

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA  
BACHARELADO EM CIÊNCIAS DA COMPUTAÇÃO

Desenvolvimento de um Sistema para Gerenciamento de Eventos *On-Line*

Rhodney Silva Ramos

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Bacharel em Ciências da Computação.

Florianópolis - SC

Julho / 2009

Desenvolvimento de um Sistema para Gerenciamento de Eventos *On-Line*

Trabalho de Conclusão de Curso pela Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Bacharel em Ciências da Computação.

Florianópolis - SC

Julho / 2009

---

**Professora Doutora Lúcia Helena Martins Pacheco**

Orientadora

---

**Professor Doutor Vitório Bruno Mazzola**

Banca Avaliadora

---

**João Bosco A. P. Filho**

Banca Avaliadora

## RESUMO

Este Trabalho de Conclusão de Curso apresenta os processos de análise e implementação de um Sistema para Gerenciamento de Eventos (Congressos, Simpósios, *Workshop*) via *WEB*.

Espera-se ter ao final deste processo uma ferramenta funcional e útil nos processos de organização de eventos, no que diz respeito às tarefas que antecedem seu início, como inscrições de participantes e trabalhos; às que ocorrem durante a sua realização, como avaliações e emissão de pareceres; e às que ocorrem após sua conclusão, como emissão de certificados e relatórios.

**Palavras -chave:** Eventos, PHP, SQL, Internacionalização de Software, HTTPS, ADODB

## **ABSTRACT**

This work shows the analysis and implementation processes of a Web Events Manager System.

At the end of this process, it is expected to have a functional tool to help in the events organization activities, with respect to the tasks before its beginning, as registration of participants and papers; to the ones that occur during its realization, as assessments and issue of opinions; and those that occur after its completion, as issuance of certificates and reports.

**Keywords:** Events, PHP, SQL, Software Internacionalization, HTTPS, ADODB

# Índice

<b>1. Introdução.....</b>	<b>p.6</b>
1.1 Justificava .....	p.7
1.2 Objetivos .....	p.7
<b>2. Sistemas de Gerenciamento de Eventos.....</b>	<b>p.8</b>
<b>3. Método de Desenvolvimento .....</b>	<b>p.10</b>
3.1 Desenvolvimento Ágil .....	p.10
3.2 Metodologia XP .....	p.11
3.3 Aplicação da XP neste trabalho .....	p.12
<b>4. Tecnologias de Desenvolvimento .....</b>	<b>p.14</b>
4.1. HTML .....	p.14
4.2. CSS .....	p.14
4.3 Páginas Dinâmicas .....	p.15
4.4 Linguagem de Programação: <i>PHP</i> .....	p.16
4.5 Bancos de Dados .....	p.17
4.5.1 ODBC.....	p.17
4.5.2 MySql .....	p.18
4.6 Servidor: Apache .....	p.18
4.6.1 WAMP.....	p.19
4.7 Segurança.....	p.19
4.7.1 HTTPS.....	p.20
<b>5. Internacionalização e Localização de Software .....</b>	<b>p.21</b>
<b>5.1 Internacionalizando a Aplicação.....</b>	<b>p.22</b>
<b>6. Modelagem da Aplicação .....</b>	<b>p.25</b>
6.1 Atores .....	p.25
6.2 Casos de Uso .....	p.26
6.3 Diagramas de Atividades .....	p.27
6.4 Modelagem do Banco de Dados .....	p.28
<b>7. Conclusão e Trabalhos Futuros.....</b>	<b>p.32</b>

## **1. INTRODUÇÃO**

Uma questão importante na decisão por iniciar este trabalho foi a constatação de uma demanda, no meio acadêmico, não atendida pelos Sistemas de Gerenciamento de Eventos atualmente disponíveis.

Dentre os sistemas atualmente disponíveis, encontram-se aqueles ligados a entidades específicas, utilizados para atender suas próprias necessidades. Assim, quem necessita desse tipo de serviço se vê desprovido de uma boa referência com características de fácil adaptação; ou seja, um sistema não direcionado a qualquer grupo de interesse específico, e que forneça uma estrutura adaptável de maneira simples a qualquer tipo de necessidade.

Uma segunda opção, são as ferramentas oferecidas comercialmente por empresas especializadas. Neste caso, pode-se contratar toda a estrutura necessária, bem como o suporte técnico que estas atividades demandam. Mesmo sendo uma boa opção, em alguns casos seus custos podem ser proibitivos, considerando-se que muitos dos possíveis clientes estarão envolvidos em eventos de pequeno porte, sem grandes pretensões de arrecadação.

## 1.1 JUSTIFICATIVA

Durante as pesquisas sobre este assunto foi possível observar que algumas universidades - *citadas no próximo capítulo* - dispõem de sistemas próprios para a realização dos eventos promovidos por seus departamentos e entidades a elas relacionadas.

De forma semelhante, a ferramenta aqui proposta poderia se estabelecer como solução para o gerenciamento dos eventos organizados pelos departamentos da UFSC.

## 1.2 OBJETIVOS

O objetivo deste trabalho é desenvolver e disponibilizar um *software* de funcionamento *on-line*, flexível e capaz de atender com a mesma eficiência às necessidades dos diversos tipos de usuários, independente de qual seja sua área de atuação.

Deverá apresentar as seguintes características:

**Flexibilidade:** o sistema poderá ser adaptado em sua interface exibindo a identidade visual determinada por seus utilizadores, bem como em questões mais estruturais como critérios de avaliação, tipos de usuários ou módulos a serem

utilizados.

**Praticidade:** espera-se automatizar tarefas repetitivas e tornar as tarefas de coleta e manipulação das informações uma atividade simples.

**Eficácia:** deverá garantir que as informações coletadas, sejam conduzidas corretamente do início ao fim de um evento, cumprindo todas as expectativas de seus usuários.

**Simplicidade:** deverá ser simples o bastante para ser manuseada por qualquer tipo de usuário, tendo ou não conhecimentos técnicos sobre programação ou bancos de dados. A partir de um tutorial simples e de um processo automatizado de instalação, poderá ser implementada em qualquer servidor disponível sem grandes dificuldades.

**Segurança:** deverá garantir sobretudo, a segurança das informações que serão coletadas. Dados pessoais, bancários e os trabalhos enviados deverão ser manipulados sob canais seguros, criptografados, e com as devidas restrições de acesso.

## 2 SISTEMAS DE GERENCIAMENTO DE EVENTOS

Um Sistema de Gerenciamento de Eventos é um programa de funcionamento *on-line* usado para facilitar o trabalho de organização de eventos como Congressos, Simpósios, *Workshops*, sejam eles esportivos, sociais, culturais ou científicos. São utilizados desde a fase de captação de inscrições e trabalhos até as etapas finais com a emissão de certificados, apresentação de

estatísticas e emissão de relatórios.

Existem vários sistemas desse tipo disponíveis na *Internet*. Alguns são oferecidos por empresas que alugam seus sistemas e oferecem todo o suporte que seja necessário. Um exemplo desta categoria é *Syge*, que pode ser encontrado no endereço <http://www.syge.com.br>.

Outras opções são aquelas mantidas por entidades que tem um foco de aplicação específico. Neste caso pode-se citar é o FLISYS, empregado durante no Festival Latino-americano de Instalação de *Software* Livre – FLISOL. [<http://sourceforge.net/projects/flisys/>]

Algumas Universidades também mantêm este tipo de serviço para dar suporte aos eventos realizados em seus espaços.

A FCA (Faculdade de Ciências Agrônomicas) da Universidade Estadual Paulista (Unesp) dispõe do seu próprio sistema para gerenciar os eventos que organiza. [<http://www.eventos.fca.unesp.br/>]

Na UFSC, o departamento de Engenharia Civil conta o *InfoHAB-Evento*. [[http://www.infohab.org.br/gestao\\_veja.aspx](http://www.infohab.org.br/gestao_veja.aspx)]

Fora das Universidades, mas não muito longe delas, entidades como as Sociedades Brasileiras de Química e Física, SBQ e SBF, respectivamente, também dispõe de estrutura própria para dar suporte às atividades que desenvolvem. [[www.s bq.org](http://www.s bq.org)]

### 3. MÉTODO DE DESENVOLVIMENTO

#### 3.1 Desenvolvimento Ágil

##### Manifesto Ágil

*“Estamos descobrindo maneiras melhores de desenvolver software fazendo-o nós mesmos e ajudando outros a fazê-lo. por meio desse trabalho, passamos a valorizar:*

*Indivíduos e interação entre eles mais que processos e ferramentas;*

*Software em funcionamento mais que documentação abrangente;*

*Colaboração com o cliente mais que negociação de contratos;*

*Responder a mudanças mais que seguir um plano.*

*Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda.”* [<http://www.manifestoagil.com.br>]

O Manifesto Ágil, publicado em 2001, descreve um conjunto de princípios que fundamentam o Desenvolvimento Ágil de sistemas informatizados (programas de computador).

O Desenvolvimento Ágil consiste em um conjunto de metodologias de cujo objetivo é melhorar o processo de desenvolvimento de *software*. [HIGHSMITH]

São vários os métodos de desenvolvimento ágil de software. A maioria

deles busca a minimização de riscos, sobretudo no que diz respeito aos custos envolvidos quando há a necessidade de alterações em um projeto. Isto é alcançado pela divisão do projeto em partes menores, conhecidas como iterações, cada qual compreendendo todas as etapas de desenvolvimento. A conclusão de cada iteração traz uma funcionalidade nova ao projeto, oferecendo flexibilidade e agilidade na correção de erros e redirecionamentos à medida que eles apareçam. [TELES, 2004]

Dentre as metodologias ágeis uma das mais conhecidas é a *Extreme Programming – XP*. [TELES, 2004]

### **3.2 Extreme Programming – XP**

A Programação Extrema é baseada em alguns valores, princípios e várias práticas que fazem com que ela seja uma solução muito eficiente para projetos de curto e médio prazos e que envolvam equipes pequenas de programadores. [TELES, 2004]

De maneira geral, a XP baseia-se muito na **Comunicação** entre os membros da equipe e desta com o cliente. Desta forma, cada nova funcionalidade apresentada pode ser avaliada, aprovada ou reprovada a tempo de se fazer as alterações necessárias sem que isto implique em grandes custos extras; na **Simplicidade**, para evitar excessos e a criação de algo além do realmente necessário; e na **Resposta** rápida às decisões tomadas.

A aplicação destes princípios é garantida pelo conjunto de práticas seguidas pela XP.

Há sempre a fase de **Planejamento**, quando os desenvolvedores se reúnem com o cliente e determinam quais as necessidades são prioritárias. Em seguida, os trabalhos são desenvolvidos em **Fases Pequenas** que sempre trazem funcionalidades novas ao sistema. Preza-se muito pelo **Design Simples**, de modo que os projetos atendam a cada passo somente o que foi pedido. A cada passo que se avança devem ser feitos os **Testes**, que devem ser escritos de preferência antes do desenvolvimento e sempre devem ser executados de forma automatizada. a **Refatoração** também é bastante aplicada para deixar o código mais fácil de entender e ser mantido. Em XP é comum a **Programação Pareada**; quer dizer que sempre há dois programadores trabalhando juntos no mesmo computador. Assim, enquanto um programador digita, o outro observa e pensa em melhorias. A **Propriedade** do código é **Coletiva**, todos os programadores podem mexer em todo o código. A cada nova funcionalidade ocorre a sincronização entre os desenvolvedores e esta **Integração** é **Contínua**. Sempre deve-se respeitar os limites dos programadores, evitando-se horas extras de trabalho e o **Cliente está Sempre Presente**, pronto para tirar as dúvidas dos desenvolvedores. [TELES, 2004]

### 3.3 Aplicação da XP neste trabalho

A escolha da XP para modelagem deste TCC está diretamente ligada ao formato desse tipo de trabalho, que exige bastante agilidade em seus processos de elaboração e implementação.

Inicialmente foi realizada uma análise geral e o levantamento dos requisitos

mínimos necessários para a obtenção de uma primeira versão funcional.

O resultado desta análise foi dividido em pequenas etapas implementadas e testadas individualmente. Desta forma, a cada novo ciclo de implementação foi gerado um novo programa que, embora ainda incompleto, apresentava uma nova funcionalidade.

Esta divisão foi feita de acordo com o tipo de usuário previsto no sistema: Autor, Avaliador, Coordenador e Administrador.

Na primeira etapa foi implementada a interface do Autor, para a qual foram previstas as atividades de Cadastro (comum a todos os tipos de usuário), Envio de Artigo e Atualização de Informações. A conclusão de cada uma destas funções marcava o fim de um pequeno ciclo.

A seguir foi programada a interface do Avaliador, usuário de relação direta com o Autor. A atividade básica prevista aqui é a Avaliação de Artigos.

As atividades previstas para o Coordenador são a Distribuição de Artigos para avaliação, Escolha de Avaliadores e Criação de Grupos Temáticos.

O Administrador não tem tarefas específicas dentro do Evento. É o responsável pela configuração do Sistema e pelo suporte técnico aos usuários.

Assim, trabalhando-se no desenvolvimento de pequenos módulos foi possível manter um sistema sempre funcional e ao mesmo tempo passível de receber a qualquer momento uma nova funcionalidade.

## 4. TECNOLOGIAS DE DESENVOLVIMENTO

### 4.1 HTML

HTML é uma linguagem de marcação utilizada para criar páginas estáticas na *WWW*. É o resultado da união de duas outras tecnologias:

*HyTime*, um padrão para representação estruturada de hipermídia e conteúdo baseado em tempo. Significa que um documento é visto como um conjunto de eventos que dependem do tempo - áudio e vídeo, por exemplo - conectados por hiper-ligações. [FREEMAN, 2006]

SGML é um padrão de formatação de textos. Apesar de não ter sido desenvolvido para hipertexto, tornou-se conveniente para transformar documentos em hiper-objetos e para descrever as ligações.

HTML consiste de uma coleção de etiquetas (*TAGS*) que descrevem a forma como o documento será exibido quando interpretado pelo navegador.

### 4.2 CSS

CSS é uma linguagem de estilos usada na definição de *layouts* de

documentos HTML. Com CSS pode-se controlar fontes, cores, margens, linhas, alturas, larguras, imagens de fundo, posicionamentos, enfim, tudo que esteja relacionado com a disposição dos elementos de uma interface.

O uso de CSS confere maior precisão e sofisticação às páginas HTML e é suportado por todos os navegadores atuais.

### **4.3 Páginas Dinâmicas**

Páginas HTML sozinhas não fornecem os requisitos necessários ao desenvolvimento de aplicações complexas. Sobretudo quando é necessária a integração com o usuário é fundamental o uso de páginas dinâmicas.

Este dinamismo pode ser conseguido com a introdução de *JavaScript*, uma linguagem interpretada que roda no lado do cliente, portanto dependente dos recursos oferecidos pelo navegador; ou pela inserção no HTML de alguma linguagem de programação interpretada e executada pelo servidor. Esta última forma é especialmente útil em situações que necessitam acessar informações centralizadas, como bancos de dados no servidor, ou quando por razões de segurança e praticidade, a exibição de informações depende das permissões garantidas a quem as solicite ou da sua necessidade num dado momento. [FREEMAN, 2006]

## 4.4 Linguagem de Programação: PHP 5

PHP é uma linguagem de programação orientada a objetos, executada no lado do servidor, de código aberto muito utilizada no desenvolvimento de conteúdo dinâmico *World Wide Web*.

Trata-se de uma linguagem altamente modularizada, o que permite a construção de grandes programas por meio de sua divisão em etapas menores, chamadas de módulos. Isto a torna ideal para uso em servidores *Web* já que diversos módulos são criados no repositório de extensões PECL (*PHP Extension Community Library*) sendo que alguns destes módulos podem ser utilizados em novas versões da linguagem.

No que diz respeito à sintaxe, tipos de dados e definição de funções, PHP se parece bastante com as linguagens C e C++. Em geral, aparece inserida em arquivos HTML no ponto onde se necessita executar alguma ação, como um acesso a banco de dados, por exemplo. O servidor recebe a requisição e devolve ao navegador uma página com os resultados, delimitados por marcadores HTML.

PHP provê suporte a um grande número de bancos de dados: *Oracle*, *Sybase*, *PostgreSQL*, *InterBase*, *MySQL*, *SQLite*, *MSSQL*, *Firebird*, sendo possível também abstrair o banco com a biblioteca *ADODB*.

É uma linguagem bastante veloz e robusta; pode ser modelada de maneira estruturada ou por orientação a objetos; é uma linguagem portátil e de tipagem fraca.[FREEMAN, 2006]

## 4.5 Banco De Dados

### 4.5.1 ODBC

*ODBC - Open Data Base Connectivity* - é um padrão para acesso a sistemas gerenciadores de bancos de dados. Utilizado em conjunto com alguma linguagem de programação, neste caso PHP, permite o acesso a uma vasta gama de bases de dados distintas sem que seja necessária a codificação dos métodos de acesso de forma diferente para cada uma delas.

Por meio de uma ODBC, e de sua implementação específica da linguagem SQL, a aplicação pode comunicar-se com a base de dados de forma transparente, permitindo que um mesmo programa possa utilizar simultaneamente o *MySQL*, o *Access* e o *SQL Server*, *PostGre*, *Oracle* sem a necessidade de quaisquer tipos de mudanças na sua camada de dados. Para isto, basta que estejam presentes os *drivers* ODBC específicos para as bases de dados que se deseja acessar.[GONZAGA, 2006]

A opção pelo uso da ODBC justifica-se pela intenção de manter este *software* flexível quando ao tipo de banco de dados escolhido pelo usuário final, deixando-o livre para usar o sistema que mais lhe agrade.

### **4. 5. 2 MySql**

Mesmo com a aplicação do controlador ODBC, a implementação inicial do banco de dados neste sistema foi feita com o *MySQL*.

O *MySql* é um Sistema de Gerenciamento de Banco de Dados (SGBD) de grande popularidade que, assim como o PHP, tem seu código livre. É um sistema de excelentes desempenho e estabilidade, de fácil uso e que demanda poucos recursos de *hardware*.

É um concorrente direto para os maiores sistemas de banco de dados existentes para aplicações de pequeno e médio porte. Uma das grandes vantagens do *MySQL* é suportar mais de uma dúzia de tipos de dados. Desta forma, o *MySQL* é altamente recomendado para aqueles que irão desenvolver aplicações de pequeno e médio porte e querem bons resultados a um custo reduzido.[GONZAGA, 2006].

### **4. 6 Servidor**

Um servidor *Web* é o programa responsável por processar as solicitações HTTP (*Hyper-Text Transfer Protocol*), o protocolo padrão da *Web*. Quando se usa um navegador de *Internet* para acessar um *site*, as solicitações são feitas ao

servidor *Web* do *site* , que as processa e retorna o conteúdo correspondente.[\[http://pt.wikipedia.org/wiki/Servidor\]](http://pt.wikipedia.org/wiki/Servidor)

O servidor *Web* utilizado na implementação deste trabalho foi o Apache 2.2.11.

#### **4.6.1 WAMP**

Existem, atualmente, alguns pacotes de instalação que, em poucos segundos, preparam um computador com todos os recursos necessários ao desenvolvimento de sistemas *Web* dinâmicos aqui citados.

Com o *WAMP* servidor Apache, o SGBD *MySql* e todos os recursos necessários à linguagem PHP são rapidamente instalados poupando o desenvolvedor do longo trabalho que costumava ser a preparação de uma plataforma de desenvolvimento com todos estes itens.[\[http://www.criarweb.com/artigos/327.php\]](http://www.criarweb.com/artigos/327.php)

O pacote utilizado aqui foi o *WAMP5* 2.0 que inclui também alguns recursos extras muito úteis na administração dos bancos de dados.

#### **4.7 Segurança**

Uma aplicação como esta que está sendo proposta envolve um grande fluxo de informações.

Eventos movimentam dados pessoais de dezenas de participantes, recebem seus trabalhos e, em alguns casos, pode ser necessário lidar com informações bancárias como números de cartões de crédito usados no pagamento de inscrições.

É essencial, portanto, que todas as movimentações sejam feitas com segurança, de modo a evitar que qualquer tipo de informação possa ser indevidamente acessada.

#### **4. 7. 1 HTTPS**

HTTPS (*HyperText Transfer Protocol Secure*), é uma implementação do protocolo HTTP sobre uma camada SSL . Essa camada adicional permite que os dados sejam transmitidos por meio de uma conexão criptografada e que se verifique a autenticidade do servidor e do cliente por meio de certificados digitais.

O protocolo HTTPS é utilizado quando se deseja evitar que a informação transmitida entre o cliente e o servidor seja visualizada por terceiros..

*[<http://pt.wikipedia.org/wiki/HTTPS>]*

A presença do ícone de um cadeado na barra de tarefas permite ao usuário saber que está navegando em uma página construída sobre a camada SSL.

Além disso, nas URLs dos sites, após o "http" é inserida a letra "s".

Alguns navegadores mais atuais indicam um *site* seguro por meio de barras de endereço na cor verde.

## 5. Internacionalização e Localização de *Software*

Internacionalização e Localização de *Software*, são processos de adaptação de um programa de computador para uma língua e cultura diferentes daquelas onde o programa foi desenvolvido. Significa que todo o texto exposto ao usuário final, tanto de interface quanto da documentação do produto, tem que ser traduzível para a língua de cada novo local onde pretenda ser distribuído. [PRUDÊNCIO, 2005]

Mas não se trata apenas de tradução textual. O texto deve ser neutro, livre de expressões características de uma certa cultura. Além disso, aspectos como a direção da escrita e leitura do texto, ordenação alfabética, a representação de números e formatos de data e hora também devem ser cuidadosamente observados.

A Internacionalização tem como função tornar o *software* flexível e neutro em termos culturais, financeiros e legais de um dado país. Ou seja, um programa internacionalizado deve ser genérico de tal modo que seja capaz de tratar as peculiaridades de diversas línguas e culturas sem que, para isso, seja necessário realizar alterações em seu código.

A Localização cuida da adaptação do *software* a um local específico onde ele será distribuído. Esta etapa compreende todo o processo de tradução e adaptação cultural do produto ao mercado alvo.

## 5.1 Internacionalização da Aplicação

A Aplicação dos conceitos de Internacionalização e Localização neste trabalho aconteceu de uma forma relativamente simples. Durante o processo de implementação tomou-se o cuidado de não se misturar ao código-fonte qualquer tipo de texto que não fosse parte do código-fonte. Assim, todos os textos relativos à interface com o usuário foram escritos em arquivos XML separados; dentro do código colocou-se apenas as referências, em forma de variáveis, para cada fragmento de texto utilizado.

Desta forma, para cada língua em que se pretenda ver o programa traduzido, basta que se proceda à tradução destes arquivos XML, um para cada novo idioma, e a interface do sistema se apresenta totalmente traduzida sem que sejam necessárias quaisquer alterações no código. *[PRUDÊNCIO, 2005]*

O uso de arquivos XML torna o processo de tradução simples, uma vez que não é necessário que se mexa na estrutura do programa, exigindo-se apenas o conhecimento das línguas de origem e de destino; seguro, já que não havendo contato com o código, elimina-se o risco do aparecimento de erros; flexível, pois a qualquer momento pode ser introduzida uma nova linguagem de apresentação.

Claro que, a interface não é formada apenas por textos. Considera-se também, a presença de um grande número de imagens: ícones, botões, logotipos. A adaptação destes componentes é um pouco mais trabalhosa pois exigirá do

tradutor algum conhecimento em manipulação de imagens. Mas é só. Traduzidas as imagens, basta que estas sejam colocadas em uma pasta separada, em local determinado e nos padrões especificados na documentação para que a tradução de uma nova interface esteja completa.

Nas imagens abaixo, pode-se observar um exemplo do que foi dito até aqui. No fragmento de código apresentado na Figura 1, destacam-se a variável que recebe uma informação de um arquivo XML e o método de acesso a esta informação. Há também, dois arquivos XML (Figuras 2 e 3) escritos em idiomas diferentes e o resultado final para cada uma das duas leituras (Figuras 4 e 5).

**As Informações de Interface são acessadas**

```
$arqXML = new DOMDocument();
$arqXML->load('xml/texto_pt_br.xml');
$meusDados = $arqXML->getElementsByTagName( "MEUSDADOS" );
$meusArtigos = $arqXML->getElementsByTagName( "MEUSARTIGOS" );
```

```
<TBODY>
<TR>
  <TD onmousedown="initializedragie(this)" style="CURSOR:move" width="100%">
    <ILAYER width="100%">
      <LAYER onmouseout="dragswitch=0" onmouseover="dragswitch=1;dragswitch=0" style="background-color:#ffffff" face="Verdana"><STRONG><SMALL><a href="#">
        <?PHP echo $meusDados = $meusDados->item(0)->nodeValue; ?></a>
      </LAYER>
    </TD>
```

**O Conteúdo carregado é apresentado**

**Figura1:** Código de acesso (DOM e PHP) para acesso dos arquivos XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<TEXTOS>
  <MEUSDADOS>Meus Dados</MEUSDADOS>
  <MEUSARTIGOS>Meus Artigos</MEUSARTIGOS>
</TEXTOS>
```

**Informações variáveis em Português**

**Figura2:** Arquivo XML com as informações de interface em Português

```
<TEXTOS>
  <MEUSDADOS>My Personal Information</MEUSDADOS>
  <MEUSARTIGOS>My Articles List</MEUSARTIGOS>
</TEXTOS>
```

### Informações Variáveis em Inglês

Figura3: Arquivo XML com as informações de interface em Inglês

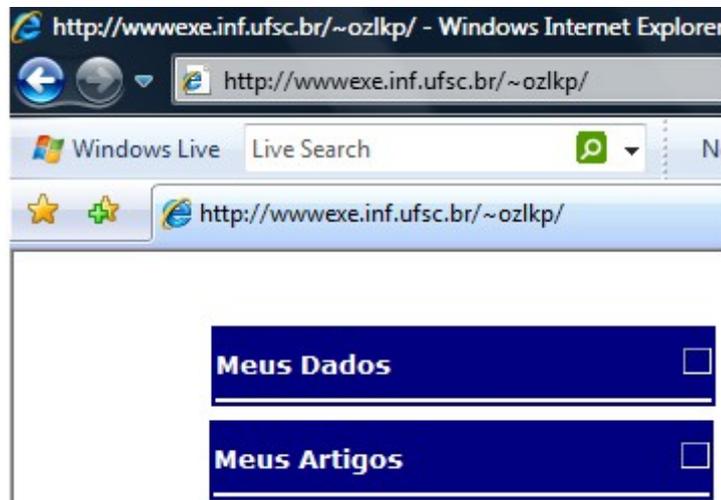


Figura4: Resultado da leitura do arquivo XML em Português

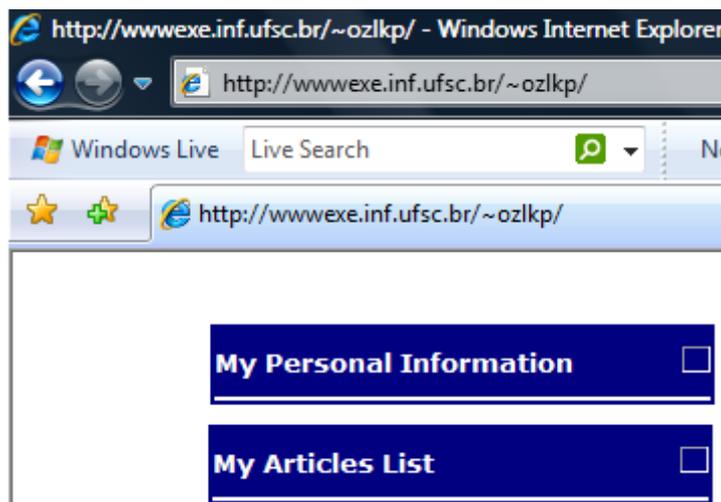


Figura5: Resultado da leitura do arquivo XML em Inglês

## 6. Modelagem

### 6.1 Atores

Para esta implementação foram previstos 5 tipos diferentes de usuários, chamados aqui de atores.

Para cada um deles foi determinado um nível de acesso ao sistema, de acordo com as atividades que execute durante um evento.

Este nível de acesso pode variar, aumentando ou diminuindo, de acordo com as necessidades observadas pelos coordenadores.

Também está previsto que novos Atores possam ser criados caso isto seja necessário.

Assim, segue abaixo uma breve descrição das atividades previstas para cada um destes Atores já implementados.

**Usuário:** Este é o primeiro nível de acesso. Todos os participantes se inscrevem como usuário. Diferentes níveis são estabelecidos posteriormente pelo coordenador. Usuários têm acesso às informações gerais de um evento, como datas e locais. Equivale a um visitante.

**Autor:** Após o pagamento da taxa de inscrição, o usuário pode enviar um artigo. Torna-se parte do ciclo ativo do evento. Além do acesso de Usuário, pode

acessar todas as informações relativas aos trabalhos que inscreveu.

**Avaliador:** É o participante responsável pela avaliação dos trabalhos submetidos. Em geral há vários avaliadores trabalhando juntos, reunidos em diversos grupos temáticos. Tem acesso aos trabalhos que avalia e aos que fazem parte de seu grupo temático.

**Coordenador:** Coordena um evento distribuindo trabalhos e nomeando os avaliadores. Também pode eleger outros coordenadores caso julgue necessário.

**Administrador:** É o responsável pelo gerenciamento do programa, não dos eventos. Cuida da preparação do arquivo de configuração, responde pelas dúvidas dos usuários e trata dos problemas técnicos que ocorram. É ele quem dá acesso ao primeiro coordenador. Tem acesso a todos os detalhes do sistema.

Além destes papéis sugeridos, o sistema permite que outros sejam criados conforme seja necessário.

## 6.2 Casos De Uso

Casos De Uso são recursos que servem para definir uma seqüência de ações executadas pelo sistema.

Cada Caso De Uso representa um objetivo específico para um Ator em particular e fornece uma nova funcionalidade ao sistema. [PRESSMAN]

Apesar de serem considerados opcionais nas Metodologia Ágeis (XP, neste caso), os Casos De Uso são bem vindos para facilitar a compreensão do comportamento do sistema. [<http://www.infoq.com/br/news/2009/02/Use-Cases-Valuable->

*But-Optional]*

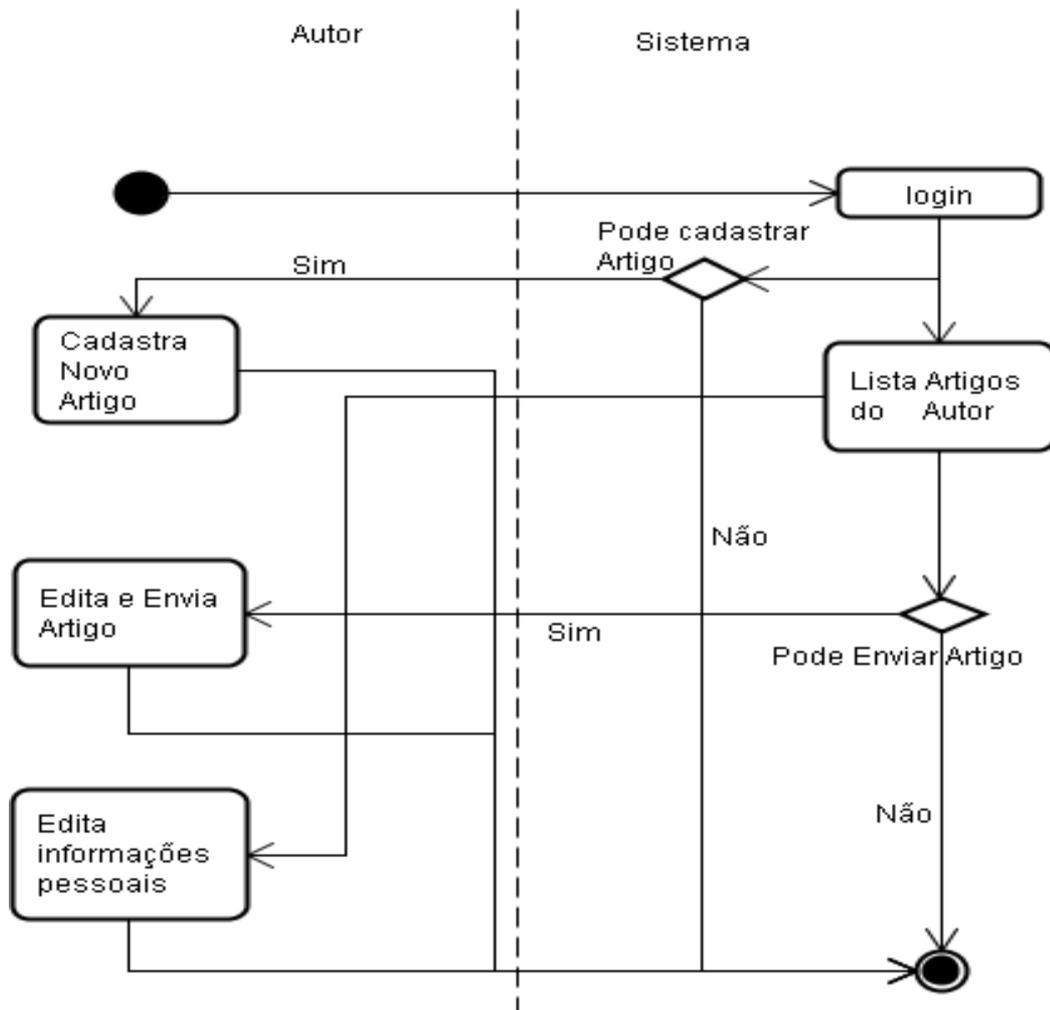
A seguir, será apresentado um caso de uso, a título de exemplo. Os demais podem ser encontrados no fim deste relatório, na seção de anexos.

<b>Caso De Uso:</b> Altera Dados Pessoais
<b>Atores:</b> Usuários
<b>Pré-Condições:</b> Estar inscrito no sistema; ter feito <i>login</i>
<b>Pós-Condições:</b> Os novos dados são inseridos no sistema
<b>Fluxo Principal:</b>  1. O usuário faz o <i>login</i>  2. Acessa o espaço para alteração dos seus dados  3. Altera o que for necessário
<b>Tratamento De Exceções:</b>  <b>3a. A conexão é perdida antes do término do processo</b>  3a.1. O usuário retoma o fluxo principal no passo 1

### 6.3 Diagramas de Atividade

O Diagrama de atividade é um diagrama definido pela Linguagem de Modelagem Unificada (UML), e representa os fluxos conduzidos por processamentos. É essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra. Em geral isso envolve a modelagem das etapas seqüenciais em um processo computacional.*[PRESSMAN]*

A seguir é apresentado o diagrama de atividades para o usuário “Autor”. Os demais diagramas podem ser encontrados na seção de Anexos.



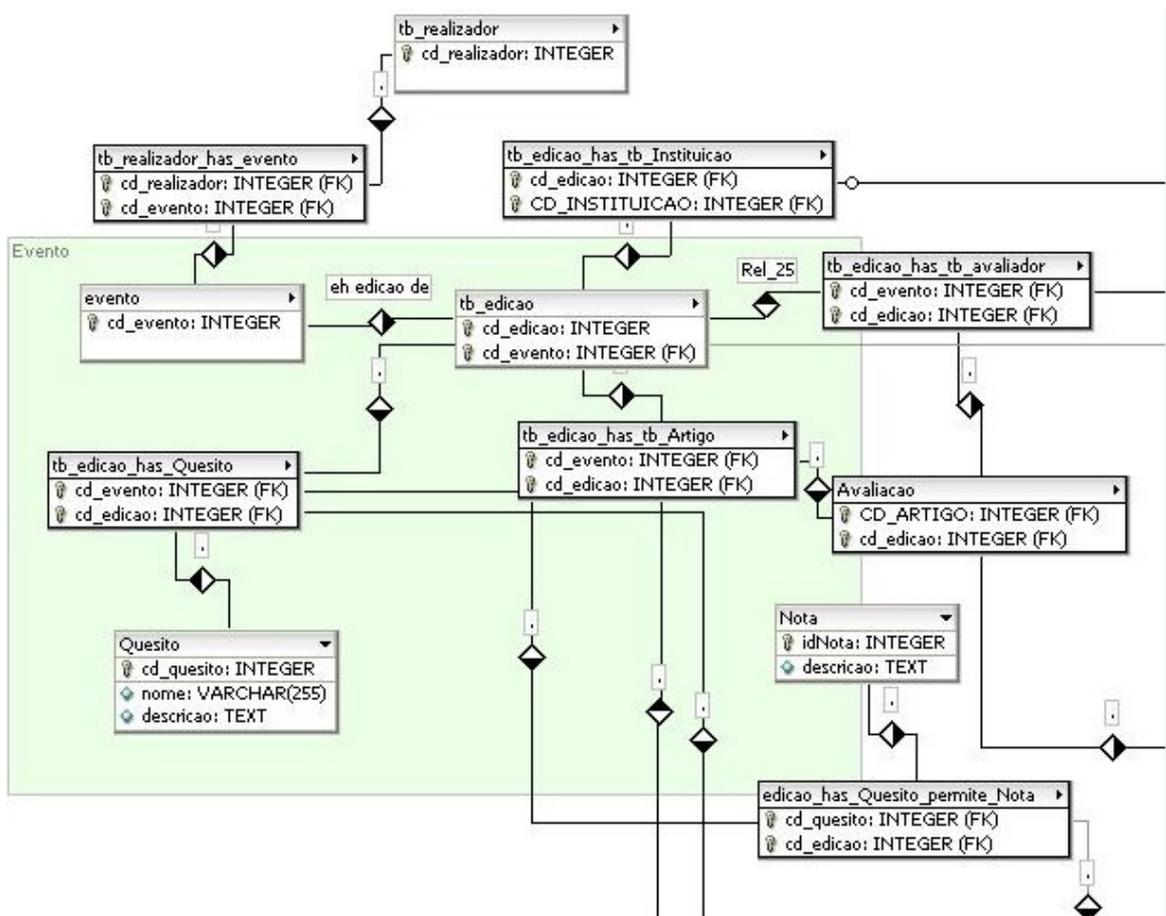
**Figura 6:** Representação das atividades do Autor

## 6.4 Banco de Dados

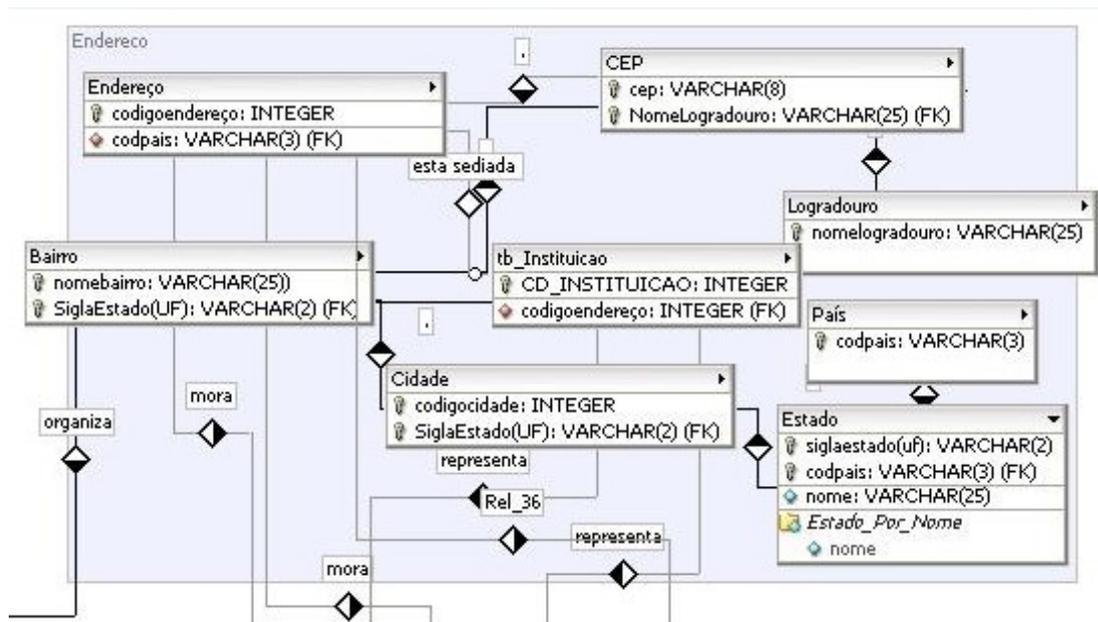
O modelo de banco de dados apresentado a seguir representa a estrutura completa implementada para este sistema.

Foi feito um agrupamento aproximado das tabelas por área, de acordo com o local ou a classe do sistema que as utiliza.

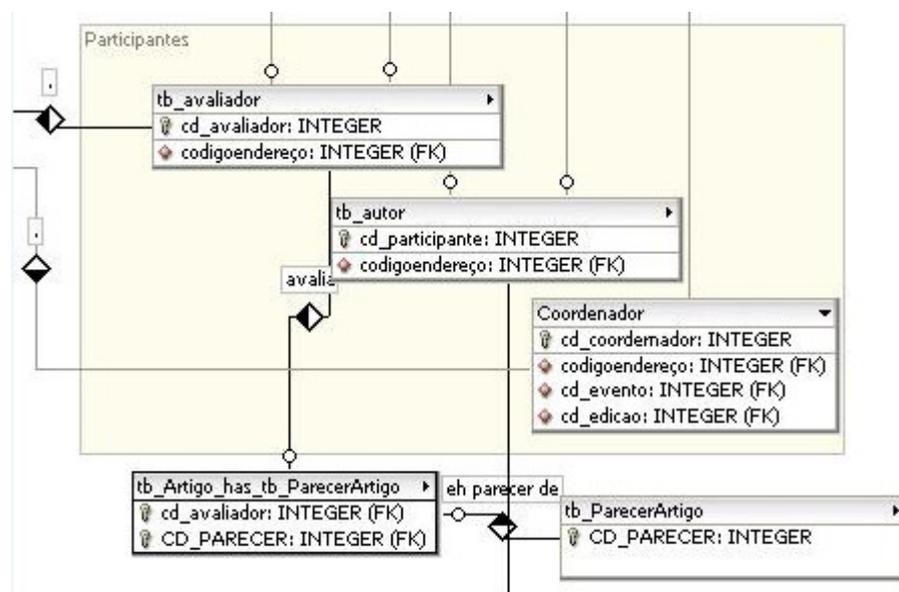
Assim, na Figura 9 está em destaque um grupo mais geral de tabelas, ligado ao evento propriamente dito. Na Figura 10 está representada a estrutura de registro de endereços dos participantes. A Figura 11 traz os Atores do evento. E, por fim, a Figura 12 mostra as tabelas que guardam as informações dos Artigos.



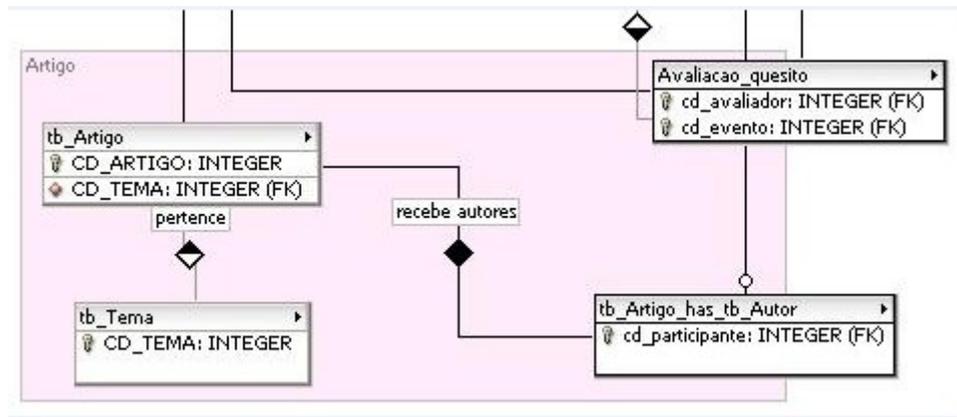
**Figura 9: Representação das tabelas ligadas ao Evento**



**Figura 10:** Tabelas que montam a estrutura de endereços



**Figura 11:** Atores principais de um Evento



**Figura 12:** Tabelas que representam um Artigo

## **7. CONCLUSÃO E SUGESTÃO PARA TRABALHOS FUTUROS**

A proposta inicial para este projeto foi realizada sem grandes problemas.

Foi gerado um sistema funcional e simples, em princípio, mas com possibilidades de ser incrementado pela adição de novos módulos.

As Atores criados e as tarefas que eles executam constituem o núcleo principal de atividades do sistema, que diz respeito ao cadastro de pessoas e trabalhos e o processo de avaliação e emissão de pareceres.

Entretanto, até que possa ser utilizado este sistema necessita ser incrementado com algumas funcionalidades essenciais, tais como os módulos de geração dos relatórios e de estatísticas. A interface do Administrador também precisa de ajustes importantes, sobretudo no que diz respeito aos recursos de personalização do sistema.

### **TRABALHOS FUTUROS**

Para o prosseguimento deste projeto, além dos ajustes citados nos parágrafos anteriores, seria interessante a implementação de novos módulos, oferecidos ao usuário de maneira a possibilitar a configuração do sistema de acordo com as suas necessidades.

Outra característica bastante interessante, relacionada às tendências atuais de comunicação, seria a possibilidade de consultas por meio de telefonia celular. Desta forma, os participantes receberiam avisos e poderiam consultar informações como pareceres e datas importantes.

A inclusão de um fórum de discussões também seria bem vinda. Neste caso, existe uma opção pronta e de código aberto chamada PHPBB [<http://www.phpbb.com/>] que pode ser incluída no sistema como um novo módulo.

### Referências

FLISYS (Laboratório de Camarões Marinhos). Disponível em <<http://sourceforge.net/projects/flisys/>>. Acesso em: 02 de maio de 2009.

Manifesto Ágil. Disponível em <http://www.manifestoagil.com.br/>. Acesso em: 05 de maio de 2009

Desenvolvimento Ágil. Disponível em <http://devagil.wordpress.com/>. Acesso em 05 de maio de 2009

TELES, Vinícius Manhães. **Extreme Programming**: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade. Novatec Editora, 2004.

[HIGHSMITH] Extreme Programming, Highsmith, J. Orr, K. Cockburn, A., Ebusiness Application Delivery, 2000

FREEMAN, Elisabeth; FREEMAN, Eric. **Use a Cabeça! (Head First)**: HTML com CSS e XHTML. 1a. Edição. 2006.

[PRESSMAN] Engenharia de Software, Pressman, Roger S., Makron Books, 1995.

GONZAGA, Flávio S.; BIRCKAN, Guilherme. Curso de PHP e MySQL. Florianópolis: 2000.

WAMP. Disponível em <http://www.criarweb.com/artigos/327.php>. Acesso em 07 de maio de 2009.

PRUDÊNCIO, Achilles Colombo; VALOIS, Djali Avelino; DE LUCCA, José Eduardo. **Introdução à Internacionalização e à Localização de Software**. Florianópolis, 2005.

Casos De Uso em XP. Disponível em:

<http://www.infoq.com/br/news/2009/02/Use-Cases-Valuable-But-Optional>

Acesso em 25 de maio de 2009

## Anexos:

### Casos de Uso:

<b>Caso De Uso:</b> Faz Inscrição
<b>Atores:</b> Usuários
<b>Objetivo:</b> Tornar-se usuário do sistema no nível inicial
<b>Pré-Condições:</b> Não ter registro anterior no sistema
<b>Pós-Condições:</b> O usuário é criado no sistema. É fornecido um <i>login</i> e uma senha.
<b>Fluxo Principal:</b> <ol style="list-style-type: none"><li>1. O interessado acessa a página geral do sistema</li><li>2. Acessa à página de inscrição.</li><li>3. Insere seus dados pessoais</li><li>4. Escolhe uma forma de pagamento da inscrição (boleto ou cartão de crédito)</li></ol>
<b>Tratamento De Exceções:</b> <b>3a. O usuário já está cadastrado</b> <ol style="list-style-type: none"><li>3a.1. O usuário é redirecionado à página inicial</li><li>3a.2. Caso não lembre seus dados, deve solicitar o reenvio do <i>login</i> e da senha</li></ol> <b>3b. A conexão é encerrada antes da conclusão da inscrição</b> <ol style="list-style-type: none"><li>3b.1. O usuário retoma o fluxo principal no passo 1</li></ol>

<b>Caso De Uso:</b> Envia Artigo
<b>Atores:</b> Usuário - Autor
<b>Pré-Condições:</b> Estar registrado no sistema Autor; ter pago a taxa de inscrição;
<b>Pós-Condições:</b> O usuário ganha acesso à área de participantes e pode acompanhar a avaliação do seu trabalho no evento
<b>Fluxo Principal:</b> <ol style="list-style-type: none"> <li>1. O Autor acessa o sistema</li> <li>2. Escolhe a opção de envio de artigo</li> <li>3. Preenche um formulário com as informações do trabalho</li> </ol>
<b>Tratamento De Exceções:</b> <ol style="list-style-type: none"> <li><b>2a. O usuário não pagou a inscrição</b> <ol style="list-style-type: none"> <li>2a.1. Deve realizar o pagamento</li> <li>2a.2. Retoma o fluxo principal no passo 3</li> </ol> </li> <li><b>2b. A inscrição já foi paga mas o sistema ainda não registrou o pagamento</b> <ol style="list-style-type: none"> <li>2b.1 Deve entrar em contato com o suporte</li> <li>2b.2 Retoma o fluxo principal no passo 3</li> </ol> </li> </ol>

<b>Caso De Uso:</b> Reenvia Artigo
<b>Atores:</b> Usuário – Autor
<b>Pré-Condições:</b> Estar registrado no sistema como Autor; O Artigo ter sido avaliado pelo menos UMA vez; O Artigo ter sido avaliado com ressalvas
<b>Pós-Condições:</b> O Artigo volta para o Avaliador
<b>Fluxo Principal:</b> <ol style="list-style-type: none"> <li>1. O Autor acessa o sistema</li> <li>2. Observa as anotações feitas pelo Autor</li> <li>3. Faz as adaptações necessárias</li> <li>4. Anexa novamente o Artigo ao Sistema</li> </ol>

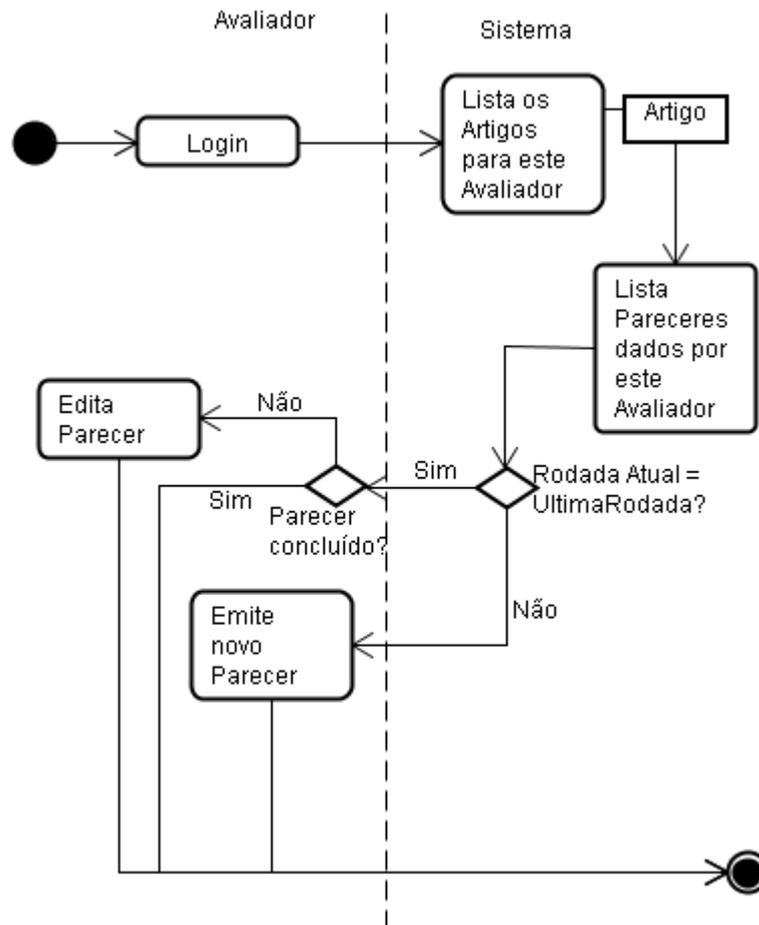
<b>Caso De Uso:</b> Avalia Artigo
<b>Atores:</b> Usuário - Avaliador
<b>Pré-Condições:</b> Estar registrado no sistema como Avaliador; Estar relacionado a um grupo temático; Ter recebido um artigo para avaliar
<b>Pós-Condições:</b> O artigo recebe uma avaliação e fica disponível para o seu autor. Caso seja aprovado com ressalvas, entra na próxima rodada de avaliações.
<b>Fluxo Principal:</b> <ol style="list-style-type: none"> <li>1. O Avaliador recebe o artigo para avaliar.</li> <li>2. Analisar o artigo</li> <li>3. Emite um parecer</li> </ol>
<b>Tratamento De Exceções:</b> <ol style="list-style-type: none"> <li>3a. A conexão é perdida <ol style="list-style-type: none"> <li>3a.1. Deve retornar ao Fluxo Principal no passo 3</li> </ol> </li> </ol>

<b>Caso De Uso:</b> Altera Nível dos Usuários
<b>Atores:</b> Usuário – Coordenador
<b>Pré-Condições:</b> Estar registrado no sistema como Coordenado
<b>Pós-Condições:</b> O nível do Usuário é alterado
<b>Fluxo Principal:</b> <ol style="list-style-type: none"> <li>1. O Coordenador acessa o sistema</li> <li>2. Escolhe um Usuário em uma lista</li> <li>3. Altera o tipo do Usuário para o nível desejado</li> </ol>

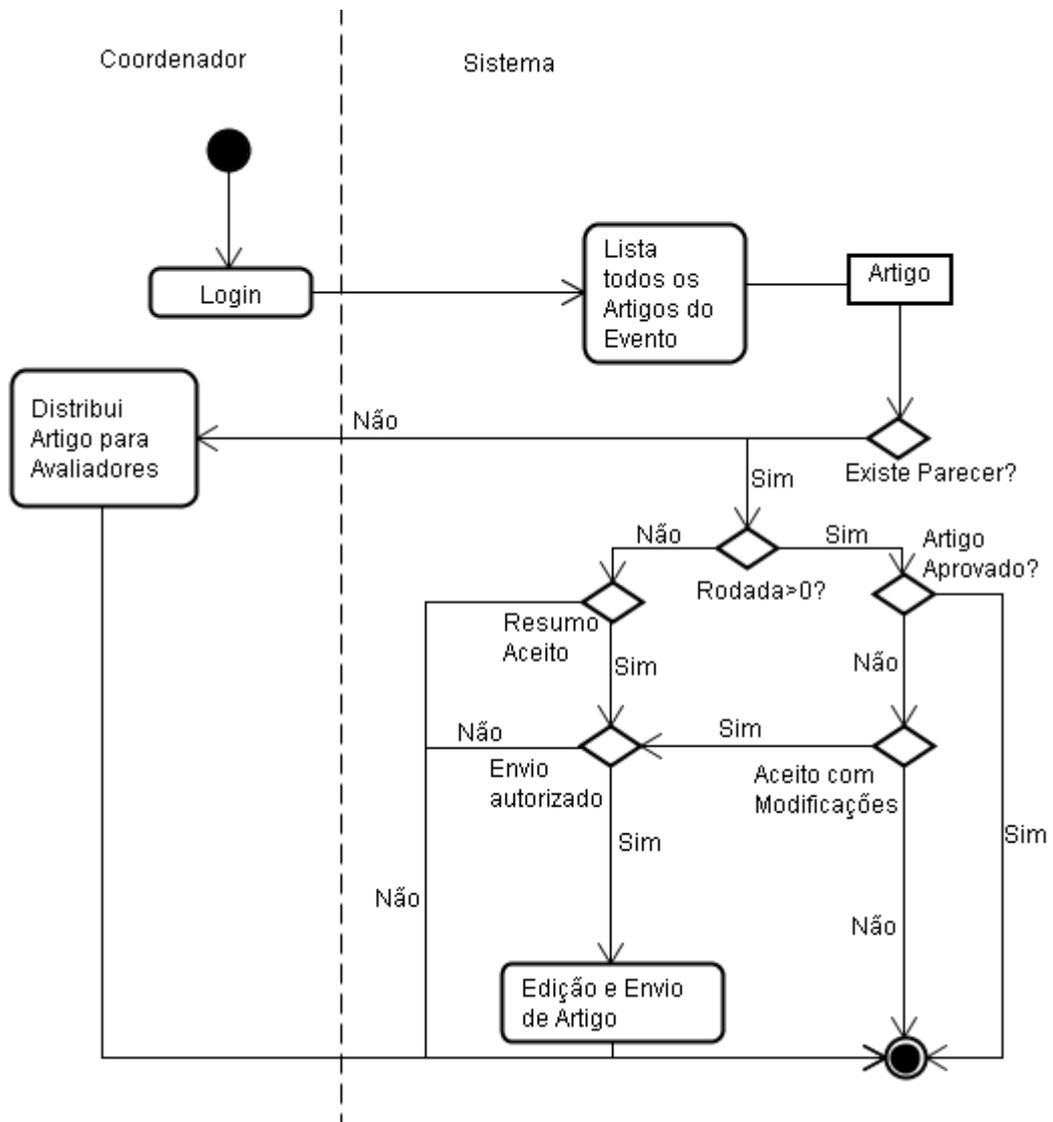
<b>Caso De Uso:</b> Cria Grupos Temáticos e associa Avaliadores
<b>Atores:</b> Usuário - Coordenador
<b>Pré-Condições:</b> Estar registrado no sistema como Coordenador
<b>Pós-Condições:</b> Novos Grupos Temáticos são criados e novos avaliadores são associados
<b>Fluxo Principal:</b> <ol style="list-style-type: none"> <li>1. O Coordenador acessa o sistema</li> <li>2. Cria um novo grupo temático</li> <li>3. Escolhe Avaliadores para participarem do grupo</li> </ol>

<b>Caso De Uso:</b> Distribui Artigos
<b>Atores:</b> Usuário – Coordenador
<b>Pré-Condições:</b> Estar registrado no sistema como Coordenador
<b>Pós-Condições:</b> Um Artigo é associado a um ou mais Avaliadores
<b>Fluxo Principal:</b> <ol style="list-style-type: none"> <li>1. O Coordenador acessa o sistema</li> <li>2. Seleciona um Artigo e associa aos Avaliadores</li> </ol>

## Diagramas de Atividades



**Figura 7:** Representação das atividades do Avaliador



**Figura 8:** Representação das atividades do Coordenador

## Códigos

busca\_artigos.php

```
<?php
header('Content-Type: text/xml');
header("Cache-Control: no-cache, must-revalidate");
//A date in the past
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");
echo "<?xml version='1.0' encoding='ISO-8859-1'\".\"?\".\">";
$cd_estado = $_GET['uf'];//RECEBE PARÂMETRO
$con = new PDO('mysql:host=localhost;dbname=evento', 'root', '');
$stmt = $con->prepare("SELECT * FROM tb_artigo WHERE cd_artigo =1");
$stmt->execute();//EXECUTA A QUERY
$qtasLinhas = $stmt->rowCount();
echo "<artigos>";
while ($rs = $stmt->fetch(PDO::FETCH_OBJ)) {
    echo "<cd_artigo>$rs->CD_ARTIGO </cd_artigo>";
    echo "<cd_tema >$rs->CD_TEMA </cd_tema>";
    echo "<titulo >$rs->TITULO </titulo>";
    echo "<resumo >$rs->RESUMO </resumo>";
    echo "<cad_data >$rs->DATA_CADASTRO</cad_data>";
}
echo "</artigos>";
?>
```

ver\_usuarios

```
<?PHP
/*Estas linha fazem com que o documento de saída seja validado como XML */
header('Content-Type: text/xml');
header("Cache-Control: no-cache, must-revalidate");
//A date in the past
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");
echo "<?xml version='1.0' encoding='ISO-8859-1'\".\"?\".\">";
```

```

$usuario = $_GET['usuario'];//RECEBE PARÂMETRO
$senha = $_GET['senha'];//RECEBE PARÂMETRO
$con = new PDO('mysql:host=localhost;dbname=evento', 'root', '');
$stmt = $con->prepare("SELECT tipo FROM tb_participante WHERE nome = '$usuario'
AND senha = '$senha'");
$stmt->execute();//EXECUTA A QUERY
$qtasLinhas = $stmt->rowCount();
echo "<usuario_tipo>";
while ($rs = $stmt->fetch(PDO::FETCH_OBJ)) {
    echo $rs->tipo;
}
echo "</usuario_tipo>";
?>

```

busca\_dados\_usuario.php

```

<?php
header('Content-Type: text/xml');
header("Cache-Control: no-cache, must-revalidate");
//A date in the past
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");
echo "<?xml version='1.0' encoding='ISO-8859-1'?.??.?>";

```

```

$cd_estado = $_GET['uf'];//RECEBE PARÂMETRO
$con = new PDO('mysql:host=localhost;dbname=evento', 'root', '');//CONECTA AO BANCO
E BUSCA AS CIDADES
$stmt = $con->prepare("SELECT * FROM tb_artigo WHERE cd_artigo =1");
$stmt->execute();//EXECUTA A QUERY
$qtasLinhas = $stmt->rowCount();//QUANTAS LINHAS FORAM ENCONTRADAS NO
BANCO
echo "<dados>";
while ($rs = $stmt->fetch(PDO::FETCH_OBJ)) {
    echo "<cd_participante>$rs->CD_ARTIGO </cd_participante>";
    echo "<codigoendereço >$rs->CD_TEMA </codigoendereço>";
    echo "<CD_INSTITUICAO >$rs->TITULO </CD_INSTITUICAO>";
    echo "<nome >$rs->RESUMO </nome>";
    echo "<ultimo_nome >$rs->DATA_CADASTRO</ultimo_nome>";
}
echo "</dados>";
?>

```

idades.php

```
<?php
// ENGINER <XML>
/*Estas linha fazem com que o documento de saída seja validado como XML */
header('Content-Type: text/xml');
header("Cache-Control: no-cache, must-revalidate");
//A date in the past
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");
echo "<?xml version='1.0' encoding='ISO-8859-1'".'?'.">";
$cd_Estado = $_GET['uf'];//RECEBE PARÂMETRO
$con = new PDO('mysql:host=localhost;dbname=evento', 'root', '');//CONECTA AO BANCO
E BUSCA AS CIDADES
$stmt = $con->prepare("SELECT cid_nom, cid_cod FROM tb_cidades WHERE uf_cod IN
(SELECT uf_cod FROM tb_uf WHERE uf_sigla='$cd_Estado')");//SELECIONA TODAS AS
CIDADES para o estado
$stmt->execute();//EXECUTA A QUERY
$qtasLinhas = $stmt->rowCount();//QUANTAS LINHAS FORAM ENCONTRADAS NO
BANCO
print "<cidades>";
while ($rs = $stmt->fetch(PDO::FETCH_OBJ)) {
//echo $rs->uf_sigla;
echo "<cidade id='".$rs->cid_cod.">$rs->cid_nom</cidade>";
$cont++;
}
print "</cidades>";
?>
```

inscricao.php

```
<HTML>
<HEAD>
<TITLE>Inscrição</TITLE>
</HEAD>
<BODY>
<?
require("camada_dados.php");
$bd = new Camada_dados();
```

```

$bd->conecta();

//VARIÁVEIS PARA O BANCO
$nome    = $_POST[nome];
$senha   = $_POST[senha];
$rua     = $_POST[rua];
$numero  = $_POST[numero];
$bairro  = $_POST[bairro];
$cidade  = $_POST[cidade];
$cep     = $_POST[cep];
$uf      = $_POST[uf];
$pais    = $_POST[pais];
$completo = $_POST[complemento];
//SEPARA AS PARTES DO NOME
$temp = split(" ", $nome);
$sultimo_nome = $temp[sizeof($temp)-1];
$bd->escreve_tb_participante($rua,$numero,$bairro,$nome,$sultimo_nome,$senha);
?>
</BODY>
</HTML>

```

instala\_bd.php

```

<?PHP
//INSTALA O BANCO DE DADOS
include("adodb/adodb.inc.php");

if($_POST[Tipo_Banco] == "Mysql")
{
    $db = NewADOConnection('mysql');
}
$con = $db->Connect("localhost", "root", "");
mysql_query("CREATE DATABASE evento");
mysql_query("USE evento");

//CRIA AS TABELAS NO BANCO DE DADOS 'evento'
$nome_do_arquivo = 'bd/cria_bd.txt';
$arquivo = Array();
$arquivo = file($nome_do_arquivo);//PEGA CADA LINHA DO ARQUIVO E JOGA NA

```

MATRIZ CRIADA ANTERIORMENTE

```
$prepara = "";  
foreach($arquivo as $v) $prepara.=$v;//PEGA CADA VALOR DA MATRIZ E CONCATENA  
NA VARIÁVEL $prepara  
$sql = explode(";", $prepara);//DIVIDE A VARIÁVEL EM VÁRIOS PEDACOS  
INDEPENDENTES DE INSTRUÇÕES SQL DO ARQUIVO SEPARADOS POR UM ';' ;'  
foreach($sql as $v) mysql_query($v);//EXECUTA CADA UMA DAS INSTRUÇÕES SQL  
//POVOA A AS TABELAS ESTADOS E CIDADES  
$nome_do_arquivo = 'bd/cidades_estados.txt';  
$arquivo = Array();  
$arquivo = file($nome_do_arquivo);//PEGA CADA LINHA DO ARQUIVO E JOGA NA  
MATRIZ CRIADA ANTERIORMENTE  
$prepara = "";  
foreach($arquivo as $v) $prepara.=$v;//PEGA CADA VALOR DA MATRIZ E CONCATENA  
NA VARIÁVEL $prepara  
$sql = explode(";", $prepara);//DIVIDE A VARIÁVEL EM VÁRIOS PEDACOS  
INDEPENDENTES DE INSTRUÇÕES SQL DO ARQUIVO SEPARADOS POR UM ';' ;'  
foreach($sql as $v) mysql_query($v);//EXECUTA CADA UMA DAS INSTRUÇÕES SQL  
?>
```

limpa\_BD.PHP

```
<?PHP  
include("adodb/adodb.inc.php");  
$db = NewADOConnection('mysql');  
$db->Connect("localhost", "root", "");  
mysql_query("USE evento");//TEM QUE ADAPTAR PARA ADO!!!!  
//CODIGO PARA EXECUTAR UM SCRIPT DE CRIAÇÃO DE BANCO DE DADOS...  
//PEGA O ARQUIVO COM O SCRIPT  
$nome_do_arquivo = 'bd/limpa_bd.txt';  
$arquivo = Array();  
//PEGA CADA LINHA DO ARQUIVO E JOGA NA MATRIZ CRIADA ANTERIORMENTE  
$arquivo = file($nome_do_arquivo);  
$prepara = "";  
//PEGA CADA VALOR DA MATRIZ E CONCATENA NA VARIÁVEL $prepara  
foreach($arquivo as $v) $prepara.=$v;  
//DIVIDE A VARIÁVEL EM VÁRIOS PEDACOS INDEPENDENTES DE INSTRUÇÕES SQL  
DO ARQUIVO SEPARADOS POR UM ';' ;'  
$sql = explode(";", $prepara);
```

```

//EXECUTA CADA UMA DAS INSTRUÇÕES SQL
foreach($sql as $v) mysql_query($v);//TEM QUE ADAPTAR PARA ADO!!!!
?>

index.php

<html>

<head>
    <script language="JavaScript" src="js/codigosJs.js"></script>
    <link rel="stylesheet" type="text/css" href="css/evento.css">
</head>

<body>
<?PHP
$config          = simplexml_load_file("xml/config.xml");//INVOCA O ARQUIVO DE
CONFIGURAÇÕES
$dir_raiz      = $config->dir_raiz;
$caminho_modelo = $config->modelo_dir."/". $config->modelo_nom;//MONTA O CAMINHO
PARA O MODELO DO BANCO DE DADOS
$caminho_cidades= $config->php_dir."/cidades.php";//ARQUIVO QUE GERA XML COM
AS CIDADES PARA UM ESTADO4
require($caminho_modelo);
$dbd          = new Camada_dados();
$dbd->conecta();
$lista_estados = $dbd->busca_estados();//PARA POVOAR A LISTA DE ESTADOS
?>
<!--FORMULARIO DE LOGIN-->
<div class="inscricao">
<form name="login">
    Usuario: <input type="text" id="usuario" name="usuario" value="jack pipo pupo"/>
<br>
    Senha: <input type="text" id="senha" name="senha" value="dfgdfgdfg"/>
    <input type="button" value="entrar" onclick="ver_usuario()"/>
</form>
<a href="#" onclick="mostra_form_inscr()">Inscreva-se</a>
</div>
<!--FORMULARIO DE LOGIN-->
<!--FORMULARIO DE INSCRIÇÃO-->

```

```

<div class="inscricao" id="inscricao">
<?PHP
ECHO "
<form id="\inscricao\" name="\inscricao\" method="\POST\" action="\$config-
>modelo_dir/inscricao.php\">; ?>
nome:<input type="text" name="nome" /><br>
senha:<input type="password" name="senha" /><br>
Instituição:<input type="text" name="instituicao" /><br><br>
Endereço:<br>
Rua:<input type="text" name="rua" /><br>
Numero:<input type="text" name="numero" /><br>
Bairro:<input type="text" name="bairro" /> <br>
Complemento:<input type="text" name="complemento" /> <br>
<!--FORMULARIO DE INSCRIÇÃO-->
<!--O Métodos 'ajaxComboBox' aciona os métodos AJAX-->
<?PHP ECHO
"Estado:<select name="\estados\" id="\estados\"
onChange="\ajaxComboBox('$caminho_cidades','cidades');\">
<option value='NDA'>escolha um estado</option>";

for($conta=0; $conta<25; $conta++){
    $uf_=$lista_estados[$conta][2];
    echo "<option value='$uf_'>";
    echo $lista_estados[$conta][1];
    echo "</option>";
}
?>
</select><br>
Cidade: <select name="cidades" id="cidades">
<option>--...</option>
</select><br>
CEP:<input type="text" name="cep" /> <br>
Pais:<input type="text" name="pais" />
<input type="submit" />
</form>
</div>
<body>
</html>

```

Avaliador.php

```
<html>
<head>
  <title>Avaliador</title>
  <LINK REL=StyleSheet HREF="estilo.css" TYPE="text/css">
  <script type="text/javascript" src="js/jquery.js"></script>
  <script language="javaScript" src="js/codigosJs.js"></script>
  <link rel="stylesheet" type="text/css" href="css/evento.css">
</head>
<body>
  <div class="menu" style="width: 100px; height: 55px; position:absolute; left:30; top:51;
background-color: #B0C4DE;" id="td_artigo"
  onClick="AlternarAbas('td_artigo','div_artigo')">Artigos</div>
  <div class="menu" style="width: 100px; height: 55px; position:absolute; left:132; top:51;
background-color: #B0C4DE;" id="td_consulta"
  onClick="AlternarAbas('td_consulta','div_consulta')">Meus Dados </div>
  <div class="menu" style="width: 30; height: 55px; position:absolute; left:260; top:51;"
id="td_manutencao"
  onClick="AlternarAbas('td_manutencao','div_manutencao')"></div>
  <div class="" style="width: 250; height: 250px; position:absolute; left:830; top:60;">Lúcia Helena
Pacheco - Avaliadora</div>
  <div id="div_artigo" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
  <div STYLE="position: absolute; top: 30px; left: 10px; width: 900px; background-color:
#FFA500; ">
  <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font color="white">-
</font><br></div>
  <div id="" STYLE="position: absolute; top: 40px; left: 10px; width: 900px;">
  <p><b>Autor:</b> Rhodney Silva Ramos</p>

  <br><b>Resumo:</b> <br>
  <br>
  <b>Palavras-Chave:</b> Eventos, PhP, SQL, Internacionalização de Software, HTTPS,
ADODB <br>

  <input type="Radio">Aceito |<input type="Radio">Recusado |<input type="Radio">Aceito
com Modificações <br> <br>
```

```

Observações:
<textArea rows="5" cols="100"></textArea><br>
<input type="submit" value="Avaliar"/>
</div>
<div STYLE="position: absolute; top: 360px; left: 10px; width: 900px; background-color:
#B0C4DE; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>
<div STYLE="position: absolute; top: 380px; left: 10px; width: 900px; background-color:
#FFA500; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>
<div STYLE="position: absolute; top: 400px; left: 10px; width: 900px; background-color:
#B0C4DE; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>
<div STYLE="position: absolute; top: 420px; left: 10px; width: 900px; background-color:
#FFA500; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>

</div>
<!--fim ECHO-->
</div>
<div id="div_consulta" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
    <div id="div_manutencao" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
        </div>
    </body>
</html>

Autor.php

<html>

```

```

<head>
  <title>Autor</title>
  <script type="text/javascript" src="js/jquery.js"></script>
  <script language="JavaScript" src="js/codigosJs.js"></script>
  <link rel="stylesheet" type="text/css" href="css/evento.css">
</head>
<body>
  <div class="menu" style="width: 100px; height: 55px; position:absolute; left:30; top:51;
background-color: #B0C4DE;" id="td_artigo"
  onClick="AlternarAbas('td_artigo','div_artigo')">Artigos</div>
  <div class="menu" style="width: 100px; height: 55px; position:absolute; left:132; top:51;
background-color: #B0C4DE;" id="td_consulta"
  onClick="AlternarAbas('td_consulta','div_consulta')">Meus Dados</div>
  <div class="menu" style="width: 100px; height: 55px; position:absolute; left:242; top:51;
background-color: #B0C4DE;" id="td_manutencao"
  onClick="AlternarAbas('td_manutencao','div_manutencao')">Enviar Artigo</div>
  <div class="" style="width: 250; height: 250px; position:absolute; left:870; top:60;"
id="td_manutencao"
  onClick="AlternarAbas('td_manutencao','div_manutencao')">Rhodney Silva Ramos -
Autor</div>
  <div id="div_artigo" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
  <!--<?PHP>for(){ echo="";-->
  <div STYLE="position: absolute; top: 30px; left: 10px; width: 900px; background-color:
#FFA500; ">
  <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font color="white">-
</font><br></div>
  <div id="" STYLE="position: absolute; top: 40px; left: 10px; width: 900px;">
<p><b>Autor:</b> </p>

  <br><b>Resumo:</b>.<br>
  <br>
  <b>Palavras-Chave:</b> Eventos, PhP, SQL, Internacionalização de Software, HTTPS,
ADODB
</div>
  <div STYLE="position: absolute; top: 180px; left: 10px; width: 900px; background-color:
#B0C4DE; ">
  <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>

```

```

</div>
<div STYLE="position: absolute; top: 200px; left: 10px; width: 900px; background-color:
#FFA500; ">
<b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>
<div STYLE="position: absolute; top: 220px; left: 10px; width: 900px; background-color:
#B0C4DE; ">
<b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>
<div STYLE="position: absolute; top: 240px; left: 10px; width: 900px; background-color:
#FFA500; ">
<b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>
</div>
<div id="div_consulta" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
<br>
<p>Nome:</p> <input type='text' value="Rhodney Silva Ramos"/> <br>
<p>Endereço</p>
Rua: <input type='text' value="Protenor Vidal"/>
Número: <input type='text' value="255"/>
Bairro: <input type='text' value="Pantanal"/><br>
Complemento: <input type='text' value=""/> <br><br>
Cidade <input type='text' value="Protenor Vidal, 255"/>
CEP <input type='text' value="Protenor Vidal, 255"/>
<p>País</p> <input type='text' value="Brasil"/> <br>
<p>Entidade:</p> <input type='text' value="UFSC"/> <br>
</div>
<div id="div_manutencao" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
<br>
<p>Título:</p> <input type='text'/> <br>
<p>Resumo:</p> <textArea cols="70" rows="5"></textArea>
<p>Tema:</p> <select name="temas" id="temas" onChange="">
<option value='NDA'>Temas</option>;
for($conta=0; $conta<25; $conta++){

```

```

        $uf_=$lista_temas[$conta][2];
        echo "<option value='$tema'>";
        echo $lista_temas[$conta][1];
        echo "</option>";
    }
    ?>
</select><br>
</div>
</body>
</html>

```

Administrador.php

```

<html>
<head>
    <title>Avaliador</title>
    <LINK REL=StyleSheet HREF="estilo.css" TYPE="text/css">
    <script type="text/javascript" src="js/jquery.js"></script>
    <script language="JavaScript" src="js/codigosJs.js"></script>
    <link rel="stylesheet" type="text/css" href="css/evento.css">
</head>
<body>
    <div class="menu" style="width: 100px; height: 55px; position:absolute; left:30; top:51;
background-color: #B0C4DE;" id="td_artigo"
        onClick="AlternarAbas('td_artigo','div_artigo')">Artigos</div>
    <div class="menu" style="width: 100px; height: 55px; position:absolute; left:132; top:51;
background-color: #B0C4DE;" id="td_consulta"
        onClick="AlternarAbas('td_consulta','div_consulta')">Meus Dados </div>
    <div class="menu" style="width: 30; height: 55px; position:absolute; left:260; top:51;"
id="td_manutencao"
        onClick="AlternarAbas('td_manutencao','div_manutencao')"></div>
    <div class="" style="width: 250; height: 250px; position:absolute; left:830; top:60;">Lúcia Helena
Pacheco - Avaliadora</div>
    <div id="div_artigo" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
    <div STYLE="position: absolute; top: 30px; left: 10px; width: 900px; background-color:
#FFA500; ">
        <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font color="white">-
</font><br></div>

```

```

<div id="" STYLE="position: absolute; top: 40px; left: 10px; width: 900px;">
<p><b>Autor:</b> Rhodney Silva Ramos</p>

<br><b>Resumo:</b> .<br>
    <br>
    <b>Palavras-Chave:</b> Eventos, PHP, SQL, Internacionalização de Software, HTTPS,
ADODB    <br>

    <input type="Radio">Aceito |<input type="Radio">Recusado |<input type="Radio">Aceito
com Modificações <br> <br>
    Observações:
    <textArea rows="5" cols="100"></textArea><br>
    <input type="submit" value="Avaliar"/>
</div>

<div STYLE="position: absolute; top: 360px; left: 10px; width: 900px; background-color:
#B0C4DE; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>

<div STYLE="position: absolute; top: 380px; left: 10px; width: 900px; background-color:
#FFA500; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>

<div STYLE="position: absolute; top: 400px; left: 10px; width: 900px; background-color:
#B0C4DE; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>

<div STYLE="position: absolute; top: 420px; left: 10px; width: 900px; background-color:
#FFA500; ">
    <b>Título</b>: Sistema Web Para Gerenciamento de Eventos <font
color="white">+</font><br>
</div>

</div>
<!--fim ECHO-->
</div>

<div id="div_consulta" class="conteudo" style="position:absolute; left:30; top:76;"

```

```

id="td_manutencao; style="display: none">
    <div id="div_manutencao" class="conteudo" style="position:absolute; left:30; top:76;"
id="td_manutencao; style="display: none">
    </div>
</body>
</html>

```

camada\_dados.php

```

<?PHP
class Camada_dados
{
    public $con      = null;//VAI ARMAZENAR O OBETO PDO
    public $tipo_banco = "mysql";
    public $servidor  = "localhost";
    public $usuario   = "root";
    public $senha     = "";
    public $banco     = "evento";
    public $persistent = false;

    public function Camada_dados( $persistent=false ){
        if( $persistent != false){
            $this->persistent = true;
        }
    }

    public function conecta(){
        try{
            $this->con = new PDO($this->tipo_banco." :host=".$this->servidor.";dbname=".$this->banco, $this->usuario, $this->senha,
                array( PDO::ATTR_PERSISTENT => $this->persistent ) );
            return $this->con;
        }catch ( PDOException $ex ){ echo "Erro: ".$ex->getMessage(); }
    }

    public function desconecta(){
        if( $this->con != null )
            $this->con = null;
    }
}

```

```

public function
escreve_tb_participante($rua,$numero,$bairro,$nome,$ultimo_nome,$senha){
    //OBTEM O CODIGO DO ENDERECO DO USUARIO RECEM CADASTRADO
    $stmt = $this->con->prepare("SELECT codigoendereco FROM tb_endereco where
rua='$rua' AND numero='$numero' AND bairro='$bairro'");
    $stmt->execute();//EXECUTA A QUERY

    $rs = $stmt->fetch(PDO::FETCH_OBJ);//EXTRAI O RESULTADO
    $cod= $rs->codigoendereco;//OBTEM O CODIGO

    $this->con->execute("insert into tb_participante
(codigoendereco,nome,ultimo_nome,senha,tipo)
values
('$cod','$nome','$ultimo_nome','$senha','0')");
}

public function escreve_tb_endereco($rua,$numero,$bairro,$complemento,$cep,$uf){
    $this->db->execute("insert into
tb_endereco(rua,numero,bairro,complemento,cep,uf)
values('$rua','$numero','$bairro','$complemento','$cep','$uf')");
}

public function busca_estados(){
    $stmt = $this->con->prepare("SELECT * FROM tb_uf");
    $stmt->execute();//EXECUTA A QUERY
    $qtasLinhas = $stmt->rowCount();
    $cont =0;
    while ($rs = $stmt->fetch(PDO::FETCH_OBJ)) {
        $ls[$cont][0] = $rs->uf_cod;
        $ls[$cont][1] = $rs->uf_nom;
        $ls[$cont][2] = $rs->uf_sigla;
        $cont++;
    }
    return $ls;
}

public function busca_cidades(){
    $stmt = $this->con->prepare("SELECT * FROM tb_cidades");
    $stmt->execute();//EXECUTA A QUERY

    while ($rs = $stmt->fetch(PDO::FETCH_OBJ)) {

```

```

        $ls = $ls.$rs->cid_nom."<BR>";
    }
    return $ls;
}
}
?>

```

## JAVASCRIPT

```

var HttpReq = null;
var dest_combo = null;

//Faz a verificação o usuario utilizando uma requisição ajax. Redireciona para a
//Página correta
function busca_artigos(){
    var usuario = 'rodney';
    ajax_artigos("php/busca_artigos.php?usuario="+usuario);
}
function ver_usuario(){
    var usuario = document.getElementById('usuario').value;
    var senha = document.getElementById('senha').value;
    ajax_login("php/ver_usuario.php?usuario="+usuario+"&senha="+senha);
}
//BUSCA E TRATAMENTO DOS ARTIGOS
function ajax_artigos(url){
    if (document.getElementById) { //Verifica se o Browser suporta DHTML.
        if (window.XMLHttpRequest) {
            HttpReq = new XMLHttpRequest();
            HttpReq.onreadystatechange = XMLHttpRequestChangeArtigos;
            HttpReq.open("GET", url, true);
            HttpReq.send(null);
        } else if (window.ActiveXObject) {
            HttpReq = new ActiveXObject("Microsoft.XMLHTTP");
            if (HttpReq) {
                HttpReq.onreadystatechange = XMLHttpRequestChangeArtigos;
                HttpReq.open("GET", url, true);
                HttpReq.send();
            }
        }
    }
}

```

```

    }
}
function XMLHttpRequestChangeArtigos() {
    if (HttpReq.readyState == 4 && HttpReq.status == 200){
        var result      = HttpReq.responseXML;
        exibe_artigos(result);//EXIBE OS DADOS DO ARTIGO NA TELA
    }
}
function exibe_artigos(result){
    var cd_artigo  = result.getElementsByTagName("cd_artigo")[0].childNodes[0].data;
    var titulo    = result.getElementsByTagName("titulo"  )[0].childNodes[0].data;
    var resumo    = result.getElementsByTagName("resumo"  )[0].childNodes[0].data;
    var cad_data  = result.getElementsByTagName("cad_data" )[0].childNodes[0].data;
    $(document).ready(function(){
        $('#div_artigo').append('<p>'+cd_artigo+'</p>');
        $('#div_artigo').append('<p>'+titulo  +'</p>');
        $('#div_artigo').append('<p>'+resumo  +'</p>');
        $('#div_artigo').append('<p>'+cad_data +'</p>');
    });
}
//FIM - BUSCA E TRATAMENTO DOS ARTIGOS

//VERFICCA OS DADOS DE UM LOGIN
function ajax_login(url){
    if (document.getElementById) { //Verifica se o Browser suporta DHTML.
        if (window.XMLHttpRequest) {
            HttpReq = new XMLHttpRequest();
            HttpReq.onreadystatechange = XMLHttpRequestChangeLogin;
            HttpReq.open("GET", url, true);
            HttpReq.send(null);
        } else if (window.ActiveXObject) {
            HttpReq = new ActiveXObject("Microsoft.XMLHTTP");
            if (HttpReq) {
                HttpReq.onreadystatechange = XMLHttpRequestChangeLogin;
                HttpReq.open("GET", url, true);
                HttpReq.send();
            }
        }
    }
}
}

```

```

}
function XMLHttpRequestChangeLogin() {
    if (HttpReq.readyState == 4 && HttpReq.status == 200){
        var result      = HttpReq.responseXML;
        //Caso o login seja valido, retorna o nivel do usuario
        var usuario_tipo = result.getElementsByTagName("usuario_tipo");
        var usuario_tipo = usuario_tipo[0].childNodes[0].data;//ESTA VARIABEL TEM O
NIVEL(AUTOR, AVALIADOR...)DO USUARIO
        direciona_usuario(usuario_tipo);//DIRECIONA O USUARIO PARA A INTERFACE
ADEQUADA
    }
}
//FIM - VERFICCA OS DADOS DE UM LOGIN

//DIRECIONA O USUARIO PARA SUA PAGINA
function direciona_usuario(usuario_tipo){
    switch (usuario_tipo) {
        case "0":
            window.location = "Autor.php";
            break;
        case "1":
            window.location = "Avaliador.php";
            break;
        case "2":
            window.location = "Coordenador.php";
            break;
    }
}
//AJAX PARA CRIAR XML COM AS CIDADES DE UM ESTADO; EXPLICAÇÃO DO
FUNCIONAMENTO DE UM PROCESSO
//AJAX NO FIM DESTE ARQUIVO
function ajaxComboBox(url, comboBox){
    dest_combo = comboBox;
    var indice = document.getElementById('estados').selectedIndex;
    var sigla = document.getElementById('estados').options[indice].getAttribute('value');
    url = url + '?uf=' + sigla;

    if (document.getElementById) { //Verifica se o Browser suporta DHTML.
        if (window.XMLHttpRequest) {

```

```

        HttpReq = new XMLHttpRequest();
        HttpReq.onreadystatechange = XMLHttpRequestChangeCombo;
        HttpReq.open("GET", url, true);
        HttpReq.send(null);
    } else if (window.ActiveXObject) {
        HttpReq = new ActiveXObject("Microsoft.XMLHTTP");
        if (HttpReq) {
            HttpReq.onreadystatechange = XMLHttpRequestChangeCombo;
            HttpReq.open("GET", url, true);
            HttpReq.send();
        }
    }
}

function XMLHttpRequestChangeCombo() {
    if (HttpReq.readyState == 4 && HttpReq.status == 200){
        var result = HttpReq.responseXML;
        var cidades = result.getElementsByTagName("cidade");
        var new_opcao = "";
        document.getElementById(dest_combo).innerHTML = "";
        for (var i = 0; i < cidades.length; i++) {
            new_opcao = create_opcao(cidades[i]);
            document.getElementById(dest_combo).appendChild(new_opcao);
        }
    }
}

function create_opcao(cidade) {
    var new_opcao = document.createElement("option");
    var texto = document.createTextNode(cidade.childNodes[0].data);
    new_opcao.setAttribute("value", cidade.getAttribute("id"));
    new_opcao.appendChild(texto); //Adiciona o texto a OPTION.
    return new_opcao; // Retorna a nova OPTION.
}

function mostra_form_inscr(){
    //window.showModalDialog("inscricao.php"); ABRE JANELA E BLOQUEIA A JNELA
    PAI
    var div_formulario = document.getElementById('div_form');

```

```
div_formulario.style.visibility="visible";  
}
```

//SCRIPTS PARA CONTROLE DE ABAS

```
function stAba(menu,conteudo)  
{  
    this.menu = menu;  
    this.conteudo = conteudo;  
}  
var arAbas = new Array();  
arAbas[0] = new stAba('td_artigo','div_artigo');  
arAbas[1] = new stAba('td_consulta','div_consulta');  
arAbas[2] = new stAba('td_manutencao','div_manutencao');  
  
function AlternarAbas(menu,conteudo)  
{  
    for (i=0;i<arAbas.length;i++)  
    {  
        m = document.getElementById(arAbas[i].menu);  
        m.className = 'menu';  
        c = document.getElementById(arAbas[i].conteudo)  
        c.style.display = 'none';  
    }  
    m = document.getElementById(menu)  
    m.className = 'menu-sel';  
    if(m == td_artigo)  
    {  
        busca_artigos();  
    }  
    c = document.getElementById(conteudo)  
    c.style.display = "";  
}
```

CSS

```
div.inscricao {  
    width: 470px;  
    float:left;
```

```
margin: 10px 0;
background: url(images/bg-gradiente.gif) repeat-X;
border: 1px solid #CCC;
padding: 30px;
clear:left;
}
```

```
body, table {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 10px;
    color: #000000;
}
```

```
.menu {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 10px;
    font-weight: normal;
    color: #000033;
    background-color: #FFFFFF;
    border-right: 1px solid #000000;
    border-left: 1px solid #000000;
    border-top: 1px solid #000000;
    //border-bottom: 1px solid #000000;
    padding: 5px;
    cursor: hand;
}
```

```
.menu-sel {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 10px;
    font-weight: bold;
    color: #000033;
    background-color: #CCCCCC;
    border-right: 1px solid #000000;
    border-left: 1px solid #000000;
    border-top: 1px solid #000000;
    padding: 5px;
    cursor: hand;
}
```

```
.tb-conteudo {  
    border-right: 1px solid #000000;  
    border-bottom: 1px solid #000000;  
}
```

```
.conteudo {  
    font-family: Verdana, Arial, Helvetica, sans-serif;  
    font-size: 10px;  
    font-weight: normal;  
    color: #000033;  
    background-color: white;  
    border-left: 1px solid #000000;  
    border-top: 1px solid #000000;  
    padding: 5px;  
    width: 1000px;  
    height: 460px;  
}
```

# Desenvolvimento de um Sistema para Gerenciamento de Eventos *On-Line*

Rhodney S. Ramos

Departamento de Informática e Estatística – Universidade Federal de Santa Catarina  
Caixa Postal 15.064 CEP 88040-900 Campus Universitário – Florianópolis, SC – Brasil

ozlkp@inf.ufsc.br

**Abstract.** *This work shows the analysis and implementation processes of a Web Events Manager System. At the end of this process, it is expected to have a functional tool to help in the events organization activities, with respect to the tasks before its beginning, as registration of participants and papers; to the ones that occur during its realization, as assessments and issue of opinions; and those that occur after its completion, as issuance of certificates and reports.*

**Resumo.** *Este Trabalho de Conclusão de Curso apresenta os processos de análise e implementação de um Sistema para Gerenciamento de Eventos (Congressos, Simpósios, Workshop) via WEB. Espera-se ter ao final deste processo uma ferramenta funcional e útil nos processos de organização de eventos, no que diz respeito às tarefas que antecedem seu início, como inscrições de participantes e trabalhos; às que ocorrem durante a sua realização, como avaliações e emissão de pareceres; e às que ocorrem após sua conclusão, como emissão de certificados e relatórios.*

## 1. Introdução

Um Sistema de Gerenciamento de Eventos é um programa de funcionamento on-line usado para facilitar o trabalho de organização de eventos como Congressos, Simpósios, Workshops, sejam eles esportivos, sociais, culturais ou científicos. São utilizados desde a fase de captação de inscrições e trabalhos até as etapas finais com a emissão de certificados, apresentação de estatísticas e emissão de relatórios.

Existem vários sistemas desse tipo disponíveis na Internet. Alguns são oferecidos por empresas que alugam seus sistemas e oferecem todo o suporte que seja necessário. Um exemplo desta categoria é Syge, que pode ser encontrado no endereço <http://www.syge.com.br>.

Outras opções são aquelas mantidas por entidades que tem um foco de aplicação específico. Neste caso pode-se citar é o FLISYS, empregado durante no Festival Latino-americano de Instalação de Software Livre – FLISOL. [<http://sourceforge.net/projects/flisys/>]

Algumas Universidades também mantém este tipo de serviço para dar suporte aos eventos realizados em seus espaços.

A FCA (Faculdade de Ciências Agrônomicas) da Universidade Estadual Paulista

(Unesp) dispõe do seu próprio sistema para gerenciar os eventos que organiza. [<http://www.eventos.fca.unesp.br/>]

Na UFSC, o departamento de Engenharia Civil conta o InfoHAB-Evento. [[http://www.infohab.org.br/gestao\\_veja.aspx](http://www.infohab.org.br/gestao_veja.aspx)]

Fora das Universidades, mas não muito longe delas, entidades como as Sociedades Brasileiras de Química e Física, SBQ e SBF, respectivamente, também dispõe de estrutura própria para dar suporte às atividades que desenvolvem. [[www.sbj.org](http://www.sbj.org)]

## **2. Desenvolvimento**

O desenvolvimento deste projeto foi baseado em uma abordagem dinâmica - *Extreme Programming* - visando uma melhor relação de ganho entre o tempo disponível e quantidade de esforço demandado durante este processo.

Inicialmente foi realizada uma análise geral e o levantamento dos requisitos mínimos necessários para a obtenção de uma primeira versão funcional.

O resultado desta análise foi dividido em pequenas etapas implementadas e testadas individualmente. Desta forma, a cada novo ciclo de implementação foi gerado um novo programa que, embora ainda incompleto, apresentava uma nova funcionalidade.

Esta divisão foi feita de acordo o tipo de usuário previsto no sistema: Autor, Avaliador, Coordenador e Administrador.

Na primeira etapa foi implementada a interface do Autor, para a qual foram previstas as atividades de Cadastro (comum a todos os tipos de usuário), Envio de Artigo e Atualização de Informações. A conclusão de cada uma destas funções marcava o fim de um pequeno ciclo.

A seguir foi programada a interface do Avaliador, usuário de relação direta com o Autor. A atividade básica prevista aqui é a Avaliação de Artigos.

As atividades previstas para o Coordenador são a Distribuição de Artigos para avaliação, Escolha de Avaliadores e Criação de Grupos Temáticos.

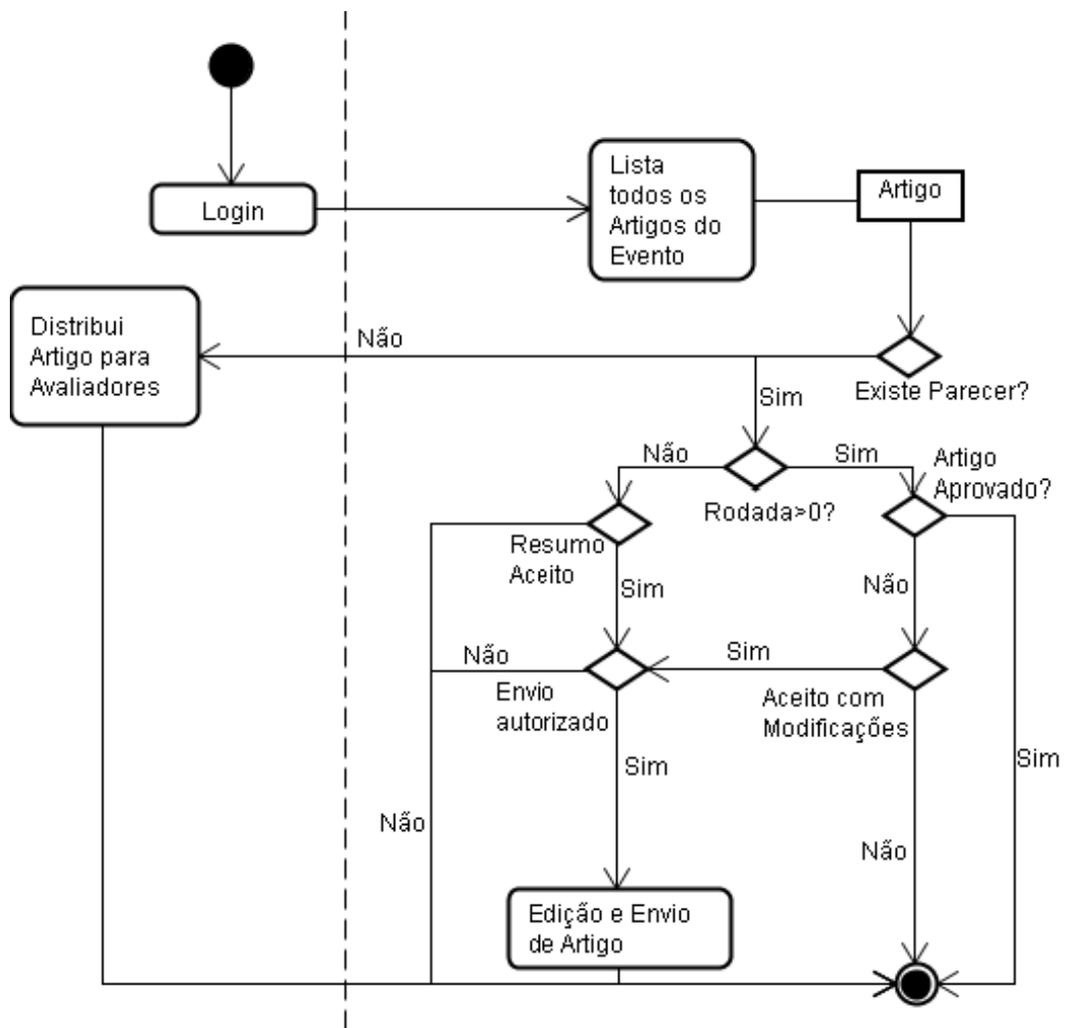
O Administrador não tem tarefas específicas dentro do Evento. É o responsável pela configuração do Sistema e pelo suporte técnico aos usuários.

Assim, trabalhando-se no desenvolvimento de pequenos módulos foi possível manter um sistema sempre funcional e ao mesmo tempo passível de receber a qualquer momento uma nova funcionalidade.

### **2.1. Funcionamento**

O sistema funciona da seguinte maneira: Os autores interessados submetem seus trabalhos, inicialmente na forma de um resumo. Este resumo é encaminhado pelo coordenador do evento a um Avaliador, por ele determinado, que decidirá se o resumo será ou não aceito. Caso o resumo seja aceito, seu Autor receberá permissão para submeter um artigo que passará por uma nova série de avaliações. A quantidade de avaliações permitidas é nefinida durante a configuração de um novo evento, de acordo com as necessidades observadas.

Este é fluxo principal do processo de captação e avaliação de documentos e pode ser observado no diagrama a seguir:



**Figura1: Estrutura Geral da Captação de Documentos**

Durante a realização das demais atividades, outras ferramentas estarão disponíveis aos seus usuários e administradores.

São estas os fóruns de debates, os módulos de geração de relatórios e de estatísticas e um painel de controle, gerenciado por um Administrador, através do qual é possível fazer as adaptações necessárias ao sistema, para que este funcione conforme as necessidades de seus organizadores.

## 2.2. Atores

Para esta implementação foram previstos 5 tipos diferentes de usuários, chamados aqui de atores, sendo que para cada um deles foi determinado um nível de acesso, de acordo com as atividades que executem durante um evento.

Este nível de acesso pode variar, aumentando ou diminuindo, conforme com as

necessidades observadas pelos coordenadores, além de ser possível a criação de novos Atores.

Assim, segue abaixo uma breve descrição das atividades previstas para cada um destes Atores já implementados.

**Usuário:** Este é o primeiro nível de acesso. Todos os participantes se inscrevem como usuário. Diferentes níveis são estabelecidos posteriormente pelo coordenador. Usuários têm acesso às informações gerais de um evento, como datas e locais. Equivale a um visitante.

**Autor:** Após o pagamento da taxa de inscrição, o usuário pode enviar um artigo. Torna-se parte do ciclo ativo do evento. Além do acesso de Usuário, pode acessar todas as informações relativas aos trabalhos que inscreveu.

**Avaliador:** É o participante responsável pela avaliação dos trabalhos submetidos. Em geral há vários avaliadores trabalhando juntos, reunidos em diversos grupos temáticos. Tem acesso aos trabalhos que avalia e aos que fazem parte de seu grupo temático.

**Coordenador:** Coordena um evento distribuindo trabalhos e nomeando os avaliadores. Também pode eleger outros coordenadores caso julgue necessário.

**Administrador:** É o responsável pelo gerenciamento do programa, não dos eventos. Cuida da preparação do arquivo de configuração, responde pelas dúvidas dos usuários e trata dos problemas técnicos que ocorram. É ele quem dá acesso ao primeiro coordenador. Tem acesso a todos os detalhes do sistema.

### 3. Tecnologias Aplicadas

#### 3.1. HTML

HTML é uma linguagem de marcação utilizada para criar páginas estáticas na *WWW*. É o resultado da união de duas outras tecnologias:

*HyTime*, um padrão para representação estruturada de hipermídia e conteúdo baseado em tempo. Significa que um documento é visto como um conjunto de eventos que dependem do tempo - áudio e vídeo, por exemplo - conectados por hiperligações. [FREEMAN, 2006]

SGML é um padrão de formatação de textos. Apesar de não ter sido desenvolvido para hipertexto, tornou-se conveniente para transformar documentos em hiper-objetos e para descrever as ligações.

HTML consiste de uma coleção de etiquetas (*TAGS*) que descrevem a forma como o documento será exibido quando interpretado pelo navegador.

#### 3.2. CSS

CSS é uma linguagem de estilos usada na definição de *layouts* de documentos HTML. Com CSS pode-se controlar fontes, cores, margens, linhas, alturas, larguras, imagens de fundo, posicionamentos, enfim, tudo que esteja relacionado com a disposição dos elementos de uma interface.

O uso de CSS confere maior precisão e sofisticação às páginas HTML e é suportado por todos os navegadores atuais.

#### 3.3. Linguagem de Programação: PHP 5

PHP é uma linguagem de programação orientada a objetos, executada no lado do servidor, de código aberto muito utilizada no desenvolvimento de conteúdo dinâmico *World Wide Web*.

Trata-se de uma linguagem altamente modularizada, o que permite a construção de grandes programas por meio de sua divisão em etapas menores, chamadas de módulos. Isto a torna ideal para uso em servidores *Web* já que diversos módulos são criados no repositório de extensões PECL (*PHP Extension Community Library*) sendo que alguns destes módulos podem ser utilizados em novas versões da linguagem.

No que diz respeito à sintaxe, tipos de dados e definição de funções, PHP se parece bastante com as linguagens C e C++. Em geral, aparece inserida em arquivos HTML no ponto onde se necessita executar alguma ação, como um acesso a banco de dados, por exemplo. O servidor recebe a requisição e devolve ao navegador uma página com os resultados, delimitados por marcadores HTML.

PHP provê suporte a um grande número de bancos de dados: *Oracle*, *Sybase*, *PostgreSQL*, *InterBase*, *MySQL*, *SQLite*, *MSSQL*, *Firebird*, sendo possível também abstrair o banco com a biblioteca *ADODB*.

É uma linguagem bastante veloz e robusta; pode ser modelada de maneira estruturada ou por orientação a objetos; é uma linguagem portátil e de tipagem fraca. [FREEMAN, 2006]

### 3.4. ODBC

*ODBC - Open Data Base Connectivity* - é um padrão para acesso a sistemas gerenciadores de bancos de dados. Utilizado em conjunto com alguma linguagem de programação, neste caso PHP, permite o acesso a uma vasta gama de bases de dados distintas sem que seja necessária a codificação dos métodos de acesso de forma diferente para cada uma delas.

Por meio de uma ODBC, e de sua implementação específica da linguagem SQL, a aplicação pode comunicar-se com a base de dados de forma transparente, permitindo que um mesmo programa possa utilizar simultaneamente o *MySQL*, o *Access* e o *SQL Server*, *PostGre*, *Oracle* sem a necessidade de quaisquer tipos de mudanças na sua camada de dados. Para isto, basta que estejam presentes os *drivers* ODBC específicos para as bases de dados que se deseja acessar. [GONZAGA, 2006]

A opção pelo uso da ODBC justifica-se pela intenção de manter este *software* flexível quando ao tipo de banco de dados escolhido pelo usuário final, deixando-o livre para usar o sistema que mais lhe agrade.

### 3.5. MySql

Mesmo com a aplicação do controlador ODBC, a implementação inicial do banco de dados neste sistema foi feita com o *MySQL*.

O *MySQL* é um Sistema de Gerenciamento de Banco de Dados (SGBD) de grande popularidade que, assim como o PHP, tem seu código livre. É um sistema de excelentes desempenho e estabilidade, de fácil uso e que demanda poucos recursos de *hardware*.

É um concorrente direto para os maiores sistemas de banco de dados existentes para aplicações de pequeno e médio porte. Uma das grandes vantagens do *MySQL* é suportar mais de uma dúzia de tipos de dados. Desta forma, o *MySQL* é altamente recomendado para aqueles que irão desenvolver aplicações de pequeno e médio porte e querem bons resultados a um custo reduzido. [GONZAGA, 2006].

### 3.6. Servidor

Um servidor *Web* é o programa responsável por processar as solicitações HTTP (*Hyper-*

*Text Transfer Protocol*), o protocolo padrão da *Web*. Quando se usa um navegador de *Internet* para acessar um *site*, as solicitações são feitas ao servidor *Web* do *site*, que as processa e retorna o conteúdo correspondente. [<http://pt.wikipedia.org/wiki/Servidor>]

O servidor *Web* utilizado na implementação deste trabalho foi o Apache 2.2.11.

### 3.7. WAMP

Existem, atualmente, alguns pacotes de instalação que, em poucos segundos, preparam um computador com todos os recursos necessários ao desenvolvimento de sistemas *Web* dinâmicos aqui citados.

Com o *WAMP* servidor Apache, o SGBD *MySql* e todos os recursos necessários à linguagem PHP são rapidamente instalados poupando o desenvolvedor do longo trabalho que costumava ser a preparação de uma plataforma de desenvolvimento com todos estes itens. [<http://www.criarweb.com/artigos/327.php>]

O pacote utilizado aqui foi o *WAMP5 2.0* que inclui também alguns recursos extras muito úteis na administração dos bancos de dados.

### 3.8. Segurança

Uma aplicação como esta que está sendo proposta envolve um grande fluxo de informações.

Eventos movimentam dados pessoais de dezenas de participantes, recebem seus trabalhos e, em alguns casos, pode ser necessário lidar com informações bancárias como números de cartões de crédito usados no pagamento de inscrições.

É essencial, portanto, que todas as movimentações sejam feitas com segurança, de modo a evitar que qualquer tipo de informação possa ser indevidamente acessada.

### 3.9. HTTPS

*HTTPS (HyperText Transfer Protocol Secure)*, é uma implementação do protocolo HTTP sobre uma camada SSL. Essa camada adicional permite que os dados sejam transmitidos por meio de uma conexão criptografada e que se verifique a autenticidade do servidor e do cliente por meio de certificados digitais.

O protocolo HTTPS é utilizado quando se deseja evitar que a informação transmitida entre o cliente e o servidor seja visualizada por terceiros.. [<http://pt.wikipedia.org/wiki/HTTPS>]

A presença do ícone de um cadeado na barra de tarefas permite ao usuário saber que está navegando em uma página construída sobre a camada SSL.

Além disso, nas URLs dos sites, após o "http" é inserida a letra "s".

Alguns navegadores mais atuais indicam um *site* seguro por meio de barras de endereço na cor verde.

## 4. Internacionalização e Localização de *Software*

Internacionalização e Localização de *Software*, são processos de adaptação de um programa de computador para uma língua e cultura diferentes daquelas onde o programa foi desenvolvido. Significa que todo o texto exposto ao usuário final, tanto de interface quanto da documentação do produto, tem que ser traduzível para a língua de cada novo local onde pretenda ser distribuído. [*PRUDÊNCIO, 2005*]

Mas não se trata apenas de tradução textual. O texto deve ser neutro, livre de expressões características de uma certa cultura. Além disso, aspectos como a direção da escrita e leitura do texto, ordenação alfabética, a representação de números e formatos de data e hora também devem ser cuidadosamente observados.

A Internacionalização tem como função tornar o *software* flexível e neutro em termos culturais, financeiros e legais de um dado país. Ou seja, um programa internacionalizado deve ser genérico de tal modo que seja capaz de tratar as peculiaridades de diversas línguas e culturas sem que, para isso, seja necessário realizar alterações em seu código.

A Localização cuida da adaptação do *software* a um local específico onde ele será distribuído. Esta etapa compreende todo o processo de tradução e adaptação cultural do produto ao mercado alvo.

## 5. Conclusão

A proposta inicial para este projeto foi realizada sem grandes problemas.

Foi gerado um sistema funcional e simples, em princípio, mas com possibilidades de ser incrementado pela adição de novos módulos.

As Atores criados e as tarefas que eles executam constituem o núcleo principal de atividades do sistema, que diz respeito ao cadastro de pessoas e trabalhos e o processo de avaliação e emissão de pareceres.

Entretanto, até que possa ser utilizado este sistema necessita ser incrementado com algumas funcionalidades essenciais, tais como os módulos de geração dos relatórios e de estatísticas. A interface do Administrador também precisa de ajustes importantes, sobretudo no que diz respeito aos recursos de personalização do sistema.

Para o prosseguimento deste projeto, além dos ajustes citados nos parágrafos anteriores, seria interessante a implementação de novos módulos, oferecidos ao usuário de maneira a possibilitar a configuração do sistema de acordo com as suas necessidades.

Outra característica bastante interessante, relacionada às tendências atuais de comunicação, seria a possibilidade de consultas por meio de telefonia celular. Desta forma, os participantes receberiam avisos e poderiam consultar informações como pareceres e datas importantes.

A inclusão de um fórum de discussões também seria bem vinda. Neste caso, existe uma opção pronta e de código aberto chamada PHPBB [<http://www.phpbb.com/>] que pode ser incluída no sistema como um novo módulo.

## 6. Referências

FLISYS (Laboratório de Camarões Marinhos). Disponível em

<<http://sourceforge.net/projects/flisys/>>. Acesso em: 02 de maio de 2009.

Manifesto Ágil. Disponível em <http://www.manifestoagil.com.br/>. Acesso em: 05 de maio de 2009

Desenvolvimento Ágil. Disponível em <http://devagil.wordpress.com/>. Acesso em 05 de maio de 2009

TELES, Vinícius Manhães. **Extreme Programming: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade.** Novatec Editora, 2004.

[HIGHSMITH] Extreme Programming, Highsmith, J. Orr, K. Cockburn, A., Ebusiness

Application Delivery, 2000

FREEMAN, Elisabeth; FREEMAN, Eric. **Use a Cabeça!** (*Head First*): HTML com CSS e XHTML. 1a. Edição. 2006.

[PRESSMAN] Engenharia de Software, Pressman, Roger S., Makron Books, 1995.

GONZAGA, Flávio S.; BIRCKAN, Guilherme. Curso de PHP e MySQL. Florianópolis: 2000.

WAMP. Disponível em <http://www.criarweb.com/artigos/327.php>. Acesso em 07 de maio de 2009.

PRUDÊNCIO, Achilles Colombo; VALOIS, Djali Avelino; DE LUCCA, José Eduardo. **Introdução à Internacionalização e à Localização de Software**. Florianópolis, 2005.

Casos De Uso em XP. Disponível em:

<http://www.infoq.com/br/news/2009/02/Use-Cases-Valuable-But-Optional> Acesso em 25 de maio de 2009