

UNIVERSIDADE DO MINHO



ENGENHARIA WEB
ENGENHARIA DE SISTEMAS DE SOFTWARE

Betting House Web App - Parte 1

Autores:

Manuel Sousa
Tiago Alves
Lázaro Lima

Identificação:

A78869
A78218
ID8120

18 de Abril de 2019

Conteúdo

1	Introdução	2
1.1	Requisitos Não-Funcionais	2
2	Solução proposta	3
2.1	Modelo de dados	3
2.1.1	Utilizadores	3
2.1.2	Eventos	4
2.1.3	Apostas	4
2.1.4	Administrador	4
2.2	WebRatio	5
3	Conclusões	8
4	Anexos	9

1 Introdução

Os sistemas de apostas online desportivas converteram-se num grande negócio e, simultaneamente, transformaram a Internet numa espécie de casa de apostas onde há cenários de intensa procura e grandes investimentos são realizados.

Um sistema de apostas online baseia-se nos ganhos e nas perdas que são obtidos com as apostas, sendo o mais comum as apostas desportivas. À primeira vista, a adopção de um sistema de apostas possibilita poupar e rentabilizar o dinheiro do apostador que visa a maximização dos lucros das suas apostas desportivas.

Com esta primeira fase, o grupo pretende expor o trabalho desenvolvido aquando a modelação de um sistema deste tipo, desde a especificação do modelo de dados até à modelação do sistema e respetivas operações.

Requisitos

A aplicação desenvolvida deve ter em conta utilizadores classificados em três tipos de perfil, são eles:

- **Administrador:** perfil responsável pela gestão da informação na aplicação, como os eventos, utilizadores, apostas, etc. Deverá ter ao seu dispor uma interface para que possa cumprir com os seus deveres.
- **Utilizador Premium:** paga uma quantia que lhe dá acesso a informações extra de eventos, eventos restritos, etc.
- **Utilizador Regular:** este perfil representa os apostadores regulares.

1.1 Requisitos Não-Funcionais

- **Real-time responsiveness:** a aplicação lida com apostas em eventos, logo apostas em eventos que já terminaram não devem ser permitidas.
- **Escalabilidade:** supostamente a aplicação deve poder aguentar com grandes quantidades de pedidos em certos eventos, como por exemplo, Liga dos Campeões, Super Bowl, etc., e deve responder quase instantaneamente.
- **Interface:** deve seguir pelo menos o nível A do WCAG (Web Content Accessibility Guidelines)

2 Solução proposta

2.1 Modelo de dados

Tendo em conta os requisitos do sistema Betting House mencionados anteriormente, foi elaborado o seguinte modelo de dados visto na figura 5:

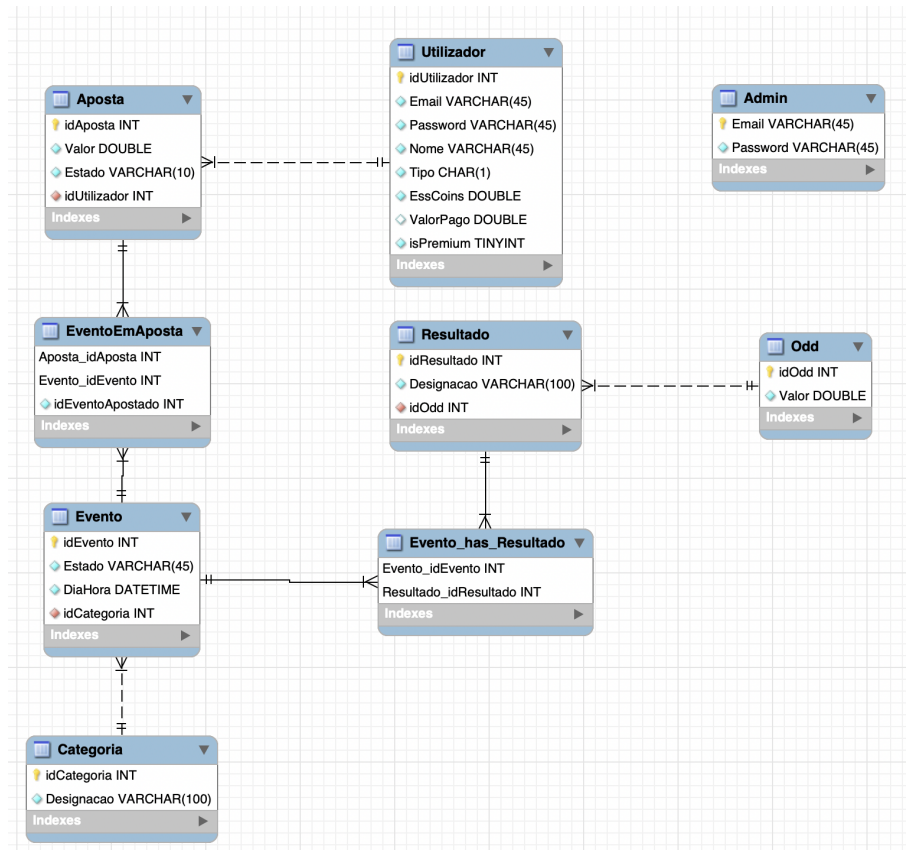


Figura 1: Base de dados do sistema Betting House

2.1.1 Utilizadores

Os utilizadores que constituem esta aplicação são o *Utilizador Normal* e o *Utilizador Premium*. Cada um deles precisa de se registar no sistema fornecendo o email e password. Além disso, terão também um nome, e as *Ess Coins* associadas a cada um deles.

Estes são os campos comuns de ambos os utilizadores, logo gerou-se uma tabela **Utilizador**, com o intuito de armazenar esta informação.

2.1.2 Eventos

Um evento pode ser derivado de diferentes categorias, por exemplo, Futebol, Corrida de cavalos, etc. Por isto, foi criada uma tabela ***Categoria***, que caracteriza a entidade categoria, com o atributo *Designação*.

As características de um determinado evento são o seu estado, que pode estar aberto ou fechado, o dia e hora, e o resultado. Relativamente a este “Resultado”, é tido em conta como se fosse um participante, ou seja, numa corrida de cavalos, temos como resultados possíveis o cavalo vermelho, o verde, o amarelo, etc. Num jogo de futebol, temos como resultados possíveis a equipa visitada, o empate ou a equipa visitante. Logo, foi gerada uma tabela ***Resultado***, com o atributo *Designação*. O relacionamento desta tabela com a tabela Evento é de muitos para muitos, porque um evento tem vários resultados a si associados, e um resultado pode estar em diversos eventos.

Visto que um resultado pode estar associado a vários eventos, foi criada a tabela ***odd***, para armazenar as odds relativas a um resultado de um certo evento.

2.1.3 Apostas

Um utilizador pode realizar várias apostas. Cada uma dessas apostas pode ser relativa a vários eventos. No entanto, este sistema não permite apostas múltiplas, ou seja, não é permitida a aposta num evento em mais do que um resultado. Foi, então, gerada uma tabela ***Aposta***, que contém os atributos Valor, que representa o valor apostado, e o estado da aposta, que pode ser aberto ou fechado. O estado de uma aposta muda para “fechado” apenas quando todos os eventos constituintes da mesma estiverem fechados.

2.1.4 Administrador

Como responsável pela gestão da plataforma, como adicionar eventos, alterar estados de eventos, etc., existe o utilizador ***Admin***, que se autentica no sistema com o email e password.

2.2 WebRatio

Depois de construído o modelo de dados sobre o qual iria ser suportada toda a aplicação, passamos então para a modelação no *WebRatio*[1]. O primeiro passo foi a criação do Modelo de Domínio baseado no modelo de dados anterior, depois a conexão à base de dados *MySQL*. Nessa etapa foram gerados scripts *SQL* capazes de criar as respetivas tabelas de todas as entidades. Posto isto, foram então criadas as vistas para o website, nas quais eram modeladas todas as operações.

Aquando a modelação das vistas, surgiram diversos desafios e dificuldades, das quais algumas não foram possíveis serem ultrapassadas. O facto de certas operações serem complexas de se desenvolver e o acesso a documentação ser escasso e não muito específico, levou a que o grupo produzisse um modelo geral da aplicação. Mais tarde, a falha na manipulação de chaves estrangeiras de outras tabelas, levou à criação de um novo modelo de dados, o qual seria uma simplificação do modelo anteriormente produzido.

Visto isto, certas associações não eram possíveis, e por essa mesma razão, o controlo de certas operações não existe. Para além disso, a dificuldade na manipulação de certos componentes levou a que campos fossem tratados de forma estática, o que retira o realismo do modelo.

Por fim, foi então produzido o seguinte modelo, capaz de processar operações básicas, e contendo 2 tipos de menus. Um menu de administrador, o qual controla os eventos, utilizadores e apostas, e um menu do utilizador, o qual permite apostar nos eventos previamente criados pelo administrador. É importante realçar que, no modelo de domínio, as entidades *User*, *Group* e *Module*, as quais são automaticamente criadas pelo *WebRatio*, teriam sido usadas para a implementação de autenticação. Visto não ser possível a correta manipulação destas entidades, não foi possível desenvolver a componente de autenticação.

Por motivos de espaço, as imagens relativas aos menus de cada entidade foram movidas para a secção dos anexos.

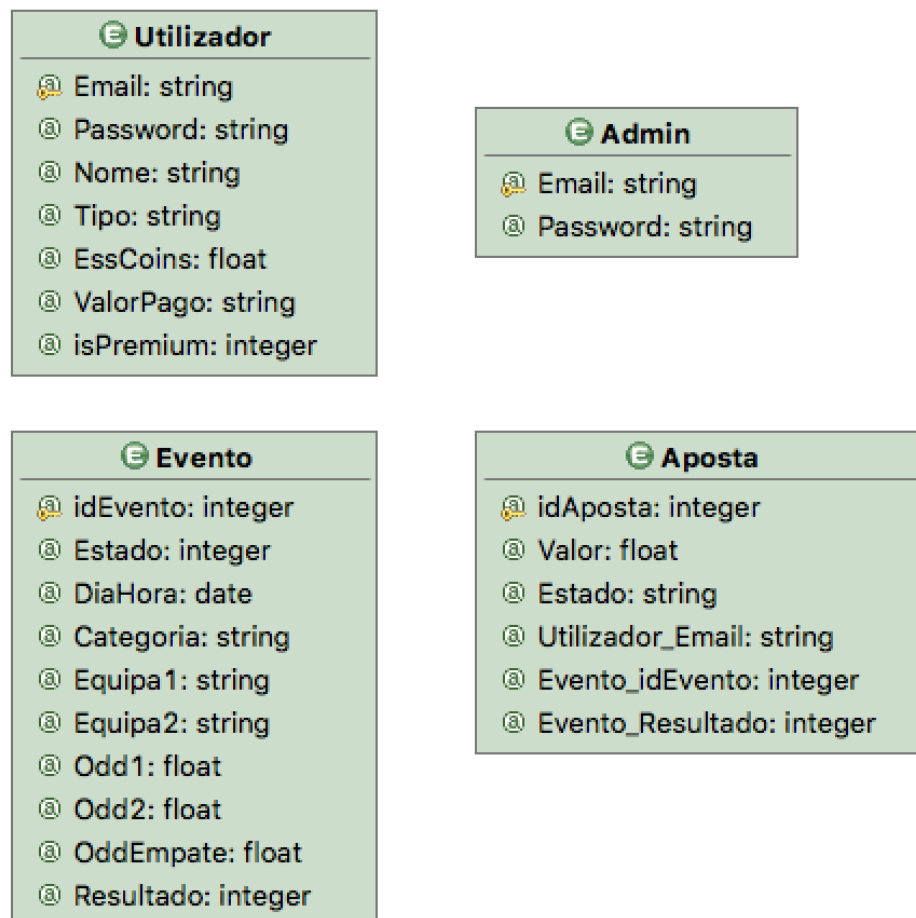


Figura 2: Modelo de Domínio do sistema desenvolvido

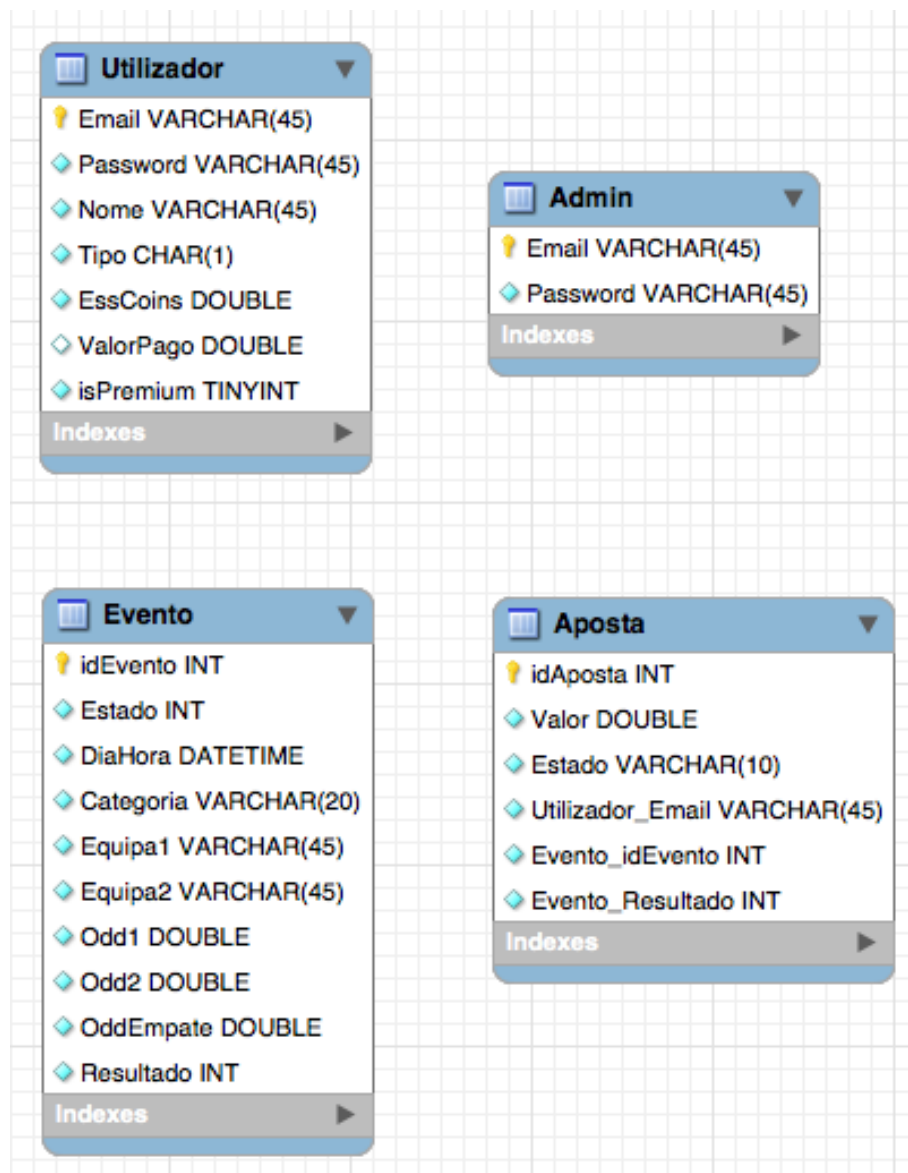


Figura 3: Modelo Lógico produzido para o WebRatio

3 Conclusões

A solução proposta pelo grupo para a implementação do sistema *Betting House* foi pensada de tal forma a que um utilizador da aplicação possa realizar apostas em diversos tipos de eventos, ou seja, incluem-se categorias neste sistema, como por exemplo, Futebol, Corridas de Cavalos, Atletismo, etc. O modelo de dados apresentado nos tópicos acima baseia-se exatamente nesta ideia generalizada de apostas.

Um sistema generalizado como este traz sempre algumas dificuldades de implementação. Neste caso, as maiores dificuldades encontraram-se no WebRatio. Primeiramente, a instalação da plataforma em cada uma das máquinas dos elementos do grupo teve um certo grau de dificuldade, devido às dependências de tecnologias que a ferramenta apresentava. Em segundo, depois de finalmente arrancada a aplicação, foram encontrados desafios ao estabelecer a conexão à base de dados do sistema.

Grande parte do tempo foi dedicado a entender a forma de funcionamento da ferramenta mencionada, e como se processavam certas operações usando os componentes disponíveis, visto que eram tarefas bastante específicas, nas quais a documentação usada não satisfaz as necessidades do grupo.

Referências

- [1] Marco Brambilla, Sara Comai, Piero Fraternali, and Maristella Matera. Designing web applications with webml and webratio. In *Web Engineering: Modelling and Implementing Web Applications*, pages 221–261. Springer, 2008.

4 Anexos

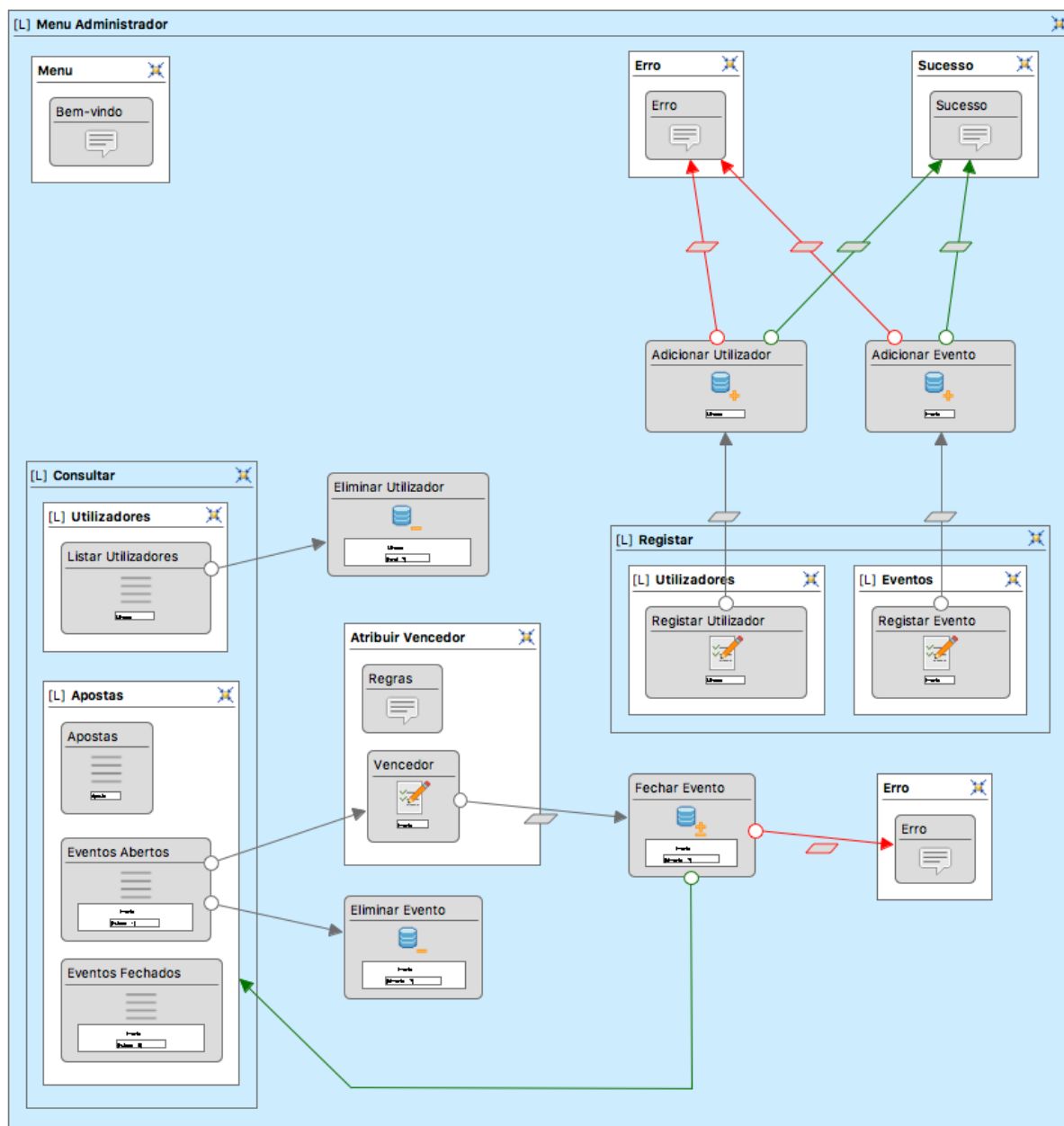


Figura 4: Menu do Administrador - WebRatio

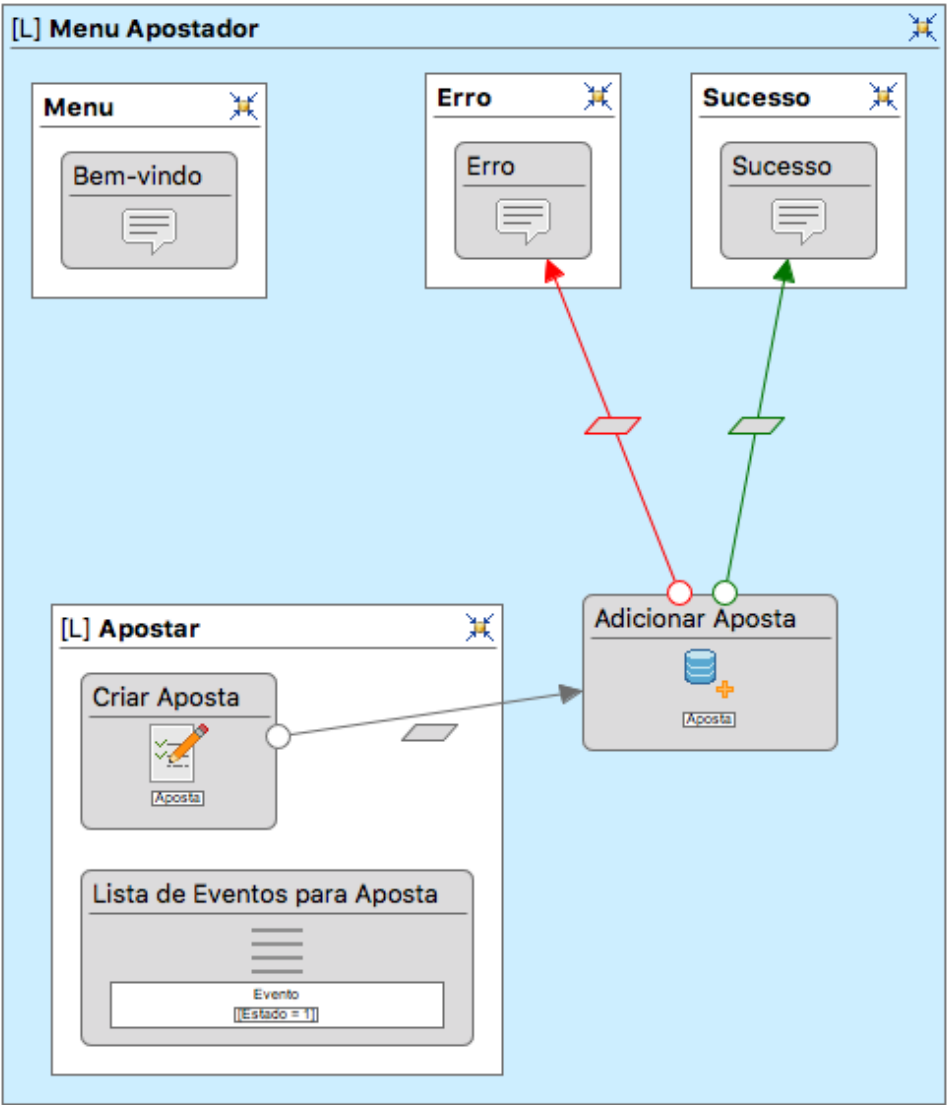


Figura 5: Menu do Apostador - WebRatio