

Integração de Tecnologias para Desenvolvimento de Sistemas Web, utilizando a metodologia AJAX

Hudson Simões Ribeiro Jefferson Grijó Ferreira Pereira Luís Paulo de Jesus Nunes Eduardo Barrére
Hudsonsr@gmail.com Jefferson.grijo@gmail.com linweda@yahoo.com.br barrere@inf.aedb.br

Associação Educacional Dom Bosco (AEDB) - Resende, RJ, Brasil

RESUMO

Este artigo aborda o processo de desenvolvimento de sistemas Web no que se refere às tecnologias atualmente disponíveis, suas características, vantagens e impactos. Aborda também a metodologia AJAX como uma possibilidade viável para esse tipo de desenvolvimento. Também é apresentado um estudo de caso desenvolvido pelos autores ilustrando o uso dessas tecnologias.

Palavras-Chave: Sistemas Web. AJAX. Plataforma Java. MVC.

1. INTRODUÇÃO

Nos primórdios da Internet, consistia apenas em conteúdos estáticos escritos em HTML (*Hypertext Markup Language*), tornando qualquer um que desenvolvesse páginas, especialista da internet.

Algum tempo depois, através da tecnologia CGI (*Common Gateway Interface*) foi possível a criação de páginas com o seu conteúdo dinâmico. O CGI permite ao servidor passar solicitações http a um programa externo, e receber o resultado deste programa, enviando ao browser cliente e assim o Perl tornou-se a linguagem mais popular para escrever programas CGI.

Após o CGI, surgiram novas e melhores tecnologias que sempre disputaram o ambiente de desenvolvimento de aplicativos web, como:

- **ColdFusion:** Produzido pela Allaire, oferece tags personalizadas como HTML, que podem ser usada para realizar uma série de operações especialmente consultar banco de dados. Durante um período de tempo foi considerada a principal tecnologia para a programação de aplicativo web.
- **Server-Side JavaScript (SSJS):** É uma extensão da linguagem JavaScript, a linguagem de script que ainda regula a programação web do lado cliente. SSJS pode acessar classes Java distribuídas do lado servidor usando a tecnologia *LiveWire* da Netscape.

- **PHP:** É uma tecnologia de fonte aberta que amadureceu nos últimos anos, oferecendo fácil desenvolvimento de aplicativo web, com seu gerenciamento de sessão e inclui uma funcionalidade interna, tal com upload de arquivo. Hoje, como PHP 5.0 partiu para uma abordagem orientada a objeto, possui uma fatia considerável de adeptos.
- **Servlet:** É a base do desenvolvimento de aplicativo web usando a linguagem de programação Java.
- **Java Server Pages (JSP):** É uma extensão da tecnologia Servlet.
- **Active Server Pages (ASP):** ASP da Microsoft emprega tecnologias de script que funcionam em plataforma Windows, mesmo havendo esforços para transportar essa tecnologia para outros sistemas operacionais. ASP Windows funciona com o servidor web *Internet Information Server* (IIS – Servidor de Informações de Internet).
- **Active Server Pages .NET (ASP .NET):** Essa tecnologia é parte da iniciativa .NET da Microsoft. Interessantemente, a .NET Framework emprega um tempo de execução chamado *Common Language Runtime* que é muito semelhante a Java Virtual Machine e oferece uma vasta biblioteca de classe, disponível a todas as linguagem .NET e de páginas ASP .NET. Ela introduziu várias novas tecnologias, inclusive gerenciamento de estado, que não depende de cookies ou reescrita de URL.

No passado, ASP e Servlet/JSP eram as principais tecnologias usadas no desenvolvimento de aplicativo web. Com o lançamento de ASP .NET não é difícil prever que essa tecnologia se tornará o principal concorrente de Servlet/JSP, cada qual tem seus próprios fãs e não é fácil prever qual será a vencedora, o resultado mais provável é que nenhuma das duas será a vencedora absoluta do mercado; Ao invés, provavelmente as tecnologias correrão lado a lado nos próximos anos. Kurniawan (2002).

A Seguir são descritas as tecnologias para desenvolvimento Web e o estudo de caso desenvolvido pelos autores.

2. JSP (JAVA SERVER PAGES), SERVLETS E JAVABEANS

A tecnologia Servlet é a base do desenvolvimento de aplicativo web usando a linguagem de programação Java. Ela é uma das tecnologias Java mais importante, e é a tecnologia subjacente para outra tecnologia Java, popular para desenvolvimento de aplicativo:

Java Server Pages (JSP). Portanto, entender a tecnologia Servlet e sua arquitetura é importante, se você quer ser um desenvolvedor de Servlet. Ainda que você planeje desenvolver o seu aplicativo web Java usando apenas páginas JSP, entender a tecnologia Servlet ajuda a montar um aplicativo JSP mais eficiente e efetivo, pois esta página será compilada em Servlets pelo servidor. Kurniawan (2002).

Java oferece ainda a tecnologia JavaBeans, que nada mais é do que uma classe que possui um construtor sem argumentos e com métodos de acesso do tipo get e set. A combinação destas tecnologias torna a programação e a criação de páginas um processo um tanto complicado quanto as maneiras tradicionais, porém com acesso a inúmeras funcionalidades da biblioteca Java. Uma vez criada uma página, todo seu código pode ser utilizado por outras. Além da reutilização das classes, estas tecnologias tornam mais rápida a apresentação do conteúdo para o visitante, já que todo trabalho pesado de executar os Servlets fica a cargo do contentor de Servlet no servidor, e após processar uma solicitação ele permanece na memória, aguardando por uma outra solicitação.

Outras funcionalidades são a portabilidade, que semelhante as outras tecnologias Java, os aplicativos Servlets são portáteis, podendo movê-los para outros sistemas operacionais sem dificuldades; Robustez, Servlets são gerenciados pela Java Virtual Machine, não havendo a necessidade de se preocupar com galha de memória ou coleta de resíduos; e aceitação difundida no mercado, visto que um grande número de fabricantes trabalham com tecnologias baseadas em Java.

3. CSS LAYOUT

Cascading Style Sheets (CSS - Folhas de Estilo em Cascata), é uma tecnologia que nos permite criar páginas web de uma maneira mais exata. Esta tecnologia permite fazer muitas coisas que não era possível utilizando somente HTML, como incluir margens, tipos de letra, fundos, cores, etc.

CSS pode ser escrito dentro do código HTML da página web ou em um arquivo a parte e “linkar” a página com esse arquivo. Antes, toda a estrutura do site era feita em incontáveis tabelas, e hoje, com a criação de CSS, toda a arte de estruturação de um site contendo milhares de páginas pode ser feita através de um único documento, definido desde as folhas padrões até a largura do caracter espaço.

Outro fator importante a se destacar nesta tecnologia é o tamanho em bytes. Por não mais serem “acorrentadas” as tabelas, a estrutura do site se torna mais leve e dinâmica. A codificação gerada para se manter uma estrutura do tipo CSS LAYOUT é bem menor se comparada às formas tradicionais de se criar um layout. Em média, seriam gastas seis linhas de código para se criar uma tabela contendo uma linha e uma coluna. Já no CSS layout, uma simples estrutura pode ser criada com apenas duas linhas de código. ANGEL ALVAREZ.

4. JAVASCRIPT

JavaScript é uma linguagem de programação Web que é executada do lado cliente, desenvolvida pela Netscape. Essa linguagem possibilita adicionar recursos dinâmicos às páginas HTML.

4.1 FUNCIONAMENTO DO JAVASCRIPT

Em sua essência, a linguagem JavaScript atua inserida no meio do código HTML de páginas Web. Essa inserção pode ser feita de várias formas, desde a inclusão de código numa área determinada ou a inserção em vários pontos da página. Tudo depende da utilidade necessária. Quando o navegador de Internet “lê” a página e se possuir tal recurso, deve “entender” as instruções do JavaScript e executá-las.

O fato do JavaScript não exigir a instalação de softwares especiais para execução (assim como ocorre com linguagens mais sofisticadas, como PHP e ASP) o fez ter grande popularidade, já que qualquer página HTML pode conter recursos em JavaScript. Tudo depende, basicamente, da capacidade do navegador em entender as instruções (hoje em dia, praticamente todos os navegadores possuem essa aptidão).

Embora esta tecnologia seja antiga, com o surgimento de novas técnicas, ela vem sendo amplamente utilizada para a apresentação dinâmica de algum conteúdo, ou mesmo nas tradicionais verificações de formulários. ALECRIM.

5. XML

XML é um subtipo de SGML (*Standard Generalized Markup Language* - Linguagem Padronizada de Marcação Genérica) capaz de descrever diversos tipos de dados. Seu propósito principal é a facilidade de compartilhamento de informações através da Internet.

A linguagem XML é definida como o formato universal para dados estruturados na Web. Esses dados consistem em tabelas, desenhos, parâmetros de configuração, etc. A linguagem então trata de definir regras que permitem escrever esses documentos de forma que sejam adequadamente visíveis ao computador. ANGEL ALVAREZ.

5.1 DIFERENÇA ENTRE O HTML E O XML

O HTML e o XML tem lá suas semelhanças, sendo a principal o fato de utilizar tags (palavras-chaves e parâmetros). Em ambas as linguagens, cada tag consiste em duas partes, uma que inicia e outra que fecha o comando. No entanto, em muitos casos, se uma tag é aberta no HTML e não é fechada, a página é exibida mesmo assim. Já no XML, se houver qualquer erro desse tipo, a aplicação simplesmente pára. Percebe-se com esse exemplo, que o HTML é uma linguagem mais tolerante, enquanto o XML é altamente rígido. Isso pode até parecer uma desvantagem, mas se for, é compensada pela extensibilidade do XML.

Para um melhor entendimento, veja o seguinte fato: no HTML, a tag `<p>` `</p>` indica o início e o fim de um parágrafo. No XML, as tags são usadas para definir blocos de dados. O que isso quer dizer? Quer dizer que, `<p>` `</p>` podem significar qualquer coisa que o programador desejar. Por exemplo, `<p>` `</p>` podem significar peso, pessoa, nome, endereço, classe, carro, enfim, o que o usuário quiser que represente. Por essa característica, o XML é até considerado por muitos uma linguagem capaz de gerar outras linguagens, visto que quem define os comandos e suas funções é o programador. A praticidade é tanta que torna-se possível um usuário criar uma coleção própria de tags e aplicá-las nas páginas e documentos que desejar. Como exemplo de linguagens baseadas em XML podemos citar: RDF, SMIL, MathML, NCL, XSIL e SVG.

O XML vem sendo amplamente explorado pelo seu grande poder de portabilidade, sendo capaz de transmitir informações entre diferentes plataformas, assim como também sua maleabilidade e versatilidade ao se poder criar a tag que é necessária para sua aplicação. ANGEL ALVAREZ.

6. AJAX

O *Asynchronous JavaScript and XML* não é uma tecnologia e sim uma técnica de programação que possui a capacidade de retornar uma informação de forma dinâmica sem a necessidade de se recarregar toda a uma página e seu conteúdo.

Esta técnica utiliza várias tecnologias citadas anteriormente, principalmente JavaScript, gerenciando a comunicação cliente servidor, e XML encapsulando a informação, cada uma a evoluir a sua maneira e a convergir de uma maneira poderosa, podemos ainda citar:

- Apresentação baseada em padrões, que utilizam XHTML e CSS.
- Exibição e interação dinâmicas por meio de DOM (*Document Object Model*).
- Troca e manipulação de dados por meio do uso de XML e XSLT.
- Recuperação assíncrona de dados com XMLHttpRequest.
- E JavaScript, que junta tudo.

A implementação mais comum do Ajax é baseada na classe *XMLHttpRequest*, que faz o meio de campo entre o servidor e o browser. A técnica ganhou fama em serviços como o Gmail, Orkut, Google News, Amazon e Google Local ” Google Desktop Search”, que empregam muitos recursos em Ajax. MELGAR.

6.1 VANTAGENS

As grandes vantagens de utilizar Ajax num site é que permite proceder a alterações na página com muito menos transferência de informação, pois podem ser transferidos apenas os dados novos em cada alteração, sendo o HTML do site reutilizado ou criado no cliente.

Por outro lado, como apenas partes das páginas são atualizadas, e essa atualização é normalmente muito mais rápida do que se toda a página fosse atualizada, o utilizador chega a ter a sensação de que tudo acontece em tempo real, localmente no seu computador.

Não há necessidade de instalação ou configurações especiais, ou ficar preso a uma única plataforma, todas as tecnologias utilizadas no desenvolvimento são multiplataformas e rodam na maioria dos browsers. Como o Ajax é composto por várias tecnologias, cada qual com a sua função, fica mais fácil a sua manutenção.

6.2 DESVANTAGENS

A grande desvantagem da utilização massiva de Ajax são a usabilidade e a indexação dos sites. Pela parte da usabilidade, o fato de uma grande quantidade das funcionalidades do site serem implementadas em Javascript, torna mais difícil a manutenção de funções a que os utilizadores estão habituados, como o botão *Voltar/Back* do browser, como as sucessivas páginas são criadas no browser do cliente esta operação não funciona.

Por outro lado, se as alterações são mais que alterações do conteúdo, mas navegação no próprio conteúdo, pode tornar-se impossível (ou pelo menos complexo) guardar links diretos para determinado conteúdo que se pretende lembrar.

Esta questão da navegação coloca-se ainda com mais relevância quando se trata a indexação do site, pois se as listagens são geradas em Javascript, e os “robôs” de indexação dos sites de pesquisa não executam o Javascript, na maioria dos casos não vão indexar corretamente o site.

Outro problema por vezes apresentado é o dos tempos de resposta, que numa rede com tempos de resposta elevados podem gerar timeouts e tempos de espera que os utilizadores não percebem. Nesta situação os utilizadores devem ser alertados visualmente. MELGAR.

7. ESTUDO DE CASO SPEL – SISTEMA DE PUBLICAÇÃO ELETRÔNICA

Com o objetivo principal de ajudar o editor nas tarefas de publicação e gerenciamento de uma revista, foi proposta a elaboração de um sistema que visa facilitar todo processo de publicação, desde o envio de artigos pelos colaboradores, até a sua efetiva publicação, o SPEL.

Ele é dividido em 4 módulos:

- **Edição:** responsável por todas as definições de Sessões da Revista, quais artigos serão publicados naquele número e a edição dos mesmos.
- **Gerenciamento:** responsável pelo controle de artigos já publicados e a publicar, assim como estatísticas de visualização de cada artigo, visualização de sessões e visitação da revista.
- **Publicação/Exibição:** responsável pela geração dos arquivos-fonte da revista.

- **Controle de Usuário:** responsável pelo controle de acesso à revista, liberando ao usuário o acesso apenas ao que tem direito.

O grande diferencial desta ferramenta é a integração de todas as etapas até a publicação, e também pós-publicação. A navegação da revista é bastante intuitiva e a princípio em dois (2) idiomas (português e inglês), e todo o conteúdo da revista está armazenado em Banco de Dados, permitindo melhor atualização e manutenção da revista.

O Sistema pode ser utilizado tanto no meio corporativo, quanto no acadêmico, disponibilizando uma revista/jornal em sua rede interna e também pelos sites que já trabalham com publicação de revistas e desejam uma ferramenta versátil e de fácil utilização.

7.1 ARQUITETURA E IMPLEMENTAÇÃO

A figura 1 apresenta a arquitetura do SPEI, apresentando o modelo MVC (*Model-View-Controller*), os usuários e permissões, as classes de dados, o fluxo dos objetos e o processo de desenvolvimento da aplicação. O MVC garante a separação de tarefas, facilitando assim a reescrita de alguma parte e a manutenção do código.

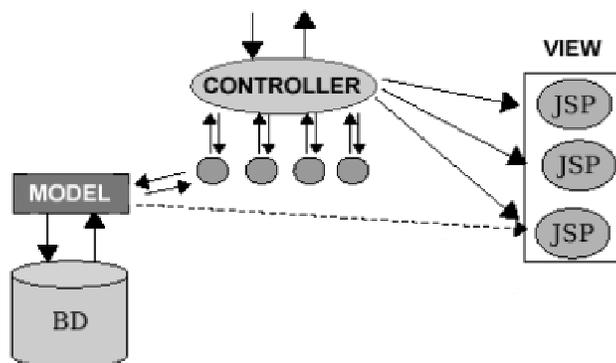


Figura 1. Modelo MVC.

Na camada VIEW, responsável por apresentar os resultados na página Web, utilizamos a tecnologia JSP (Java Server Pages) aproveitando seus melhores recursos, e visando melhorar a interação com o usuário, utilizamos em alguns componentes a metodologia AJAX (pacote `org.ajaxtags.demo.servlet`), como na tabela inferior e a busca de um determinado registro das páginas de cadastro e a navegação entre abas nas páginas de configuração da revista e configuração de seção.

Na camada *CONTROLLER*, utilizamos um servlet (pacote controle) que despacha todas as solicitações http às páginas de apresentação correspondentes, com base na solicitação de URL, parâmetros de entrada e estado da aplicação. Deste modo, temos um único ponto de controle para segurança e registro, encapsulando as entradas de dados e facilitando o controle de sessão.

Na camada *MODEL*, utilizamos logicamente JavaBeans (pacote Serviço), representando as classes de modelo da nossa aplicação, facilitando o armazenamento e a recuperação de dados.

Utilizamos ainda outros Servlets (pacote ClassesApoio) responsáveis por funções específicas como conexão com banco de dados, configuração de idioma, upload de arquivos, etc.

7.2 USUÁRIOS E PERMISSÕES

Os usuários do sistema são divididos em 4 perfis básicos, os quais são representados pela classe Perfis. Nela, são descritos os perfis de usuário *ADMINISTRADOR*, responsável pelo cadastro da revista, cadastrar e delegar permissões ao editor e consultar as estatísticas de visitação; usuário *EDITOR*, responsável pelo cadastro dos colaboradores, configuração do layout da revista e dos artigos, manter artigo, escolha de artigos que serão publicadas naquele determinado nº, disponibilizar a revista, consultar as estatísticas de visitação, consulta de artigos publicados e a publicar; usuário *COLABORADOR*, que poderão postar e consultar artigos; E finalmente, usuário *VISITANTE*, Só tem acesso a uma revista já publicada.

O sistema prevê o cadastro de perfis intermediários para atender qualquer nível de acesso.

A figura 2 abaixo ilustra a interação do usuário Editor com o sistema, de acordo com a diagramação da UML.

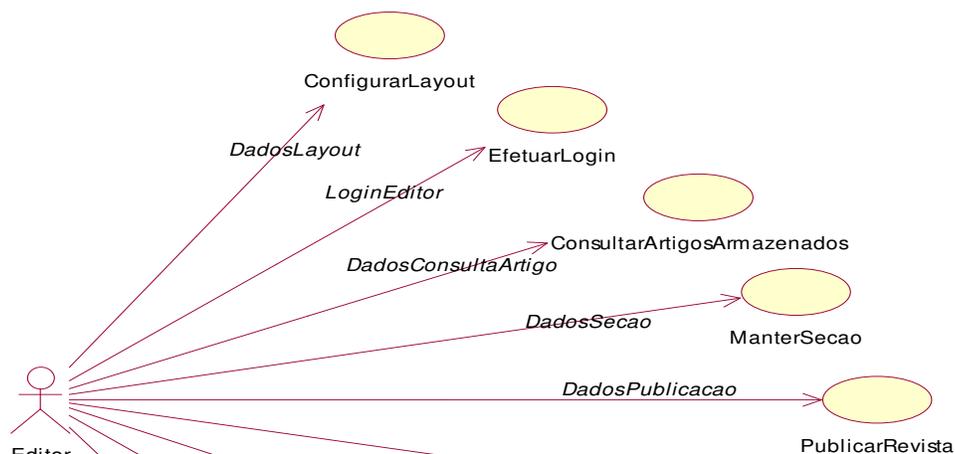


Figura 2. Diagrama de caso de uso do ator editor.

7.3 CLASSES DE DADOS

O sistema possui as seguintes classes de dados:

- Revista, responsável pelo cadastro da revista e por disponibilizá-la para edição e publicação;
- Secao, responsável pelo cadastro de seções da revista;
- Artigos, responsável pelo cadastro dos artigos;
- Foto, responsável pelo cadastro de fotos dos artigos;
- Usuario, responsável pelo cadastro de usuários;
- Perfil, responsável pelo cadastro de perfis de acesso à revista;
- Configuracao, responsável pela configuração da revista;
- ConfSecao, responsável pela configuração das seções;
- ArtigosPublicados, responsável pelo controle de artigos publicados;
- Visitacao, responsável pelo controle de visitação.

7.4 FLUXO DOS OBJETOS

Os objetos criados pela camada VIEW são objetos Java (JavaBeans- objetos com propriedades e métodos get/set, e método de negócio). Eles trafegam da interface para a

camada de controle sofrendo os processos de negócio necessários, até a persistência no banco de dados.

7.5 PROCESSO DE DESENVOLVIMENTO

O processo de análise e projeto do sistema segue as bases Prototipação Evolutiva. Baseado em ciclos evolutivos, a modelagem do sistema fora balizada pela construção de casos de uso e o agrupamento de tecnologias a serem utilizadas.

Seguido pela prototipação da interface de usuário, bem como o esquema navegacional, definiu-se o modelo de interação do usuário. Em paralelo, os objetos de dados foram modelados e as devidas informações relativas à persistência foram inseridas. Na seqüência, a implementação das regras de negócio e ajustes mais finos na interface do usuário foram efetuados. Finalmente, testes de usabilidade, performance, correção de erros e ajustes foram aplicados.

8. CONCLUSÃO

Este artigo aborda de forma sucinta a Integração de tecnologias para desenvolvimento de sistema em JSP, utilizando a metodologia AJAX, com o estudo de caso do sistema SPEI. Embora existam várias maneira de se implementar uma aplicação, foi visto que utilizar JSP, Servlet e JavaBeans aplicando o modelo MVC obtém-se um código mais limpo e de fácil manutenção, e agregando a metodologia AJAX no desenvolvimento da interface, torna o software mais atrativo aos olhos do usuário.

E importante lembrar que utilizar ou não o modelo MVC depende muito da aplicação, no case de aplicações robustas ou escaláveis, este modelo é altamente recomendado independente da linguagem utilizada.

A utilização das tecnologias abordadas, mostraram-se capazes de suprir as necessidades no desenvolvimento do SPEI, cada qual com sua singularidade, tornando possível adequar o sistema com as novas tendências no desenvolvimento web. Além deste aspecto, utilizar estas tecnologias, até então, pouco aplicadas, nos auxiliou a enxergar uma forma pratica de criar aplicativos web e que ainda sim, fosse tão eficaz quanto as formas mais difundidas.

9. REFERÊNCIAS

- [1] KURNIAWAN, BUDI. Java para a Web com Servlets, JSP e EJB. Ciência Moderna, 2002. ed.1.
- [2] GUIMARÃES, RENATO. ASP.NET - Uma mudança radical no desenvolvimento web... <<http://www.imasters.com.br/artigo/1624>>. Acessado em 10 de Abril de 2006.
- [3] LIMA, GLEYDSON. Introdução a Enterprise JavaBeans – PARTE 1. <http://www.j2eebrasil.com.br/jsp/tutoriais/tutorial.jsp?idTutorial=010_001>. Acessado em 3 de Março de 2006.
- [4] EIS, DIEGO. Browsers e seus diferenciais. <<http://www.tableless.com.br/>>. Acessado em 21 de Abril de 2006.
- [5] ANGEL, MIGUEL ALVAREZ. O que é CSS. <<http://www.criarweb.com/artigos/173.php>>. Acessado em 29 de Maio de 2006.
- [6] ANGEL, MIGUEL ALVAREZ O que é XML. <<http://www.criarweb.com/artigos/203.php>>. Acessado em 12 de Março de 2006.
- [7] ALECRIM, EMERSON. Linguagem JavaScript. <<http://www.infowester.com/lingxml.php>>. Acessado em 5 de Maio de 2006.
- [8] CAELUM, Ensino e soluções em Java. Java para desenvolvimento web. <<http://www.caelum.com.br/caelum/index.jsp>>. Acessado em 23 de Abril de 2006.
- [9] MELGAR, ROBINSON. Ajax Asynchronous Javascript and XML. Revista www.com.br, n. 69, p.26-27, abril. 2006. [LARMAN 2003] Larman, Craig. : *A Manager's Guide*. Addison-Wesley Professional, 2003.