

Fabiano Yukio Otsuka

**DESENVOLVIMENTO DE UM APLICATIVO PARA GESTÃO DE
EVENTOS UTILIZANDO A FERRAMENTA DE PROGRAMAÇÃO
DELPHI**

Monografia de conclusão de curso apresentada ao
Departamento de Ciência da Computação da Uni-
versidade Federal de Lavras como parte das exigên-
cias para obtenção do título de Bacharel em Ciência
da Computação

Orientador
Prof. Reginaldo Ferreira de Souza

Lavras
Minas Gerais - Brasil
2004

Fabiano Yukio Otsuka

**DESENVOLVIMENTO DE UM APLICATIVO PARA GESTÃO DE
EVENTOS UTILIZANDO A FERRAMENTA DE PROGRAMAÇÃO
DELPHI**

Monografia de conclusão de curso apresentada ao
Departamento de Ciência da Computação da Uni-
versidade Federal de Lavras como parte das exigên-
cias para obtenção do título de Bacharel em Ciência
da Computação

Aprovada em Junho de 2004

Prof. André Luiz Zambalde

Prof. Reginaldo Ferreira de Souza
(Orientador)

Lavras
Minas Gerais - Brasil

Agradecimentos

Agradeço primeiramente aos meus pais que não mediram esforços para que eu concluísse mais essa etapa da minha vida. Agradeço aos meus irmãos, ao apoio dos meus colegas de curso que ao longo dos períodos acabaram se tornando meus amigos, aos integrantes da república São Nunca pela amizade e boa convivência. Agradeço também ao meu orientador Reginaldo Ferreira de Souza pelo apoio e orientação e a todos que de alguma forma contribuíram para que eu pudesse concluir o curso.

Resumo

O mercado de serviços exige cada vez mais excelência na qualidade. Não é conveniente, portanto, qualquer ação que não tenha sido analisada e planejada minuciosamente. A organização de um evento tem que satisfazer totalmente seus objetivos. Com esse intuito foi desenvolvido um Sistema para Gestão de Eventos (SGE), voltado para o organizador de um evento, que recebesse as informações necessárias para sua realização de forma organizada e que propiciasse consultas e gerasse relatórios para uma gestão mais eficiente.

Palavra-chave: Gestão de eventos, banco de dados e Delphi

Abstract

The market of services demands each time more excellency in the quality. It is not convenient, therefore, any action that has not been analyzed and planned minutely. The organization of an event has that to satisfy its objectives total. With this intention a System for Management of Eventos was developed (SGE), come back toward the organizador of an event, that received the necessary information for its accomplishment from organized form and that it propitiated consultations and it generated reports for a more efficient management.

Keywords: Management of events, data base and Delphi

Sumário

1	Introdução	1
1.1	Considerações Iniciais	1
1.2	Objetivos e Justificativas	2
1.3	Descrição dos capítulos	2
2	Referencial Teórico	5
2.1	Tecnologia da Informação	5
2.2	Sistemas de Informação	6
2.2.1	Sistema de Informação para gestão de eventos	9
2.3	Interface Homem Máquina	9
2.4	Engenharia de Software	13
2.4.1	Paradigmas da engenharia de software	14
3	Metodologia	23
3.1	Tipos de pesquisa	23
3.2	Ambiente de trabalho	24
3.3	Ferramentas utilizadas	25
3.3.1	Ferramenta de programação Delphi	25
3.3.2	Bando de dados Paradox	26
4	Resultados e Discussão	29
4.1	Funcionalidades	29
4.2	Modelagem dos dados	30
4.3	Desenvolvimento do sistema	34
4.4	Descrição do Sistema	35
4.4.1	Tela de login do sistema	36
4.4.2	Tela principal	36

4.4.3	Telas de cadastros	38
4.4.4	Telas de consultas	43
4.4.5	Tela de criação do arquivo de backup	47
5	Conclusões	49
6	Referências bibliográficas	51
A	Detalhamentos sobre as entidades do Modelo Entidade Relacionamento	53

Lista de Figuras

2.1	Modelo classico	15
2.2	Prototipação	18
2.3	Modelo espiral	19
2.4	Tecnicas de quarta geração	20
4.1	Modelo Entidade Relacionamento do banco de dados	32
4.2	Diagrama de casos de uso	35
4.3	Menu de cadastros	37
4.4	Menu de consultas	37
4.5	Tela principal do sistema	37
4.6	Menu de cadastros	38
4.7	Menu de consultas	38
4.8	Cadastro de um evento	39
4.9	Cadastro do(s) responsável(s) pelo evento	39
4.10	Mensagem de confirmação para exclusão	39
4.11	Cadastro do local de realização do evento	40
4.12	Mensagem de campo obrigatório	40
4.13	Cadastro dos detalhes do evento	41
4.14	Cadastro do ministrante de um curso	42
4.15	Cadastro dos participantes, patrocinadores e organizadores	42
4.16	Cadastro de um novo usuário	43
4.17	Seleção de um evento	43
4.18	Detalhes sobre o evento selecionado	44
4.19	Consulta sobre os participantes de um evento	45
4.20	Relatório da listagem dos participantes de um evento	46
4.21	Menu de cadastros	47
4.22	Menu de consultas	47

Lista de Tabelas

4.1	Tabela de eventos	33
4.2	Tabela dos dados pessoais do responsável pelo evento	33
A.1	Tabela dos dados sobre um curso	53
A.2	Tabela dos dados pessoais do ministrante de um curso	54
A.3	Tabela dos dados de uma palestra	54
A.4	Tabela dos dados pessoais de um palestrante	55
A.5	Tabela dos dados pessoais de um participante	55
A.6	Tabela dos dados sobre o patrocinador	56
A.7	Tabela dos dados sobre um organizador	56
A.8	Tabela dos dados sobre o local do evento	57
A.9	Tabela dos dados sobre o responsável pelo local do evento	57
A.10	Tabela dos dados sobre um seminário	58
A.11	Tabela dos dados sobre um seminarista	58
A.12	Tabela dos dados sobre uma mesa redonda	59
A.13	Tabela dos dados sobre um integrante de uma mesa redonda	59

Capítulo 1

Introdução

1.1 Considerações Iniciais

O gerenciamento da informação atualmente tem se tornado um dos itens mais importantes para que uma empresa seja bem sucedida em seu ramo de atuação. A necessidade de informação aumenta em relação ao desenvolvimento de uma sociedade. Com isso vem a necessidade de organizar todos esses dados, que separados representam apenas um conjunto de caracteres que não transmitem nenhuma mensagem, em uma informação relevante.

A forma com que uma empresa gerencia suas informações pode ser decisiva para poder se manter ou prosperar em um mercado instável e competitivo como o atual. Por isso cada vez mais empresas investem um grande capital na modernização da sua Tecnologia da Informação (TI). Entretanto somente o investimento em tecnologia de ponta não é garantia de sucesso, pois é necessária a utilização de hardware e software específicos para necessidades de cada empresa.

Cada empresa deve dispor da informática de forma otimizada, adequando o seu uso ao seu porte e a suas peculiaridades. Agir de outra forma pode acarretar desperdício em tecnologias caras e desnecessárias ou privar seu negócio dos recursos que ele precisa para se desenvolver plenamente. Instalações diversas e a compra de equipamentos vão melhorar e agilizar os processos internos de gestão e o fluxo de informações, caso seja implementado de forma correta. Desta forma, uma empresa poderá diminuir custos, melhorar o atendimento aos clientes e aumentar a sua eficiência.

A agilidade com que se gerencia informações diversas é um fator importante para o bom funcionamento de uma empresa promotora de eventos. Devido a

grande quantidade de informações geradas na organização de eventos como congressos, seminários, convenções e outros, o investimento em um software que possa auxiliar no gerenciamento dos dados gerados, poderia diminuir custos e melhorar a sua eficiência.

Como atualmente o intercâmbio de informações e tecnologias está se tornando cada vez mais freqüente, as pessoas estão tendo a necessidade de reciclar seus conhecimentos, não só para melhorar suas performances profissionais como empresariais. Geralmente essa troca ocorre em eventos como semana de cursos, palestras, congressos, etc, transformado a organização de eventos em um setor bastante promissor. Portanto o desenvolvimento de um aplicativo específico capaz de otimizar e melhorar a gestão de um evento seria muito interessante, pois melhoraria a competitividade da empresa em relação às outras.

Como uma empresa de eventos está sempre organizando novos eventos ou simplesmente realizando eventos nas datas já predefinidas todos os anos, a possibilidade de se utilizar todas as informações dos eventos já cadastrados evitaria erros cometidos anteriormente e a necessidade de começar a organização de um evento do início, através do reaproveitamento dos dados anteriores.

1.2 Objetivos e Justificativas

O mercado de serviços exige cada vez mais excelência na qualidade. Não é conveniente, portanto, qualquer ação que não tenha sido analisada e planejada minuciosamente. A organização de um evento tem que satisfazer totalmente seus objetivos. Com esse intuito foi desenvolvido um Sistema para Gestão de Eventos (SGE), para ser utilizado pelo organizador e que recebesse as informações necessárias para sua realização de forma organizada e que propiciasse consultas e gerasse relatórios para uma gestão mais eficiente.

1.3 Descrição dos capítulos

Nesta seção é feita uma descrição de cada capítulo deste trabalho que se encontra dividido em seis capítulos. No capítulo 2, é feita uma descrição teórica sobre os assuntos necessários para o desenvolvimento do projeto. No capítulo 3, são descritas a metodologia de pesquisa realizada para obtenção do conhecimento, os ambientes de trabalho e as ferramentas utilizadas no desenvolvimento do software. No capítulo 4, são mostrados os resultados e discussões sobre o desenvolvimento

do Sistema de Gestão de Eventos que foi definido a partir das descrição das funcionalidades, modelagem dos dados, modelagem do sistema e uma descrição do seu funcionamento. No capítulo 5, é feita uma conclusão e possíveis trabalhos futuros para complementar o projeto. E para finalizar é apresentado as referencias bibliográficas.

Capítulo 2

Referencial Teórico

2.1 Tecnologia da Informação

Com o intuito de aumentar sua competitividade no mercado as organizações estão utilizando uma variada e complexa gama de tecnologias. Uma que esta sendo bastante difundida atualmente é a Tecnologia da Informação (TI), que se tornou uma peça muito importante em organizações de todos os tipos que desejam se manter ou crescer na disputa cada vez mais acirrada por espaço no mercado.

De acordo com [Souza 2001] citando Laudon e Laudon¹, as tecnologias de informação contemporâneas vão além do computador isolado e abrangem as redes de comunicações, equipamentos de fax, impressoras e copiadoras "inteligentes", workstations (estações de trabalho), processamento de imagens, gráficos, multimídia, e comunicações em vídeo. É notório que os problemas não serão solucionados por mainframes ou um microcomputador, mas por um conjunto de dispositivos digitais dispostos em rede.

A TI fornece uma ampla gama de ferramentas e técnicas para o gerenciamento mais eficiente das informações importantes estrategicamente para uma organização. Ela influencia o modo como as pessoas utilizam e pensam sobre a tecnologia na organização. Com isso há um maior aproveitamento do potencial de recursos disponíveis, resultando em ganho de eficiência dos trabalhadores.

Todos esses pontos positivos proporcionados pela TI não serão alcançados se não houver um estudo antecipado sobre o funcionamento da organização em relação aos setores a serem informatizadas. Uma implantação feita sem um planeja-

¹Laudon, K. C. e Laudon, J. P. Sistemas de informação, Rio de Janeiro: LTC, 1999, 389 p.

mento prévio traria grandes prejuízos à empresa, pois seria feito um investimento em equipamentos de hardware e software que seriam desnecessários.

Segundo[Freitas 2003] citando Adriana Beal², diz que o principal benefício que a tecnologia da informação traz para as organizações é a sua capacidade de melhorar a qualidade e a disponibilidade de informações e conhecimentos importantes para a empresa, seus clientes e fornecedores. Os sistemas de informação mais modernos oferecem às empresas oportunidades sem precedentes para a melhoria dos processos internos e dos serviços prestados ao consumidor final. Portanto, um estudo detalhado sobre o funcionamento de uma empresa seria o passo principal para uma implantação da Tecnologia da informação de forma correta. Cabe a empresa primeiramente fazer um levantamento dos processos a serem informatizados e posteriormente dos equipamentos necessários para o prosseguimento do processo de informatização. Com isso as chances de um investimento sem retorno diminuiriam bastante. Isso é descrito em[Souza 2001], citando autores como Mañas³, onde diz que a utilização de TI exige um planejamento que não é um processo puramente técnico, realizado por especialistas em TI, mas um procedimento gerencial que envolve a organização como um todo. O risco da utilização sem planejamento é citado por [Cruz 1998], que afirma que o uso crescente da TI, ao mesmo tempo em que potencializa a capacidade das organizações em obter, manter ou combater vantagens competitivas também eleva os riscos de gerenciamento, inerentes a qualquer tipo de decisão.

2.2 Sistemas de Informação

Alguns conceitos genéricos de sistema são, segundo [Rezende 2002], o conjunto de partes que interagem entre si, integrando-se para atingir um objetivo ou resultado; partes integrantes e interdependentes que conjuntamente formam um todo com determinados objetivos e efetuam determinadas funções; em informática, é o conjunto de software, hardware e recursos humanos; componentes da Tecnologia da Informação e seus recursos integrados; empresa e seus vários subsistemas.

Com esses conceitos de sistema podemos entender melhor o conceito de Sistemas de Informação (SI) segundo [Rezende 2002], são todos os sistemas que produzem e/ou geram informações, que são dados trabalhados (ou com valor atribuído ou agregado a eles) para execução de ações e para auxiliar nos processos de tomadas de decisões.

²Citação retiradas do site <http://gwarh.com.br/p62.htm>

³ Mañas, Vico A. Administração de sistemas de informação, São Paulo: Érica, 1999, 285 p.

Segundo [Souza 2001] o tratamento e a estruturação de dados geram informações, que devem apresentar as seguintes características:

- Precisa - a informação precisa não tem erros. Em alguns casos, a informação imprecisa é gerada pela entrada de dados incorretos no processo de transformação. Isto é comumente chamado de entra lixo, sai lixo (ELSL).
- Completa - a informação completa contém todos os fatos importantes. Por exemplo, um relatório de investimento que não inclui todos os custos importantes não está completo.
- Econômica - a informação também deve ser de produção relativamente econômica. Os tomadores de decisões devem sempre fazer um balanço do valor da informação com o custo de sua produção.
- Flexível - a informação flexível pode ser usada para diversas finalidades. Por exemplo, a informação de quanto se tem de estoque disponível de uma determinada peça pode ser usada pelos representantes de vendas no fechamento de uma venda, por um gerente de produção para determinar se mais estoque é necessário, e por um diretor financeiro para determinar o valor total que a empresa tem investido em estoque.
- Confiável - a informação confiável pode ser dependente. Em muitos casos, a confiabilidade da informação depende da confiabilidade do método de coleta dos dados. Quer dizer, a confiabilidade depende da fonte da informação. Um boato vindo de fonte desconhecida que os preços do petróleo devem subir pode não ser confiável.
- Relevante - a informação relevante é importante para o tomador de decisões. A informação de que os preços da madeira de construção devem cair pode não ser relevante para um fabricante de chips de computador.
- Simples - a informação também deve ser simples, não deve ser muito complexa. A informação sofisticada e detalhada pode não ser necessária. Na realidade, informação em excesso pode causar sobrecarga de informação, quando um tomador de decisões tem informação demais e não consegue determinar o que realmente importante.
- Em tempo - a informação em tempo é enviada quando necessário. Saber as condições do tempo da semana passada não ajudará a decidir qual o agasalho vestir hoje.

- Verificável - finalmente, a informação deve ser verificável. Isto significa que se pode checá-la para saber se está correta, talvez checando várias fontes da mesma informação.

Independente do tipo de sistema de informação em análise, tem ficado claro seu papel nas organizações nos seguintes aspectos segundo [Souza 2001], citando Roque⁴:

- Seus processos - um processo é um sistema de atividades, um conjunto de ações coordenadas com o intuito de atingir um determinado objetivo; um sistema de informações pode tornar um processo mais eficiente, rápido, econômico, etc;
- Seus produtos - produtos são as saídas dos processos que se diferem pelas características próprias que podem ser aprimoradas com um sistema de informação que, em última instância, controla suas características (informações);
- Seu gerenciamento, uma vez que permite que os gerentes voltem-se para suas atividades principais;
- Aumento da qualidade, através dos feedback constantes e da facilidade de ajustes de processos fornecidos pelos sistemas de informações;
- Tomada de decisões mais adequadas, a partir do momento em que estas irão basear-se em informações mais consistentes;
- Aquisição e manutenção de vantagens competitivas, quando permite que esta organização consiga mudar sua própria natureza através das possibilidades anteriormente citadas.

Desempenhando um papel de tamanha importância, um SI é parte integrante de uma organização e é um produto de três componentes: tecnologia, organizações e pessoas. Não existe sistema de informação eficiente sem o conhecimento de suas dimensões no que se refere à organização, pessoas, assim como de suas dimensões técnicas conforme [Souza 2001] citando Laudon⁵.

⁴Roque, R. F. Estudo comparativo de metodologias de desenvolvimento de informação utilizando a técnica delphi, 1998 <<http://www.eps.ufsc.br/disserta98/ruth/index.htm>> (01/03/2000).

⁵Laudon, K. C. e Laudon, J. P. Sistemas de informação, Rio de Janeiro: LTC, 1999, 389 p.

Empresas que possuem seu sistema de informação implementada de forma correta possuem uma vantagem competitiva e estratégia em relação às outras, pois esses são um dos objetivos incorporados ao sistema de informação, que não são mais apenas uma ferramenta facilitadora das tarefas rotineiras nas organizações, mas sim uma peça importante no complexo quebra cabeça na disputa por sucesso no mundo dos negócios.

2.2.1 Sistema de Informação para gestão de eventos

Conforme [Rithcher 2003], sistemas de gestão de eventos são sistemas que minimizam o tempo gasto para fazer operações administrativas. E que, se fossem realizadas de maneira convencional, tomariam um tempo considerável para serem realizadas. Eles permitem a concentração do fluxo de informações de uma forma rápida, segura e totalmente dinâmica. Fornecem aos organizadores uma visão das atividades regulares da empresa, de modo que possam controlar, organizar e planejar, de maneira eficaz e eficiente, as atividades e eventos.

2.3 Interface Homem Máquina

Inicialmente os computadores e softwares eram desenvolvidos por engenheiros para serem usados por engenheiros. Como esse grupo de usuários não representa a grande maioria, que é composta atualmente por usuários comuns, a fase de desenvolvimento da interface ganhou grande importância diante das outras. Como o uso de computadores pela sociedade tem crescido continuamente, a interface se tornou um fator determinante no sucesso ou fracasso de um produto.

A definição de Interface Homem Máquina de acordo com [Zambalde 2003], é parte do software de um sistema interativo responsável por traduzir ações do usuário em ativações das funcionalidades do sistema (aplicação), e permitir que os resultados possam ser observados, coordenando esta interação. Em outras palavras, a interface é responsável pelo mapeamento das ações do usuário sobre dispositivos de entrada e pedidos de processamento fornecidos pela aplicação, e pela apresentação em forma adequada dos resultados produzidos.

A interface homem-máquina é uma preocupação da indústria muito antes de se falar em interface de programas de computador. Existe muita experiência adquirida sobre o assunto, como por exemplo, a utilização de mais de um perfil de usuário e níveis de conhecimento ou experiência (iniciante, intermediário e expert).

De acordo com [Zambalde 2003], o termo amigável (user-friendly) é comumente atribuído a interfaces. Seguem algumas características presentes em sistemas onde a interface recebe este adjetivo:

- Facilidade de usar e aprender (easy-to-use and easy-to-learn): interface invisível, previsível, flexível e as pessoas gostam dela (o usuário pode concentrar-se nas tarefas que necessita realizar em termos de aplicação);
- Taxa de erro mínimo: em sistemas críticos (nucleares, controle de tráfego aéreo, etc.) dificultar a ocorrência de erros cometidos pelos usuários é imprescindível. Convém lembrar que erros também afetam o desempenho do usuário;
- Recordação rápida: o usuário esporádico, idealmente, não deve ter que recorrer a manuais quando for usar o sistema;
- Atratividade: nem sempre o sistema mais poderoso é preferido pelo usuário. O usuário pode selecionar um sistema em que seu desempenho é inferior comparando com um outro, mas com o qual se sente mais confortável.

Segundo [Zambalde 2003], a interface pode influir na produtividade do usuário, que nem sempre prefere um sistema com mais recursos ou eficiência do ponto de vista computacional. Um exemplo real é o comentário de Bill Gates, acerca da frustração parcial com o Word 2.0.

"Merecemos a culpa por não termos facilitado o seu aprendizado. No tocante aos recursos o produto era fantástico, mas no que se refere a facilidades dos primeiros passos não nos saímos muito bem."

De acordo com [Zambalde 2003], o foco da atenção atualmente dado às interfaces possui uma linha simples de justificativa:

- O uso de computadores tem crescido continuamente. Prevê-se que praticamente todo ser humano irá utilizar computadores no futuro de uma ou de outra forma;
- O mercado tem mostrado que, nas vendas entre produtos similares, sobressai o que melhor permite o acesso do usuário à funcionalidade fornecida pelo sistema. Convém ressaltar que em alguns casos a funcionalidade e o desempenho não são suficientes para agradar o usuário, que faz opção por outro sistema com interface atrativa. Ou seja, se um produto deseja ser competitivo, necessariamente sua interface deve ser considerada de forma séria.

- O custo de um sistema não se limita a hardware + software. É preciso treinar usuários. Quanto mais difícil de aprender, mais oneroso é o treinamento e quanto mais difícil de usar, menor é o desempenho do usuário através de erros constantes, lentidão de operação do sistema e outros;
- Softwares com interface inadequada tendem a ser rejeitados pelos usuários.

Devido a todo esse enfoque dado a Interface Homem Máquina vários estudos feitos sobre a dificuldade de desenvolvimento de um projeto de interface são discutidos. Segundo [Kiko 2003], embora os benefícios da melhora da usabilidade de uma interface sejam indiscutíveis, não estão resolvidos os problemas que levam à dificuldade do projeto de uma Interface. Discute-se algumas dificuldades abaixo:

- A dificuldade em entender as tarefas e os usuários - A necessidade da interface estar diretamente ligada ao modo com que será usado requer compreensão profunda dos usuários e de suas habilidades e expectativas. Levantar este tipo de informação é difícil, sobretudo porque programadores têm dificuldade de se imaginarem efetivamente na condição de usuários comuns.
- A complexidade inerente às tarefas e aplicações - Em geral, o domínio da aplicação a ser criado envolve situações de difícil modelagem, seja porque a tarefa em si é complicada, seja porque a aplicação se propõe a resolver problemas de gama extensa.
- A variedade de aspectos e requisitos diferentes - Além das limitações inerentes a qualquer projeto, interfaces com o usuário envolvem questões como padrões, design gráfico, documentação, internacionalização, e performance, entre outras. Estas questões associadas contribuem para aumentar a complexidade do desenvolvimento da interface.
- Teoria e métodos não são suficientes para resolver o problema - Embora existam muitas metodologias para a criação de uma interface boa, a maior parte dos estudos feitos a seu respeito revela que a habilidade dos projetistas é o fator primário para a qualidade das interfaces geradas. O fato de existir grande proporção de casos que sejam exceções às regras propostas nos métodos contribui para a dificuldade de se criar um método abrangente.
- Dificuldade de se fazer um projeto iterativo - Embora se veja como ideal o processo de se refinar ciclicamente uma interface, este processo em si já é difícil de ser executado. Muitas vezes, as modificações trazem uma

piora de usabilidade, e é difícil saber quando a interface está realmente bem-elaborada. Além disso, é difícil obter resultados do uso da interface diretamente dos usuários primários, que muitas vezes não são os seus compradores e nem responsáveis.

De acordo com [Xchart 2003], em recente pesquisa sobre dezenas de projetos [Mye 1992], verifica-se que em média 48% do código de um sistema interativo é dedicado à interface. Quanto ao tempo total de desenvolvimento, aquele consumido na interface equivale a 48% do tempo de projeto de todo o sistema, 50% de toda a implementação e 37% da manutenção. Assim fica evidente a dificuldade de se implementar uma boa interface. Segundo [Kiko 2003], há razões importantes que explicam porque software relacionado à interface estará entre os mais difíceis de se implementar.

- A necessidade de se fazer um projeto iterativo - Implementar a interface iterativamente significa entrelaçar as etapas de projeto e implementação e teste; o impacto direto disto é que o processo de software tradicional se torna inadequado, e o processo utilizado em si, mais difícil de administrar e documentar.
- Programação reativa - Uma das diferenças principais entre a implementação de software para a interface e software de outros tipos é que o software deve ser escrito contemplando seu controle pelo usuário, e não pela aplicação. Esta diferença tem mostrado resultar em uma maior dificuldade na modularização e organização do código.
- Multiprocessamento - Além de existirem diversas tarefas simultâneas que necessitem responder à atenção do usuário, o tempo de execução destas tarefas varia, e é importante manter o sistema respondendo instantaneamente. Por este motivo, os mesmos problemas relacionados ao multiprocessamento, sincronização, controle de concorrência e consistência das informações, existem no desenvolvimento de interfaces.
- Programação em tempo real - Há limitações importantes do tempo de resposta e do redesenho das informações exibidas pela interface; esta característica embute preocupações que a maior parte dos sistemas não contempla a fundo.
- Necessidade de robustez - Toda entrada do usuário deve ser atendida satisfatoriamente; isto difere em muito da robustez costumeira que se deseja

em um software. As entradas dos usuários podem ser de qualquer natureza, e muitas vezes é difícil prever todos os tipos diferentes de interação que ocorrerão. A necessidade de cancelar e desfazer as operações que o usuário executa acrescenta a esta dificuldade.

- Dificuldade de sistematizar testes - Sistemas de manipulação direta à interface são em geral pouco úteis para efetuar seu teste, porque não simulam bem a atividade caótica do usuário, e porque alterações na interface implicam em se ter de fazer mudanças nos testes. Além disso, grande parte da qualidade da interface é não-avaliável por um sistema, por ser resultado de apreciações essencialmente humanas.
- Complexidade das ferramentas - Como a linguagem de programação muitas vezes não suporta completamente as necessidades da interface, cria-se uma necessidade por ferramentas que enderecem esta questão. No entanto, estas ferramentas em geral são incompatíveis e requerem longo treinamento por parte dos desenvolvedores.
- Dificuldade de modularização - Uma das maiores dificuldades relacionadas à criação de interfaces é a dificuldade de se separar a forma da interface com o conteúdo a ser apresentado – muitas vezes, mudanças na interface implicam em mudanças na funcionalidade da aplicação. Além disso, o uso extensivo de funções call-back nas ferramentas torna o problema ainda maior, porque cada interação com uma parte da interface (um widget) requer uma função correspondente.

Para se atingir todos os requisitos de uma boa interface essas dificuldades devem ser sanadas através de novas técnicas e teorias desenvolvidas no campo da Interface Homem Máquina. Certamente todos os problemas citados não serão resolvidos, além disso, novas formas de interação irão aparecer constantemente.

2.4 Engenharia de Software

Atualmente as maiorias dos países possuem uma dependência de sistemas complexos computadorizados. Cada vez mais processos são automatizados utilizando computadores e software de controle. Como softwares cada vez mais específicos e complexos estão sendo desenvolvidos, o seu custo em uma implantação possui um peso considerável no planejamento de um projeto. Portanto, a produção de

um software de forma que sua relação custo/benefício seja boa é muito importante para o bom funcionamento da empresa desenvolvedora.

Antes de entendermos a definição de Engenharia de Software vamos entender a definição de software segundo [Sommerville 2003], que define como sendo não apenas o programa, mas também toda a documentação associada e os dados de configuração necessários para fazer com que esses programas operem corretamente. Um sistema de software, usualmente, consiste em uma série de programas separados, arquivos de configuração que são utilizados para configurar esses programas, documentação do sistema, que descreve a estrutura desse sistema, e documentação do usuário, que explica como utilizar o sistema e, no caso dos produtos de software, sites Web para os usuários fazerem o download das informações recentes sobre o produto.

Segundo [Sommerville 2003] os engenheiros definem dois tipos de produtos de software:

- Produtos genéricos - São sistemas do tipo stand-alone, produzidos por uma organização de desenvolvimento e vendidos no mercado a qualquer cliente capaz de adquirí-los. Algumas vezes, referimo-nos a eles como pacotes de software. Ex: processadores de texto, pacotes de desenho, etc.
- Produtos sob encomenda (ou personalizados) - São sistemas encomendados por um cliente em particular. O software é desenvolvido especialmente para aquele cliente por uma empresa de software. Ex: Sistema de tráfego aéreo, sistema de controle para dispositivos eletrônico, etc.

Com essas definições podemos entender melhor o que é engenharia de software de acordo com [Sommerville 2003], é uma disciplina da engenharia que se ocupa de todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até a manutenção desse sistema, depois que ele entrou em operação.

2.4.1 Paradigmas da engenharia de software

Um modelo de processo de software é uma representação abstrata de um software. Cada modelo de processo representa um processo a partir de uma perspectiva particular, de uma maneira que proporciona apenas informações parciais sobre o processo. Abaixo são apresentados alguns modelos ou paradigmas genéricos a partir de uma perspectiva arquitetural.

Os paradigmas não são descrições definitivas de processos de software, mas sim abstrações úteis, que podem ser utilizadas para explicar variadas abordagens de desenvolvimento de software. Para o desenvolvimento de grandes sistemas não é utilizado apenas um paradigma, mas um conjunto de modelos para partes diferentes do sistema.

O ciclo de vida clássico ou modelo cascata

Esse modelo considera as atividades de especificação, desenvolvimento, validação e evolução, que são fundamentais ao processo, e as representa como fases separadas do processo, como a análise e engenharia de sistemas, análise de requisitos de software, projeto (design), implementação ou codificação, testes e manutenção. O ciclo de vida clássico é o paradigma mais antigo e o mais amplamente usado na engenharia de software, esse modelo pode ser visto na Figura 2.1.

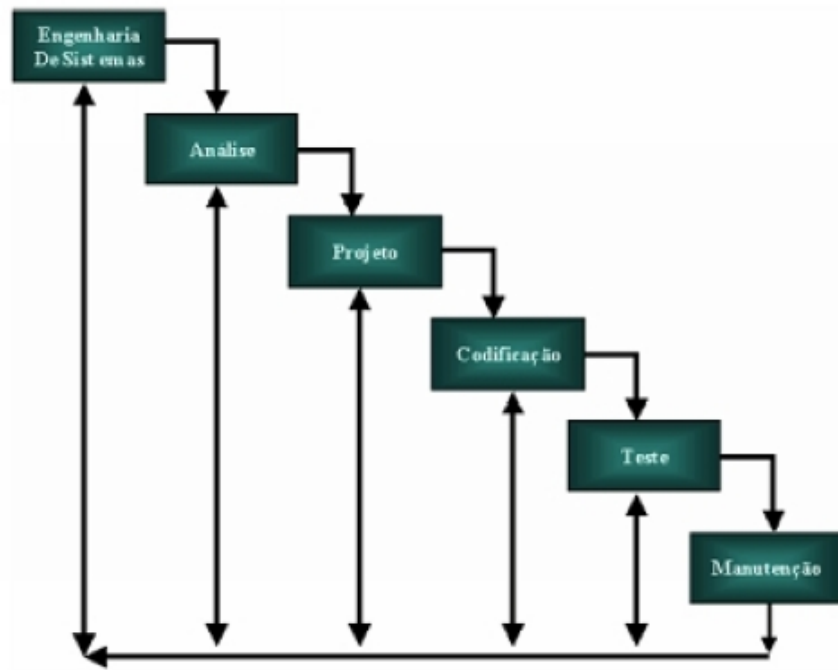


Figura 2.1: Modelo classico
Fonte: Pressman 1995

- Engenharia de sistemas: essencial quando o software deve fazer interface com outros elementos, tais como, hardware, pessoas e banco de dados.
- Análise de requisitos de software: o engenheiro de software deve compreender o domínio da informação para o software, bem como a função, desempenho e interface exigidos. Os requisitos são documentados e revistos com o cliente.
- Projeto: quatro atributos: estrutura de dados, arquitetura de software, detalhes procedimentais e caracterização da interface. O projeto traduz as exigências em uma representação que pode ser avaliada quanto à qualidade antes que a codificação se inicie. O projeto é documentado e torna-se parte do software.
- Codificação: O projeto deve ser traduzido em uma forma legível por máquina.
- Testes: Assim que o código for gerado, inicia a realização de testes do programa. Descobrir erros e garantir que a entrada definida produza resultados reais que correspondem com os resultados exigidos.
- Manutenção: Indubitavelmente, o software sofrerá mudanças depois que for entregue ao cliente. Porque foram encontrados erros; adaptado devido a mudanças em seu ambiente externo; ou porque o cliente exige acréscimos funcionais ou de desempenho.

Teoricamente uma fase seguinte não deve começar até que a fase anterior tenha sido concluída. Mas na prática essas fases se sobrepõem e trocam informações entre si. O processo de desenvolvimento de software não é um modelo linear, ele envolve uma seqüência de iterações das várias atividades de desenvolvimento.

O modelo em cascata possui um problema, a sua inflexível divisão do projeto em estágios distintos. Os acordos devem ser feitos em um estágio inicial do processo, ou seja, é difícil responder aos requisitos do cliente, que sempre quer modificar alguma coisa. Com isso o modelo em cascata deveria ser utilizado quando os requisitos forem bem definidos.

Prototipação de software

A definição de todos os requisitos necessários ao sistema pelo cliente ou usuário geralmente é uma tarefa muito difícil. É quase impossível prever como o sistema irá afetar o funcionamento das práticas de trabalho, como será a interação com

outros sistemas e que operações dos usuários devem ser automatizadas. Mas para poder testar os requisitos de uma forma mais eficiente, seria necessária a utilização de um protótipo do sistema, esse modelo pode estar representado na Figura 2.2.

Um protótipo segundo [Sommerville 2003], é uma versão inicial de um sistema de software, que é utilizada para mostrar conceitos, experimentar opções de projeto e, em geral, para conhecer mais sobre os problemas e suas possíveis soluções. O desenvolvimento rápido de um protótipo é essencial para que os custos sejam controlados e os usuários possam fazer experiências com o protótipo no início do processo de software.

De acordo com [Sommerville 2003], um protótipo de software apóia duas atividades do processo de engenharia de requisitos:

- Levantamento de requisitos - Os protótipos de sistema permitem que os usuários realizem experiências para ver como o sistema apóia seu trabalho. Eles obtêm novas idéias para os requisitos e podem identificar pontos positivos e negativos do software. Eles podem, então, propor novos requisitos de sistema.
- Validação de requisitos - O protótipo pode revelar erros e omissões nos requisitos propostos. Uma função descrita em uma especificação pode parecer útil e bem-definida. Contudo, quando essa função é utilizada com outras, os usuários muitas vezes acham que sua visão inicial era incorreta e incompleta. A especificação de sistema pode então ser modificada para refletir sua compreensão alterada dos requisitos.

De acordo com [Freitas 2003] citando Brooks⁶, na maioria dos projetos, o primeiro sistema construído dificilmente será usável. Ele pode ser muito lento, muito grande, desajeitado em uso, ou todos os três. A questão administrativa, não é se deve construir um sistema-piloto e jogá-lo fora. Isso será feito. A única questão é se deve planejar antecipadamente a construção de algo que se vai jogar fora ou prometer entregar isso aos clientes.

A prototipação como paradigma da engenharia de software pode ser problemática pelas seguintes razões:

- Quando informamos que o produto precisa ser reconstruído, o cliente grita impropérios e exige que alguns acertos sejam aplicados para tornar o protótipo um produto; muito freqüentemente, a gerência de desenvolvimento de software cede.

⁶Brooks F. The Mythical Man-Month. Addison-Wesley, 1975



Figura 2.2: Prototipação
 Fonte: Pressman 1995

- O desenvolvedor muitas vezes faz concessões de implementação a fim de colocar um protótipo em funcionamento rapidamente. Depois de algum tempo, o desenvolvedor pode familiarizar-se com essas opções e esquecer-se de todas as razões pelas quais elas são inadequadas - a opção menos ideal se tornou então parte integrante do sistema.

De acordo com [Sommerville 2003], as melhorias na facilidade de uso e melhores requisitos de usuário, provenientes do uso de protótipo, não significa necessariamente um aumento geral dos custos de desenvolvimento de sistema. A prototipação geralmente aumenta os custos dos primeiros estágios do processo de software, mas reduz os custos posteriores. A razão principal disso é que o retrabalho durante o desenvolvimento é evitado, quando os clientes solicitam menos mudanças no sistema. Contudo, eles constataram que uma consequência negativa da prototipação é que o desempenho geral do sistema é, por vezes, degradado quando um código de protótipo ineficiente é reutilizado.

O modelo espiral

Segundo [Pressman 1995], o modelo espiral foi criado visando abranger as melhores características do modelo clássico e da prototipagem. De certa maneira, este

modelo é semelhante à programação exploratória. A cada ciclo da espiral, versões progressivamente mais completas do software são construídas e, antes de cada ciclo, uma análise de riscos é feita. Ao fim de cada ciclo, é feita uma avaliação e então deve prosseguir para o próximo ciclo.

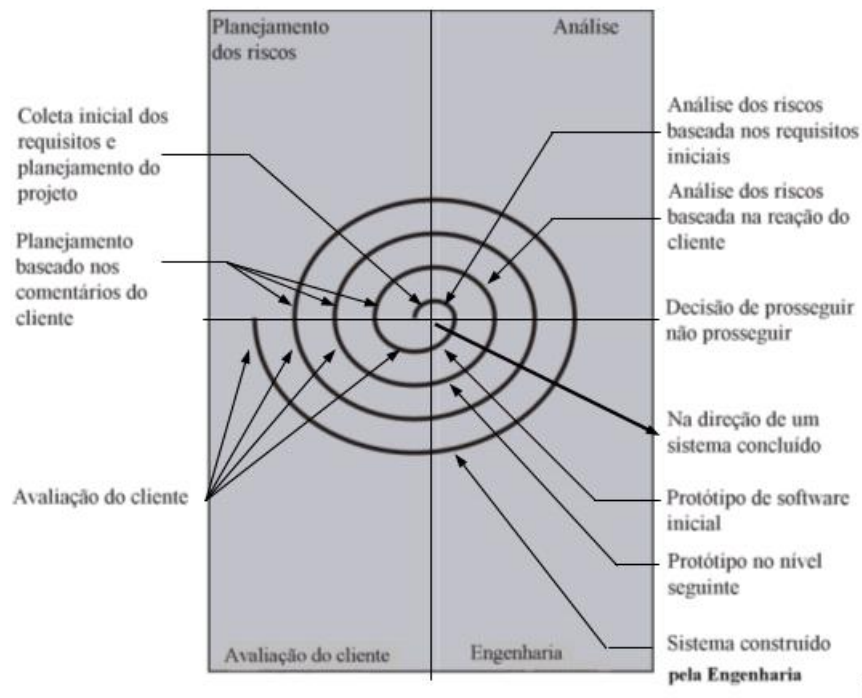


Figura 2.3: Modelo espiral
Fonte: Pressman 1995

Segundo [Pressman 1995], as etapas do modelo espiral se dividem em:

- Planejamento: determinação dos objetivos, alternativas e restrições;
- Análise dos riscos: análise de alternativas e identificação e resolução dos riscos;
- Engenharia: desenvolvimento do produto no nível seguinte;
- Avaliação feita pelo cliente: avaliação dos resultados da engenharia.

Esse paradigma da engenharia de software é considerado a abordagem mais realista no desenvolvimento de um software de grande escala. Ele minimiza os riscos de erros utilizando a prototipação, que pode ser aplicada em qualquer etapa da evolução do produto. O modelo espiral exige uma consideração direta dos riscos técnicos em todas as etapas do projeto e, se adequadamente aplicado, deve reduzir os riscos antes que eles se tornem problemáticos.

Técnicas de quarta geração

As técnicas de quarta geração (4GT) possuem um variado conjunto de ferramentas de software, que tem uma característica em comum que é a possibilidade do desenvolvedor de software especificar alguma característica num nível mais elevado. A característica mais importante desse paradigma é a capacidade de se especificar software a uma máquina em um nível que esteja próximo à linguagem natural ou de se usar uma notação que comunique uma função significativa.

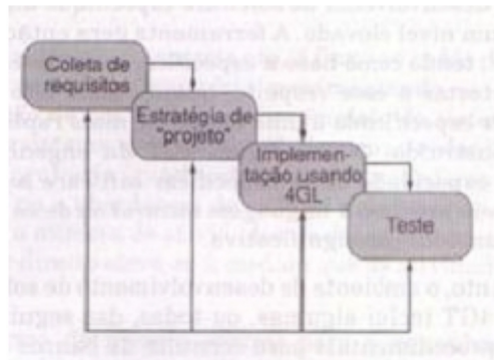


Figura 2.4: Técnicas de quarta geração
Fonte: Pressman 1995

A primeira etapa deste paradigma consiste em coletar os requisitos. Idealmente o cliente descreveria os requisitos, e estes seriam diretamente convertidos em um protótipo operacional. Mas isso não acontece devido o cliente não ter certeza daquilo que realmente quer.

De acordo com [Pressman 1995], para pequenas aplicações, talvez seja possível passar diretamente da etapa de coleta das exigências para a implementação, utilizando uma linguagem de quarta geração (4GL). Porém, para esforços maiores, é preciso desenvolver uma estratégia de projeto para o sistema, mesmo que se pretenda usar uma 4GL. O uso da 4GL sem planejamento causará as mesmas dificul-

dades que temos encontrado quando desenvolvemos software usando abordagens convencionais.

Segundo [Pressman 1995], tem havido muito exagero e considerável debate em torno do uso do paradigma 4GT. Os proponentes reivindicam uma drástica redução no tempo de desenvolvimento de software e produtividade fortemente elevada para as pessoas que constroem software. Os opositores afirmam que as atuais ferramentas 4GT não são assim tão mais fáceis de se usar do que as linguagens de programação, que o código produzido por tais ferramentas é ineficiente e que a manutenibilidade de grandes sistemas desenvolvidos pelas técnicas de 4GL está aberta a questionamentos.

Capítulo 3

Metodologia

3.1 Tipos de pesquisa

O desenvolvimento do projeto foi dividido em duas fases, onde o objetivo da primeira foi a obtenção de informação sobre o tema, tecnologias e ferramentas a serem utilizadas, para isso as metodologias utilizadas foram às pesquisas bibliográfica e documental. Segundo [Gil 1991], a pesquisa bibliográfica é desenvolvida a partir de material já elaborado, constituído principalmente de livros e artigos científicos. Embora em quase todos os estudos seja exigido algum tipo de trabalho desta natureza, há pesquisas desenvolvidas exclusivamente a partir de fontes bibliográficas.

A pesquisa documental segundo [Gil 1991], assemelha-se muito à pesquisa bibliográfica. A diferença essencial entre ambas está na natureza das fontes. Enquanto a pesquisa bibliográfica se utiliza fundamentalmente das contribuições dos diversos autores sobre determinado assunto, a pesquisa documental vale-se de materiais que não receberam ainda um tratamento analítico, ou que ainda podem se reelaborados de acordo com os objetivos da pesquisa.

O sistema de gestão de eventos implementado é uma continuação do projeto Desenvolvimento de um Aplicativo para Gestão de Eventos Utilizando a Tecnologia J2SE realizado por [Rithcher 2003] que foi implementado utilizando a linguagem de programação Java. Sua implementação dava suporte para utilização via intranet com arquitetura cliente/servidor. Ele podia ser utilizado tanto pelos organizadores quanto pelos participantes dos eventos, mediante a identificação por nome de login e uma senha. O organizador possui acesso as principais funcionalidades como: cadastrar, alterar, excluir e consultar sobre algum determinado

evento. Enquanto que a funcionalidade permitida aos participantes se restringia apenas a realizar consultas sobre determinado evento.

A partir disso foi desenvolvido o presente projeto, mas com o objetivo de desenvolver um sistema voltado totalmente para o organizador do evento. Um sistema que auxiliasse na organização do evento por parte do organizador, onde o participante não teria nenhum acesso. Portanto, a primeira fase do presente projeto consistiu em analisar o projeto anterior e realizar mais pesquisas sobre o assunto.

A segunda parte do projeto consistiu na modelagem e desenvolvimento do SGE. Para isso a metodologia utilizada foi a pesquisa-ação que segundo [Gil 1991] citando Thiollent¹ é um tipo de pesquisa com base empírica que é concebida e realizada em estreita associação com uma ação ou com a resolução de um problema coletivo e no qual os pesquisadores e participantes representativos da situação ou do problema estão envolvidos de modo cooperativo ou participativo.

Foi utilizada essa metodologia não somente pela sua flexibilidade, mas também devido aos aspectos referentes à pesquisa propriamente dita, pois envolveu a ação do pesquisador e da empresa interessada Berin Eventos.

3.2 Ambiente de trabalho

Como a primeira fase de desenvolvimento do projeto se concentrou principalmente na pesquisa e obtenção de informações os ambientes de trabalho escolhidos foram os laboratórios do Departamento de Ciência da Computação - DCC e a Biblioteca Central da Universidade Federal de Lavras - UFLA, por possuírem acesso a Internet e periódicos para pesquisa e a empresa organizadora de eventos Berin Eventos para obtenção de mais informações referentes ao projeto.

O ambiente da segunda fase foi praticamente realizado nos laboratórios do Departamento de Ciência da Computação - DCC, pois essa etapa consistiu no desenvolvimento do sistema de gestão de eventos. Esse ambiente foi escolhido devido à possibilidade de realizar mais pesquisas complementares caso houvesse necessidade.

¹Thiollent, Michel. Metodologia da pesquisa-ação. São Paulo: Cortez, 1985.

3.3 Ferramentas utilizadas

3.3.1 Ferramenta de programação Delphi

O sistema para gestão de eventos foi desenvolvido para ser utilizado pelo organizador de um evento utilizando o sistema operacional Windows². Para isso foi utilizada a ferramenta de programação visual Delphi³, pois segundo [Teixeira 2000] ele possui as seguintes características:

- A qualidade do ambiente de desenvolvimento visual - Geralmente, o ambiente de desenvolvimento visual pode ser dividido em três componentes: o editor, o depurador e o Form Designer. Como na maioria das modernas ferramentas RAD (Rapid Application Development - desenvolvimento rápido de aplicação), esses três componentes funcionam em harmonia enquanto você projeta uma aplicação. O editor do Delphi pode ser comparado aos utilizados por outras ferramentas. Ele possui uma tecnologia chamada de CodeInsight que poupa tempo na digitação auto completando as palavras no momento da digitação. A depuração do Delphi permite que as janelas sejam colocadas e travadas onde o programador quiser durante a depuração e possibilitando que esse estado seja salvo como um parâmetro de desktop. O Form Designer é um recurso exclusivo das ferramentas RAD, como Delphi. Uma característica do Form Designer do Delphi, que o torna realmente especial, é o fato de que o Delphi é construído em cima de uma verdadeira estrutura orientada a objeto. Por essa razão, as alterações que você faz nas classes básicas irão se propagar para qualquer classe ancestral.
- A velocidade do compilador contra a eficiência do código compilado - Um compilador rápido permite que você desenvolva softwares de modo incremental e dessa forma possa fazer freqüentes mudanças no seu código-fonte, recompilando, testando, alterando, recompilando, testando novamente e assim por diante, o que lhe proporciona um ciclo de desenvolvimento muito eficiente.
- A potência da linguagem de programação contra sua complexidade - A linguagem usada pelo Delphi é o object Pascal, essa linguagem consegue manter um bom equilíbrio entre complexidade e potência. Ela limita os recursos disponíveis com o objetivo de induzir o programador a um projeto lógico.

²Sistema operacional desenvolvida pela Microsoft.

³Ferramenta de programação desenvolvida pela Borland.

- A flexibilidade e a capacidade de redimensionar a arquitetura do banco de dados - O Delphi mantém o que pensamos ser uma das mais flexíveis arquiteturas de banco de dados de qualquer ferramenta. Na prática, o gerenciador de banco de dados BDE(Borland Database Engine) funciona maravilhosamente bem e executa bem a maioria das aplicações em uma ampla gama de plataformas de banco de dados local, cliente/servidor.
- O projeto e os padrões de uso impostos pela estrutura - A habilidade para manipular componentes em tempo de projeto, projetar componentes e herdar o comportamento de outros componentes usando técnicas orientadas a objeto (OO) é de fundamental importância para o nível de produtividade do Delphi.
- Criação avançada de protótipos - O Delphi se destaca na primeira faceta do desenvolvimento de aplicação, que tem sido a ruína de muitos programadores do Windows: o projeto da interface do usuário.O Delphi lhe permite usar os controles personalizados para a criação de belas interfaces de usuário em pouco tempo agilizando o desenvolvimento de protótipos de aplicação.

3.3.2 Bando de dados Paradox

Como a organização de um evento gera uma grande quantidade de informação, a necessidade de realizar o seu armazenamento de forma que suas informações não se encontrem isoladas umas das outras é de suma importância. Além de uma forma adequada para definir o armazenamento destas informações, os usuários desejam realizar consultas em um determinado subconjunto de dados, atualizar ou modificar a estrutura dos dados e eliminar informações não mais necessárias.

Para resolver esse problema surgiu a tecnologia de Banco de Dados que segundo [Date 1991] nada mais é do que um sistema de manutenção de registros por computador. O próprio banco de dados pode ser considerado uma espécie de sala de arquivo eletrônica - ou seja, um depósito de um conjunto de arquivos de dados computadorizados que oferece diversos recursos ao usuário, possibilitando-lhe a realização de várias operações, incluindo, entre outras, as seguintes:

- A adição de novos (vazios) arquivos ao banco de dados;
- A inserção de novos dados nos arquivos existentes;
- A recuperação de dados dos arquivos existentes;

- A atualização de dados nos arquivos existentes;
- A eliminação de dados nos arquivos existentes;
- A renovação permanente de arquivos existentes (vazios ou outros) do bando de dados.

Para a criação do banco de dados do sistema foi utilizado o Paradox que é um sistema-cliente no qual os dados são completamente manipulados por clientes individuais. Quando o usuário quiser que os dados sejam lidos ou manipulados, eles devem ser transportados para a aplicação. Cada aplicação manipula todo o processamento por si só.

Em suma, o Paradox é veloz em um disco rígido local, mas a sua performance diminui drasticamente em uma rede. A rede fica sobrecarregada, com um grande volume de processamentos sendo realizado repetidamente na máquina cliente. Mesmo em uma rede pequena a performance é rapidamente afetada quando um novo usuário é conectado.

Como o Paradox foi basicamente desenvolvido para um único usuário, sendo que características que visavam permitir o atendimento a múltiplos usuários foram depois sendo adicionadas. Como objetivo do projeto inicialmente foi desenvolver um programa mono-usuário o Paradox atendeu muito bem às necessidades do desenvolvimento.

Capítulo 4

Resultados e Discussão

Este capítulo apresenta os resultados do desenvolvimento do sistema de gestão de eventos, começando com a descrição de suas funcionalidades, depois a modelagem dos dados, desenvolvimento do sistema e a explicação do seu funcionamento de acordo com sua interface.

4.1 Funcionalidades

Como a organização de um evento gera uma grande quantidade de informações a necessidade de organização e agilidade com relação à utilização dessas informações é muito importante. Visando isso foi desenvolvido o sistema para a gestão de eventos para organizar as informações de forma a agilizar a inserção, alteração e consulta desses dados.

O desenvolvimento do sistema foi focado para ser utilizado exclusivamente pelos organizadores do evento, portanto a sua principal funcionalidade está no cadastramento das informações sobre os organizadores, participantes, locais de realização, responsáveis pelos locais e outros dados referentes ao tipo de evento a ser organizado. Essa funcionalidade tenta armazenar o máximo de informações sobre cada detalhe do evento.

Todos as funcionalidades como alterar, consultar e excluir os dados do sistema foram englobados na funcionalidade de cadastrar, visando minimizar o tempo gasto para se acessar uma outra funcionalidade. A consulta citada é apenas para se ter uma visão simplificada dos dados no momento do cadastro.

Na funcionalidade de consultas referentes a todos as informações cadastradas no banco de dados do sistema, foi acrescentadas a possibilidade de alteração, ex-

clusão e geração de relatórios sobre os dados cadastrados. Uma consulta pode ser mais geral como listar todos os eventos organizados, todos os participantes de um evento, todos os cursos cadastrados, não apresentando todos os detalhes. Ou pode ser mais detalhada, mostrando os dados detalhados de um participante específico como: nome, RG, CPF, endereço, telefone de contato, e-mail e profissão.

A funcionalidade de geração de relatórios referentes é acessada a partir das consultas que podem ser realizadas. Isso é muito útil para se ter um maior controle sobre a organização do evento, a partir de uma listagem de todos os participantes, organizadores, patrocinadores de um determinado evento. Podem ser gerados também relatórios sobre a programação de um evento mostrando dados como horário, local de realização, número de vagas, valor de um curso, palestra ou seminário.

Como o sistema realiza o cadastro de muitas informações pessoais e bem detalhadas sobre a organização de um evento, foi implementada uma função de criação de um arquivo de backup. Onde todos os dados referentes ao evento são compactados e podem ser armazenados em algum outro lugar como em disquetes ou em um cd gravável.

4.2 Modelagem dos dados

Como o objetivo principal do aplicativo é a gestão de informações referentes a um evento, a criação de um banco de dados bem estruturado que armazene essas informações de forma que as inserções, consultas e alterações sejam feitas de forma mais otimizada possível é muito importante.

Para a modelagem dos dados foi utilizado o MER (Modelo Entidade - Relacionamento), pois segundo [Rocha 2003], citando Ferreira¹ o MER é um modelo baseado na percepção do mundo real, que consiste em um conjunto de objetos básicos chamados de entidades e nos relacionamentos entre esses objetos. Além do mais o MER foi desenvolvido segundo [Korth 1989] a fim de facilitar o projeto de banco de dados permitindo a especificação de um esquema de empreendimento. Tal esquema representa a estrutura lógica global do banco de dados.

Segundo [Rocha 2003] citando Ferreira, o MER possui as seguintes características:

- Modela regras de negócio e não a implementação. A modelagem é dos dados requeridos para o negócio, baseado nas funções do sistema atual ou

¹Ferreira, Paulo. Banco de Dados: Modelagem Conceitual. Disponível na World Wide Web: <<http://wzero.com.br/paulo/feevale/aula02.pdf>> [04/Nov/2003].

a ser desenvolvido. Para modelar um negócio, é necessário conhecer em detalhes sobre do que se trata;

- Possui uma sintaxe robusta, bem definida;
- Técnica amplamente difundida e utilizada. Atualmente, a maioria dos bancos de dados disponíveis no mercado utiliza a abordagem relacional como modelo de dados.

Devido a essas características e simplicidade dessa modelagem ela foi utilizada para representar as informações necessárias para construção do banco de dados do sistema. Essa modelagem pode ser observada com mais detalhes na Figura 4.1.

O detalhamento de cada entidade pertencente ao modelo é apresentado no Apêndice A, onde é mostrando todos os detalhes de cada tabela utilizada no sistema. São apresentadas as tabelas de evento e do responsável pelo evento nas tabelas 4.1 e 4.2 como exemplo.

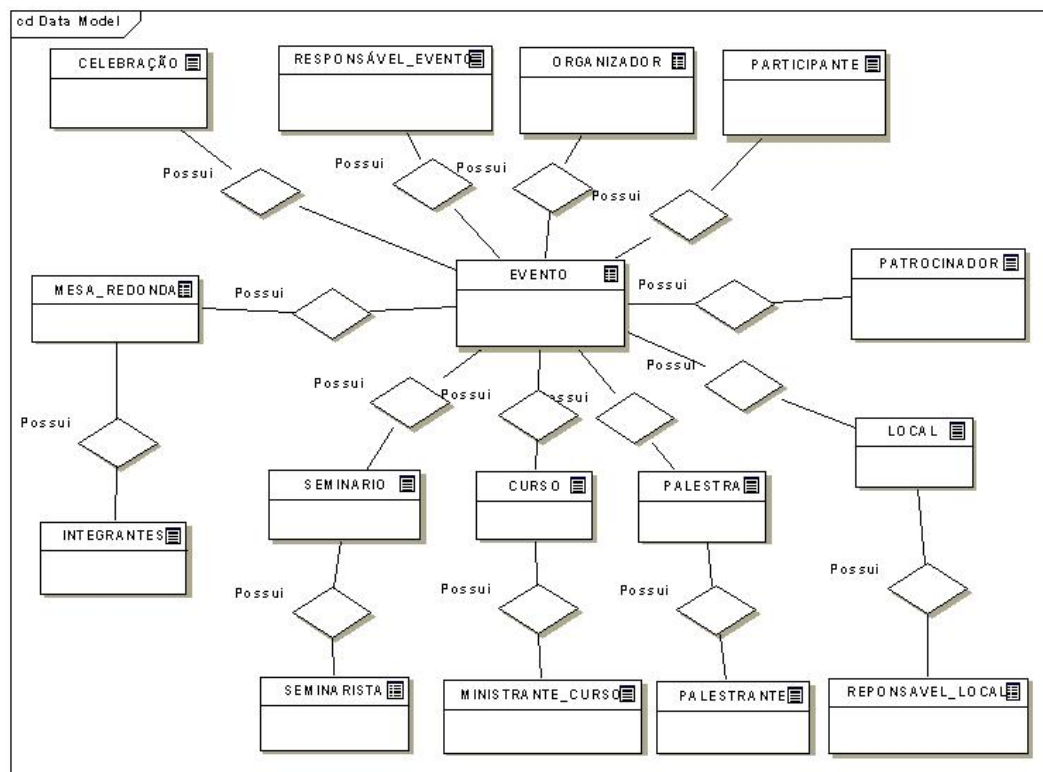


Figura 4.1: Modelo Entidade Relacionamento do banco de dados

- Tabela EVENTO:

ATRIBUTO	TIPO	DESCRIÇÃO
1- cod_evento	Inteiro	Código de controle dos eventos - Chave
2- nome_evento	String	Nome do evento
3- tipo_evento	String	Tipo de evento
4- data_inscri	Data	Data de inscrição
5- data_reali	Data	Data de realização
6- taxa_inscri	Inteiro	Valor da inscrição
7- publico_alvo	String	Público alvo

Tabela 4.1: Tabela de eventos

- Tabela RESPONSAVEL_EVENTO:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_resp	Inteiro	Cadastro de pessoa física – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - rg_resp	String	Número da carteira de identificação
4 - nome_resp	String	Nome do responsável
5 - rua_resp	String	Nome da rua do responsável
6 - num_resp	Inteiro	Número da residência do responsável
7 - comp_resp	String	Complemento do endereço como apto
8 - bairro_resp	String	Nome do bairro do responsável
9 - cidade_resp	String	Nome da cidade do responsável
10 - uf_resp	String	Unidade federativa
11 - cep_resp	Inteiro	Cep do responsável
12 - tel1_resp	Inteiro	Telefone residencial do responsável
13 - tel2_resp	Inteiro	Telefone comercial do responsável
14 - cel_resp	Inteiro	Celular do responsável
15 - mail_resp	String	E-mail do responsável

Tabela 4.2: Tabela dos dados pessoais do responsável pelo evento

4.3 Desenvolvimento do sistema

Feito a análise dos requisitos e a modelagem dos dados o próximo passo foi o desenvolvimento do aplicativo. Para auxiliar no desenvolvimento foi utilizado a UML(Unified Model Language) que segundo [Rocha 2003] citando Furlan² é a linguagem padrão para especificar, visualizar, documentar e construir artefatos de um sistema e pode ser utilizada com todos os processos ao longo do ciclo de desenvolvimento e através de diferentes tecnologias de implementação.

De acordo com [Rocha 2003], a UML fornece vários diagramas para modelagem de sistemas, os quais apresentam um resultado passível de compreensão pelo usuário e programador, facilitando o desenvolvimento da análise, projeto e implementação do sistema.

O diagrama utilizado na modelagem do aplicativo foi o diagrama de casos de uso que é uma técnica usada para descrever e definir os requisitos funcionais de um sistema. Eles são escritos em termos de atores externos, casos de uso e o sistema modelado. Os atores representam o papel de uma entidade externa ao sistema como um usuário, um hardware, ou outro sistema que interage com o sistema modelado. Os atores iniciam a comunicação com o sistema através dos casos de uso, onde representa uma seqüência de ações executadas pelo sistema e recebe do ator que utiliza dados tangíveis de um tipo ou formato já conhecido, e o valor de resposta da execução de um caso de uso (conteúdo) também já é de um tipo conhecido, tudo isso é definido juntamente com os casos de uso através de texto de documentação.

Como o sistema possui apenas um tipo de usuário, que é representado pelo responsável pelo evento, o diagrama de casos de uso da Figura 4.2 representa todas as funcionalidades que o usuário poderá utilizar no sistema. A funcionalidade de gerenciar definida no modelo está no sentido de cadastrar, alterar, excluir, buscar e gerar relatórios sobre alguma informação no banco de dados.

²Furlan, José Davi. Modelagem de Objetos através da UML. São Paulo: Makron Books, 1998.

O responsável pelo evento é o usuário principal com o poder de permitir que outro usuário utilize o sistema através do cadastramento de um login de usuário e uma senha. Feito isso todos os usuários cadastrados tem acesso a todas as funcionalidades do sistema.

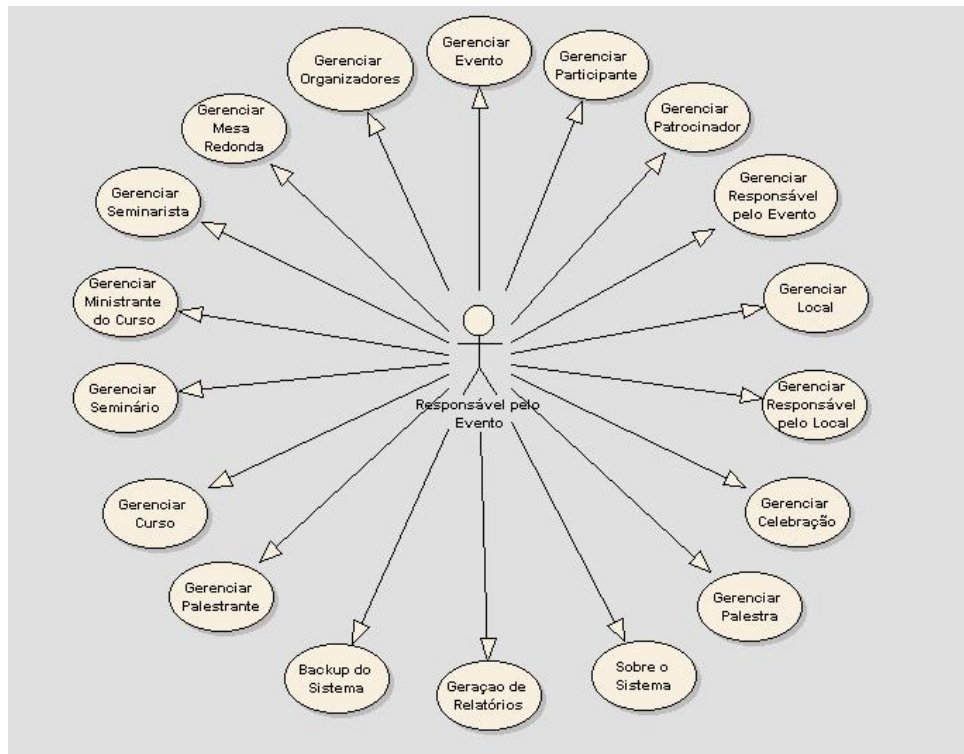


Figura 4.2: Diagrama de casos de uso

4.4 Descrição do Sistema

O SGE possibilita o cadastramento das informações geradas pela organização de um evento. As primeiras informações cadastradas no sistema são aquelas básicas a todo tipo de evento, que consiste em definir o nome do evento, o seu tipo, as datas de inscrição e realização, local onde será realizado e o responsável pelo evento. Feito isso as outras informações referentes aos detalhes do evento poderão

ser cadastradas, como: cursos, palestras, seminários, mesas redondas, celebrações, participantes, organizadores e patrocinadores.

A partir do momento em que o usuário está cadastrando dados sobre um organizador, por exemplo, ele também tem a possibilidade de estar alterando, consultando ou excluindo essas informações do bando de dados. Essa facilidade foi implementada visando agilizar a inserção e alteração dos dados no sistema, pois caso cada funcionalidade dessas fosse implementada separadamente a demanda de tempo seria enorme devido à necessidade de se ativar outra funcionalidade.

As consultas realizadas no sistema são definidas pelo nome do evento selecionado pelo usuário. Depois de selecionado o evento a ser consultado é apresentado todos os detalhes cadastrados como: cursos, palestras, seminários, mesas redondas e celebrações, caso pertençam ao evento. É dada ao usuário a possibilidade de estar incluindo, alterando e excluindo algum dado no momento da consulta.

A partir do momento que o usuário está consultando algum evento ele tem a possibilidade de gerar relatórios sobre os seus detalhes. Devido a grande variedade de relatórios que poderiam ser gerados a partir dos dados cadastrados no sistema foram definidos alguns relatórios como listas dos participantes, dos organizadores, dos patrocinadores, dos cursos oferecidos, das palestras, dos seminários e da programação.

Todas as funcionalidades citadas serão detalhadas nas próximas seções a partir da explicação das interfaces do sistema.

4.4.1 Tela de login do sistema

Como o SGE possui uma grande quantidade de informação pessoal referente aos participantes de um evento, a implementação de um acesso restrito aos usuários foi feita através de login de sistema. Como foi citado anteriormente, somente o organizador do evento possui o domínio de acesso ao sistema e a possibilidade de deixar que outro usuário também acesse mediante o seu cadastramento. A tela de login é mostrada na Figura 4.3 e o aviso de senha ou login incorreto é mostrado na Figura 4.4.

4.4.2 Tela principal

Na tela principal mostrada na Figura 4.5, são apresentadas todas as funcionalidades do sistema, elas estão agrupadas em forma de um 'menu' na parte superior da tela. As principais funcionalidades foram colocadas na lateral esquerda em forma de

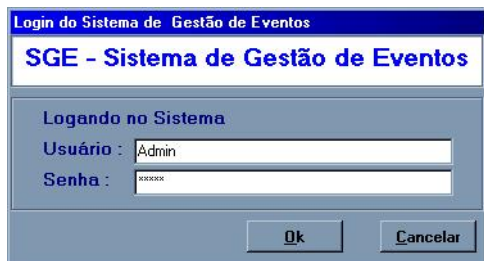


Figura 4.3: Menu de cadastros



Figura 4.4: Menu de consultas

botões para agilizar o seu acesso. O 'menu' com as funcionalidades de cadastro e consulta citadas acima pode ser mais bem visualizado nas figuras 4.6 e 4.7.

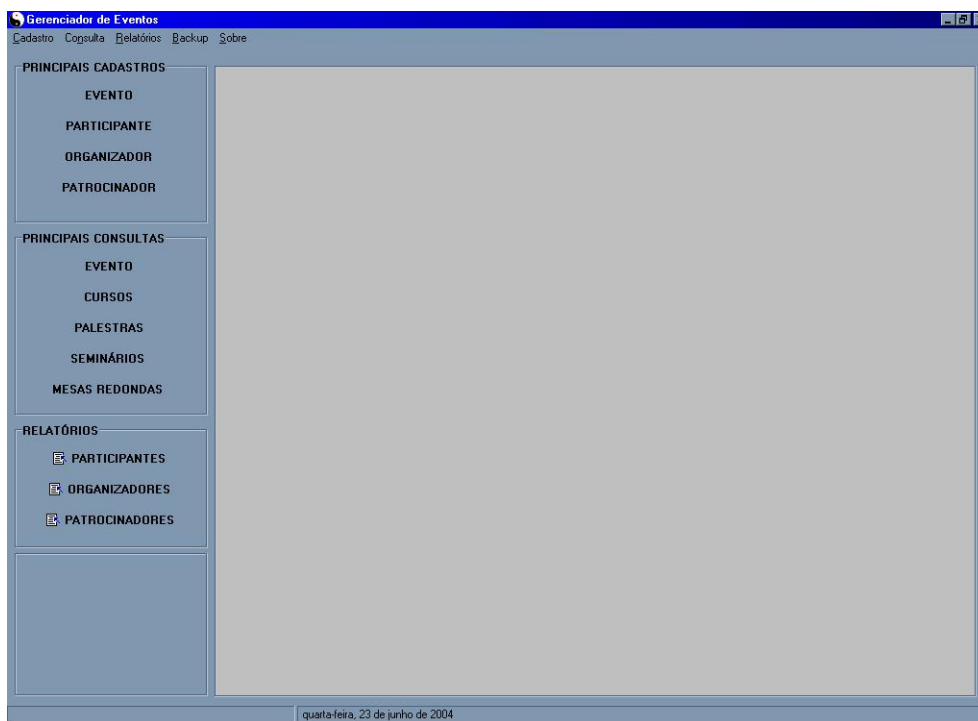


Figura 4.5: Tela principal do sistema

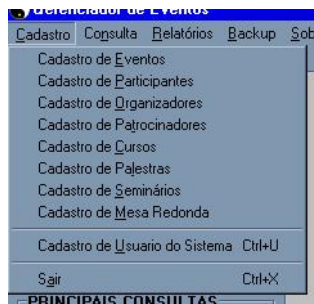


Figura 4.6: Menu de cadastros

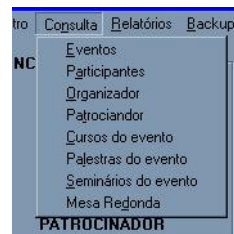


Figura 4.7: Menu de consultas

4.4.3 Telas de cadastros

A tela do cadastro de um evento é composta basicamente de todos os dados comuns a todos os tipos de eventos, como nome, datas de inscrição e realização, responsável e local da realização. Na parte superior da tela é mostrada uma lista de todos os eventos cadastrados no sistema. Os dados sobre o evento são mostrados de acordo com o evento selecionado na lista da tabela de eventos cadastrados de acordo com a Figura 4.8.

A tela possui mais abaixo, os botões para o cadastro do responsável pelo evento e local do evento, que são mostrados nas Figuras 4.9 e 4.11. Eles funcionarão se existir algum evento cadastrado, caso contrário é enviada uma mensagem de aviso ao usuário sobre não existir nenhum evento cadastrado. Os próximos botões correspondem às funcionalidades de inclusão, alteração, exclusão, gravação e cancelamento dos dados sobre o evento. A exclusão é seguida de uma mensagem mostrada na Figura 4.10 de aviso para confirmação. E para finalizar, os botões mais abaixo possuem as funções de continuar o cadastro ou sair da tela. O botão de 'continuar cadastro' será mais detalhado na próxima seção.

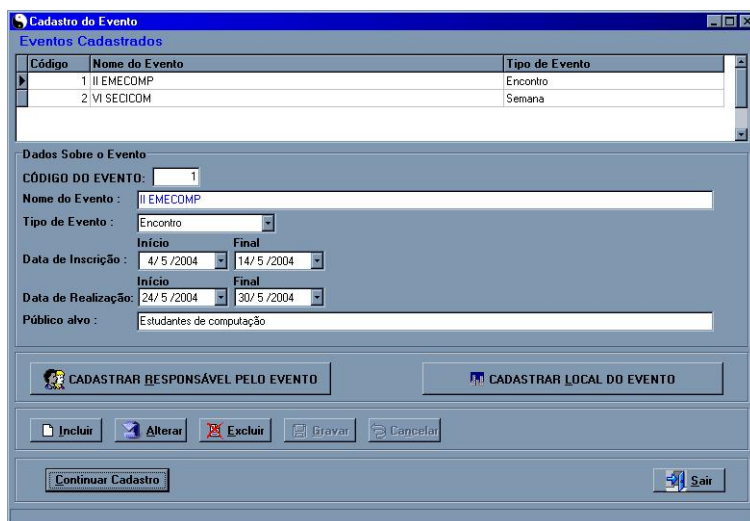


Figura 4.8: Cadastro de um evento

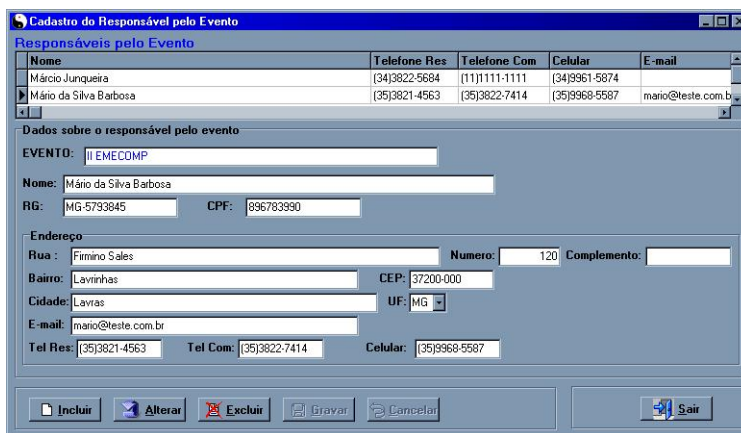


Figura 4.9: Cadastro do(s) responsável(s) pelo evento

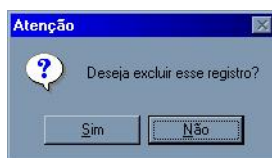


Figura 4.10: Mensagem de confirmação para exclusão

Na tela de cadastro do local do evento como mostra a Figura 4.11 existe um botão para se cadastrar as informações referentes ao responsável pelo local. Essa tela é semelhante às outras já mostradas possuindo uma lista de registros já cadastrados e todos os botões para se realizar alguma alteração nos dados. Caso não sejam preenchidos os campos obrigatórios é enviada uma mensagem de erro de acordo com a Figura 4.12.

Nome	Cidade	UF	Telefone
Universidade Federal de Lavras - UFLA	Lavras	MG	(35)3826-4589

Dados sobre o local do evento

EVENTO: VI SECICOM

Nome do Local: Universidade Federal de Lavras - UFLA

Endereço

Rua: Campus da Universidade Federal de Lavras Número: Complemento:

Bairro: CEP: 37200-000

Cidade: Lavras UF: MG

Telefone de contato: (35)3826-4589

Cadastrar Responsável pelo Local

Incluir Alterar Excluir Gravar Cancelar Sair

Figura 4.11: Cadastro do local de realização do evento

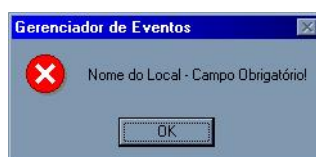


Figura 4.12: Mensagem de campo obrigatório

Tela para continuar cadastro

A tela de continuar cadastro pode ser acessada pelo botão da tela de cadastro de evento ou pelo menu na opção de cadastro. Nesta tela estão englobados todos os cadastros em forma de páginas sobre os detalhes do evento, como: cursos, palestras, seminários, mesas redondas e celebrações. Todos possuem os botões de inclusão, alteração, exclusão, gravação e cancelamento dos dados a serem cadastrados mostrado na Figura 4.13.

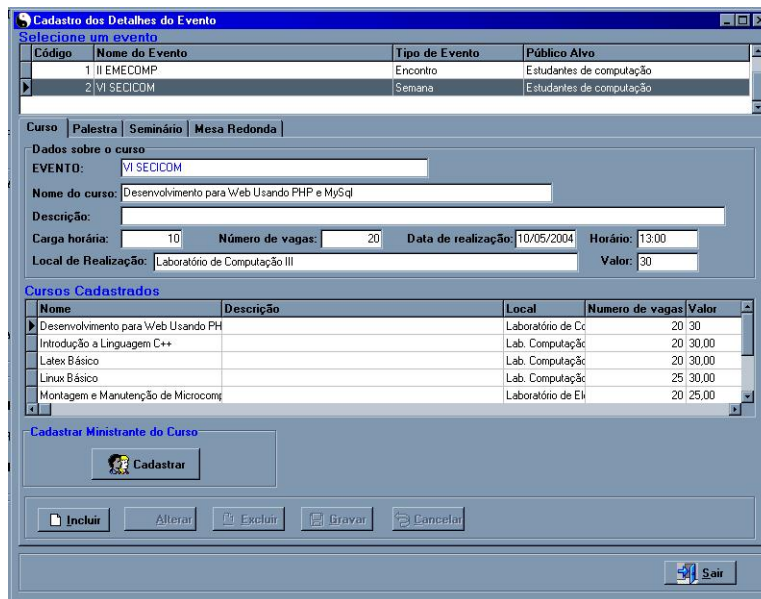


Figura 4.13: Cadastro dos detalhes do evento

Para se cadastrar um ministrante de um curso é necessário selecionar na tabela de cursos cadastrados e utilizar o botão de cadastro, feito isso uma outra tela aparecerá com todos os campos necessários para se cadastrá-lo, isso pode ser observado na Figura 4.14. Para todos os outros casos a funcionalidade é a mesma não necessitando ser mostrada.

Tela de cadastro de participantes, patrocinadores e organizadores

A tela de cadastro de participantes da Figura 4.15, patrocinadores e organizadores é semelhante à tela de cadastro dos detalhes do evento apresentado na seção anterior. O acesso a essa tela é feito através dos botões laterais de cadastro ou pelo 'menu de cadastro' da tela principal.

Cadastro do Ministrante do Curso

Nome	Telefone Res	Telefone Com	Celular	E-mail
X Samuel Pereira Dias	(35)3822-6598			samuel@teste.com.br

Dados sobre o ministrante do curso

CURSO: Desenvolvimento para Web Usando PHP e MySQL

Nome: Samuel Pereira Dias

RG: 3945734995 CPF: 8059438509438

Endereço

Rua: José Correia de Abreu Número: 254 Complemento: apto 250

Bairro: NSr de Lourdes CEP: 37200-000

Cidade: Lavras UF: MG País: Brasil

E-mail: samuel@teste.com.br

Tel Res: (35)3822-6598 Tel Com: Celular:

Instituição: Universidade Federal de Lavras - UFLA

Mini currículo: Coordenador de Informática do DCC

Incluir Alterar Excluir Gravar Cancelar Sair

Figura 4.14: Cadastro do ministrante de um curso

Cadastro dos detalhes do evento

Selecione um evento

Código	Nome do Evento	Tipo de Evento	Público Alvo
1	II EMECOMP	Encontro	Estudantes de computação
X 2	VI SECICOM	Semana	Estudantes de computação

Patrocinador | Participante | Organizador

Dados sobre o participante

EVENTO: VI SECICOM

Nome: Fabiana A. Costa

RG: 456546 CPF: 46546546546

Endereço

Rua: Firmino Sales Número: 556 Complemento:

Bairro: Centro CEP: 35622-000

Cidade: Diamantina UF: MG País:

E-mail: fabiana@teste.com.br

Tel Res: (31)6582-1144 Tel Com: () - Celular: () -

Profissão: Estudante

Participantes

Nome	Telefone Residencial	Telefone Comercial	Celular	E-mail
Décio Bras Rezende	(34)3851-0162	() -	() -	decio@teste.com.br
X Fabiana A. Costa	(31)6582-1144	() -	() -	fabiana@teste.com.br

Incluir Alterar Excluir Gravar Cancelar Sair

Figura 4.15: Cadastro dos participantes, patrocinadores e organizadores

Tela de cadastro dos usuários do sistema

A tela da Figura 4.16 é acessada apenas pelo 'menu' da tela principal se localizando na parte de cadastro. Como em qualquer outro cadastro apresentado anteriormente é possível incluir, alterar, excluir e gravar um novo usuário do sistema.

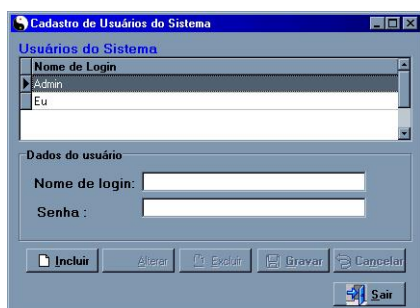


Figura 4.16: Cadastro de um novo usuário

4.4.4 Telas de consultas

As consultas foram feitas para se ter uma dimensão maior da organização dos dados cadastrados. Elas se baseiam em mostrar listagem dos dados referentes a algum evento selecionado. Isso é feito através da tela mostrada na Figura 4.17 que é acessada através do 'menu' de consultas ou pelo botão lateral da tela principal.

Selecionado o evento e acionado o botão de localizar é apresentada a tela da Figura 4.18 que possui uma listagem dos detalhes do evento como: local da realização, responsável, cursos, palestras, seminários, mesas redondas e celebrações caso existam. Cada detalhe pode ser alterado através do botão presente em seu campo. As telas de alteração são semelhantes as apresentadas na seção de cadastro.

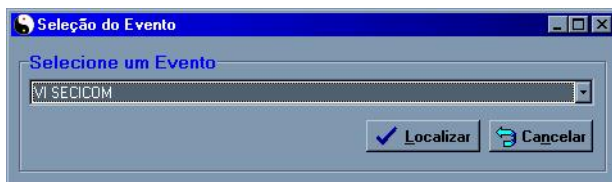


Figura 4.17: Seleção de um evento

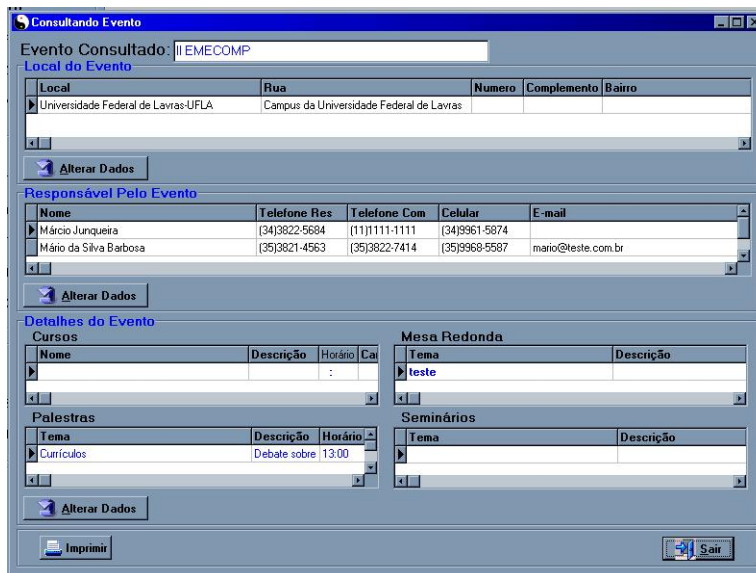


Figura 4.18: Detalhes sobre o evento selecionado

Telas de consultas sobre os participantes, organizadores e patrocinadores

As consultas referentes aos participantes, organizadores e patrocinadores possuem a interface apresentada na Figura 4.19. Nesta tela a consulta é feita selecionando o evento desejado e de acordo com isso são apresentados a listagem de todos os participantes do evento selecionado. A localização também pode ser realizada digitando-se o nome do participante no campo especificado. A localização é feita no decorrer da digitação realizada pelo usuário através da seleção do participante na listagem apresentada. Existe a possibilidade de organizar a busca com relação aos participantes da mesma cidade, profissão ou em ordem alfabética pelo nome.

A tela de consulta apresenta a possibilidade de acessar os dados do participante selecionado no modo de ficha. Esse modo é semelhante às fichas de cadastro dos participantes, permitindo a visualização detalhada de seus dados. Os dois modos de visualização possuem o botão de impressão que gera um relatório mostrado na Figura 4.20 da listagem de todos os participantes ou a ficha de um em específico. Todas essas funcionalidades se aplicam às consultas dos organizadores e patrocinadores.

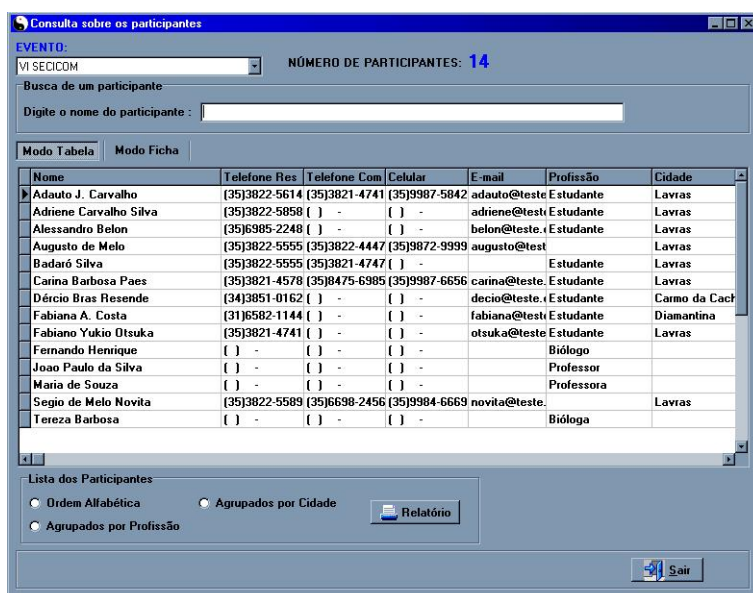


Figura 4.19: Consulta sobre os participantes de um evento

Os relatórios são compostos de um cabeçalho onde é mostrado o nome do evento, o seu tipo, o público alvo e o número de participantes no caso da consulta de participantes. Isso é válido para as consultas dos organizadores e patrocinadores também. No rodapé é apresentado o número da página e a data da criação do relatório.

Relatório

Lista de Participantes do evento VI SECICOM

Tipo de Evento Semana

Público Alvo Estudantes de computação **Número de Participantes** 14

Nome	Telefone	E-mail	Cidade	Profissão
Adauto J. Carvalho	(35)3822-5614	adauto@teste.com.br	Lavras	Estudante
Adriene Carvalho Silva	(35)3822-5858	adriene@teste.com.br	Lavras	Estudante
Alessandro Belon	(35)6985-2248	belon@teste.com.br	Lavras	Estudante
Augusto de Melo	(35)3822-5555	augusto@teste.com.br	Lavras	
Badaró Silva	(35)3822-5555		Lavras	Estudante
Carina Barbosa Paes	(35)3821-4578	carina@teste.com.br	Lavras	Estudante
Décio Bras Resende	(34)3851-0162	decio@teste.com.br	Carmo da Cachoeira	Estudante
Fabiana A. Costa	(31)6582-1144	fabiana@teste.com.br	Diamantina	Estudante
Fabiano Yúki Otsuka	(35)3821-4741	otsuka@teste.com.br	Lavras	Estudante
Fernando Henrique	() -			Biólogo
Joao Paulo da Silva	() -			Professor
Maria de Souza	() -			Professora
Segio de Melo Novita	(35)3822-5589	novita@teste.com.br	Lavras	
Tereza Barbosa	() -			Bióloga

Figura 4.20: Relatório da listagem dos participantes de um evento

4.4.5 Tela de criação do arquivo de backup

A função de criação de backup é acessada a partir do 'menu' da tela principal. Essa tela é composta dos comandos de selecionar os arquivos que serão copiados, de criar o arquivo de backup compactado e de descompactar os arquivos conforme a Figura 4.21. A tela da Figura 4.22 é acionada com o acionamento do comando de descompactar arquivo.

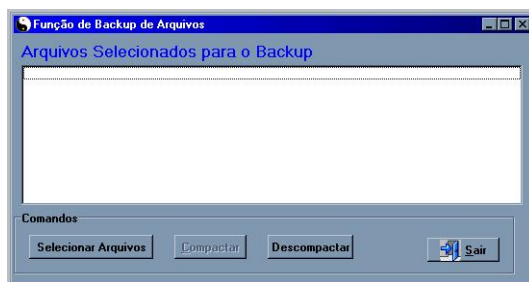


Figura 4.21: Menu de cadastros



Figura 4.22: Menu de consultas

Capítulo 5

Conclusões

Atualmente, o gerenciamento de informação se tornou fundamental para o sucesso ou fracasso de uma organização. E para isso cada vez mais pessoas têm necessidade de reciclar seus conhecimentos, trocar informações e tecnologia, não só para melhorar suas performances profissionais, empresariais e acadêmicas. E isso pode ser feito através de eventos como congressos, seminários, palestras, workshops e etc. Portanto, a organização de eventos se tornou um setor bastante promissor para empresas organizadoras de eventos.

Visando esse setor foi desenvolvido o Sistema para Gestão de Eventos que realizasse o cadastro, consulta e gerasse relatórios sobre as informações referentes aos vários tipos de eventos, com seus detalhes como cursos, palestras, participantes etc. Como o sistema foi desenvolvido para ser genérico, ou seja, o sistema foi feito para cadastrar desde uma simples celebração a um congresso. Como a quantidade de informações sobre um evento é muito grande, a possibilidade de realizar cadastros, consultas e gerar relatórios é enorme, portanto foi feita uma restrição com relação aos tipos de eventos para que fosse possível definir os dados a serem cadastrados de forma consistente no curto prazo para o desenvolvimento do projeto.

Todas as funcionalidades foram implementadas com sucesso possibilitando ao usuário realizar inclusões, alterações, exclusões, consultas e geração de relatórios sobre todos os registros do banco de dados. Sua interface foi desenvolvida tentando dar o máximo de conforto ao usuário, utilizando uma nomenclatura bem definida e botões com símbolos sugestivos.

De acordo com os objetivos definidos para o projeto pode-se concluir que foram atingidos. Contudo várias melhorias podem ser realizadas futuramente para melhorar e complementar o sistema, tais como:

- Utilização de um banco de dados mais robusto do que o Paradox, ou seja, caso o sistema seja implementado para ser utilizado em rede recomenda-se utilizar um banco de dados que dê mais suporte a esse tipo de aplicação;
- Complementar as funcionalidades com a possibilidade de cadastro dos dados sobre os hotéis disponíveis e horários de ônibus;
- Implementar um sistema de controle de pagamentos e emissão de recibo;
- Realizar a inscrição de Sócio ou não-sócio local ou via internet, com possibilidade de pagamento via cartão, cheque, dinheiro ou boleto bancário.

Referências Bibliográficas

- [Cruz 1998] CRUZ, T. - Sistemas de informações gerenciais: tecnologia da informação e a empresas do século XXI, São Paulo, Atlas, 1998, 231 p.
- [Freitas 2003] FREITAS, Vinícius de M. - Modelagem e desenvolvimento de um software de controle de alunos e sócios e geração de relatórios para a APAE de Lavras - MG 2003 Monografia(Graduação em Ciência da Computação) - Universidade Federal de Lavras - MG
- [Mye 1992] BRAD, A. Myers Mary Beth Rosson - Survey on User Interface Programming in Human Factors in Computing Systems CHI'92 Conference Proceedings, May 1992, Monterey, California, pp. 195-202.
- [Pressman 1995] PRESSMAN, Roger - S. Engenharia de software / Roger S. Pressman; - São Paulo : Makron Books, 1995
- [Souza 2001] SOUZA, Reginaldo Ferreira de - Sistemas de Informações Gerenciais - Lavras; DCC/UFLA, 2001.
- [Rezende 2002] REZENDE, Denis Alcides - Engenharia de software e sistemas de informação/ Denis Alcides Rezende. - 2.ed.- Rio de Janeiro: Brasport, 2002
- [Kiko 2003] Disponível em <http://www.async.com.br/kiko/ware.html>. Acessado em 20/10/2003
- [Rithcher 2003] BAPTISTE JUNIOR, Michel-Aange Rithcher Jean - Desenvolvimento de um aplicativo para gestão de eventos utilizando a tecnologia J2SE 2003 Monografia(Graduação em Ciência da Computação) - Universidade Federal de Lavras - MG

- [Sommerville 2003] SOMMERVILLE, Ian - Engenharia de software/ Ian Sommerville. - São Paulo: Addison Wesley, 2003
- [Teixeira 2000] TEIXEIRA, Steve - Delphi 5, guia do desenvolvedor / Steve Teixeira, Xavier Pacheco ; tradução de Daniel Vieira. - Rio de Janeiro : Campus, 2000
- [Xchart 2003] Disponível em <http://www.dcc.unicamp.br/proj-chart/start/porque.html>. Acessado em 20/10/2003
- [Zambalde 2003] ZAMBALDE, André Luiz - Material de aula de Interface Homem Maquina encontrado em <http://www.comp.ufla.br/zambalde/ihtm/aulas/>. Acessado em 26/09/2003
- [Gil 1991] GIL, Antônio Carlos, 1946 - Como elaborar projetos de pesquisa / Antônio Carlos Gil. - 3.ed. - São Paulo : Atlas, 1991.
- [Date 1991] DATE, C.J., 1941 - Introdução a sistemas de bancos de dados / C. J. Date; tradução (da 4^o edição original) de Contexto Traduções. - Rio de Janeiro: Campus, 1991 Tradução de: An Introduction to database systems.
- [Rocha 2003] ROCHA, Paulo Rogério Souza - Carteira de Trabalho On-Line - MG 2003 Monografia (Graduação em ciência da Computação) - Universidade Federal de Lavras
- [Adriana 2003] Disponível em <http://gwarh.com.br/p62.htm>. Acessado em 15/09/2003
- [Korth 1989] KORTH, Henry F. - Sistema de banco de dados / Henry F. Korth & Abraham Silberschatz; revisão técnica Routh Terada. - São Paulo : MacGraw-hill, 1989
- [Teixeira 2000] TEIXEIRA, Steve - Delphi 5, guia do desenvolvedor / Steve Teixeira, Xavier Pacheco; tradução de Daniel Vieira. Rio de Janeiro: Campus, 2000

Apêndice A

Detalhamentos sobre as entidades do Modelo Entidade Relacionamento

- Tabela CURSO:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - <u>cod_curso</u>	Inteiro	Código do curso – Chave
2 - <u>cod_evento</u>	Inteiro	Código do evento – Chave estrangeira
3 - nome_curso	String	Nome do curso
4 - descri_curso	String	Descrição do curso
5 - hor_curso	String	Horário de realização do curso
6 - carga_hor_curso	Inteiro	Carga horária do curso
7 - data_curso	Data	Data de realização do curso
8 - local_curso	String	Local de realização do curso
9 - num_vagas_curso	Inteiro	Número de vagas do curso
10- valor_curso	Inteiro	Valor do curso

Tabela A.1: Tabela dos dados sobre um curso

- Tabela MINISTRANTE_CURSO:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_ministrante	Inteiro	Cadastro de pessoa física – Chave
2 - cod_curso	Inteiro	Código do curso – Chave estrangeira
3 - nome_ministrante	String	Nome do ministrante
4 - rg_ministrante	String	Número da carteira de identificação
5 - rua_ministrante	String	Nome da rua do ministrante
5 - num_ministrante	Inteiro	Número da residência do ministrante
6 - comp_ministrante	String	Complemento do endereço como apto
7 - bairro_ministrante	String	Nome do bairro do ministrante
8 - cidade_ministrante	String	Nome da cidade do ministrante
9 - uf_ministrante	String	Unidade federativa
10- pais_ministrante	String	País do ministrante
11- cep_ministrante	Inteiro	Cep do ministrante
12- tel1_ministrante	Inteiro	Telefone residencial do ministrante
13- tel2_ministrante	Inteiro	Telefone comercial do ministrante
14- cel_ministrante	Inteiro	Celular do ministrante
15- mail_ministrante	String	E-mail do responsável
16- institu_ministrante	String	Instituição do ministrante
17- minicurr_ministrante	String	Mini currículo do ministrante

Tabela A.2: Tabela dos dados pessoais do ministrante de um curso

- Tabela PALESTRA:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cod_palestra	Inteiro	Código da palestra – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - nome_palestra	String	Nome da palestra
4 - descri_palestra	String	Descrição da palestra
5 - hor_palestra	String	Horário de realização da palestra
6 - carga_hor_palestra	Inteiro	Carga horária da palestra
7 - data_palestra	Data	Data de realização da palestra
8 - local_palestra	String	Local de realização da palestra
9 - num_vagas_palestra	Inteiro	Número de vagas da palestra

Tabela A.3: Tabela dos dados de uma palestra

- Tabela PALESTRANTE:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_palestrante	Inteiro	Cadastro de pessoa física – Chave
2 - cod_palestra	Inteiro	Código da palestra – Chave estrangeira
3 - nome_palestrante	String	Nome do palestrante
4 - rg_palestrante	String	Número da carteira de identificação
5 - rua_palestrante	String	Nome da rua do palestrante
6 - num_palestrante	Inteiro	Número da residência do palestrante
7 - comp_palestrante	String	Complemento do endereço como apto
8 - bairro_palestrante	String	Nome do bairro do palestrante
9 - cidade_palestrante	String	Nome da cidade do palestrante
10 - uf_palestrante	String	Unidade federativa
11 - pais_palestrante	String	País do palestrante
12 - cep_palestrante	Inteiro	Cep do palestrante
13 - tel1_palestrante	Inteiro	Telefone residencial do palestrante
14 - tel2_palestrante	Inteiro	Telefone comercial do palestrante
15 - cel_palestrante	Inteiro	Celular do palestrante
16 - mail_palestrante	String	E-mail do palestrante
17 - institu_palestrante	String	Instituição do palestrante
18 - incurr_palestrante	String	Mini currículo do palestrante

Tabela A.4: Tabela dos dados pessoais de um palestrante

- Tabela PARTICIPANTE:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_part	Inteiro	Cadastro de pessoa física – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - nome_part	String	Nome do participante
4 - rg_part	String	Número da carteira de identificação
5 - rua_part	String	Nome da rua do participante
6 - num_part	Inteiro	Número da residência do participante
7 - comp_part	String	Complemento do endereço como apto
8 - bairro_part	String	Nome do bairro do participante
9 - cidade_part	String	Nome da cidade do participante
10 - uf_part	String	Unidade federativa
11 - pais_part	String	País do participante
12 - cep_part	Inteiro	Cep do participante
13 - tel1_part	Inteiro	Telefone residencial do participante
14 - tel2_part	Inteiro	Telefone comercial do participante
15 - cel_part	Inteiro	Celular do participante
16 - mail_part	String	E-mail do participante
17 - prof_part	String	Profissão do participante

Tabela A.5: Tabela dos dados pessoais de um participante

- Tabela PATROCINADOR:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - nome_patro	Inteiro	Nome do patrocinador - Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - rua_patro	String	Nome da rua do patrocinador
4 - num_patro	String	Número da residência do patrocinador
5 - comp_patro	String	Complemento do endereço como apto
6 - bairro_patro	Inteiro	Nome do bairro do patrocinador
7 - cidade_patro	String	Nome da cidade do patrocinador
8 - uf_patro	String	Unidade federativa
9 - pais_patro	String	País do patrocinador
10 - cep_patro	String	Cep do patrocinador
11 - tell_patro	String	Telefone residencial do patrocinador
12 - tel2_patro	Inteiro	Telefone comercial do patrocinador
13 - cel_patro	Inteiro	Celular do patrocinador
14 - mail_patro	Inteiro	E-mail do patrocinador
15 - pg_web_patro	String	Página na Internet do patrocinador

Tabela A.6: Tabela dos dados sobre o patrocinador

- Tabela ORGANIZADOR:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_org	Inteiro	Cadastro de pessoa física – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - nome_org	String	Nome do organizador
4 - rg_org	String	Número da carteira de identificação
5 - rua_org	String	Nome da rua do organizador
6 - num_org	Inteiro	Número da residência do organizador
7 - comp_org	String	Complemento do endereço como apto
8 - bairro_org	String	Nome do bairro do organizador
9 - cidade_org	String	Nome da cidade do organizador
10 - uf_org	String	Unidade federativa
11 - pais_org	String	País do organizador
12 - cep_org	Inteiro	Cep do organizador
13 - tell_org	Inteiro	Telefone residencial do organizador
14 - tel2_org	Inteiro	Telefone comercial do organizador
15 - cel_org	Inteiro	Celular do organizador
16 - mail_org	String	E-mail do organizador
17 - funcao_org	String	Função exercida pelo organizador

Tabela A.7: Tabela dos dados sobre um organizador

- Tabela LOCAL:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cod_local	Inteiro	Código do local – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - nome_local	String	Nome do local de realização do evento
4 - rua_local	String	Nome da rua do local
5 - num_local	Inteiro	Número do local
6 - comp_local	String	Complemento do endereço como apto
7 - bairro_local	String	Nome do bairro do local
8 - cidade_local	String	Nome da cidade do local
9 - uf_local	String	Unidade federativa
10 - pais_local	String	País do local
11 - cep_local	Inteiro	Cep do local
12 - tel_local	Inteiro	Telefone de contato do local

Tabela A.8: Tabela dos dados sobre o local do evento

- Tabela RESPONSÁVEL_LOCAL:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_resp_local	Inteiro	Cadastro de pessoa física – Chave
2 - cod_local	Inteiro	Código do local – Chave estrangeira
3 - nome_resp_local	String	Nome do responsável pelo local
4 - rua_resp_local	String	Nome da rua do responsável pelo local
5 - num_resp_local	Inteiro	Número da residência do responsável pelo local
6 - comp_resp_local	String	Complemento do endereço como apto
7 - bairro_resp_local	String	Nome do bairro do responsável pelo local
8 - cidade_resp_local	String	Nome da cidade do responsável pelo local
9 - uf_resp_local	String	Unidade federativa
10 - pais_resp_local	String	País do responsável pelo local
11 - cep_resp_local	Inteiro	Cep do responsável pelo local
12 - tel1_resp_local	Inteiro	Telefone residencial do responsável pelo local
13 - tel2_resp_local	Inteiro	Telefone comercial do responsável pelo local
14 - cel_resp_local	Inteiro	Celular do responsável pelo local
15 - mail_resp_local	String	E-mail do responsável pelo local

Tabela A.9: Tabela dos dados sobre o responsável pelo local do evento

- Tabela SEMINÁRIO:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cod_seminario	Inteiro	Código do seminário – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - tema_seminario	String	Tema do seminário
4 - descri_seminario	String	Descrição do seminário
5 - hor_seminario	Inteiro	Horário de realização do seminário
6 - carga_hor_seminario	Inteiro	Carga horária do seminário
7 - data_seminario	Data	Data de realização do seminário
8 - local_seminario	String	Local de realização do seminário
9 - num_vagas_seminario	Inteiro	Número de vagas do seminário

Tabela A.10: Tabela dos dados sobre um seminário

- Tabela SEMINARISTA:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cpf_seminarista	Inteiro	Cadastro de pessoa física – Chave
2 - cod_seminario	Inteiro	Código do seminário – Chave estrangeira
3 - nome_seminarista	String	Nome do seminarista
4 - rg_seminarista	String	Número da carteira de identificação
5 - rua_seminarista	String	Nome da rua do seminarista
6 - num_seminarista	Inteiro	Número da residência do seminarista
7 - comp_seminarista	String	Complemento do endereço como apto
8 - bairro_seminarista	String	Nome do bairro do seminarista
9 - cidade_seminarista	String	Nome da cidade do seminarista
10 - uf_seminarista	String	Unidade federativa
11 - pais_seminarista	String	País do seminarista
12 - cep_seminarista	Inteiro	Cep do seminarista
13 - tel1_seminarista	Inteiro	Telefone residencial do seminarista
14 - tel2_seminarista	Inteiro	Telefone comercial do seminarista
15 - cel_seminarista	Inteiro	Celular do seminarista
16 - mail_seminarista	String	E-mail do seminarista
17 - institu_seminarista	String	Instituição do seminarista
18 - minicurr_seminarista	String	Mini currículo do seminarista

Tabela A.11: Tabela dos dados sobre um seminarista

- Tabela MESA_REDONDA:

ATRIBUTO	TIPO	DESCRIÇÃO
1 - cod_mesa	Inteiro	Código da mesa redonda – Chave
2 - cod_evento	Inteiro	Código do evento – Chave estrangeira
3 - tema_mesa	String	Tema da mesa redonda
4 - descri_mesa	String	Descrição da mesa redonda
5 - hor_mesa	Inteiro	Horário de realização da mesa redonda
6 - carga_hor_mesa	Inteiro	Carga horária da mesa redonda
7 - data_mesa	Data	Data de realização da mesa redonda
8 - local_mesa	String	Local de realização da mesa redonda
9 - num_vagas_mesa	Inteiro	Número de vagas da mesa redonda

Tabela A.12: Tabela dos dados sobre uma mesa redonda

- Tabela INTEGRANTES:

ATRIBUTO	TIPO	DESCRIÇÃO
1. - cpf_integrante	Inteiro	Cadastro de pessoa física – Chave
2. - cod_mesa	Inteiro	Código da mesa redonda – Chave estrangeira
3. - nome_integrante	String	Nome do integrante da mesa
4. - rg_integrante	String	Número da carteira de identificação
5. - rua_integrante	String	Nome da rua do integrante da mesa
6. - num_integrante	Inteiro	Número da residência do integrante da mesa
7. - comp_integrante	String	Complemento do endereço como apto
8. - bairro_integrante	String	Nome do bairro do integrante da mesa
9. - cidade_integrante	String	Nome da cidade do integrante da mesa
10. - uf_integrante	String	Unidade federativa
11. - pais_integrante	String	País do integrante da mesa
12. - cep_integrante	Inteiro	Cep do integrante da mesa
13. - tell_integrante	Inteiro	Telefone residencial do integrante da mesa
14. - tel2_integrante	Inteiro	Telefone comercial do integrante da mesa
15. - cel_integrante	Inteiro	Celular do integrante da mesa
16. - mail_integrante	String	E-mail do integrante da mesa
17. - institu_integrante	String	Instituição do integrante da mesa
18. - minicurr_integrante	String	Mini currículo do integrante da mesa

Tabela A.13: Tabela dos dados sobre um integrante de uma mesa redonda