

DOCUMENTO DE REQUISITOS BOLÃO WEB

Versão 2.0

Integrantes	RA
BRUNO HENRIQUE KUNIEDA	61926
GUILHERME GANASSIN	61528
THIAGO BUCALÃO	68962

Sumário

1. Introdução	4
1.1. Objetivo do Documento.....	4
1.2. Escopo do Produto	4
2. Descrição geral do produto.....	4
2.1. Interfaces com Produto	4
2.2. Requisitos Funcionais	5
2.3. Requisitos não Funcionais	8
3. Estudo de Viabilidade	8
3.1. Quanto às contribuições à empresa	8
3.2. Quanto à viabilidade econômica	8
3.3. Quanto à viabilidade técnica	8
3.4. Quanto à legalidade	8
3.5. Quanto às alternativas	9
4. Avaliação dos Riscos	9
4.1. Riscos de Projeto	9
4.2. Risco técnico.....	9
5. Diagramas de Pacotes	10
5.1. Diagrama de Pacotes Genérico	10
5.2. Diagrama de Pacotes com ênfase na atuação de Frameworks	10
6. Arquitetura de Software.....	11
6.1. Diagrama de Arquitetura de Software	11
7. Frameworks	12
7.1. JSF.....	12
7.2. BootsFaces	12
7.3. Hibernate	13
8. Modelo de Casos de Uso	13
8.1. Identificação dos casos de uso	13
8.2. Diagrama de Casos de Uso	14
8.3. Detalhamento dos Casos de Uso	15
9. Diagrama de Classes	28
10. Planejamento das Sprints	28
9.1. Priorização dos Requisitos.....	29

9.2. Primeira Sprint	29
9.3. Segunda Sprint	31

1. INTRODUÇÃO

Este documento especifica os requisitos do **Bolão Web**, um sistema em que os usuários poderão fazer apostas sobre o placar de jogos de futebol dos principais campeonatos de futebol do mundo. Esta documentação irá fornecer aos desenvolvedores as informações necessárias para o projeto e implementação, assim como para a realização dos testes e homologação da aplicação.

O Bolão Web será uma aplicação disponível na rede mundial de computadores (*word wide web*) publicamente, através de um registro de usuário. O seu principal objetivo é angariar visibilidade e acessos ao domínio, isso será possível através do sistema web que simula a prática de bolão, já difundida no meio esportivo principalmente no futebol.

1.1. Objetivo do Documento

Este documento tem como público alvo todos os envolvidos no projeto, desde sua análise até a concepção, com o intuito de guiar a todos em direção a seu objetivo e, conseqüentemente, sua conclusão satisfatória.

1.2. Escopo do Produto

O objetivo do projeto é elevar a audiência de seu domínio na internet, gerando visibilidade e credibilidade à empresa, para alavancar o crescimento da mesma através da obtenção de novos clientes e projetos.

O projeto tem como foco a diversão, e com integração as redes sociais, pretendemos obter engajamento dos usuários, levando a um aumento exponencial dos novos registros e, conseqüentemente, dos acessos.

O objetivo do projeto é a criação de uma plataforma web onde haja interação entre usuários através de um sistema de “Bolão”, onde usuários cadastrados podem sugerir sugestões de resultados de jogos de futebol previamente cadastrados, recebendo assim pontuações por seus acertos. Sua pontuação pode ser vista por um ranking dos melhores apostadores.

2. DESCRIÇÃO GERAL DO PRODUTO

2.1. Interfaces com o Produto

Ator	Descrição
Administrador	Usuário do sistema que deverá moderar e gerenciar os usuários e definir políticas de uso e pontuação das apostas.
Operador	Usuário do sistema que deverá manter os cadastros da aplicação atualizados e atender as solicitações dos usuários apostadores. A principal função do operador será atualizar o sistema com os dados das partidas de futebol disponíveis para aposta.
Apostador	Usuário final do sistema que fará as apostas do bolão e

	receberá pontos de acordo com seus resultados.
Visitante	Usuário final da sistema interessado somente em visualizar informações como, resultado de jogos, situação do ranking, ver campeonatos.

2.2. Requisitos Funcionais

[RF001] Gerenciar Operadores

O sistema deverá permitir incluir, alterar, excluir, e consultar os usuários operadores. O cadastro deve conter um identificador único gerado pelo sistema, nome, e-mail e senha.

Prioridade:	X	Essencial		Importante		Desejável
Ator(es):	<i>Administrador</i>					
Requisitos associados:						

[RF002] Gerenciar Campeonatos

O sistema deverá permitir incluir, alterar, excluir, e consultar os campeonatos. O cadastro deve conter um identificador único gerado pelo sistema, nome, escopo (regional, estadual, nacional, continental ou mundial), data de início, data de fim, tipo (pontos ou mata-mata) e observações.

Prioridade:	X	Essencial		Importante		Desejável
Ator(es):	<i>Administrador, Operador</i>					
Requisitos associados:						

[RF003] Gerenciar Times

O sistema deverá permitir incluir, alterar, excluir, e consultar os times participantes de cada campeonato ou torneio. O cadastro deve conter um identificador único gerado pelo sistema, tipo (time ou seleção), nome e local de origem. O local de origem depende do campo tipo: Se for time, considera-se um campo composto por cidade, estado e país. Se for uma seleção, considera-se apenas o país.

Prioridade:	X	Essencial		Importante		Desejável
--------------------	----------	-----------	--	------------	--	-----------

Ator(es):	<i>Administrador, Operador</i>
Requisitos associados:	RF010

[RF004] Gerenciar Partidas

O sistema deverá permitir incluir, alterar, excluir, e consultar as partidas. O cadastro deve conter um identificador único gerado pelo sistema, o identificador de ambos os times, a data e hora da partida e a quantidade de gols marcados por cada time.

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	<i>Administrador, Operador</i>					
Requisitos associados:	<i>[RF004]</i>					

[RF005] gerenciar apostadores

Através do administrador ou operador responsável, o sistema deve permitir o cadastro de usuários apostadores. O cadastro deve conter um identificador único gerado pelo sistema, nome, sobrenome, apelido, e-mail, data de nascimento, senha e foto.

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	<i>Administrador e Operador</i>					

[RF006] Gerenciar Palpite

O sistema deverá permitir o apostador incluir, alterar, excluir, e consultar seus palpites de resultado para cada jogo. A alteração e exclusão podem ser feitas até o momento de início do jogo.

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	<i>Apostador</i>					
Requisitos associados:						

[RF007] Processar resultado e distribuir pontos

Ao fim de cada partida, os usuários operadores serão responsáveis em registrar o resultado do jogo, e ao salvar o sistema deverá processar esse resultado e distribuir os pontos aos apostadores que deram um palpite em determinada partida. O apostador recebe dois pontos caso acerte o placar exato e um ponto caso acerte o time vencedor. O Administrador também possui autonomia de realização desta tarefa.

Prioridade:	X	Essencial		Importante		Desejável
Ator(es):	<i>Administrador, Operador</i>					
Requisitos associados:						

[RF008] Ranking de apostadores

O sistema deverá disponibilizar de forma pública uma página com o ranking dos apostadores classificado em ordem da quantidade de pontos de forma decrescente. A página deve conter a classificação, nome e quantidade de pontos de cada apostador. Caso o apostador esteja logado, a posição dele deve estar destacada.

Prioridade:	X	Essencial		Importante		Desejável
Ator(es):	<i>Administrador, Visitante</i>					
Requisitos associados:						

[RF009] Consulta fácil de palpites

O sistema deverá possuir uma tela inicial para o apostador constando seus palpites, os resultados dos jogos já realizados, dados de registro, posição no ranking e quantidade de pontos.

Prioridade:		Essencial	X	Importante		Desejável
Ator(es):	<i>Apostador</i>					
Requisitos associados:	RF008					

[RF010] Gerenciar Local Origem

Para que seja possível ao sistema cadastrar futuramente times antes é necessário que seja cadastrado da local origem desse time. Essa função fica a cargo do Operador.

Prioridade:	X	Essencial		Importante		Desejável
Ator(es):						
Requisitos associados:						

2.3. Requisitos Não Funcionais

[NF001] Plataforma Web

O sistema deverá ser implantado em plataforma web, para que seja possível acesso de qualquer lugar, sem a necessidade de instalação do software e também através de dispositivos móveis.

Prioridade:	X	Essencial		Importante		Desejável
Ator(es):						

3. ESTUDO DE VIABILIDADE

3.1. Quanto às contribuições à empresa

O projeto visa, para o cliente, gerar visibilidade ao seu domínio, um crescente em termos de usuários e credibilidade perante aos mesmos. Visando também garantir futuros novos clientes e projetos. Dessa forma, o usuário final poderá interagir e, muitas vezes, se informar sobre seu o futebol.

3.2. Quanto à viabilidade econômica

O projeto acaba por ter um custo relativamente baixo, pois é feito através de um sistema web, deixando de lado problemas mais custosos e físicos. Ressaltando que mesmo levando-se em consideração este motivo ainda assim, há custos tais como: recursos humanos, manutenção.

3.3. Quanto à viabilidade técnica

A tecnologia atual oferece uma gama de recursos que podem contribuir com a construção do projeto.

Em relação a alguma restrição de custo de processamento, cremos que não haverá maiores problemas, no nosso entendimento os requisitos para o uso do sistema, em termos de hardware e estrutura, estão dentro da faixa de preço e processamento que o mercado brasileiro oferece - necessitando apenas acesso à internet e um navegador web.

Além disso, o prazo a se considerar, pois, como campeonatos e demais torneios geralmente são eventos com data estabelecida, existe um prazo válido para a entrega do projeto. Mas considerando as necessidades e o tempo hábil, cremos que o tempo é suficiente.

3.4. Quanto à legalidade

O projeto é baseado em um sistema de palpites, que não visa lucro direto ao usuário, mas também o usuário não terá prejuízos, devido ao fato em que não será permitido palpites de resultados através de apostas de dinheiro. Tratando-se assim, apenas de uma opção de descontração e informação.

Em nenhum momento foi constatada alguma infração ou ilegalidade em tempo de desenvolvimento do projeto.

3.5. Quanto às alternativas

O desenvolvimento web nos oferece muitas possibilidades, dentre a que mais se destacou durante o estudo foi a linguagem JAVA, pois ela nos oferece codificação no servidor, um ponto importante para o desenvolvimento. Revisados todos os nossos fatores que colocamos em consideração, chegamos a decisão em usar esta tecnologia, em um SGBD MySQL. A escolha foi devido à familiaridade entre os desenvolvedores do projeto, por não ser um projeto de grande porte, oferecer mais rapidez em seu desenvolvimento, entre outros fatores menos específicos.

4. AVALIAÇÃO DOS RISCOS

4.1. Riscos de Projeto

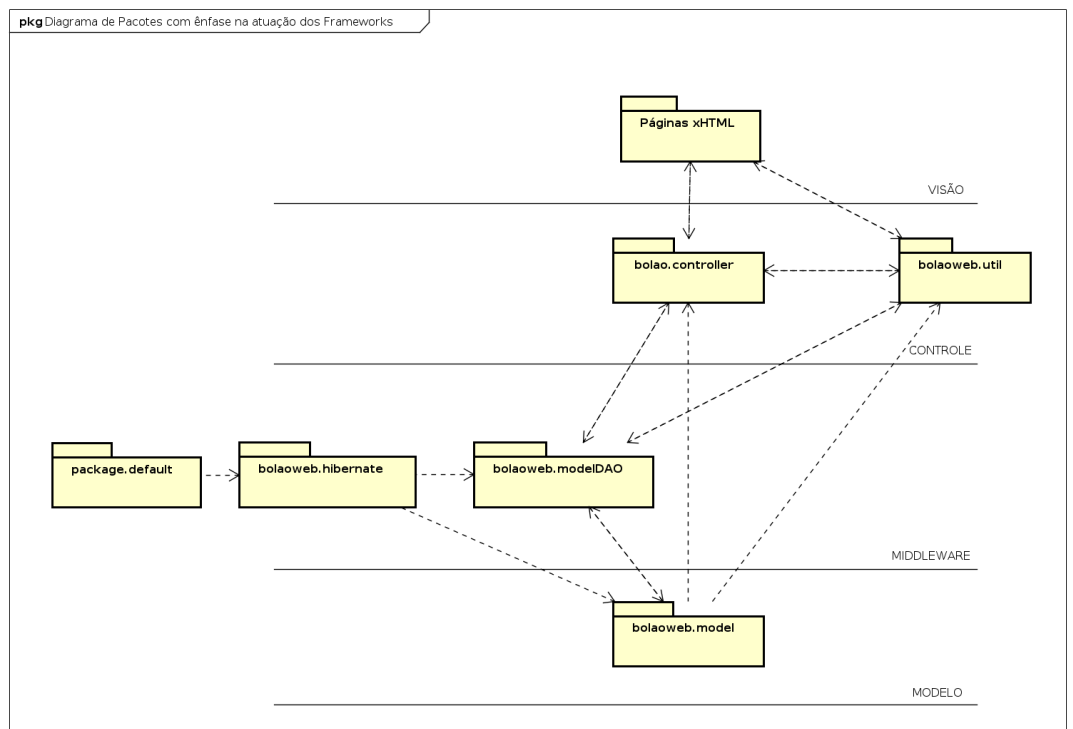
- Falta de tempo, uma vez que somos todos universitários e trabalharemos no contra turno.
- Como os nossos usuários são pessoas fictícias, não há tempo hábil para se fazer um estudo mais aproximado do que irá agradá-los, a falta desse estudo poderá gerar mudanças no escopo do projeto.

4.2. Risco técnico

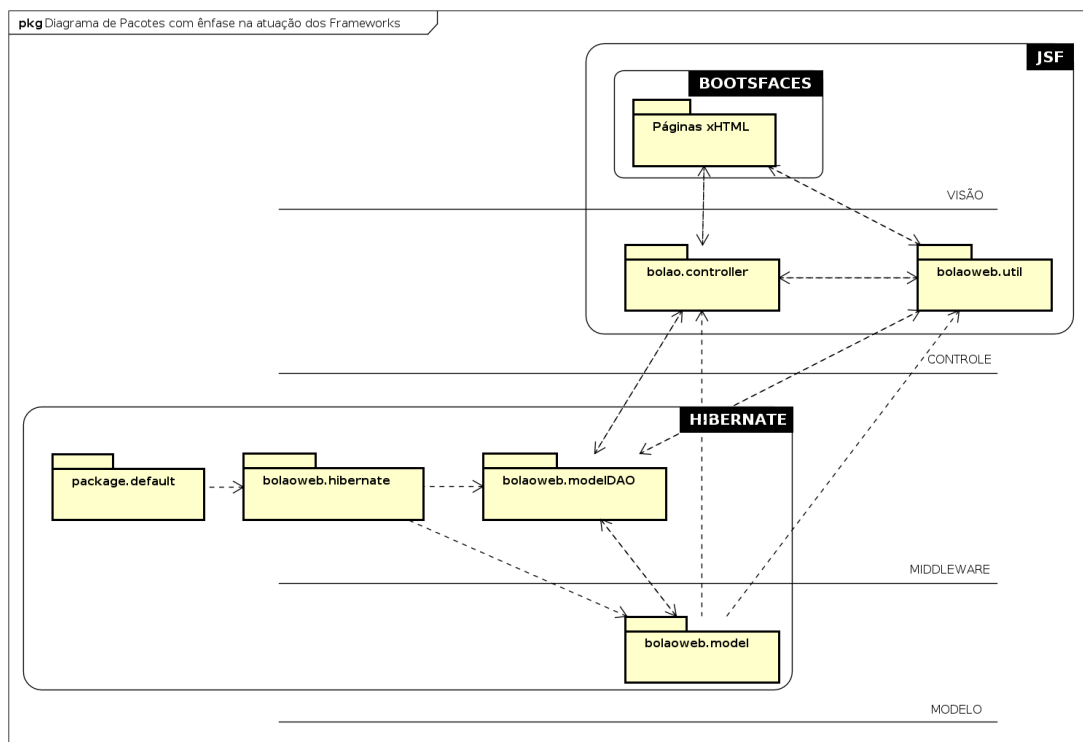
- Falta de informação, caso aconteça de não conseguirmos informações com tempo suficiente para cadastrar os jogos no Bolão WEB e os apostadores fazerem suas apostas.

5. DIAGRAMA DE PACOTES

5.1. Diagrama de Pacotes Genérico



5.2. Diagrama de Pacotes com ênfase na atuação de Frameworks



6. ARQUITETURA DE SOFTWARE

Definimos que o tipo de arquitetura de sistema que utilizaremos para o desenvolvimento do sistema web é o MVC. Onde especificamos que:

Camada de Visão: Será a camada responsável com a interação com o usuário, a partir de um navegador. Responsável pela entrada de dados, requisitar os serviços ao controle e apresentar os resultados;

Camada de Controle: Será a responsável em definir e realizar as chamadas de método dentro da aplicação, fazer o controle dos dados recebidos da camada de apresentação e se comunicar com as classes de persistência quando solicitado algum dado presente nesta.

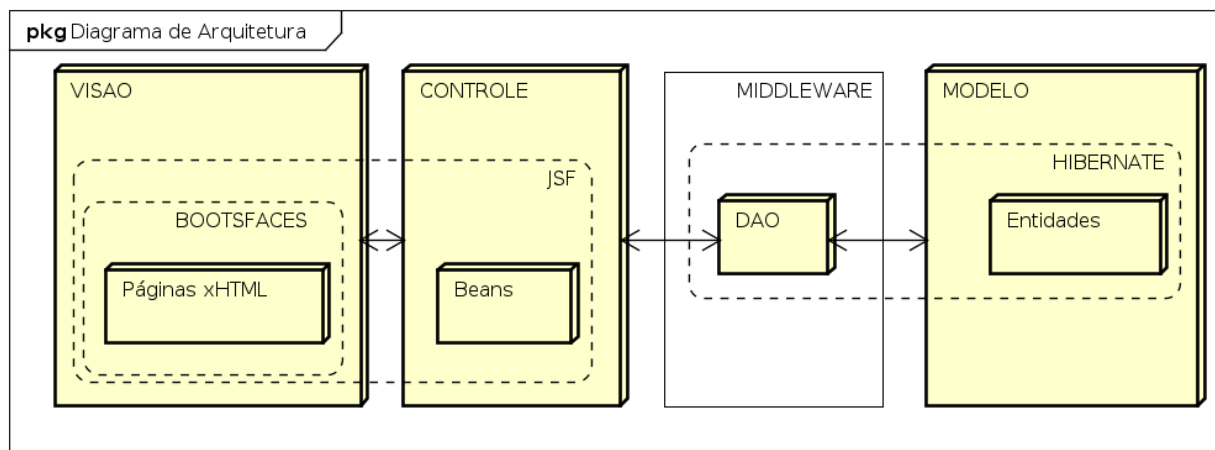
Camada de Modelo: Será a responsável em armazenar as regras de negócio e os dados recebidos da camada de aplicação.

No diagrama a seguir, consideramos também uma subcamada de Middleware, localizada entre as camadas de Controle e Modelo. A função principal desta subcamada é portar as classes DAO, que são responsáveis por intermediar o acesso do controle as entidades do Modelo. Esta subcamada de Middleware também é responsável por prover suporte e configuração das classes de persistência.

6.1. Diagrama de Arquitetura de Software

O padrão arquitetural Model-View-Controller (MVC) é uma forma de quebrar uma aplicação, ou até mesmo um pedaço da interface de uma aplicação, em três partes: o modelo, a visão e o controlador.

No padrão MVC, teríamos então o mapeamento de cada uma dessas três partes para o padrão MVC. Optamos utilizar esse padrão de projeto pois através da sua separação facilita manipular os objetos além de facilitar quando está sendo implementado.



7. FRAMEWORKS

7.1. JSF

O *JavaServer Faces* (JSF) é uma tecnologia que nos permite criar aplicações Java para Web utilizando componentes visuais pré-prontos, de forma que o desenvolvedor não se preocupe com o *Javascript* e HTML, o que define o JSF como um framework. Basta adicionar os componentes que forem necessários ao projeto e eles serão renderizados e exibidos em formato HTML.

Com a característica *stateful* podemos criar formulários de várias páginas e navegar nos vários passos dos componentes com o estado das telas sendo mantidos automaticamente.

A separação de camadas de apresentação e de aplicação também é uma característica muito marcante da arquitetura do JSF. Pensando no modelo MVC, o JSF possui uma camada de visualização bem separada do conjunto de classes de modelo.

O JSF ainda tem a vantagem de ser uma especificação do Java EE, isto é, todo servidor de aplicações Java tem que vir com uma implementação dela e há diversas outras disponíveis.

Em nosso projeto, o JSF será usado na construção de toda a interface gráfica de interação com o usuário, a utilização deste framework visa a maior eficiência nesta fase da implementação e o principal objetivo é a diminuição do tempo gasto na implementação da interface gráfica.

O JSF irá fazer o mapeamento das páginas xhtml, recebendo as ações pretendidas pelo usuário do sistema, enviando as mesmas para as classes BEANS (responsável pela integração com as classes de persistência) também mapeadas pelo mesmo.

7.2. BootsFaces

O *BootsFaces* é um framework para desenvolvimento de estilos de páginas HTML que surge como uma opção do famoso *Bootstrap* para desenvolvimento *JavaWeb* com suporte JSP.

O *Bootstrap* por si só, já traz como características diversas vantagens, tais como, vasta documentação, otimização para layouts responsivos e a praticidade para o desenvolvimento de layouts padronizados. Com a popularidade do *Bootstrap*, o *BootsFaces* foi criado para ser uma poderosa ferramenta JSF que tras o melhor do *Bootstrap* e que tem foco no desenvolvimento rápido e fácil de aplicações Java Web.

Necessariamente, o *BootsFaces* se concentra na estrutura de layout da página, permitindo a possibilidade de implementar sistemas bem desenhados com pouco esforço.

Outra característica importante do *BootsFaces* que influenciou diretamente na escolha do mesmo para nosso projeto é o fato de ser bem leve e consumir pouco espaço. Comparado com outros frameworks JSF, o *BootsFaces* pode ser até 8 vezes menor.

Infelizmente, por se tratar de uma tecnologia nova, não há disponibilidade de temas previamente prontos (como no caso do *PrimeFaces*) e a customização dos componentes do *template* padrão disponibilizado pelo *BootsFaces* não é possível de ser realizada devido a utilização do Maven, ou seja, sempre que a dependência é atualizada, as alterações dos componentes é sobrescrita pela sua versão padronizada.

7.3. Hibernate

O *Hibernate* é um *framework* para o mapeamento objeto-relacional. Este framework facilita o mapeamento dos atributos entre uma base tradicional de dados relacionais e o modelo objeto de uma aplicação, mediante o uso de arquivos (XML) ou anotações.

O objetivo do *Hibernate* é diminuir a complexidade entre os programas Java, baseado no modelo orientado a objeto, que precisam trabalhar com um banco de dados do modelo relacional. Em especial, no desenvolvimento de consultas e atualizações dos dados.

Sua principal característica é a transformação das classes em Java para tabelas de dados (e dos tipos de dados Java para os da SQL). O *Hibernate* gera as chamadas SQL e libera o desenvolvedor do trabalho manual da conversão dos dados resultante, mantendo o programa portátil para quaisquer bancos de dados SQL, porém causando um pequeno aumento no tempo de execução.

Em nosso sistema o *Hibernate* será utilizado para gerenciar o acesso às informações contidas no banco de dados composto pelas classes de modelo.

O *Hibernate*, conforme o padrão de projeto *Singleton*, fará a instanciação única das classes, gerando um “canal” de acesso ao banco de dados.

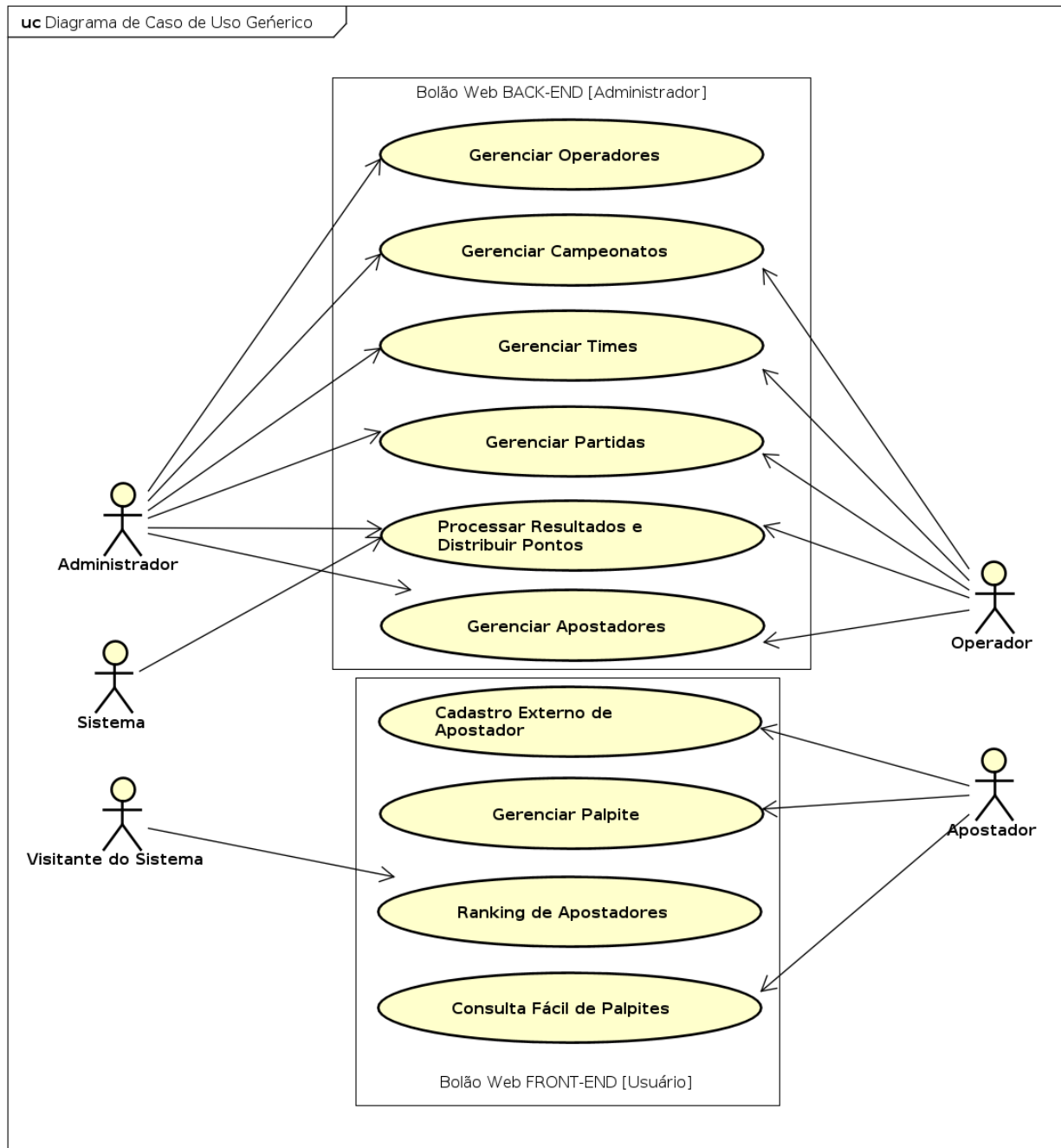
8. MODELO DE CASOS DE USO

8.1. Identificação dos casos de uso

Casos de uso de forma prioritária:

- [RF001] Gerenciar Operadores – Guilherme
- [RF002] Gerenciar Campeonatos - Bruno
- [RF003] Gerenciar Times - Bruno
- [RF004] Gerenciar Partidas - Bruno
- [RF005] Gerenciar Apostador - Thiago
- [RF006] Gerenciar Palpite - Thiago
- [RF007] Processar resultado e distribuir pontos - Thiago
- [RF008] Ranking de apostadores - Guilherme
- [RF009] Consulta fácil de palpites – Guilherme
- [RF010] Gerenciar Local Origem – Bruno

8.2. Diagrama de Casos de Uso



Obs.: O diagrama de caso de uso acima representa todos os casos de uso do sistema relacionados com seus respectivos atores. Nos detalhamentos a seguir, foi implementado um diagrama para cada caso de uso representando seus detalhes específicos.

8.3. Detalhamento dos Casos de Uso

[RF001] USE CASE<gerenciar operadores> - Guilherme

Objetivo:

Permite o gerenciamento e manipulação de usuários responsáveis pela alimentação das informações operacionais do sistema. Estas informações operacionais são pertinentes ao lançamento de resultados, gerenciamento de times, gerenciamento de partidas, gerenciamento de campeonatos e processamento de apostas.

Pré-condições:

O usuário deve estar logado no sistema e possuir permissão de administrador.

Iniciado por:

Qualquer usuário administrador no intuito de efetuar uma inclusão, alteração ou consulta por usuários operacionais.

Fluxo principal:

1. O usuário administrador acessa a opção de controle de usuários operacionais;
2. O sistema mostra a tela para escolha da rotina que será;
3. O usuário escolhe a opção para inclusão/alteração de novo usuário operacional;
4. O sistema mostra o formulário de cadastro/alteração;
5. O usuário informa os dados de inclusão de novo usuário, com atenção especial aos atributos obrigatórios;
6. O sistema valida a corretude dos atributos, e verifica o preenchimento dos atributos obrigatórios;
7. O sistema grava as informações na devida tabela do BD;
8. O sistema envia uma mensagem para a tela do usuário informando o sucesso da operação e atualiza o usuário como ativo.

Fluxo alternativo 1: Dados pessoais incorretos

No item 6, caso as informações não corresponderem aos padrões de validação o sistema informa através de mensagem de erro a ocorrência e indica quais campos precisam ser corrigidos. O fluxo principal é retomado com as informações preenchidas corretamente.

Fluxo alternativo 2: Problema com o banco de dados

No item 7, caso o banco de dados apresente algum problema ou as informações de persistência não forem definidas corretamente, o sistema não grava os dados e exibe uma mensagem de erro na tela indicando o problema.

Fluxo alternativo 3: Exceder tempo

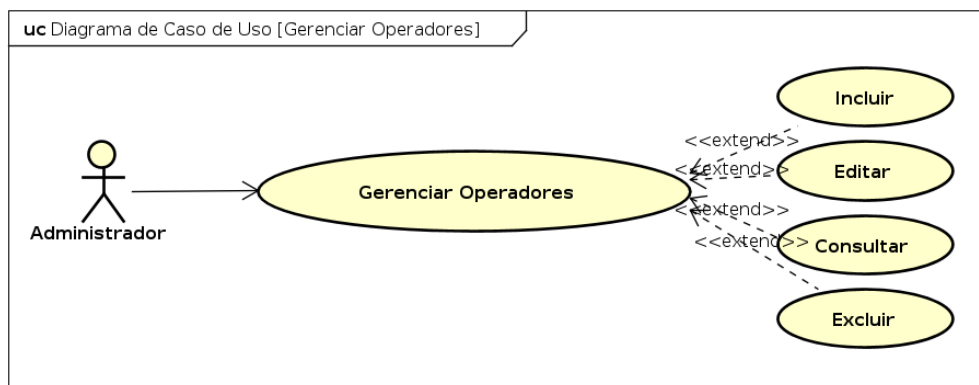
No item 5, caso o usuário não confirme os dados e fique mais de 30 minutos inativo, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Fluxo alternativo 4: Cancelar cadastro.

Em qualquer momento, o usuário pode cancelar, e o sistema retornando ao início no Menu Principal.

Pós-condições:

Um usuário administrador deve ser criado e estar devidamente ativo, ou o processo é abortado na ocorrência do Fluxo alternativo 2.



[RF002] USE CASE<gerenciar campeonatos> - Bruno

Objetivo:

O sistema deverá permitir manipular os dados referente aos campeonatos através da página do Bolão. O cadastro deve conter um identificador único gerado pelo sistema, nome, escopo (regional, estadual, nacional, continental ou mundial), data de início, data de fim, tipo (pontos ou mata mata) e observações.

Pré-Condições:

O cadastramento e demais gerenciamento de Campeonato é uma atividade exclusiva para usuários Administradores ou Operadores.

Iniciado por:

O Administrador ou o Operador, que estiver designado a cadastrar os campeonatos, fará o cadastro manualmente através do site.

Fluxo principal:

1. O usuário seleciona a opção de cadastrar/alteração campeonato
2. O sistema apresenta um formulário de cadastro/alteração
3. O usuário preenche o formulário de campeonato e submete o formulário
4. O sistema valida os dados cadastrais
5. O sistema reapresenta as informações na tela e pede confirmação
6. O usuário confirma os dados
7. O sistema efetua a inclusão dos dados informados no cadastro de campeonato
8. O sistema agradece o cadastramento e convida o usuário a cadastrar um novo campeonato.

Fluxo alternativo 1: Corrigir dados cadastrais

No item 4, o sistema ao validar os dados cadastrais, detecta que alguns dados considerados como obrigatórios não foram preenchidos ou que dados foram preenchidos incorretamente.

5a. O sistema reapresenta os dados cadastrais indicando o(s) erro(s)

5b O usuário corrige o(s) erro(s) cadastrais

Retorna ao fluxo principal no item 6

Fluxo alternativo 2: Exceder tempo

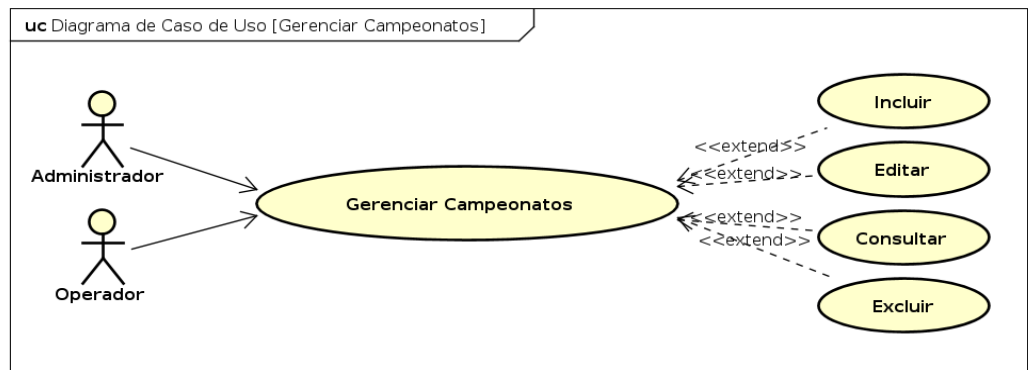
No item 6, caso o usuário não confirme os dados e fique mais de 30 minutos inativo, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Fluxo alternativo 3: Cancelar cadastro.

Em qualquer momento, o usuário pode cancelar, e o sistema retornando ao início no Menu Principal.

Pós-condições:

O campeonato foi cadastrado no sistema ou o sistema exibiu a sua página inicial, por cancelamento da operação.



[RF003] USE CASE<gerenciar times> - Bruno

Objetivo:

Permitir ao usuário o gerenciamento e manipulação das informações das equipes no sistema.

Pré-Condições:

Usuário logado no sistema e com permissões ao nível operacional ou de administrador.

Iniciado por:

Usuário com nível de permissão devidamente logado no sistema, com a intenção de incluir, alterar, excluir ou consultar as equipes no sistema.

Fluxo principal:

1. O usuário seleciona a opção de gerenciamento de times;
2. O sistema retorna ao usuário uma tela de formulário para inserção/alteração;
3. O usuário informa que tipo de equipe será manipulada, se clube ou federação (país);
4. O usuário informa o nome, dados de origem da equipe e seleciona a opção para gravação no sistema;
5. O sistema gera o identificador do time inscrito, caso inclusão, ou gera "decremento" do identificador, caso exclusão;
6. O sistema salva as informações no banco de dados;
7. O sistema exibe uma mensagem de confirmação da operação.

Fluxo alternativo 1: Cancelar cadastro.

Em qualquer momento, o usuário pode cancelar o cadastro da equipe, com o sistema retornando ao início do formulário com os campos vazios.

Fluxo alternativo 2: Exceder tempo

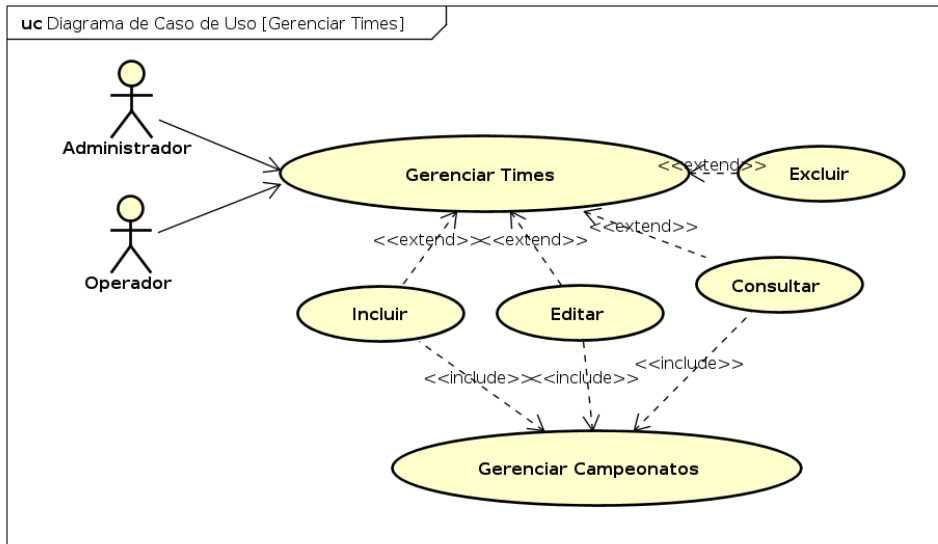
No item 3 ou 4, caso o usuário não confirme os dados e fique mais de 30 minutos inativo, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Fluxo alternativo 3: Dados da equipe

No item 6, o sistema ao validar encontra uma inconsistência, sendo um dado inválido, time já inserido ou a falta de um campo requerido.

Pós-condições:

Um time foi criado, excluído, alterado ou consultado caso o usuário não tenha



abortado o processo.

[RF004] USE CASE<gerenciar partidas> - Bruno

Objetivo:

Permitir ao usuário administrador ou operador toda a manipulação de informações relativas as partidas.

Pré-condições:

O usuário deve estar logado no sistema, deve existir ao menos um campeonato cadastrado e ao menos dois times cadastrados.

Iniciado por:

Usuário administrador ou operador com a intenção de incluir, alterar, excluir ou consultar as partidas.

Fluxo principal:

1. O usuário acessa a opção de gestão de partidas.
2. O sistema exibe o formulário de cadastro.
3. O usuário seleciona o campeonato na qual a partida faz parte, o time da casa e o time visitante, a data e hora da partida.
4. O sistema gera o identificador da partida.
5. O sistema salva as informações no banco de dados.

6. O sistema exibe uma mensagem informando o sucesso da operação.

Fluxo alternativo 1: Exceder tempo

No item 3, caso o usuário não confirme os dados e fique mais de 30 minutos inativo, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Fluxo alternativo 2: Cancelar cadastro.

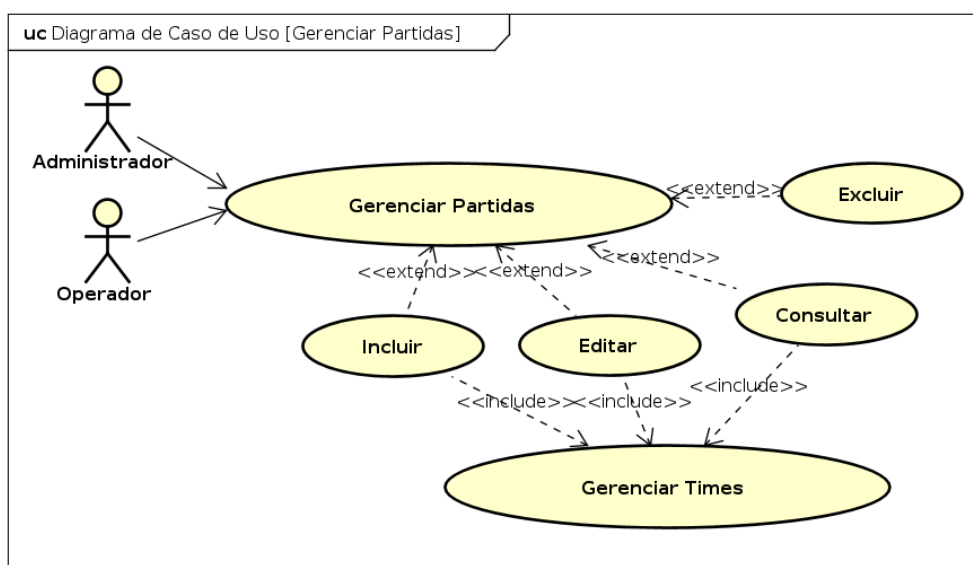
Em qualquer momento, o usuário pode cancelar, e o sistema retornando ao início no Menu Principal.

Fluxo alternativo 3: Data e hora da partida são inferiores a atual

Antes do item 4, caso o sistema identifique que a data e hora da partida são menor que a atual (partida já ocorreu), o sistema deve obrigar o usuário a informar o placar do hoje antes de salvar o registro.

Pós-condições:

Uma partida foi criada, excluída, alterada ou consultada caso o usuário não tenha abortado o processo.



[RF005] USE CASE<gerenciar apostador> - Thiago

Objetivo:

O sistema deverá permitir os processos de cadastro e gerenciamento do usuário apostador.

Pré-Condições:

O usuário deve estar logado.

Iniciado Por:

Pelo Administrador ou Operador, quando necessário, através do sistema administrador do Bolão.

Fluxo principal:

1. O usuário seleciona a opção de gerenciar cadastro de apostador;
2. O sistema apresenta o formulário para preenchimento dos dados;
3. O usuário incluir/altera os dados;
4. O sistema valida os dados do formulário;
5. O sistema reapresenta os dados ao usuário e pede confirmação
6. O usuário confirma a operação
7. O sistema grava no banco de dados as informações;
8. O sistema notifica o sucesso da operação e volta à página inicial do sistema.

Fluxo alternativo 1: Corrigir dados

No item 4, o sistema ao validar os dados, detecta que alguns dados considerados como obrigatórios não foram preenchidos ou que dados foram preenchidos incorretamente.

5a. O sistema reapresenta os dados indicando o(s) erro(s)

5b O usuário corrige o(s) erro(s)

Retorna ao fluxo principal no item 6

Fluxo alternativo 2: Exceder tempo

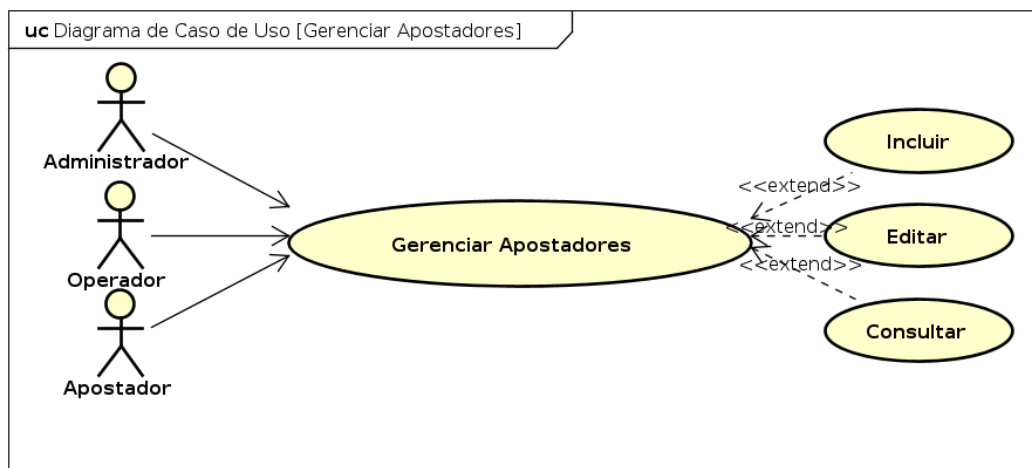
No item 3, caso o usuário não confirme os dados e fique mais de 30 minutos inativo, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Fluxo alternativo 3: Cancelar cadastro.

Em qualquer momento, o usuário pode cancelar, e o sistema retornando ao início no Menu Principal.

Pós-condições:

Os dados do usuário foram alterados no sistema ou o sistema exibiu a sua página inicial, por cancelamento da operação.



[RF007] USE CASE<gerenciar palpites> - Thiago

Objetivo:

Permitir que o usuário tente adivinhar o placar dos jogos que estiverem disponíveis no site do Bolão.

Pré-Condições:

O usuário deve estar logado no sistema.

Iniciado por:

De acordo com cada partida cadastrada pelo Administrador no sistema, o usuário poderá definir seu palpite até um dia antes da rodada corrente.

Fluxo principal:

1. O usuário acessa a página inicial do site do Bolão;
2. <<include>>Autenticar usuário;
3. O usuário seleciona a opção no sistema de fazer o palpite;
4. O usuário seleciona a partida na qual deseja cadastrar um palpite;
5. O usuário informa o palpite da partida selecionada;
6. O sistema mostra as informações dos dados inseridos e das partidas selecionadas;
7. O usuário confirma os dados passados ao sistema;
8. O sistema realiza a inclusão dos dados informados no cadastro de palpite na partida desejada;
9. O sistema confirma o palpite e convida o usuário a fazer um novo ou retornar a página inicial do site.

Fluxo alternativo 1: Corrigir dados do palpite

No item 7, o sistema ao validar os dados do palpite, detecta que alguns dados considerados como obrigatórios não foram preenchidos ou que dados foram preenchidos incorretamente.

5a. O sistema reapresenta os dados do palpite indicando o(s) erro(s);

5b. O usuário corrige o(s) erro(s) do palpite;

Retorna ao fluxo principal no item 6.

Fluxo alternativo 2: Cancelar operação

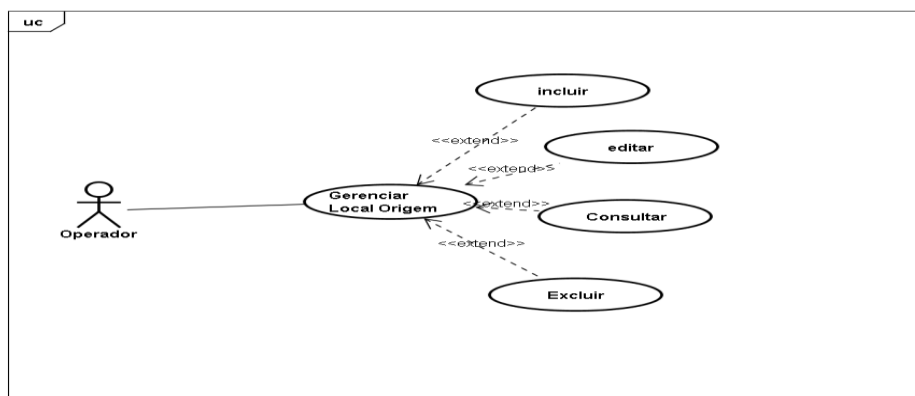
Nos itens 3 ou 4 ou 5 ou 7, o usuário cancela a operação. O sistema retorna à página inicial do sistema.

Fluxo alternativo 3: Exceder tempo

No item 7, caso o usuário não confirme os dados e fique mais de 30 minutos inativo, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Pós-condições:

O palpite do usuário foi cadastrado no sistema ou o sistema exibiu a sua página inicial, por cancelamento da operação.



[RF008] USE CASE<processar resultados e distribuir pontos> - Thiago

Objetivo:

Ao fim de uma partida, os operadores são alertados e devem cadastrar o placar e o sistema deve percorrer todos os palpites da respectiva partida distribuindo aos apostadores dois pontos caso acerte o placar exato e um ponto caso acerte o time vencedor.

Pré-condições:

Existir uma partida com pelo menos um palpite e com data e hora inferior a atual.

Iniciado por:

Processo do sistema que verifica constantemente se existe alguma partida já ocorrida.

Fluxo principal:

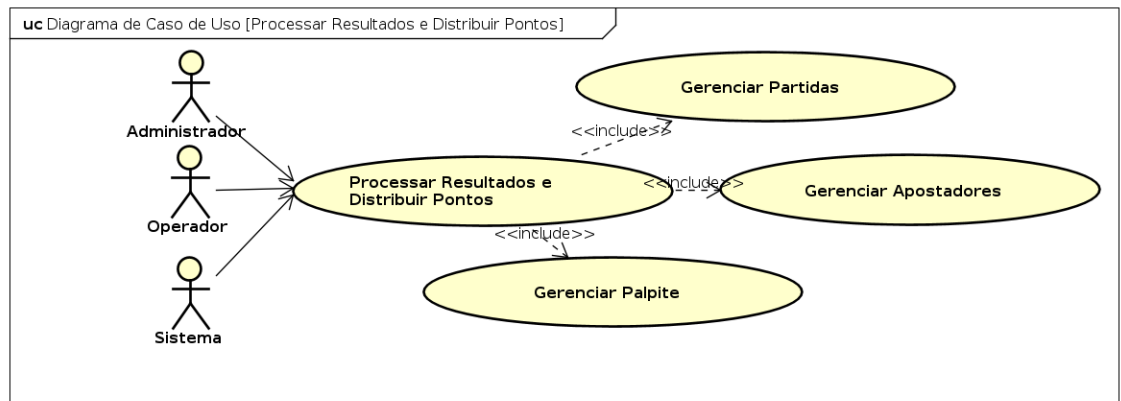
1. O sistema identifica uma partida com data e hora inferior a atual.
2. O sistema notifica a todos os operadores por e-mail com um link direto para a partida.
3. O usuário recebe o e-mail e acessa o link.
4. O usuário informa o placar do jogo e seleciona a opção para salvar os dados.
5. O sistema atualiza a partida no banco de dados.
6. O sistema exibe uma mensagem informando o sucesso da operação.
7. O sistema carrega todos os palpites relacionados a essa partida.
8. O sistema, para cada palpite, soma aos pontos do apostador dois pontos caso acerte o placar exato e um ponto caso acerte o time vencedor e envia um e-mail ao apostador informando de seus resultados.

Fluxo alternativo: Cadastro do placar já efetuado

Após o item 3, caso a partida já possua cadastro de placar, é exibida uma mensagem informando o usuário e solicitando que ele valide e confirme o placar. Após isso, o processo é finalizado.

Pós-condições:

A partida já realizada fica com o placar cadastrado e os apostadores que deram palpites nela, recebem os seus pontos.



[RF009] USE CASE<ranking de apostadores> - Guilherme

Objetivo:

O sistema deverá permitir consultar o ranking dos Apostadores através da página do Bolão. A consulta é permitida a qualquer visitante do site. No ranking

deve constar classificação, nome e pontos de cada apostador em ordem decrescente da classificação.

Pré-Condições:

Nenhuma pré-condição, o usuário simplesmente deve ir ao sistema e realizar a consulta.

Iniciado por:

Qualquer visitante ou usuário do site.

Fluxo principal:

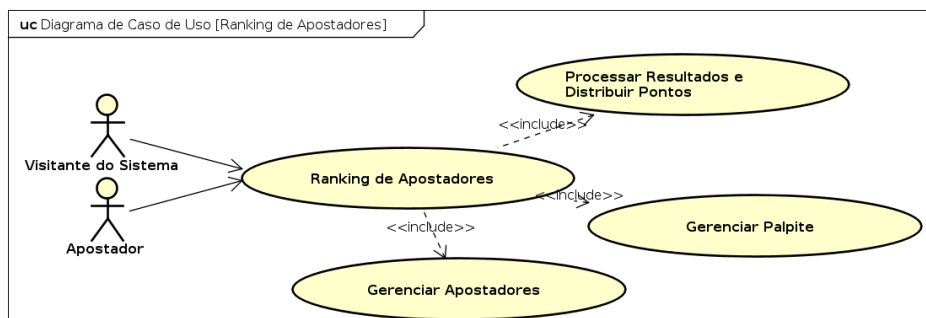
1. O usuário acessa a página iniciado do sistema, o site do Bolão.
2. O usuário seleciona a opção de consultar ranking de apostadores
3. O sistema apresenta o ranking

Fluxo alternativo: Mostrar posição

No item 3, caso o usuário logado seja um apostador, a posição dele deverá estar em destaque.

Pós-condições:

O ranking foi exibido na tela.



[RF010] USE CASE<consulta fácil de palpites> - Guilherme

Objetivo:

Permitir ao usuário realizar consultados palpites já inseridos no sistema, assim como as partidas já realizadas da rodada, posição no ranking e a quantidade de pontos.

Pré-Condições:

Usuário logado no sistema e com permissões ao nível de apostador.

Iniciado por:

Usuário com nível de permissão devidamente logado no sistema.

Fluxo principal:

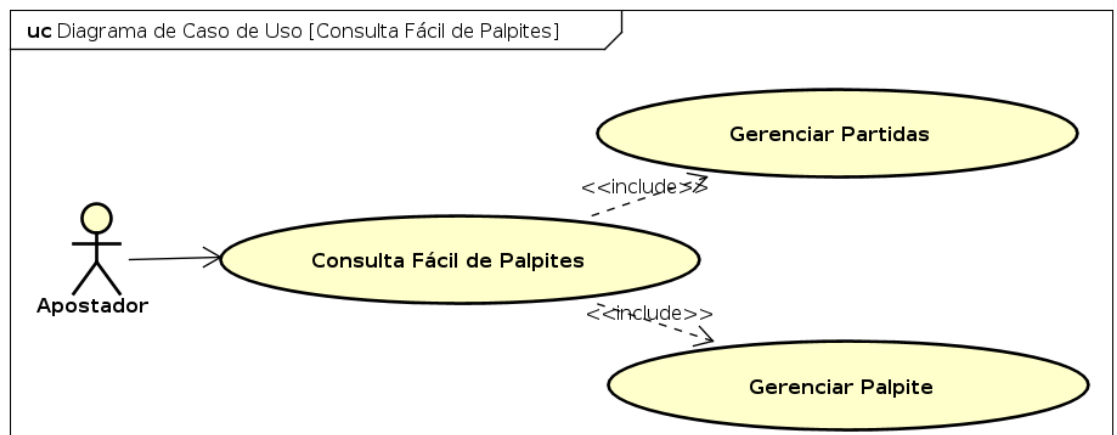
1. O sistema exibe a tela inicial do site;
2. O usuário seleciona a consulta fácil de palpites;
3. O sistema retorna uma página onde constam as informações de palpites já realizados, pontuação e a posição no ranking geral;
4. O usuário confere os dados e seleciona voltar, retornando a página inicial do sistema.

Fluxo alternativo: Exceder tempo

No item 7, caso o usuário não retorne nenhuma ação dentro de 10 minutos, a operação é cancelada pelo sistema. Nesse caso o sistema retorna à sua página inicial.

Pós-condições:

O sistema retorna a página inicial do sistema após o usuário conferir os dados ou exceder o tempo limite.



[RF011] USE CASE<Gerenciar Local Origem> - Bruno

Objetivo:

Para que os cadastros dos times sejam realizados com sucesso é necessário que haja um cadastro dos estados ou país a qual este time pertence. Essas informações podem ser acessadas pelo operador na hora de cadastro do time.

Pré-condições:

Os campeonatos deverão estar encerrados.

Iniciado por:

Operador realiza o cadastro, alteração ou exclusão de locais origem.

Fluxo principal:

1. Operador solicita um novo cadastro de local origem.
2. Operador informa o estado e país.
3. O sistema aceita o cadastro desse estado e país.

Fluxo alternativo: Local Origem já cadastrado Anteriormente.

1. Sistema verifica se local origem informado já consta no Banco de Dados.
2. Operador informa o novo local origem ou retorna a outra atividade do sistema.

Pós-condições:

Ao final encontra-se cadastrado no Banco de Dados do sistema todos os possíveis locais de origem.

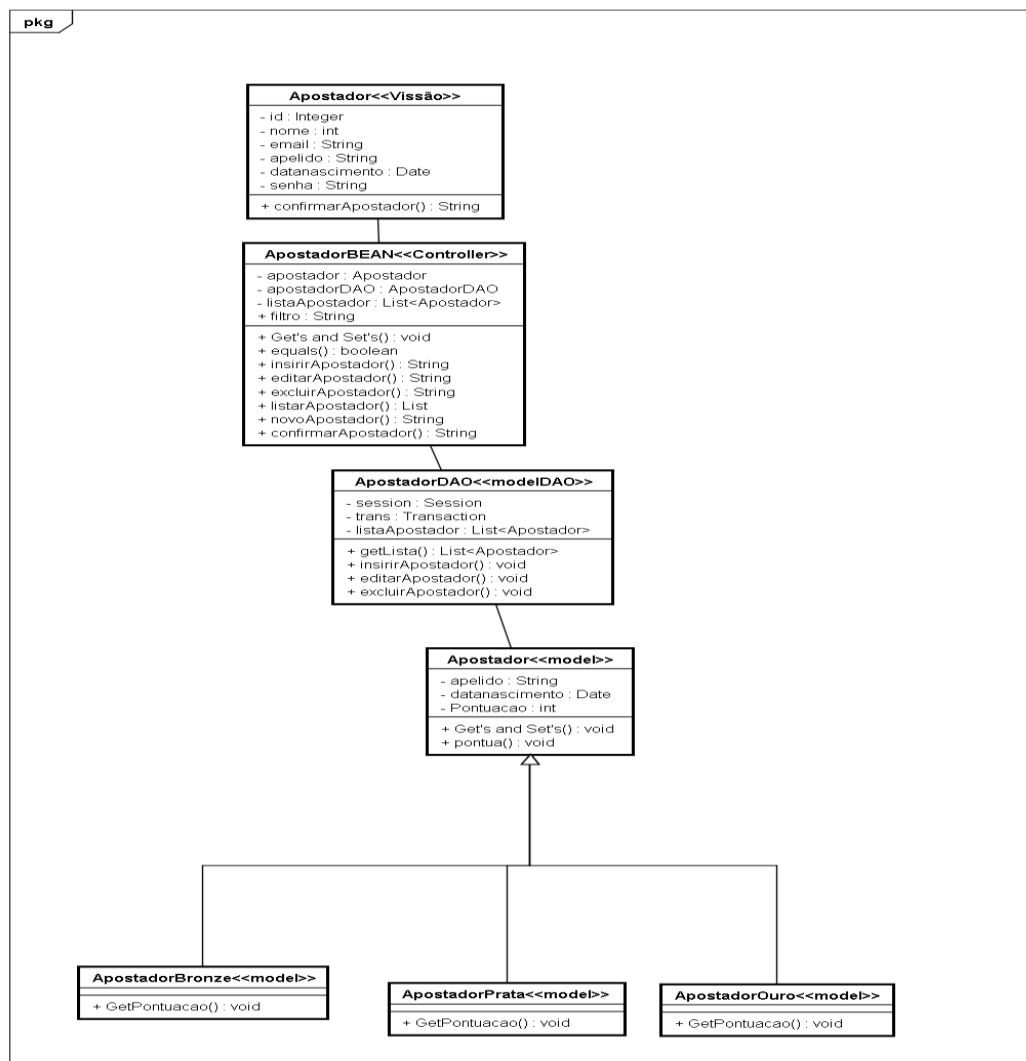
9. Padrão de Projeto:

Padrão de projeto: STATE

É um padrão de projeto de software usado quando um componente de um objeto, dependendo do seu estado. Ou seja, defini uma interface para encapsular o comportamento com um estado particular do contexto.

A vantagem de se utilizar o padrão de projeto *state* é devido a fácil localização das responsabilidades de estados específicos, devido a que os encontram nas classes que correspondem a cada estado. Isso proporciona maior clareza no desenvolvimento, e na manutenção posteriormente. Pelo fato que diferentes estados são representados por um único atributo (estado) e não envolvido em diferentes variáveis e grandes condicionais.

A motivação encontrada em nosso sistema para a implementação desse padrão de projeto. E de encapsular diferentes comportamentos da classe Apostador sendo que caso o Apostador tenha uma quantidade pontuação dentro sistema ele é categorizado em bronze, prata, ouro. Abaixo temos o diagrama de classe do padrão de projeto.



10. PLANEJAMENTO DAS SPRINTS

10.1. BACKLOG

REQUISITO - Descrição	PRIORIDADE
[RF001] Gerenciar Operadores – Cadastrar, alterar, excluir e consultar Operador.	ALTA
[RF002] Gerenciar Campeonatos - Cadastrar, alterar, excluir e consultar Campeonatos.	ALTA
[RF003] Gerenciar Times - Cadastrar, alterar, excluir e consultar Times.	ALTA
[RF005] Gerenciar Apostador - Cadastrar, alterar, excluir e consultar Apostadores.	ALTA
[RF004] Gerenciar Partidas -	MÉDIA
[RF007] Gerenciar Palpite - Cadastrar, alterar, excluir e consultar Palpites.	MÉDIA
[RF008] Processar resultado e distribuir pontos	MÉDIA
[RF006] Cadastro Externo de Apostador	BAIXA
[RF009] Ranking de apostadores	BAIXA
[RF010] Consulta fácil de palpites	BAIXA
[RF011] Gerenciar Local de Origem	MÉDIA

[illegible]

11.1. Primeira SPRINT

REQUISITO: [RF001] GERENCIAR OPERADORES – Guilherme	
TAREFA	DURAÇÃO
Modelar Processo	4h
Especificar Funcionalidade	4h
Desenvolver Funcionalidade	12h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

REQUISITO: [RF002] GERENCIAR CAMPEONATOS – Bruno	
TAREFA	DURAÇÃO
Modelar Processo	4h
Especificar Funcionalidade	4h
Desenvolver Funcionalidade	12h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

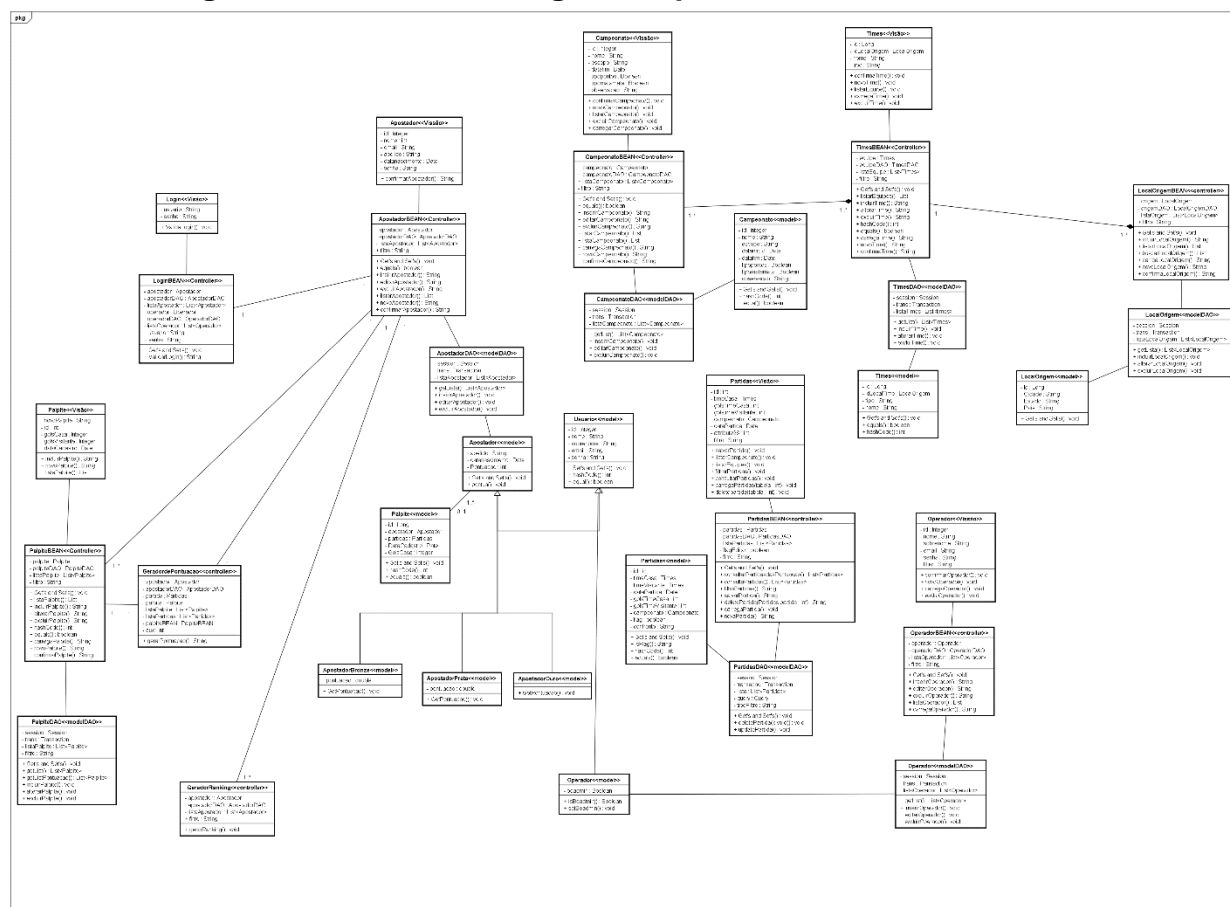
REQUISITO: [RF003] GERENCIAR TIMES - Bruno	
TAREFA	DURAÇÃO
Modelar Processo	4h
Especificar Funcionalidade	4h
Desenvolver Funcionalidade	12h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

REQUISITO: [RF005] GERENCIAR APOSTADOR - Thiago	
TAREFA	DURAÇÃO
Modelar Processo	4h
Especificar Funcionalidade	4h
Desenvolver Funcionalidade	12h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

- Período de desenvolvimento da Sprint: 1 mês.

Responsável	Previsto	Realizado
Bruno Henrique	Implementação para gerenciar os times	80%
Bruno Henrique	Implementação para gerenciar os campeonatos	80%
Guilherme Ganassin	Implementação no Gerenciamento de Operadores	80%
Thiago Bucalão	Implementação no Gerenciamento de Apostadores	80%

11.2. Diagrama de Classe da Segunda Sprint



11.3. Segunda *SPRINT*

Atividade	O que ficou sem fazer	Tempo
Gerenciar Times		
Gerenciar Operadores	<ul style="list-style-type: none">• Verificar se operador já foi cadastrado, não sendo permitido cadastrar dois operadores com o mesmo nome.• Corrigir falha de campos já pré-preenchidos ao criar novos operadores.	10h
Gerenciar Apostador	<ul style="list-style-type: none">• Verificar se um apostador já foi cadastrado, não sendo permitido cadastrar dois apostadores com mesmo nome.	8hs
Diagrama de Classes	<ul style="list-style-type: none">• Fazer diagrama classe de todos os casos de uso anteriores e o da Sprint atual.	6h
Padrão de Projeto	<ul style="list-style-type: none">• Definir e aplicar o padrão de projeto na Sprint atual	10h

REQUISITO: [RF004] GERENCIAR PARTIDAS – Bruno	
TAREFA	DURAÇÃO
Modelar Processo	4h
Especificar Funcionalidade	4h
Desenvolver Funcionalidade	12h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

REQUISITO: [RF010] CONSULTA FACIL DE PALPITE – Guilherme	
TAREFA	DURAÇÃO
Modelar Processo	2h
Especificar Funcionalidade	2h
Desenvolver Funcionalidade	10h
Testar Funcionalidade	2h
Validar e Realizar Commit	2h

REQUISITO: [RF006] GERENCIAR PALPITE – Thiago	
TAREFA	DURAÇÃO
Modelar Processo	2h
Especificar Funcionalidade	2h
Desenvolver Funcionalidade	20h
Testar Funcionalidade	2h
Validar e Realizar Commit	2h

REQUISITO: [RF008] RANKING DE APOSTADORES – Guilherme	
TAREFA	DURAÇÃO
Modelar Processo	2h
Especificar Funcionalidade	2h
Desenvolver Funcionalidade	12h
Testar Funcionalidade	2h
Validar e Realizar Commit	2h

REQUISITO: [RF010] GERENCIAR LOCAL ORIGEM – Bruno	
TAREFA	DURAÇÃO
Modelar Processo	3h
Especificar Funcionalidade	2h
Desenvolver Funcionalidade	14h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

REQUISITO: [RF007] PROCESSAR RESULTADO E DISTRIBUIR PONTOS - Thiago	
TAREFA	DURAÇÃO
Modelar Processo	4h
Especificar Funcionalidade	4h
Desenvolver Funcionalidade	18h
Testar Funcionalidade	4h
Validar e Realizar Commit	2h

- Período de desenvolvimento da Sprint: 2 meses.

Responsável	Previsto	Realizado
Bruno Henrique	Implementação para Gerenciar Partidas	90%
Bruno Henrique	Implementação para gerenciar Local Origem	100%
Guilherme Ganassin	Implementação do Ranking Apostador	80%
Guilherme Ganassin	Implementação do Consulta fácil de Palpites	80%
Thiago Bucalão	Implementação no Gerenciamento de Palpites	80%
Thiago Bucalão	Implementação de processamento de Resultados e distribuição de Pontos	80%

12. Realização dos Testes:

Funcionalidade Testada	Responsável pelo teste	Responsável pela implementação da funcionalidade
Gerenciar Times [RF003]	Guilherme	Bruno

Descrição: Testar o gerenciamento (inclusão e exclusão) de times no sistema.

Pré Condição: O usuário deve estar logado no sistema e fazer o cadastro de um time, com o local de origem (cidade onde o time possui sede) já devidamente cadastrada.

Procedimento: Localizar falhas ou validações incorretas que impeçam o Operador de fazer o cadastro de novos times no sistema.

Descrição Geral do Teste: Na tela principal do sistema, no menu superior existe a opção chamada “Times”, que é utilizada para que o Administrador ou o Operador cadastrem e gerenciem novos times, a fim de alimentar o sistema. Clicando sobre essa opção, a primeira tela que se abre mostra um painel com todos os times cadastrados no sistema, ordenados pelos seus ids (número único gerado no momento do cadastro do time). Nesta primeira tela existe o botão para cadastrar novos times, e junto a cada time mostrado estão as opções para editar ou remover o time do sistema.

A rotina de cadastro de times não apresentou falhas. Ao fazer o cadastro e não preencher um nome para o time, o sistema alerta que o nome é um campo obrigatório e não prossegue o cadastro até que este seja preenchido. A incrementação do ID para cada time também acontece de maneira correta.

A edição (alteração) de dados de times já cadastrados também ocorre de maneira correta, assim como a exclusão de registros.

Os problemas notados foram a possibilidade de cadastrar dois times com o mesmo nome e local de origem, e também quando é feita uma busca com o campo “Buscar Time” em branco, neste caso, o sistema passa a se comportar de maneira incorreta nas buscas seguintes.

Funcionalidade Testada	Responsável pelo teste	Responsável pela implementação da funcionalidade
Gerenciar Operadores [RF001]	Thiago	Guilherme

Descrição: Fazer testes sobre a rotina de cadastro e gerenciamento de Operadores no sistema

Pré Condição: O usuário precisa estar logado no sistema e possuir permissões de administrador

Procedimento: Encontrar falhas que impeçam o cadastro de novos Operadores ou o gerenciamento dos já existentes.

Descrição Geral do Teste: A tela de Operadores é acessada via menu principal, clicando sobre “Consulta > Operadores”. Feito isso, a tela de gerenciamento de Operadores se abre e é mostrada a lista de todos os Operadores cadastrados no sistema.

No cadastro existem validações sobre o nome e sobrenome do Operador, que não podem possuir menos que 4 caracteres, sobre o e-mail que precisa estar no formato padrão de e-mail (texto@texto) e também sobre a senha, que precisa ter pelo menos 5 caracteres. Caso algum dos campos não passe na validação, é exibida a mensagem em vermelho na parte superior da tela e o cadastro só é concluído após as alterações solicitadas.

Na tela principal, as rotinas de edição e exclusão dos Operadores já cadastrados também se comportaram de forma correta, e na edição foram aplicadas as mesmas validações do cadastro de novos operadores.

O sistema, de maneira incorreta, não emitiu um alerta quando foi cadastrado um novo Operador com o mesmo e-mail de um outro já cadastrado, os dados não foram gravados no banco, porém nenhuma mensagem indicando a falha foi emitida. Neste ponto, outro problema foi que o sistema permitiu que fosse gravado um novo Operador com o mesmo nome e sobrenome de outro já cadastrado no sistema.

A busca por nome também funcionou de maneira adequada, porém, ao fazer uma busca com o campo em branco o sistema traz todos os registros de Operadores gravados, porém, passa a dar erro nas buscas seguintes.

Funcionalidade Testada	Responsável pelo teste	Responsável pela implementação da funcionalidade
Gerenciar Apostador [RF005]	Bruno	Thiago

Descrição: Testar as rotinas de cadastro de Apostadores, que são os usuários que de fato irão utilizar o sistema

Pré Condição: Um usuário Operador deve estar logado no sistema

Procedimento: Localizar problemas no cadastro de Apostadores e verificar se a rotina funciona corretamente.

Descrição Geral do Teste: Para ter acesso à página de cadastro/gerenciamento de Apostadores é necessário, na página principal, clicar em “Consultas” e depois em “Apostadores”. Após isso, é aberta a tela de gerenciamento dos Apostadores, e de imediato são mostrados todos os operadores cadastrados e o botão que permite o cadastro de novos apostadores.

Selecionando a opção para cadastrar, são feitas validações de tamanho de campo (mínimo de 4 caracteres) no nome e sobrenome; no e-mail, onde é necessário que o endereço esteja no formato aceito, e também na senha, que precisa ter 5 ou mais caracteres. Nesta etapa, um erro é apresentado no campo que recebe a data, onde aparecem apenas a opção do ano de 2006 em diante (alterando manualmente no campo o sistema permite gravar a data correta).

O sistema não permite gravar dois apostadores com o mesmo e-mail, porém, ao finalizar a gravação utilizando um e-mail já cadastrado, não é gerada nenhuma mensagem, e o sistema apenas não grava as informações no banco, ou seja, nenhum alerta indicando o problema é emitido. Além disso, foi possível cadastrar dois usuários com mesmo nome, sobrenome e apelido, e também, preencher a data de nascimento de um apostador com um valor maior que a data atual.

13. Prints das telas:

Tela de Login:

Bolão Web

Usuário
 Senha

Tela Principal do Operador:

Bolão Web Consulta ▾ Sair

Bem vindo ao Bolão Web

UEM - Implementação de Sistema de Software 2015

Bolão Web Consulta ▾ Sair

Campeonato
 Operadores
 Times
 Apostadores
 Partidas
 Local Origem

B

ndo ao Bolão Web

UEM - Implementação de Sistema de Software 2015

Consulta de Campeonatos:

Bolão Web Consulta ▾ Sair

Consulta de Campeonato

Buscar Campeonato:

ID	Nome	Escopo	Data Inicio	Data Fim	Pontos	Mata-Mata	Observações	Ações		
1	Campeonato Brasileiro	Nacional	05/03/2016	04/12/2016	✓	☐	Brasileirão	Editar		Remover
2	Libertadores	Continental	17/02/2016	25/05/2016	✓	✓		Editar		Remover
3	Copa do Mundo	Mundial	05/06/2016	10/07/2016	✓	✓		Editar		Remover

UEM - Implementação de Sistema de Software 2015

Novo Campeonato / Edição de Campeonato:

Bolão Web Consulta ▾

Cadastro de Campeonato

ID:

Nome:

Escopo: Regional ▾

Dt. Início: Clique aqui!

Dt. Fim: Clique aqui!

Tipo: ☐ Pontos ☐ Mata-Mata

Observações

Confirmar Voltar

UEM - Implementação de Sistema de Software 2015

Consulta de Operador:

Bolão Web Consulta ▾ Sair

Consulta de Operador

+ Novo Operador

Buscar Operador Buscar

ID	Nome	Sobrenome	Email	Ações		
2	Guilherme	Ganassin	guilherme@teste.com	Editar		Remover
3	Thiago	Bucalon	thiago@teste.com	Editar		Remover
6	Rogério	Ceni	rogerio@saopaulo.com	Editar		Remover

UEM - Implementação de Sistema de Software 2015

Cadastro/Edição de Operador:

Bolão Web Consulta ▾

Cadastro de Operador

ID:

Nome:

Sobrenome:

Email:

Senha:

Confirmar Voltar

UEM - Implementação de Sistema de Software 2015

Consulta de Times:

Bolão Web

Consulta ▾

Sair

Consulta de Times

+ Novo Time

Buscar Time:

Buscar

ID	Local Origem	Tipo	Nome	Ações			
1	São Paulo	Time	São Paulo da Floresta	Editar		Remover	
2	Barcelona	Time	Barcelona	Editar		Remover	
3	Santos	Time	Santos	Editar		Remover	
4	Milão	Time	Milan	Editar		Remover	
5	Buenos Aires	Time	Boca Juniors	Editar		Remover	
6	Madrid	Time	Real Madrid	Editar		Remover	
7	São Paulo	Seleção	Brasil	Editar		Remover	
8	Buenos Aires	Seleção	Argentina	Editar		Remover	
9	Barcelona	Seleção	Espanha	Editar		Remover	
10	Milão	Seleção	Italia	Editar		Remover	

Retornar ao Menu Inicial

UEM - Implementação de Sistema de Software 2015

Cadastro/Edição de Times:

Bolão Web

Consulta ▾

Cadastro de Time

ID:

Local de Origem:

São Paulo ▾

Nome:

Tipo:

Time ▾

Confirmar

Cancelar

Retornar ao Menu Inicial

UEM - Implementação de Sistema de Software 2015

Consulta de Apostadores:

Bolão Web

Consulta ▾

Sair

Consulta de Apostador

+ Novo Apostador

Buscar Apostador (nome):

Buscar

ID	Nome	Sobrenome	Apelido	Email	Data Nascimento	Ações		
1	Marcelo	Pinheiro	Teste	marcelo@teste.com.br	09/08/91	Editar		Remover
9	Bruno	Kunieda	Bruno	bruno@teste.com	04/02/91	Editar		Remover
12	Paulo	Sousa	PauloSousa	paulo@sousa.com	02/02/84	Editar		Remover
13	João	Oliveira	Jão96	joao@teste.com	03/02/96	Editar		Remover
14	Guilherme	Apostador	GGApostador	guilherme@apostador.com	31/03/91	Editar		Remover

UEM - Implementação de Sistema de Software 2015

Cadastro/Edição de Apostadores:

Bolão Web Consulta ▾

Cadastro de Apostador


ID:

Nome:

Sobrenome:

Apelido:

Email:

Data Nascimento: 

Senha:

UEM - Implementação de Sistema de Software 2015

Consulta de Partidas:

Bolão Web Consulta ▾ Sair

Consulta de Partidas

Pesquisar Partida:

Data	Time Casa	Gols Time Casa	X	Gols Time Visitante	Time Visitante	Ações			
13/04/2016	São Paulo	5	X	1	Boca Juniors	<input type="button" value="Editar"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>
08/05/2016	São Paulo	4	X	0	Santos	<input type="button" value="Editar"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>
12/06/2016	Brasil	7	X	1	Alemanha	<input type="button" value="Editar"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>
26/06/2016	Italia	1	X	2	Espanha	<input type="button" value="Editar"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>	<input type="button" value="Remover"/>

UEM - Implementação de Sistema de Software 2015

Cadastro/Atualização de Partidas:

Bolão Web Consulta ▾

Cadastro de Partida

Data: 

Time Casa:

Gols Time Casa:

Time Visitante:

Gols Time Visitante:

Campeonato:

UEM - Implementação de Sistema de Software 2015

Consulta Local de Origem:

Bolão Web

Consulta ▾

Sair

Consulta de local origem

+ Novo Local Origem

Buscar local origem:

Buscar

ID	Cidade	Estado	País	Ações			
1	São Paulo	São Paulo	Brasil	Editar		Remover	
2	Barcelona	-	Espanha	Editar		Remover	
3	Santos	São Paulo	Brasil	Editar		Remover	
4	Milão	-	Italia	Editar		Remover	
5	Buenos Aires	-	Argentina	Editar		Remover	
6	Madrid	-	Espanha	Editar		Remover	
7	Berlim	-	Alemanha	Editar		Remover	

Retornar ao Menu Inicial

UEM - Implementação de Sistema de Software 2015

Cadastro/Alteração de Local de Origem:

Bolão Web

Consulta ▾

Cadastro de local de origem

ID:

Cidade:

Estado:

País:

Confirmar

Cancelar

Retornar ao Menu Inicial

UEM - Implementação de Sistema de Software 2015

Tela Principal do Apostador:

Bolão Web

Consulta ▾

Ranking

Contato

Sair

Bem vindo ao Bolão Web

Paulo

UEM - Implementação de Sistema de Software 2015

Consulta de Campeonatos em disputa:

Bolão Web

Consulta ▾

Ranking

Contato

Sair

Consulta de Campeonato

Buscar Campeonato:

Nome	Escopo	Data Inicio	Data Fim	Pontos	Mata-Mata	Observações
Campeonato Brasileiro	Nacional	05/03/2016	04/12/2016	✓	<input type="checkbox"/>	Brasileirão
Libertadores	Continental	17/02/2016	25/05/2016	✓	✓	
Copa do Mundo	Mundial	05/06/2016	10/07/2016	✓	✓	

UEM - Implementação de Sistema de Software 2015

Consulta de Partidas:

Bolão Web

Consulta ▾

Ranking

Contato

Sair

Consulta de Partidas

Pesquisar Partida:

Data	Time Casa		Time Visitante	Ações
13/04/2016	São Paulo	X	Boca Juniors	<input type="button" value="Palpitar"/>
08/05/2016	São Paulo	X	Santos	<input type="button" value="Palpitar"/>
12/06/2016	Brasil	X	Alemanha	<input type="button" value="Palpitar"/>
26/06/2016	Italia	X	Espanha	<input type="button" value="Palpitar"/>

UEM - Implementação de Sistema de Software 2015

Consulta de Palpites:

Meus Palpites

Partida	Data de Cadastro	Gols time casa	Gols time visitante
---------	------------------	----------------	---------------------

[Retornar ao Menu Inicial](#)

UEM - Implementação de Sistema de Software 2015

Raking de Apostadores:

Ranking de Apostador

Selecionar Campeonato

Selecionar Apostador [Buscar](#)

Posição	Nome	Sobrenome	Pontos
0 °	Marcelo	Pinheiro	0
0 °	Bruno	Kunieda	0
0 °	Paulo	Sousa	0
0 °	João	Oliveira	0
0 °	Guilherme	Apostador	0

[Retornar ao Menu Inicial](#)

UEM - Implementação de Sistema de Software 2015