

embarcadero®

Delphi Academy

Dicas rápidas, truques e técnicas



Primeiros passos com **Ext JS**

Fernando Rizzato
Lead Software Consultant, Latin America

AGENDA

- Ext JS Toolkits
- MVC / MVVM
- Layouts e Containers
- Data Packages
- Demos

EXT JS TOOLKITS

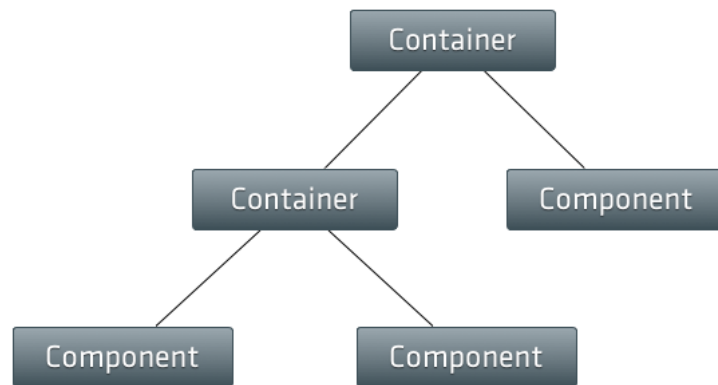
- Classic
 - para browsers desktop/tablet
- Modern
 - para browsers modernos/mobile
- Universal
 - suporta ambos Classic e Modern

MVC / MVVM

- A principal diferença entre **MVC** e **MVVM** é que o MVVM apresenta uma abstração de uma View chamada ViewModel. O ViewModel coordena as alterações entre os dados de um modelo e a apresentação desse dado usando uma técnica chamada "data binding"
- **(M)** Model - Estes são os dados para sua aplicação. Os modelos sabem como persistir no pacote de dados e podem ser vinculados a outros modelos por meio de associações.
- **(V)** View - A View é qualquer tipo de componente que é representado visualmente. Por exemplo, grids, trees e painéis são todos considerados Views
- **(C)** Controller - Controladores são usados como um local para manter a lógica da visão que faz seu aplicativo funcionar. Isso pode implicar renderização de visualizações, roteamento, instanciação de modelos e qualquer outro tipo de lógica de aplicativo
- **(VM)** ViewModel – ViewModel é uma classe que gerencia dados específicos para o View. Ele permite que componentes interessados se vinculem a ele e sejam atualizados sempre que esses dados forem alterados.

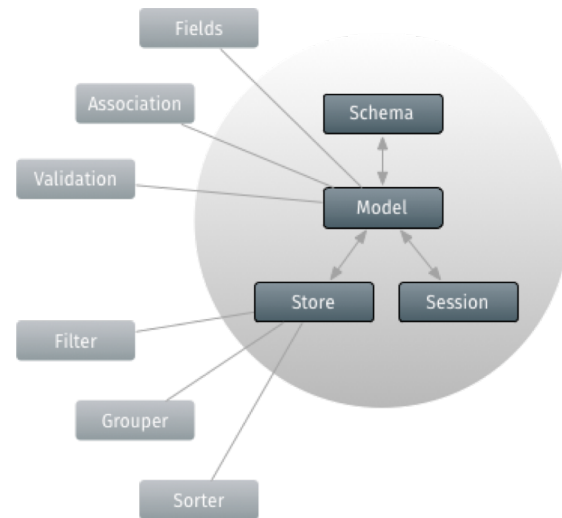
LAYOUTS E CONTAINERS

- O sistema de layout é uma das partes mais poderosas do Ext JS. Ele lida com o dimensionamento e posicionamento de todos os componentes em sua aplicação
- Uma UI do aplicativo Ext JS é composta de Componentes Um Contêiner é um tipo especial de Componente que pode conter outros Componentes. Um aplicativo Ext JS típico é composto de várias camadas de Componentes aninhados



DATA PACKAGES

- O pacote de dados é o que carrega e salva todos os dados em seu aplicativo
- A peça central do pacote de dados é **Ext.data.Model**. Um modelo representa uma entidade em um aplicativo. Por exemplo, um aplicativo de comércio eletrônico pode ter modelos para usuários, produtos e pedidos
- **Proxies** são usados por modelos e lojas para lidar com o carregamento e salvamento de dados do modelo. Existem dois tipos de proxies: Cliente e Servidor
- Modelos são normalmente usados com um **Store**, que é basicamente uma coleção de registros (instâncias de uma classe derivada de **Model**)



DEMOS

RECURSOS ADICIONAIS

Documentação

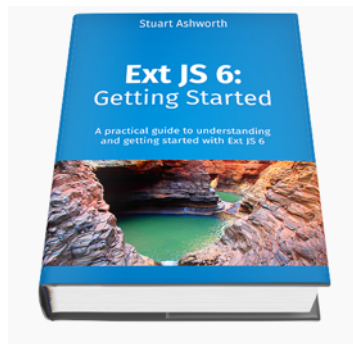
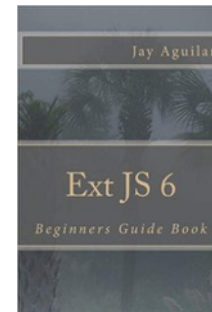
- https://docs.sencha.com/extjs/6.7.0-CE/guides/other_resources/oop_concepts.html
- https://docs.sencha.com/extjs/6.7.0-CE/guides/getting_started/open_tooling.html
- https://docs.sencha.com/extjs/6.7.0-CE/guides/core_concepts/classes.html
- https://docs.sencha.com/extjs/6.7.0-CE/guides/core_concepts/layouts.html
- https://docs.sencha.com/extjs/6.7.0-CE/guides/core_concepts/data_package.html
- <https://examples.sencha.com/extjs/6.7.0/>

Blogs

- <https://linuxize.com/post/how-to-install-node-js-on-ubuntu-18.04>
- <https://www.swarmonline.com/20-tips-for-getting-started-with-ext-js/>
- <https://www.extjs-tutorial.com/>
- <https://www.tutorialspoint.com/extjs/index.htm>
- https://www.youtube.com/playlist?list=PLIUDMEtO0HdS4_YbmTAU7GyygSBCPpUu

EXT JS BOOKS

- <https://www.extjs6gettingstarted.com/>
- <https://www.oreilly.com/library/view/ext-js-6/9781783550494/>
- <https://www.amazon.com/Sencha-Ext-JS-Bootcamp-Book/dp/1539406660>
- <https://www.amazon.com/Mastering-Ext-JS-Loiane-Groner/dp/1782164006>
- <https://www.amazon.com/Ext-JS-Beginners-Guide-Book/dp/1539380858/httpwwwtuto0a-20>
- Packt Publishing: <http://bit.do/eT8T9>



OBRIGADO!

Perguntas?

Você pode me encontrar em:

@FernandoRizzato

fernando.rizzato@embarcadero.com

Siga-nos em

fb.com/DelphiBrasil

fb.com/EmbarcaderoBR