa verificar qual delas se adequa aos objetivos da API e, dessa forma, o teste seja realizado de forma mais eficaz.

PARA SABER MAIS

No contexto de desenvolvimento e teste de APIs e Web Services, existe uma verdade absoluta: não existe ferramenta perfeita para realizar o desenvolvimento e teste. Você deve se perguntar como isso é possível. Não importa o quanto você ache estranho, a afirmativa de fato é verdadeira. Existem diversos programas que realizam o desenvolvimento e teste de APIs e Web Services, porém, nenhuma delas vai satisfazer por completo todas as necessidades que seu projeto precisa. Esse problema não é restrito a essas ferramentas, mas é muito comum como a maioria das soluções de desenvolvimento de software.

Se engana quem pensa que partir para soluções comerciais e pagas vai solucionar esse problema – exemplos dessas soluções são o Postman e Tricentis Tosca. Essas ferramentas são muito conceituadas, importantes por realizar diversas ações e auxiliarem no desenvolvimento e testes de APIs e Web Services, porém, até elas podem deixar pequenas brechas sem solução, sem contar o custo para adquirir essas ferramentas com todas as funcionalidades.

Já as ferramentas de código aberto, conhecidas como *open-source*, como o JMeter, Rest-Assured e Karate DSL, necessitam de recursos e esforço para serem utilizadas corretamente. Algumas ferramentas parecem ter um equilíbrio entre custo e outros fatores,

como a ferramenta Katalon Studio e o Postman, porém, acabam se adequando em alguns projetos específicos e em outros não.

Como resolver esse problema que parece insolucionável? Você precisa conhecer todas as ferramentas possíveis que existem para que você possa eleger aquelas mais adequadas ao seu projeto e que, possivelmente, solucionaram, senão por completo, mas o mais próximo disso possível, todos os problemas da sua aplicação. Dessa forma, você passa a estruturar fortemente seu desenvolvimento, solucionando problemas considerados críticos.

Referências bibliográficas

ALDER, A.; LOPES, M. 10 Ferramentas de Teste de APIs para você conhecer. **InfoQ**, [s.l.], 21 de agosto de 2018. Disponível em: <<https://www.infoq.com.br/articles/10-ferramentas-teste-api/>>. Acesso em: 10 set. 2021.

TEORIA EM PRÁTICA

Uma empresa de software está recrutando pessoas para trabalhar na área de desenvolvimento, testes e documentação de APIs e Web Services. Você precisa realizar uma prova prática em que terá que explicar o passo a passo de boas práticas de documentação de softwares além de indicar ferramentas que auxiliam neste procedimento.

Reflita sobre o problema proposto e responda, de acordo com seus estudos, quais seriam boas práticas para o desenvolvimento e testes de APIs e Web Services? Aponte pelo menos uma ferramenta que realiza a documentação desse desenvolvimento.

Para conhecer a resolução comentada proposta pelo professor, acesse a videoaula deste *Teoria em Prática* no ambiente de aprendizagem.

LEITURA FUNDAMENTAL

Indicações de leitura

Prezado aluno, as indicações a seguir podem estar disponíveis em algum dos parceiros da nossa Biblioteca Virtual (faça o log in por meio do seu AVA), e outras podem estar disponíveis em sites acadêmicos (como o SciELO), repositórios de instituições públicas, órgãos públicos, anais de eventos científicos ou periódicos científicos, todos acessíveis pela internet.

Isso não significa que o protagonismo da sua jornada de autodesenvolvimento deva mudar de foco. Reconhecemos que você é a autoridade máxima da sua própria vida e deve, portanto, assumir uma postura autônoma nos estudos e na construção da sua carreira profissional.

Por isso, nós o convidamos a explorar todas as possibilidades da nossa Biblioteca Virtual e além! Sucesso!

Indicação 1

Ao realizar a leitura do artigo *Uma ferramenta para verificação e validação de APIs RESTful* você vai entender a importância e a necessidade da utilização uma ferramenta de software que auxilie desenvolvedores na especificação, validação e verificação de APIs que seguem o estilo REST (APIs RESTful).

Para realizar a leitura, acesse e pesquise pelo título do artigo no seu buscador de preferência e acesse o repositório da revista.

COSTA, B.; CASTRO, I. D.; IMENES, R. ConsoleBox – Uma ferramenta para verificação e validação de APIs RESTful. **Revista da JOPIC**, Teresópolis, v. 1, n. 1, p. 8, 2016.

Indicação 2

A dissertação *Uma ferramenta de apoio ao desenvolvimento de web service* ajudará você a entender como ocorre o desenvolvimento de Web Services por meio de uma ferramenta. Você terá a oportunidade de verificar de que forma pode ser realizada a escolha de uma ferramenta e sua utilização.

Para realizar a leitura, pesquise pelo título da dissertação no seu buscador de preferência e acesse o link do repositório da instituição.

MENÉNDEZ, A. I. M. **Uma ferramenta de apoio ao desenvolvimento de Web Services**. 2002. 98 f. Dissertação (Mestrado em Ciências) – Centro de Ciências e Tecnologia, Universidade Federal de Campina Grande, Campina Grande, 2002.



QUIZ

Prezado aluno, as questões do Quiz têm como propósito a verificação de leitura dos itens *Direto ao Ponto, Para Saber Mais, Teoria em Prática e Leitura Fundamental*, presentes neste Aprendizagem em Foco.

Para as avaliações virtuais e presenciais, as questões serão elaboradas a partir de todos os itens do Aprendizagem em Foco e dos slides usados para a gravação das videoaulas, além de questões de interpretação com embasamento no cabeçalho da questão.

- O desenvolvimento de API é um processo em que existe a necessidade de tomar vários cuidados, isso porque ao fim da implementação deve-se ter uma aplicação simples, de fácil entendimento e de fácil acesso, para que todos possam acessar e utilizar as funcionalidades existentes nela. Dessa forma, a API deve estar toda documentada para que um usuário possa entender suas funcionalidades.

Sobre as ferramentas que realizam a documentação de uma API, assinale a alternativa correta que aponta uma ferramenta que realiza a documentação de uma API.

- Word.
- Writer.
- WordPad.
- Bloco de notas.
- Swagger.

- Se você desenvolveu uma _____ que utiliza o protocolo _____, você deve criar uma _____ que realize a especificação de todos os _____ que podem ser utilizados, em outras palavras, informe quais são as requisições que poder ser utilizadas como, por exemplo, GET, POST, PUT ou DELETE.
 - API; HTTP; documentação; métodos.
 - Métodos; documentação; HTTP; API.
 - Documentação; HTTP; API; métodos.
 - HTTP; API; métodos; documentação.
 - Métodos; API; HTTP; documentação.



GABARITO

Questão 1 - Resposta E

Resolução: Das alternativas apresentadas, a única ferramenta utilizada para documentar uma API, é a ferramenta Swagger, que auxilia não só na documentação, mas também a consumir serviços web RESTful.

Questão 2 - Resposta A

Resolução: Das alternativas apresentadas, a única sequência correta é API, HTTP, documentação, métodos. Isso porque se você desenvolveu uma API que utiliza o protocolo HTTP, é preciso criar uma documentação que realize a especificação de todos os métodos.



BONS ESTUDOS!