

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

PABLO DAPONT

**Palpigol – sistema Web para a criação de
disputas em grupos com base em resultados
de jogos desportivos**

Trabalho de Graduação.

Prof. Dr. Leandro Krug Wives
Orientador

Porto Alegre, novembro de 2009

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitora de Graduação: Profa. Valquiria Link Bassani

Diretor do Instituto de Informática: Prof. Flávio Rech Wagner

Coordenador do CIC: Prof. João César Netto

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

AGRADECIMENTOS

Acima de tudo, agradeço aos meus pais, irmã, tia e madrinha que jamais mediram esforços para me apoiar e auxiliar durante toda essa caminhada. Com certeza essa conquista pertence a cada um deles. À minha namorada que sempre compreendeu os momentos difíceis e esteve ao meu lado dando força para eu superar qualquer obstáculo que cruzasse meu caminho. Aos meus colegas a quem devo muito do meu aprendizado e com quem compartilhei ótimos momentos que sempre estarão guardados na memória.

Agradeço também a todos os professores, em especial ao meu orientador, que realizam seu trabalho com excelência, capacitando-nos a sermos profissionais destacados no mercado de trabalho. Por fim, a todos os membros da comunidade acadêmica que muito contribuem, a cada dia, para que nós, alunos, tenhamos prazer em cursar a faculdade.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	6
LISTA DE FIGURAS.....	7
LISTA DE TABELAS	8
RESUMO.....	9
ABSTRACT	10
1 INTRODUÇÃO	11
2 CONCEITOS RELACIONADOS	13
2.1 Linguagens e tecnologias utilizadas	13
2.1.1 A linguagem de modelagem UML	13
2.1.2 A linguagem PHP	13
2.1.3 Banco de dados MySQL.....	14
2.1.4 A linguagem de marcação XHTML	14
2.1.5 A linguagem Javascript	14
2.1.6 Folhas de estilo CSS	14
2.1.7 A tecnologia de desenvolvimento AJAX	15
2.2 Ferramentas utilizadas.....	15
2.2.1 JUDE Community	15
2.2.2 fabFORCE DBDesigner 4	16
2.2.3 Macromedia Dreamweaver 8.....	16
3 PLANEJAMENTO E ANÁLISE DE REQUISITOS.....	17
3.1 Glossário de termos	17
3.2 Requisitos do sistema	17
3.2.1 Requisitos de interface.....	18
3.2.2 Requisitos funcionais.....	18
3.2.3 Requisitos não funcionais.....	19
3.3 Casos de uso preliminares.....	20
3.4 Modelo conceitual preliminar.....	21
4 DETALHAMENTO DA SOLUÇÃO PROPOSTA	23
4.1 Casos de uso	23
4.1.1 Diagrama de casos de uso essenciais.....	23
4.1.2 Casos de uso essenciais expandidos	25
4.1.3 Casos de uso reais.....	30
4.2 Modelagem da base de dados	33
4.2.1 Modelagem inicial	33
4.2.2 Informações de controle	35
4.2.3 Organização do sistema	36
4.2.4 Otimização de consultas	37
4.3 Definição das classes e métodos.....	38
4.3.1 Modelo conceitual refinado	38

4.3.2	Diagrama de classes.....	40
5	IMPLEMENTAÇÃO	44
5.1	Gerenciador.....	44
5.1.1	Estrutura básica.....	44
5.1.2	Funcionalidades restritas	46
5.2	Website	47
5.2.1	Usabilidade	47
5.2.2	Navegação	48
5.2.3	Institucional	48
5.2.4	Otimização.....	49
5.3	Palpigol x Concorrentes	50
5.3.1	Serviços oferecidos.....	51
5.3.2	Usabilidade	52
6	CONCLUSÃO.....	54
	REFERÊNCIAS	56

LISTA DE ABREVIATURAS E SIGLAS

BD	Banco de Dados
CRUD	Create, Retrieve, Update and Delete
ER	Entidade Relacionamento
SGBD	Sistema de Gerência de Banco de Dados
UML	Unified Modeling Language
URL	Uniform Resource Locator

LISTA DE FIGURAS

Figura 3.1: Diagrama de casos de uso preliminares	20
Figura 3.2: Modelo conceitual preliminar	21
Figura 4.1: Diagrama de casos de uso essenciais	24
Figura 4.2: Modelagem ER inicial do banco de dados.....	34
Figura 4.3: Modelagem ER com a tabela usuario	35
Figura 4.4: Modelagem ER com tabelas para organização do sistema	37
Figura 4.5: Modelagem ER com a tabela ranking para consultas rápidas.....	38
Figura 4.6: Modelo conceitual refinado	39
Figura 4.7: Diagrama de classes	41
Figura 4.8: Código do método <i>load()</i> da classe <i>Competicoes</i>	42
Figura 5.1: Interface do gerenciador Palpigol	45
Figura 5.2: Cadastro de usuários no gerenciador Palpigol	46
Figura 5.3: Interface do Website Palpigol	47
Figura 5.4: Exemplo de Meta tags para uma categoria de competições.....	49

LISTA DE TABELAS

Tabela 3.1: Glossário de termos do projeto.....	17
Tabela 4.1: Caso de uso CRUD: Listagem.....	25
Tabela 4.2: Caso de uso CRUD: Consulta	26
Tabela 4.3: Caso de uso CRUD: Inclusão	26
Tabela 4.4: Caso de uso CRUD: Exclusão	27
Tabela 4.5: Caso de uso CRUD: Alteração	27
Tabela 4.6: Caso de uso fechar rodada	28
Tabela 4.7: Caso de uso login no sistema.....	29
Tabela 4.8: Caso de uso visualizar dados	29
Tabela 4.9: Caso de uso real dar palpite.....	30
Tabela 4.10: Caso de uso real cadastrar jogos.....	32
Tabela 5.1: Palpigol x Concorrentes – serviços oferecidos.....	51
Tabela 5.2: Palpigol x Concorrentes – usabilidade	52

RESUMO

Os sistemas de entretenimento são cada vez mais comuns na Internet, muitos deles surgem e são amplamente aceitos pelo público, evoluindo e ganhando, assim, a cada dia, mais adeptos. Por esse motivo é possível afirmar que a grande rede oferece ótimas oportunidades para idéias boas e inovadoras. Portanto, no país do futebol, um sistema de entretenimento para a criação de bolões sobre jogos de futebol que oferece muitas diferenças em relação a todos atuais concorrentes tem grandes chances de obter sucesso.

Assim, a partir disso, este trabalho propõe o projeto de um sistema inovador que permite a criação de bolões personalizados, onde o cliente tem a opção de criar as próprias competições, com times, jogos e regras de sua escolha. Para tanto, ao longo do texto, abordam-se todos os elementos presentes no decorrer do projeto, desde o planejamento inicial até os últimos detalhes da implementação. Ainda, realiza-se um comparativo entre o sistema proposto e quatro outros sistemas existentes visando avaliar e analisar a qualidade e os serviços oferecidos por cada um deles, destacando as diferenças e vantagens do sistema proposto.

Palavras-Chave: bolão, palpito, competição, sistema Web, jogos de futebol.

Palpigol - a Web system to creation of disputes into groups based on sports games results

ABSTRACT

Entertainment systems are increasingly becoming more common on the Internet. Many of them arise and are rapidly and widely accepted by the public, thus evolving and gaining more followers every day. Therefore it is possible state that the vast network offers great opportunities for good and innovative ideas. Consequently, in the country of football, an on-line entertainment system allowing football bids and offering many differences with respect to existing competitors is likely to succeed. In this sense, this paper proposes the design of an innovative system that allows people to define custom bids, where the customer has the option to create their own tournaments with teams, rules and games of their choice. Therefore, this document, presents all the elements used throughout the project, ranging from initial planning to the last details of implementation. Last but not least, a comparison between the proposed system and four other existing systems is presented in order to evaluate and analyze the quality and services offered by each approach, highlighting the benefits of the proposed system.

Keywords: bids, palpitate, competition, Web system, football games.

1 INTRODUÇÃO

Diante do crescente aumento e popularização da Internet, não há como negar que ela seja um ninho de oportunidades. Cada vez mais surgem sistemas Web que são amplamente aceitos pelo público, tanto provendo facilidades quanto entretenimento. Nesse contexto, encaixa-se o Palpigol – sistema Web para a criação de disputas em grupos com base em resultados de jogos desportivos.

Popularizado como “bolão”, a idéia principal do sistema consiste em palpitar sobre o placar de jogos de futebol que estejam ocorrendo no momento. Atrelado a isso, acrescenta-se um esquema de pontuação para que seja possível gerar uma tabela de classificação resultando assim em uma competição entre os participantes.

A idéia não é nova: após uma vasta pesquisa na Internet foram encontrados alguns sites que oferecem um serviço semelhante. Dentre todos esses sites encontrados, pode-se efetuar uma divisão em dois grandes grupos. No primeiro deles, são oferecidas competições pré-definidas e prêmios aos participantes que atingirem as primeiras posições. Os melhores exemplos a serem citados são o BolaoMania¹ e o Virtual Bolão². Nesse grupo, não é possível que o cliente (usuário) crie uma disputa entre seus amigos, por exemplo, ele apenas tem a opção de participar de competições juntamente com qualquer pessoa que faça o cadastro no site.

Já no segundo grupo, o cliente tem opção de criar seu próprio bolão e convidar seus amigos para participar. Esse cliente é visto como administrador e cabe a ele permitir o ingresso de novos participantes no bolão. Foram encontrados dois bons exemplos a serem citados: o bolão ESPN³ e o BolaoVip⁴. O segundo deles oferece uma quantidade enorme de competições possíveis de serem disputadas, porém, em ambos, não é possível criar uma nova competição. Além disso, nos dois casos as regras para pontuação já estão pré-definidas.

Assim, o grande diferencial oferecido pelo Palpigol é a possibilidade do cliente administrador poder criar uma competição, cadastrando times e jogos a seu critério, escolhendo suas próprias regras para pontuação e classificação. Já que, atualmente, nenhum dos sites anteriormente citados, que provêm um serviço semelhante aos seus clientes, oferece essa flexibilidade, acredita-se, portanto, ser esse um bom mercado a ser explorado.

¹ BolaoMania (<http://www.bolaomania.com/>).

² Virtual Bolão (<http://www.virtualbolao.com/>).

³ ESPN (<http://espnbolao.terra.com.br/>).

⁴ BolaoVip (<http://www.bolaovip.com.br/>).

O principal objetivo, então, é oferecer essa flexibilidade aos clientes através de um gerenciador de fácil manipulação bem como um site onde a maior preocupação será a usabilidade, ou seja, deve-se, além do diferencial informado acima, superar os concorrentes na praticidade e facilidade de uso.

Para avaliar a usabilidade do sistema é feita uma estimativa visando analisar as operações mais realizadas pelos clientes no Website. Para o acesso às áreas mais frequentemente utilizadas, é estudada a melhor forma para ter-se o menor número de cliques possíveis. É feita, ainda, uma comparação entre as funcionalidades dos sistemas mencionados anteriormente e o Palpigol, visando não só comparar os serviços oferecidos por cada um deles como também avaliar a usabilidade do sistema implementado.

O texto está dividido da seguinte forma: primeiramente são abordadas todas as ferramentas e tecnologias utilizadas para o desenvolvimento do sistema, após é apresentado todo planejamento do projeto, apresentando-se a análise detalhada dos requisitos do sistema bem como as modelagens iniciais. Posteriormente, é então detalhada a solução proposta, ou seja, é apresentada toda arquitetura do sistema utilizando-se para tal o detalhamento dos casos de uso no estilo essencial e real, a modelagem da base de dados e a especificação detalhada das classes e métodos do sistema.

Por fim então, é apresentada a implementação do sistema, isto é, tudo que foi realmente construído no projeto juntamente com a comparação com os sistemas existentes. Para a implementação é feita uma divisão em dois grandes grupos, o gerenciador do sistema e o Website. Para ambos são abordados os principais aspectos que os compõem fazendo uso de esboços da Interface de cada um deles.

2 CONCEITOS RELACIONADOS

Para a especificação do sistema foi adotada Linguagem de Modelagem Unificada – UML, considerada uma das linguagens mais expressivas para modelagem de sistemas orientados a objetos (OMG, 2003).

Para a implementação do sistema, foi utilizada a linguagem PHP⁵ em conjunto com banco de dados MySQL⁶. As páginas Web responsáveis por fornecer a interface administrativa, o gerenciador, a interface de acesso geral, e o site, foram desenvolvidos utilizando-se XHTML⁷, Javascript⁸ e folhas de estilo CSS⁹. Ainda, para algumas páginas específicas, foi utilizada a tecnologia de desenvolvimento AJAX¹⁰.

2.1 Linguagens e tecnologias utilizadas

A seguir, será feita uma breve introdução a todas as linguagens e tecnologias envolvidas no desenvolvimento do projeto, visando apontar as características que levaram à escolha das mesmas para a especificação e implementação do sistema.

2.1.1 A linguagem de modelagem UML

A Linguagem de Modelagem Unificada (UML) é uma linguagem visual para especificação, documentação, visualização e desenvolvimento de sistemas orientados a objetos (OMG, 2003). Através de diagramas padronizados, ela permite representar sistemas de softwares sob diversas perspectivas de visualização, caracterizando-se principalmente por sintetizar os principais métodos existentes na modelagem de sistemas orientados a objetos (LARMAN, 2007), o que justifica sua escolha para o projeto.

2.1.2 A linguagem PHP

PHP é acrônimo de *Hypertext Preprocessor* (pré-processador de hipertexto), uma poderosa linguagem de programação *open source*, mundialmente utilizada, principalmente no ambiente Web. Além de oferecer velocidade, robustez e portabilidade, ela tem como uma de suas características mais marcantes a capacidade de se misturar ao HTML, tornando mais fácil a geração de páginas Web dinâmicas

⁵ PHP: PHP Hipertext Preprocessor (<http://www.php.net>).

⁶ MySQL: SGBD open source (<http://dev.mysql.com>).

⁷ XHTML: eXtensible HyperText Markup Language (<http://www.w3.org/TR/xhtml11>).

⁸ Javascript: linguagem de script (<http://pt.wikipedia.org/wiki/JavaScript>).

⁹ CSS: Cascading Style Sheets (<http://www.w3.org/Style/CSS>).

¹⁰ AJAX: Asynchronous Javascript And XML (<http://pt.wikipedia.org/wiki/AJAX>).

(SOARES, 2004). Assim, todas essas características mencionadas influenciaram na escolha dessa linguagem para a implementação do sistema.

2.1.3 Banco de dados MySQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD) que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês *Structured Query Language*) como interface. É atualmente um dos bancos de dados mais populares, com mais de 10 milhões de instalações pelo mundo¹¹.

O MySQL é o líder entre os sistemas de gerenciamento de banco de dados relacional de código-fonte aberto. Ele é poderoso, estável e, o melhor de tudo, é livre e gratuito (SUEHRING, 2002), o que foi decisivo para a sua escolha no projeto.

2.1.4 A linguagem de marcação XHTML

HTML (acrônimo para a expressão inglesa *HyperText Markup Language*, que significa Linguagem de Marcação de Hipertexto) é uma linguagem de marcação amplamente utilizada para produzir páginas na Web, pois arquivos codificados em HTML são interpretados por navegadores, fornecendo, assim, a exibição desejada do conteúdo do arquivo (CASTRO, 2006).

Visando melhorar a acessibilidade, o XHTML, ou *eXtensible Hypertext Markup Language*, é uma reformulação da linguagem de marcação HTML baseada em XML¹². Essa padronização combina as *tags* de marcação HTML com regras da marcação XML tendo em vista a exibição de páginas Web em diversos dispositivos (CASTRO, 2006).

Tendo em vista o aumento crescente de dispositivos móveis como *palm* e celular, bem como a imensa difusão desses equipamentos, com certeza, deve-se utilizar o XHTML para o desenvolvimento de qualquer sistema nos dias atuais.

2.1.5 A linguagem Javascript

Javascript é uma linguagem de programação criada principalmente para atender a interação com a página no lado cliente. O uso mais freqüente é na validação de formulários. A linguagem é interpretada ao invés de compilada, ela oferece tipagem dinâmica e um bom suporte a expressões regulares (GOODMAN, 2001).

A linguagem Javascript fornece ótimos recursos para qualquer sistema na Internet em que haja interação com os clientes, seus métodos de validação oferecem a rapidez desejada em sistemas Web (NEGRINO, 2008), o que será fundamental no sistema.

2.1.6 Folhas de estilo CSS

Cascading Style Sheets (ou simplesmente CSS) é uma linguagem de estilos utilizada para definir a apresentação de documentos escritos em uma linguagem de marcação, como HTML ou XML. Seu principal benefício é prover a separação entre o formato e o conteúdo de um documento.

Ao invés de colocar a formatação dentro do documento, o desenvolvedor cria um *link* (ligação) para uma página que contém os estilos, procedendo de forma idêntica para todas as páginas de um sistema. A grande vantagem desse método é, por exemplo, ao

¹¹ MySQL. (<http://www.mysql.com/why-mysql>). Página visitada em 20 de junho de 2009.

¹² XML: Extensible Markup Language (<http://www.w3.org/XML>).

desejar a alteração da aparência do sistema, bastará modificar apenas um arquivo (CASTRO, 2006).

Essa metodologia é muito adequada para o sistema, pois certamente haverá, no futuro, atualizações ou modificações no *layout* tanto por parte do gerenciador quanto por parte do Website.

2.1.7 A tecnologia de desenvolvimento AJAX

AJAX (acrônimo em língua inglesa de *Asynchronous Javascript And XML*) é o uso metodológico de tecnologias como Javascript e XML, providas por navegadores, para tornar páginas Web mais interativas com o usuário, utilizando-se de solicitações assíncronas de informações. AJAX não é somente um novo modelo, é também uma iniciativa na construção de aplicações Web mais dinâmicas e criativas (SOARES, 2006).

Na realidade são várias tecnologias conhecidas trabalhando juntas, cada qual fazendo sua parte, e, assim, oferecendo novas funcionalidades. AJAX incorpora em seu modelo uma apresentação baseada em padrões, usando XHTML e CSS. Fornece intercâmbio e manipulação de dados usando XML assim como recuperação assíncrona de dados usando o objeto XMLHttpRequest¹³. Por fim, a união de todas essas tecnologias anteriormente mencionadas é feita através de Javascript (NEGRINO, 2008).

A maior vantagem das aplicações AJAX é que elas rodam no próprio navegador web. Assim, no sistema desenvolvido, há várias páginas em que há determinadas ações onde pouco conteúdo deve ser atualizado ou calculado e o restante mantido. Nesses casos, deseja-se a maior rapidez possível no processamento. Na forma clássica, mesmo havendo pouco conteúdo a ser atualizado, toda a página seria recarregada. A tecnologia AJAX se mostrou eficiente em atender essa necessidade, pois, com o emprego da tecnologia, apenas o pouco conteúdo é atualizado nesse exemplo, sem a necessidade da recarga total da página.

2.2 Ferramentas utilizadas

Para o desenvolvimento do sistema, foram utilizadas ferramentas para a criação dos diagramas envolvendo casos de uso, interação, seqüência e classes, assim como para a modelagem da base de dados. Também se fez necessária uma IDE para a programação na linguagem PHP.

2.2.1 JUDE Community

Através da ferramenta JUDE Community¹⁴, é possível criar praticamente todos os diagramas previstos na UML. Utilizando essa ferramenta, foram criados: o diagrama de casos de uso e o diagrama de classes, que serão apresentados no decorrer desse documento. Para tanto, foi utilizada a versão gratuita que, apesar de possuir algumas limitações em comparação à versão paga da ferramenta, atendeu a todas as necessidades do projeto.

¹³ XMLHttpRequest: objeto Javascript (<http://pt.wikipedia.org/wiki/XMLHttpRequest>).

¹⁴ Ferramenta JUDE Community: Disponível em (<http://jude.change-vision.com/jude-web/product/community.html>). Página visitada em 20 de junho de 2009.

2.2.2 fabFORCE DBDesigner 4

A ferramenta DBDesigner 4¹⁵, da fabFORCE, foi utilizada para fazer a modelagem da base de dados assim como para sincronizar, com o banco de dados MySQL criado, o modelo ER, modelo Entidade Relacionamento, cuja finalidade é descrever, de maneira conceitual, os dados a serem utilizados em um sistema.

Trata-se de uma ferramenta gratuita e poderosa, que permite, de forma visual, a criação de uma modelagem ER com facilidade. A ferramenta também permite que as mudanças realizadas na modelagem ER sejam sincronizadas com a base de dados que está sendo utilizada no sistema, facilitando, assim, a manutenção da documentação da modelagem bem como melhorando o processo de desenvolvimento.

2.2.3 Macromedia Dreamweaver 8

Toda a programação do sistema foi feita utilizando-se a IDE Dreamweaver 8.0 da Macromedia, que é a versão anterior ao Dreamweaver CS4, da Adobe¹⁶. A IDE utilizada, mesmo não sendo a versão atual, oferece, além das funcionalidades básicas como *syntax highlight*, verificadores de sintaxe específicos para PHP, Javascript, HTML entre outros, uma interface muito boa para efetuar *download/upload* de arquivos com segurança, o que torna mais rápido o desenvolvimento facilitando a vida do programador.

¹⁵ Ferramenta DBDesigner 4: Disponível em (<http://fabforce.net/dbdesigner4/>). Página visitada em 20 de junho de 2009.

¹⁶ IDE Dreamweaver: Disponível em (<http://www.adobe.com/>). Página visitada em 20 de junho de 2009.

3 PLANEJAMENTO E ANÁLISE DE REQUISITOS

Na fase de planejamento é elaborado um glossário de termos a serem utilizados no projeto bem como são levantados os principais requisitos do sistema. Posteriormente, serão definidos os casos de uso preliminares e um modelo conceitual preliminar para ter-se a idéia inicial da arquitetura do sistema.

3.1 Glossário de termos

O Glossário de termos é uma lista de termos e suas definições que serão relevantes no decorrer do projeto. Tem como principal objetivo reduzir problemas de comunicação e de ambigüidades nos requisitos, já que há termos específicos do domínio que poderiam ser usados de diferentes modos.

Tabela 3.1: Glossário de termos do projeto

Termo	Definição e Informações
Palpiteiro	Cliente que utiliza o Website para dar palpites sobre o resultado de jogos das competições que ele participa.
Palpite	Um placar para algum jogo que o Palpiteiro informa antes do jogo ser realizado.
Grupo de Palpiteiros	Conjuntos formados agrupando-se Palpiteiros para disputar uma competição.
Times	Clubes ou Seleções que integrarão os jogos das competições.
Competições	Disputas contidas de uma ou mais rodadas que Grupos de Palpiteiros disputarão.
Rodadas	Um conjunto de jogos próximos (geralmente possuindo a mesma data ou mesmo nível) dentro de uma competição.
Jogos	Partidas entre dois times.
Bolão	União de uma competição mais um grupo de palpiteiros que a dispute.

3.2 Requisitos do sistema

Fazendo-se a análise dos requisitos do sistema, primeiramente surgem os requisitos de interface, que especificam as funcionalidades inerentes à interface do sistema com o

cliente. Após, os demais requisitos do sistema podem ser classificados em dois grandes grupos, os requisitos funcionais e não funcionais.

Os requisitos funcionais descrevem o comportamento do sistema, ou seja, as ações para cada entrada. Em outras palavras, eles descrevem as funcionalidades que o sistema deve dispor. É comum a associação a “o que” tem que ser feito pelo sistema.

Já aos requisitos não funcionais, pode-se fazer uma associação a “como” deve ser feito o sistema. Em geral eles se relacionam com padrões de qualidade como usabilidade, confiabilidade, desempenho, robustez, dentre outros.

3.2.1 Requisitos de interface

O sistema deve ser composto de duas interfaces; A primeira delas consiste no gerenciador restrito aos clientes administradores. A segunda delas consiste no Website, que será acessada pelos demais clientes e também é aberta ao público em geral. Para isso, deve possuir uma área restrita aos clientes requerendo autenticação e uma área aberta com páginas informativas fazendo a divulgação do sistema.

3.2.2 Requisitos funcionais

Para os requisitos funcionais, primeiramente deve-se explicar o que significa CRUD e a forma como será representado no decorrer do projeto. CRUD é o acrônimo da expressão em língua Inglesa *Create, Retrieve, Update e Delete* (KILOV, 1998), usada para definir quatro operações básicas usadas em bancos de dados relacionais ou em interfaces para usuários para criação, consulta, atualização e exclusão de dados. A seguir então, listam-se os requisitos funcionais do sistema:

- O sistema deve prover no gerenciador uma área restrita à pessoa responsável pelo gerenciamento do sistema como um todo, ou seja, responsável por gerenciar os clientes administradores;
- O sistema deve prover no gerenciador o CRUD de Palpiteiros, onde o cliente administrador poderá cadastrar amigos para disputarem alguma competição;
- O sistema deve prover no gerenciador o CRUD de Grupos de Palpiteiros, onde o cliente administrador associará os Palpiteiros já cadastrados ao grupo;
- O sistema deve prover no gerenciador o CRUD de Times, onde o cliente administrador poderá cadastrar as equipes que integrarão as competições.
- O sistema deve prover no gerenciador o CRUD de Competições, onde o cliente administrador definirá os grupos que participam da mesma gerando assim um convite a cada palpiteiro do grupo à participar da competição;
- O sistema deve prover no gerenciador o CRUD de Rodadas, onde o cliente administrador poderá cadastrar as rodadas de cada competição;
- O sistema deve prover no gerenciador o CRUD de Jogos, onde o cliente administrador cadastrará os jogos de cada rodada da competição;
- O sistema deve prover no gerenciador a operação para fechar rodadas onde, após um resultado final de algum jogo da rodada, o cliente administrador informará o placar do jogo. Isso gerará uma pontuação para cada Palpiteiro que deu seu palpite para aquele jogo;

- O sistema deve prover no Website a possibilidade do Palpiteiro efetuar sua autenticação informando usuário e senha;
- O sistema deve prover no Website a possibilidade do Palpiteiro visualizar as competições que participa;
- O sistema deve prover no Website a possibilidade do Palpiteiro, para cada competição que participa, visualizar todas as rodadas e jogos da competição com placares finais ou em aberto;
- O sistema deve prover no Website a possibilidade do Palpiteiro, para cada competição, visualizar os palpites dados por todos Palpiteiros que disputam aquela competição;
- O sistema deve prover no Website a possibilidade do Palpiteiro dar o seu Palpite para os jogos das competições que ele disputa;
- O sistema deve prover no Website a possibilidade do Palpiteiro, para cada competição que disputa, listar a classificação atualizada da mesma.

3.2.3 Requisitos não funcionais

Os requisitos não funcionais são de extrema importância, pois definem se o sistema será ou não eficiente para a tarefa a ser executada. A seguir então, listam-se os requisitos não funcionais do sistema:

- O sistema deve tratar os erros e evitar mostrar na tela mensagens técnicas;
- A coleta dos dados para autenticação, tanto do cliente administrador como do palpiteiro, deve ser feita de maneira segura;
- O sistema deve adotar medidas para desencorajar e coibir a ação de usuários mal intencionados, nas telas de cadastro e autenticação de clientes;
- Os campos para autenticação do cliente no Website devem estar sempre disponíveis, por exemplo, no topo do site em todas as páginas, visto que para a ampla maioria das interações do cliente com o sistema é necessária a autenticação;
- Após efetuada a autenticação, o cliente sempre é direcionado para uma página que lista os próximos jogos a serem palpitados por ele das competições em que disputa, visto que essa operação, com certeza, está dentre as mais realizadas no Website pelo cliente;
- Após o cliente administrador efetuar a operação para fechar alguma rodada no gerenciador, será gerada uma pontuação para todos os palpiteiros da competição; essa pontuação deve ser salva e mantida atualizada no banco de dados visando consultas rápidas no Website, visto que ver a classificação das competições em que o palpiteiro disputa também está entre as operações mais realizadas pelo cliente;
- Dado que o sistema é uma aplicação Web e depende da velocidade da Internet do cliente, todas as interações com o sistema (com exceção da operação para fechar rodadas que tende a ser mais lenta) devem fornecer um tempo de resposta inferior a 6 segundos considerando-se uma velocidade de *download* de 200Kbps;

- O conteúdo do Website deve ser leve, sem poluição visual, em torno de 80% textual para alcançar um ótimo desempenho mesmo para clientes que não tenham uma Internet rápida;

3.3 Casos de uso preliminares

O grande objetivo dos casos de uso preliminares é sintetizar as principais funcionalidades do sistema para que no início do projeto sejam facilmente compreendidas. Para isso, é apresentado o diagrama de casos de uso preliminares, onde, de maneira simplificada, são representados os principais serviços oferecidos pelo sistema assim como são evidenciados os papéis desempenhados por cada ator¹⁷ na utilização do sistema.

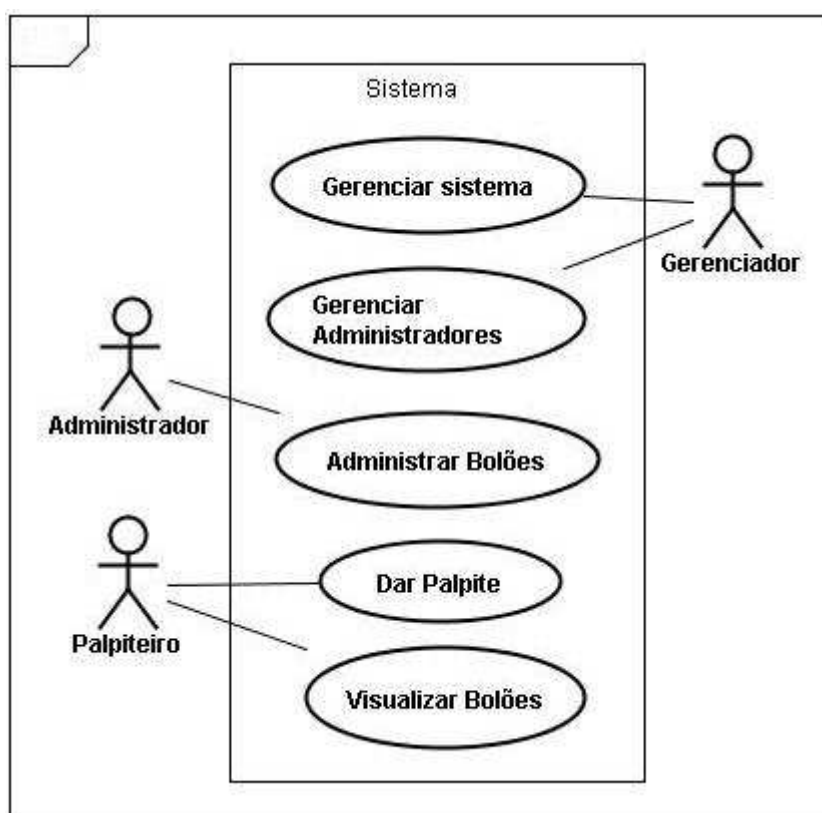


Figura 3.1: Diagrama de casos de uso preliminares

O diagrama de casos de uso preliminares da Figura 3.1 apresenta três atores, o gerenciador, o administrador e o palpiteiro. O primeiro deles é responsável pelo gerenciamento de clientes administradores e de algumas funcionalidades do próprio sistema. Já o ator administrador, tem como responsabilidade administrar tudo que está atrelado aos bolões. Por fim, o ator palpiteiro pode visualizar todas as informações referentes aos bolões que disputa e também dar palpites nesses bolões.

¹⁷ Ator: qualquer coisa com um comportamento, inclusive o próprio sistema quando invoca os serviços de outros sistemas (LARMAN,2007).

Diante disso, pode-se ter a primeira idéia das funcionalidades do sistema. É evidente que esses casos de uso precisam ser expandidos e minuciados, por isso eles são detalhados no capítulo seguinte.

3.4 Modelo conceitual preliminar

Assim como os casos de uso preliminares, o modelo conceitual preliminar tem como objetivo dar uma idéia inicial, porém agora, da arquitetura do sistema. Para tanto, é apresentado na Figura 3.2 as principais classes do sistema, pois serão elas que servirão de base para a construção das demais classes do sistema.

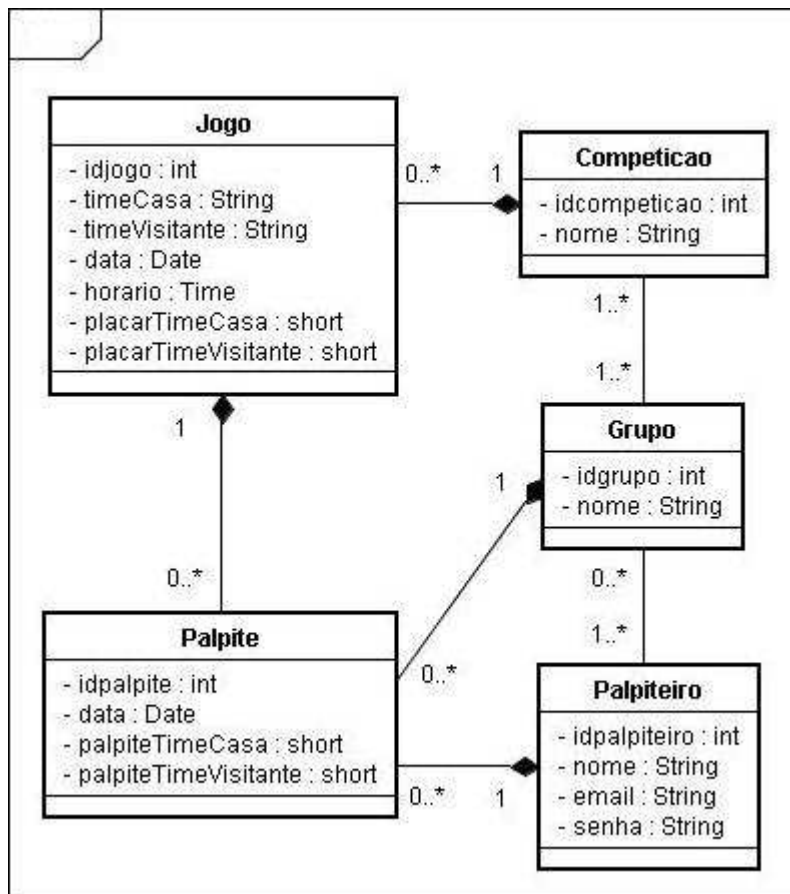


Figura 3.2: Modelo conceitual preliminar

O modelo conceitual preliminar da Figura 3.2 define cinco classes, são elas: Palpiteiro, Grupo, Competicao, Jogo e Palpite.

A classe Palpiteiro contém todos os dados do cliente necessários para sua identificação e autenticação no sistema. Já as classes Grupo e Competicao contêm o identificador único e seu nome. Há uma relação de associação entre essas três classes, de tal forma que um palpитеiro pertence a zero ou mais grupos e um grupo contém um ou mais palpитеiros. Um grupo também disputa uma ou mais competições e uma competição é disputada por um ou mais grupos.

A classe Jogo contém todos os atributos necessários para a identificação do jogo, ou seja, times, data, horário e placar. Essa classe possui uma relação de composição com a classe Competicao, ou seja, jogos podem ser vistos como partes de uma competição.

Todo jogo pertence a exatamente uma competição e uma competição contem zero ou mais jogos.

Por fim, a classe Palpite contém o seu identificador, a data e o placar do palpite. Essa classe possui relações de composição com as classes Jogo, Palpiteiro e Grupo. Palpites podem ser vistos como partes dos jogos, ou seja, cada palpite é sobre exatamente um jogo e jogos têm vários palpites, cada qual pertencente a um palpiteiro. Já com o relacionamento com a classe Grupo, abre-se a possibilidade de que um palpiteiro dê palpites diferentes para um jogo caso ele participe de dois ou mais grupos que disputem a mesma competição.

A partir do modelo conceitual preliminar tem-se a visão do núcleo do sistema, com isso parte-se para o refinamento desse modelo para então dar origem ao diagrama de classes. Esse detalhamento é feito no capítulo seguinte.

4 DETALHAMENTO DA SOLUÇÃO PROPOSTA

Para detalhar a solução proposta serão primeiramente apresentadas todas as funcionalidades desejáveis da implementação utilizando-se dos casos de uso essenciais e reais do sistema.

4.1 Casos de uso

Os casos de uso em um estilo essencial são obtidos a partir do detalhamento dos casos de uso preliminares apresentados na seção anterior. O objetivo é ter cada funcionalidade do sistema bem definida e especificada, para tanto será apresentado o diagrama de casos de uso essenciais e após serão descritos os principais casos de uso através dos casos de uso expandidos.

Ainda, para alguns casos de uso do sistema considerados críticos, é feita uma abordagem no formato completo, onde há um detalhamento maior, ou seja, uma escrita mais aprofundada. Esses casos de uso são tratados como casos de uso reais do sistema.

4.1.1 Diagrama de casos de uso essenciais

A partir do refinamento de cada caso de uso existente no diagrama de casos de uso preliminares da-se origem ao diagrama de casos de uso essenciais. Nesse diagrama são mantidos os três atores iniciais: Gerenciador, Administrador e Palpiteiro, porém todas as funcionalidades são melhor detalhadas.

Muitas funcionalidades estão sintetizadas através da expressão CRUD, porém em todas elas estão associadas operações de listagem, consulta, inclusão, alteração e exclusão de registros. Essas operações serão melhor descritas nos casos de uso expandidos da seção seguinte. Na Figura 4.1, é apresentado o diagrama de casos de uso essenciais.

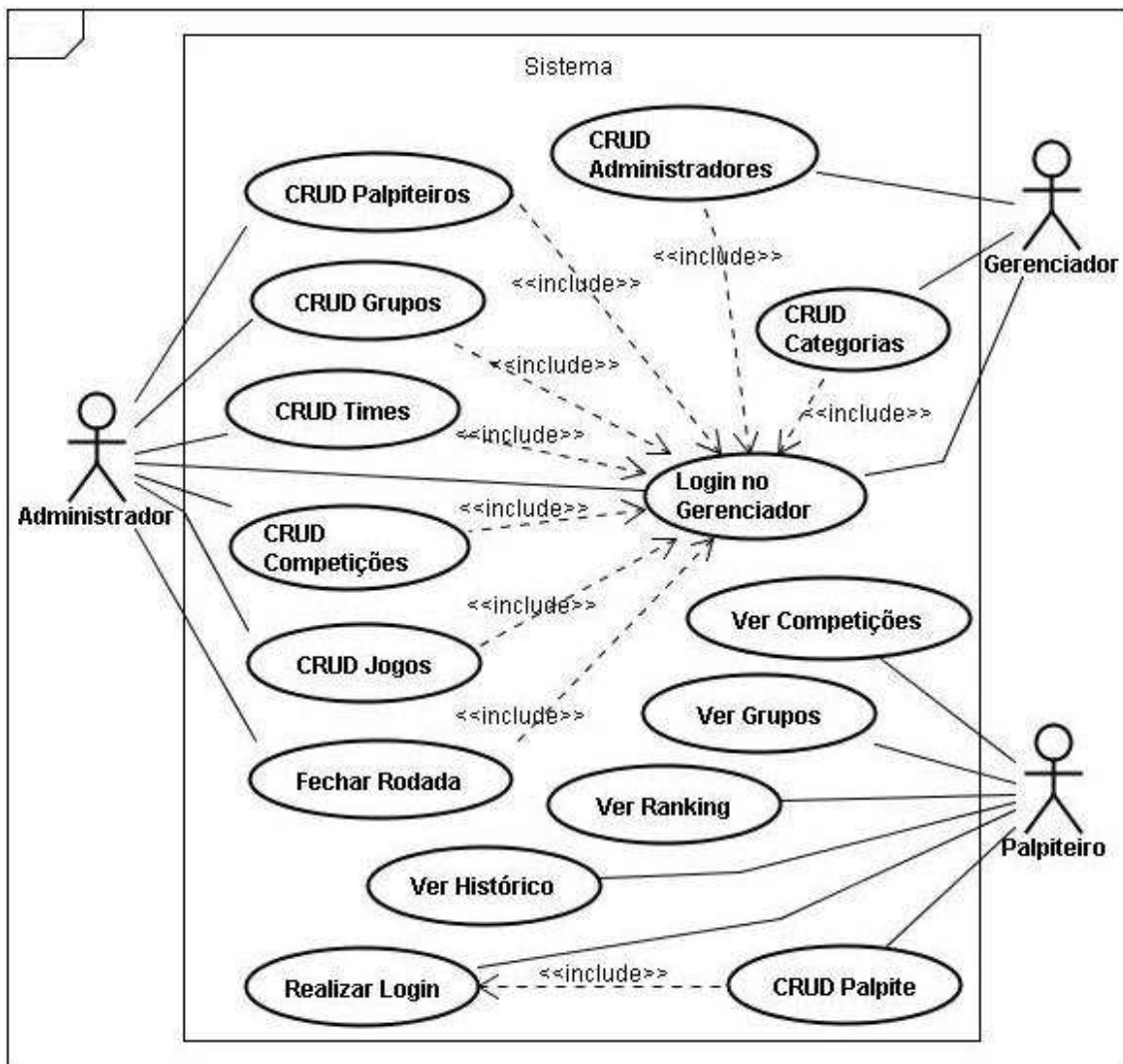


Figura 4.1: Diagrama de casos de uso essenciais

No diagrama da Figura 4.1 é possível verificar as funcionalidades minuciadas correspondentes a cada um dos três atores. O primeiro deles, o ator Gerenciador, é responsável pelo CRUD de categorias e de administradores do sistema. Para ambas as operações o ator deverá primeiramente efetuar o *login* no gerenciador.

O conceito de categorias foi criado para promover uma melhor organização do sistema como um todo, esse conceito será melhor abordado posteriormente, na seção onde é definida a modelagem da base de dados. Já os administradores serão, na verdade, clientes responsáveis por administrar os bolões, ou seja, o ator Gerenciador é responsável pelo CRUD de administradores do sistema.

O segundo ator, Administrador, será o responsável pelo CRUD de palpitesiros, grupos de palpitesiros, times, competições e jogos. Para todas as operações o ator deverá primeiramente efetuar o *login* no gerenciador. Através de todas essas funcionalidades, esse ator poderá então criar um bolão. Ele também será o responsável por gerenciar o bolão, assim ele deve inserir os placares dos jogos ao final de uma rodada, essa funcionalidade foi definida no diagrama como: fechar rodada.

O terceiro ator, Palpiteiro, terá acesso apenas ao CRUD de seus palpites. Para isso ele deverá realizar *login* no sistema informado suas credencias de acesso. Ele poderá

também visualizar os grupos ao qual pertence, as competições que disputa e sua respectiva classificação nessas competições. Ainda há possibilidade do Palpiteiro poder ver um histórico contendo informações das competições em que disputa, tais como seus palpites, palpites dos outros componentes do grupo e placares dos jogos.

4.1.2 Casos de uso essenciais expandidos

Os casos de uso expandidos são narrativas em texto com o objetivo de descobrir e registrar os requisitos do sistema. O estilo de escrita é dito essencial porque ele evita detalhes da interface com o usuário e focaliza as reais intenções do usuário (LARMAN, 2007). Assim, é possível especificar de um modo mais abrangente as principais funcionalidades do sistema independente da tecnologia utilizada e dos detalhes dos mecanismos relacionados com a interface.

Para os casos de uso expandidos não há um padrão definido nem um melhor formato. Muito pelo contrário, há vários formatos utilizáveis, alguns com seções acrescentadas ou removidas, ou então com nomes de cabeçalhos mudados, porém o mais importante é escrever os detalhes de sucesso principal e suas extensões de alguma forma (LARMAN, 2007). Assim, optou-se por utilizar o formato de duas colunas que enfatiza a ocorrência de uma interação entre atores e o sistema além de proporcionar uma clara separação visual.

Então, primeiramente, por existir CRUD para todos os elementos do sistema, tais como palpitesiros, grupos de palpitesiros, times, competições, jogos e palpites, além do CRUD de administradores e categorias, opta-se por explicar apenas o conceito de cada tipo de operação CRUD disponível, sendo que ele não muda independente do tipo de elemento com o qual ele esteja sendo usado no momento.

Tabela 4.1: Caso de uso CRUD: Listagem

Ação – CRUD: Listagem	
Ator:	Gerenciador, Administrador ou Palpiteiro
Pré-Condição:	-
Pós-Condição:	-
Entrada:	Filtro de registros relacionados.
Saída:	Registros relacionados.
Fluxo de Eventos	
Ator	Sistema
Informa dados de filtragem de registros, se assim o desejar.	
	Retorna lista de registros do banco de dados, com base nos filtros - se algum foi informado - ou exibe todos os registros.

Na Tabela 4.1 é detalhada a operação de listagem independente dos dados que estão sendo tratados. Os três atores do sistema poderão efetuar essa operação. Por exemplo, o ator Palpiteiro efetua a operação de listagem do CRUD de palpites, já o ator

Administrador poderá efetuar a operação de listagem para, por exemplo, o CRUD de jogos, assim como o ator Gerenciador poderá efetuar a operação de listagem para, por exemplo, o CRUD de categorias do sistema.

Tabela 4.2: Caso de uso CRUD: Consulta

Ação – CRUD: Consulta	
Ator:	Gerenciador, Administrador ou Palpiteiro
Pré-Condição:	Existência de registros relacionados.
Pós-Condição:	-
Entrada:	Identificador de registro relacionado.
Saída:	Dados do registro relacionado.
Fluxo de Eventos	
Ator	Sistema
Informa o identificador do registro.	
	Busca na base os dados o registro referente ao identificador.

Na Tabela 4.2 é detalhada a operação de consulta independente dos dados que estão sendo tratados para qualquer CRUD pertencente ao sistema. Os três atores do sistema poderão efetuar essa operação.

Tabela 4.3: Caso de uso CRUD: Inclusão

Ação – CRUD: Inclusão	
Ator:	Gerenciador, Administrador ou Palpiteiro
Pré-Condição:	Inexistência do registro no sistema.
Pós-Condição:	Novo registro inserido.
Entrada:	Filtro de registros relacionados.
Saída:	Dados do registro relacionado.
Fluxo de Eventos	
Ator	Sistema
1. Informa os dados de cadastro de um novo registro.	
	Valida dados de cadastro e, se estiverem corretos, inclui novo registro no banco de dados. Caso contrário, retorna para (1), informando campos incorretos.

Na Tabela 4.3 é detalhada a operação de inclusão independente dos dados que estão sendo tratados para a maior parte das operações CRUD pertencentes ao sistema. Os três atores do sistema poderão efetuar essa operação. A inclusão de palpiteiro por parte do cliente administrador sofre apenas uma pequena alteração, pois nessa inclusão é disparado via e-mail um convite ao novo palpiteiro que está sendo inserido no sistema.

Tabela 4.4: Caso de uso CRUD: Exclusão

Ação – CRUD: Exclusão	
Ator:	Gerenciador, Administrador ou Palpiteiro
Pré-Condição:	Existência de registros relacionados.
Pós-Condição:	Registro removido do banco de dados.
Entrada:	Identificador de registro.
Saída:	-
Fluxo de Eventos	
Ator	Sistema
Informa identificador do registro que será removido.	
	Remove registro da base de dados, referente ao identificador informado.

Na Tabela 4.4 é detalhada a operação de exclusão independente dos dados que estão sendo tratados para qualquer CRUD pertencente ao sistema. Os três atores do sistema poderão efetuar essa operação.

Tabela 4.5: Caso de uso CRUD: Alteração

Ação – CRUD: Alteração	
Ator:	Gerenciador, Administrador ou Palpiteiro
Pré-Condição:	Existência de registros relacionados.
Pós-Condição:	Registro modificado na base de dados.
Entrada:	Identificador de registro relacionado.
Saída:	-
Fluxo de Eventos	
Ator	Sistema
Informa o identificador do registro e os dados que deseja alterar.	
	Altera o registro no banco de dados de acordo com o identificador informado.

Na Tabela 4.5 é detalhada a operação de alteração independente dos dados que estão sendo tratados para qualquer CRUD pertencente ao sistema. Os três atores do sistema poderão efetuar essa operação.

Detalhadas todas as operações pertencentes ao CRUD do sistema, tem-se agora a maior parte das funcionalidades do diagrama de casos de uso essenciais bem especificadas através dos casos de uso expandidos. O ator Administrador, além das operações pertencentes ao CRUD, possui outros dois casos de uso: fechar rodada e *login* no gerenciador.

Tabela 4.6: Caso de uso fechar rodada

Ação – Fechar rodada	
Ator:	Administrador
Pré-Condição:	Jogos finalizados com placares a serem inseridos.
Pós-Condição:	Tabela de classificação atualizada.
Entrada:	Placares dos jogos.
Saída:	-
Fluxo de Eventos	
Ator	Sistema
1. Informa o placar do(s) jogo(s).	
	2. Valida dados de entrada. Caso estejam... <i>A) Corretos</i> Insere os dados de entrada no banco de dados e atualiza a tabela de classificação para as competições a que o jogo pertence. <i>B) Incorretos</i> Volta para (1), informando o erro.

Na Tabela 4.6 é detalhado o caso de uso fechar rodada, o qual somente o ator Administrador terá acesso.

Para todos os casos de uso, tanto para o ator Administrador quanto para o ator Gerenciador, será necessário a autenticação no sistema. Sendo assim, ambos os atores primeiramente deverão informar suas credencias de acesso no sistema. Esse caso de uso é descrito como *login* no gerenciador.

Já ao ator Palpiteiro está associado o caso de uso realizar *login*, que será necessário quando o Palpiteiro desejar efetuar alguma operação referente ao CRUD de palpites.

Embora esses dois casos de uso sejam distintos, pois o primeiro deles é executado na interface do gerenciador e o segundo no Website, no estilo essencial onde se deixa de lado a interface de usuário e dá-se enfoque na intenção do ator, eles se tornam

praticamente idênticos e assim podem ser descritos em apenas um caso de uso expandido.

Tabela 4.7: Caso de uso *login* no sistema

Ação – Login no sistema	
Ator:	Gerenciador, Administrador ou Palpiteiro.
Pré-Condição:	Realização de cadastro do ator.
Pós-Condição:	Sessão com o servidor.
Entrada:	Nome de usuário e senha.
Saída:	-
Fluxo de Eventos	
Ator	Sistema
1. Informa nome de usuário e senha.	
	2. Valida dados de entrada. Caso estejam... <i>A) Corretos</i> Valida os dados de entrada contra os dados do banco de dados. Novamente, estes podem estar... <i>a. Corretos</i> Exibe tela inicial para atores cadastrados. <i>b. Incorretos</i> Volta para (1), informando o erro. <i>B) Incorretos</i> Volta para (1), informando o erro.

Na Tabela 4.7 é então generalizada a operação de *login* no sistema para qualquer ator, seja ele Gerenciador, Administrador ou Palpiteiro. Embora as interfaces sejam distintas para esses atores, o funcionamento do mecanismo de autenticação é idêntico.

A partir de então, restam apenas os casos de uso do Palpiteiro correspondentes à visualização de informações, que estão descritos no diagrama de classes como: ver competições, ver grupos, ver ranking e ver histórico. Essas operações de visualização são muito simples podendo então ser sintetizadas em um único caso de uso a ser expandido independente dos dados que estão sendo tratados.

Portanto, a visualização de competições, de grupos, de ranking e de histórico de uma competição pode ser generalizada por meio do caso de uso da Tabela 4.8, descrito como visualizar dados.

Tabela 4.8: Caso de uso visualizar dados

Ação – Visualizar dados	
Ator:	Palpiteiro
Pré-Condição:	Existência de registros relacionados.
Pós-Condição:	-
Entrada:	Identificador de registro relacionado.
Saída:	Dados do registro relacionado.
Fluxo de Eventos	
Ator	Sistema
Informa o identificador do registro.	
	Busca na base os dados e exibe todas informações referentes ao identificador informado.

Na Tabela 4.8 é detalhada a generalização da operação visualizar dados para os casos de uso: ver competições, ver grupos, ver ranking e ver histórico, todos associados ao ator Palpiteiro.

4.1.3 Casos de uso reais

Os casos de uso reais surgem depois que muitos casos de uso tiverem sido identificados e escritos em um formato mais resumido. Então alguns, como por exemplo 10%, dos casos de uso arquiteturalmente significativos e de alto valor são escritos em detalhe (LARMAN, 2007). Nesse formato completo de escrita, ao contrário do estilo essencial, é levada em conta a interface para com o usuário.

Para os casos de uso reais, foram escolhidos os dois principais casos de uso do sistema, o cadastro de palpite realizado pelo palpiteiro e o cadastro de jogos realizado pelo administrador.

Tabela 4.9: Caso de uso real dar palpite

Ação – Dar palpite	
Ator:	Palpiteiro.
Pré-Condição:	-
Pós-Condição:	Palpite(s) inserido(s) no banco de dados.
Entrada:	Palpite para o(s) jogo(s).
Saída:	Lista de jogos com palpites do palpiteiro.
Fluxo de Eventos	
Ator	Sistema
1. Acessa o Website na área “Meus Palpites”.	

	<p>2. Verifica se o palpiteiro está <i>logado</i> no sistema, se..</p> <p>A) <i>Sim</i> Vai para (6).</p> <p>B) <i>Não</i> Vai para (3).</p>
	<p>3. Solicita em uma página auxiliar em estilo <i>lightbox</i>¹⁸ usuário e senha do palpiteiro.</p>
4. Informa usuário e senha para autenticação no sistema.	
	<p>5. Valida dados de entrada. Caso estejam..</p> <p>A) <i>Corretos</i> Valida os dados de entrada contra os dados do banco de dados. Novamente, estes podem estar...</p> <p>a. <i>Corretos</i> Vai para (6).</p> <p>b. <i>Incorretos</i> Volta para (3), informando o erro.</p> <p>B) <i>Incorretos</i> Volta para (3), informando o erro.</p>
	<p>6. Exibe, para cada competição que o palpiteiro disputa, todos os jogos possíveis de serem palpitados.</p>
7. Informa o placar para o(s) jogo(s) que deseja palpar.	
	<p>8. Valida dados de entrada. Caso estejam..</p> <p>A) <i>Corretos</i> Exibe mensagem de sucesso e volta para (6).</p> <p>B) <i>Incorretos</i> Volta para (6), informando o erro.</p>

¹⁸ *Lightbox*: aplicação *Javascript* usada para mostrar algum conteúdo sobreposto à página mostrada no navegador. ([http://en.wikipedia.org/wiki/Lightbox_\(JavaScript\)](http://en.wikipedia.org/wiki/Lightbox_(JavaScript))).

Na Tabela 4.9 é então detalhado o caso de uso dar palpite realizado pelo ator Palpiteiro incluindo todos os aspectos referentes a sucesso e insucesso das operações juntamente com alguns detalhes de interface. Um ponto a ser destacado é referente ao passo (6) do caso de uso, pois nele é dito que são exibidos todos os jogos possíveis de serem palpitados. Esses jogos são jogos com data e horário maior que o atual, incluindo jogos que o palpiteiro já deu o palpite. Isso porque, até o início do jogo, o palpiteiro poderá alterar seu palpite.

Tabela 4.10: Caso de uso real cadastrar jogos

Ação – Cadastrar jogos	
Ator:	Administrador.
Pré-Condição:	Autenticação do ator no gerenciador
Pós-Condição:	Jogo(s) inserido(s) no banco de dados.
Entrada:	Dados do(s) jogo(s).
Saída:	Lista de jogos da rodada.
Fluxo de Eventos	
Ator	Sistema
1. Acessa o gerenciador no menu “Competição → Rodadas”.	
	2. Exibe todas as competições cadastradas que o administrador possui.
3. Seleciona a competição que deseja cadastrar o(s) jogo(s).	
	4. Exibe as rodadas que a competição possui.
5. Seleciona a rodada que deseja cadastrar o(s) jogo(s).	
	6. Exibe todos os jogos já cadastrados para aquela rodada juntamente com a opção “adicionar jogo”.
7. Clica na opção “adicionar jogo”.	
	8. Exibe dois campos de seleção de time (time da casa e time visitante) e outros dois campos para data e horário do jogo respectivamente.
9. Seleciona os times e informa a data e o horário do jogo.	
	10. Valida dados de entrada. Caso estejam.. A) <i>Corretos</i> Exibe mensagem de sucesso e

	volta para (6). <i>B) Incorretos</i> Volta para (8), informando o erro.
--	---

Na Tabela 4.10 é então detalhado o caso de uso cadastrar jogos, que é realizado pelo ator Administrador através do gerenciador do sistema. Esse caso de uso tende a ser muito realizado tanto no início de uma competição, quando o cliente administrador deverá cadastrar os jogos da competição recém criada, quanto ao longo da competição, isso porque muitos campeonatos não têm seu calendário totalmente definido no início da competição, seja por não ter todos os confrontos definidos ou por não ter a data e horário estabelecidos.

4.2 Modelagem da base de dados

A partir do refinamento de todos os requisitos do sistema através da especificação detalhada dos casos de uso, é preciso então definir o relacionamento entre os elementos gerados e citados anteriormente. Para tanto, inicia-se então o projeto do banco de dados do sistema.

4.2.1 Modelagem inicial

Primeiramente então, é definida uma modelagem ER inicial contendo apenas as relações básicas dos principais elementos do sistema. O objetivo dessa modelagem é ter a primeira idéia da base de dados relacional proposta no projeto. Além disso, ela será o núcleo da base de dados do sistema, pois é a partir dela que são geradas todas as outras tabelas e relacionamentos necessários para atender todos os requisitos impostos ao projeto.

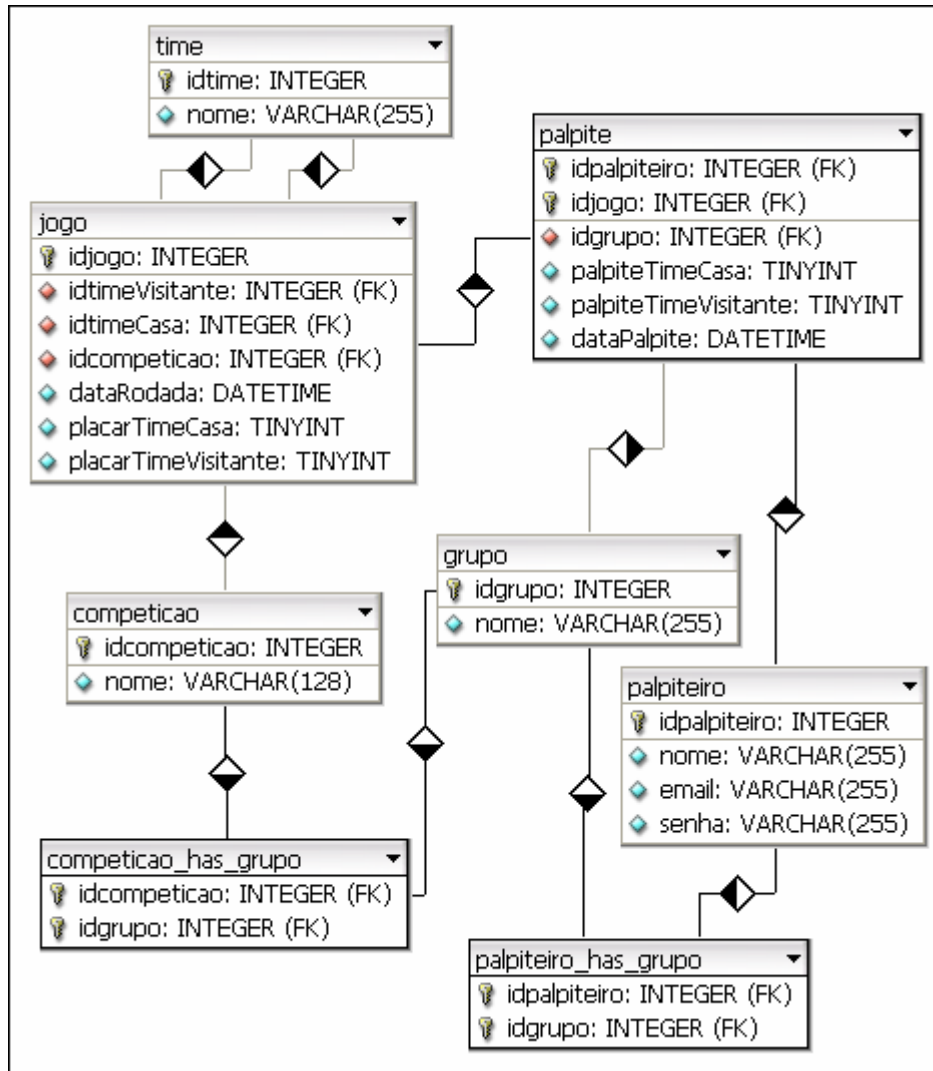


Figura 4.2: Modelagem ER inicial do banco de dados

A modelagem da Figura 4.2 é composta pela tabela *time* que contém o identificador único do time e seu nome; pela tabela *jogo* que contém dois identificadores de time, ou seja, um para o time da casa e outro para o time visitante e o identificador da competição a que o jogo pertence, esses todos como chave estrangeira. Ainda na tabela *jogo* terá o campo *dataRodada* que contém a data e o horário do *jogo*, e dois campos para o placar, ou seja, placar do time da casa e placar do time visitante.

Também há as tabelas *competicao*, com identificador único e nome da competição, *grupo*, com identificador único e nome do grupo e a tabela *palpiteiro*, essa, além de identificador único e nome do palpiteiro, contém suas credenciais de acesso ao site, ou seja, o e-mail e senha do palpiteiro. A partir dessas tabelas são geradas duas tabelas de relacionamento, *palpiteiro_has_grupo* e *competicao_has_grupo*. Essas tabelas tornam, respectivamente, a relação entre palpiteiro e grupo e a relação entre grupo e competição n para n, ou seja, um palpiteiro pode participar de vários grupos assim como um grupo é composto de vários palpiteiros e um grupo pode disputar várias competições assim como uma competição pode ser disputada por vários grupos.

Além disso, o palpiteiro sempre dará um palpite sobre o resultado de um jogo por um grupo, por isso também se tem a tabela *palpito*, contendo o identificador do palpiteiro, do jogo e do grupo, assim como o palpite para o time da casa, o palpite para

o time visitante e a data do palpite. É importante ressaltar o porquê do identificador do grupo na tabela *palpite*, isso se deve ao fato de um palpiteiro poder estar disputando uma mesma competição por dois grupos distintos. Dessa forma, é identificado o grupo ao qual o palpiteiro está palpitando e ele poderá dar palpites diferentes, se desejar, para o mesmo jogo, porém um para cada grupo.

É evidente que essas tabelas da modelagem inicial contêm outros elementos além dos apresentados na modelagem ER inicial da Figura 4.2. Como por exemplo, a tabela *Palpiteiro* contém outros dados do palpiteiro, como apelido, data de nascimento, entre outros, além de seu nome, e-mail e senha. Entretanto, somente os elementos mais relevantes do sistema foram apresentados na modelagem relacional.

A partir dessa modelagem, o núcleo da base de dados do sistema está definido, é preciso, então, partir para a análise das necessidades extras do projeto.

4.2.2 Informações de controle

No sistema haverá muitos clientes administradores, cada qual com suas competições, seus grupos de palpiteiros, seus times e jogos cadastrados. Diante disso, surge a necessidade de informações de controle para identificar quais dados pertencem a cada administrador do sistema.

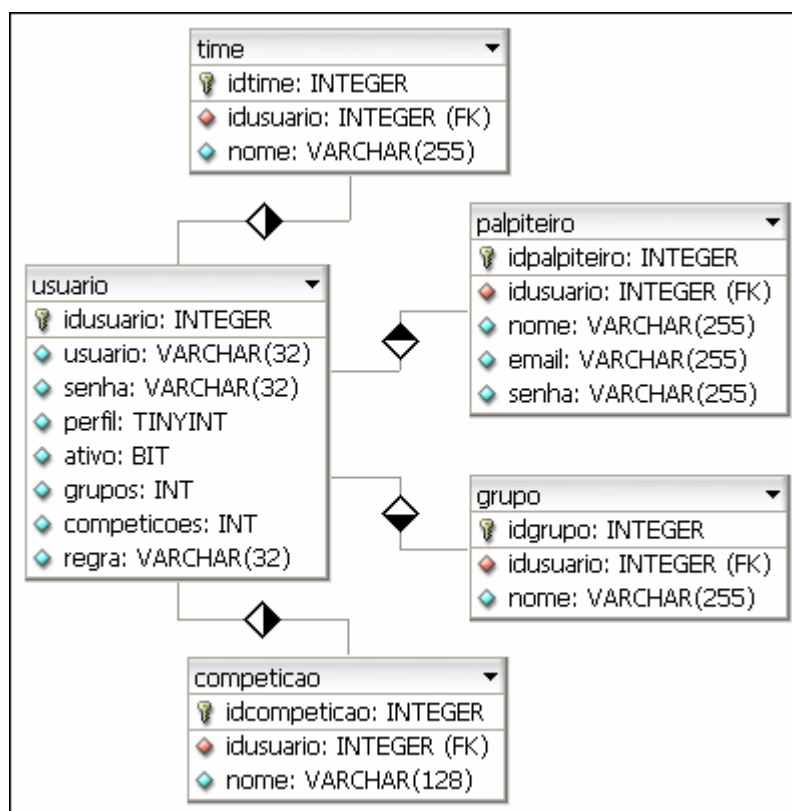


Figura 4.3: Modelagem ER com a tabela *usuario*

A tabela *usuario* irá proporcionar a adição das informações de controle necessárias no sistema. Essa tabela, além do identificador único, possui os elementos necessários para autenticação no sistema, usuário e senha. Além disso, possui o campo perfil que definirá o tipo de usuário e o campo ativo que definirá se o usuário está ou não fazendo uso do sistema. Ainda, visando impor limites ao sistema, há os campos grupos e

competições, que definem o número de grupos e o número de competições que cada usuário poderá criar no sistema.

Por fim, o campo regra define o nome da função responsável pelo cálculo da pontuação das competições do usuário, ou seja, a função onde estão definidas as regras para o bolão. O cliente administrador, ao contratar o serviço, poderá escolher um conjunto de regras já existentes disponível em alguma função de cálculo de pontuação ou informar as regras de sua escolha. No segundo caso, uma função específica para aquele cliente deverá ser criada. Posteriormente, outros clientes poderão utilizar essa mesma regra, pois a idéia é ir acumulando regras de pontuação de forma a poder oferecer uma grande variedade de regras aos clientes além, é claro, de oferecer a criação de uma própria.

Outra mudança ocorre em outras quatro tabelas que já foram descritas na modelagem inicial da seção anterior. Essa mudança é a adição do elemento *idusuario* como chave estrangeira nas quatro tabelas: *time*, *palpiteiro*, *grupo* e *competicao*. Essa informação permitirá que um cliente administrador, ao efetuar *login* no sistema, veja apenas as informações pertinentes a ele.

O campo perfil da tabela *usuario* identificará se o usuário é um cliente administrador ou se é o gerenciador do sistema. Deste modo, o gerenciador do sistema poderá cadastrar times, competições, palpites e grupos de palpites que serão visíveis a todos os clientes administradores do sistema. Para isso, simplesmente é imposta a regra de que um cliente administrador poderá visualizar os times, competições, grupos de palpites e palpites com seu próprio *idusuario* ou com um *idusuario* pertencente a um gerenciador do sistema.

Para evidenciar a grande utilidade desse mecanismo, pega-se como exemplo times conhecidos do futebol brasileiro. Esses times são cadastrados no sistema pelo gerenciador, desse modo ficarão visíveis a todos os clientes administradores. Isso evitará cadastros duplicados de times que serão usados pela maioria dos administradores.

4.2.3 Organização do sistema

Um fato importante que surge quando é estimado um crescimento no número de cadastro de times, por exemplo, é a necessidade de um nível de organização maior. Assim, a partir da modelagem inicial, são criadas duas outras tabelas para facilitar a organização do sistema.

Na Figura 4.4 é possível verificar essas novas tabelas já incluindo os relacionamentos adicionados com as tabelas anteriormente definidas, tabela *time* e tabela *competicao*.

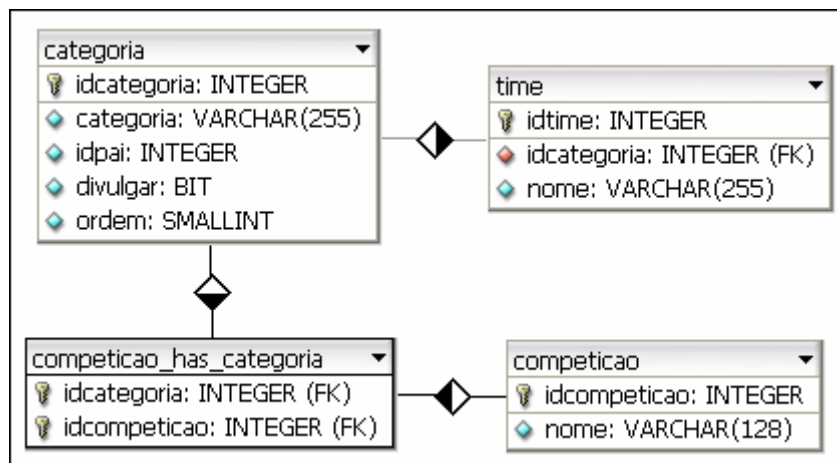


Figura 4.4: Modelagem ER com tabelas para organização do sistema

A tabela *categoria* irá prover essa organização mencionada, uma vez que sua estrutura seja em forma de árvore, permitirá uma classificação em níveis, que tanto facilitará o cadastro por parte do cliente Administrador, quanto permitirá uma melhor estruturação para a visualização no Website.

O mecanismo de classificação é bem simples, por exemplo, através da tabela *categoria* podem-se ter os seguintes registros cadastrados: “Série A”, “Série B” e “Série C” todos eles tendo como registro pai “Clube Brasileiro”. Assim, no cadastro de times é informada a categoria que o time pertence, bem como no cadastro de uma competição é informada qual ou quais categorias de times disputaram essa competição.

Essa classificação facilitará muito o cadastro de jogos, pois quando vai se cadastrar um jogo para a competição, por exemplo, Campeonato Brasileiro da série A, só serão listados os times classificados como “Clube Brasileiro → Série A”.

A visualização no site também se torna melhor, pois podemos definir um menu hierárquico para facilitar a navegação. O cliente através desse menu poderá alcançar a competição desejada através de poucos cliques.

4.2.4 Otimização de consultas

Outro ponto que foi analisado foi quanto à classificação dos participantes da competição. Uma tabela com a pontuação dos participantes seria muito mais consultada do que atualizada. Assim, visando um acesso rápido na consulta à classificação, é adicionada à modelagem uma tabela auxiliar chamada *ranking* onde contém a classificação sempre atualizada das competições. Assim a cada visualização no site não precisaria ser feito nenhum cálculo, através, apenas, de uma consulta rápida à tabela *ranking* já seria atingido o objetivo de forma eficiente.

Na Figura 4.5 tem-se essa nova tabela juntamente com os relacionamentos com as outras tabelas já existentes na modelagem inicial.

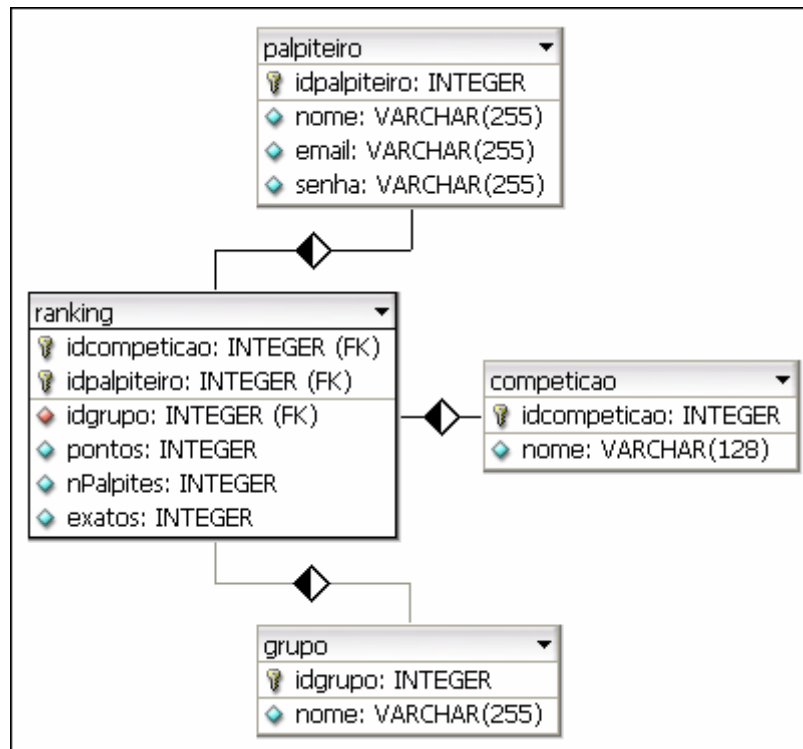


Figura 4.5: Modelagem ER com a tabela *ranking* para consultas rápidas

Na Figura 4.5 aparecem as tabelas *competicao*, *grupo* e *palpiteiro* já explicadas anteriormente e a nova tabela *ranking*. Essa tabela irá conter além dos identificadores de competição, palpiteiro e grupo, os registros *pontos*, *nPalpites* e *exatos*. O campo *pontos* conterá a pontuação atual do palpiteiro na competição. O campo *nPalpites* conterá o número atual de palpites do palpiteiro e o campo *exatos* conterá o número de placares exatos que o palpiteiro acertou. Esses dois últimos campos serão usados como critério de desempate no caso de mesmo número de pontos.

Dessa forma, para listar uma classificação, basta listar os registros da tabela *ranking* para um determinado grupo e uma determinada competição ordenados pelo campo *pontos* decrescentemente.

4.3 Definição das classes e métodos

Uma vez definida a modelagem da base de dados, a próxima etapa é a definição das classes e objetos que darão suporte às operações com o banco. Assim, primeiramente é obtido o modelo conceitual refinado a partir do modelo conceitual preliminar apresentado na seção 3.4, após, então, com base nesse segundo modelo, é gerado o diagrama de classes do sistema.

4.3.1 Modelo conceitual refinado

No modelo conceitual refinado, as classes que formam o núcleo do sistema apresentadas no modelo conceitual preliminar são complementadas, isto é, adicionam-se ao primeiro modelo classes que fazem parte do domínio do problema. Também nesse modelo, são definidas hierarquias de generalizações e agregações entre as classes.

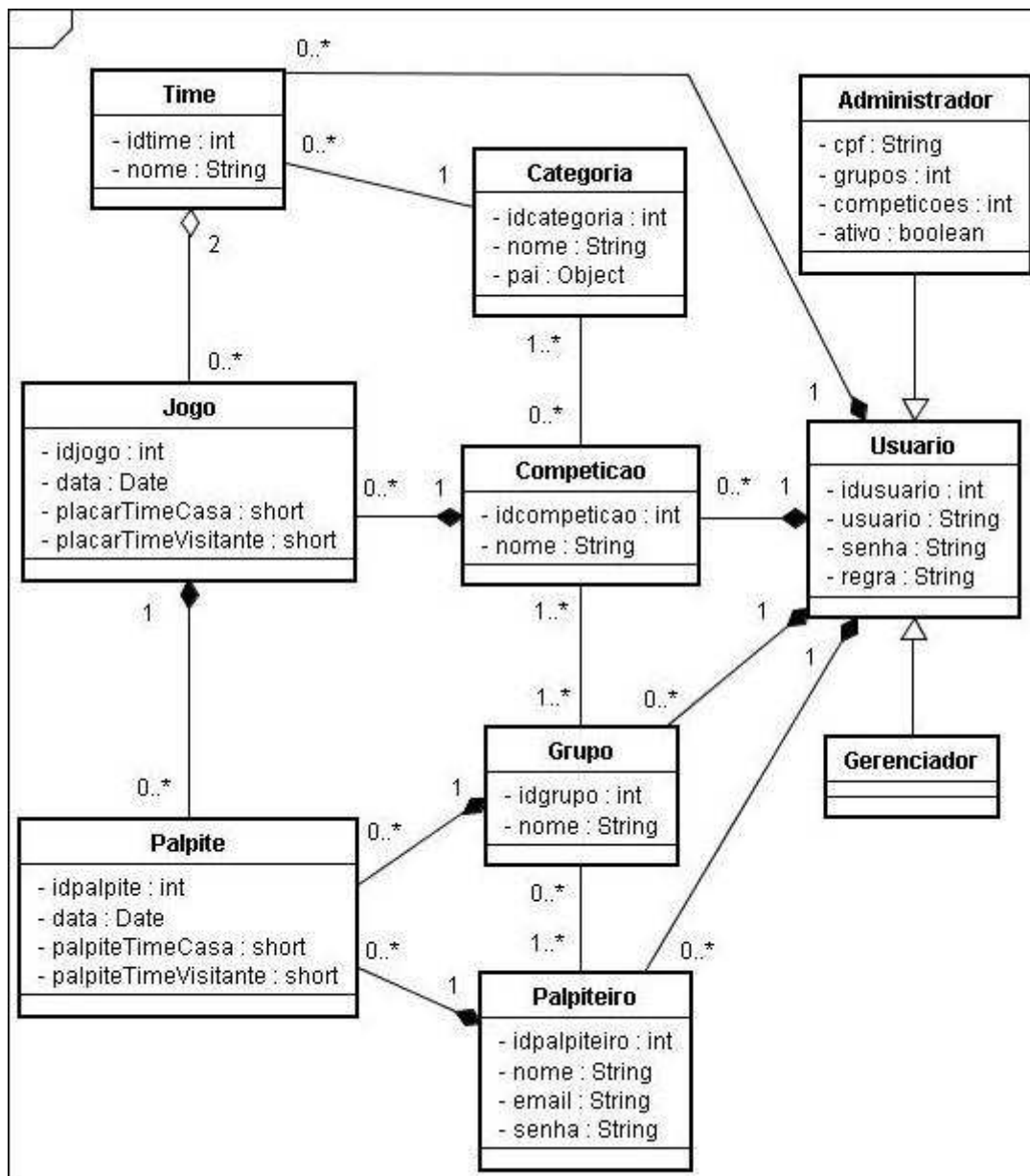


Figura 4.6: Modelo conceitual refinado

Na Figura 4.6 estão representadas as classes adicionadas ao modelo conceitual preliminar da seção anterior, são elas: Time, Categoria, Usuario, Administrador e Gerenciador.

A classe Time possui o identificador único e o nome do time. Ela possui uma relação de agregação com a classe Jogo, ou seja, o objeto da classe Jogo conterá exatamente dois objetos instanciados da classe time, o time da casa e o time visitante.

A classe Categoria possui, além do identificador único e o nome da categoria, o atributo pai que é um objeto da própria classe Categoria, possibilitando assim uma hierarquia entre as categorias do sistema. Ela possui associação com a classe Time e com a classe Competicao. Um time pertencerá a exatamente uma categoria, já uma competição poderá conter jogos envolvendo times de várias categorias, por isso a associação uma ou mais relacionando a classe Competicao com a classe Categoria.

Por fim, a classe Usuário contém os atributos necessários para autenticação do usuário no gerenciador do sistema, o atributo regra para identificar a função com o conjunto de regras de pontuação para as competições e duas generalizações, a classe Administrador e a classe Gerenciador. Isso se deve ao fato de um usuário poder ser ou administrador ou gerenciador do sistema.

No caso do usuário ser administrador, ele possuirá algumas informações extras, como cpf, grupos, competições e ativo. O campo cpf é utilizado na identificação do cliente administrador. Já os campos grupos e competições conterão respectivamente o número de grupos e número de competições que aquele cliente administrador poderá criar no sistema. O campo ativo será útil quando o responsável pelo sistema desejar, por alguma razão, bloquear o acesso do cliente administrador por algum período. Uma vez que somente clientes ativos poderão efetuar a autenticação no sistema, cria-se, assim, a possibilidade de restringir o acesso sem necessariamente excluir o cadastro do cliente.

A classe Usuario possui relação de composição com as classes Time, Competicao, Grupo e Palpiteiro. Isso porque cada time, competição, grupo ou palpiteiro inserido no sistema foi cadastrado por um usuário, assim é possível que registros inseridos pelo gerenciador fiquem disponíveis a todos, já registros inseridos por clientes administradores ficam disponíveis apenas ao seu criador.

4.3.2 Diagrama de classes

Embora no modelo conceitual refinado as classes do sistema foram complementadas, ainda se tem apenas as classes específicas do domínio do problema, assim, por exemplo, classes para interface do sistema não são abordadas nesse modelo. Da mesma forma, embora seja usada a notação do diagrama de classes UML, apenas os atributos são representados no modelo conceitual, o detalhamento dos métodos das classes não são tratados.

Assim, visando ter uma representação completa da estrutura do sistema, é elaborado o diagrama de classes do projeto. Esse diagrama espelha tudo que foi realmente implementado e construído para o núcleo do sistema.

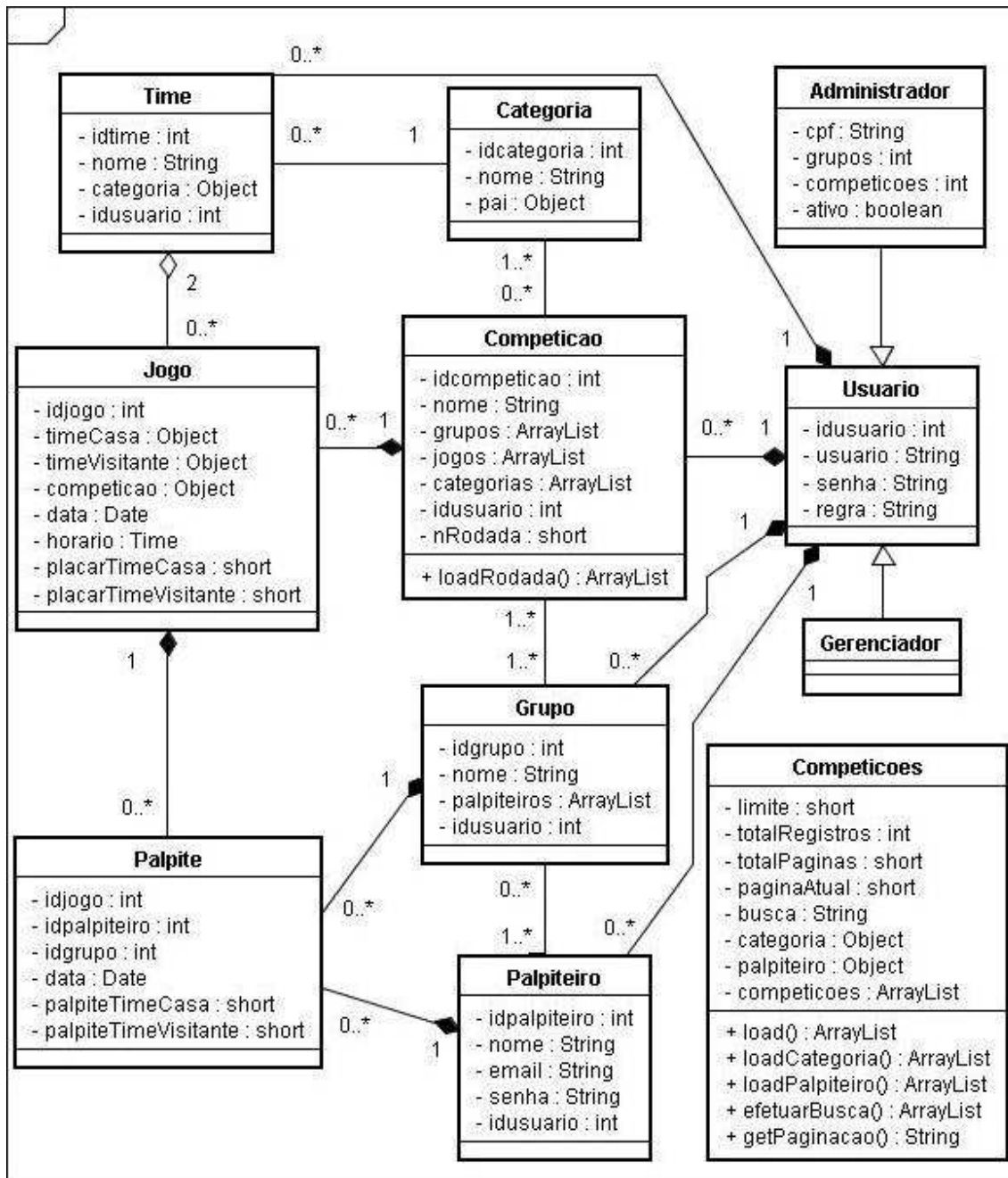


Figura 4.7: Diagrama de classes

Primeiramente então, é preciso mencionar que, embora não estejam representados no diagrama de classes da Figura 4.7, em todas as classes do sistema há métodos *get()* e *set()* para todos atributos da classe. O método *get()* simplesmente retorna o valor do atributo, já o método *set()* recebe como parâmetro um valor e seta-o ao atributo.

Para todas as classes, agora se excetuando a classe *Competicoes*, há também métodos que são responsáveis pela persistência com o banco de dados, são os métodos *load()* e *save()*. O primeiro deles ao ser chamado, receberá como parâmetro um identificador único, e então fará a conexão com o banco de dados buscando as informações de todos os atributos da classe para esse identificador. Já o segundo, ao ser chamado, pegará todas as informações contidas nos atributos da classe e fará a inserção ou atualização no banco de dados.

Esses métodos mencionados seguem o mesmo princípio de funcionamento independente da classe ou tipo de dados que estão sendo tratados. Já o método *loadRodada()* da classe *Competicao* apresenta uma pequena variação do método original *load()*, pois ao invés de receber o identificador da competição, ele recebe o identificador da rodada e busca todos os jogos da competição para aquela rodada. Esse é o motivo da criação do atributo *nRodada*, pois assim é possível organizar uma competição em rodadas, facilitando a visualização e até mesmo o cadastramento dos jogos.

No diagrama de classes, é possível perceber que surgiu uma classe nova em relação ao modelo conceitual refinado da seção anterior, a classe *Competicoes*. Essa classe está relacionada exclusivamente à exibição no site, ela possui atributos e métodos visando à paginação e métodos de carregamento que se adaptam a necessidade em questão de visualização. Esses métodos de carregamento são centralizados na função *load()*, que nessa classe é diferenciada em relação as demais classes do sistema. Na Figura 4.8 segue o trecho de código do método *load()* da classe *Competicoes*.

```
function load($idcategoria, $idpalpiteiro, $wordkey, $orderby, $ordenamento)
{
    if ($idcategoria)
    {
        $this->categoria= new Categoria();
        $this->categoria->load($idcategoria);
        $this->loadCategoria($idcategoria, $orderby, $ordenamento);
    }
    elseif ($wordkey)
    {
        $this->busca = $wordkey;
        $this->efetuarBusca($this->busca, $orderby, $ordenamento);
    }
    else
    {
        $this->palpiteiro= new Palpiteiro();
        $this->palpiteiro->load($idpalpiteiro);
        $this->loadPalpiteiro($idpalpiteiro, $orderby, $ordenamento);
    }
}
```

Figura 4.8: Código do método *load()* da classe *Competicoes*

No código tem-se o método *load()* que é responsável pelo carregamento das competições de acordo com a necessidade do usuário. Supõe-se então que o usuário deseja listar as competições de uma determinada categoria, então o método *load()* é chamado com o identificador da categoria e neste caso efetua-se o método *loadCategoria()*.

Agora, supõe-se que o usuário utilizou os mecanismos de busca do site, assim o método *load()* é chamado com uma palavra chave para a busca e o método *efetuarBusca()* é então acionado carregando todas as competições que se identificam com a palavra chave informada.

Em outro caso agora, supõe-se que o cliente efetuou a autenticação no sistema e deseja exibir apenas as competições que ele participa, então o método *load()* é chamado com o identificador desse cliente, ou seja, o *idpalpiteiro*. Nesse caso é invocado o

método *loadPalpiteiro()* e todas as competições que aquele cliente disputa são carregadas.

Os demais parâmetros, *orderby* e *ordenamento* são utilizados para oferecer diferentes possibilidades de ordenação dos registros ao cliente independentemente do tipo de registro em que está sendo feita a visualização.

5 IMPLEMENTAÇÃO

A implementação pode ser vista dividida em dois grandes grupos, o gerenciador¹⁹ do sistema, através do qual os clientes administradores criam e administram os bolões e o Website²⁰, por onde os clientes palpiteiros fazem o acesso para disputar e acompanhar os bolões em que participam. Nas seções seguintes, são então apresentadas essas duas caras do sistema aprofundando os principais detalhes de cada uma delas.

5.1 Gerenciador

O gerenciador é de uso exclusivo para clientes administradores e para o responsável pela gerencia do sistema como um todo, assim, para qualquer ação nesse sistema, é necessária a autenticação do usuário.

5.1.1 Estrutura básica

A estrutura do gerenciador basicamente consiste em um menu fixo de opções em dois níveis e o conteúdo principal, que é onde os dados correspondentes a cada uma das opções do menu são exibidos. Na Figura 5.1 é possível verificar a interface do gerenciador, nela está exemplificada o cadastro de jogos para uma determinada rodada de uma competição.

¹⁹ Gerenciador Palpigol: Disponível em (<http://www.inf.ufrgs.br/~pdapont/manager>).

²⁰ Website Palpigol: Disponível em (<http://www.inf.ufrgs.br/~pdapont>).

Gerenciador Palpigol | Usuário: fernando | Data: 25.10.2009 | SAIR

Rodadas
Filtro

Competição:

Rodada 2

« Anterior | Próxima » | Salvar | Excluir | Adicionar Jogo

<input type="checkbox"/>	Time da casa	x	Time visitante	Data	Horário
<input type="checkbox"/>	Portugal	x	Brasil	30/10/2009	14:00
<input type="checkbox"/>	Chile	x	Gana	30/10/2009	16:30
<input type="checkbox"/>	<input type="text" value="Selecione"/> <ul style="list-style-type: none"> Selecione Argentina Brasil Chile Gana Holanda Inglaterra Portugal 	x	<input type="text" value="Selecione"/>	<input type="text"/> <input type="text"/>	<input type="text"/> h <input type="text"/> min

1 2 3
Rodada 2 de 3

Figura 5.1: Interface do gerenciador Palpigol

Analisando então a Figura 5.1, além das credenciais de identificação do sistema e usuário que ficam fixas no topo, tem-se no menu lateral à esquerda as opções para o CRUD dos elementos do sistema, ou seja, CRUD de palpitesiros, de grupo de palpitesiros, de times, de competições e de rodadas e jogos. A opção escolhida no menu, assim como no exemplo, sempre é exibida à direita dele, geralmente em uma página de listagem com opções de busca ou filtro, conforme necessidade, além de opções de inserção, alteração ou exclusão de registros.

No exemplo da Figura 5.1, está sendo feito o cadastro de um jogo da segunda rodada da competição nomeada como Copa do Mundo 2010. Para esse cadastro, o cliente administrador pode fazer uso do botão “Adicionar Jogo” e ir adicionando quantos jogos forem necessários para aquela rodada. Para o cadastro de um jogo, o administrador seleciona o time da casa e o time visitante através dos campos de seleção e informa a data e horário do jogo.

Para o cadastro de jogos de outras rodadas, são disponibilizados os botões: “Anterior” e “Próxima”, assim como a numeração das rodadas, facilitando assim a navegação pelas rodadas da competição. Após alcançada a rodada desejada, basta realizar o mesmo procedimento de cadastro de jogos. Para cadastrar jogos de outras competições, basta alterar, através do campo de seleção, a competição selecionada no filtro disposta no topo da página.

Embora a Figura 5.1 apresente um exemplo específico de cadastro no sistema, ela passa a idéia da estrutura básica do gerenciador, ou seja, os demais cadastros apesar de apresentarem algumas variações, basicamente seguem o mesmo padrão.

5.1.2 Funcionalidades restritas

No exemplo da seção anterior, a interface exibida do gerenciador corresponde a um acesso por algum cliente administrador. Quando o acesso é feito pelo responsável pelo gerenciamento de todo sistema, algumas funcionalidades extras e restritas são disponibilizadas. Na Figura 5.2 é exemplificado esse acesso.

Figura 5.2: Cadastro de usuários no gerenciador Palpigol

Agora, embora a interface seja idêntica a da Figura 5.1 da seção anterior, pode-se perceber novas funcionalidades disponíveis no menu lateral esquerdo, como o CRUD de categorias e também opções administrativas do próprio sistema.

No exemplo da Figura 5.2, está sendo feito o cadastro de um cliente administrador no sistema. Nesse cadastro, além dos dados de identificação do cliente e de acesso ao sistema, o responsável pelo cadastro define privilégios ao cliente que está sendo cadastrado. Esses privilégios servirão para dar opções de planos aos clientes, por exemplo, poderá haver clientes interessados em criar e disputar somente uma competição com seu grupo de amigos, porém outros poderão ter interesse em criar várias competições e diversos grupos.

Além disso, esses privilégios permitem limitar os cadastros por parte dos clientes administradores, não apenas facilitando o controle do sistema como também evitando o uso indevido.

5.2 Website

Ao contrário do gerenciador, o Website é aberto ao público em geral, pois é através dele que é feita a divulgação do sistema visando atrair outros clientes, para tanto, ele possui um layout comercial expondo todas as informações para que qualquer pessoa ao efetuar o acesso ao site seja totalmente informada a respeito de todas as funcionalidades oferecidas pelo sistema.

A seguir então, na Figura 5.3, é apresentada a interface do Website através da página inicial do site Palpigol.

The screenshot displays the Palpigol website interface. At the top, there is a green header with the Palpigol logo on the left and a login form on the right. The login form includes fields for 'E-mail' and 'Senha' (Password), an 'Entrar' (Login) button, and a 'Cadastre-se' (Sign up) link. Below the header is a navigation menu with a search bar and links for 'Meu cadastro', 'Meus palpites', 'Minhas competições', 'Ajuda', and 'Contato'. The main content area is divided into several sections: 'Clubes Brasileiros' (with links for Série A, B, C, and D), 'Clubes Estrangeiros' (with links for Alemão, América Latina, Espanhol, Inglês, Italiano, and Português), and 'Seleções' (with links for África, América do Sul, Ásia, Europa, and Oceania). The 'Destaques' (Highlights) section features four featured events: 'Campeonato Italiano', 'Copa do Mundo 2010', 'Jogos do Juventude', and 'Copa do Brasil'. Below this is an 'Outras' (Other) section with four more events: 'Grenal 2009', 'Libertadores da América', 'Brasileirão 2009', and 'Mundial Interclubes'. At the bottom of the page, there is a footer with the Palpigol logo, a navigation menu for 'Nossa Empresa', 'Como Funciona', 'Como Participar', 'Regulamento', 'Mapa do Site', and 'Contato', and a yellow banner for 'SENAWEB' with a cartoon character and a bomb. The footer also includes copyright information and the text 'Seja também um patrocinador PALPIGOL' and 'Desenvolvido por: bnrlemos / pdapont'.

Figura 5.3: Interface do Website Palpigol

Na Figura 5.3, além de apresentar o layout do Website, é possível perceber sua estrutura básica, composta de topo, menu lateral esquerdo, conteúdo e rodapé.

5.2.1 Usabilidade

Tida como prioridade desde o início do projeto, a usabilidade do sistema é verificada através de pontos que simplificam as operações realizadas pelos clientes palpiteiros. Assim, no projeto da estrutura do Website sempre se tentou estudar a melhor forma para

dispor as informações ao cliente de forma organizada e tendo o menor número de cliques possíveis para serem alcançadas.

Dessa forma, por exemplo, quando disputando uma competição, a maior parte dos acessos do palpiteiro ao Website será para palpitar e visualizar informações referentes ao seu bolão. Para isso, será necessária a autenticação no sistema, assim as informações para efetuar *login* sempre ficam fixas no topo do site, disponíveis em todas as páginas.

Também sempre disponível no topo, está o menu horizontal de opções para o palpiteiro sempre ter acesso rápido às informações que deseja sobre o seu bolão. Nas três principais opções desse menu, está a opção “Meu cadastro” através da qual o palpiteiro poderá editar seus dados cadastrados, por exemplo alterando sua senha se desejar; a opção “Meus palpites” onde são exibidos todos os palpites já realizados e os que estão disponíveis para palpitar e a opção “Minhas competições” onde o palpiteiro poderá visualizar todas informações referentes às competições que disputa.

5.2.2 Navegação

Na página inicial do Website, são exibidas algumas competições que são destaque e outras escolhidas aleatoriamente. Ao escolher alguma, são exibidos então todos os detalhes e informações referentes àquela competição. Contudo nem sempre a competição desejada estará na capa do site, assim são disponibilizadas outras duas formas de navegação em busca de alguma competição específica.

A primeira delas é disponibilizada no menu lateral esquerdo que contém as categorias do sistema organizadas hierarquicamente em dois níveis. Dessa forma, sempre que uma opção é acionada, são carregadas na área de conteúdo todas as competições pertencentes àquela categoria selecionada.

A segunda delas é disponibilizada no menu horizontal, que é o mecanismo de busca do site, através desse mecanismo o cliente informa uma palavra chave e então é feita uma pesquisa na base de dados buscando competições que contém essa palavra em seu nome ou sua descrição. Assim, todas as competições encontradas são listadas na área de conteúdo ficando disponíveis ao cliente.

5.2.3 Institucional

Além do conteúdo vital para o funcionamento do sistema mencionado nas seções anteriores, outras informações se tornam necessárias no Website. Uma vez que a divulgação do sistema é feita através do Website, conforme mencionado no início desse capítulo, é necessário páginas institucionais com informações sobre o sistema.

Assim, é disponibilizado no menu horizontal do topo as opções de “Ajuda” e “Contato”. A primeira delas é atrelada ao menu de opções contidas no rodapé do site, com páginas contendo informações do regulamento, de como funciona, de como participar de bolões, entre outras. Já a opção para contato, é um atendimento oferecido para que clientes possam esclarecer suas dúvidas, solicitar serviços, fazer reclamações ou sugestões, tudo tendo em vista a melhoria do sistema como um todo.

No rodapé do site também é reservada uma área para publicidade, onde poderão ser feitos anúncios e propagandas através de *banners* animados, tudo isso tendo em vista atrair patrocinadores para o sistema.

5.2.4 Otimização

A otimização do Website tem como principal objetivo aumentar a sua visibilidade para os mecanismos de busca da Internet e assim, conseqüentemente, aos possíveis clientes. Dessa forma, o objetivo dessa seção é demonstrar como algumas técnicas básicas de otimização de sites foram aplicadas no projeto.

Primeiramente, então, um dos fatores mais importantes para a otimização de um site é o conteúdo das páginas, que precisa ser trabalhado para oferecer qualidade nas informações passadas além de possuir palavras-chave ao longo de sua descrição (LEDFOURD, 2008). Assim, foram criados os campos para resumo e para descrição das competições. Dessa forma, competições abertas ao público cadastradas pelo responsável pelo gerenciamento do sistema e, até mesmo, competições cadastradas por clientes administradores, poderão ter os textos muito bem trabalhados visando sempre a otimização do site.

Para exemplificar o uso desses textos, pode-se citar, por exemplo, uma competição contendo apenas os jogos do Brasil na Copa do Mundo 2010. Para o campo resumo dessa competição, um bom texto a ser cadastrado seria: “Bolão dos jogos do Brasil na Copa do Mundo 2010 na África do Sul. Convide seus amigos e palpite sobre todos os jogos do Brasil na Copa da África do Sul”. Esse texto apresenta de forma resumida uma boa descrição para a competição além de palavras-chave como: bolão, palpite, Brasil, Copa do Mundo, África do Sul. O texto de resumo é específico para as páginas de listagem de produtos ou para a página inicial. Para a página de competição, é apresentado o texto descrição que será muito maior, podendo, assim, conter mais informações sobre a competição e também mais palavras-chave.

Outro fator muito importante para a otimização são as Meta tags²¹, pois elas ajudam na identificação e na classificação do conteúdo das páginas do site.

```

<!-- meta tags -->
<meta name="DC.title" content="Clubes Brasileiros > Série A - Palpigol
- Portal de palpites e Bolões personalizados" />
<meta name="DC.identifier" scheme="DCTERMS.URI" content=
"http://www.palpigol.com.br/competicoes/clubes-brasileiros/serie-a/6" />
<meta name="description" content="Aqui você pode criar e personalizar
seus bolões, definindo os jogos e as regras para o bolão! Convide seus
amigos e palpite sobre jogos do campeonato brasileiro, copa do brasil,
estaduais ou apenas de seu time ou times de sua escolha." />
<meta name="keywords" content="Palpigol, bolão, bolão campeonato
brasileiro, bolão do brasileirão, bolão série A, campeonato brasileiro,
brasileirão, bolão de futebol, criar bolão, bolões, bolão jogos futebol,
palpite, palpites, palpiteiro, palpite campeonato brasileiro, palpite
jogos brasileirão" />
<meta name="rating" content="general" />
<meta name="robots" content="all" />
<!-- / meta tags -->

```

Figura 5.4: Exemplo de Meta tags para uma categoria de competições

²¹ Meta tags: Meta tags são linhas de código HTML, ou "etiquetas", que, entre outras coisas, descrevem o conteúdo do site para os buscadores (LEDFOURD, 2008).

Na Figura 5.4, são apresentadas algumas Meta tags fundamentais para a otimização do site. O exemplo de código pertence à página de listagem de competições quando acessada para a subcategoria Série A pertencente à categoria Clubes Brasileiros. No exemplo, primeiramente aparecem as Meta tags nomeadas como *DC.title* e *DC.identifier*, elas são responsáveis, respectivamente, pelo título e identificação do documento para o mecanismo de busca.

Já a Meta tag com nome *description* contém uma descrição resumida de todo conteúdo da página, que embora seja de impacto pouco relevante em posicionamento do site, é um influenciador direto do clique do usuário em sua busca, por isso da importância de uma Meta tag *description* chamativa e com qualidade no seu conteúdo.

A Meta tag com nome *keywords* conterá as palavras-chave do conteúdo da página, ou seja, as palavras que o usuário informará nos mecanismos de busca. Assim, é fundamental conter nesse campo as palavras importantes da página e listá-las juntamente com suas variações e sinônimos.

Por fim, as Meta tags com nome *rating* e *robots* são usadas, respectivamente, para classificação do conteúdo da página e para garantir a indexação das páginas do site nos mecanismos de busca.

Outros fatores importantes também para a otimização são os títulos das páginas e as URL²²s do site (LEDFORD, 2008). Assim, para ambos, são adotados padrões para ter sempre uma descrição única, criativa e intimamente ligada ao conteúdo da página em questão. Um exemplo dessa aplicação pode ser a página de listagem de categorias, onde um de seus títulos poderá ser: “Clubes Brasileiros > Série A – Palpigol – Portal de palpites e Bolões personalizados”. Nesse título é possível perceber a padronização: categoria > subcategoria – nome do site – complemento.

Na URL dessa página também conterá informações pertinentes ao conteúdo que está sendo exibido de forma amigável ao usuário, ou seja, padronizada, com palavras-chave e curta, facilitando assim a divulgação e memorização dela. Um bom exemplo de URL para essa página seria, após o endereço do site, “/competicoes/clube-brasileiro/serie-a”. É possível perceber a padronização “pagina/categoria/subcategoria” nesse exemplo.

5.3 Palpigol x Concorrentes

Conforme introduzido no início do texto, há sites que oferecem um serviço semelhante a seus clientes. Diante disso, torna-se essencial, ao final do projeto, uma comparação com a concorrência destacando pontos que são considerados importantes para alcançar o sucesso nesse modelo de negócio.

Para essa comparação com o Palpigol, então, foram analisados os sites BolaoMania, Virtual Bolão, ESPN e BolaoVip. A análise basicamente pode ser dividida em dois grandes grupos: serviços oferecidos e usabilidade.

²² URL: (acrônimo para a expressão inglesa *Uniform Resource Locator*), em português Localizador de Recursos Universal, é o endereço de um recurso, nesse caso, o endereço dos arquivos do site.

5.3.1 Serviços oferecidos

A análise das funcionalidades bem como dos serviços oferecidos tem como objetivo verificar o que cada sistema disponibiliza a seus clientes. Assim, é apresentada na Tabela 5.1 uma lista de funcionalidades e serviços considerados essenciais para o modelo de sistema proposto. Então, para cada um deles, foi analisado se os quatro sistemas mencionados no início da seção e o Palpigol o disponibilizavam e, em alguns casos, a maneira que o disponibilizavam.

Tabela 5.1: Palpigol x Concorrentes – serviços oferecidos

<i>Serviço</i>	<i>Palpigol</i>	<i>BolaoVip</i>	<i>ESPN</i>	<i>Virtual Bolão</i>	<i>BolaoMania</i>
Criar bolão pessoal	✓	✓	✓	✗	✗
Criar uma competição	✓	✗	✗	✗	✗
Inclusão de times	✓	✗	✗	✗	✗
Definir regras para a pontuação	✓	✗	✗	✗	✗
Definir pesos para os jogos	✗	✗	✗	✗	✗
Oferece prêmios	✗	✗	✓	✓	✓
Número de competições oferecidas ao público	reduzido	alto	baixo	baixo	baixo
Serviços gratuitos	variável	maioria	todos	nenhum	todos

Analisando a Tabela 5.1 é possível perceber que o grande diferencial do Palpigol é oferecer a personalização dos bolões, ou seja, definir quais serão os jogos, times e regras para o bolão. A idéia de definir pesos para os jogos é algo que poderá ser acrescentado no futuro ao sistema, sendo, assim, mais um diferencial, uma vez que nenhum concorrente apresenta essa flexibilidade.

Já quanto ao oferecimento de prêmios, a princípio não é o objetivo do sistema. Agora quanto ao número de competições oferecidas ao público, é reduzido porque a idéia principal consiste em vender a personalização ao cliente, mesmo assim será bem maior que os concorrentes que apresentam um número baixo de competições. Para ter idéia dos valores, baixo representa uma ou duas competições, já alto representam mais de cem competições. O número atribuído ao Palpigol é dito reduzido, pois a meta é ter em torno de dez competições na primeira divulgação do sistema.

Por fim, excetuando-se o BolaoVip e o Virtual Bolão, os outros sistemas não tem custo. O BolaoVip disponibiliza todos serviços gratuitos para bolões pessoais, já para bolões empresariais é mencionado um custo. Já o Virtual Bolão não possui nenhum serviço gratuito, para participar de qualquer bolão há um custo. Agora o Palpigol, nesse item, é descrito como variável porque se acredita que, ao mesmo tempo em que é feita a divulgação do sistema através dos bolões gratuitos, é vendida a personalização das competições para os bolões aos clientes. Assim, o valor desse campo está diretamente

relacionado com a quantidade de vendas, sendo o grande objetivo ter, com o tempo, apenas alguns serviços gratuitos e a maior parte deles estarem sendo pagos pelos clientes.

5.3.2 Usabilidade

Já a análise da usabilidade tem como objetivo verificar como e com que qualidade cada sistema oferece seus serviços aos clientes.

Tabela 5.2: Palpigol x Concorrentes – usabilidade

<i>Item</i>	<i>Palpigol</i>	<i>BolaoVip</i>	<i>ESPN</i>	<i>Virtual Bolão</i>	<i>BolaoMania</i>
<i>Login na página inicial</i>					
<i>Login em todas páginas</i>					
<i>Regulamento</i>					
<i>Atendimento/Contato</i>					
<i>Busca no site</i>					
<i>Mapa do site</i>					
<i>Cliques para palpitar</i>	2	3	3	4	3
<i>Qualidade das informações</i>	alta	alta	alta	baixa	baixa
<i>Poluição visual</i>	baixa	média	média	reduzida	alta

Analisando a Tabela 5.2 é possível perceber que, mesmo que para a maioria das operações no sistema é necessária a autenticação do cliente, não são todos os sistemas que deixam sempre disponíveis as informações para o *login* no site. Já as informações pertinentes ao regulamento e também o atendimento via mensagem eletrônica para o esclarecimento de dúvidas do cliente estão disponíveis em todos os sistemas.

A busca no site e o mapa do site são importantes para auxiliar o usuário na navegação pelo site, porém apenas um concorrente os disponibilizam. Agora quanto ao número de cliques para palpitar, o Palpigol consegue superar todos os concorrentes, pois uma vez que é a razão responsável pela maior parte dos acessos ao site por parte dos atuais clientes, o objetivo foi sempre ter esse número o menor possível.

Por fim, então, a qualidade das informações juntamente com a poluição visual são fatores importantes que farão um visitante permanecer no site. A análise da qualidade das informações foi baseada principalmente no quão bem os sistemas esclarecem seus serviços oferecidos ao longo dos textos do site, ou seja, se fica claro ao visitante todas as funcionalidades que o site dispõe e os custos associados a elas sem a necessidade do visitante ter que fazer uso de atendimento através de e-mail ou telefone.

Já a poluição visual foi baseada principalmente na quantidade de publicidade que os sistemas exibem ao longo das páginas do site e também na qualidade de exibição do conteúdo. Essa poluição pode ser considerada baixa no Palpigol, inclusive inferior ao

Virtual Bolão, no qual ela é reduzida, porque, além da preocupação constante ao longo do projeto com a exibição de informações e a criação do *layout* do sistema visando principalmente à usabilidade, inicialmente não está previsto (mesmo já tendo áreas reservadas para isso no Website) nenhum patrocinador ou publicidade para o sistema.

Assim, mesmo vindo a adquirir patrocinadores no decorrer do tempo, esse nível de poluição visual poderá se manter inferior à maioria dos atuais concorrentes. Isso porque a principal fonte de renda do Palpigol tende a ser a personalização das competições, ao contrário dos outros sistemas que parece ser a publicidade mesmo.

6 CONCLUSÃO

Nesse trabalho foi feito o projeto de um sistema Web para a criação de disputas em grupos com base em acertos de placares para jogos desportivos. Para tanto, foi realizada a especificação detalhada de toda arquitetura do sistema juntamente com o detalhamento de sua implementação. Da implementação desse sistema, então, nasceu o Palpigol oferecendo diferencias em relação aos concorrentes existentes como a criação de bolões com o cadastro de competições, times e jogos a critério do cliente.

Assim, conforme pode ser verificado nas seções anteriores, o Palpigol vem a ser um sistema Web de entretenimento que, mesmo permitindo jogos de qualquer modalidade esportiva, tem seu foco centrado no futebol por ser a paixão nacional do brasileiro, atingindo, dessa forma, um público alvo mais representativo.

Diante disso, o próximo passo consiste em disponibilizar o sistema para pelo menos um grupo de amigos disputar uma competição, pois essa validação junto a usuários reais é de extrema importância para detectar eventuais problemas ou melhorias nos serviços oferecidos, além de ser um excelente modo de testar o sistema. Após essa validação, o grande desafio será, então, efetivamente disponibilizar o sistema ao grande público visando os principais campeonatos nacionais e internacionais do próximo ano, criando bolões abertos ao público para não só atrair clientes como também divulgar o sistema.

Embora, ao final do projeto, seja possível afirmar que o grande objetivo, ter o sistema finalizado e apto a ser disponibilizado na grande rede, foi atingido, muitas melhorias e possibilidades de expansão podem ser sugeridas. Porém isso certamente depende da aceitação do público, pois conforme o aumento da procura pelo sistema, consequentemente poderá ser desenvolvido novas funcionalidades com o objetivo de não só aumentar a flexibilidade do sistema como também melhorar a qualidade dos serviços já disponíveis.

Nesse contexto, então, enquadra-se a possibilidade de automatizar o cadastro de placares para os jogos. Atualmente, tanto o cliente administrador quanto a pessoa responsável pela gerência do sistema devem sempre imputar os resultados dos jogos ao término de uma rodada. A automatização consiste em oferecer a possibilidade desses resultados serem buscados em sites que realizam cobertura esportiva e disponibilizam arquivos com esses placares para leitura e, após, inseridos automaticamente no sistema.

Ainda como melhoria do sistema, entra o cadastro de regras de pontuação para as competições criadas pelo cliente. Atualmente há regras pré-definidas para a pontuação e, ao contratar o serviço, o cliente poderá fazer uso dessas regras já existentes ou informar as suas próprias regras. Nesse segundo caso, uma função específica de cálculo de pontos deverá ser criada para aquele cliente. Contudo, com o crescimento do número de clientes, far-se-á necessário o aperfeiçoamento desse método.

A idéia que surge então é ter um cadastro de regras amplo, no qual constam todas as regras possíveis de serem aplicadas no cálculo da pontuação das competições. O responsável por esse cadastro seria a pessoa que gerencia todo o sistema e caberia ao cliente apenas, no momento em que cadastra sua competição, marcar se deseja ou não aplicar tal regra para a competição.

Agora mais como possibilidade de expansão, algo que, embora simples de ser implementado, aumentaria a flexibilidade do sistema é o cadastro de pesos para os jogos. Assim, independente das regras de pontuação escolhidas, o cliente administrador poderia arbitrariamente definir pesos para quaisquer jogos de suas competições. Isso possibilitaria aumentar a pontuação para acertos de placares, por exemplo, de um clássico local ou de uma final de campeonato.

Ainda como sugestão de trabalhos futuros, entra o cadastro de dinheiro fictício. Isso permitiria ao cliente administrador definir custo para palpites assim como prêmios para os primeiros colocados. Assim, no momento da criação do bolão, por exemplo, o cliente administrador informa que cada palpite custa n unidades do dinheiro ao palpiteiro e também define os percentuais do arrecadado que serão destinados a cada colocação na tabela de pontuação do bolão. De posse desses valores, o sistema, ao longo da competição, mantém cada palpiteiro informado do valor que ele está ganhando no momento. Como é esperado que muitos clientes utilizem o sistema para apostas com dinheiro real entre seus grupos, certamente esse mecanismo facilitaria muito o controle das apostas ao administrador.

Portanto, há muito que pode ser explorado ainda nesse modelo de negócio, porém, assim como muitos sistemas que surgiram simples, foram evoluindo devido à aceitação do público e hoje são referências na Internet, o Palpigol será expandido em conjunto com o crescimento e o interesse do público.

REFERÊNCIAS

- SOMMERVILLE, I. **Software Engineering**. Readings, 2003.
- PRESSMAN, R. **Engenharia de Software**. Makron Books, 2006.
- LARMAN, C. **Utilizando UML e padrões**. 3rd ed. Bookman, 2007.
- OBJECT MANAGEMENT GROUP. **OMG: UML 2.0 Superstructure Specification**. Disponível em: <<http://www.omg.org/>>. Acesso em: set. 2009.
- SOARES, W. **PHP 5**. Érica, 2004.
- CONVERSE, T.; PARK, J. **PHP a Bíblia**. 2rd ed. Campus / Elsevier, 2003.
- SUEHRING, S. **MySQL a Bíblia**. Campus / Elsevier, 2002.
- CASTRO, E. **HTML, XHTML, and CSS**. 6rd ed. Peachpit Press, 2006.
- GOODMAN, D. **Javascript a Bíblia**. Campus / Elsevier, 2001.
- SOARES, W. **AJAX – Guia Prático para Windows**. Érica, 2006.
- NEGRINO, T.; SMITH, D. **JavaScript and Ajax for the Web**. 7rd ed. Peachpit Press, 2008.
- LIGHSTONE, S.; TEOREY, T.; NADEAYU, T. **Projeto e Modelagem de Banco de Dados**. Elsevier, 2006.
- CORREIA, C.; TAFNER, M. **Análise Orientada a Objetos**. 2rd ed. Visual Books, 2006.
- KILOV, H. **Business Specifications: The Key to Successful Software Engineering**. Prentice Hall, 1998.
- LEDFORD, J. **SEO: Otimização para Mecanismos de Busca - Bíblia**. Alta Books, 2008.
- BOLAOVIP. **Bolão de futebol é aqui**: Melhor referencia de sistema que oferece a possibilidade de criar bolões pessoais para um grande número de competições. Disponível em: <<http://www.bolaovip.com.br/>>. Acesso em: nov. 2009.

ESPN. **Bolão ESPN Brasileiro 2009**: Sistema com boa estrutura que oferece a possibilidade de criar bolões pessoais apenas para os campeonatos nacionais. Disponível em: <<http://espnbolao.terra.com.br/>>. Acesso em: nov. 2009.

VIRTUALBOLAO. **O maior bolão de futebol do Brasil**: Sistema que oferece a possibilidade de participar de bolões apenas para os campeonatos nacionais. Disponível em: <<http://www.virtualbolao.com/>>. Acesso em: nov. 2009.

BOLAOMANIA. **Bolão do futebol brasileiro e mundial inteiramente grátis**: Sistema que oferece a possibilidade de participar de bolões com jogos de campeonatos nacionais e internacionais. Disponível em: <<http://www.bolaomania.com/>>. Acesso em: nov. 2009.