

UNIVERSIDADE DO EXTREMO SUL CATARINENSE
CURSO DE CIÊNCIA DA COMPUTAÇÃO

DIEGO BÚRIGO ZACARÃO

APLICAÇÃO EXPERIMENTAL DO CONCEITO DE *E-PROCUREMENT*
UTILIZANDO AJAX: UM CASO DE USO NA UNESC

CRICIÚMA, NOVEMBRO DE 2007

DIEGO BÚRIGO ZACARÃO

**APLICAÇÃO EXPERIMENTAL DO CONCEITO DE *E-PROCUREMENT*
UTILIZANDO AJAX : UM CASO DE USO NA UNESC**

Trabalho de Conclusão de Curso
Apresentado para Obtenção do Grau de
Bacharel em Ciência da Computação da
Universidade do Extremo Sul Catarinense.

Orientador: Prof. MSc. Rogério Antônio
Casagrande

Co-orientador: Esp. Fabrizio Colombo
Machado

CRICIÚMA, NOVEMBRO DE 2007

DIEGO BÚRIGO ZACARÃO

**Aplicação Experimental do Conceito de *E-procurement* Utilizando
AJAX : Um caso de uso na UNESC**

Submetido ao corpo docente do Curso de Ciência da Computação da
Universidade do Extremo Sul Catarinense como um dos requisitos para obtenção do
grau de Bacharel em Ciência da Computação.

Profa. MSc. Ana Claudia Garcia Barbosa
Coordenadora do Curso de Ciência da Computação

Banca Examinadora:

Prof. MSc Rogério Antônio Casagrande
Orientador

Esp. Fabrizio Colombo Machado
Co-Orientador

Prof. Fabrício Giordani

Prof. Esp. Arildo Sônego

Aos familiares e amigos que sempre souberam me compreender e incentivar para que chegássemos ao fim de mais uma jornada.

AGRADECIMENTOS

Primeiramente gostaria de agradecer por ser um dos contemplados, sabe-se lá porque, por habitar este planeta, que independentemente do que se possa crer, está a disposição de todos àqueles que tocam-no com seus pés.

Entretanto, isso para mim não seria possível se não fosse os meus pais, Renato e Marise, que definitivamente me trouxeram ao mundo para que eu pudesse desfrutar de tudo isso que nos é proporcionado.

Agradeço também pela existência dos meus dois irmãos, Eduarda e em especial ao Darlan, em nossa família, a nossa eterna criança, que por algum motivo na fase de gestação, foi impedido de desenvolver seu raciocínio e inteligência normalmente durante sua fase de crescimento. Sem ele, nossa família não seria nem pior, nem melhor, simplesmente não seria a mesma!

Aos meus avós, não posso deixar de citá-los e agradecê-los por terem em mim, seu primeiro neto, as expectativas e a fé de que eu irei alcançar todos os meus objetivos nesta vida.

Agradeço de coração a minha namorada Dayana, por ter pulado de alegria e me dado todo o apoio quando fiz a surpresa de aparecer com a pasta do curso de Ciência da Computação, dizendo-lhe que havia me matriculado, mesmo sem saber ao certo como iria fazer para pagar as mensalidades.

Sou grato também a todos os familiares e amigos do peito, que sempre estão ao meu lado em todas as circunstâncias. Amo todos vocês.

Por último e não menos importante, agradeço ao meu orientador Rogério e ao meu co-orientador Fabrizio, pelo tempo dedicado a este trabalho.

“Talk is cheap. Show me the code.”

Linus Torvalds

RESUMO

Este trabalho tem como objetivo apresentar uma alternativa para tornar a Web mais interativa, proporcionando maior usabilidade para os usuários, denominada AJAX. Sendo o AJAX uma técnica de programação, a mesma foi aplicada no desenvolvimento de um protótipo de Pregão Eletrônico para a realização de cotações na Universidade do Extremo Sul Catarinense – UNESC. Para que tal implementação pudesse ser desenvolvida, conceitos de *business to business* e *e-procurement* foram estudados, além de várias tecnologias de programação Web que compõem o AJAX. Ao final do estudo, pode-se observar como o AJAX pode tornar o ambiente Web mais interativo, com a conotação de tornar as aplicações Web algo mais próximo das aplicações *Desktop* convencionais.

Palavras-chaves: AJAX, Interatividade na Web, Usabilidade na Web, Aplicações Web, *E-procurement*, Pregão Eletrônico.

ABSTRACT

The aim of this paper is to present an alternative to make the Web more interactive, providing greater usability for users, denominated AJAX. As AJAX is a programming technique, it has been applied in the development of a Web application prototype to realize quotations of prices at the University of UNESC (Universidade do Extremo Sul Catarinense). In order to develop this implementation, concepts of Business to Business and E-procurement were studied, besides several programming technologies that composes AJAX. In the end of this study, it is possible to observe how AJAX can make Web environment more interactive, with the connotation to turn the Web applications something close of conventional Desktop applications

Key-words: AJAX, Web Iterative, Web Usability, Web Application, *E-procurement*.

LISTA DE ILUSTRAÇÕES

Figura 1. Hierarquia DOM em uma página HTML.....	22
Figura 2. Exemplo da utilização da API DOM.....	23
Figura 3. Resultado final da página de exemplo DOM.....	24
Figura 4. Exemplo de documento XML.....	25
Figura 5. Comparação entre os formatos XML e JSON.....	27
Figura 6. Instanciando um objeto <i>XMLHttpRequest</i>	28
Figura 7. Lógica de aplicativo Web clássico.....	33
Figura 8. Lógica de aplicação AJAX.....	34
Figura 9. Comparação de tráfego de dados.....	35
Figura 10. Seleção de cidade por estados utilizando AJAX.....	37
Figura 11. Fluxo de interações entre as tecnologias que compõem o AJAX.....	38
Figura 12. Diagrama de atividades do processo de compra atual da UNESCO.....	52
Figura 13. Carregando a aba via AJAX.....	57
Figura 14. <i>Autocomplete</i> para procura de produtos na requisição.....	58
Figura 15. Tabela ordenada por produto utilizando o <i>plugin tablesorter</i>	59
Figura 16. Validação de formulário.....	60
Figura 17. Calendário dinâmico escrito em <i>JavaScript</i>	60
Figura 18. Bloqueio da interface com <i>feedback</i> para o usuário.....	61
Figura 19. Formulário de manipulação de requisição de compra.....	62

LISTA DE SIGLAS

AJAX	<i>Asynchronous Javascript And Xml</i>
B2B	<i>Business to Business</i>
B2C	<i>Business to Consumer</i>
C2C	<i>Consumer to Consumer</i>
CSS	<i>Cascading Style Sheets</i>
DOM	<i>Document Object Model</i>
ERP	<i>Enterprise Resource Planning</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
HTTPS	<i>HyperText Transfer Protocol Secure</i>
JSON	<i>JavaScript Object Notation</i>
MVC	<i>Model-view-controller</i>
PHP	<i>PHP: Hipertext PreProcessor</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	<i>Structured Query Language</i>
URI	<i>Uniform Resource Identifier</i>
XML	<i>eXtensible Markup Language</i>
XSL	<i>eXtensible Stylesheet Language</i>
XSLT	<i>eXtensible Stylesheet Language Transformation</i>
W3C	<i>World Wide Web Consortium</i>

SUMÁRIO

1 INTRODUÇÃO	14
1.1 OBJETIVO GERAL.....	15
1.2 OBJETIVOS ESPECÍFICOS	15
1.3 JUSTIFICATIVA.....	15
1.4 ESTRUTURA DO TRABALHO.....	16
2 TECNOLOGIAS DE PROGRAMAÇÃO WEB.....	18
2.1 HTML.....	18
2.2 CSS.....	19
2.3 <i>JAVASCRIPT</i>	20
2.4 DOM.....	21
2.5 XML.....	24
2.5.1 XSL	26
2.5.2 XSLT.....	26
2.6 JSON.....	26
2.7 <i>XMLHTTPREQUEST</i>	27
2.8 PHP.....	28
2.8.1 Framework CodeIgniter para desenvolvimento em PHP.....	29
2.9 MYSQL.....	29
3 TÉCNICA AJAX	31
3.1 REPENSANDO AS APLICAÇÕES WEB.....	31
3.2 CONHECENDO O AJAX.....	32
3.3 OS PRINCÍPIOS DO AJAX.....	32
3.3.1 O Navegador Hospeda Uma Aplicação, não o Conteúdo.....	33
3.3.2 O Servidor Fornece Dados, não Conteúdo.....	35
3.3.3 A Interação do Usuário com o Aplicativo pode ser Fluente e Contínua.....	36

3.3.4 A Codificação de um Aplicativo AJAX Requer Disciplina.....	36
3.4 AJAX NA PRÁTICA.....	37
3.5 QUANDO SE DEVE UTILIZAR O AJAX.....	39
3.6 SEGURANÇA NO AJAX.....	40
3.7 BIBLIOTECAS <i>JAVASCRIPT</i> COM SUPORTE AO AJAX.....	41
4 E-BUSINESS: CONCEITOS DE NEGÓCIOS APLICADOS A INTERNET....	43
4.1 <i>BUSINESS TO BUSINESS</i>	43
4.2 <i>E-PROCUREMENT</i> : APERFEIÇOANDO O PROCESSO DE COMPRA.....	44
4.2.1 Entendendo o Processo de Compras.....	45
4.2.2 As Metas do <i>E-Procurement</i>.....	46
5 TRABALHOS CORRELATOS.....	48
5.1 UTILIZAÇÃO DE AJAX NO DESENVOLVIMENTO DE SISTEMAS WEB....	48
5.2 APLICAÇÕES WEB ENRIQUECIDAS COM AJAX.....	48
5.3 AJAX APLICADO A UM SISTEMA DE PLANO DE NEGÓCIOS.....	49
5.4 MODELAGEM DE UM SISTEMA <i>E-PROCUREMENT</i>	49
5.5 ANÁLISE DO ENTENDIMENTO DA UTILIZAÇÃO DO <i>E-PROCUREMENT</i> .	50
6 CASO DE USO DA UNESC.....	51
6.1 O ATUAL PROCESSO DE COTAÇÕES DA UNESC.....	51
6.2 O PROCESSO DE COTAÇÃO PROPOSTO.....	53
6.2.1 Definindo as partes envolvidas no processo de cotação.....	54
6.2.1.1 Solicitante.....	55
6.2.1.2 Compras.....	55
6.2.1.3 Fornecedor.....	55
6.2.1.4 Financeiro.....	56
6.3 FUNCIONALIDADES IMPLEMENTADAS NO PROTÓTIPO.....	56
6.3.1 <i>Tabs</i>.....	57

6.3.2 Autocomplete.....	58
6.3.3 Tablesorter.....	59
6.3.4 Form.....	59
6.3.5 Validate.....	60
6.3.6 Calendar.....	60
6.3.7 BlockUI.....	61
6.3.8 Implementações específicas da aplicação.....	61
CONCLUSÃO.....	64
REFERÊNCIAS.....	67
APÊNDICE A – DIAGRAMA ER DO PROTÓTIPO.....	69
APÊNDICE B – DIAGRAMA DE CASO DE USO - SOLICITANTE.....	70
APÊNDICE C – DIAGRAMA DE CASO DE USO - COMPRAS.....	71
APÊNDICE D – DIAGRAMA DE CASO DE USO - FORNECEDOR.....	72
APÊNDICE E – DIAGRAMA DE CASO DE USO - FINANCEIRO.....	73
APÊNDICE F – ARTIGO SOBRE O TRABALHO.....	74

1 INTRODUÇÃO

A realização de tomadas de preço em uma corporação, seja ela privada ou pública, é um processo muitas vezes burocrático e de alto custo. Com isso a idealização de um processo de cotação informatizado, que faça uso dos recursos de Internet para gerenciar os pedidos de compras de mercadorias, denominado *e-procurement*, é uma alternativa que pode trazer redução de custos às instituições.

O processo de cotação atual não informatizado da UNESC, devido a demanda e tamanho da instituição, é cansativo e lento, ocupando o tempo que poderia ser utilizado para a execução de outras atividades. Além disso, o custo com telefonia e o tempo dispensado nos atendimentos ao telefone de fornecedores nas negociações, é um ponto extremamente crítico do processo. Outros problemas como o histórico e controle da relação de fornecedores e produtos, também são pontos relevantes levados em consideração. A falta de uma estrutura informatizada e disponível na Web, torna o gerenciamento desse processo complexo e de alto custo.

A utilização de ferramentas integradas à Internet que sejam dinâmicas e que possam proporcionar maior agilidade e interatividade aos usuários, além de somente poder consultar páginas Web, é uma tendência mundial. Com este propósito a técnica AJAX utiliza-se de tecnologias aplicadas a Web para fazer requisições assíncronas ao servidor. Este fato proporciona inúmeras maneiras de se trabalhar com componentes dinâmicos em páginas Web, pois permite requisitar dados de forma assíncrona, sem a necessidade de recarregar uma página por completo.

Com o intuito de resolver o problema relacionado ao *e-procurement*, usufruindo de uma aplicação que siga as novas tendências de aplicações Web, será realizado um estudo de caso referente a utilização do conjunto de tecnologias que compõem o AJAX, no desenvolvimento de um Pregão Eletrônico com base nas necessidades da UNESC. O protótipo proposto usará a Web para entrar em contato com

os fornecedores e assim os mesmos poderão optar em cotar ou não os materiais requisitados em um formulário no sistema da própria instituição. Por meio deste trabalho pode-se fornecer uma maneira ágil para a realização de tomadas de preço. Este processo deve comportar os pedidos de mercadorias aos fornecedores, juntamente com controle e privilégio de acesso aos usuários.

1.1 OBJETIVO GERAL

Desenvolver uma aplicação *e-procurement* experimental utilizando as tecnologias que compõem a técnica AJAX: Estudo de caso na UNESC.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos deste trabalho são:

- a) compreender conceitos de programação;
- b) entender o funcionamento de um sistema de *e-procurement*;
- c) compreender conceitos de *e-business* e *business-to-business* (B2B);
- d) aplicar a técnica AJAX no desenvolvimento de um protótipo de Pregão Eletrônico;
- e) verificar as vantagens da utilização do AJAX.

1.3 JUSTIFICATIVA

O fato das aplicações tenderem ao ambiente Web, faz com que o comércio e negociações do cotidiano, estejam cada vez mais presentes no meio eletrônico. Isto impulsiona o surgimento de novas definições e termos usados no mundo dos negócios. Uma destas definições é o *e-procurement*, uma modalidade *Business-to-Business*, que

refere-se a conceitos relacionados a negociações de mercadorias a partir da oferta de compra, por meio de cotações eletrônicas.

A evolução das aplicações Web faz com que a busca por novas técnicas e tecnologias de programação seja constante, a fim de otimizar e obter maior performance das aplicações. O AJAX vem como uma alternativa a esse contexto e tem como princípio básico a otimização da quantidade de dados transferidos nas requisições feitas entre cliente e servidor, fazendo com que a comunicação se dê de uma forma mais eficiente e rápida, requisitando ao servidor somente os dados necessários a cada interação do usuário com a página, além de proporcionar um considerável aumento das possibilidades destas interações.

Com o objetivo de explanar os benefícios da técnica AJAX, foi escolhido um caso de uso na UNESCO referente a um Pregão Eletrônico, no qual o mesmo deve abordar conceitos de relacionamento entre empresa e fornecedor (*business-to-business*), a fim de oferecer uma alternativa que possa agilizar o processo de aquisição e manutenção de estoque, reduzir custos nas negociações e controlar o fluxo de aprovação da compra de materiais utilizando os recursos de Internet.

1.4 ESTRUTURA DO TRABALHO

O presente trabalho de pesquisa tem como objetivo o desenvolvimento de um protótipo de um Pregão Eletrônico Web para a UNESCO, baseando-se no conceito de *e-procurement*, aplicando a utilização de uma técnica das mais atuais no que se refere a desenvolvimento Web, o AJAX. Para tanto, o mesmo está disposto em seis capítulos, os quais incluem a introdução, o trabalho e a conclusão.

O Capítulo 2 identifica as principais tecnologias de programação Web que compõem o AJAX e faz uma abordagem de como basicamente elas funcionam. O Capítulo 3 tem por objetivo explanar os princípios do AJAX e como as aplicação Web

comportam-se ao agregar essa técnica no seu desenvolvimento. O capítulo seguinte apresenta as diretivas de relacionamento entre empresas (*Business-to-Business*) focando o *e-business* e principalmente os conceitos relacionados ao *e-procurement*.

O Capítulo 6 e os apêndices do trabalho, explanam o caso de uso na UNESCO e a proposta de informatização do processo de cotação, para que se faça uso da Web, por meio de um sistema Internet, utilizando um sistema que agregue novas concepções de desenvolvimento, que aprimorem os conceitos de interatividade sobre a Web.

2 TECNOLOGIAS DE PROGRAMAÇÃO WEB

Em 1990 quando a primeira proposta para a *Word Wide Web* foi lançada, a idéia era bastante simples: criar um emaranhado de informações interconectadas utilizando hipertexto e *Uniform Resource Identifier* (URI). Todavia não se mensurou que o fato de ter informações, quase que imediatas, para referência no mundo acadêmico, alavancaria a proposta potencializando o aperfeiçoamento da mesma (ZAKAS; MCPEAK; FAWCETT, 2006).

No decorrer de vários anos de evolução da Web, muitos conceitos tecnológicos surgiram para suprir necessidades específicas, que cada vez mais são almeçadas para que se possibilite um maior aproveitamento da Internet. Em contrapartida, a técnica de programação AJAX é um conceito que agrega uma série de tecnologias já existentes, trabalhando em conjunto, cada uma fazendo sua parte, para oferecer novas funcionalidades, que vão além de páginas Web dinâmicas que não possibilitam maior interatividade ao usuário.

A seguir são explanadas essas tecnologias utilizadas pelo AJAX, entretanto não serão feitas considerações mais profundas sobre as mesmas, pelo fato de não ser o objetivo principal deste trabalho.

2.1 HTML

O *HyperText Markup Language* (HTML) é uma linguagem de programação que permite apresentar informações de forma gráfica na Internet. Esta tecnologia utiliza-se de marcações, também chamadas de *tags*, sobre o texto para informar ao *browser*, navegador de Internet, como cada elemento deve ser manipulado e apresentado na tela (GRAHAM, 1998).

A seguir tem-se um exemplo de código fonte HTML que demonstra como o navegador interpreta estas marcações:

```
A palavra a seguir esta em <strong>negrito</strong>.
```

O resultado produzido deverá ser o seguinte:

```
A palavra a seguir esta em negrito.
```

São vários os atributos possíveis a serem utilizados em uma página HTML para representação gráfica e visualização de elementos, entretanto os mesmos não serão abordados. Estes atributos são facilmente encontrados na Internet ou em literaturas do gênero.

2.2 CSS

O *Cascading Style Sheets* (CSS) é uma diretiva bem estabelecida do Web design sendo uma das tecnologias mais importantes para o AJAX. Uma folha de estilo (CSS) oferece uma maneira centralizada de definir todas as categorias de estilos visuais que farão parte de uma página e aplicá-las concisamente em seus elementos. Além de feitos básicos de estilização como cores, bordas, imagens de fundo e tamanho, as folhas de estilo podem definir a disposição dos elementos em relação um ao outro na janela do navegador (CRANE; PASCARELLO; JAMES, 2006).

Sem a utilização de CSS para formatação da página Web, se faz necessário a inserção de código HTML a cada atributo que se deseje, tornando assim o código muito maior e visualmente de difícil leitura. Utilizando-se das propriedades do CSS, é possível definir uma formatação para cada tipo de atributo em um arquivo separado, por exemplo, deixando desta forma o código da página padronizado e claro.

Em aplicações Web clássicas, as folhas de estilos provêm um meio útil de definição de um estilo em um único local que pode ser reutilizado por muitas páginas.

Com AJAX, não se pensa em termos da alteração rápida do visual de páginas, mas as folhas de estilos oferecem um repositório prático de visuais predefinidos que podem ser aplicados aos elementos dinamicamente fazendo-se uso do mínimo de código, aplicando desta maneira conceitos relacionados a padrões de desenvolvimento Web (*Webstands*) (GONÇALVES, 2006).

2.3 JAVASCRIPT

A linguagem *JavaScript* foi desenvolvida pela *Netscape Navigation* e lançada na versão 2.0 de seu navegador em 1995, possibilitando uma alternativa aos desenvolvedores de alterar a forma de como as páginas Web sempre interagem com os usuários. A proposta básica do *JavaScript* era que não se fizesse constantes requisições ao servidor para realizar tarefas simples, como validação de dados, exibição de mensagens na tela, cálculos diversos, entre outras tarefas. A responsabilidade de executar estas tarefas seria repassada ao navegador do usuário (ZAKAS, 2006).

Sendo o *JavaScript* executado pelo navegador do usuário que está acessando a página Web, ele depende estritamente do navegador para que funcione adequadamente. Entretanto alguns navegadores não seguem a padronização da *World Wide Web Consortium (W3C)* e isso leva um código *JavaScript* a ser muitas vezes incompatível entre os vários navegadores existentes (KOCH, 2006).

O papel central no conjunto de ferramentas do Ajax é sem dúvida desempenhado pelo *JavaScript*. Um aplicativo Ajax carrega um cliente completo na memória, combinando dados, apresentação e a lógica do programa; o *JavaScript* é a ferramenta utilizada para implementar essa lógica. O *JavaScript* é uma linguagem de programação de uso geral de descendência híbrida, com uma leve semelhança com a família das linguagens C (CRANE; PASCARELLO; JAMES, 2006, p. 23).

A linguagem *JavaScript* possui uma codificação flexível, é interpretada e fracamente tipada. Fracamente tipada significa que as variáveis não são declaradas

especificamente como *strings* ou inteiros e que é possível fazer atribuições de diferentes tipos de valores à mesma variável (CRANE; PASCARELLO; JAMES, 2006).

Por exemplo, o código a seguir é totalmente válido para um *script JavaScript*:

```
<SCRIPT LANGUAGE="JavaScript">  
    var x = 123456;  
    x = 'JavaScript';  
</SCRIPT>
```

Ser uma linguagem de finalidade geral significa ser apropriada para uso com a maioria dos algoritmos e tarefas de programação. O núcleo da linguagem *JavaScript* contém suporte para números, *strings*, datas e horas, vetores expressões regulares para processamento de texto, em funções matemáticas tal como trigonometria e geração de número aleatório. Essas funcionalidades são muito abrangentes possibilitando definir objetos estruturados, exigindo assim, princípios e regras de projeto para o uso em códigos mais complexos.

Na próxima seção será apresentado um código, utilizando-se de um conjunto de tecnologias, no qual ficará mais claro o papel do *JavaScript* e como o mesmo pode funcionar dentro de uma página HTML.

2.4 DOM

O *Document Object Model* (DOM) é uma especificação que define uma interface e uma plataforma neutra que irão permitir que programas e *scripts* preencham e alterem o conteúdo, a estrutura e o estilo dos documentos, de uma forma normalizada, dando condições para uma maior portabilidade de programas e *scripts*.

O DOM possibilita que uma página Web seja manipulada representando um conjunto de nós, permitindo acesso direto a qualquer elemento da página, por meio de um *JavaScript*. O DOM apresenta os documento HTML como uma hierarquia de nós

que podem implementar interfaces especializadas. Por exemplo, o elemento <TABLE> do HTML está hierarquicamente acima dos elementos <TR>, os quais representam as linhas dentro da tabela, estes por sua vez, estando acima dos elementos <TD> de sua estrutura, que representam as colunas dentro de uma linha (SOARES, 2006).

Esta habilidade do DOM é particularmente útil para se ter a disposição ao escrever uma aplicação AJAX. Em uma aplicação Web clássica a atualização é regularmente à página inteira com novos fluxos de texto HTML, e redefinições de grande parte da interface através destes novos fluxos. Em uma aplicação AJAX, a maioria das alterações na interface do usuário será feita usando o DOM. Os elementos HTML em uma página Web estão organizados em uma estrutura de árvore. A raiz desta árvore é o elemento <HTML>, o qual representa o próprio documento (GONÇALVES, 2006).

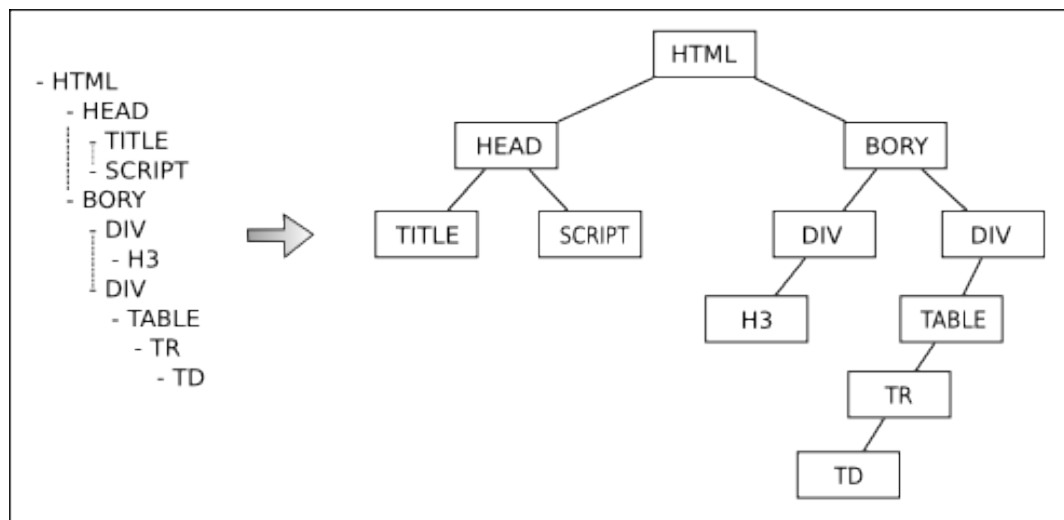


Figura 1. Hierarquia DOM em uma página HTML.

Com o DOM pode-se adicionar qualquer elemento em uma página, definindo seu conteúdo e os seus atributos. Assim, é possível colocar novos parágrafos, itens em listas, e até mesmo criar uma página inteira somente utilizando DOM. O relacionamento entre elementos no DOM deve refletir a mesma relação entre os elementos HTML, e vice versa. Modificar o DOM alterará as marcações HTML e conseqüentemente a apresentação da página (GONÇALVES, 2006).

A Figura 2 mostra um código fonte HTML que inclui noções de CSS, para estilizar a página, juntamente com *JavaScript*, para implementar um função que utiliza DOM para incluir dinamicamente novos itens em uma lista.

```

<html>
<head>
  <title>Exemplo da utilização da API DOM</title>

  <!--CSS-->
  <style>
    P {
      color: black;
    }
    .item {
      background-color: silver;
      color: black;
      font-weight: bold;
    }
  </style>

  <!--JavaScript-->
  <script language="JavaScript">
    var contador = 0;

    function adicionar() {
      contador++;
      var e = document.getElementById('lista');
      var no = document.createElement('DIV');
      var t = document.createTextNode('Item ' + contador + ' da Lista');
      e.appendChild(no);
      no.appendChild(t);
      no.className = 'item';
    }
  </script>
</head>

<body>
  <p>Exemplo da utilização da API DOM</p>
  <div id="lista"></div>
  <a href="javascript:adicionar();">Incluir novo item</a>
</body>
</html>

```

Figura 2. Exemplo da utilização da API DOM.

A cada clique no *link* “Incluir novo item” a função *adicionar()* insere no nó “lista” um novo nó, elemento do tipo `div`, e altera sua classe para “item”. Desta maneira a lista de itens reflete, na página HTML resultante, exatamente a quantidade de cliques que o *link* “Incluir novo item” receber.

Após quatro cliques, tem-se a seguinte página Web como mostra a Figura 3:



Figura 3. Resultado final da página de exemplo DOM.

As funcionalidades da API DOM em um conjunto com *JavaScript* não limitam-se somente a isso, existem diversas propriedades do DOM que possibilitam ainda um infinidade de possibilidades de interações e manipulação de elementos dispostos em uma página HTML, por meio dos seus nós. Os nós são a base para que o DOM possa navegar e conhecer a estrutura da página, partindo deste princípio, pode-se usar o DOM para manipular páginas Web como se fossem documentos XML (SOARES, 2006).

2.5 XML

O eXtensible Markup Language (XML) que em português significa Linguagem de Marcação Extensível, é uma tecnologia utilizada não apenas em programação, mas também em toda a área de Tecnologia de Informação (BORBA, 2006).

Apesar de haver semelhança, o XML não é uma variante do HTML, nem mesmo foi originado a partir do mesmo. O HTML é uma linguagem para estruturar conteúdos levando em consideração a lógica de apresentação. Ao invés disso, o XML preocupa-se somente em armazenar dados de forma estruturada. XML permite definir

um conjunto de elementos ou *tags* ilimitado para definir um padrão de documento, ao contrário do HTML, onde a variedade de *tags* é limitada. A tecnologia XML pode ser vista como uma metalinguagem, o que significa que é possível criar outras linguagens a partir dela (BORBA, 2006).

Quando é necessário que se haja troca de informações entre elementos, é preciso estabelecer uma linguagem de comunicação por meio de regras. O XML tem o propósito de definir e ser a forma de manipulação dessas regras, utilizando-se de *tags*, que serão usadas para que os elementos se comuniquem de uma forma simples. A estruturação de dados usando o XML nas aplicações AJAX, tem o papel de fazer a com que a parte cliente, neste caso o navegador do usuário, e o servidor Web, consigam trocar informações por meio de regras de comunicação pré-estabelecidas (BORBA, 2006).

A Figura 4 exemplifica um arquivo XML simples:

```
<?xml version="1.0" encoding="UTF-8"?>
<filme>
  <filme cdfilme="1">
    <titulo>Eu no Mundo</titulo>
    <genero>Aventura</genero>
  </filme>
  <filme cdfilme="2">
    <titulo>Eu em Marte</titulo>
    <genero>Ficção</genero>
  </filme>
</filme>
```

Figura 4. Exemplo de documento XML

O documento XML acima descreve uma lista com dois filmes. Para cada filme existe um elemento ou *tag* definido que corresponde a um dado sobre o filme. No exemplo, “titulo” é um nome atribuído a um campo no banco de dados, o mesmo acontece com “genero” e o atributo “cdfilme” da *tag* <filme>. Quando um documento deste gênero é recebido de um servidor Web, estes elementos podem perfeitamente serem interpretados possibilitando transformar este documento em uma página Web

facilmente ou simplesmente usar os dados para preencher uma página HTML já existente utilizando AJAX (BORBA, 2006).

2.5.1 XSL

O *eXtensible Stylesheet Language* (XSL) é uma linguagem para definir a apresentação de um documento XML. A definição do estilo é compatível com o CSS, embora use a sintaxe XML. O XSL engloba a mesma finalidade do CSS, porém sendo mais flexível, pelo fato de um documento XML não ter um número limitado de *tags*, podendo-se definir qualquer elemento (BORBA, 2006).

2.5.2 XSLT

O *eXtensible Stylesheet Language Transformation* (XSLT) é uma linguagem baseada em XML cuja principal função é transformar um documento XML em um outro documento.

Por meio do XSLT pode-se utilizar um documento XML dinâmico e um documento XSL, geralmente estático, combinando-os para produzir um resultado final o qual frequentemente é associado a um documento HTML ou XHTML (CRANE; PASCARELLO; JAMES, 2006).

2.6 JSON

O *JavaScript Object Notation* (JSON) é um formato de dados construído em *JavaScript* em alternativa ao XML. Apesar do XML ser praticamente um padrão de interoperabilidade entre sistemas e em servidores Web para transmissão de dados, alguns consideram o mesmo demasiadamente complexo somente para transmissão de

dados, necessitando enviar mais *bytes* pela Internet do que realmente haveria necessidade (GONÇALVES, 2006).

O JSON surge com o objetivo de ser um formato de dados muito leve baseado em um subconjunto da sintaxe *JavaScript*, ou seja, *arrays* e objetos literais.

```
// Array literal
var nome = ["Diego", "João", "José"];

// Objeto literal
var pessoa = {"nome": "Diego", "idade": 22, "sexo": "M"};
```

Como o JSON usa sintaxe *JavaScript*, pode-se incluir definições JSON em códigos *JavaScript*, podendo-se acessá-las sem análise gramatical extra por linguagens baseadas em XML (GONÇALVES, 2006).

Formato XML	Formato JSON
<pre><departamentos> <departamento> <codigo>1</codigo> <nome>Informática</nome> </departamento> </departamentos></pre>	<pre>"departamentos" : [{ "codigo" : 1, "nome" : "Informática" }]</pre>
102 bytes	51 bytes

Figura 5. Comparação entre os formatos XML e JSON
Fonte: GONÇALVES, E. (2007)

Conforme pode-se observar na Figura 5, devido a sua sintaxe simplificada, o JSON consegue ser consideravelmente menor comparado ao XML, o que lhe traz vantagens levando em conta o tamanho dos arquivos gerados e conseqüentemente na velocidade de sua transmissão (GONÇALVES, 2006).

2.7 XMLHTTPREQUEST

O *XMLHttpRequest* é o principal fundamento do AJAX. Ele é responsável por possibilitar que os navegadores dos usuários consigam fazer requisições assíncronas

ao servidor Web, de forma transparente. Isto significa que um objeto *XMLHttpRequest* pode servir de intermédio, oferecendo o suporte, para receber ou enviar dados para o servidor, assincronamente, sem precisar carregar toda a página Web a cada requisição (CRANE; PASCARELLO; JAMES, 2006).

A implementação de um objeto *XMLHttpRequest* não é um padrão regido pela W3C. Cada navegador é responsável pela adequação do mesmo em seu ambiente. Isto significa que nem todos os navegadores possuem a mesma forma de implementação do objeto. Na Figura 6 será apresentado uma função *JavaScript* que tem a finalidade de criar um objeto *XMLHttpRequest* com sucesso na maiorias dos navegadores Web.

```
//Cria e retorna um objeto XMLHttpRequest
function makeObjRequest(){

    var requestObj;

    if (window.XMLHttpRequest){ // se Firefox, Safari, Opera, ...
        requestObj = new XMLHttpRequest()
        try {
            if (xmlhttp.overrideMimeType)
                xmlhttp.overrideMimeType('text/xml');
        }catch (e){}
    }else if (window.ActiveXObject){ // se Internet Explorer
        try {
            requestObj = new ActiveXObject("Msxml2.XMLHTTP")
        }catch (e){
            try {
                requestObj = new ActiveXObject("Microsoft.XMLHTTP")
            }catch (e){}
        }
    }
    if (!requestObj) {
        alert('Não foi possível criar uma instância XMLHttpRequest');
        return false;
    }else{
        return(requestObj);
    }
}
```

Figura 6. Instanciando um objeto *XMLHttpRequest*

2.8 PHP

O *PHP: Hypertext PreProcessor*, sendo seu acrônimo recursivo PHP, é uma linguagem de programação Web interpretada com a finalidade de gerar páginas HTML

dinamicamente. Todo código PHP é interpretado pelo servidor que, então, retorna ao cliente a página resultante do processamento. Desta forma o PHP consegue interagir com outros serviços disponíveis no servidor, como por exemplo, um banco de dados, possibilitando assim que se possa consultar dados e apresentá-los em páginas HTML de forma simples (ANSELMO, 2000).

O PHP não é uma tecnologia base para o funcionamento do AJAX, qualquer linguagem que consiga gerar páginas HTML dinamicamente pode substituí-la, como por exemplo, Java, Python ou ASP. Entretanto esta linguagem, além de ser livre e gratuita, é uma tecnologia bem difundida no âmbito de programação para Internet. Assim sendo a mesma provém uma considerável quantidade de funções e bibliotecas para interações com outras tecnologias.

2.8.1 Framework *CodeIgniter* para desenvolvimento em PHP

O *CodeIgniter* é um *framework* MVC para desenvolvimento de aplicações escritas em PHP, que contém uma considerável quantidade de bibliotecas para implementações cotidianas de código, além de também abstrair a camada de comunicação com o banco de dado. O *CodeIgniter* oferece uma estrutura lógica de acesso as suas bibliotecas utilizando camadas de modelo, visão e controle (MVC) e tem como objetivo minimizar a quantidade de código para se implementar um projeto de software (*CodeIgniter*, 2007).

2.9 MYSQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Structured Query Language - Linguagem Estruturada para Pesquisas), a qual é o padrão utilizado pela maioria dos bancos de dados relacionais. O

MySQL se tornou um banco de dados popular, para utilização na Internet, devido sua robustez, rapidez e concepções de multi-tarefa e multi-usuário. Além disto, o MySQL é multiplataforma e contém bibliotecas para interação com várias linguagens de programação como JAVA, C/C++, Python, Perl, PHP e Ruby (MYSQL AB, 2007).

Dentre as características citadas acima o MySQL também se destaca por ser um software livre, que pode ser utilizado em sistemas de produção com alta carga e missão crítica bem como pode ser embutido em programa de uso em massa.

3 TÉCNICA AJAX

Neste capítulo será abordado em algumas seções as principais diretrizes desta técnica que pode ser uma alternativa à interatividade em páginas Web. Estas poderão explicar os objetivos e esclarecer alguns pontos desta nova técnica de desenvolvimento para a Internet, que tem por finalidade tornar a Web mais interativa e dinâmica.

3.1 REPENSANDO AS APLICAÇÕES WEB

Ao longo de vários anos de evolução de Internet, pode-se notar uma tendência do mercado, a qual se aplica em uma projeção para que aplicativos funcionem de forma parcial ou totalmente no ambiente Web. Esta nova concepção leva os desenvolvedores a explorarem novos horizontes, a fim de tornar, as até então conhecidas páginas Web, em aplicações mais ricas em recursos e funcionalidades, que possam oferecer uma melhor experiência de usabilidade e flexibilidade para os usuários (BORBA, 2006).

As aplicações tradicionais contêm uma vasta quantidade de recursos visuais e são repletas de funcionalidades. Por exemplo, em um editor de planilha de cálculo, o qual observar-se que ao digitar uma fórmula matemática, pode-se interagir com a planilha de várias maneiras. Pode-se editar dados, navegar pelos dados com o teclado ou *mouse*, reorganizar dados utilizados com recursos de arrastar e soltar e que uma série de itens fazem deste editor uma aplicação com uma interface rica em recursos (CRANE; PASCARELLO; JAMES, 2006).

O *Asynchronous Javascript And Xml* (AJAX) tem como proposta básica, oferecer a uma página ou aplicação Web funcionalidades próximas ou idênticas ao de

uma aplicação tradicional. Desta forma os usuários poderão usufruir de aplicações Web com interfaces mais ricas e com maior interatividade .

3.2 CONHECENDO O AJAX

Ao contrário do que se possa muitas vezes imaginar, o AJAX não é nenhuma nova tecnologia. AJAX na realidade é uma técnica que surgiu como uma alternativa, que utiliza tecnologias já existentes, as quais foram citadas no capítulo anterior, cada uma fazendo sua parte, para que, em conjunto, possibilitem que a Web seja mais dinâmica, quebrando o paradigma de páginas estáticas e de baixa interação com o usuário (CRANE; PASCARELLO; JAMES, 2006).

Uma das principais funcionalidade do AJAX é a requisição assíncrona ao servidor Web. Desta maneira é possível realizar consultas sem que uma página seja novamente carregada inteiramente, evitando tráfego desnecessário de informações não relevantes à ação que se esteja executando no momento. Contudo, esta funcionalidade não se limita somente a redução de tráfego, e sim, também a uma considerável gama de possibilidades de interações que o assincronismo de requisições possibilita em um aplicativo baseado sobre a Web.

3.3 OS PRINCÍPIOS DO AJAX

O modelo clássico de aplicação baseado em páginas é estruturado embasando-se na forma de pensar do ser humano. Segundo Crane, Pascarello e James (2006), um aplicativo Web AJAX é projetado com um conceito de quatro suposições:

- a) O navegador hospeda uma aplicação, não o conteúdo;
- b) o servidor fornece dados, não conteúdo;
- c) a interação do usuário com o aplicativo pode ser fluente e contínua;

d) isso é um programa e requer disciplina.

Baseando-se nestas suposições é possível delimitar que o AJAX é utilizado em aplicações com forte integração com o navegador, sendo este o recipiente básico destas aplicações na arquitetura Web.

3.3.1 O Navegador Hospeda Uma Aplicação, não o Conteúdo

Nas aplicações Web clássicas baseadas em páginas, o navegador não possui nenhum controle sobre o que o usuário está realmente realizando. Simplesmente a cada interação do usuário o servidor envia uma nova página para o navegador, mesmo que a alteração seja mínima, e o mesmo exibe-a na tela.

Todas as informações nesse modelo de aplicação são mantidas no servidor e o navegador obedientemente descarta o documento exibido anteriormente pelo novo, a cada requisição, conforme mostra a Figura 7 (CRANE; PASCARELLO; JAMES, 2006):

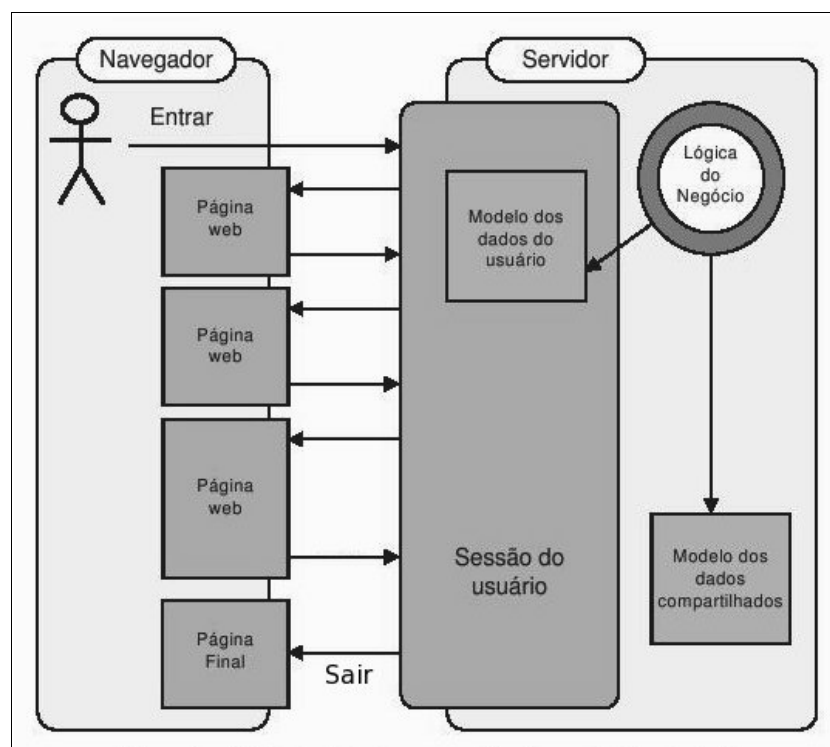


Figura 7. Lógica de aplicativo Web clássico.
Fonte: CRANE, D.; PASCARELLO, E.; JAMES, D. (2006)

Em uma aplicação Web AJAX move-se parte da lógica do aplicativo para o navegador. Essa lógica normalmente encontra-se em código *JavaScript* e é entregue ao navegador no primeiro acesso realizado ao servidor pelo usuário. Esse documento permanecerá com o usuário por toda a sua sessão e será capaz de decidir se deve tratar ele mesmo os dados inseridos pelo usuário, passar uma solicitação para o servidor Web, que tem acesso ao banco de dados e outros recursos ou fazer uma combinação de ambos (CRANE; PASCARELLO; JAMES, 2006).

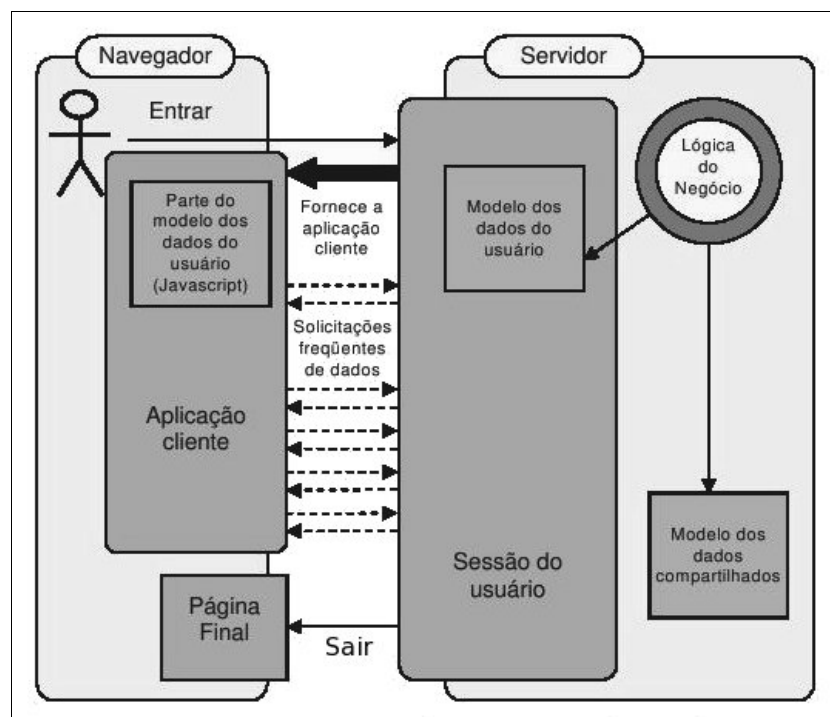


Figura 8. Lógica de aplicação AJAX.

Fonte: CRANE, D.; PASCARELLO, E.; JAMES, D. (2006)

A Figura 8 demonstra a lógica de uma aplicação Web utilizando AJAX, no qual a interação com o servidor é frequentemente feita por meio da aplicação *JavaScript*, hospedada no navegador do usuário durante a permanência de sua sessão com o servidor Web. Esta aplicação controla o fluxo de interações com o usuário e é responsável por manipular as transações e exibir seus resultados na tela.

3.3.2 O Servidor Fornece Dados, não Conteúdo

Como observado, as aplicações Web clássicas retransmitem, a cada requisição, a mesma mistura de cabeçalhos, conteúdo e dados. Na hipótese de se ter uma aplicação de vendas e se queira adicionar um item à lista compras, tudo que realmente se precisa fazer é informar o novo valor atualizado ou notificar algum erro no processo. Desta maneira, em uma aplicação Web clássica, há muito tráfego que necessariamente não seria relevante a essa ação, tendo em vista que uma nova página é carregada por inteiro (Figura 9) (CRANE; PASCARELLO; JAMES, 2006).

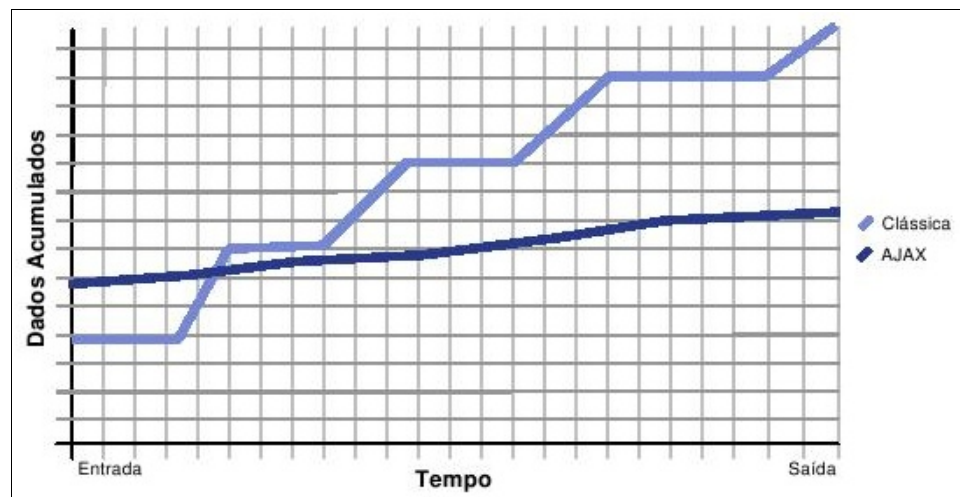


Figura 9. Comparação de tráfego de dados.

Fonte: CRANE, D.; PASCARELLO, E.; JAMES, D.(2006)

Na mesma situação descrita usando um aplicativo Web AJAX, pode-se utilizar requisições assíncronas ao servidor para tornar a transferência de dados mais eficiente. O cabeçalho, alguns conteúdos e outras características referentes ao *lay-out* da página, já foram anteriormente carregados, portanto o servidor precisa somente enviar os dados relevantes.

O aplicativo Web AJAX poderia fazer isso de várias formas, como retornar um fragmento de *JavaScript*, um simples fluxo de texto ou um pequeno documento XML, que seria manipulado e exibido na página (CRANE; PASCARELLO; JAMES, 2006).

3.3.3 A Interação do Usuário com o Aplicativo pode ser Fluente e Contínua

Ao se enviar qualquer dado por meio de um formulário ao servidor ou ao acessar um *hyperlink* numa aplicação Web clássica, geralmente o usuário permanece efetivamente aguardando por uma resposta do servidor. Entretanto com o uso do AJAX, caso a aplicação do lado cliente seja robusta o suficiente, esse processo poderá ser gerenciado facilmente pela aplicação e o usuário poderá executar várias requisições simultâneas e continuar com sua tarefa normalmente.

Uma outra questão que pode ser abordada é a possibilidade de associação de eventos a um aspecto amplo de ações de usuários. Desta maneira, conceitos mais sofisticados, como arrastar e soltar, tornam-se mais simples de serem implementados com o uso do AJAX, trazendo a experiência do usuário para algo mais próximo dos recursos de aplicativos *desktop* (CRANE; PASCARELLO; JAMES, 2006).

3.3.4 A Codificação de um Aplicativo AJAX Requer Disciplina

As clássicas aplicações Web utilizam *JavaScript* apenas para controles de preenchimento de formulários e outros recursos básicos e geralmente não permanecem por muito tempo na sessão do usuário.

Codificar um aplicação completa em AJAX é uma gestão mais complexa. O código deve ser executado do início ao fim da sessão do usuário sem interrupção, sem tornar-se lento e sem gerar consumo indevido de memória. Para cumprir estes objetivos, é preciso escrever código de fácil manutenção e de alto desempenho, utilizando fundamentos de estruturação de código e projeto (CRANE; PASCARELLO; JAMES, 2006).

3.4 AJAX NA PRÁTICA

A técnica AJAX pode ser utilizada em diversas situações em uma aplicação Web que requeiram algum tipo de interação com o usuário. Uma situação comum e simples, onde se utiliza AJAX é em formulários, no qual existem campos de seleção, em que, conforme a opção escolhida no primeiro campo, uma requisição é feita para carregar informações no próximo campo baseadas na seleção anterior.

Pode-se imaginar nesse exemplo um formulário de busca de cidades por estados. Desta forma, a cada estado que se selecione, o campo de cidades é preenchido somente com cidades referentes àquele estado sem que a página necessite ser carregada novamente por completo (Figura 10).

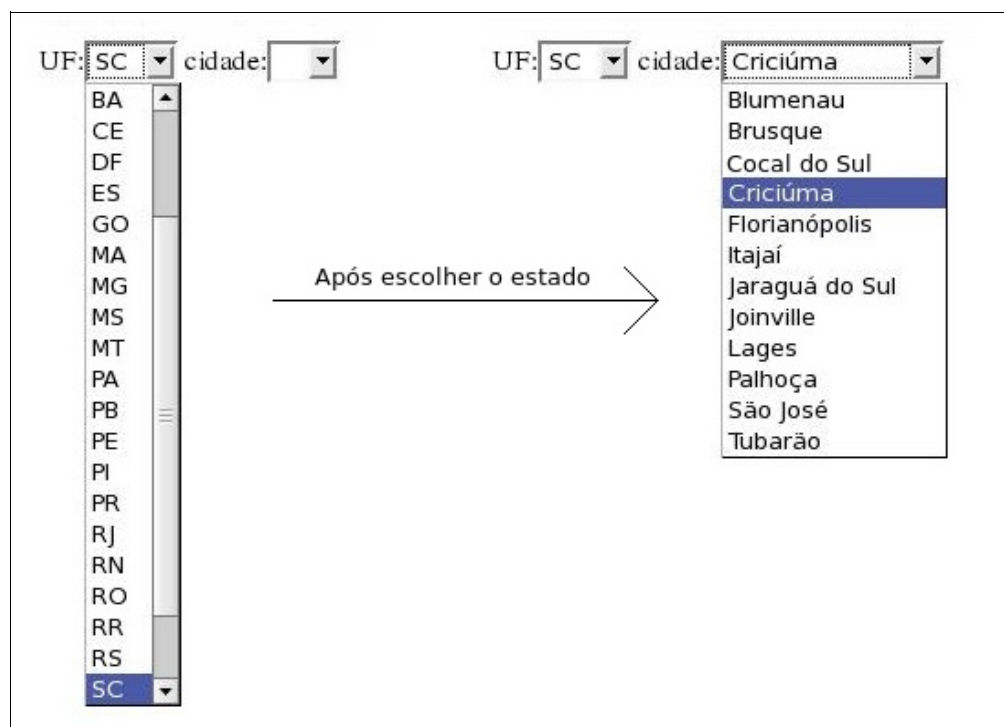


Figura 10. Seleção de cidade por estados utilizando AJAX.

No exemplo da Figura 10 um função *JavaScript* é acionada, pelo evento *onChange*, quando existe um mudança no campo que contém a lista de estados do Brasil. Após ser feito um seleção de estado, esta função instancia um objeto *XmlHttpRequest*, que por sua vez, com o parâmetro obtido na seleção, faz uma

requisição ao servidor Web, solicitando as cidades que estão relacionadas àquele estado. O servidor utiliza um código PHP para acessar o banco de dados MySQL e a partir da consulta, gera um documento XML e retorna o mesmo para a aplicação cliente, por meio do objeto *XmlHttpRequest*.

Este documento XML contém somente dados estruturados referentes as cidades como, código e nome. Assim sendo, por meio do DOM, o código *JavaScript* lê as entradas do documento XML e atribui cada uma delas ao campo do formulário que corresponde as cidades. Desta forma o preenchimento das cidades pode ser feito de maneira ágil e transparente.

É importante ressaltar que não se utilizou CSS no exemplo da Figura 10, entretanto, nota-se que não há nenhum tipo de estilo ou formatação para a página de exemplo simplória. Caso houvesse um estilo CSS atribuído ao documento, o que poderá fazer mais sentido em uma aplicação real, o mesmo seria automaticamente aplicado assim que a página fosse carregada, acontecendo o mesmo com os elementos que seriam adicionados posteriormente pela utilização da técnica AJAX.

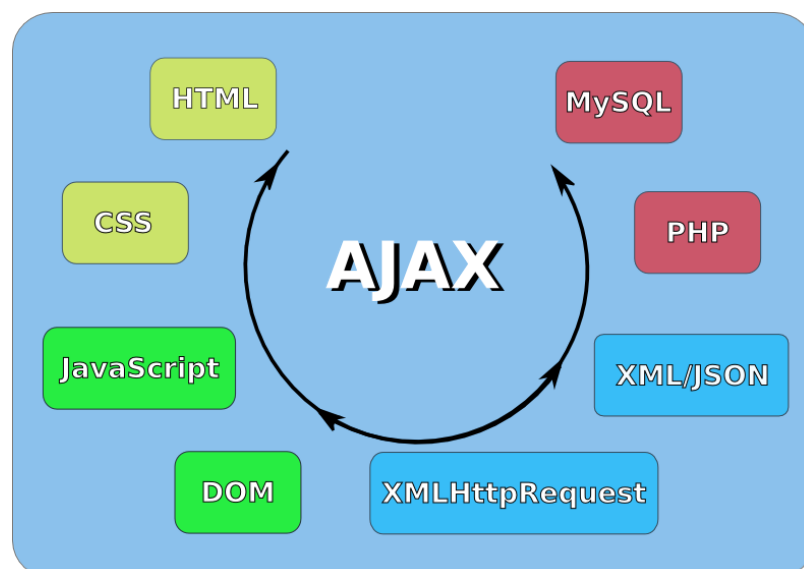


Figura 11. Fluxo de interações entre as tecnologias que compõem o AJAX.

A Figura 11 ilustra o fluxo das interações entre as tecnologias componentes do AJAX, do início ao fim de uma requisição assíncrona.

3.5 QUANDO SE DEVE UTILIZAR O AJAX

Existe ocasiões as quais a utilização do AJAX nem sempre será a melhor solução. O AJAX pode ser utilizado de várias maneiras em aplicações na Internet, podendo-se construir aplicações convencionais totalmente em AJAX ou aplicações híbridas, que utilizarão esta técnica somente em funções específicas do Website.

Uma aplicação completa em AJAX terá que carregar um código *JavaScript* que permanecerá na sessão do usuário até que o mesmo saia do sistema. Esta aplicação geralmente se assemelha ao máximo com as aplicações *desktop* convencionais e portando deverá ser projetada com as mesmas preocupações e funcionalidades inerentes. Desta forma, o custo de se desenvolver um sistema completo em AJAX pode se tornar de alto, comparado com o desenvolvimento baseado em páginas Web clássicas. Em contra partida, este é o investimento que poderá se transformar em um aplicativo de grande interatividade e usabilidade (CRANE; PASCARELLO; JAMES, 2006).

Em alternativa ao custo de aplicações completamente escritas em AJAX, pode-se criar aplicações híbridas, que no seu escopo geral, serão desenvolvidas no modelo clássico de páginas Web, porém usufruindo do AJAX em determinadas funções na aplicação, onde se necessite de maior interatividade e melhor usabilidade, como por exemplo em:

- a) Inserções, exclusões e atualizações de itens em uma lista;
- b) interações com formulários;
- c) navegações em árvores de grande hierarquia;
- d) comunicação rápida entre usuários;
- e) enquetes e caixas de marcação;
- f) filtro e aperfeiçoamento de manipulação de dados;
- g) autocompletar;

- h) erros interativos;
- i) operações computacionais de alto custo.

O AJAX possibilita que as operações citadas acima possam ser realizadas com rapidez e menor custo no tráfego de dados, também podendo tornar a aplicação mais atrativa aos usuários.

Em aplicações híbridas é possível que determinadas aplicações tenham uma melhor abordagem sobre seu contexto. Desta maneira, páginas na Internet não necessariamente precisam de uma interface rica fazendo-se uso do AJAX, para que as mesmas cumpram com eficiência seu papel. Uma aplicação híbrida pode reduzir tempo de implementação e o custo necessário desta aplicação comparado a uma aplicação completa em AJAX (NIEDERAUER, 2007).

Em aplicações Web que se deseje uma freqüente mudança da interface, com navegação básica, onde se substitua uma grande quantidade de texto, pode não ser uma boa alternativa a utilização do AJAX, sendo a melhor alternativa a utilização de páginas Web clássicas, tendo em vista que as vantagens da utilização do AJAX não poderão ser aplicadas com eficiência a ponto de tornar a aplicação viável.

3.6 SEGURANÇA NO AJAX

As crescentes evoluções das tecnologias em torno da Internet trouxeram muitos benefícios em geral, mas também alguns problemas, como a segurança dos dados que trafegam pela rede. A Web, por ser um ambiente que atua sobre a Internet, herda essas características e adicionar medidas de segurança adequadas a um aplicativo AJAX pode fazer um grande diferencial na aplicação. Muitos dos problemas de segurança que um aplicativo AJAX é submetido são os mesmos de um aplicativo Web clássico, porém existem diretivas sobre segurança que implicam especificamente ao uso

do AJAX, que devem ser tratadas dependendo da situação (CRANE; PASCARELLO; JAMES, 2006).

Destre estas diretivas pode-se citar:

- a) Tratamento das medidas que os navegadores Web adotam para execução de *scripts* executáveis;
- b) proteger dados confidenciais utilizando *HyperText Transfer Protocol Secure* (HTTPS) ou criptografia por meio do AJAX sobre *HyperText Transfer Protocol* (HTTP);
- c) controle de acesso a fluxo de dados do AJAX.

Um outro ponto de vista, focando o lado cliente, também deve ser levado em consideração. Ao deixar que seu navegador carregue uma aplicação AJAX, um usuário está depositando uma confiança significativa nos autores no aplicativo em questão. Este fato põe em discussão uma abordagem de como garantir que o código que se esteja executando no lado cliente pode ser considerado seguro (CRANE; PASCARELLO; JAMES, 2006).

Segundo Crane, Pascarello e James (2006) deve-se lembrar que segurança não é um estado absoluto. Nada permanece completamente seguro. O melhor que se pode fazer é estar um passo à frente de quaisquer invasores.

Segurança é um tema muito abrangente e não é o escopo deste trabalho explicar mais especificamente este assunto. Para implementações de códigos e/ou maiores informações sobre segurança no AJAX, consulte as referências do trabalho.

3.7 BIBLIOTECAS *JAVASCRIPT* COM SUPORTE AO AJAX

Em prol da produtividade do desenvolvimento Web aliada a um considerável número de características comuns agregadas aos sites, focando a interatividade com o usuário, pessoas começaram a implementar bibliotecas *JavaScript*

e disponibilizá-las na Internet. Algumas dessas iniciativas atualmente são capazes de auxiliar na interatividade com o usuário, que vão desde simples verificações de formulários, o que é comum no desenvolvimento Web, até sofisticados sistemas de *autocomplete* (GONÇALVES, 2006).

Dentre as bibliotecas mais reconhecidas que dão suporte a requisições assíncronas (AJAX), pode-se citar:

- a) *JQuery*
- b) *Prototype e Script.aculo.us*
- c) *Dojo Toolkit*
- d) *Yahoo! UI*
- e) *Ext*

Cada biblioteca tem sua peculiaridade e forma de implementação, entretanto todas focam a interatividade na Web como objetivo principal. *JQuery* foi a biblioteca escolhida para construção deste trabalho, pois apresenta-se como uma biblioteca leve, de simples implementação, com uma série de *plugins* disponíveis, e que se pode manipular facilmente a estrutura de documento HTML/XML por meio da API DOM, suprimindo os requisitos para o desenvolvimento da aplicação experimental proposta (GONÇALVES, 2007).

A biblioteca *JQuery* provê uma relevante quantidade de recursos que podem manipular o DOM pesquisando em sua estrutura por meio dos nós. Esses nós podem sofrer diversos tipos de ações como adição, remoção e alteração de atributos, classes, estilos, efeitos, eventos, dados, entre outros. Com a utilização do *JQuery* existe a possibilidade que se alterar qualquer característica de qualquer elemento gráfico da aplicação Web, de forma simples e muitas vezes com poucas linhas de código (JQUERY, 2007).

4 *E-BUSINESS*: CONCEITOS DE NEGÓCIOS APLICADOS A INTERNET

Neste capítulo serão abordados alguns conceitos relacionados ao mundo dos negócios, nos quais empresas interagem entre si para realização de compra e venda de bens e serviços por meio da Internet.

Com o desenvolvimento da tecnologia da informação, observa-se o surgimento de sistemas informatizados para cobrir ou suportar muitas das atividades das áreas de gestão de negócios. Estas áreas são divididas em classes dependendo do seu enfoque de atuação. Estas classes podem fazer parte de sistemas de gestão empresariais (ERP) estando diretamente ligadas à automação do processo operacional, permitindo agilização e melhores controles (TEIXEIRA, 2003).

Segundo Franco Junior (2001) dentre as diferentes áreas do *e-business*, cada uma é responsável por algum tipo de relacionamento entre entidades podendo-se citar algumas classes como:

- a) *Business to Consumer* (B2C);
- b) *Consumer to Consumer* (C2C);
- c) *Business to Business* (B2B).

O B2B é a classe responsável pelo relacionamento entre empresas para compra e venda de bens e serviços. Esta será abordada com maior ênfase na seção seguinte, para que se possa compreender melhor suas características.

4.1 *BUSINESS TO BUSINESS*

O comércio eletrônico B2B refere-se às transações entre empresas realizadas eletronicamente pela Internet, extranets, intranets ou redes privadas. Estas transações podem ser feitas entre uma empresa e sua cadeia de suprimentos ou simplesmente com uma outra empresa qualquer. Neste contexto, o termo *empresa*

indica qualquer organização, privada ou pública, com ou sem fins lucrativos. A principal característica do B2B consiste em que as empresas possam tentar automatizar os processos de transações comerciais com o intuito de melhorá-los (TURBAN; KING, 2004).

4.2 *E-PROCUREMENT*: APERFEIÇOANDO O PROCESSO DE COMPRA

O *e-procurement* pode ser relacionado com o “departamento de compras” de uma empresa tradicional e tem como conceito transferir para a Web o processo e gerenciamento de compras de suprimentos aliviando, assim, a carga de trabalho e os custos dessa área nas corporações. Este sistema baseia-se na oferta de compra e os fornecedores podem fazer suas cotação diretamente pela Internet, quando alertados por meio de um e-mail, por exemplo, sem que haja um outro tipo de contato prévio (TEIXEIRA, 2003).

O *e-procurement* traz a eliminação do papel, cotações/licitações de preços mais abrangentes e a possibilidade de acompanhar melhor a performance dos fornecedores.

Quando desenvolvidos para Web, os sistemas podem elaborar cadastros eletrônicos, onde se é possível analisar produtos e preços, indicando as melhores opções de compra de acordo com parâmetros pré-estabelecidos (TEIXEIRA, 2003).

Segundo Teixeira (2003), entre os resultados imediatos sobre a adoção de um sistema de *e-procurement* estão: o aumento efetivo dos níveis de eficiência e controle em compras, além de uma redução sensível dos custos operacionais e dos preços dos produtos comprados.

E-procurement é o termo usado para o relacionamento eletrônico entre uma empresa e fornecedores, principalmente na automatização dos processos de compras *on-line*. Esse relacionamento abrange desde pesquisas de produtos e preços até

transações propriamente ditas, passando por consultas a informações e outros serviços que podem estar associados às atividades de fornecimento. Os sistemas de *e-procurement* podem ser desenvolvidos ou implementados de várias formas, dependendo da posição que uma empresa específica se encontra dentro da cadeia produtiva e de como estão organizados seus fornecedores (TEIXEIRA, 2003).

4.2.1 Entendendo o Processo de Compras

Conforme afirma Teixeira (2003), toda empresa tem uma área de compras, que na maioria das vezes é composta por três processos básicos:

- a) **Processo de Direção e Planejamento:** define as políticas, procedimentos, os direcionamentos estratégicos de compras, o planejamento das necessidades de compras do ano, alinhado com o planejamento geral de vendas, produção e investimentos, entre outras funções;
- b) **Processo Estratégico de Compras:** esse processo avalia e prioriza as categorias de compras que serão negociadas no ano, as metas de economia e a qualidade para cada categoria selecionada. Analisa, também, o mercado fornecedor, buscando novas alternativas tanto de especificações como de novos fornecedores e definição detalhada de preços, considerando o escopo do que vai ser comprado, bem como a integração do fornecedor no processo operacional da empresa;
- c) **Processo Operacional:** envolve o envio do pedido ou a programação da entrega, o acompanhamento do transporte e entrega, recebimento, atualização de estoques e monitoramento do desempenho do fornecedor.

De uma maneira geral toda empresa, de médio ou grande porte, deveria desenvolver essas atividades em maior ou menor grau. Independentemente da existência

de tecnologia ou não, estes processos fazem parte da vida de uma área de compras e devem ser respeitados, inclusive para o uso adequado de ferramentas como a Internet (TEIXEIRA, 2003).

4.2.2 As Metas do *E-Procurement*

No gerenciamento do *procurement*, que é a coordenação de todas as atividades pertinentes à compra de bens e serviços necessários para cumprir a missão de uma empresa, perde-se muito tempo com atividades sem valor agregado, como: registrar dados, corrigir erros em documentos, acelerar a expedição ou resolver problemas de qualidade (TURBAN; KING, 2004).

O *e-procurement* vem a ser uma alternativa à automatização e simplificação das rotinas de *procurement*, assim os profissionais de compras podem dedicar-se a compras mais estratégicas, podendo atingir as seguintes metas (TURBAN; KING, 2004):

- a) Aumentar a produtividade do agente de compra;
- b) reduzir os preços de compra por meio de padronização do produto e da consolidação das compras;
- c) aperfeiçoar o fluxo e o gerenciamento de informações (informações sobre fornecedores e preços, por exemplo);
- d) minimizar as compras feitas de fornecedores não contratados (eliminando a compra não programada);
- e) melhor processo de pagamento;
- f) simplificar o processo de compra, tornando-o simples e rápido;
- g) reduzir o custo administrativo de processamento por pedido em até 90 por cento;

- h) descobrir fornecedores que forneçam bens e serviços mais rapidamente e/ou mais baratos;
- i) integrar o processo de compra empresarial como o controle orçamentário de modo eficiente e efetivo;
- j) minimizar erros humanos no processo de compra ou expedição.

Os benefícios mais palpáveis se concentram em atividades operacionais específicas, como o envio, recebimento e equalização das cotações. As atividades nobres e que agregam maior valor, como avaliação de especificações alternativas, análise da competitividade do mercado fornecedor, avaliação de outros aspectos da cotação que não apenas os preços e condução de negociações estratégicas, devem ser realizadas pelos profissionais da área de compras (TURBAN; KING, 2004).

5 TRABALHOS CORRELATOS

No decorrer dos estudos referentes a este trabalho foram analisados algumas publicações nesta área a nível nacional, estadual e regional, que a seguir estão relacionadas:

5.1 UTILIZAÇÃO DE AJAX NO DESENVOLVIMENTO DE SISTEMAS WEB

Trabalho de conclusão de curso apresentado como requisito parcial para a obtenção do grau de especialista pelo curso de Especialização em Web e Sistemas de Informática da Universidade Federal do Rio Grande do Sul - Instituto de Informática, publicado em Novembro de 2006.

O trabalho tem por objetivo fazer um estudo sobre o desenvolvimento de aplicações Web utilizando AJAX, abordando as tecnologias envolvidas neste desenvolvimento e a maneira como são integradas. Cita também o futuro do AJAX e é mostrado um estudo de caso de uma aplicação que utilize deste recurso (LIMEIRA, 2006).

5.2 APLICAÇÕES WEB ENRIQUECIDAS COM AJAX

Monografia intitulada como “Estudo da Concepção, Definição e Projeto de Aplicações Web Enriquecidas com AJAX”, apresentada como requisito parcial para conclusão do curso de graduação em Licenciatura Plena em Informática da Universidade Federal de Mato Grosso – Departamento de Matemática, publicada em 2006.

Neste trabalho é realizado em breve estudo sobre o AJAX e oferece uma concepção e definição para este termo, caracterizando-o como um técnica para

atualização assíncrona de conteúdo com tecnologias já existentes; e como um novo modelo no projeto de aplicação para Web. Após a análise da técnica é proposta uma abordagem da concepção de padrão de projeto *Model-view-controller* (MVC), para o desenvolvimento de aplicações AJAX, além de relatar locais onde se deveria utilizar, ou não, esta metodologia e problemas comumente ocorridos (GONÇALVES, 2006).

5.3 AJAX APLICADO A UM SISTEMA DE PLANO DE NEGÓCIOS

Trabalho intitulado como “Estudo da Metodologia de Programação AJAX Aplicada a um Sistema de Plano de Negócios Gratuito para a Web”, apresentado como requisito parcial para obtenção do grau de Bacharelado em Sistemas de Informação da Sociedade Educacional de Santa Catarina – SOCIESC - Joinville, publicado em 2006.

Este trabalho mostra o AJAX como uma possível solução para uma maior interatividade na Web e redução de tráfego entre estações e servidores. Nele são descritas todas as tecnologias que compõem esta metodologia e também um sistema de plano de negócios é apresentado usufruindo-se da mesma (KOCH, 2006).

5.4 MODELAGEM DE UM SISTEMA *E-PROCUREMENT*

Trabalho de conclusão de curso intitulado como “Modelagem de uma Plataforma de Compras Eletrônica *E-procurement* Auxiliando a Indústria Têxtil na Venda de Produtos”, apresentado como requisito parcial para obtenção do grau de Bacharelado em Sistemas de Informação da Universidade do Sul de Santa Catarina – UNISUL – Campus Grande Florianópolis, publicado em 2006.

O trabalho que segue apresenta a modelagem de um sistema para a utilização do meio digital nos processos de obtenção de pedidos. Para tanto, pode-se utilizar a linguagem de modelagem UML e o conceito de *e-procurement* que visa

melhorar a eficiência através da automatização do processo de compra. A abordagem inicial é feita explicando o significado de se modelar um software e os conceitos da linguagem UML. Em seguida são apresentados os conceitos de *e-commerce*, *business-to-business* e *e-procurement*. A modelagem do negócio e a modelagem do sistema são demonstradas pelos diversos diagramas e elementos sustentados pela linguagem UML. Neste âmbito o sistema é validado através de um *checklist* e também pela implementação (BROILO, 2006).

5.5 ANÁLISE DO ENTENDIMENTO DA UTILIZAÇÃO DO *E-PROCUREMENT*

Monografia intitulada como “Verificação do Entendimento e da Utilização do *E-procurement* em Duas Empresas Globalizadas”, apresentada como requisito parcial para obtenção do Certificado de Especialização pelo Curso de Pós Graduação MBA em Gerência Financeira e Controladoria do Departamento de Economia, Contabilidade, Administração e Secretariado – ECASE da Universidade de Taubaté, publicada em 2003.

O trabalho explana que o *e-procurement* é uma ferramenta de comércio eletrônico e pode ser definida como um conceito que transfere para a Web o processo e gerenciamento de compras tornando, assim, a carga de trabalho e os custos dessa área nas corporações problemas menos inerentes.

Apesar de revolucionária a ferramenta ainda é pouco conhecida e utilizada até mesmo em grandes corporações. Essa conclusão pode ser obtida através de uma pesquisa com pessoas ligadas as áreas de compras de duas empresas globalizadas, que foram submetidas a um questionário. Foram realizadas três rodadas tentando, assim, formar um consenso de conhecimentos entre os pesquisados (TEIXEIRA, 2003).

6 CASO DE USO DA UNESC

O processo de *procurement* da Universidade do Extremo Sul Catarinense - UNESC, atualmente funciona do modo tradicional. Realiza-se em média três cotações à empresas e então os profissionais desta área analisam as propostas com base em alguns fatores, como preço e qualidade do produto.

Nas reuniões realizadas com os profissionais das áreas de compras e finanças pode-se constatar que devido ao tamanho da instituição, este modo tradicional torna-se lento e de difícil controle e gerenciamento, demandando muito tempo, inclusive para correções de erros decorrentes dos processos manuais. Os custos operacionais, levando em consideração, as ligações telefônicas e a carga de trabalho dos profissionais do setor de compras também acabam se elevando. Desta maneira, a utilização de uma estrutura informatizada e compatível com a Web, pode tornar o gerenciamento desse processo mais eficiente e diminuir os custos.

A fim de propor uma solução para auxiliar o processo de compras da universidade, uma aplicação experimental de um Pregão Eletrônico Web, baseada em conceitos de *e-procurement*, foi desenvolvida agregando funcionalidades referentes a interatividade e usabilidade utilizando AJAX.

6.1 O ATUAL PROCESSO DE COTAÇÕES DA UNESC

A solicitação de bens ou produtos para uso e consumo iniciam-se por meio de uma requisição de compra dentro do próprio sistema interno da instituição. Vários setores da universidade tem acesso a esse módulo e podem fazer solicitações ao departamento de compras, acompanhadas de um memorando assinado pelo responsável do setor. O departamento de compras dá prosseguimento nos trâmites seguintes que

envolvem: orçamentos, aprovação e finalmente a compra propriamente dita dos itens requisitados.

As requisições de compras normalmente são abrangentes, cabendo ao departamento de compras pesquisar as várias características dos produtos requisitados, salvo casos em que a compra deve ser específica de um determinado produto, marca ou fornecedor. Por exemplo, um dado tipo de substância para uso no curso de medicina, ou, um servidor de aplicação para o setor de informática.

Nos orçamentos realizados pelo departamento de compras, as negociações de prazos e tipo de entrega, valores e forma de pagamento são feitas por meio de e-mail ou telefone. Nessas negociações são levadas em consideração as exigências de cada requisição. Muitas vezes são sugeridos os fornecedores na própria requisição, além de outras informações como quantidade, item, unidade, descrição, centro de custo, observações e data limite para entrega do material. Dependendo desses fatores o departamento de compras algumas vezes aprova a compra antes que a requisição chegue ao gerente financeiro. Estas situações acontecem geralmente em compras corriqueiras da instituição, que são feitas frequentemente. Na Figura 12 pode-se observar o diagrama de atividades do processo geral de compras da UNESC.

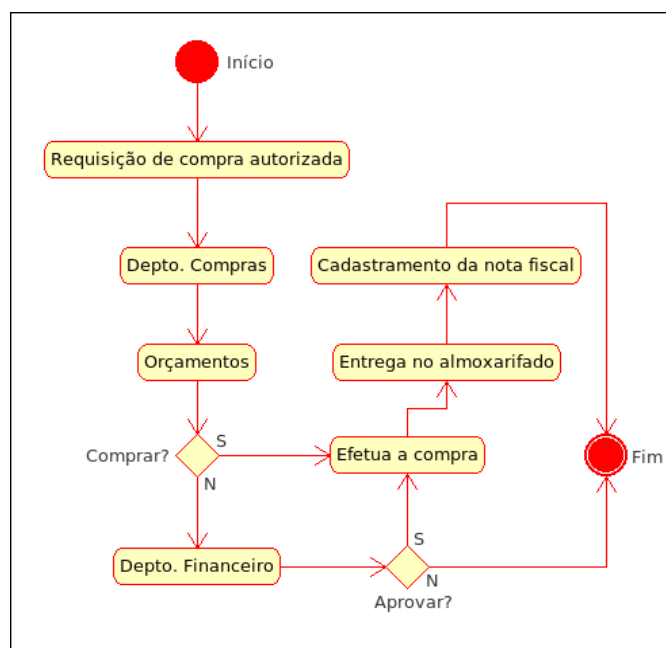


Figura 12. Diagrama de atividades do processo de compra atual da UNESC

O gerente financeiro da instituição é o responsável pela aprovação ou não das compras, ele analisa as requisições e decide quais devem ser deferidas ou não, com base nos orçamentos e informações contidas nas mesmas. Após a aprovação, a requisição volta para o departamento de compras, que por sua vez confirma a compra com o fornecedor do orçamento escolhido e aguarda que o produto seja entregue até a data estipulada.

Quando a mercadoria chega na instituição o departamento de compras recebe o documento fiscal e o cadastra, para fins de escrituração fiscal. Este processo também pode implicar no cadastramento do fornecedor, caso o mesmo não esteja, pois atualmente não é necessário que o fornecedor esteja previamente cadastrado para que o mesmo possa participar de uma cotação.

Este processo se repete a cada requisição de compra da instituição, sem que se possa ter um controle ou acompanhamento mais detalhado de todo o andamento do processo pelos setores envolvidos e principalmente pelo setor financeiro, o qual é responsável por todas as aprovações de compras.

6.2 O PROCESSO DE COTAÇÃO PROPOSTO

A nova proposta para o processo de compras da UNESCO não difere muito da maneira atual em termos de fluxo do processo, mas sim do ponto de vista em termos de informatização. A informatização pode tornar o processo seguro, delimitando poderes e níveis de acesso às partes que interagem entre si na realização de uma cotação. Além disso a aplicação Web pode proporcionar um acesso mais rápido à informação podendo torná-la acessível a partir de qualquer microcomputador com acesso a Internet.

A mudança mais relevante proposta na estruturação do processo de compras está na forma em que a instituição interage com seus fornecedores. Basicamente a instituição pode realizar suas cotações pela oferta de compra na Internet, onde o

fornecedor passa a ser o maior interessado na realização das cotações de preços dos itens disponibilizados para cotação pela instituição (*e-procurement*).

A abordagem de cotações pela oferta de compra implica na disponibilização de acesso restrito aos fornecedores via Web, que devem estar previamente cadastrados na UNESCO. Com este acesso os fornecedores podem observar quais itens estão sendo cotados no momento, no ramo de atuação dos mesmos, e decidirem efetuar ou não cotações para cada produto. Isto significa que o setor de compras da UNESCO pode somente informar os fornecedores através de um e-mail automático, que um determinado produto está disponível para cotação, até uma determinada data e hora. Desta maneira, evitando contatos telefônicos ou por e-mail para realização de tomadas de preços.

6.2.1 Definindo as partes envolvidas no processo de cotação

Os quatro atores que podem participar do processo de cotação da UNESCO estão definidos nos seguintes perfis:

- a) **Solicitante:** usuários dos diversos setores da instituição autorizados a requisitar bens e materiais de uso e consumo ao setor de compras;
- b) **Compras:** profissional do setor de compras responsável por analisar as requisições, disponibilizá-las aos fornecedores por um determinado período para cotações e depois disso repassá-las para o setor financeiro para aprovação;
- c) **Fornecedor:** fornecedor previamente cadastrados que podem participar das cotações de preços da instituição por meio de um acesso restrito.
- d) **Financeiro:** usuário do setor de finanças responsável por analisar e deferir as compras da instituição;

6.2.1.1 Solicitante

O solicitante basicamente pode requisitar um ou vários produtos em uma requisição informando quantidade, unidade, data limite para entrega e especificações sobre cada produto. Além disso o mesmo pode consultar o andamento de todas as requisições realizadas por ele.

6.2.1.2 Compras

No setor de compras todas as novas requisições podem ser alteradas, podendo-se acrescentar ou retirar itens, refinar as especificações e também determinar a data e hora em que a cotação para cada produto encerra. Após uma requisição de cotação ser publicada para os fornecedores cotarem os preços dos itens, não se pode mais alterar a requisição, somente cancelá-la.

O término do período de cotação de cada item é a diretiva que indica ao usuário do setor de compras ou financeiro que a cotação para aquele item deve ser analisada e então uma tomada de decisão deve ser feita. Geralmente a tomada de decisão é a escolha do melhor preço e emissão de uma ordem de compra ao fornecedor para o produto em questão.

6.2.1.3 Fornecedor

O papel do fornecedor consiste em cotar o preço, forma de pagamento, prazo de entrega, dentre outras informações para cada item disposto no seu espaço restrito por um dado período de tempo. Ao término de cada cotação o mesmo recebe um informativo via *e-mail*, informando-lhe caso ele não tenha vencido a cotação, ou então uma ordem de compra para o faturamento do(s) produto(s) nela contida(s).

6.2.1.4 Financeiro

O gerente financeiro é o responsável direto por todas as compras efetuadas na instituição. Seu papel consiste em analisar as cotações e deferir ou não a compra dos produtos, além de poder deferir uma cotação baseando-se em outros critérios que não seja preço do produto.

6.3 FUNCIONALIDADES IMPLEMENTADAS NO PROTÓTIPO

As tecnologias que compõem o AJAX podem ser utilizadas de muitas maneiras para diversas finalidades. Juntas elas podem formar um conjunto poderoso de alternativas que dependem essencialmente da criatividade do desenvolvedor em aplicá-las de forma a tornar uma aplicação Web mais interativa e produtiva. Algumas dessas alternativas de certo modo já são consideradas implementações cotidianas, e muitas vezes já encontram-se em forma de bibliotecas disponibilizadas gratuitamente na Internet.

A biblioteca escrita em *JavaScript JQuery*, que possui um considerável leque de implementações para manipulação da estrutura de um documento HTML/XML por meio da API DOM e para tratamento de requisições assíncronas (AJAX), possui também vários *plugins*, que oferecem ao desenvolvedor funcionalidades de interação à uma aplicação Web, já implementados. Alguns desses *plugins* foram utilizados neste trabalho os quais encontram-se no site oficial da biblioteca *JQuery*, podendo-se citar (JQUERY, 2007):

- a) *Tabs*
- b) *autocomplete*;
- c) *tablesorter*;
- d) *form*;

- e) *validate*;
- f) *calendar*;
- g) *blockUI*.

A seguir são explanados cada um desses *plugins* bem com as implementações de funcionalidades específicas para este projeto com base na biblioteca *jQuery*, citando as tecnologias envolvidas em cada uma delas.

6.3.1 Tabs

Plugin para implementação de abas no aplicativo Web que permitem o carregamento assíncrono do conteúdo exibido em cada painel, sem a necessidade de recarregar todo o documento. Também é possível a partir da versão 3 do *plugin* manipular facilmente adições e remoções de abas dinamicamente. Na Figura 13 pode-se observar o funcionamento do *plugin* na prática.

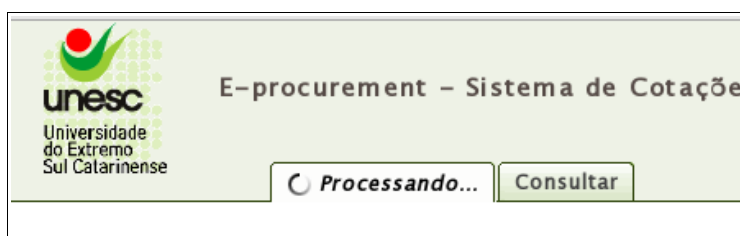


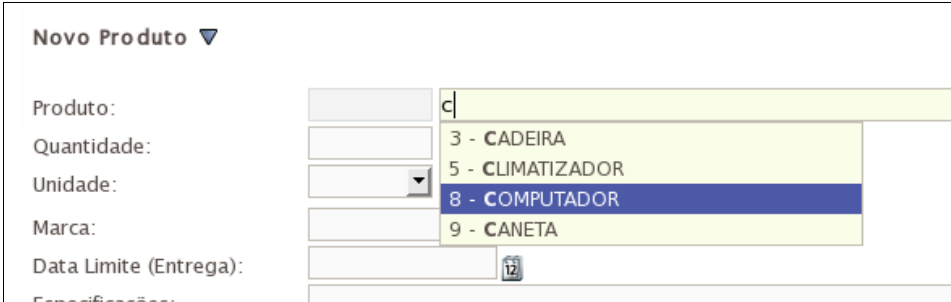
Figura 13. Carregando a aba via AJAX

No protótipo ao clicar pela primeira vez em uma aba uma requisição assíncrona é iniciada, solicitando ao servidor os dados referentes a mesma. Em seguida o servidor envia a resposta resultante do processamento de um *script* PHP e esses dados são incorporados a estrutura do documento com a ajuda da API DOM dentro de uma *tag* DIV, para que o usuário possa visualizá-lo.

6.3.2 Autocomplete

Este *plugin* permite que o usuário, ao digitar algo em determinado campo de texto, receba uma lista de possíveis alternativas que combinem parcialmente com o que se está digitando.

Esta é uma funcionalidade muito utilizada nas aplicações que utilizam AJAX, na qual as várias tecnologias envolvidas estão visivelmente presentes trabalhando em conjunto. A primeira é o *JavaScript*, responsável por capturar os eventos sobre o campo do formulário e a partir disto efetuar uma requisição assíncrona por meio de um objeto *HttpXmlRequest* ao servidor, que utiliza o PHP para consultar as combinações possíveis no banco de dados MySQL e então retorna um objeto do tipo JSON. Este objeto por sua vez é tratado novamente pelo *Javascript*, que junto aos estilos CSS predefinidos, apresenta uma lista de produtos para o usuário, como consequência da manipulação da estrutura do documento por meio da API DOM.



Novo Produto ▼	
Produto:	<input type="text" value="c"/>
Quantidade:	<input type="text"/>
Unidade:	<input type="text"/>
Marca:	<input type="text"/>
Data Limite (Entrega):	<input type="text"/>
Especificações:	<input type="text"/>

- 3 - CADEIRA
- 5 - CLIMATIZADOR
- 8 - COMPUTADOR
- 9 - CANETA

Figura 14. *Autocomplete* para procura de produtos na requisição.

No protótipo essa funcionalidade é implementada no campo de procura por produto, como mostra a Figura 14, onde somente se pode incluir produtos na requisição que já estejam previamente cadastrados no banco de dados. Caso o produto não exista na lista disponível, deve-se solicitar o cadastramento do mesmo junto ao setor de compras da UNESCO.

Outra característica implementada no protótipo é a possibilidade de procura de produto por código no mesmo campo de procura pela descrição. Isto flexibiliza a

digitação da requisição e pode tornar mais ágil a inclusão de uma grande lista de produtos para o usuário, melhorando a usabilidade o sistema.

6.3.3 Tablesorter

Este *plugin* tem como finalidade manipular ordenações de linhas de uma tabela de dados a partir de uma ou mais colunas. Basicamente o *plugin*, utilizando a API DOM, identifica o conteúdo de cada linha referente a(s) coluna(s) selecionada(s) para ordenação e manipula na estrutura do documento a ordem das mesmas.

Código	Produto	Qtde.	Uni.	Marca	Data Lim.	
3	CADEIRA	10	UN		06/12/2007	 
8	COMPUTADOR	10	UN		06/12/2007	 
4	MESA	10	UN		06/12/2007	 

Figura 15. Tabela ordenada por produto utilizando o *plugin tablesorter*.

Como mostra a Figura 15, com este *plugin* pode-se definir quais colunas do cabeçalho de uma tabela podem ser ordenadas ao clicá-las e também aplicar estilo CSS para diferenciar as linhas, facilitando a visualização por parte do usuário.

6.3.4 Form

Com este *plugin* pode-se submeter formulários de forma assíncrona para o servidor de forma simplificada. A resposta do servidor pode ser tratada por uma função após a submissão, para avisar ao usuário sobre o sucesso ou não da submissão, por exemplo.

6.3.5 Validate

O *plugin validate* em conjunto com o *form* oferece uma estrutura para validação de formulário podendo-se customizar mensagens de erro e seu estilo caso um campo não seja preenchido ou preenchido de forma incorreta (Figura 16), além de também possibilitar validações de campos de e-mail, por exemplo, com o uso de expressões regulares.

Figura 16. Validação de formulário.

Outra funcionalidade do *plugin validate* refere-se a possibilidade de estipular quantidades máxima e mínima da caracteres para um campo e também tratá-lo como sendo um campo numérico ou não (JQUERY, 2007).

6.3.6 Calendar

Calendar é *plugin* para auxílio no preenchimento de campos do tipo data que permite fazer várias configurações tais como tradução da interface e customização de períodos de seleção. Este calendário interativo pode oferecer ao usuário uma forma simples e ágil de informar uma data (Figura 17).

Figura 17. Calendário dinâmico escrito em *JavaScript*.

6.3.7 BlockUI

O *plugin blockUI* prove de forma simples o bloqueio das ações do usuário podendo fornecer uma mensagem para o mesmo (Figura 18).



Figura 18. Bloqueio da interface com *feedback* para o usuário.

Dependendo da situação, algumas vezes se faz necessário o bloqueio das ações do usuário para que várias requisições assíncronas possam ser concluídas de forma segura, sem que o usuário tenha mudado de tela ou execute novamente a ação enquanto a aplicação recebe as respostas do servidor.

O funcionamento do *plugin* basicamente consiste no carregamento do mesmo antes do início de uma ou mais requisições assíncronas e então descarregá-lo ao término da operação. Desta maneira o usuário fica impossibilitado de efetuar qualquer mudança na interface da aplicação até que a ação corrente esteja finalizada.

6.3.8 Implementações específicas da aplicação

No protótipo foram utilizando *plugins* de terceiros e também implementados funcionalidades específicas e necessárias para torná-lo mais interativo e melhorar sua usabilidade. Tanto os *plugins* prontos quanto as implementações específicas para a aplicação utilizam a biblioteca *jQuery*. Uma dessas implementações, consistem na utilização de um estrutura de dados *JavaScript* para armazenamento dos produtos de uma requisição na aplicação cliente, que consiste em uma matriz.

Em uma inclusão de requisição de compra é possível manipular uma lista de produtos por parte do usuário de forma bastante ágil. Cada produto da requisição, ao ser incluído na mesma, é adicionado em uma estrutura de dados do tipo matriz (*array* de *arrays*), na qual cada linha corresponde a um produto, e também em uma tabela de dados para que o usuário possa visualizá-lo (Figura 19). Todo este processo envolve apenas a manipulação da estrutura do documento, o que significa que não existe a necessidade de uma comunicação com o servidor para que todos os produtos de uma requisição sejam inclusos, exceto para a procura do produto no banco de dados utilizando o *plugin autocomplete*, entretanto este processo não implica no carregamento da página por completo.

Lista de Produtos

Código	Produto	Qtde.	Uni.	Marca	Data Lim.	
3	CADEIRA	10	UN		06/12/2007	[Ícone]
8	COMPUTADOR	10	UN		06/12/2007	[Ícone]
4	MESA	10	UN		06/12/2007	[Ícone]

Novo Produto

Produto: MESA

Quantidade:

Unidade:

Marca:

Data Limite (Entrega):

Especificações:

Observações:
Mesa de escritório para computador

Figura 19. Formulário de manipulação de requisição de compra.

Ao incluir cada produto na lista também são atribuídos alguns eventos aos mesmos. Todos esses eventos estão relacionados com as linhas na tabela de dados onde os produtos se encontram e podem ser observados na Figura 18, sendo respectivamente os eventos para edição e exclusão presentes na última coluna da tabela de dados e a visualização das observações ao passar o *mouse* sobre cada uma das linhas.

Ao editar um produto, a estrutura de dados que contém a lista de produtos é consultada para a obtenção de todas as informações do produto em questão e assim os

dados são disponibilizados para alteração no respectivos campos do formulário. A exclusão de um item basicamente implica na eliminação de suas informações tanto na estrutura de dados quando tabela de visualização na página.

Na sua maioria, as requisições assíncronas feitas pela aplicação recebem dados do servidor no formato JSON, podendo também variar para HTML ou XML. Esses dados são tratados como objetos na aplicação cliente e então manipulados para que sejam apresentados ao usuário. A implementação desse tipo de estrutura não é simples, pois todas as interações devem ser previstas e manipuladas pelo código *JavaScript*, dependendo estritamente do funcionamento completo do início ao fim da sessão do usuário.

A grande maioria das implementações citadas nas seções anteriores são utilizadas na aplicação e conforme a necessidade são customizadas para suprir necessidades específicas de cada parte do sistema, sempre buscando melhorar os aspectos de usabilidade e interatividade da aplicação.

CONCLUSÃO

No mundo da informática é imprescindível estar atento às novas tendências computacionais que se renovam rapidamente. Analisando os últimos dez anos de evolução, observa-se que a Internet deixou de ser um instrumento que poucos tinham acesso e hoje já é utilizada por um público maior e para os mais diversos fins.

Com a difusão e popularização da Internet muitos sites começaram a disponibilizar uma quantidade maior de serviços na rede, aproveitando dessa maneira as vantagens como facilidade e rapidez de acesso as informações que a Internet propicia. Em contra partida, os usuários se tornaram mais exigentes, aumentando cada vez mais a complexidade das aplicações que disponibilizam os serviços em comparação de uma página Web convencional. Isso significa que as aplicações Web agregam funcionalidades mais abrangentes e que por definição devem trazer o ambiente Web para algo mais próximo das aplicações *Desktop* tradicionais, ou seja, uma aplicação Web deve contemplar conceitos de interatividade e usabilidade, mesmo rodando dentro de um navegador, a fim de suprir as expectativas do usuário.

Como uma alternativa para tornar a Web mais interativa tem-se o AJAX. O AJAX é uma das mais atuais técnicas para o desenvolvimento de aplicações Web e seu princípio básico consiste em possibilitar que dados sejam trafegados entre o cliente e o servidor, por meio de requisições assíncronas, sem a necessidade de carregar novamente uma página Web por completo.

A biblioteca JQuery, por prover uma grande quantidade de funções para a manipulação da estrutura de documentos do tipo HTML/XML, além de dar suporte ao AJAX, demonstra ser uma boa alternativa para o desenvolvimento de aplicações Web mais interativas. O *JQuery*, possibilita que o desenvolvedor implemente as interações da aplicação Web escrevendo menos código e produzindo mais. Entretanto, isto ainda não

é o suficiente para que o desenvolvimento e a depuração de *scripts JavaScript* sejam consideradas tarefas fáceis de serem executadas.

Utilizando a biblioteca *jQuery*, pôde-se implementar algumas funcionalidades no protótipo de Pregão Eletrônico proposto, podendo-se assim, verificar as reais expectativas que esta nova abordagem pode trazer para o desenvolvimento Web e também verificar as dificuldades que implicam na sua utilização.

O desenvolvimento de uma aplicação Web utilizando AJAX, pode ser uma tarefa complexa, dependendo estritamente da aplicação e das funcionalidades que se deseja implementar. Desenvolver um aplicativo completamente com AJAX exige planejamento e estrutura de projeto, pois a aplicação deve rodar no navegador do cliente do início ao fim da sessão do usuário. Isso torna o desenvolvimento ainda mais delicado, tendo em vista a existência de vários navegadores, os quais são os recipientes para a execução da aplicação cliente. Desta forma, o aplicativo deve ser desenvolvido com suporte necessário aos navegadores distintos, que nem sempre seguem os padrões determinados pela W3C.

Ao se utilizar a biblioteca *jQuery* juntamente com seus *plugins*, outra dificuldade pode ser encontrada em relação as diferentes versões do *jQuery* que os *plugins* requerem, além da compatibilidade entre si dos diferentes *plugins*. Isso pode se tornar um incômodo em alguns casos, principalmente quando há a necessidade de aplicar *plugins* distintos em um mesmo elemento no desenvolvimento da aplicação.

O desenvolvimento de uma aplicação com AJAX entretanto também tem seus pontos positivos. No protótipo proposto e também no decorrer deste trabalho pôde-se observar as facilidades e novas concepções que o uso dessa técnica pode trazer para o ambiente Web. A possibilidade de se fazer requisições assíncronas ao servidor e então incorporar os dados obtidos em partes da aplicação, é algo que abre as portas para o encontro da interatividade na Web.

Seguindo esta tendência, o protótipo de Pregão Eletrônico desenvolvido utilizando conceitos de *e-procurement* para a UNESCO, pode se tornar uma boa ferramenta para a realização de tomadas de preços da instituição ao final de sua implementação, agregando o que há de mais novo em relação ao desenvolvimento e interatividade na Web. O desenvolvimento deste protótipo foi iniciado com o intuito de que possa ser utilizado pela UNESCO em produção e, por decorrência disto, estudou-se o caso de uso referente à ela.

Para o desenvolvimento deste trabalho, houve também a preocupação de se utilizar somente ferramentas e tecnologias livres, que possam ser aplicadas nos diversos seguimentos, abrindo ainda mais o horizonte quanto a independência de ferramentas ou tecnologias proprietárias obtidas por meio de licenças.

Aliado a esse fato, o trabalho apresentado pode vir a ser utilizado como base para futuras aplicações Web, que pretendam agregar questões relacionadas a interatividade e melhor usabilidade. Desta forma, este trabalho deixa como forma de contribuição ao curso e aos alunos de Ciência da Computação, conceitos estudados ao longo do curso empregados sobre a aplicação experimental, além da explanação das diretrizes de uma das mais atuais concepções no que se refere a interatividade e usabilidade no desenvolvimento Web, o AJAX.

Com o estudo e a implementação do trabalho proposto, pôde-se aprender sobre as reais dificuldades e vantagens de se implementar uma aplicação Web utilizando AJAX, aprimorando assim, o conhecimento sobre o desenvolvimento de aplicações Web interativas. Além disto, conceitos de negócios aplicados no meio eletrônico, também foram assimilados, podendo-se perceber o quanto a Internet e as aplicações Web podem contribuir para melhorar as relações comerciais entre empresas, sobre tudo, com aplicações Web mais interativas.

REFERÊNCIAS

- AMOR, Daniel. **A Revolução do E-Business**. São Paulo: Makron Books, 2000.
- ANSELMO, Fernando. **PHP e MySQL para Windows**. São Paulo: Visual Books, 2000.
- BORBA, Fernando Emanuel. **AJAX: Guia de programação**. São Paulo: Érica, 2006.
- BROILO, Marcus. **Modelagem de uma plataforma de compras eletrônica e-procurement**. 2006. 175 f. Trabalho de Conclusão de Curso – Universidade do Sul de Santa Catarina, Palhoça, 2006.
- CHEN, Peter. **Gerenciando banco de dados: a abordagem entidade-relacionamento para projeto lógico**. São Paulo: McGraw-Hill, 1990.
- CODEIGNITER. **CodeIgniter User Guide Version 1.5.4**. Disponível em: <http://codeigniter.com/user_guide/>. Acesso em: 13 novembro 2007.
- CONVERSE, Tim; PARK, Joyce. **PHP 4: a bíblia**. Rio de Janeiro: Campus, 2001.
- CRANE, Dave; PASCARELLO, Eric; JAMES, Darren. **Ajax em ação**. São Paulo: Pearson Prentice Hall, 2006.
- CRONIN, Mary J. **Fazendo business via Internet: como a via eletrônica está transformando as empresas americanas**. São Paulo: Érica, 1995.
- FRANCO JUNIOR, Carlos F. **e-Business: tecnologia de informação e negócios na Internet**. São Paulo: Atlas, 2001.
- GONÇALVES, Bruno Leandro Morais Freitas. **Estudo da concepção, definição e projeto de aplicações Web enriquecidas com AJAX**. 2006. 70 f. Trabalho de Conclusão de Curso – Universidade Federal do Mato Grosso, Ponta do Araguaia, 2006.
- GONÇALVES, Edson. **Ajax na prática**. Rio de Janeiro: Ed. Ciência Moderna, 2007.
- GONÇALVES, Edson. **Dominando Ajax**. Rio de Janeiro: Ed. Ciência Moderna, 2006.
- GRAHAM, Ian S. **HTML: a referência completa para HTML 3.2 e das extensões HTML**. Rio de Janeiro: Ed. Campus, 1998.
- GLOOR, Peter A. **Transformando a empresa em e-business: como ter sucesso na economia digital**. São Paulo: Atlas, 2001.
- JQUERY. **jQuery is a new type of JavaScript library**. Disponível em: <<http://jquery.com/>>. Acesso em: 13 setembro 2007.
- KOCH, Rodrigo de Souza. **Estudo da metodologia de programação AJAX aplicada a um sistema de plano de negócios gratuito para a Web**. 2006. 73 f. Trabalho de

Conclusão de Curso – Sociedade Educacional de Santa Catarina – Sociesc Instituto Superior Tupy – IST, Joinville, 2006.

LIMEIRA, José Luiz Silveira. **Utilização de AJAX no desenvolvimento de sistemas Web**. 2006. 44 f. Monografia de Especialização – Universidade Federal do Rio Grande do Sul – Instituto de informática, Porto Alegre, 2006.

MYSQL AB. *MySQL 5.1 Reference Manual*. Disponível em: <<http://dev.mysql.com/doc/refman/5.1/en/>>. Acesso em: 20 maio 2007.

NIEDERAUER, Juliano. **Livro Web Interativa com Ajax e PHP**. São Paulo: Novatec, 2007.

SILVA, Luciano Carlos da. **Banco de dados para Web: do planejamento à implementação**. São Paulo: Érica, 2001.

SOARES, Wallace. **AJAX (Asynchronous Javascript and XML): Guia prático para Windows**. 2. ed. São Paulo: Érica, 2006.

TEIXEIRA, Flávia Silva. **Verificação do entendimento e da utilização do e-procurement em duas empresas globalizadas**. 2003. 66 f. Monografia (MBA em Gerência Financeira e Controladoria) - Departamento de Economia, Contabilidade, Administração e Secretariado – ECASE, Universidade de Taubaté, Taubaté, 2003.

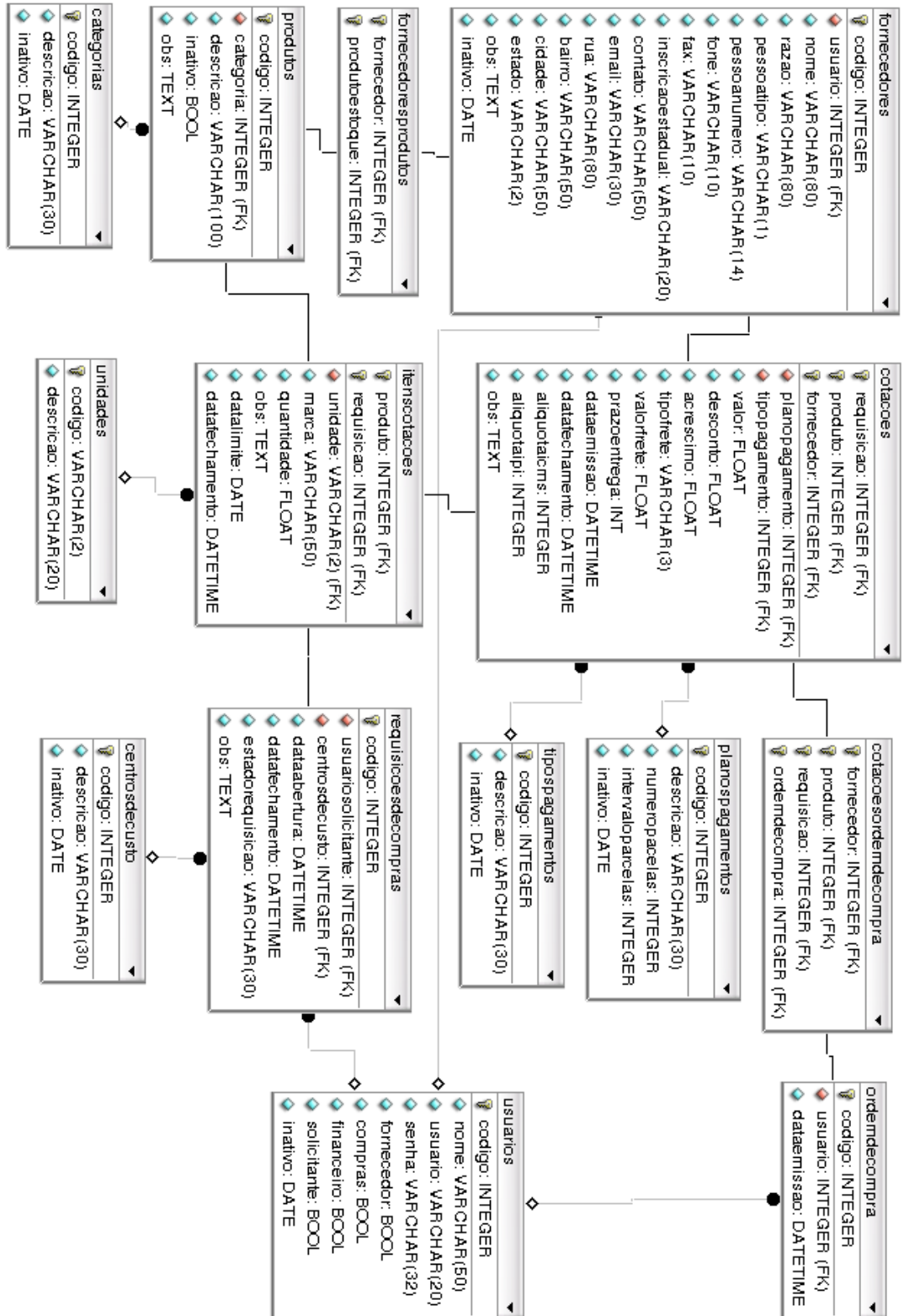
TURBAN, Efraim; KING, David. **Comércio Eletrônico: Estratégia e Gestão**. São Paulo: Pearson, 2004.

VIEIRA, Victor. **Www.com.br**. Por dentro do Ajax. São Paulo: Europa, 2006.

ZAKAS, Nicholas C.; MCPEAK, Jeremy; FAWCETT, Joe. **Professional AJAX**. Indianapolis: Wiley Publishing, 2006.

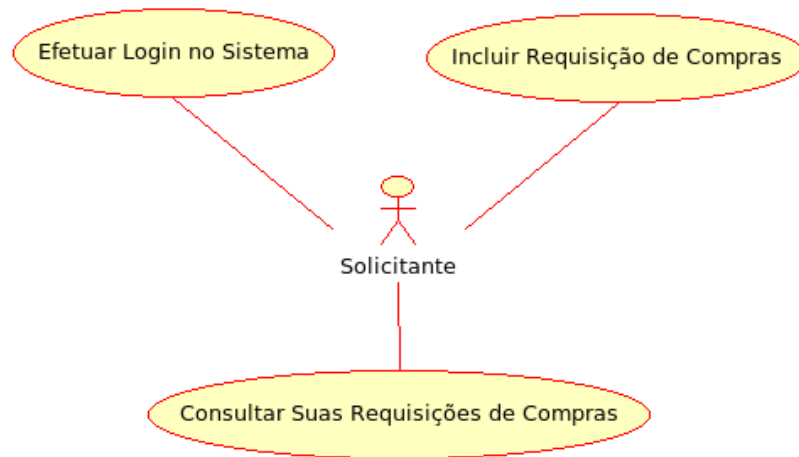
APÊNDICE A – DIAGRAMA ER DO PROTÓTIPO

Diagrama ER inicial do protótipo.



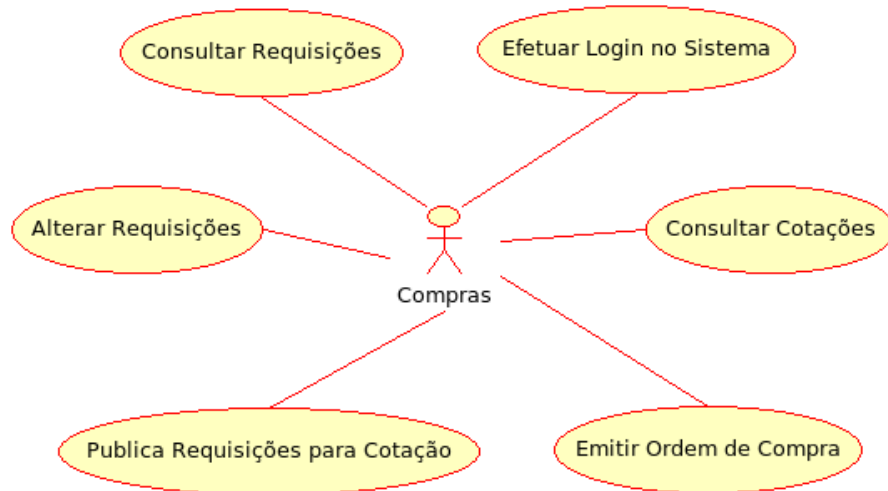
APÊNDICE B – DIAGRAMA DE CASO DE USO - SOLICITANTE

Diagrama de caso de uso do perfil solicitante.



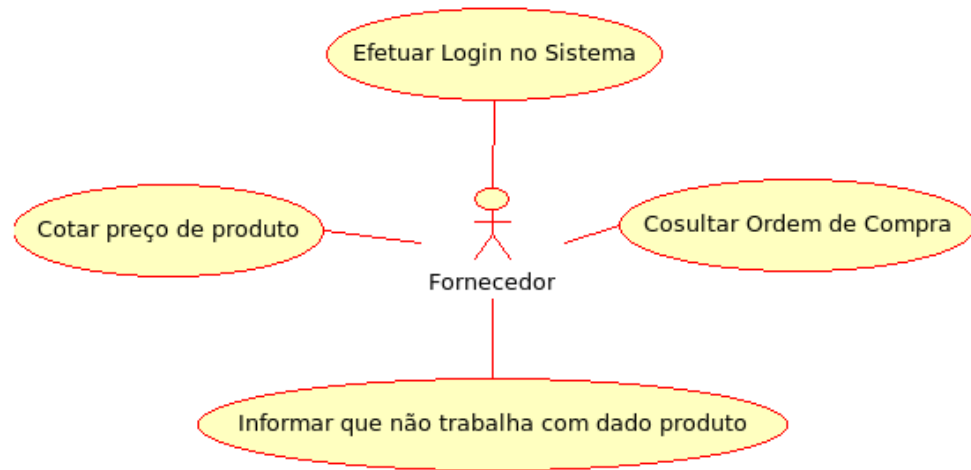
APÊNDICE C – DIAGRAMA DE CASO DE USO - COMPRAS

Diagrama de caso de uso do perfil compras.



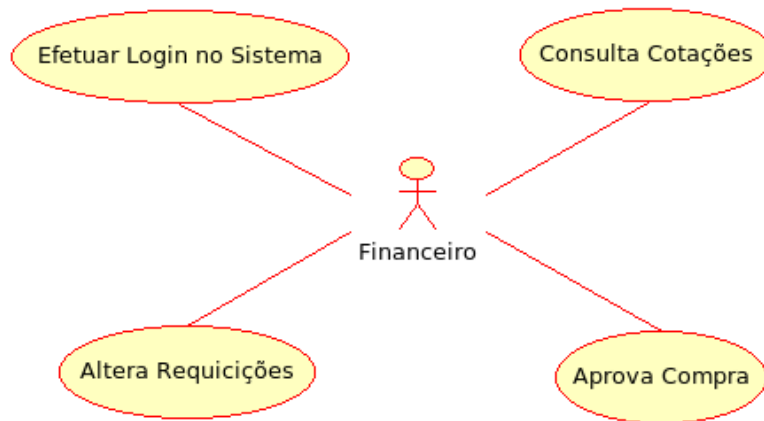
APÊNDICE D – DIAGRAMA DE CASO DE USO - FORNECEDOR

Diagrama de caso de uso do perfil fornecedor.



APÊNDICE E – DIAGRAMA DE CASO DE USO - FINANCEIRO

Diagrama de caso de uso do perfil financeiro.



APÊNDICE F – ARTIGO SOBRE O TRABALHO

Aplicação experimental do conceito de *e-procurement* utilizando AJAX: Um caso de uso na UNESC

**Diego Búrigo Zacarão¹, Rogério Antônio Casagrande¹,
Fabrizio Colombo Machado¹**

¹Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC)
Criciúma – SC – Brazil

diegobz@gmail.com, {roger,fabrizio}@unesc.net

Abstract. *The aim of this paper is to present an alternative to make the Web more interactive, providing greater usability for users, denominated AJAX. As AJAX is a programming technique, it has been applied in the development of a Web application prototype to realize quotations of prices at the University of UNESC (Universidade do Extremo Sul Catarinense). In order to develop this implementation, concepts of Business to Business and E-procurement were studied, besides several programming technologies that composes AJAX. In the end of this study, it is possible to observe how AJAX can make Web environment more interactive, with the connotation to turn the Web applications something close of conventional Desktop applications*

Resumo. *Este artigo tem como objetivo apresentar uma alternativa para tornar a Web mais interativa, proporcionando maior usabilidade para os usuários, denominada AJAX. Sendo o AJAX uma técnica de programação, a mesma foi aplicada no desenvolvimento de um protótipo de Pregão Eletrônico para a realização de cotações na Universidade do Extremo Sul Catarinense – UNESC. Para que tal implementação pudesse ser desenvolvida, conceitos de Business to Business e E-procurement foram estudados, além de várias tecnologias de programação Web que compõem o AJAX. Ao final do estudo, pode-se observar como o AJAX pode tornar o ambiente Web mais interativo, com a conotação de tornar as aplicações Web algo mais próximo das aplicações Desktop convencionais.*

1 Introdução

Em uma instituição o processo de compras e cotações de mercadorias, denominado *procurement*, pode ser dificultoso devido a demanda de requisições de compras. Na Universidade do Extremo Sul Catarinense – UNESC esse processo acaba gerando um alto custo, principalmente na carga de trabalho dos profissionais deste setor. Devido ao tamanho da instituição, esse processo torna-se complexo e de difícil gerenciamento, tomando muito tempo em correções decorrentes dos processos manuais envolvidos e em atendimentos aos fornecedores por meio de e-mail e/ou telefone.

A utilização de ferramentas integradas a Internet que sejam dinâmicas e que possam proporcionar maior agilidade e interatividade aos usuários, pode ser uma alternativa para o processo de compras de uma instituição. Com este propósito, surge o conceito de *e-procurement*, que por definição tem como objetivo transferir para a Web o processo e gerenciamento de compras de suprimentos aliviando, assim, a carga de trabalho e os custos dessa área nas corporações.

A evolução das aplicações Web faz com que a busca por novas técnicas e tecnologias de programação seja constante, a fim de otimizar e obter maior performance das aplicações. O AJAX vem como uma alternativa a esse contexto e tem como princípio básico a otimização da quantidade de dados transferidos nas interações feitas entre cliente e servidor. O AJAX utiliza-se de tecnologias aplicadas a Web para fazer requisições assíncronas ao servidor. Este fato proporciona inúmeras maneiras de se trabalhar com componentes dinâmicos em páginas Web, pois permite requisitar dados de forma assíncrona, sem a necessidade de recarregar uma página por completo.

Com o objetivo de explanar os benefícios do AJAX, usufruindo de uma maior interatividade e melhor usabilidade, foi escolhido um caso de uso na UNESCO referente ao desenvolvimento de um protótipo de um Pregão Eletrônico Web. Esta aplicação baseia-se em conceitos de *e-procurement*, a fim de oferecer uma alternativa que possa agilizar o processo de aquisição e manutenção de estoque, reduzir custos nas negociações e controlar o fluxo de aprovação da compra de materiais utilizando os recursos de Internet e principalmente agregando concepções referentes a interatividade sobre a Web.

2 Tecnologias de Programação

No decorrer de vários anos de evolução da Web, muitos conceitos tecnológicos surgiram para suprir necessidades específicas, que cada vez mais são almejadas para que se possibilite um maior aproveitamento da Internet. Em contra partida, a técnica de programação AJAX é um conceito que agrega uma série de tecnologias já existentes, trabalhando em conjunto, cada uma fazendo sua parte, para oferecer novas funcionalidades, que vão além de páginas Web dinâmicas que não possibilitam maior interatividade ao usuário.

2.1 HTML

O *HyperText Markup Language* (HTML) é uma linguagem de programação que permite apresentar informações de forma gráfica na Internet. Esta tecnologia utiliza-se de marcações, também chamadas de *tags*, sobre o texto para informar ao *browser*, navegador de Internet, como cada elemento deve ser manipulado e apresentado na tela (GRAHAM, 1998).

2.2 CSS

O *Cascading Style Sheets* (CSS) é uma diretiva bem estabelecida do Web design sendo uma das tecnologias mais importantes para o AJAX. Uma folha de estilo (CSS) oferece uma maneira centralizada de definir todas as categorias de estilos visuais que farão parte de uma página e aplicá-las concisamente em seus elementos. Além de feitos básicos de estilização como cores, bordas, imagens de fundo e tamanho, as folhas de estilo podem definir a disposição dos elementos em relação um ao outro na janela do navegador (CRANE; PASCARELLO; JAMES, 2006).

2.3 JavaScript

A proposta básica do *JavaScript* consiste em não se fazer constantes requisições ao servidor para realizar tarefas simples, como validação de dados, exibição de mensagens na tela, cálculos diversos, entre outras tarefas. A responsabilidade de executar estas tarefas é repassada ao navegador do usuário (ZAKAS, 2006).

O papel central no conjunto de tecnologias que compõem o AJAX é sem dúvida desempenhado pelo *JavaScript*. Uma aplicação AJAX carrega um cliente completo na

memória, combinando dados, apresentação e a lógica do programa. O *JavaScript* é a ferramenta utilizada para implementar essa lógica, a qual deve ser robusta o suficiente para que o código funcione adequadamente nas diferentes versões de navegadores existentes (CRANE; PASCARELLO; JAMES, 2006).

2.4 DOM

O *Document Object Model* (DOM) é uma especificação que define uma interface e uma plataforma neutra permitindo que programas e *scripts* preencham e alterem o conteúdo, a estrutura e o estilo dos documentos, de uma forma normalizada, dando condições para uma maior portabilidade de programas e *scripts* (SOARES, 2006). O DOM possibilita que uma página Web seja manipulada representando um conjunto de nós, permitindo acesso direto a qualquer elemento da página, por meio de um *JavaScript*. O DOM apresenta os documento HTML como uma hierarquia de nós que podem implementar interfaces especializadas. Por exemplo, o elemento <TABLE> do HTML está hierarquicamente acima dos elementos <TR>, os quais representam as linhas dentro da tabela, estes por sua vez, estando acima dos elementos <TD> de sua estrutura, que representam as colunas dentro de uma linha (SOARES, 2006).

2.5 XML

Quando é necessário que se haja troca de informações entre elementos, é preciso estabelecer uma linguagem de comunicação por meio de regras. O XML tem o propósito de definir e ser a forma de manipulação dessas regras, utilizando-se de *tags*, que serão usadas para que os elementos se comuniquem de uma forma simples. A estruturação de dados usando o XML nas aplicações AJAX, tem o papel de fazer a com que a parte cliente, neste caso o navegador do usuário, e o servidor Web, consigam trocar informações por meio de regras de comunicação pré-estabelecidas (BORBA, 2006).

2.6 JSON

O *JavaScript Object Notation* (JSON) é um formato de dados construído em *JavaScript* em alternativa ao XML. Apesar do XML ser praticamente um padrão de interoperabilidade entre sistemas e em servidores Web para transmissão de dados, alguns consideram o mesmo demasiadamente complexo somente para transmissão de dados, necessitando enviar mais *bytes* pela Internet do que realmente haveria necessidade (GONÇALVES, 2007).

Formato XML	Formato JSON
<pre data-bbox="367 1668 893 1870"><departamentos> <departamento> <codigo>1</codigo> <nome>Informática</nome> </departamento> </departamentos></pre> <p data-bbox="742 1892 885 1926" style="text-align: right;">102 bytes</p>	<pre data-bbox="933 1668 1364 1870">"departamentos" : [{ "codigo" : 1, "nome" : "Informática" }]</pre> <p data-bbox="1236 1892 1372 1926" style="text-align: right;">51 bytes</p>

Figura 1. Comparação entre os formatos XML e JSON
 Fonte: GONÇALVES, E. (2007)

O JSON surge com o objetivo de ser um formato de dados muito leve baseado em um subconjunto da sintaxe *JavaScript*, podendo-se incluir definições JSON em códigos *JavaScript* e acessá-las sem análise gramatical extra por linguagens baseadas em XML. Além disso, devido a sua sintaxe simplificada, o JSON consegue ser consideravelmente menor comparado ao XML (Figura 1), o que lhe traz vantagens levando em conta o tamanho dos arquivos gerados e conseqüentemente na velocidade de sua transmissão (GONÇALVES, 2007).

2.7 XMLHttpRequest

O *XMLHttpRequest* é o principal fundamento do AJAX. Ele é responsável por possibilitar que os navegadores dos usuários consigam fazer requisições assíncronas ao servidor Web, de forma transparente. Isto significa que um objeto *XMLHttpRequest* pode servir de intermediário, oferecendo o suporte, para receber ou enviar dados para o servidor, assincronamente, sem precisar carregar toda a página Web a cada requisição (CRANE; PASCARELLO; JAMES, 2006).

2.8 PHP

O *PHP: Hypertext PreProcessor* é uma linguagem de programação Web interpretada com a finalidade de gerar páginas HTML dinamicamente. Todo código PHP é interpretado pelo servidor que, então, retorna ao cliente a página resultante do processamento. Desta forma o PHP consegue interagir com outros serviços disponíveis no servidor, como por exemplo, um banco de dados, possibilitando assim que se possa consultar dados e apresentá-los em páginas HTML de forma simples (ANSELMO, 2000).

O PHP não é uma tecnologia base para o funcionamento do AJAX, qualquer linguagem que consiga gerar páginas HTML dinamicamente pode substituí-la.

2.9 MySQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Structured Query Language - Linguagem Estruturada para Pesquisas), a qual é o padrão utilizado pela maioria dos bancos de dados relacionais. O MySQL se tornou um banco de dados popular, para utilização na Internet, devido sua robustez, rapidez e concepções de multi-tarefa e multi-usuário. Além disto, o MySQL é multiplataforma e contém bibliotecas para interação com várias linguagens de programação como JAVA, C/C++, Python, Perl, PHP e Ruby (MYSQL AB, 2007).

3 AJAX

Ao contrário do que se possa muitas vezes imaginar, o AJAX não é nenhuma nova tecnologia. AJAX na realidade é uma técnica que surgiu como uma alternativa, que utiliza tecnologias já existentes, cada uma fazendo sua parte, para que, em conjunto, possibilitem que a Web seja mais dinâmica, quebrando o paradigma de páginas estáticas e de baixa interação com o usuário (CRANE; PASCARELLO; JAMES, 2006).

Uma das principais funcionalidade do AJAX é a requisição assíncrona ao servidor Web. Desta maneira é possível realizar consultas sem que uma página seja novamente carregada inteiramente, evitando tráfego desnecessário de informações não relevantes à ação que se esteja executando no momento. Contudo, esta funcionalidade não se limita somente a redução de tráfego, e sim, também a uma considerável gama de

possibilidades de interações que o assincronismo de requisições possibilita em um aplicativo baseado sobre a Web.

A Figura 2 demonstra o fluxo de uma requisição assíncrona.

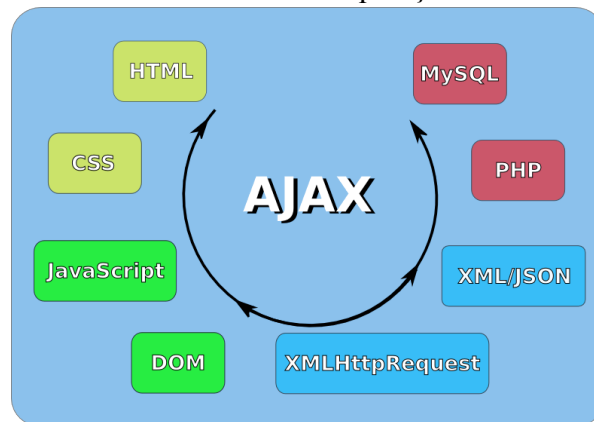


Figura 2. Fluxo de interações entre as tecnologias que compõem o AJAX.

Existe ocasiões as quais a utilização do AJAX nem sempre será a melhor solução. O AJAX pode ser utilizado de várias maneiras em aplicações na Internet, podendo-se construir aplicações convencionais totalmente em AJAX ou aplicações híbridas, que utilizarão esta técnica somente em funções específicas do Website.

Uma aplicação completa em AJAX terá que carregar um código *JavaScript* que permanecerá na sessão do usuário até que o mesmo saia do sistema. Esta aplicação geralmente se assemelha ao máximo com as aplicações *desktop* convencionais e portando deverá ser projetada com as mesmas preocupações e funcionalidades inerentes. Desta forma, o custo de se desenvolver um sistema completo em AJAX pode se tornar de alto, comparado com o desenvolvimento baseado em páginas Web clássicas. Em contra partida, este é o investimento que poderá se transformar em um aplicativo de grande interatividade e usabilidade (CRANE; PASCARELLO; JAMES, 2006).

Em alternativa ao custo de aplicações completamente escritas em AJAX, pode-se criar aplicações híbridas, que no seu escopo geral, serão desenvolvidas no modelo clássico de páginas Web, porém usufruindo do AJAX em determinadas funções na aplicação, onde se necessite de maior interatividade e melhor usabilidade, como por exemplo em:

- a) Inserções, exclusões e atualizações de itens em uma lista;
- b) interações com formulários;
- c) navegações em árvores de grande hierarquia;
- d) comunicação rápida entre usuários;
- e) enquetes e caixas de marcação;
- f) filtro e aperfeiçoamento de manipulação de dados;
- g) autocompletar;
- h) erros interativos;
- i) operações computacionais de alto custo.

O AJAX possibilita que as operações citadas acima possam ser realizadas com rapidez e menor custo no tráfego de dados, também podendo tornar a aplicação mais atrativa aos usuários.

3.1 Bibliotecas *JavaScript* com Suporte ao AJAX

Em prol da produtividade do desenvolvimento Web aliada a um considerável número de características comuns agregadas aos sites, focando a interatividade com o usuário, pessoas começaram a implementar bibliotecas *JavaScript* e disponibilizá-las na Internet. Algumas dessas iniciativas atualmente são capazes de auxiliar na interatividade com o usuário, que vão desde simples verificações de formulários, o que é comum no desenvolvimento Web, até sofisticados sistemas de *autocomplete* (GONÇALVES, 2006).

Dentre as bibliotecas mais reconhecidas que dão suporte a requisições assíncronas (AJAX), pode-se citar:

- a) *JQuery*
- b) *Prototype e Script.aculo.us*
- c) *Dojo Toolkit*
- d) *Yahoo! UI*
- e) *Ext*

Cada biblioteca tem sua peculiaridade, entretanto todas focam a interatividade na Web como objetivo principal. *JQuery* foi a biblioteca escolhida para construção deste trabalho, pois apresenta-se como uma biblioteca leve, de simples implementação, com uma série de plugins disponíveis, e que se pode manipular facilmente a estrutura de documento HTML/XML por meio da API DOM, suprimindo os requisitos para o desenvolvimento da aplicação experimental proposta (GONÇALVES, 2007).

4 *E-business*

Segundo Franco Junior (2001) dentre as diferentes áreas do *e-business*, cada uma é responsável por algum tipo de relacionamento entre entidades podendo-se citar algumas classes como:

- a) *Business to Consumer (B2C)*;
- b) *Consumer to Consumer (C2C)*;
- c) *Business to Business (B2B)*.

O B2B é a classe responsável pelo relacionamento entre empresas, sendo o *e-procurement*, uma de suas ramificações, a qual abrange mais especificamente os conceitos do processo de compra de bens e serviços eletronicamente em uma instituição.

4.1 *E-procurement*: Aperfeiçoando o Processo de Compra

O *e-procurement* pode ser relacionado com o “departamento de compras” de uma empresa tradicional e tem como conceito transferir para a Web o processo e gerenciamento de compras de suprimentos aliviando, assim, a carga de trabalho e os custos dessa área nas corporações. Este sistema baseia-se na oferta de compra e os fornecedores podem fazer suas cotação diretamente pela Internet, quando alertados por meio de um e-mail, por exemplo, sem que haja um outro tipo de contato prévio (TEIXEIRA, 2003).

O *e-procurement* traz a eliminação do papel, cotações/licitações de preços mais abrangentes e a possibilidade de acompanhar melhor a performance dos fornecedores.

5 Caso de Uso na UNESC

O processo de *procurement* da Universidade do Extremo Sul Catarinense - UNESC, atualmente funciona do modo tradicional. Realiza-se em média três cotações à empresas e então os profissionais desta área analisam as propostas com base em alguns fatores, como preço e qualidade do produto.

Devido ao tamanho da instituição, este modo tradicional torna-se lento e de difícil controle e gerenciamento, demandando muito tempo, inclusive para correções de erros decorrentes dos processos manuais. Os custos operacionais, levando em consideração, as ligações telefônicas e a carga de trabalho dos profissionais do setor de compras também acabam se elevando. Desta maneira, a utilização de uma estrutura informatizada e compatível com a Web, pode tornar o gerenciamento desse processo mais eficiente e diminuir os custos.

5.1 O Atual Processo de Cotações da UNESC

A solicitação de bens ou produtos para uso e consumo iniciam-se por meio de uma requisição de compra dentro do próprio sistema interno da instituição. Vários setores da universidade tem acesso a esse módulo e podem fazer solicitações ao departamento de compras, acompanhadas de um memorando assinado pelo responsável do setor. O departamento de compras da prosseguimento nos trâmites seguintes que envolvem: orçamentos, aprovação e finalmente a compra propriamente dita dos itens requisitados.

5.2 O Processo de Cotação Proposto

A nova proposta para o processo de compras da UNESC não difere muito da maneira atual em termos de fluxo do processo, mas sim do ponto de vista em termos de informatização. A informatização pode tornar o processo seguro, delimitando poderes e níveis de acesso às partes que interagem entre si na realização de uma cotação para compra. Além disso a aplicação Web pode proporcionar um acesso mais rápido à informação podendo torná-la acessível a partir de qualquer microcomputador com acesso a Internet.

A mudança mais relevante proposta na estruturação do processo de compras esta na forma em que a instituição interage com seus fornecedores. Basicamente a instituição pode realizar suas cotações pela oferta de compra na Internet, onde o fornecedor passa a ser o maior interessado na realização das cotações de preços dos itens disponibilizados para cotação pela instituição (*e-procurement*).

5.3 Definindo as Partes Envolvidas no Processo de Cotação

A Os quatro atores que podem participar do processo de cotação da UNESC estão definidos nos seguintes perfis:

- a) **Solicitante:** usuários dos diversos setores da instituição autorizados a requisitar bens e materiais de uso e consumo ao setor de compras;
- b) **Compras:** profissional do setor de compras responsável por analisar as requisições, disponibilizá-las aos fornecedores por um determinado período para cotações e depois disso repassá-las para o setor financeiro para aprovação;
- c) **Fornecedor:** fornecedor previamente cadastrados que podem participar das cotações de preços da instituição por meio de um acesso restrito.

- d) **Financeiro**: usuário do setor de finanças responsável por analisar e deferir as compras da instituição;

5.4 Funcionalidades Implementadas no Protótipo

As tecnologias que compõem o AJAX podem ser utilizadas de muitas maneiras para diversas finalidades. Juntas elas podem formar um conjunto poderoso de alternativas que dependem essencialmente da criatividade do desenvolvedor em aplicá-las de forma a tornar uma aplicação Web mais interativa e produtiva. Algumas dessas alternativas de certo modo já são consideradas implementações cotidianas, e muitas vezes já encontram-se em forma de bibliotecas disponibilizadas gratuitamente na Internet.

A biblioteca escrita em *JavaScript* JQuery, que possui um considerável leque de implementações para manipulação da estrutura de um documento HTML/XML por meio da API DOM e para tratamento de requisições assíncronas (AJAX), possui também vários *plugins*, que oferecem ao desenvolvedor funcionalidades de interação à uma aplicação Web, já implementados. Alguns desses *plugins* foram utilizados neste trabalho os quais encontram-se no site oficial da biblioteca JQuery, podendo-se citar (JQUERY, 2007):

- a) *Tabs*
- b) *autocomplete*;
- c) *tablesorter*;
- d) *form*;
- e) *validate*;
- f) *calendar*;
- g) *blockUI*.

5.4.1. Tabs

Plugin para implementação de abas no aplicativo Web que permitem o carregamento assíncrono do conteúdo exibido em cada painel, sem a necessidade de recarregar o documento todo. Também é possível a partir da versão 3 do *plugin* manipular facilmente adições e remoções de abas dinamicamente. Na Figura 3 pode-se observar o funcionamento do *plugin* na prática.

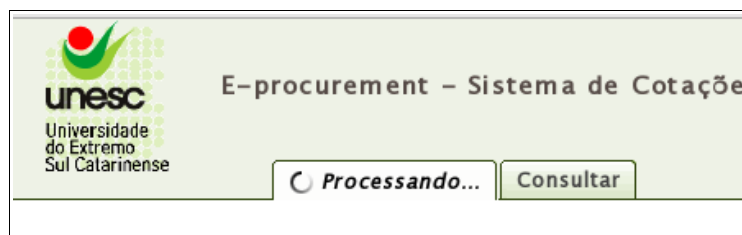


Figura 3. Carregando a aba via AJAX

5.4.2. Autocomplete

Plugin para Este *plugin* permite que o usuário, ao digitar algo em determinado campo de texto, receba uma lista de possíveis alternativas que combinem parcialmente com o que se está digitando.

Esta é uma funcionalidade muito utilizada nas aplicações que utilizam AJAX, na qual as várias tecnologias envolvidas estão visivelmente presentes trabalhando em conjunto. A primeira é o *JavaScript*, responsável por capturar os eventos sobre o campo do formulário e a partir disto efetuar uma requisição assíncrona por meio de um objeto *HttpXmlRequest* ao servidor, que utiliza o PHP para consultar as combinações possíveis no banco de dados MySQL e então retorna um objeto do tipo JSON. Este objeto por sua vez é tratado novamente pelo *JavaScript*, que junto aos estilos CSS predefinidos, apresenta uma lista de produtos para o usuário, como consequência da manipulação da estrutura do documento por meio da API DOM.

A imagem mostra um formulário web com o título "Novo Produto" e um ícone de seta para baixo. O formulário contém os seguintes campos: "Produto:", "Quantidade:", "Unidade:", "Marca:", "Data Limite (Entrega):" e "Especificação:". O campo "Produto:" contém o caractere "c" e está aberto, exibindo uma lista de sugestões de produtos: "3 - CADEIRA", "5 - CLIMATIZADOR", "8 - COMPUTADOR" (destacado em azul) e "9 - CANETA".

Figura 4. *Autocomplete* para procura de produtos na requisição.

6 Conclusão

No mundo da informática é imprescindível estar atento às novas tendências computacionais que se renovam rapidamente. Analisando os últimos dez anos de evolução, observamos que a Internet deixou de ser um instrumento que poucos tinham acesso e hoje já é utilizada por um público maior e para os mais diversos fins.

Com a difusão e popularização da Internet muitos sites começaram a disponibilizar uma quantidade maior de serviços na rede, aproveitando dessa maneira as vantagens de facilidade e rapidez de acesso às informações que a Internet propicia. Em contrapartida, os usuários se tornaram mais exigentes, aumentando cada vez mais a complexidade das aplicações que disponibilizam os serviços em comparação de uma página Web convencional. Isso significa que as aplicações Web agregam funcionalidades mais abrangentes e que por definição devem trazer o ambiente Web para algo mais próximo das aplicações *Desktop* tradicionais, ou seja, uma aplicação Web deve contemplar conceitos de interatividade e usabilidade, mesmo rodando dentro de um navegador, a fim de suprir as expectativas do usuário.

Como uma alternativa para tornar a Web mais interativa tem-se o AJAX. O AJAX é uma das mais atuais técnicas para o desenvolvimento de aplicações Web e seu princípio básico consiste em possibilitar que dados sejam trafegados entre o cliente e o servidor, por meio de requisições assíncronas, sem a necessidade de carregar novamente uma página Web por completo.

A biblioteca JQuery, por prover uma grande quantidade de funções para a manipulação da estrutura de documentos do tipo HTML/XML, além de dar suporte ao AJAX, demonstra ser uma boa alternativa para o desenvolvimento de aplicações Web mais interativas. O JQuery possibilita que o desenvolvedor implemente as interações da aplicação Web escrevendo menos código e produzindo mais. Entretanto, isto ainda não é o suficiente para que o desenvolvimento e a depuração de *scripts JavaScript* sejam consideradas tarefas fáceis de serem executadas.

O desenvolvimento de uma aplicação Web utilizando AJAX, pode ser uma tarefa complexa, dependendo estritamente da aplicação e das funcionalidades que se deseja implementar. Desenvolver um aplicativo completamente com AJAX exige

planejamento e estrutura de projeto, pois a aplicação deve rodar no navegador do cliente do início ao fim da sessão do usuário. Isso torna o desenvolvimento ainda mais delicado, tendo em vista a existência de vários navegadores, os quais são os recipientes para a execução da aplicação cliente. Desta forma, o aplicativo deve ser desenvolvido com suporte necessário aos navegadores distintos, que nem sempre seguem os padrões determinados pela W3C.

O desenvolvimento de uma aplicação com AJAX entretanto também tem seus pontos positivos. No protótipo proposto e também no decorrer deste artigo pôde-se observar as facilidades e novas concepções que o uso dessa técnica pode trazer para o ambiente Web. A possibilidade de se fazer requisições assíncronas ao servidor e então incorporar os dados obtidos em partes da aplicação, é algo que abre as portas para o encontro da interatividade na Web.

Seguindo esta tendência, o protótipo de Pregão Eletrônico desenvolvido utilizando conceitos de *e-procurement* para a UNESCO, pode se tornar uma boa ferramenta para a realização de tomadas de preços da instituição ao final de sua implementação, agregando o que há de mais novo em relação ao desenvolvimento e interatividade na Web.

Com o estudo e a implementação do trabalho proposto, pôde-se aprender sobre as reais dificuldades e vantagens de se implementar uma aplicação Web utilizando AJAX, aprimorando assim, o conhecimento sobre o desenvolvimento de aplicações Web interativas. Além disto, conceitos de negócios aplicados no meio eletrônico, também foram assimilados, podendo-se perceber o quanto a Internet e as aplicações Web podem contribuir para melhorar as relações comerciais entre empresas, sobre tudo, com aplicações Web mais iterativas.

Referências

- ANSELMO, Fernando. **PHP e MySQL para Windows**. São Paulo: Visual Books, 2000.
- BORBA, Fernando Emanuel. **AJAX: Guia de programação**. São Paulo: Érica, 2006.
- CRANE, Dave; PASCARELLO, Eric; JAMES, Darren. **Ajax em ação**. São Paulo: Pearson Prentice Hall, 2006.
- FRANCO JUNIOR, Carlos F. **e-Business: tecnologia de informação e negócios na Internet**. São Paulo: Atlas, 2001.
- GONÇALVES, Edson. **Ajax na prática**. Rio de Janeiro: Ed. Ciência Moderna, 2007.
- GONÇALVES, Edson. **Dominando Ajax**. Rio de Janeiro: Ed. Ciência Moderna, 2006.
- GRAHAM, Ian S. **HTML: a referência completa para HTML 3.2 e das extensões HTML**. Rio de Janeiro: Ed. Campus, 1998.
- JQUERY. ***jQuery is a new type of JavaScript library***. Disponível em: <<http://jquery.com/>>. Acesso em: 13 setembro 2007.
- MYSQL AB. ***MySQL 5.1 Reference Manual***. Disponível em: <<http://dev.mysql.com/doc/refman/5.1/en/>>. Acesso em: 20 maio 2007.
- SOARES, Wallace. **AJAX (Asynchronous Javascript and XML): Guia prático para Windows**. 2. ed. São Paulo: Érica, 2006.
- TEIXEIRA, Flávia Silva. **Verificação do entendimento e da utilização do e-procurement em duas empresas globalizadas**. 2003. 66 f. Monografia - ECASE, Universidade de Taubaté, Taubaté, 2003.