



APRENDIZAGEM EM FOCO

---

# ***FRAMEWORKS BACK END***



# APRESENTAÇÃO DA DISCIPLINA

Autoria: Leandro C. Cardoso

Leitura crítica: Arthur Gonçalves Ferreira

Os *frameworks back end* é uma das principais ferramentas que os profissionais de desenvolvimento web *full stack* utilizam em seus projetos. Nesta disciplina, você irá conhecer e obter o direcionamento para saber selecionar *frameworks back end*, nos quais são apropriados para o desenvolvimento de projetos do lado servidor, principalmente para soluções distribuídas por meio do uso de plataforma. A disciplina trará opções de utilização prática dos *frameworks back end*, na criação de APIs para o *front end*, e outras possíveis aplicações consumidoras, como também gerenciar equipes que podem atuar utilizando *frameworks back end*.

De modo simplificado, os *frameworks back end* são conjuntos de códigos de uma linguagem de programação específica que auxiliam o desenvolvimento de projetos como de web, softwares, games, aplicativos, entre outros. A importância do conhecimento *frameworks back end*, pelo fato de possuir muitas vantagens, como o baixo nível de programação e o maior controle da programação do projeto que será desenvolvido. Mesmo com todas as diretrizes dos frameworks é necessário utilizar da criatividade para se diferenciar dos outros projetos.

O Framework Back End Express utiliza a linguagem de programação JavaScript e Node.js. JS; em relação ao JavaScript, trata-se de uma linguagem de programação que o seu uso mais comum é para scripts dinâmicos. Em relação ao Node.js ou simplesmente Node, é considerado um ambiente em tempo que executa, código aberto ou *open-source* e multiplataforma, permitindo para os desenvolvedores a possibilidade de criar diversos tipos de ferramentas e aplicativos

(*backend*) em JavaScript. Além desses *frameworks* nesta disciplina serão apresentados Laravel e Framework Back End Spring Boot, que são excelentes ferramentas para os profissionais que trabalham com desenvolvimento *web full stack*.

Bons estudos!

## 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!

A photograph of a young man with dark, curly hair, smiling broadly. He is wearing a light grey t-shirt. He is in the center of a crowd of other people, who are slightly out of focus. The background shows green trees and a building. The image is overlaid with several geometric shapes: a teal triangle in the top right, a light green triangle on the left, and a dark blue triangle at the bottom. A yellow rectangle is also visible, partially obscured by the teal triangle.

TEMA 1

# Introdução à *frameworks back end*

---

Autoria: Leandro C. Cardoso

Leitura crítica: Arthur Gonçalves Ferreira



## DIRETO AO PONTO

Os profissionais que trabalham com desenvolvimento *web full stack*, é importante o conhecimento de *frameworks back end* e ter parâmetros para saber selecionar para os seus projetos. Seja para a plataforma Android, do Google, seja para a plataforma iOS, da Apple – é importante, primeiramente, possuir noções básicas de *Architecture Information* (AI; em português, arquitetura de informação), de *User Experience* (UX; em português, experiência do usuário), e de *User Interface* (UI; em português, interface do usuário).

No cotidiano, as pessoas deparam com uma grande quantidade de informações. Fica evidente que, quando usamos projetos desenvolvidos com *web full stack*, aplicativos e sites na internet, a arquitetura da informação existe de alguma maneira. A AI (*Architecture Information*), estuda como se organizam as informações de uma interface, com o intuito de facilitar a compreensão do usuário.

Em um projeto de desenvolvimento *web full stack*, o momento do planejamento da AI é importante que deva ser bem prático. Para iniciar o planejamento, noções de UX (*User Experience*) e de UI (*User Interface*) são fundamentais. Observando com maior rigor vários projetos nos quais são facilitados com o conhecimento de *frameworks back end*, é possível perceber o processo de experiência do usuário. Este processo é constituído por cinco camadas, cujas sobreposições vão de um nível mais abstrato a um mais concreto.

A primeira camada, vários autores a denominam como estratégia, e está relacionado ao que esperamos para o produto (website, sistema, aplicativo), mas não apenas aquilo que queremos, mas o que os usuários desejam. A segunda camada recebe o nome de escopo que, na prática, seria o levantamento de requisitos e especificações de funcionalidades e qualidades que o produto deve



ter, e como o projeto será conduzido. A terceira camada recebe o nome de estrutura, em que é importante levar em consideração os seguintes questionamentos: Como o conteúdo deverá ser organizado? O que é menos e o que é mais importante a ser exibido e encontrado pelos usuários? Como encaixar isso nas capacidades e limitações técnicas, mercadológicas e de usabilidade do produto?

A quarta camada pode ser identificada como esqueleto, no qual é o momento em que se desenha a organização dos elementos que constituirão a interface do projeto, os seus componentes – inclusive pensando quais os tipos de componentes mais adequados para exibir as informações e habilitar funcionalidades para os usuários. A quinta e a última camada, estudiosos, autores e profissionais da área denominam como superfície, que nada mais é que a interface final, aquilo que o consumidor, cliente, usuário, terá diante de si, e com a qual vai interagir. Conforme Garrett (2003), é importante conhecer essas camadas para se aprofundar no conhecimento e na escolha de *frameworks back end*.

O planejamento da AI está intimamente ligado à UX. Dessa maneira, existem muitas características emocionais abstratas que devem ser levadas em consideração antes mesmo de iniciarmos a concepção do desenvolvimento *Web full stack*, os profissionais para se diferenciar não devem se limitar apenas ao conhecimento do desenvolvimento de *frameworks*, por exemplo. Pois, são esses aspectos que gastam a maior parte do tempo de planejamento de um projeto, pois é preciso ter a certeza de que a ideia e a estratégia de como dispor o conteúdo do projeto funcionam. A AI se responsabiliza por aspectos mais técnicos: se o projeto apresenta boa navegação; e se responde às solicitações do usuário que interage com a sua interface. Nessa etapa, ainda segundo Garrett (2003), o profissional de desenvolvimento *web full stack* utiliza recursos e técnicas, como o HTML5, para construir a arquitetura da

informação que será utilizada em um aplicativo, em um sistema, em um website etc., facilitado pelo desenvolvimento de *frameworks*.

Com os avanços da tecnologia, que são constantes e velozes, é importante que os profissionais da área tenham sinergia com os departamentos de marketing e de design, para o desenvolvimento de soluções web, de aplicativos mobile que estejam antenados e sintonizados nas constantes inovações, nos quais são facilitados com o desenvolvimento de *frameworks*. É importante, ainda, ser avaliado o perfil do novo consumidor, que tem a característica de, no momento de comprar e/ou de baixar um aplicativo, julgar fatores que, antes, passavam despercebidos, como:

- Pesquisar previamente a reputação da empresa que desenvolve o produto, o serviço ou o aplicativo.
- Buscar informações de como a empresa se comporta na sociedade – por exemplo, se se envolve com causas ambientais e sociais.
- Verificar se os processos subjacentes aos produtos, aos serviços e aos aplicativos mantêm práticas sustentáveis.

Para tais aspectos mercadológicos, embora os profissionais de desenvolvimento *web full stack* não trabalhem diretamente, é importante ter a preocupação e principalmente utilizar *frameworks* para facilitar e otimizar o seu trabalho. A internet é um dos motivos das mudanças de comportamento do consumidor, visto que lhe proporciona uma grande quantidade de informações de forma rápida e facilitada pela interface do usuário. A Figura 1, a seguir, ilustra os diversos *frameworks* de trabalho.

**Figura 1 – Frameworks back end**



Fonte: <https://pixabay.com/pt/illustrations/rede-desenvolvedor-web-1935737/>. Acesso em: 4 out. 2021.

As interfaces desenvolvidas pelos profissionais de desenvolvimento *web full stack*, além de desenvolver projetos web, podem ser aliados com profissionais que trabalham especificamente com design de aplicativos, os quais devem contemplar a criatividade e se aliar a uma estratégia de marketing. O objetivo é tornar o comportamento do consumidor digital favorável à aquisição do produto ofertado. O novo consumidor é muito mais exigente, de maneira que, se forem encontradas quaisquer dificuldades, buscará, em questão de segundos, produtos alternativos e encontrará os concorrentes facilmente. Nesse sentido, utilizar o campo de buscas das lojas dos aplicativos dos sistemas operacionais, como o Google Play e a Apple Store, é uma estratégia que indica games e aplicativos similares aos da busca executada, desta maneira, é importante conhecer as diretrizes para o desenvolvimento de *frameworks* dos principais sistemas operacionais.

Portanto, é necessário entender o comportamento do consumidor digital e o modo como são executadas suas ações na internet, tornando a coleta informações algo de importância fundamental



para o departamento de marketing, que poderá transmitir para os profissionais de desenvolvimento *web full stack* as demandas dos consumidores. Assim, devem ser desenvolvidos projetos bem atrativos, para que os consumidores os utilizem pelo máximo de tempo possível, resultando em uma boa experiência e, conseqüentemente, na aquisição do produto e/ou na execução de seu download.

O uso das tecnologias digitais possibilita novas maneiras de o consumidor relacionar-se com as empresas e com seus produtos. Dependendo do produto, o consumidor será capaz de experimentar e até de interagir. Nos quais são facilitados com conhecimento de *frameworks*, pelos profissionais que trabalham com *web full stack*.

## Referências bibliográficas

GARRETT, J. J. **The elements of the user experience**. 2. ed. New York: New Riders, 2003.



## PARA SABER MAIS

Os *frameworks back end* precisam possuir mecânica funcional com linguagem de programação que evite o máximo de travamento nos diversos dispositivos que serão executados. Ao mesmo tempo, é importante que resultam em projetos de fácil navegação, sendo esse aspecto proporcionado pelos profissionais de desenvolvimento *web full stack*. É importante sempre procurar melhorar a usabilidade de um projeto, agregando diversos valores, fazendo com que se destaque em relação aos outros projetos ou apps dos concorrentes. Dessa forma, é preciso planejar os elementos que compõem a interface nos mínimos detalhes, e como o *framework* irá trabalhar com cada detalhe. Como a escolha correta da tipografia, que deve considerar boa legibilidade, facilitando a leitura das informações

textuais e, se possível, famílias de fontes que conversem com a identidade visual do produto ou serviço que o projeto representa. As fotografias, imagens, composições visuais e ilustrações precisam ter boa qualidade de resolução para que não fiquem distorcida, conforme as adaptações em diferentes dispositivos que serão executados o projeto, ou aplicativo por exemplo. Esses elementos de tipografia, informações textuais e visuais devem ser organizados de maneira coerente no layout e atraente para os usuários, com objetivo de chamar e prender a atenção do usuário pelo maior tempo possível.

No desenvolvimento *web full stack* é importante salientar que o desenvolvedor tem de se colocar na posição do usuário, conhecer seu perfil, nível de conhecimento em relação às tecnologias dos dispositivos e o projeto, por exemplo um aplicativo pode proporcionar. Diante disso, no momento do desenvolvimento do *framework*, é fundamental pensar nas possibilidades de customização para que ele melhor se adapte ao usuário e não o contrário. Esse aspecto melhora a experiência e a satisfação dos usuários do projeto. É importante considerar a composição das cores de fundo e das tipografias e o fato de os usuários poderem utilizar os aplicativos em ambientes diferentes, internos ou externos. Imagine um layout com letras brancas com fundo claro sendo visualizado sob a luz do sol. Seria praticamente impossível visualizar as informações. Embora os próprios dispositivos móveis como os tablets e smartphones tenham controle automático de brilho e contraste das telas, todos esses aspectos técnicos são relevantes e devem ser considerados no planejamento e desenvolvimento do *framework*, embora esteja no *back end*, interfere no *front end*, ou seja, na interface.

O *framework* também precisa atender às particularidades que cada sistema operacional e verificar se é necessário elaborar adaptações das funcionalidades. Nos projetos é importante sempre ter o

objetivo de aprimorar a experiência dos usuários em relação aos tamanhos e adaptações das imagens conforme cada dispositivo móvel, as cores, entre outros elementos. É imprescindível conhecer sobre as cores e manter a identidade visual da empresa, produto ou serviço que o projeto representa, escolhendo a composição e tonalidade ideal. Mesmo que haja inúmeras possibilidades de uso de cores é importante que tenha harmonia com tema do aplicativo, sendo um diferencial para o profissional com este conhecimento.

Para manter a identidade visual em relação às cores, é fundamental consultar o manual da marca da empresa. Caso não exista, a referência inicial é o logotipo da empresa, que é primordial para analisar as cores aplicadas e, se for conveniente, criar paleta com essa referência. Para cada tipo de design de aplicativo, existe uma solução ideal para as cores. O contraste é um aspecto importante que deve ser considerado, pois, além de organizar, ele permite destacar elementos que são importantes além de organizá-los. É possível, por exemplo, aplicar uma tonalidade suave e contrastá-la com outros tons mais escuros, utilizando a quantidade de cores da marca. O importante é testar as opções e verificar quais se adequam melhor ao objetivo do projeto. A simplicidade, em muitos casos, é a solução ideal no desenvolvimento *web full stack* uma vez que evita a poluição visual e conseqüentemente melhora a visualização da marca ou dos elementos que são necessários destacar no layout, e todos esses aspectos devem ser considerados no desenvolvimento de *frameworks*.



## TEORIA EM PRÁTICA

Para o desenvolvimento *web full stack* e especificamente para os *frameworks back end*, é importante o conhecimento das particularidades das plataformas e sistemas operacionais que irão

executar os projetos, como Windows da Microsoft, Android do Google. A Apple oferece binários para MacOS e para Linux, que podem compilar códigos para iOS, para MacOS, para watchOS, para tvOS e para Linux. Elabore um exemplo de como é a sua codificação utilizando uma *String*, cujo objetivo é chegar um resultado de um nome de uma pessoa.

**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 tem o objetivo de apresentar *framework* de referência que possa ser utilizado na construção de modelos organizacionais. O *framework* que é apresentado no artigo é fundamentado na ideia do hipertexto, destacando a incorporação da dimensão espaço na estrutura do *framework*.

LIMA, Edson; LEZANA, Álvaro. Desenvolvendo Um Framework Para Estudar A Ação Organizacional: Das Competências Ao Modelo Organizacional. **Gestão e Produção**, São Carlos v.12, n. 2, p. 177-190, mai./ago. 2005.

## Indicação 2

O artigo refere-se ao resultado de um estudo com objetivo de desenvolver um *framework* que padronizasse processos em um periódico científico, a partir de uma equipe que executa rotinas operacionais reguladas por normas (externas) e padrões (internos e externos).

GARCIA, José A. C. et al. Definição de framework para a padronização de processos em um periódico científico com base em revisão da literatura e nos objetivos de desempenho. **GEPROS. Gestão da Produção, Operações e Sistemas**, Bauru, v. 9, n. 4, p. 153-170, 2014.

## 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. Para o desenvolvimento de *frameworks back end*, é importante que os profissionais de desenvolvimento *web full stack* tenham conhecimentos técnicos da área representados pela sigla:
  - a. AI (*Architecture Information*; Arquitetura de Informação).
  - b. DF (*Development Frameworks*; Desenvolvimento de *Frameworks*).
  - c. SF (*Select Frameworks*; Seleção de *Frameworks*).
  - d. KWF (*Know How Frameworks*; Estudo de *Frameworks*).
  - e. DDF (*Design Digital Framework*).
  
2. Os *frameworks* embora a sua função nativa seja de utilização no *back end*, podem ser utilizados para melhorar a usabilidade dos projetos de qual maneira?
  - a. Por meio da adaptação do projeto para atender as funções dos *frameworks* já disponíveis no mercado.
  - b. Por meio do modo que os *frameworks* irão trabalhar com os detalhes dos elementos da interface.
  - c. Por meio do planejamento do desenvolvimento do projeto tendo como base os *frameworks* já prontos.
  - d. Por meio da seleção dos *frameworks* de usabilidade e posteriormente adaptação ao projeto.
  - e. Por meio da seleção dos *frameworks*, tendo como base de algum projeto finalizado.



## GABARITO

### Questão 1 - Resposta A

**Resolução:** AI (Arquitetura da Informação) analisa como se dá a organização das informações de uma interface, desta maneira, é bem similar de como deve ser o desenvolvimento de *frameworks*, que também deve ser de maneira organizada. Embora saber desenvolver, selecionar, e estudar *frameworks* seja importante as siglas DF, SF, e KWF, não representam conhecimentos técnicos da área.

### Questão 2 - Resposta B

**Resolução:** Os *frameworks* podem ser um facilitador para usabilidade, por meio do planejamento dos principais componentes da interface, e é importante que os *frameworks* se adaptem ao projeto e não ao contrário. Caso não seja encontrado um *framework* que atendam os projetos é recomendado o seu desenvolvimento.



TEMA 2

# Utilização do Framework Express

---

Autoria: Leandro C. Cardoso

Leitura crítica: Arthur Gonçalves Ferreira



## DIRETO AO PONTO

Para criar um aplicativo utilizando Framework Express, é importante observar que o app objeto denota convencionalmente o aplicativo Express, sendo necessário ser criado e chamando a `express()` função de nível superior exportada pelo módulo Express, conforme o exemplo a seguir:

```
var express = require('express')

var app = express()

app.get('/', function (req, res) {

  res.send('aprendendo express')

})

app.listen(4000)
```

Sendo que o app objeto possui métodos para roteamento de solicitações HTTP; `app.METHOD` e `app.param`, é importante também a configuração de *middleware* e o `app.route`, além das renderizações e visualizações HTML `app.render`. E o registro de um mecanismo de modelo, ou *template* pelo `app.engine`, no quais também possui configurações, ou seja, propriedades que afetam o comportamento do aplicativo. Desta forma, o objeto de aplicativo Express pode ser referido a partir do objeto de solicitação e do objeto de resposta como `req.app`, e `res.app`, respectivamente.

Já em relação às propriedades, o `app.locals` objeto possui propriedades que são variáveis de locais dentro do próprio aplicativo como exemplificado a seguir.

```
console.dir(app.locals.title)
```

```
// => 'Meu Aplicativo'
```

```
console.dir(app.locals.email)
```

```
// => 'meunome@meuaplicativo.com.br'
```

Uma vez definido, o valor das `app.locals` as propriedades persistem durante a vida útil do aplicativo, em contraste com as propriedades `res.locals` que são válidas apenas durante a vida útil da solicitação. É possível acessar variáveis locais em modelos renderizados no aplicativo. Isso é útil para fornecer funções auxiliares a modelos, bem como dados em nível de aplicativo, variáveis locais estão disponíveis no middleware via `req.app.locals`. Observe um exemplo a seguir:

```
app.locals.title = 'Meu Aplicativo'
```

```
app.locals.strftime = require('strftime')
```

```
app.locals.email = 'meunome@meuaplicativo.com.br'
```

A propriedade `app.mountpath` contém um ou mais padrões de caminho nos quais um subaplicativo foi montado. Para entender melhor, um subaplicativo é uma instância de Express, nos quais pode ser usada para lidar com a solicitação de uma rota, veja o exemplo a seguir.

```
var express = require('express')
```

```
var app = express() // the main app
```

```
var admin = express() // the sub app
```

```
admin.get('/', function (req, res) {
```

```
    console.log(admin.mountpath) // /admin
```



```
res.send('Admin Homepage')

})
```

```
app.use('/admin', admin) // mount the sub app
```

É semelhante à propriedade base Url do req objeto, exceto que req.baseUrl retorna o caminho da URL correspondente, em vez de os padrões correspondentes. Caso um subaplicativo for montado em vários padrões de caminho, app.mountpath retorna à lista de padrões em que está montado, conforme o exemplo a seguir.

```
var admin = express()
```

```
admin.get('/', function (req, res) {
```

```
    console.dir(admin.mountpath) // [ '/adm*n', '/manager' ]
```

```
    res.send('Admin Homepage')
```

```
})
```

```
var secret = express()
```

```
secret.get('/', function (req, res) {
```

```
    console.log(secret.mountpath) // /secr*t
```

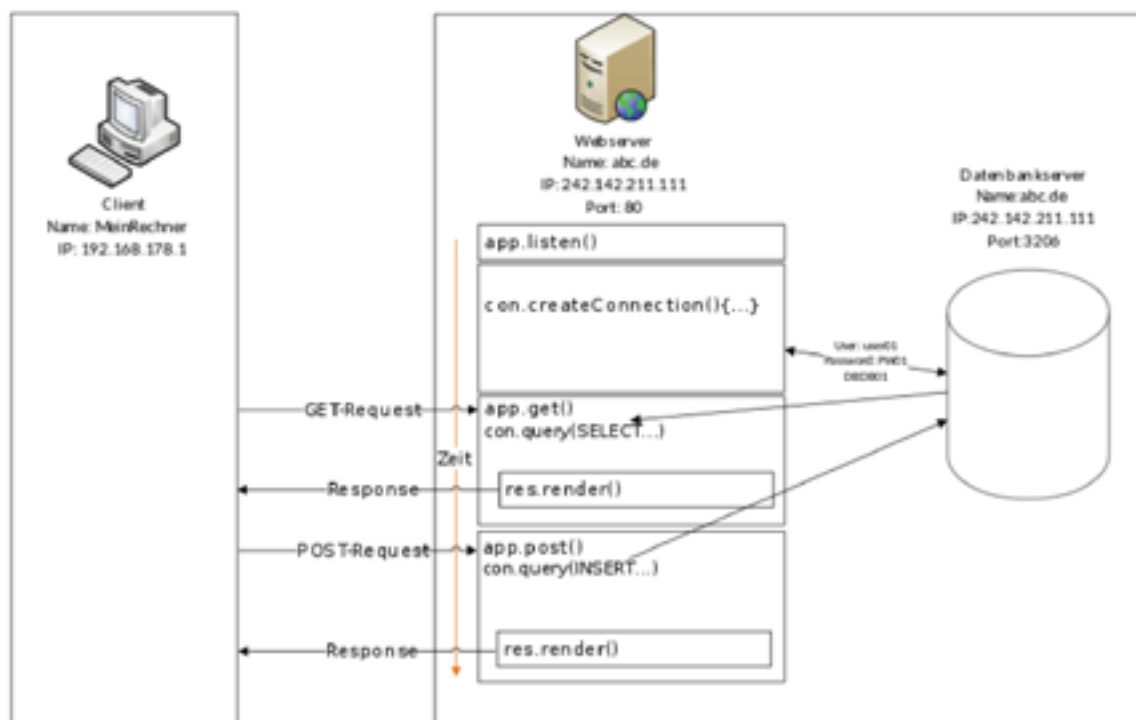
```
    res.send('Admin Secret')
```

```
})
```

```
admin.use('/secr*t', secret) // load the 'secret' router
on '/secr*t', on the 'admin' sub app
```

```
app.use(['/adm*n', '/manager'], admin) // load the 'admin'
router on '/adm*n' and '/manager', on the parent app
```

**Figura 1 – Modelo HTTP-Cliente-Servidor – Node.js-App com Express-Framework**



Fonte: <https://upload.wikimedia.org/wikipedia/commons/4/44/Client-Server-HTTP.svg>.  
Acesso em: 4 out. 2021t.

Desta forma, exemplos mais práticos facilitam a compreensão de Express, que, de maneira resumida, trata-se de um *Framework Back End* com JavaScript e Node.js.

## PARA SABER MAIS

No Framework Express, existem diversos métodos `app.all` que pode definir caminho, *callback* sendo método semelhante aos métodos `app.METHOD()` padrão, exceto que corresponde a todos os verbos HTTP. Observe o quadro a seguir descrevendo argumento, descrição e padrão.

## Quadro 1 – Argumento, descrição e padrão

Argumento	Descrição	Padrão
path	<p>O caminho para o qual a função de <i>middleware</i> é chamada; pode ser qualquer um dos seguintes:</p> <p>Uma string que representa um caminho. Um padrão de caminho. Um padrão de expressão regular para combinar caminhos. Uma matriz de combinações de qualquer uma das opções acima.</p>	'/' (caminho raiz)
callback	<p>Funções de retorno de chamada; pode ser:</p> <p>Uma função de <i>middleware</i>. Uma série de funções de <i>middleware</i> (separadas por vírgulas). Uma série de funções de <i>middleware</i>. Uma combinação de todas as opções acima.</p> <p>Pode fornecer várias funções de retorno de chamada que se comportam como <i>middleware</i>, exceto que esses retornos de chamada podem invocar <code>next('route')</code> para ignorar o (s) retorno de chamada de rota restante. Esse mecanismo pode ser usado para impor pré-condições em uma rota e, em seguida, passar o controle para as rotas subsequentes se não houver motivo para prosseguir com a rota atual.</p> <p>Como o roteador e o aplicativo implementam a interface de <i>middleware</i>, pode usá-los como faria com qualquer outra função de <i>middleware</i>.</p>	Nenhum

Fonte: elaborado pelo autor.

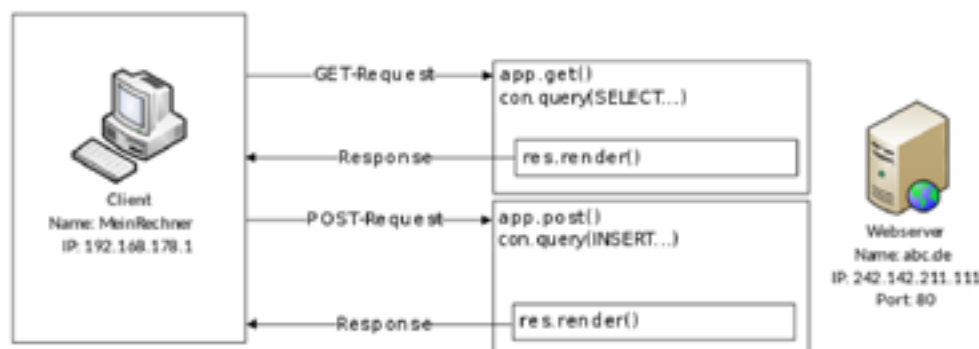
Veja a seguir um exemplo prático, no qual o retorno de chamada é executado para solicitações /secret usando GET, POST, PUT, DELETE ou quaisquer outros métodos de solicitação HTTP:

```
app.all('/secret', function (req, res, next) {  
  
  console.log('Accessando a seção secreta...')  
  
  next() // pass control to the next handler  
  
})
```

Assim, o `app.all()` method é útil para mapear a lógica “global” para prefixos de caminho específicos ou correspondências arbitrárias. Por exemplo, caso coloque o seguinte no topo de todas as outras definições de rota, isso requer que todas as rotas daquele ponto em diante exijam autenticação e carreguem um usuário automaticamente. Lembre-se de que esses retornos de chamada não precisam atuar como pontos finais: `loadUser` pode executar uma tarefa e, em seguida, chamar `next()` para continuar correspondendo as rotas subsequentes.

Observe na Figura 2, a seguir que o roteamento é a forma como um servidor recebe solicitações do cliente e responde novamente.

**Figura 2 – Roteamento com POST e GET na estrutura Express**



Fonte: <https://upload.wikimedia.org/wikipedia/commons/9/94/POST-GET-Express.svg>.  
Acesso em: 4 out. 2021.

## **TEORIA EM PRÁTICA**

O Node.js pode ser usado para criar um simples servidor web, a partir do uso do pacote denominado como Node http, desta forma crie um ser servidor web no qual escuta qualquer tipo de requisição HTTP na URL fictícia `http://157.0.0.1:9000/` em que no momento que a aquisição for recebida, o script responda com a string de texto *“Teoria em Prática”*. Não se esqueça que para desenvolver este servidor é necessário instalar o Node, a partir do terminal, por exemplo, o prompt da linha de comando do sistema operacional Windows, e crie uma pasta como nome *“tep”* e o nome do arquivo como *“teoria”*.

**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

Este trabalho, a partir da página 24, apresenta um aprofundamento do Node.js, mostrando que é uma plataforma construída sobre o motor Javascript baseado em um modelo assíncrono orientado a evento I/O, que suporta execuções concorrentes sem a necessidade de múltiplos *threads*, prevenindo a aplicação de ser bloqueada enquanto espera alguma operação I/O.

Referência completa conforme ABNT:

OLIVEIRA JÚNIOR, Laércio G. de. **ATLOM.JS: um Framework NODE.JS para aplicações Web baseado em componentes**. 2017. 67 f. Trabalho de Conclusão de Curso (Graduação em Engenharia de Software) – UFCE, Quixadá, 2017.

## Indicação 2

Este trabalho corresponde a uma pesquisa sobre a pilha de desenvolvimento MEAN no qual são citados o Node.js, o Framework Express, JavaScript entre outros, no quais é um aprofundamento do conteúdo estudado nesta unidade de ensino.

OLIVEIRA Ariane, *et al.* MEAN STACK: uma solução para o desenvolvimento de aplicações Web. **Interface Tecnológica**, Taquaritinga, v. 17, n. 2, 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. Para o desenvolvimento de um app usando o Express, após ter criado e chamando a `express()` pode ser utilizar a função de que nível?
  - a. Superior.
  - b. Inferior.
  - c. Intermediária.
  - d. Todos os níveis.
  - e. Inferior ou intermediária.

2. Para os profissionais da área é importante conhecer os diversos métodos no Framework Express sendo que um deles podem definir caminho, mas apenas existe uma exceção. Qual seria?
- a. Que corresponde a todos os intransitivos HTTP.
  - b. Que corresponde a todos os verbos HTTP.
  - c. Que corresponde a todos os pronomes HTTP.
  - d. Que corresponde a morfologia HTTP.
  - e. Que correspondem a todos os verbos e pronomes HTTP.



## GABARITO

### Questão 1 - Resposta A

**Resolução:** Após de ser criado e chamando a `express()` é a função de nível superior exportada pelo módulo Express, e não a inferior, ou intermediária, ou todos os níveis cujo não é nem um termo utilizado nas codificações.

### Questão 2 - Resposta B

**Resolução:** Nos diversos métodos `app.all` que pode definir caminho, callback no Framework Express, a única exceção é a corresponde a todos os verbos HTTP, sendo que intransitivos, pronomes, morfologia, não são termos utilizados na codificação de códigos.



TEMA 3

# Framework Back End Laravel

---

Autoria: Leandro C. Cardoso

Leitura crítica: Arthur Gonçalves Ferreira



## DIRETO AO PONTO

Para desenvolvedores experientes, o Laravel oferece ferramentas robustas para injeção de dependência, teste de unidade, filas, eventos em tempo real entre outras, sendo que o container de serviço Laravel é uma ferramenta robusta para gerenciamento de dependências de classes, e executar injeção de dependências. Para entender melhor a injeção de dependência da classe é, de certa forma, “injetada” na classe por meio do construtor ou, em alguns casos, dos métodos “setter”. Observe o exemplo a seguir:

```
<?php

namespace App\Http\Controllers;

use App\Http\Controllers\Controller;

use App\Repositories\UserRepository;

use App\Models\User;

class UserController extends Controller
{

    /**
     * The user repository implementation.
     *
     * @var UserRepository
     */
```



```
protected $users;

/**
 * Create a new controller instance.
 *
 * @param UserRepository $users
 * @return void
 */
public function __construct(UserRepository $users)
{
    $this->users = $users;
}

/**
 * Show the profile for the given user.
 *
 * @param int $id
 * @return Response
 */
public function show($id)
{
```

```
$user = $this->users->find($id);

return view('user.profile', ['user' => $user]);

}

}
```

Neste exemplo, fica fácil de entender que UserController é necessário para recuperar usuários de uma fonte de dados. Neste caso será simulada a situação de injetar um serviço capaz de recuperar usuários. Sendo assim, o UserRepository provavelmente utiliza o Eloquent para recuperação de informações do usuário do banco de dados. Mas, pelo motivo de o repositório ser injetado, permite que seja trocado de maneira fácil por outra implementação. Outra situação é “simular” e executar a criação de implementação fictícia do UserRepository ao testar o aplicativo. O entendimento avançado do contêiner de serviço do Laravel é importante para construção de um aplicativo robusto e, também, para contribuição do próprio núcleo do Laravel. Mas, caso uma classe não possua dependências ou dependa apenas de outras classes concretas, mas não interfaces, o contêiner não precisa ser instruído sobre como resolver essa classe. Observe o exemplo de utilizar a seguinte codificação no routes/web.php arquivo:

```
<?php

class Service

{

    //

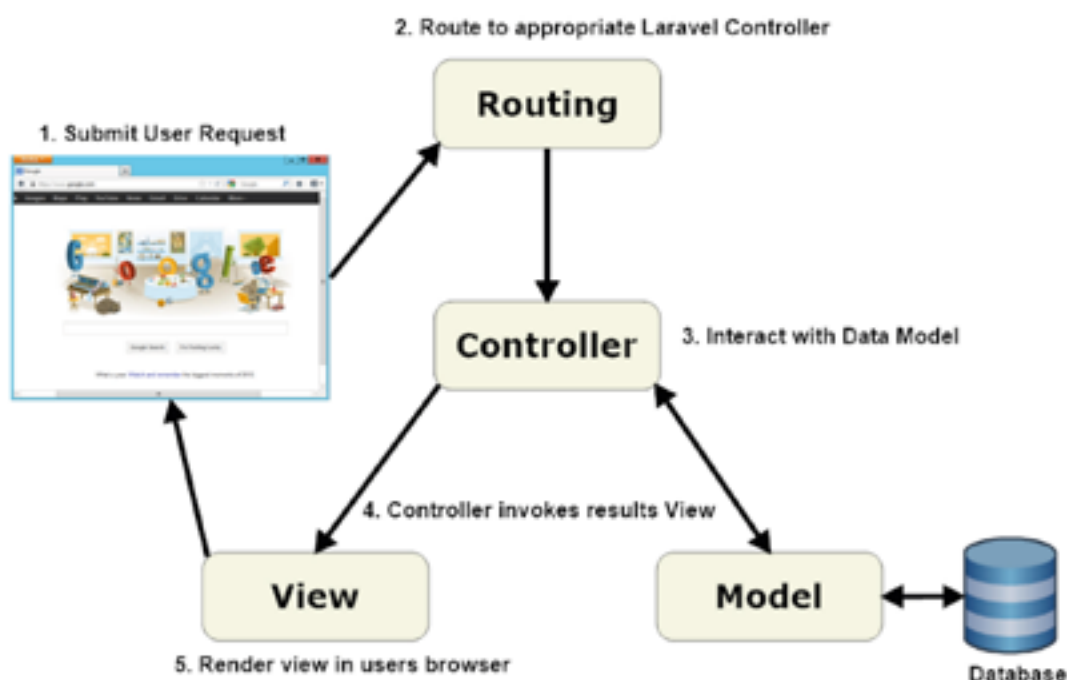
}
```

```
Route::get('/', function (Service $service) {  
  
    die(get_class($service));  
  
});
```

Neste exemplo, atingir a /rota do aplicativo que está em desenvolvimento irá resolver de forma automática Serviceclasse e injetá-la no manipulador de sua rota, podemos considerar como uma mudança de jogo. Desta maneira, permite desenvolver o aplicativo aproveitando as vantagens da injeção de dependência sem se preocupar com arquivos de configuração inchados.

Outro motivo que faz do Laravel o melhor *framework* PHP é que ele suporta arquitetura como MVC (*Model-View-Controller*), garantindo clareza entre lógica e apresentação. MVC melhora o desempenho e tem muitas funções integradas. Veja como o MVC funciona para o Laravel, na Figura 1, a seguir:

**Figura 1 – Suporte MVC (*Model-View-Controller*) no Laravel**



Fonte: [levanphu.info/public//lptech.asia](http://levanphu.info/public//lptech.asia).

Um dos pontos positivos é que várias classes que deverão ser escritas ao construir um aplicativo Laravel recebem automaticamente suas dependências. Por meio do contêiner, incluindo controladores, ouvintes de eventos, *middleware* e muito mais, e também permite que o desenvolvedor insira dependências de dica, no *handle* método de trabalhos em fila.



## PARA SABER MAIS

O Laravel foi construído com vários recursos para execução de testes, na prática, para o suporte para teste com PHPUnit está incluído fora da caixa e um `phpunit.xml`, arquivo já está configurado para o aplicativo que está em desenvolvimento. A estrutura também vem com métodos auxiliares convenientes que permitem testar os aplicativos de forma expressiva, para criar uma nova situação de teste, deve-se utilizar o `make:test` comando Artisan. Por padrão, os testes serão colocados no `tests/Feature` diretório:

```
php artisan make:test UserTest
```

Caso queira desenvolver um teste dentro do `tests/Unit` diretório, pode-se utilizar a `-unit`, opção ao executar o `make:test` comando:

```
php artisan make:test UserTest -unit
```

Sendo que uma vez que o teste foi gerado, é possível definir métodos de teste como faria normalmente usando PHPUnit, para execução de testes, execute o comando `vendor/bin/phpunit` ou `php artisan test` no terminal:

```
<?php
```

```
namespace Tests\Unit;
```

```
use PHPUnit\Framework\TestCase;

class ExampleTest extends TestCase
{
    /**
     * A basic test example.
     *
     * @return void
     */
    public function test_basic_test()
    {
        $this->assertTrue(true);
    }
}
```

Além do phpunit comando, pode utilizar o test comando Artisan para executar os testes, o executor de teste Artisan fornece relatórios de teste detalhados para facilitar o desenvolvimento e a depuração:

```
php artisan test
```

Quaisquer argumentos que podem ser passados para o phpunitcomando também podem ser passados para o test

comando Artisan com o seguinte comando: `php artisan test --testsuite=Feature --stop-on-failure`.

## **TEORIA EM PRÁTICA**

Para os profissionais ou estudantes que trabalham com *Frameworks back end*, é possível desenvolver um projeto no Laravel, utilizando Composer diretamente. Também existem outras possibilidades no qual é de realizar a instalação do instalador como uma dependência global do Composer. Apresente qual é o comando para realizar este processo, levando em consideração que o nome do diretório receberá o nome de “teoria-pratica3”. Apresente, ainda, o comando para acessar o diretório, e qual será o resultado obtido.

**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**

Trabalho sobre o desenvolvimento de uma ferramenta para organização e gerenciamento de atividades de docentes, para o desenvolvimento foram utilizados o *framework* de *front-end* Vue.js e o *back-end framework* Laravel. A partir da página 13 o tema é dedicado ao *framework* Laravel, sendo um aprofundamento dos assuntos aqui apresentados.

MENDES, Rafael. **Desenvolvimento de uma ferramenta para organização e gerenciamento de atividades de docentes**. 2018. 31 f. Trabalho de Conclusão de Curso (Bacharelado em Sistemas da Informação) – Faculdade de Computação, Universidade Federal De Uberlândia, Uberlândia. 2018.

## **Indicação 2**

Este trabalho se relaciona ao desenvolvimento de um sistema web para o gerenciamento de atendimentos no setor de saúde, fundamentado nos princípios do ciclo de vida do software seguindo o modelo de prototipação. Para a codificação do sistema web, foram utilizados o Laravel Framework, sendo assim, mais um exemplo prático da sua aplicação.

SOUSA, Claudiany; Silva, ENNIO. SIAS: Sistema web desenvolvido para gerenciamento de atendimentos no setor de saúde do IFTO – Campus Araguatins. **Braz. J. of Develop.**, Curitiba, v. 5, n. 12, p. 30140-30152, dez./2019.





## 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. É importante que os profissionais e estudantes da área saibam que quando é usado o *Framework Back End* Laravel. É possível o desenvolvimento de aplicativos utilizando os benefícios do(a):
  - a. Web semântica.
  - b. HTML5
  - c. Hiperlinks.
  - d. Injeção de dependência.
  - e. Navegadores web.
  
2. Umas das grandes vantagens de utilizar o *Framework Back End* Laravel, dispõe de algumas ferramentas para testes. Quais são os comandos para a criação de um novo teste?
  - a. fake:test.
  - b. make:test
  - c. test:now.
  - d. know:test.

e. right:test.



## GABARITO

### Questão 1 - Resposta D

**Resolução:** Utilizando o *Framework Back End* Laravel, é possível o desenvolvimento de aplicativos utilizando as vantagens da injeção de dependência, sem ter a preocupação com arquivos de configuração inchados. Já a web semântica é modo que proporciona computadores e usuários trabalharem em cooperação. HTML5, hiperlinks e navegadores web estão relacionados com linguagem de marcação nos quais os browsers visualizam os códigos.

### Questão 2 - Resposta B

**Resolução:** O make:test comando Artisan, é uns dos comandos destinados para execução de test no Laravel, sendo que fake:test, test:now, know:test e right:test não são comandos que fazem parte da codificação do Laravel.



TEMA 4

## ***Framework Back End Spring Boot***

---

Autoria: Leandro C. Cardoso

Leitura crítica: Arthur Gonçalves Ferreira



## DIRETO AO PONTO

Existem várias maneiras de desenvolvimento de aplicativo web simples com Spring Boot, sendo que uma delas basta apenas adicionar alguns serviços úteis no qual o Spring Boot, por exemplo, examina o caminho de classe e os *beans* que foram configurados. Também executa suposições razoáveis sobre o que está faltando e adiciona esses itens. Desta maneira, permite ao desenvolvedor concentrar-se mais nos recursos de negócios e menos na infraestrutura. É importante certificar se o Spring MVC está no caminho de classe. Para entender melhor, o Spring MVC trata-se de uma estrutura MVC completa orientada a HTTP, gerenciada pela Spring Framework e baseada em *servlets*. Os *servlets*, de certa forma, são servidores, como uma classe Java utilizada para estender as funcionalidades de um servidor, que também podem ser caracterizados como um componente semelhante a um servidor. No qual gera dados HTML e XML para a camada de apresentação de uma aplicação web, processando dinamicamente requisições e respostas.

Outro fator importante de observar é se existem vários *beans* específicos nos quais o desenvolvedor quase sempre utiliza, e o Spring Boot os adiciona automaticamente. Sendo que um aplicativo Spring MVC também precisa de um contêiner de *servlet*, portanto, o Spring Boot configura automaticamente o Tomcat (servidor web Java) integrado. Também é importante considerar se o Jetty (servidor HTTP e Servlet Container 100% escrito em Java) está no caminho de classe. E caso afirmativo, o desenvolvedor provavelmente não utilizará o Tomcat, mas sim o Jetty incorporado, sendo que estas configurações o Spring Boot configura de maneira automática pelas características de cada projeto.

Também é importante considerar se Thymeleaf (mecanismo de modelo Java XML / XHTML / HTML5 que pode funcionar em ambientes web e não web) está no caminho de classe. Nesse caso, há alguns *beans* que sempre devem ser adicionados ao contexto do aplicativo; e com essa identificação, o Spring Boot os adiciona automaticamente. Estas são apenas alguns exemplos da configuração automática que o Spring Boot fornece. Ao mesmo tempo, o Spring Boot não atrapalha nenhum funcionamento. Por exemplo, se Thymeleaf estiver no caminho, Spring Boot adiciona automaticamente um `SpringTemplateEngine` ao contexto do aplicativo. Mas, caso definir o próprio `SpringTemplateEngine` com as próprias configurações, Spring Boot não adiciona um.

O Spring Boot não gera código ou faz edições nos arquivos, ao invés disso, quando inicia o aplicativo, ele liga dinamicamente os *beans* e as configurações e os aplica ao contexto do aplicativo. Caso o desenvolvedor utilize o Maven, pode-se acessar o site Spring Initializr no endereço <https://start.spring.io/>, para gerar um novo projeto com as dependências necessárias Spring Web. A lista a seguir apresenta um exemplo do arquivo `pom.xml` no qual é desenvolvido quando é escolhido a opção Maven:

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
      https://maven.apache.org/xsd/maven-4.0.0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <parent>

    <groupId>org.springframework.boot</groupId>
```

```
        <artifactId>spring-boot-starter-parent</
artifactId>

        <version>2.5.0</version>

        <relativePath/> <!-- lookup parent from
repository -->

    </parent>

    <groupId>com.example</groupId>

    <artifactId>spring-boot-initial</artifactId>

    <version>0.0.1-SNAPSHOT</version>

    <name>spring-boot-initial</name>

    <description>Projeto de demonstracao Spring Boot</
description>

    <properties>

        <java.version>1.8</java.version>

    </properties>

    <dependencies>

        <dependency>

            <groupId>org.springframework.boot</groupId>

            <artifactId>spring-boot-starter-web</
artifactId>

        </dependency>

        <dependency>
```

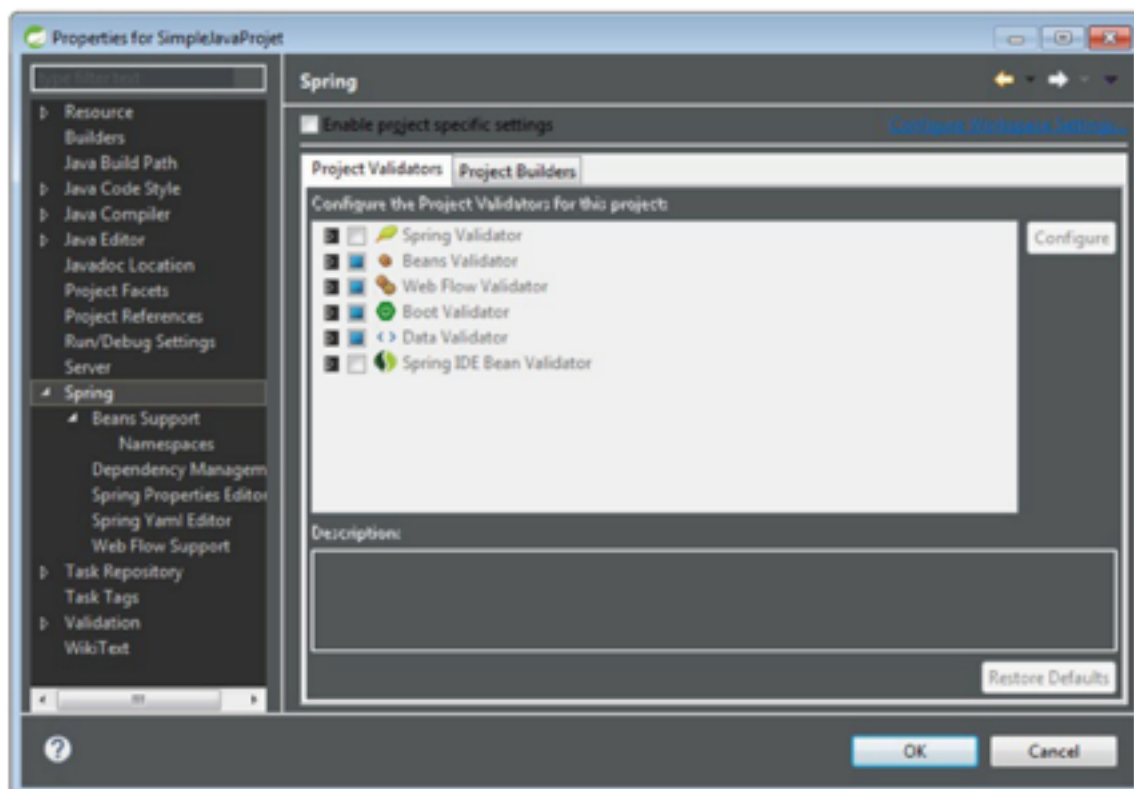


```
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</
artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</
groupId>
            <artifactId>spring-boot-maven-plugin</
artifactId>
        </plugin>
    </plugins>
</build>
</project>
```

Para ilustrar, vejamos na Figura 1, a seguir, a janela de propriedades do Spring.



**Figura 1 – Janela de propriedades do Spring**



Fonte: [https://upload.wikimedia.org/wikipedia/commons/d/d0/Spring\\_Tool\\_Suite\\_3.7.3\\_-\\_Spring\\_properties\\_windows\\_-\\_2017-04-20.png](https://upload.wikimedia.org/wikipedia/commons/d/d0/Spring_Tool_Suite_3.7.3_-_Spring_properties_windows_-_2017-04-20.png). Acesso em: 4 out. 2021.

Para o desenvolvimento *web full stack*, é importante o conhecimento de Java com o *framework back end* Spring Boot, pois otimiza muito o trabalho.

## PARA SABER MAIS

Caso o desenvolvedor utilize o Gradle para o desenvolvimento do projeto, basta acessar a página <https://start.spring.io/> para gerar um novo projeto com as dependências necessárias Spring Web. Veja o exemplo de lista que apresenta o arquivo build.gradle desenvolvido quando a opção Gradle é selecionada:

```
plugins {
```

```
    id 'org.springframework.boot' version '2.5.0'

    id 'io.spring.dependency-management' version '1.0.11.
RELEASE'

    id 'java'
}

group = 'com.example'

version = '0.0.1-SNAPSHOT'

sourceCompatibility = '1.8'

repositories {

    mavenCentral()

}

dependencies {

    implementation 'org.springframework.boot:spring-boot-
starter-web'

    testImplementation 'org.springframework.boot:spring-
boot-starter-test'

}

test {

    useJUnitPlatform()

}
```

Caso o desenvolvedor opte por inicializar o projeto manualmente, uma das maneiras é primeiro acessar o endereço <https://start.spring.io>. Este serviço irá extrair todas as dependências necessárias

para um aplicativo e escolher a opção Gradle ou Maven, e a linguagem de programação no qual é indicada que seja o Java. Depois, basta clicar em ADD DEPENDENCIES (Dependências) ou CTRL +B e selecionar Spring Web, depois clicar em GENERATE (Gerar). O próximo procedimento é baixar o arquivo ZIP resultante, no qual é um arquivo de um aplicativo da web configurado conforme personalização. Caso o IDE do desenvolvedor possuir a integração Spring Initializr, ele pode ser utilizado.

## TEORIA EM PRÁTICA

Para os profissionais de desenvolvimento *web full stack*, a utilização do Java com o *Framework Back End* Spring Boot, proporciona diversas vantagens para os desenvolvedores. Uma delas é a utilização da plataforma que está disponível no endereço: [start.spring.io](https://start.spring.io). Exemplifique quais são os principais procedimentos para iniciar um novo projeto Spring Boot, utilizando esta plataforma. E para otimizar o tempo dos desenvolvedores, algumas ações podem ser utilizadas, como teclas de atalhos. Apresente, também, quais são as principais.

**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

Este trabalho apresenta os recursos necessários para o desenvolvimento de uma *single page application* para o gerenciamento de estoques. No desenvolvimento do *back-end* foi utilizado o *framework* de Java, Spring Boot. Os tópicos 2.1, 2.3 e 2.4 tratam dos temas JavaScript, Java e Spring Framework, respectivamente.

SOUZA, William; PINTO, Giuliano. A UTILIZAÇÃO DO VUE.JS COM UMA API REST EM SPRING BOOT. **Interface Tecnológica**, Taquaritinga, v. 17, n. 2, 2020.

### Indicação 2

Este trabalho apresenta uma aplicação desenvolvida, com objetivo de ajudar desenvolvedores e atuantes da área de tecnologia a encontrar colaboradores, para participar na criação, implementação e execução de suas ideias, no tópico 2.3 sobre *back-end* apresenta aprofundamento do Spring Initializr.

ULGUIM, Yulle. **Rede social para pesquisa e criação de projetos colaborativos**. 2017. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Sistema de Informação) – UFSC, Florianópolis. 2017.

## 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. É fundamental que os profissionais da área de desenvolvimento *web full stack*, no momento que está trabalhando com *frameworks back end*, conhecer as particularidades de cada *frameworks*, por exemplo, são diversos modos para criar um aplicativo utilizando o Spring Boot, sendo que uma delas para iniciar basta somente:
  - a. Adicionar alguns serviços.
  - b. Adicionar *framework*.
  - c. Adicionar hiperlinks.
  - d. Adicionar browser.
  - e. Adicionar *tags*.
2. No desenvolvimento de um serviço da Web RESTful, para dar início a um projeto utilizando a plataforma, no qual extrai

todas as dependências de que são necessárias para um aplicativo os profissionais da área podem usar vários métodos, uma delas é por meio do Gradle. Qual seria o primeiro passo para utilizar esta ferramenta de modo mais otimizado?

- a. Desenvolver a codificação padrão.
- b. Escolher a linguagem de programação.
- c. Definir o *framework* com Node.js
- d. Utilizando os recursos da plataforma start.spring.io.
- e. Definir o *framework* com JavaScript.js.



## GABARITO

### Questão 1 - Resposta A

**Resolução:** Uns dos modos mais fáceis de iniciar um projeto com Spring Boot, é adicionar alguns serviços úteis, e o Spring Boot verifica o caminho de classe e os *beans* que foram configurados. O Spring já é um *framework*, hiperlinks e *tags* são adicionados nas páginas, o browser é um navegador, desta forma, estas opções não se relacionam a criação de aplicativo usando o Spring Boot.

### Questão 2 - Resposta D

**Resolução:** Quando o optar por usar o Gradle é recomendo como primeiro passo acessar a página <https://start.spring.io/> para gerar um novo projeto com as dependências necessárias Spring Web. Desenvolver a codificação, a linguagem de programação como JavaScript.js ou Node.js, não é necessário pois o Spring Boot já define de maneira automática.



**BONS ESTUDOS!**