

# Integração de frameworks front-end para desenvolvimento de interfaces ricas com JavaServer Faces

Marcelo Soares da Silva, Fabiano Cutigi Ferrari

**Resumo:** Com a evolução das tecnologias utilizadas para o desenvolvimento de aplicações web atualmente é possível disponibilizar aplicações cada vez mais robustas, funcionais e interativas, melhorando a experiência do usuário. Neste artigo será apresentada a integração de dois frameworks front-end: PrimeFaces e Bootstrap, ambos para o desenvolvimento de interfaces ricas, sobre a especificação Java ServerFaces. Serão analisados as vantagens e desvantagens obtidos pela integração dos frameworks através de um estudo piloto simples de cadastro de carros e um caso de uso de uma aplicação real, com o intuito de ajudar os desenvolvedores a conhecerem as tecnologias e ferramentas que ajudam a implementar interfaces ricas com produtividade e qualidade, melhorando o desempenho da aplicação

**Palavras-Chave:** Primefaces, Bootstrap, JavaServer Faces, RIAs, Interfaces ricas para web

## *Integrating front-end frameworks to develop rich interfaces with JavaServer Faces*

**Abstract:** *With the evolution of technologies used for developing web applications, it is now possible to provide increasingly robust, functional and interactive applications, improving the user experience. This paper will present the integration of two front-end frameworks: PrimeFaces and Bootstrap, both for developing rich interfaces on the Java ServerFaces specification. The advantages and disadvantages of integrating frameworks through a simple pilot study of registration of cars and a use case in a real application will be analyzed in order to help developers to know the technologies and tools that help implement rich interfaces with productivity and quality, improving application performance..*

**Keywords:** *Primefaces, Bootstrap, JavaServer Faces, RIAs, Rich Interfaces for Web.*

## I. INTRODUÇÃO

A cada dia a tecnologia inova gerando cada vez mais dados que precisam ser transformados em informação, gerando consigo várias vertentes no ramo do desenvolvimento de software. Atualmente pela internet vários serviços são disponibilizados como, por exemplo, fazer um reserva em um hotel, comprar uma passagem e acessar a conta no banco, ou seja, o mundo está conectado (INFOSYS, 2011).

A disponibilidade de internet está cada vez maior, com navegadores cada vez mais robustos possibilitando uma melhor apresentação da informação ao usuário através de interfaces sofisticadas utilizando RIAs (Rich Internet Applications). RIAs representam uma evolução das páginas web estáticas tradicionais, que apresentavam poucos recursos do lado cliente e com muitas limitações na interatividade e usabilidade. Segundo Bush et al. (2009), RIAs são uma variedade de tecnologias baseadas na web para promover front-ends (interfaces de usuário) sofisticados, funcionais, intuitivas, rápidas e bonitas melhorando o desempenho da aplicação e a experiência do usuário.

Com a utilização de RIAs é possível gerar conteúdo dinamicamente, armazenar e processar dados do processo no

lado cliente, fazer comunicação assíncrona, dar suporte a multimídia. RIAs criam interatividade e usabilidade sofisticadas, com um front-end rico em recursos parecidos a uma aplicação desktop tradicional, obtendo uma maior atratividade por parte do usuário (TOFFETTI, 2011).

A implementação de uma aplicação utilizando RIAs pode ser uma tarefa complexa, desde que o desenvolvedor tenha que planejar e desenvolver seus próprios componentes sem a ajuda de alguma ferramenta. Observa-se que, com a velocidade que a tecnologia evolui a produtividade é essencial. Através de frameworks front-end com o apoio da arquitetura do framework JSF<sup>1</sup> (JavaServer Faces), implementar interfaces ricas se torna uma tarefa mais fácil, poupando horas de desenvolvimento e estudo sobre inúmeras tecnologias tais como HTML, CSS, JAVA SCRIPT, JSON, XML, AJAX.

O estudo desenvolvido neste artigo, tem como objetivo apresentar a integração de frameworks front-end sobre a arquitetura JSF, através de um estudo piloto simples de

<sup>1</sup>JavaServer Faces Technology. Disponível em: <<http://www.oracle.com/technetwork/java/javae/javaxserverfaces-139869.html>>. Acesso em: 18 out. 2014.

cadastro de carros, seguida por uma apresentação de um estudo de caso do sistema Gestão de Qualificação desenvolvido no decorrer de um curso de pós-graduação lato-sensu em Computação com foco em desenvolvimento de Software para Web oferecido pela UFSCar. Com a intenção de mostrar que é possível implementar interfaces ricas com produtividade e qualidade levando o usuário a uma melhor experiência da aplicação. A metodologia empregada no desenvolvimento do trabalho será a apresentação da integração entre os frameworks de maneira prática, identificando as vantagens e as desvantagens obtidos com esta prática.

O artigo está organizado da seguinte forma: na sessão II apresenta-se uma explanação sobre RIAs; na sessão III apresenta-se uma abordagem sobre os frameworks utilizados: PrimeFaces, Bootstrap e JSF; na sessão IV apresenta-se o desenvolvimento do estudo piloto através dos frameworks propostos; na sessão V apresenta-se o estudo de caso de uma aplicação real, Gestão de Qualificação utilizando os conceitos abordados; na sessão VI apresenta-se os trabalhos relacionados; na sessão VII apresenta-se a conclusão e trabalhos futuros.

## II. RICH INTERNET APPLICATIONS (RIAs)

O conceito de RIA (Aplicações Ricas para Internet) não é novo. Desde 1998, a Microsoft já falava sobre interfaces ricas com o Remote Scripting<sup>2</sup>, que visava atualizar apenas uma parte da página dando ao usuário uma experiência mais agradável a utilizar páginas HTML estáticas. Uma aplicação utilizando o conceito de RIA, obtém riqueza de conteúdo, facilidade de uso, melhor desempenho e interatividade (ROSEINDIA, 2011).

Na estrutura de uma RIA, como ilustrada na Figura 1 pode exigir a instalação de um Runtime específico ao cliente (navegador web, não exibido na figura) através de plug-ins, por exemplo, ou pode-se utilizar das técnicas sofisticadas de recursos como o AJAX para retornar um conteúdo específico sem necessidade de qualquer instalação adicional por parte do usuário, o que se torna uma forma mais elegante e simples.

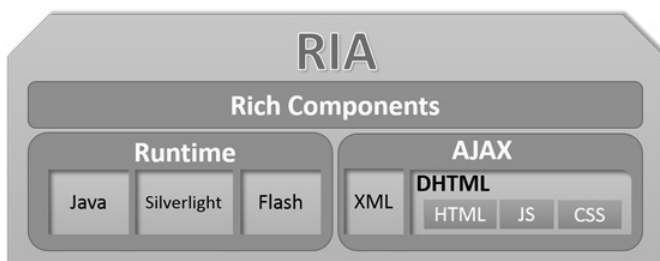


Figura 1. Diagrama de blocos, mostra a estrutura de uma RIA, adaptada do trabalho de ÁVILA (2011).

Neste conceito o processamento também ocorre do lado cliente. Mais especificamente, o processamento ocorre no navegador, implementando assim a arquitetura do 'cliente

gordo' que visa deixar uma parte do trabalho do lado cliente. Desta forma tem-se uma menor sincronização das páginas deixando o acesso mais suave e prazeroso.

Na atualidade muitas aplicações estão utilizando RIAs, desde um sistema de grande porte a um editor de texto, devido à demanda de usuários mais exigentes por interfaces mais funcionais e uma experiência mais próxima ao mundo real (WARD, 2007).

Através da implementação de interfaces ricas é possível criar front-ends com uma variedade de componentes, que podem ser: mapas, animações, grades, formulários complexos, gráficos, tabelas interativas, caixas de diálogo entre outros que tornam a aplicação mais robusta e com um visual bem próximo as aplicações desktop.

Devido a esta tendência no mercado apareceram vários frameworks front-end, que são ferramentas para ajudar na implementação de RIAs, sendo alguns proprietários como: .NET Framework; Unity; Canoo RIA Suite; Curl; DevForce e outros gratuitos tais como: PrimeFaces; Bootstrap; RichFaces; ICEFaces; Vaadin; JQuery UI; Ext JS; Fuel UX; HTML5; My Faces Trinidad entre vários outros.

A Figura 2 mostra a evolução das aplicações web relativamente a evolução dos servidores e clientes (navegadores). Identifica-se que, com a evolução da velocidade da internet, o processamento dos servidores e processamento dos clientes, um número maior de funcionalidades nas páginas web surgirão, melhorando a usabilidade das aplicações neste processo evolutivo constante.

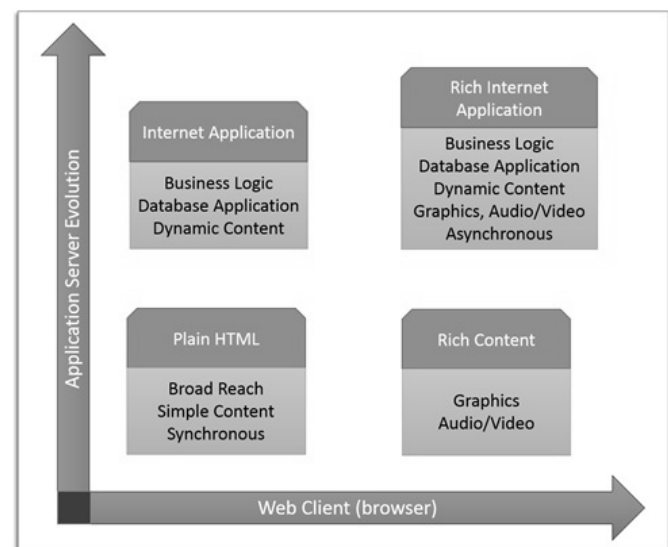


Figura 2. Evolução das aplicações web, adaptada do trabalho de INFOSYS (2011)

A interface de uma aplicação tem um papel essencial, pois é onde o usuário vai interagir com o sistema. Para garantir o sucesso de uma aplicação é necessário pensar no usuário, analisando as iterações a serem feitas, a usabilidade e o desempenho que o front-end deva disponibilizar tornando o acesso mais agradável, fácil e ágil de forma equilibrada e harmônica (FERNANDES, 2009).

Para se desenvolver uma front-end bem estruturado, otimizado e performático é necessário fazer um bom

<sup>2</sup>Get Ready for Ajax. Disponível em:

<<http://msdn.microsoft.com/en-us/magazine/cc163580.aspx>>. Acesso em: 09 nov. 2014.

planejamento do projeto, pois geralmente o front-end gasta um tempo de desenvolvimento maior que o back-end (códigos de programação).

Segundo Souders (2012) a regra de ouro de desempenho de uma aplicação se deve ao tempo que o usuário deva gastar no front-end, cerca de 80% à 90%. Ou seja, utiliza-se o back-end somente quando necessário, melhorando o desempenho da

aplicação e poupando recursos. Na Figura 3, apresenta-se um gráfico dos 10 sites mais acessados mundialmente. O gráfico aponta que o tempo gasto pelos usuários no front-end é de 76% em relação ao back-end 24%. Observa-se que os sites por terem poucos recursos visuais como, por exemplo, o Google, ficam um pouco abaixo da ‘regra de ouro’.

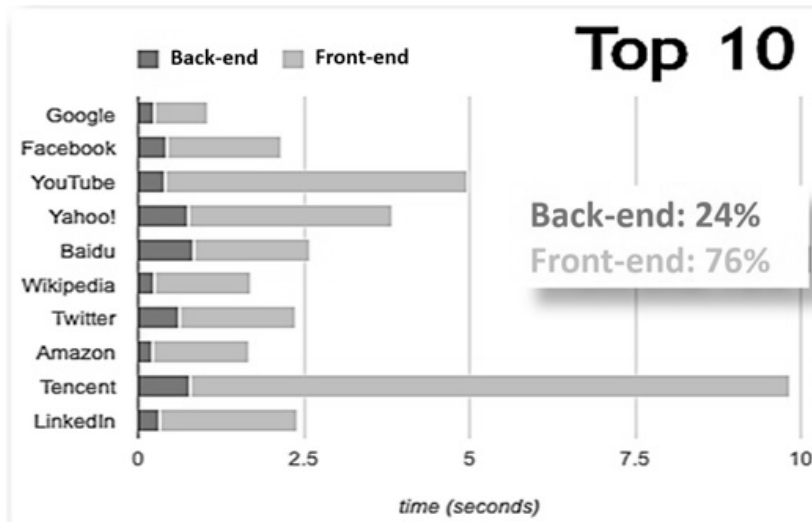


Figura 3. Tempo que o usuário gasta entre back-end e front-end no Top 10 sites mais acessados mundialmente (SUDORES, 2012).

Na Figura 4 o gráfico mostra que as startups que são empresas recém-criadas com o foco em inovação, mantêm os seus usuários em 84% no front-end, e como as startups

seguem tendências podemos ver que estão na faixa da ‘regra de ouro’.

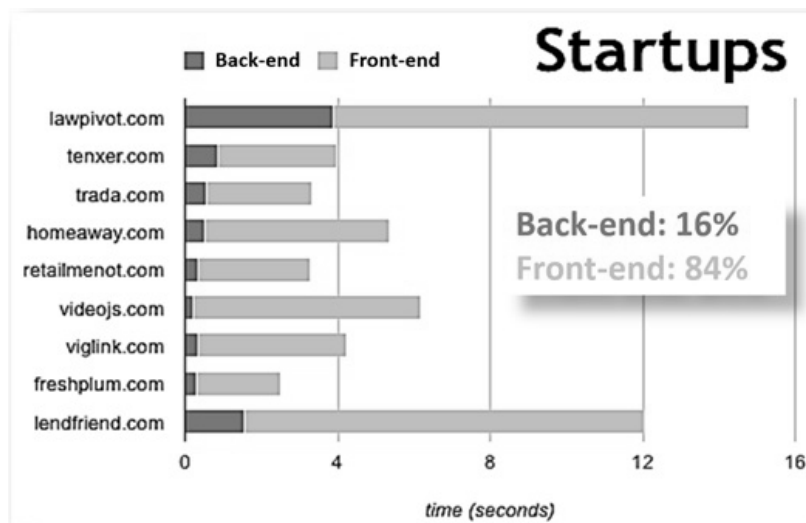


Figura 4. Tempo que o usuário gasta entre back-end e front-end nas startups (SUDORES, 2012).

Com base destes resultados mostra-se a importância de uma aplicação com interfaces bem planejadas e desenvolvidas, pois é uma tendência natural da evolução da web sempre com o intuito de disponibilizar uma aplicação mais funcional ao usuário.

### III. FRAMEWORKS

Para facilitar o desenvolvimento de aplicações surgiram os frameworks, que são estruturas com camadas e padrões a

serem seguidos, com códigos de qualidade e testados, exigindo um menor esforço por parte do desenvolvedor e gerando produtividade.

Um framework é uma plataforma que fornece uma estrutura base permitindo o desenvolvimento de aplicações a partir de classes ou funções pré-definidas. Os frameworks podem ser multiplataforma ou específicos de uma, em alguns casos necessitam de instalações adicionais para o funcionamento, como certas aplicações da Microsoft que exigem o Microsoft NET Framework (TECHTERMS, 2013).

### A) PrimeFaces

O framework front-end PrimeFaces é uma biblioteca de componentes JSF baseada em JQuery e HTML5 de código aberto para criação de interfaces ricas e funcionais para usuário (PRIMEFACES). Atualmente a produtividade com qualidade é imprescindível para o desenvolvimento de softwares. Nesse sentido, o PrimeFaces vem com a proposta de levar ao usuário a experiência de uma interface via navegador similar às nativas instaladas nos computadores levando a uma navegação mais confortável e agradável.

O PrimeFaces contém uma coletânea de componentes<sup>3</sup> vasta, desde um simples input a um dataTable complexo. A cada versão surgem novos componentes e a correção de

outros. A documentação é muito completa e contém exemplos de utilização de cada componente, deixando o entendimento ainda mais fácil. A comunidade que utiliza esta ferramenta é muito ativa, o que traz uma maior facilidade na resolução de problemas ou dúvidas. Além de todos os componentes, o PrimeFaces disponibiliza uma série de temas que podem ser adicionados facilmente, possibilitando a alteração da aparência dos componentes, deixando mais compatível com o designer da aplicação.

Em pouco tempo o PrimeFaces superou seus concorrentes como IceFaces e RichFaces como mostra a Figura 5, devido a qualidade de seus componentes.

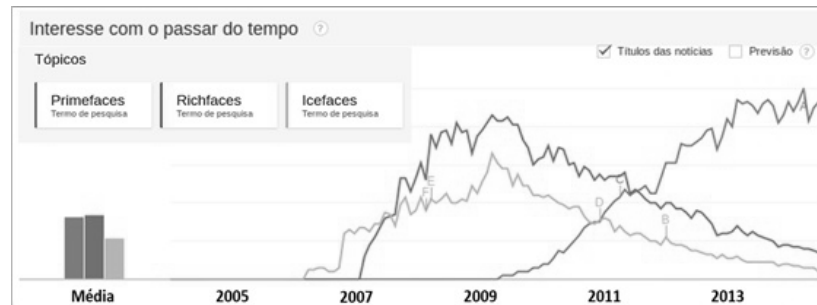


Figura 5. Gráfico gerado pela Google Treands em 19/07/2014, compara a popularidade/interesse sobre os frameworks front-end de componentes (GOOGLE, 2014).

Grandes empresas como a Ford, BMW, Siemens, Dell entre outras utilizam o framework como padrão front-end em suas aplicações, isso mostra a maturidade da plataforma (PRIMEFACES).

Uma das grandes vantagens do PrimeFaces é o uso de AJAX por padrão em seus componentes. Para adicionar o framework ao projeto basta fazer o download da biblioteca que é um arquivo de extensão .jar do site oficial<sup>4</sup>, mapear no projeto e pronto, tudo isso sem dependências e complicações. Os requisitos mínimos para utilização são Java 5 e JSF 1.2.

### B) Bootstrap

O framework front-end Bootstrap é uma coleção de

convenções HTML5, CSS3 e uma biblioteca JavaScript. Atualmente um dos mais populares frameworks front-end em seu seguimento como apresentado na Figura 6. A popularidade se deve a sua alta compatibilidade entre os navegadores e a responsividade de seus componentes<sup>5</sup> que se ajustam a várias resoluções com o intuito de criar uma aplicação única (BOOTSTRAP).

O Bootstrap conta com uma documentação muito completa e exemplos para facilitar o entendimento de seus componentes. Uma das vantagens mais atuais do Bootstrap é o suporte a Less e Sass que são co-processadores e funcionam como otimizadores de código, trazendo maior produtividade sem causar muitos impactos ao código.



Figura 6. Gráfico gerado pela Google Treands em 19/07/2014, comparando a popularidade/interesse sobre os frameworks front-end HTML5/CSS3 (GOOGLE, 2014).

<sup>3</sup>Lista de componentes do PrimeFaces. Disponível em: <<http://www.primefaces.org/showcase/>>. Acesso em 18 out. 2014.

<sup>4</sup>Site oficial do PrimeFaces. Disponível em: <<http://www.primefaces.org/>>. Acesso em 18 out. 2014.

Cada desenvolvedor pode fazer seu Bootstrap personalizado estudando o código e fazendo as devidas alterações como os

<sup>5</sup>Coleção de componentes do Bootstrap. Disponível em: <<http://getbootstrap.com/>>. Acesso em: 18 out. 2014.

desenvolvidos pela [globo.com](http://globo.com), [yahoo.com](http://yahoo.com) e [dropbox.com](http://dropbox.com) ou utilizar um tema fornecido gratuitamente pela [bootswatch.com](http://bootswatch.com).

Como citado o framework utiliza layout responsivo, o que é uma significativa vantagem, pois com a gama de dispositivos disponíveis no mercado com múltiplas resoluções, fazer um layout que se adeque a qualquer dispositivo é produtivo e crucial para a aplicações atuais.

### C) *JavaServer Faces (JSF)*

O framework UI (User Interface) JavaServer Faces é destinado ao desenvolvimento de interfaces de usuário para

aplicações Java web, com o intuito de facilitar a escrita e manutenção do código utilizando a arquitetura MVC. A comunicação entre os Managed Beans e as páginas JFS ocorrem por componentes gráficos baseados em eventos, validadores, conversores entre outros como identificado na Figura 7. Com esta arquitetura o JSF traz uma maior facilidade para a integração de outros frameworks, pois cada camada fica devidamente separada. Com JavaServer Faces é possível construir interfaces ricas através de componentes utilizando frameworks front-end, facilitando o desenvolvimento (ANDRADE, 2010).

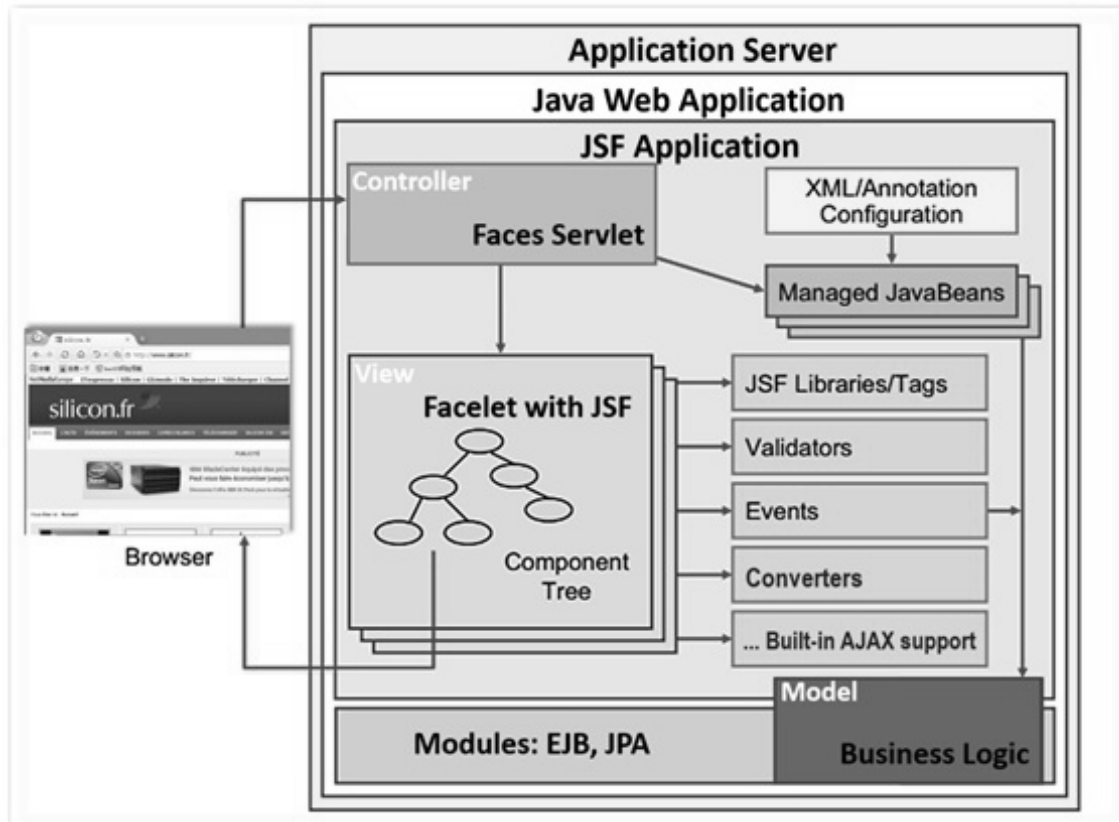


Figura 7. Arquitetura de uma aplicação JSF, adaptada do trabalho de KEEN (2012)

O uso de bibliotecas de componentes para JSF é natural, pois as implementações geralmente vêm com poucas opções, justamente para que o desenvolvedor possa fazer as alterações necessárias no visual da sua aplicação web. Dessa forma é possível utilizar os frameworks front-end de componentes como o PrimeFaces que contém vários widgets (componentes de interface gráfica) avançados e estáveis melhorando a robustez e usabilidade da aplicação.

O framework possui módulos de eventos, componentes, validações, conversões, navegabilidade e back-ends fornecendo uma melhor divisão das funções necessárias para a construção da aplicação. Com o JSF os papéis dos desenvolvedores ficam bem definidos como por exemplo: desenvolvedores de front-end, back-end, componentes, arquitetura entre outros (ANDRADE, 2010).

### IV. ESTUDO PILOTO

Neste estudo piloto será desenvolvido um projeto simples de cadastro de automóveis utilizando a integração dos frameworks já apresentados anteriormente com o intuito de implementar uma interface rica com qualidade e produtividade.

No projeto foram utilizadas as tecnologias: framework UI JSF implementação Mojarra 2.2, os frameworks front-end PrimeFaces e Bootstrap. Para desenvolver o projeto foi utilizada a IDE (ambiente integrado para desenvolvimento de software) NetBeans 7.3.1, plataforma Java EE (Enterprise Edition) versão 7 e o servidor de aplicação GlassFish 4.0.

Para iniciar o estudo foi criada uma aplicação web através no NetBeans, na sequência foi adicionado o PrimeFaces ao projeto fazendo o download do arquivo de extensão .jar disponível na página oficial em <http://www.primefaces.org/>.

É aconselhável baixar a última versão, pois a documentação disponível no site sempre está de acordo com a mesma. Após o download colocar o arquivo em um diretório dentro do

projeto e mapear a biblioteca como mostra a Figura 8, assim o projeto estará utilizando o framework PrimeFaces.

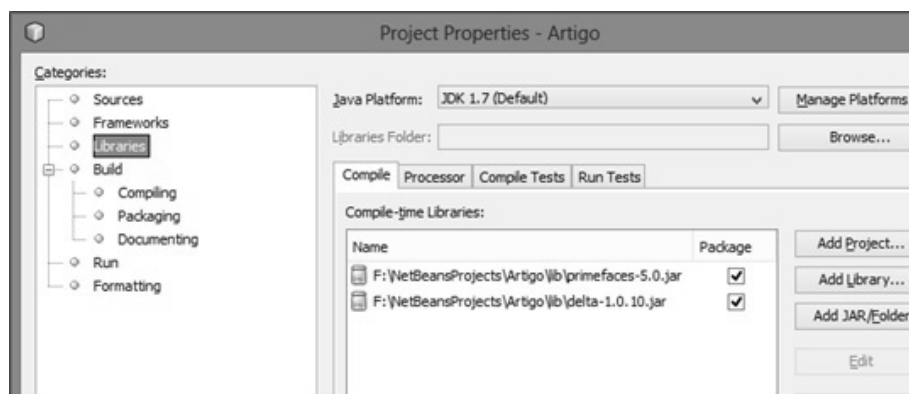


Figura 8. Adicionar biblioteca PrimeFaces ao projeto

Para utilizar as tags do PrimeFaces nas páginas XHTML basta adicionar o namespace `xmlns:p="http://primefaces.org/ui"` dentro da tag HTML da página com identificado na Figura 9.



Figura 9. Namespace do PrimeFaces adicionado na tag HTML da página XHTML.

Para adicionar um componente a página utilizar o namespace p: seguido do nome do componente como segue: `<p:commandButton />`.

Para adicionar o Bootstrap ao projeto baixar o arquivo de extensão .zip do site oficial <http://getbootstrap.com>, descompactar o arquivo dentro do diretório web do projeto, depois adicionar os links de referência dentro da tag HEAD da página XHTML como segue na Figura 10.

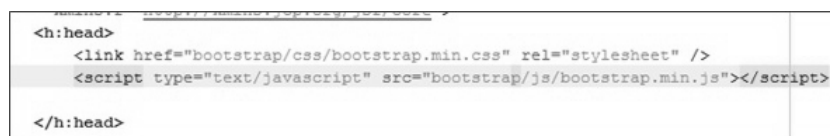


Figura 10. Mostra a adição do Bootstrap ao projeto

Desta forma o projeto está configurado com o Bootstrap. Para utilizar seus recursos adicionar as classes junto a tag desejada como segue: `<h:commandButton styleClass="btn btn-success" />`. O melhor meio para saber mais sobre como utilizar os frameworks é acessando a documentação disponível nos sites oficiais.

No projeto proposto foram utilizados os seguintes

componentes do Bootstrap: o sistema de grids responsivas, o CSS dos botões Insert, Alter, nos inputs correspondentes a Brand e Price, nas divs que formam o painel Registered Cars e o menu. Os componentes do PrimeFaces foram: os inputs referentes a Year e Status, o botão Cancel e o datable que lista as informações como mostra a Figura 11.

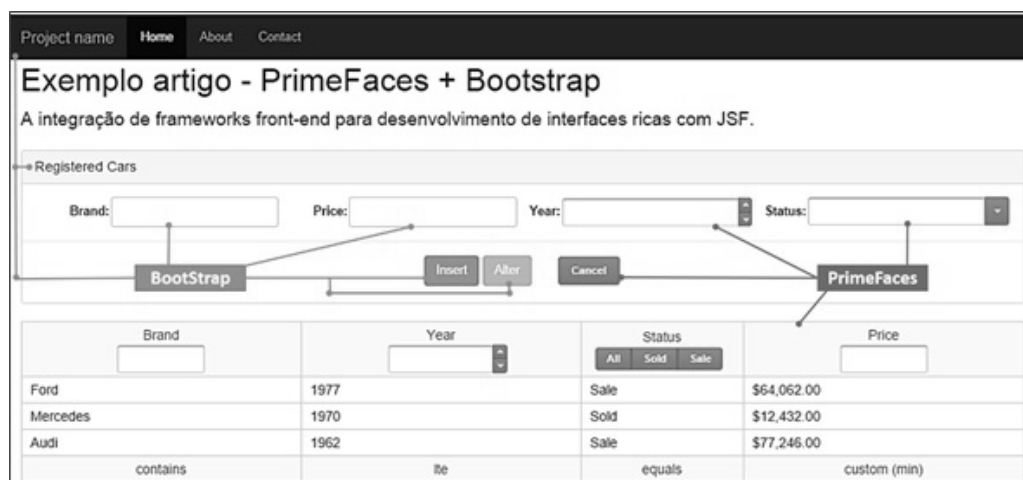


Figura 11. Front-end desenvolvido com a integração dos frameworks PrimeFaces e Bootstrap sobre a implementação Mojarra JSF.

Nesta interface mostra-se a integração dos frameworks proporcionando uma interface mais funcional e rica. Utiliza-se o CSS responsivo do Bootstrap, que traz maior flexibilidade de acesso para resoluções de diferentes navegadores, pois a página e componentes se adaptam ao tamanho da tela, melhorando a experiência visual comparado ao desenvolvimento web tradicional. E com os componentes ricos do PrimeFaces como o datatable que manipula todos os dados do lado cliente executando filtros e ordenação sem a necessidade de sincronização com o servidor, com isso melhora-se o desempenho da aplicação, pois no modo tradicional a página teria de ser atualizada constantemente exigindo maior fluxo de dados na rede.

## V. ESTUDO DE CASO

A implementação do estudo de caso proposto neste artigo foi um projeto desenvolvido por sete alunos do curso de pós-graduação Lato Sensu em Desenvolvimento web da Universidade Federal de São Carlos, com o intuito de atender as necessidades da empresa Schnellecke Logistics com o sistema de Gestão de Qualificação.

O projeto foi desenvolvido com os requisitos estabelecidos pela empresa Schnellecke com o intuito de melhorar o

gerenciamento dos treinamentos realizados por seus funcionários e facilitar a organização com relação aos prazos de realização das avaliações. Busca-se manter assim o funcionário com os treinamentos atualizados, ficando aptos a exercerem suas atividades em sua área de trabalho, evitando problemas com as auditorias e fiscalizações.

O sistema visa criar e gerenciar os treinamentos que os funcionários da empresa deverão realizar periodicamente para ocupar os seus cargos em determinadas áreas. O projeto foi desenvolvido em JSF, utilizando a integração dos frameworks front-end PrimeFaces e Bootstrap para implementar as interfaces de usuário.

A utilização do JSF se deve a arquitetura MVC, assim o projeto poderia ser desenvolvido pelos integrantes do grupo em paralelo e o outro motivo é a utilização de bibliotecas de componentes para JSF.

Através da integração dos frameworks front-end foi obtida uma produtividade e qualidade de interface atrativos. A interface ficou rica com componentes funcionais e inteligentes do PrimeFace e através do layout responsivo do Bootstrap as páginas podem ser abertas em várias resoluções, trazendo uma melhor experiência ao usuário como mostra a Figura 12.

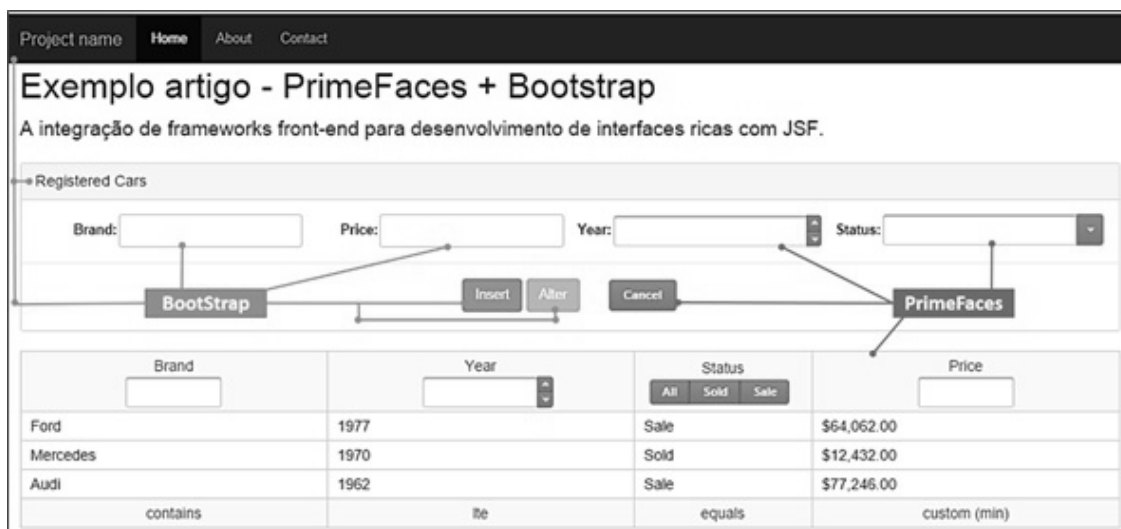


Figura 12. Página com componentes PrimeFaces e Bootstrap apresentados em uma janela redimensionada para resolução de 480 X 500 pixels.

A integração dos frameworks trouxe benefícios, porem para alguns componentes do PrimeFaces o CSS entra em conflito. Foram testados 30 componentes e em 5 apresentaram algum problema de posicionamento ou deformação, foram eles: Fieldset, OneMenu, Stick do DataTable, ColumnToggler do DataTable, Resize do DataTable. Na Figura 13 apresenta-se o FieldSet, onde o mesmo fica deformado. É possível alterar o CSS deste elemento para obter a forma desejada, mas às vezes a tarefa de entender o código do componente é complexa e pode ser morosa, ficando como melhor opção trocar de componente ou criá-lo a partir de outros componentes do próprio framework.

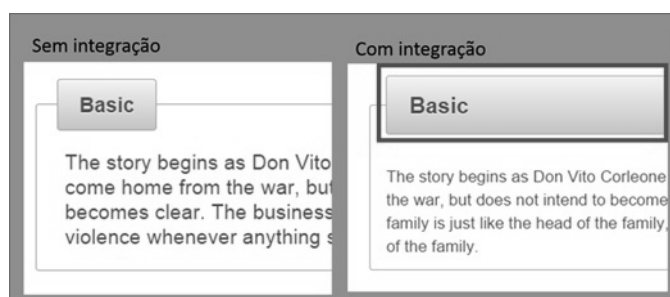


Figura 13. Do lado esquerdo o FieldSet do PrimeFaces sem intervenção do Bootstrap, do lado direito com a integração o FieldSet fica deformado

Com a documentação disponível dos dois frameworks front-end utilizados torna-se uma tarefa fácil criar interfaces ricas, basta ter um mínimo de experiência em HTML, CSS e JSF para começar a desenvolver. A produtividade é aumentada, afinal está quase tudo pronto, bastando dispor da forma desejada e fazer os mapeamentos. Com a utilização dos frameworks uma tarefa com esta proposta do estudo piloto, poderia levar muitas horas para ser desenvolvida sem ajuda destas ferramentas que diminuiriam esse tempo consideravelmente.

Através da integração obteve-se vantagens interessantes como mostra a Tabela 1, mas as desvantagens são inevitáveis com apresentado na Tabela 2. Outro benefício da integração é que na maioria das vezes é necessária alguma funcionalidade que um certo framework tenha que outro não esteja implementado.

Tabela 1. Vantagens obtidas com o uso da integração dos frameworks.

Vantagens	Descrição
Padronização	O código segue um padrão pré-definido.
Produtividade	Após o entendimento dos componentes de cada <i>framework</i> é só utilizá-lo.
Qualidade de código	As empresas que fornecem os <i>frameworks</i> com códigos já testados.
Manutenção	Para dar manutenção no código fica mais fácil devido a padronização e com um número reduzido de linhas.
Ajuda da comunidade	Com o uso de <i>frameworks</i> conhecidos como o PrimeFaces e Bootstrap a comunidade de utilizados é muito ativa.
Redução de custos	Com a produtividade aumentada a redução de custos é uma consequência.
Flexibilidade	É possível através da integração é possível escolher entre vários componentes ou até mesmo outros <i>frameworks</i> .

Tabela 2. Desvantagens obtidas com o uso da integração dos frameworks

Desvantagens	Descrição
Engessamentos da implementação	Com o uso de frameworks a implementação fica limitada ao uso de seus recursos para seguir o padrão.
Curva de aprendizado	Com a integração dos <i>frameworks</i> a curva de aprendizado tende a ser maior e mais complexa.
Configuração	Dependendo da integração a ser realizada pode complicada.
Customização	Para fazer a customização dos componentes tem de um conhecimento aprimorado dos <i>frameworks</i> .
Conflitos	Após a integração pode ser gerado conflitos entre códigos dos <i>frameworks</i> .

Adicionado PrimeFaces ao projeto é possível não utilizar o tema CSS do mesmo apenas adicionado dois parâmetros ao arquivo web.xml da aplicação como mostra a Figura 14. Desta forma fica a cargo do desenvolvedor escolher o melhor CSS para sua aplicação, neste caso pode-se utilizar o CSS do Bootstrap para dar outra aparência aos componentes do PrimeFaces como mostra a Figura 15.

```
<context-param>
  <param-name>primefaces.THEME</param-name>
  <param-value>none</param-value>
</context-param>
```

Figura 14. Configuração do parâmetro no arquivo web.xml para remover o tema CSS dos componentes do PrimeFaces.



Figura 15. Diferença entre botões do PrimeFaces com CSS nativo e CSS do Bootstrap

Este estudo de caso mostrou algumas das possibilidades com a integração dos frameworks. Com a integração é possível criar front-ends complexos, ricos em recursos utilizando o que tem de melhor em cada ferramenta, disponibilizando ao usuário uma experiência agradável mediante uma interface funcional e com um desempenho melhorado. Como apresentado neste artigo utilizou-se dois frameworks front-end. Contudo, a utilização de uma quantidade maior de frameworks, vai depender da necessidade do desenvolvedor e da aplicação com isso pode aumentar a complexidade da integração. No mercado existem vários frameworks disponíveis e gratuitos prontos para serem adicionados ao projeto web, trazendo assim uma maior produtividade e qualidade nas interfaces com diversos componentes. Com esta grande variedade de ferramentas, caso a integração não fique muito agradável, pode-se substituir por outro até achar a integração que melhor atende ao desenvolvimento em questão.

## VI. TRABALHOS RELACIONADOS

Em uma pesquisa realizada por Toffetti (2011), identifica-se que as aplicações que implementam RIAs têm um custo maior de complexidade no seu desenvolvimento, apontando que é necessário seguir uma metodologia com o apoio do uso de ferramentas disponíveis na internet como os frameworks Silverlight, Flex, OpenLaszlo entre outros. O estudo apresenta mais de 150 abordagens propostas para a implementação de interfaces ricas, melhorando o processo de desenvolvimento e manutenção da aplicação.

No trabalho realizado por Neil (2010) mostra-se que é difícil utilizar os paradigmas de interfaces desktop em interfaces web para criar uma aplicação rica, intuitiva e ágil. Segundo Neil para obter sucesso a interface tem que ser pensada levando em conta o tipo de usuário que irá interagir com a aplicação, modelando assim o designer da página de acordo com o mesmo, utilizando frameworks baseados em RIA para facilitar o desenvolvimento e atingir as metas com um menor esforço.

Na integração proposta por Qui (2013) entre os frameworks Bootstrap e jQuery UI sobre a plataforma ASP.NET, o objetivo é obter mais opções para o desenvolvimento de interfaces de aplicações web. Ao fazer a integração ocorreram incompatibilidades entre os temas CSS dos frameworks, principalmente nos botões. Para solucionar o problema, Qui faz alterações no arquivo onde o CSS dos frameworks são importados para o projeto e indica qual CSS terá prioridade, solucionado o problema de incompatibilidade.



Segundo Arun (2009), às vezes ao desenvolver um projeto web surge um bloqueio ao tentar utilizar um componente de um determinado framework front-end. Porém, existe o componente ideal em outra ferramenta que atende a necessidade. Então surge a necessidade de fazer a integração de frameworks para agilizar o desenvolvimento. No projeto proposto em seu artigo, faz-se a integração entre RichFaces e extJS sobre a plataforma JSF para criar interfaces ricas. O seu trabalho apresentou que é possível fazer a troca de dados entre componentes de frameworks diferentes através da integração.

Em estudo realizado por The Server Labs (2008), mostra-se que as aplicações web tradicionais têm um desempenho ruim, pois cada interação tem de passar pelo servidor e não disponibiliza uma interface funcional ao usuário. Mostra-se que através de RIAs é possível executar processos do lado cliente e proporcionar ao usuário um melhor desempenho da aplicação e uma melhor experiência de interface. Para o desenvolvimento de interfaces ricas aponta-se a adoção de frameworks, pois a criação de componentes é muito cara em termos de tempo. O estudo aponta que, com a evolução das tecnologias RIAs, cada vez mais oferece-se um melhor suporte e maior nível de abstração com uma variedade de opções.

## VII. CONCLUSÃO E TRABALHOS FUTUROS

Com a necessidade cada vez maior de disponibilizar interfaces de usuários com riqueza e com um curto prazo a adoção de frameworks front-end para o desenvolvimento de aplicações web se torna imprescindível, trazendo maior produtividade e qualidade a aplicação. Como apresentado o framework JSF, dá todo suporte aos frameworks front-ends, deixando a tarefa de fazer interfaces ricas muito mais fácil e rápido.

Como apresentado no decorrer deste artigo a integração entre frameworks front-end traz inúmeras possibilidades, desta forma o desenvolvedor tem várias opções para montar a interface da aplicação. Entretanto, ao fazer esta prática, podem ocorrer incompatibilidades entre alguns componentes das ferramentas, o que poderá demandar um tempo e um conhecimento mais aprofundado para resolver. Mas mesmo com as incompatibilidades é muito válido fazer a integração, pois assim o desenvolvedor poderá utilizar recursos de dois ou mais frameworks conseguindo maior liberdade no desenvolvimento da aplicação.

Atualmente no mercado existem e estão surgindo vários frameworks front-ends trazendo ainda mais opções. Fazendo o uso da integração entre eles as possibilidades de se fazer interfaces ricas aumentam, com a possibilidade de experimentar várias integrações optando pelas que melhor se ajustam ao projeto. Através das integrações cria-se uma flexibilidade no desenvolvimento, desta forma o desenvolvedor não fica preso a apenas uma solução que pode trazer algumas restrições ao faltar algo para dar continuidade a implementação do projeto. Cada ferramenta pode ser utilizada de acordo com a necessidade da aplicação web, com isso não será necessário 'reinventar a roda', pois existem

muitas soluções prontas e estáveis, bastando entender seu funcionamento e adicionar ao projeto.

O desenvolvimento de interfaces para aplicações web que tenham uma melhor usabilidade já faz parte da atualidade e é inevitável o uso de RIAs para atingir este objetivo e assim tornar o projeto mais atraente ao usuário. Utilizar frameworks é a chave para um desenvolvimento com qualidade, afinal os códigos são testados e milhares de pessoas utilizam as mesmas ferramentas. Caso contrário o desenvolvedor terá de criar muitos recursos que os frameworks já disponibilizam demandando um conhecimento vasto sobre várias tecnologias abordadas neste artigo, além dos testes a serem realizados nos componentes que desenvolver, tudo isso leva muito tempo, concluindo que o uso de frameworks alavanca o projeto, reduz o tempo de desenvolvimento e melhora a qualidade do código.

Para os trabalhos futuros pode-se experimentar a integração entre três ou mais frameworks front-end como RichFaces, IceFaces, JQuery UI, Ext JS sobre o framework UI JSF para desenvolver interfaces ricas, analisando os benefícios e limitações gerados pelas integrações realizadas da mesma forma como foi a proposto no estudo piloto e no caso de uso descrito neste artigo. Cada ferramenta tem os seus benefícios e integrá-las traz maior liberdade ao desenvolvedor para utilizar este ou aquele recurso. É fato que terão de ser feitos alguns ajustes, mas o fato de não desenvolver o componente do zero ajuda muito na produtividade.

Outra possibilidade para um trabalho futuro é fazer a integração de frameworks front-end e medir o desempenho da aplicação com relação à quantidade de ferramentas utilizadas, pois pode ser que dependendo da quantidade de frameworks utilizados influencie negativamente no desempenho da aplicação.

## REFERÊNCIAS

- ANDRADE, T. F. Desenvolvimento Web com JavaServer Faces. 2010. Disponível em: <<https://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0CB0QFjAA&url=http%3A%2F%2Fwww.superhospedagens.com%2Fdl.php%3Ftype%3Dd%26id%3D18&ei=pDozVIHPDle1sQSyvIGABg&usg=AFQjCNFgB5xQC9AbLwqaT5mJPF-g2gfxLA&sig2=ulBJH48uHVeJcnC-VLGnHA>>. Acesso em: 06 out. 2014.
- ARUN. Proof of Concept(PoC): Using Richfaces and extJS widgets on the same JSF page. 2009. Disponível em: <<http://thus-spoke-the-computer.blogspot.com.br/2009/12/proof-of-conceptpoc-using-richfaces-and.html>>. Acesso em: 07 set. 2014.
- BOOTSTRAP. Disponível em: <<http://getbootstrap.com/>>. Acesso em: 06 set. 2014.
- BUSCH, M.; KOSH, N. Rich Internet Applications State-of-the-Art. 2009. Disponível em: <[http://uwe.pst.ifi.lmu.de/publications/maewa\\_rias\\_report.pdf](http://uwe.pst.ifi.lmu.de/publications/maewa_rias_report.pdf)>. Acesso em: 15 jun. 2014.
- DEB, B. et al. Rich Internet Applications (RIA). 2007. Disponível em: <<http://www.infosys.com/IT-services/application-services/white->

- papers/Documents/rich-internet-applications.pdf>. Acesso em: 18 jun. 2014.
- FERNANDES, G. G. Interface Humano Computador: prática pedagógica para ambientes virtuais. 2009. Disponível em: <[http://cead.ufpi.br/conteudo/material\\_online/disciplinas/video/livro\\_gildasio.pdf](http://cead.ufpi.br/conteudo/material_online/disciplinas/video/livro_gildasio.pdf)>. Acesso em: 04 out. 2014.
- GOOGLE TRENDS. Disponível em: <<https://www.google.com.br/trends/>>. Acesso em: 06 set. 2014.
- KAY, R. Rich Internet Applications. 2009. Disponível em: <<http://www.computerworld.com/article/2551058/networking/rich-internet-applications.html>>. Acesso em: 18 jun. 2014.
- KEEN, M. et al. Developing Web Applications using JavaServer Faces. 2012. Disponível em: <<http://www.redbooks.ibm.com/redpapers/pdfs/redp4883.pdf>>. Acesso em: 10 jul. 2014.
- NEIL, T. Rich Internet Application Screen Design. 2010. Disponível em: <<http://uxmag.com/articles/rich-internet-application-screen-design>>. Acesso em: 14 jun. 2014.
- PENA, R. Era da Informação. Disponível em: <<http://www.mundoeducacao.com/geografia/era-informacao.htm>>. Acesso em: 18 jun. 2014.
- PRIMEFACES. Disponível em: <<http://primefaces.org/>>. Acesso em: 06 set. 2014.
- QUI, X. Make jQuery UI and Bootstrap 2 work together in MVC template. Disponível em: <<http://blogs.msdn.com/b/webdev/archive/2013/08/01/make-jquery-ui-and-bootstrap-2-working-together-in-mvc-template.aspx>>. Acesso em: 07 set. 2014.
- ROSEINDIA. Rich Internet Application. 2011. Disponível em: <<http://www.roseindia.net/Technology-revolution/rich-internet-application.shtml>>. Acesso em: 04 out. 2014.
- SUDORES, S. The Performance Golden Rule. 2012. Disponível em: <<http://www.stevesouders.com/blog/2012/02/10/the-performance-golden-rule/>>. Acesso em: 13 set. 2014.
- TECHTERMS. Framework. 2013. Disponível em: <<http://www.techterms.com/definition/framework>>. Acesso em: 04 out. 2014.
- THE SERVER LABS. Rich Internet Applications. <<http://www.theserverlabs.com/resources/whitepapers.html?download=4:rich-internet-applications-frameworks-evaluation>>. Acesso em: 11 out. 2014.
- TOFFETTI et. Al. STATE-OF-THE-ART AND TRENDS IN THE SYSTEMATIC DEVELOPMENT OF RICH INTERNET APPLICATIONS. 2011. Disponível em: <[http://www.researchgate.net/publication/220538098\\_State-of-the-Art\\_and\\_trends\\_in\\_the\\_Systematic\\_Development\\_of\\_Rich\\_Internet\\_Applications](http://www.researchgate.net/publication/220538098_State-of-the-Art_and_trends_in_the_Systematic_Development_of_Rich_Internet_Applications)>. Acesso em: 14 jun. 2014.