

**MINISTÉRIO DA EDUCAÇÃO**

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO**

***CAMPUS* BIRIGUI**

CNPJ 10.882.594/0001-65

Rua Pedro Cavalo, 709, Portal da Pérola II – Birigui – SP – CEP 16201-407

Telefone (18) 3643-1160

**CURSO SUPERIOR DE TECNOLOGIA**

**EM SISTEMAS PARA INTERNET**

**Bruno Wake Moraes Gonçalves**

**TRABALHO DE CONCLUSÃO DE CURSO**

**Moradia de Estudante**

Birigui-SP

2017

**BRUNO WAKE MORAES GONÇALVES**

**BRUNO WAKE MORAES GONÇALVES**

**Moradia de Estudante**

Trabalho de conclusão de curso apresentado à banca examinadora do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Birigui, como requisito à obtenção do grau de Tecnólogo em Sistemas para Internet sob a orientação do professor   
Me. Murilo Varges Da Silva.

Birigui – SP

2017

**AGRADECIMENTOS**

Agradeço aos meus pais, amigos e familiares que me deram forças para terminar este trabalho.

Agradeço ao meu professor Murilo Varges da Silva que me auxiliou na conclusão desse curso.

**BRUNO WAKE MORAES GONÇALVES**

**RESUMO**

Este projeto foi desenvolvido com o propósito de ajudar os discentes e docentes que querem alugar um dormitório compartilhado através da internet. Com base no sistema de filtragem por perfil, a Moradia de Estudante é uma alternativa para encontrar um bom dormitório com parceiros que gostem e façam as mesmas coisas que você. O software prioriza a privacidade de seus dados, este site mostra seu telefone ou celular apenas quando as pessoas interessadas são aprovadas pelos proprietários.

**PALAVRAS-CHAVE**: Hibernate, JSF, MYSQL, Primefaces, Bootstrap.

**ABSTRACT**

This project was developed with the purpose of helping students and teachers who want to rent a shared dormitory through the internet. Based on the profile filtering system, the Moradia de Estudante is an alternative to find a good dormitory with partners who enjoy and do the same things as you. The software prioritizes the privacy of your data, this website shows your phone or mobile only when the interested people are approved by the owners. The purpose of this project is to help the emerging niche market for real estate rentals to the academic people.

**Lista de Tabelas**

Tabela 1 - Requisitos funcionais do sistema 8

Tabela 2 - Casos de Uso Usuário (selecionar estado) 11

Tabela 3 - Casos de Uso Usuário (Visualizar lista de anúncios) 12

Tabela 4- Casos de Uso Usuário (filtrar imóvel) 12

Tabela 5 - Casos de Uso Usuário (Cadastrar) 13

Tabela 6 - Casos de Uso Inquilino (*Login*) [15](#_heading=h.2xcytpi)

Tabela 7 - Casos de Uso Inquilino (Recuperar senha) 15

Tabela 8 - Casos de Uso Inquilino (Demonstrar interesse no imóvel) 16

Tabela 9 - Casos de Uso Inquilino (Alterar senha) 16

Tabela 10 - Casos de Uso Inquilino (Alterar usuário) 17

Tabela 11 - Casos de Uso Inquilino (Alterar perfil usuário) 17

Tabela 12 - Casos de Uso Inquilino (Consultar anúncios que tenho interesse) 17

Tabela 13 - Casos de Uso Anunciante (Consultar interessados no meu anúncio) 19

Tabela 14 - Casos de Uso Anunciante (Criar anúncio) 21

Tabela 15 - Casos de Uso Anunciante (Consultar anúncios do meu usuário) 22

Tabela 16 - Casos de Uso Anunciante (Excluir anúncio) 22

Tabela 17 - Casos de Uso Anunciante (Alterar anúncio) 23

Tabela 18 - Casos de Uso Anunciante (alterar perfil do anuncio) 23

Tabela 19 - Casos de Uso Anunciante (Adicionar fotos) 24

Tabela 20 - Casos de Uso Anunciante (Consultar fotos) 24

Tabela 21 - Casos de Uso Anunciante (Excluir fotos) 25

Tabela 22- Casos de Uso Administrador (Acessar área administrativa) 26

Tabela 23 - Casos de Uso Administrador (Alterar qualquer usuário) 27

Tabela 24 - Casos de Uso Administrador (Alterar perfil) 27

Tabela 25 - Casos de Uso Administrador (Excluir usuário) 28

Tabela 26 - Casos de Uso Administrador (Consultar anúncio qualquer usuário) 29

Tabela 27 - Casos de Uso Administrador (alterar anúncio) 29

Tabela 28 - Casos de Uso Administrador (Alterar perfil do anúncio) 30

Tabela 29 - Casos de Uso Administrador (Excluir anúncio) 30

Tabela 30 - Casos de Uso Administrador (Adicionar fotos) 31

Tabela 31 - Casos de Uso Administrador (Consultar fotos) 31

Tabela 32 - Casos de Uso Administrador (Excluir fotos) 32

Tabela 33 - Projeto de dados (anúncio) 51

Tabela 34 - Projeto de dados (perfil) 52

Tabela 35 - Projeto de dados (usuário) 53

Tabela 36 - Projeto de dados (anuncio\_vaga) 53

Tabela 37 - Projeto de dados (imagens) 54

Tabela 38 - Projeto de dados (log\_login) 54

Tabela 39 - Projeto de dados (log\_operacoes) 54

**Lista de Figuras**

Figura 1- Diagrama de domínio 9

Figura 2 - Diagrama de casos de uso (usuário) 10

Figura 3 -Diagrama de casos de uso (inquilino) 14

Figura 4 -Diagrama de casos de uso (anunciante) 19

Figura 5 - Diagrama de casos de uso (anunciante parte 2) 21

Figura 6 - Diagrama de casos de uso (administrador) 26

Figura 7 - Diagrama de casos de uso (administrador2) 28

Figura 8 - EER Diagram Mysql workbench concluído 33

Figura 9 - Diagrama Mysql workbench (ferramentas) 34

Figura 10 - Diagrama Mysql workbench (criando tabelas e colunas) 35

Figura 11 - Diagrama Mysql workbench (relacionando tabelas) 36

Figura 12 - Mysql workbench foward engineer - conexão com bando de dados 37

Figura 13 - Mysql workbench foward engineer - opções de criação de banco 38

Figura 14 - Mysql workbench foward engineer - seleção de objetos 39

Figura 15 - Mysql workbench foward engineer - código sql 40

Figura 16 - Mysql workbench foward engineer – conclusão 41

Figura 17 - Camadas hibernate 42

Figura 18 - Pacotes do Hibernate 42

Figura 19 - Opções do projeto 43

Figura 20- Arquivos hibernate 44

Figura 21 - Arquivos criados Hibernate 45

Figura 22 - Hibernate.cfg.xml 46

Figura 23 - Hibernate.revenge.xml 47

Figura 24 - Classe de mapeamento Hibernate (Anuncio) 48

Figura 25 - Classe Java – Engenharia reversa Hibernate (Anuncio) 49

Figura 26 - Código hibernateutil 49

Figura 27 - Menu/Filtro 55

Figura 28 - Adicionar anúncio 56

Figura 29 - Meus anúncios 57

Figura 30 - Demonstrar interesse 57

Figura 31 - Confirmar interesse 58

Figura 32 - Alteração senha 59

Figura 33 - Alteração usuário 59

Figura 34 - Confirmar interesse 60

Figura 35 - Dialog telefone 61

Figura 36 - Dialog perfil interessado 61

Figura 37 - Dialog aceitar 62

Figura 38 - Filtro administrador 63

**Sumário**

[1.](#_heading=h.gjdgxs) Introdução 6

[2.](#_heading=h.30j0zll) Objetivo do sistema 6

[2.1 Objetivo geral 6](#_heading=h.1fob9te)

[2.2 Objetivo específico 6](#_heading=h.3znysh7)

[3.](#_heading=h.2et92p0) Justificativa 7

[4.](#_heading=h.tyjcwt) Descrição geral do produto 7

[5.](#_heading=h.3dy6vkm) Requisitos 7

[6.1 Requisitos do Sistema 7](#_heading=h.1t3h5sf)

[6.2 Requisitos funcionais do sistema 7](#_heading=h.4d34og8)

[6.](#_heading=h.17dp8vu) Diagrama de domínio 9

[7.](#_heading=h.26in1rg) Diagrama de casos de uso 10

[7.1.](#_heading=h.lnxbz9) Usuário 10

[7.2.](#_heading=h.1y810tw) Inquilino 12

[7.3.](#_heading=h.2p2csry) Anunciante 17

[7.4.](#_heading=h.3tbugp1) Administrador 24

[8.](#_heading=h.1rvwp1q) Tecnologias usadas 31

[8.1.](#_heading=h.4bvk7pj) Mysql Workbench 31

[8.2.](#_heading=h.xvir7l) Hibernate 40

[8.3.](#_heading=h.2nusc19) JSF 49

[8.4.](#_heading=h.1302m92) Primefaces 49

[8.5.](#_heading=h.3mzq4wv) Bootstrap 50

[9.](#_heading=h.2250f4o) Projeto de dados 50

[10.](#_heading=h.1tuee74) Projeto de interface 54

[10.1.](#_heading=h.2szc72q) Inclusão de anúncios e gerencia de imóveis 55

[10.2.](#_heading=h.279ka65) Demonstrar interesse: 56

[10.3.](#_heading=h.1ljsd9k) Alterar usuário e senha 58

[10.4.](#_heading=h.zu0gcz) Gerenciar interessados no imóvel e ver telefone 60

[10.5.](#_heading=h.1d96cc0) Utilizar à area administrativa 62

[11.](#_heading=h.2ce457m) Considerações Finais 63

[12.](#_heading=h.rjefff) Melhorias Futuras 63

[13.](#_heading=h.3bj1y38) Referências bibliográficas 64

# Introdução

No ano de 2014, o Brasil tinha cerca de 2314 instituições de ensino superior, sendo 245 instituições públicas e 2069 particulares. Essas instituições podem ser classificadas da seguinte maneira: 186 universidades, 127 centros universitários e 1.966 faculdades. Além disso, 35 instituições federais públicas de educação profissional e tecnológica oferecem cursos superiores. Segundo Fernando Haddad, ministro da educação, “A evolução dos números da educação brasileira tem sido satisfatória, e o mais importante é que essa evolução tem ocorrido com qualidade no ensino”. “Total de estudantes formados no País dobrou em sete anos, segundo Censo da Educação.” (Governo Brasil, 2014)

Esse aumento tem sido considerável se compararmos a quantidade de matriculas de alunos entre o ano de 2002 e 2009, cuja a diferença é de mais de 2 milhões de matriculados. A estimativa geral de matriculas, em 2009, era aproximadamente 5,9 milhões de alunos e o índice de ingressão era superior a 50%. Além de uma grande quantidade de novos alunos, temos cerca de 340.817 professores em atividade. (Governo Brasil, 2014)

Uma das maiores dificuldades enfrentadas por professores e estudantes que necessitam se deslocar é achar uma moradia, principalmente entre os meses de janeiro e fevereiro, onde as buscas aumentam em torno de 50% no ano de 2015. (Governo Brasil, 2014)

Levando em consideração os dados citados, este projeto visa auxiliar as partes interessadas na obtenção de moradia, através de um site de anúncio de imóveis, onde o intermédio de terceiros não seja necessário e as informações pessoais (telefone e celular) do anunciante só seja mostrado para quem tem interesse, dessa forma, evitando possíveis transtornos.

# Objetivo do sistema

Concentrar anúncios de imóveis para o público universitário em um local específico onde, o conforto e convivência dos moradores sejam levados em consideração desde o cadastro até a hora da contratação.

## 2.1 Objetivo específico

Auxiliar o aluguel e a locação de imóveis coletivos em cidades emergentes no “meio universitário”, facilitando a busca e o anúncio de imóveis através de perfis e filtros de busca, evitando contatos desnecessários entre o anunciante e o locador.

# Justificativa

Atualmente é notado uma dificuldade, relacionada ao tempo, na hora de buscar um imóvel através dos meios convencionais. Gastos supérfluos para o locador e a falta de comunicação entre os envolvidos agravam ainda mais o problema, principalmente quando uma das partes reside em outra cidade. O Moradia de estudante vem como alternativa para suprir essas dificuldades de forma prática, confortável e direta, evitando contatos excessivos entre as partes interessadas.

# Descrição geral do produto

O Moradia de Estudante é um *software* voltado inteiramente para o ambiente *web*, focado para o aluguel de imóveis baseado em perfis. O intuito do *software* é facilitar a vida de docentes e discentes que procuram um local para morar. O próprio nome da aplicação e a forma que é construído colabora para ser um "filtro" natural onde, a maioria dos acessos será de pessoas do meio acadêmico. O sistema de perfil permite que um imóvel seja alugado por pessoas que tenham os mesmos interesses, dessa forma reduzindo as chances de que o inquilino mude de residência pela má convivência com seus companheiros. O *software* também visa a integridade dos seus dados, e para evitar trotes ou ligações desnecessárias para o número de telefone informado no imóvel, o interessado só tem acesso ao contato após ser aceito pelo anunciante.

# Requisitos

Os requisitos são a capacidade e a condição de compreender o sistema e, de forma mais ampla, o projeto. O desafio principal do requisito é encontrar, lembrar e comunicar o que é realmente necessário de uma forma clara que o cliente e a equipe de desenvolvimento compreendam. (LARMAN, 2001).

Os requisitos funcionais do sistema são a base para o desenvolvimento dos demais diagramas do projeto e o desenvolvimento do *software*. Os requisitos expressam de forma direta e simples as funcionalidades do *software*. O diagrama de casos de uso complementa os requisitos de uma forma mais exemplificada.

## 5.1 Requisitos funcionais do sistema

Tabela 1 - Requisitos funcionais do sistema

|  |  |  |
| --- | --- | --- |
| **Indicador** | **Descrição** | **Depende** |
| C1 | O programa deverá mostrar uma lista de estados onde tem anúncios disponíveis |  |
| C2 | O programa deverá permitir que o USUARIO visualize os anúncios do site |  |
| C4 | O programa deverá permitir que o USUARIO filtre os anúncios por perfil, preço, bairro e valor |  |
| C5 | O USUARIO poderá fazer o cadastro para obter acesso ao site, deverá usar os seguintes campos: nome, senha, *e-mail*, foto, rg, cpf, sexo, Data de nascimento, telefone, celular. |  |
| C6 | O USUARIO deverá fazer o cadastro do perfil referente ao usuário com os seguintes campos: fuma, bebe, chega em casa de madrugada, solteiro, faz festas em casa, cozinha e limpa a casa. | C5 |
| C7 | O INQUILINO poderá fazer o *login* com o *e-mail* e com a senha | C5 |
| C8 | O INQUILINO poderá demonstrar interesse no imóvel que desejar | C7 |
| C9 | O INQUILINO poderá alterar a senha do usuário | C7 |
| C10 | O INQUILINO poderá alterar os seguintes dados do usuário: nome, *e-mail*, foto, rg, cpf, sexo, data de nascimento, telefone e celular. Também poderá alterar os seguintes campos do perfil do usuário: fuma, bebe, chega em casa de madrugada, solteiro, faz festas em casa, cozinha e limpa a casa. | C7 |
| C11 | O INQUILINO poderá consultar os anúncios que tem interesse e se for aceito poderá ver o telefone para contato do imóvel | C7, C16 |
| C12 | O anunciante poderá incluir, alterar e excluir anúncios com os seguintes campos: rua, bairro, estado, cidade, descrição, quantidade de vagas, valor total. | C7 |
| C13 | O anunciante poderá incluir um perfil para o imóvel com os seguintes campos: fuma, bebe, chega em casa de madrugada, solteiro, faz festas em casa, cozinha e limpa a casa. | C12 |
| C14 | O anunciante poderá incluir um máximo de 5 fotos por anuncio. A inclusão de foto será uma por vez. | C12 |
| C15 | O anunciante poderá excluir as fotos adicionadas ao anuncio. | C14 |
| C16 | O anunciante poderá consultar a lista de interessados nos anúncios feitos. | C12, C8 |
| C17 | O anunciante poderá aceitar o interesse do inquilino e poderá analisar o perfil do interessado antes da ação de aceite. | C12, C8 |
| C18 | O Administrador poderá alterar e consultar qualquer usuário. | C7 |
| C19 | O administrado poderá consultar, alterar e excluir o anuncio de qualquer usuário. | C12, C5 |
| C20 | O administrador poderá adicionar e excluir fotos de qualquer usuário. | C14 |

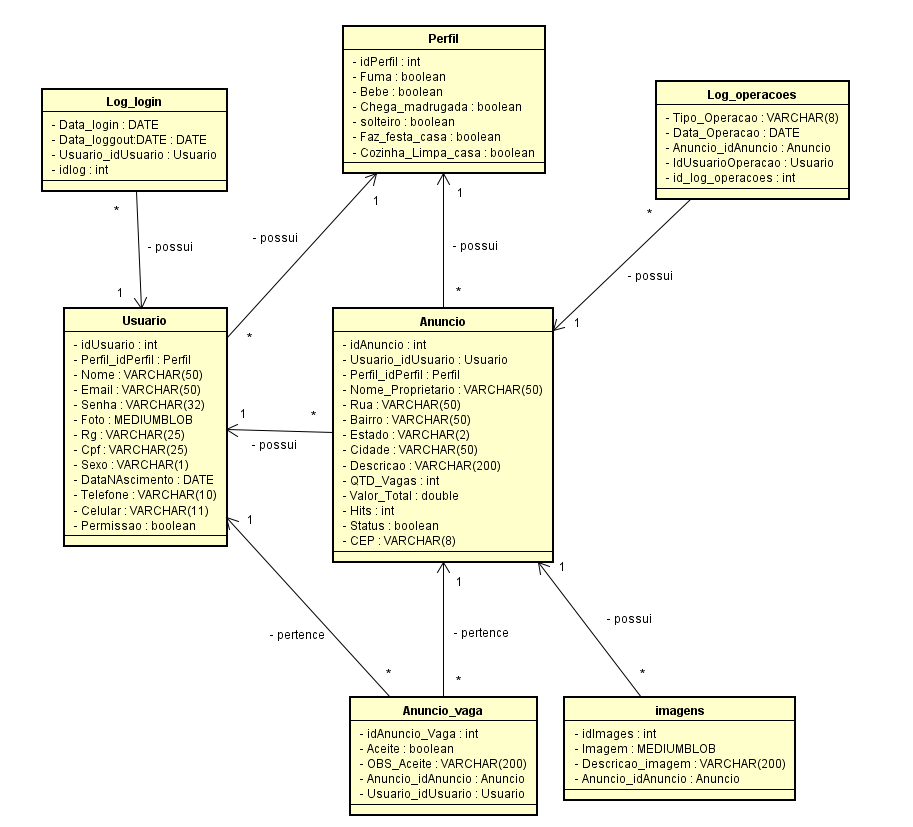
Fonte: Elaborado pelo autor (2017).

# Diagrama de domínio

O diagrama de domínio é amplamente usado como fonte de inspiração para o design dos objetos do *software* e ilustra classes conceituais significativas em um domínio problemático. Este diagrama é o artefato mais importante para ser feito durante uma análise orientada a objetos. (LARMAN, 2001)

O diagrama de domínio foi escolhido para auxiliar a identificação das classes a serem utilizadas durante o desenvolvimento do sistema, dessa forma, é mais fácil e prático imaginar o funcionamento do sistema como um todo e o relacionamento entre das classes.

Figura 1- Diagrama de domínio

 Fonte: Elaborado pelo autor (2017).

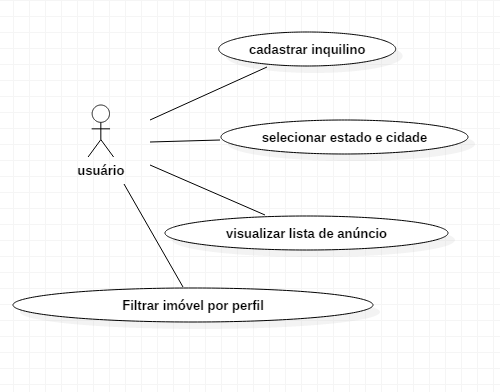
# Diagrama de casos de uso

O diagrama de casos de uso é uma técnica para captar os requisitos funcionais de um sistema. Ele tem como objetivo mostrar as interações mais comuns entre usuário e sistema através de uma narrativa sobre o uso do sistema. (FOWLER,2005)

Nesse projeto será utilizado o diagrama de casos de uso para definir e mostrar ao usuário as funcionalidades do sistema através de uma forma visual e pratica.

## Usuário

No sistema, o usuário é todo aquele que tem acesso ao *site* sem a necessidade de *login*. Esse tipo de cliente poderá visualizar os anúncios e realizar toda a parte de filtro para achar o imóvel que mais lhe adequa. Quando o usuário clicar no botão de mostrar interesse, ele será direcionado para a página de *login*/cadastro. O usuário também poderá fazer o cadastro para ter acesso as demais funcionalidades do site.

Figura 2 - Diagrama de casos de uso (usuário) 

Fonte: Elaborado pelo autor (2017).

Tabela 2 - Casos de Uso Usuário (selecionar estado e cidade)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Selecionar estado e cidade** |
| Ator principal | Usuário |
| Resumo | Ao acessar o *site*, o usuário será direcionado para a *homepage* onde poderá selecionar o estado que deseja verificar os anúncios |
| Pré-Condições | Ter o endereço do site e acesso à *internet* |
| **Fluxo principal** | |
| **Ações do ator** | **Ações do sistema** |
| Selecionar estado e cidade |  |
|  | Gerar uma lista de anúncios do estado e cidade selecionado |
|  | Direcionar o usuário para a página de anúncios |

Fonte: Elaborado pelo autor (2017).

Tabela 3 - Casos de Uso Usuário (Visualizar lista de anúncios)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Visualizar lista de anúncios** |
| Ator principal | Usuário |
| Resumo | Poderá navegar em uma lista e analisar todos os atributos e fotos de um anúncio. É mostrado um anúncio por vez. Ex: Figura 30. |
| Pré-Condições | Ter selecionado o estado |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para carregar mais anúncios |  |
|  | Carregar 1 a 1 os anúncios da lista |

Fonte: Elaborado pelo autor (2017).

Tabela 4- Casos de Uso Usuário (filtrar imóvel por perfil)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Filtrar imóvel por perfil** |
| Ator principal | Usuário |
| Resumo | O usuário poderá filtrar o anúncio por valor, cidade, preço e perfil. |
| Pré-Condições | Ter selecionado o estado e a cidade |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher os filtros |  |
|  | Retornar uma lista com os filtros selecionados pelo usuário |
| **Fluxo alternativo** |  |
| limpar pesquisa |  |
|  | Carregar a lista de anúncios por estado |

Fonte: Elaborado pelo autor (2017).

Tabela 5 - Casos de Uso Usuário (Cadastrar)

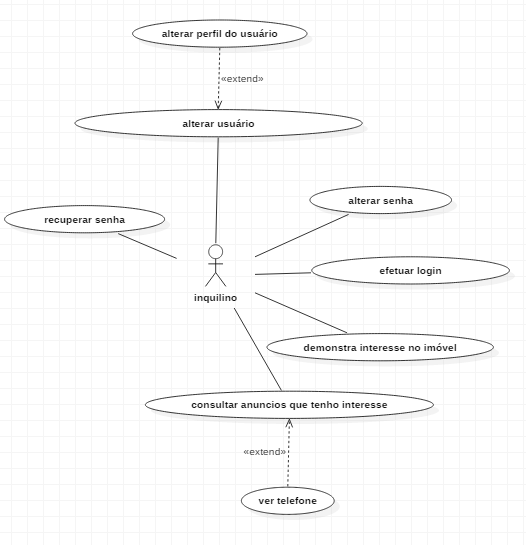
|  |  |
| --- | --- |
| **Nome do caso de uso** | **Cadastrar** |
| Ator principal | Usuário |
| Resumo | O usuário poderá se cadastrar no site para ter acesso a área do inquilino/anunciante |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher os campos para o cadastro |  |
|  | Validar dados do cadastro |
|  | Salvar dados no banco |
|  | Direcionar o usuário para tela de *login* |
| **Fluxo alternativo** |  |
|  | 1 Dados do tipo errado |
|  | 2 Senha não condiz com o campo de autenticação |
|  | 3 Campo nulo |
|  | Em todos os casos o sistema apresentara uma mensagem de erro |

Fonte: Elaborado pelo autor (2017).

## Inquilino

Quando o usuário faz o cadastro e passa ter acesso ao sistema por meio do *login*, ele é considerado um inquilino/usuário. O inquilino poderá alterar os dados do cadastro realizado anteriormente, tais como: nome, rg, cpf, perfil. Em uma opção separada, o inquilino poderá fazer a alteração de senha também. Além dessas duas opções de uso do *software*, o inquilino terá acesso a duas funcionalidades: demonstrar interesse no anúncio e acesso a lista de anúncios que demonstrou interesse. A primeira delas, que é demonstrar interesse, consiste em um botão que aparece junto ao anúncio onde, ao apertar o botão, o inquilino poderá escolher entre a opção sim e não. Se um usuário fizer o mesmo procedimento, ele será direcionado para página de cadastro para que possa se tornar um inquilino. A outra funcionalidade é uma lista de anúncios em que o usuário demonstrou interesse, nesta lista irá conter nome do anunciante, nome do interessado, id do interesse, um campo para o anunciante informar o motivo pelo qual o inquilino foi aceito ou não e um botão para verificar o contato do proprietário imóvel caso seja aceito. Inquilino e usuário estão divido por funcionalidades. Após efetuar o *login*, os usuários possuem o mesmo privilégio de acesso as funções, com exceção do administrador que possui o acesso a área administrativa da aplicação.

Figura 3 -Diagrama de casos de uso (inquilino)



Fonte: Elaborado pelo autor (2017).

Tabela 6 - Casos de Uso Inquilino (*Login*)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Login** |
| Ator principal | Inquilino |
| Resumo | O inquilino poderá fazer o acesso ao sistema |
| Pré-condições | Tem um usuário cadastrado com sucesso |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher o campo de *e-mail* e senha |  |
|  | Valida os dados |
|  | Gera *log* de *login* |
|  | redireciona para pagina de dados pessoais |
| **Fluxo alternativo** |  |
|  | Em caso de *e-mail* e/ou senha incorretas o sistema mostrará uma mensagem de erro |

Fonte: Elaborado pelo autor (2017).

Tabela 7 - Casos de Uso Inquilino (Recuperar senha)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Recuperar Senha** |
| Ator principal | Inquilino |
| Resumo | O inquilino poderá recuperar a senha |
| Pré-condições | Saber CPF e *e-mail* cadastrado |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher CPf e *e-mail* cadastrado |  |
| Colocar nova senha |  |
|  | Verificar se CPF e *e-mail* existem |
|  | Salvar nova senha no banco |
| **Fluxo alternativo** |  |
|  | Caso o *e-mail* e o CPF não sejam encontrados no banco, irá aparecer uma mensagem de erro. |

Fonte: Elaborado pelo autor (2017).

Tabela 8 - Casos de Uso Inquilino (Demonstrar interesse no imóvel)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Demonstrar interesse no imóvel** |
| Ator principal | Inquilino |
| Resumo | O inquilino poderá demonstrar o interesse no imóvel |
| Pré-condições | Ter feito o *login* no sistema |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão presente no corpo do anuncio do imóvel |  |
|  | Seleciona os dados do anuncio e do usuário e grava no banco de dados |

Fonte: Elaborado pelo autor (2017).

Tabela 9 - Casos de Uso Inquilino (Alterar senha)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar senha** |
| Ator principal | Inquilino |
| Resumo | O inquilino e poderá fazer a alteração da senha de *login* ao sistema |
| Pré-condições | Ter feito o *login* no sistema |
| Pré-condições | Ter a senha antiga de *login* |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher o campo de senha antiga e digitar a nova senha em dois campos para ser autenticado |  |
|  | Verifica se a nova senha está correta com o campo de verificação |
|  | Salvar nova senha no banco de dados |
| **Fluxo alternativo** |  |
|  | Mostra mensagem de erro se a senha antiga ou o campo de verificação da senha nova estiverem errados. |

Fonte: Elaborado pelo autor (2017).

Tabela 10 - Casos de Uso Inquilino (Alterar usuário)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar usuário** |
| Ator principal | Inquilino |
| Resumo | O inquilino/anunciante poderá fazer a alteração dos dados cadastrais feito pelo usuário |
| Pré-condições | Efetuar *login* |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher os campos que deseja alterar |  |
|  | Verificar se os dados estão corretos |
|  | Fazer alteração no banco de dados |
| **Fluxo alternativo** |  |
|  | Mostra mensagem de erro se algum campo estiver nulo ou incorretamente preenchido |

Fonte: Elaborado pelo autor (2017).

Tabela 11 - Casos de Uso Inquilino (Alterar perfil usuário)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar perfil usuário** |
| Ator principal | Inquilino |
| Resumo | O inquilino/anunciante poderá fazer a alteração dos dados do perfil |
| Pré-condições | Estar alterando os dados do usuário |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher os campos que deseja alterar |  |
|  | Verificar se os dados estão corretos |
|  | Fazer alteração no banco de dados |
| **Fluxo alternativo** |  |
|  | Mostra mensagem de erro se algum campo estiver nulo ou incorretamente preenchido |

Fonte: Elaborado pelo autor (2017).

Tabela 12 - Casos de Uso Inquilino (Consultar anúncios que tenho interesse)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Consultar anúncios que tenho interesse** |
| Ator principal | Inquilino |
| Resumo | O inquilino terá acesso a uma página onde consultará uma lista de anúncios que demonstrou interesse e, se aceito, poderá visualizar o telefone do anunciante |
| Pré-condições | Efetuar *login* |
| Pré-condições | Demonstrar interesse em 1 ou mais anúncios |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Acessar a página lista de vagas |  |
|  | Listar todos os interesses demonstrados pelo inquilino/anunciante |
| Consultar *status* do interesse |  |
| **Fluxo alternativo** |  |
| Visualizar telefone do imóvel |  |
|  | Mostrar um *dialog* com o telefone cadastrado no anuncio do imóvel |

Fonte: Elaborado pelo autor (2017).

## Anunciante

O anunciante tem acesso a uma lista com todos os interessados nos seus anúncios. Nesta lista ele poderá visualizar o perfil do usuário interessado e comparar com o perfil do anúncio, tomando a decisão se aceita o inquilino ou não. Para fazer o aceite do interesse, o usuário terá dois botões em uma coluna dessa lista onde, uma é para aceitar e outro para recusar e, quando isso for feito ele poderá inserir uma observação informando o motivo da decisão.

Figura 4 -Diagrama de casos de uso (anunciante)

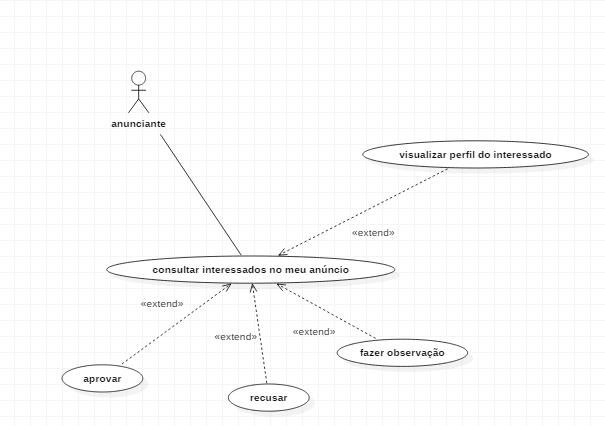
 Fonte: Elaborado pelo autor (2017).

Tabela 13 - Casos de Uso Anunciante (Consultar interessados no meu anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Consultar interessados no meu anuncio** |
| Ator principal | anunciante |
| Resumo | O anunciante poderá consultar os interessados no anuncio e decidir quem aprovar ou não |
| Pré-condições | Efetuar *login* |
| Pré-condições | Ter 1 ou mais interessado (os) em um imóvel anunciado |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Acessar a página lista de vagas |  |
|  | Listar todos os interessados em um ou mais imóveis anunciados |
| Aceitar ou recusar interesse |  |
|  | Mostrar um *dialog* com as opções de: aceitar, recusar e fazer observação |
| Aceitar ou recusar e preencher campo comentário se necessário |  |
|  | Salvar dados no banco |
| **Fluxo alternativo** |  |
| Visualizar perfil do interessado |  |
|  | Mostrar um *dialog* comparando o perfil do interessado com o perfil do imóvel |

Fonte: Elaborado pelo autor (2017).

Oanunciante poderá incluir anúncios no sistema onde poderá preencher os campos como endereço, valor total e quantidade de vagas disponíveis. O anunciante também poderá incluir fotos nesse anúncio totalizando um máximo de cinco. O anunciante terá acesso a uma página onde conseguirá fazer o gerenciamento dos anúncios feitos. Nessa página o anunciante poderá excluir, alterar o anuncio e o perfil do anuncio além de fazer a adição e a exclusão das fotos pertencentes a cada anúncio. Para o gerenciamento das fotos, o anunciante terá acesso a uma lista separada que contém todas as fotos de um anúncio em específico, tornando mais fácil inclusão e/ou alteração dessas fotos.

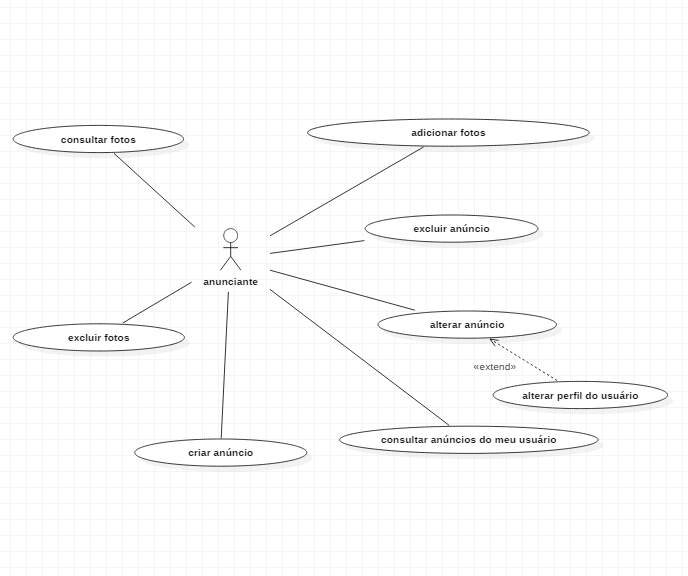
Figura 5 - Diagrama de casos de uso (anunciante parte 2)  Fonte: Elaborado pelo autor (2017).

Tabela 14 - Casos de Uso Anunciante (Criar anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Criar anuncio** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá incluir um anuncio no sistema |
| Pré-condições | Efetuar *login* |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preencher os campos referente ao anuncio |  |
|  | Validar os dados |
|  | Salvar os dados no banco |
| **Fluxo alternativo** |  |
|  | Mostrar mensagem de erro caso um campo esteja vazio ou preenchido incorretamente |

Fonte: Elaborado pelo autor (2017).

Tabela 15 - Casos de Uso Anunciante (Consultar anúncios do meu usuário)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Consultar anúncios do meu usuário** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá consultar os anúncios feitos |
| Pré-condições | Efetuar *login* e ter 1 ou mais anúncio(s) feito(s) |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Acessar pagina “meus anúncios “ |  |
|  | Carregar uma lista com os anúncios do anunciante |

Fonte: Elaborado pelo autor (2017).

Tabela 16 - Casos de Uso Anunciante (Excluir anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Excluir anúncio** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá excluir um anuncio no sistema |
| Pré-condições | Ter feito o caso de uso “consultar anúncios do meu usuário” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão de excluir |  |
|  | Mostrar um *dialog* para confirmar ou não a exclusão |
| Selecionar opção |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 17 - Casos de Uso Anunciante (Alterar anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Al0terar anúncio** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá alterar um anuncio no sistema |
| Pré-condições | Ter feito o caso de uso “consultar anúncios do meu usuário” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão alterar |  |
|  | Mostrar um *dialog* com os campos do anúncio |
| Preencher os campos à serem alterados |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 18 - Casos de Uso Anunciante (alterar perfil do anuncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar perfil do anúncio** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá alterar o perfil do anúncio no sistema |
| Pré-condições | Ter feito o caso de uso “consultar anúncios do meu usuário” e estar alterando um anúncio |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preenche os campos à serem alterados |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 19 - Casos de Uso Anunciante (Adicionar fotos)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Adicionar fotos** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá adicionar fotos ao anúncio |
| Pré-condições | Ter feito o caso de uso “consultar anúncios do meu usuário” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para adicionar fotos por anúncio |  |
|  | mostrar *dialog* para inserir comentário e a foto |
| Clicar no botão para selecionar foto |  |
|  | Enviar foto, comentário e salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 20 - Casos de Uso Anunciante (Consultar fotos)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Consultar fotos** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá consultar as fotos do anúncio |
| Pré-condições | Ter fotos adicionadas |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para mostrar fotos por anúncio |  |
|  | Carregar uma lista com as fotos do anúncio |

Fonte: Elaborado pelo autor (2017).

Tabela 21 - Casos de Uso Anunciante (Excluir fotos)

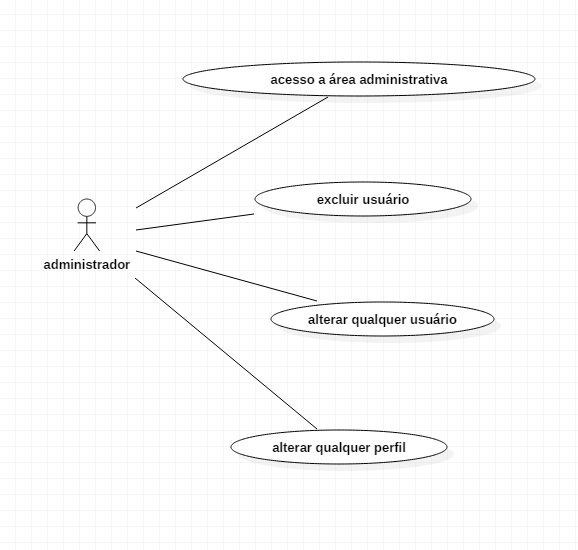
|  |  |
| --- | --- |
| **Nome do caso de uso** | **Excluir fotos** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá excluir fotos do anúncio |
| Pré-condições | Ter feito o caso de uso “consultar fotos” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para excluir a foto desejada |  |
|  | Mostrar um *dialog* para confirmar ação |
| decidir se exclui a foto ou não |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017)

## Administrador

O administrador do sistema será o primeiro usuário cadastrado. Após ter um administrador definido, ele poderá alterar a permissão das contas de qualquer usuário do sistema, tornando possível que tenha mais de um administrador. Como dito anteriormente, o administrador pode alterar qualquer usuário e até efetuar a exclusão se necessário. Essas ações poderão ser feitas por botões localizados em uma coluna na lista de usuários, sendo um botão para alteração e um para exclusão e outro para atualizar uma lista de anúncios referentes ao usuário selecionado, essa lista fica localizada logo abaixo da lista de usuários.

Figura 6 - Diagrama de casos de uso (administrador)



Fonte: Elaborado pelo autor (2017).

Tabela 22- Casos de Uso Administrador (Acessar área administrativa)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Acessar área administrativa** |
| Ator principal | Administrador |
| Resumo | O administrador poderá acessar a área administrativa |
| Pré-condições | Ser administrador e efetuar *login* |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
|  | Carregar uma lista com todos os usuários do site |

Fonte: Elaborado pelo autor (2017).

Tabela 23 - Casos de Uso Administrador (Alterar qualquer usuário)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar qualquer usuário** |
| Ator principal | Administrador |
| Resumo | O administrador poderá alterar qualquer usuário do sistema |
| Pré-condições | Ter feito o caso de uso “acesso à área administrativa” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão alterar |  |
|  | Mostrar um *dialog* com os campos do usuário |
| Preencher os campos à serem alterados |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 24 - Casos de Uso Administrador (Alterar perfil)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar perfil** |
| Ator principal | Administrador |
| Resumo | O administrador poderá alterar o perfil de qualquer usuário |
| Pré-condições | Ter feito o caso de uso “acesso a área administrativa” e estar alterando um usuário |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preenche os campos à serem alterados |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 25 - Casos de Uso Administrador (Excluir usuário)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Excluir usuário** |
| Ator principal | Administrador |
| Resumo | O administrador poderá excluir qualquer usuário |
| Pré-condições | Ter feito o caso de uso “acesso a área administrativa” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão de exclusão |  |
|  | Mostrar um *dialog* com a opção de excluir ou não |
| Confirmar procedimento |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Na lista de anúncios o administrador poderá fazer a alteração de qualquer dado do anúncio e perfil do anúncio além de excluir caso necessário. As fotos serão incluídas através de um botão localizado em uma coluna na lista de anúncios. A exclusão da foto será feita na lista de fotos referentes à um anuncio, o botão estará localizado na coluna da lista.

Figura 7 - Diagrama de casos de uso (administrador2)

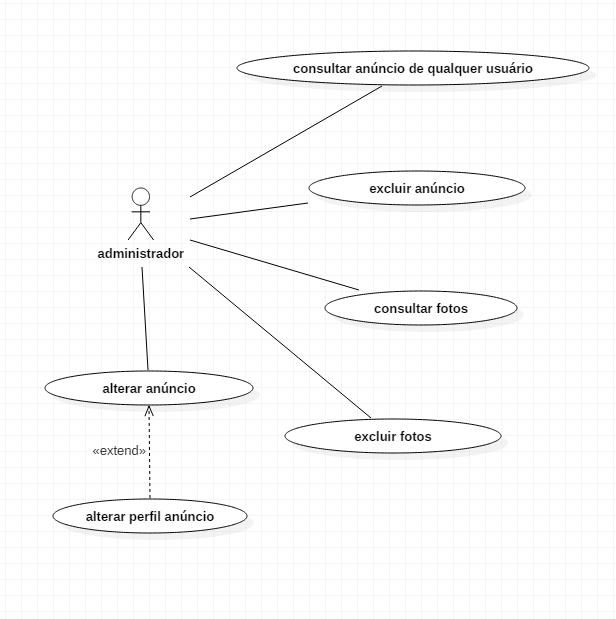
Fonte: Elaborado pelo autor (2017).

Tabela 26 - Casos de Uso Administrador (Consultar anúncio qualquer usuário)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Consultar anúncio de qualquer usuário** |
| Ator principal | Administrador |
| Resumo | O administrador poderá consultar qualquer anúncio de qualquer usuário |
| Pré-condições | Ter feito o caso de uso “acesso a área administrativa” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para consultar os anúncios do usuário |  |
|  | Carregar uma lista com os anúncios do usuário selecionado |

Fonte: Elaborado pelo autor (2017).

Tabela 27 - Casos de Uso Administrador (alterar anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar anúncio** |
| Ator principal | Administrador |
| Resumo | O administrador poderá alterar um anuncio no sistema |
| Pré-condições | Ter feito o caso de uso “consultar anúncio de qualquer usuário” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão alterar |  |
|  | Mostrar um *dialog* com os campos do anuncio |
| Preencher os campos à serem alterados |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 28 - Casos de Uso Administrador (Alterar perfil do anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Alterar perfil do anúncio** |
| Ator principal | Administrador |
| Resumo | O administrador poderá alterar o perfil do anuncio no sistema |
| Pré-condições | Ter feito o caso de uso “consultar anúncio de qualquer usuário” e estar alterando um anúncio |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Preenche os campos à serem alterados |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 29 - Casos de Uso Administrador (Excluir anúncio)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Excluir anúncio** |
| Ator principal | Administrador |
| Resumo | O administrador poderá excluir um anúncio no sistema |
| Pré-condições | Ter feito o caso de uso “consultar anúncio de qualquer usuário” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão de excluir |  |
|  | Mostrar um *dialog* para confirmar ou não a exclusão |
| Selecionar opção |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 30 - Casos de Uso Administrador (Adicionar fotos)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Adicionar fotos** |
| Ator principal | Administrador |
| Resumo | O administrador poderá adicionar fotos ao anúncio criado por ele mesmo ou por um anunciante. |
| Pré-condições | Ter feito o caso de uso “consultar anúncio de qualquer usuário” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para adicionar fotos por anúncio |  |
|  | Carregar um dialog para inserir comentário e inserir foto. |
| Clicar no botão para selecionar foto |  |
|  | Enviar foto, comentário e salvar no banco |

Fonte: Elaborado pelo autor (2017).

Tabela 31 - Casos de Uso Administrador (Consultar fotos)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Consultar fotos** |
| Ator principal | Anunciante |
| Resumo | O anunciante poderá consultar as fotos do anúncio |
| Pré-condições | Ter fotos adicionadas |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para mostrar fotos por anúncio |  |
|  | Carregar uma lista com as fotos do anúncio |

Fonte: Elaborado pelo autor (2017).

Tabela 32 - Casos de Uso Administrador (Excluir fotos)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Excluir fotos** |
| Ator principal | Administrador |
| Resumo | O administrador poderá excluir fotos do anúncio |
| Pré-condições | Ter feito o caso de uso “consultar fotos” |
| **Fluxo principal** |  |
| **Ações do ator** | **Ações do sistema** |
| Clicar no botão para excluir a foto desejada |  |
|  | Mostrar um *dialog* para confirmar ação |
| decidir se exclui a foto ou não |  |
|  | Salvar no banco |

Fonte: Elaborado pelo autor (2017).

# Tecnologias usadas

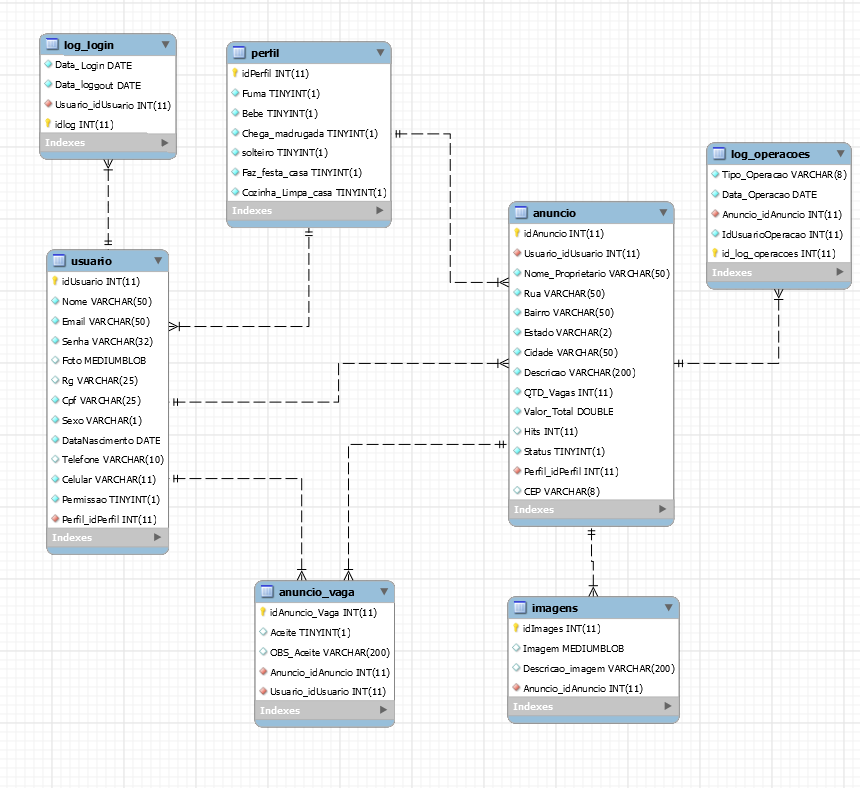
Nesta seção serão abordadas as tecnologias utilizadas para o desenvolvimento deste projeto. Para o desenvolvimento do projeto de banco de dados foi utilizado o *Mysql* como banco de dados e a ferramenta *Mysql* Workbench para construção do banco de dados e diagramas de forma visual. Para fazer a persistência de um objeto em um banco de dados relacional foi utilizado o *framework* *Hibernate*, que de forma automática, faz esse mapeamento de um Objeto para um banco de dados relacional. Para desenvolvimento web foi utilizado *Bootstrap*, JSF e *Primefaces* como *frameworks*. Como IDE para programação em JAVA foi utilizado o *Netbeans* e todos os *frameworks* citados foram incorporados em um projeto dentro desta IDE. Desta forma, após configurar todo o ambiente, é necessário usar apenas o Workbench e o *Netbeans* para desenvolver e alterar qualquer parte do projeto.

## Mysql Workbench

O *Mysql* *workbench* é uma ferramenta para gerenciamento do banco de dados *mysql* de forma visual mantido pela empresa *Oracle*. Além de gerenciar bancos de dados *MySql* a ferramenta *Workbench* oferece funcionalidades que auxiliam a criação das tabelas do sistema a ser desenvolvido.

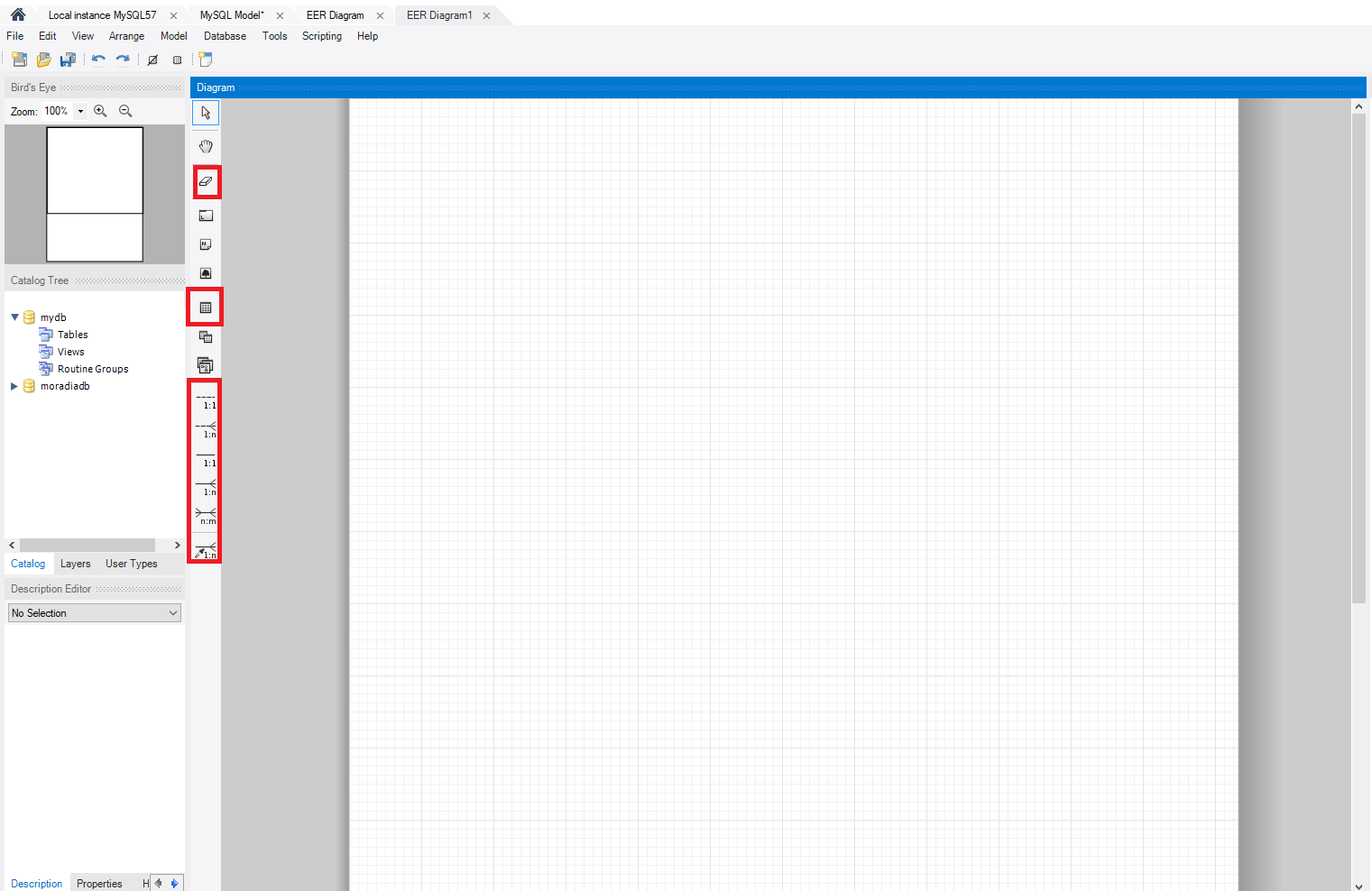
Neste projeto foi utilizado a função de *EER Diagrams (enhanced entity–relationship Diagrams),* que em tradução livre, significa ‘relacionamento de entidade aprimorada’. O *EER Diagram* é um diagrama de alto nível que incorpora extensões ao relacionamento de entidades, a Figura 8 apresenta o EER do *software* proposto.

Figura 8 - EER Diagram Mysql workbench concluído

 Fonte: Elaborado pelo autor (2017).

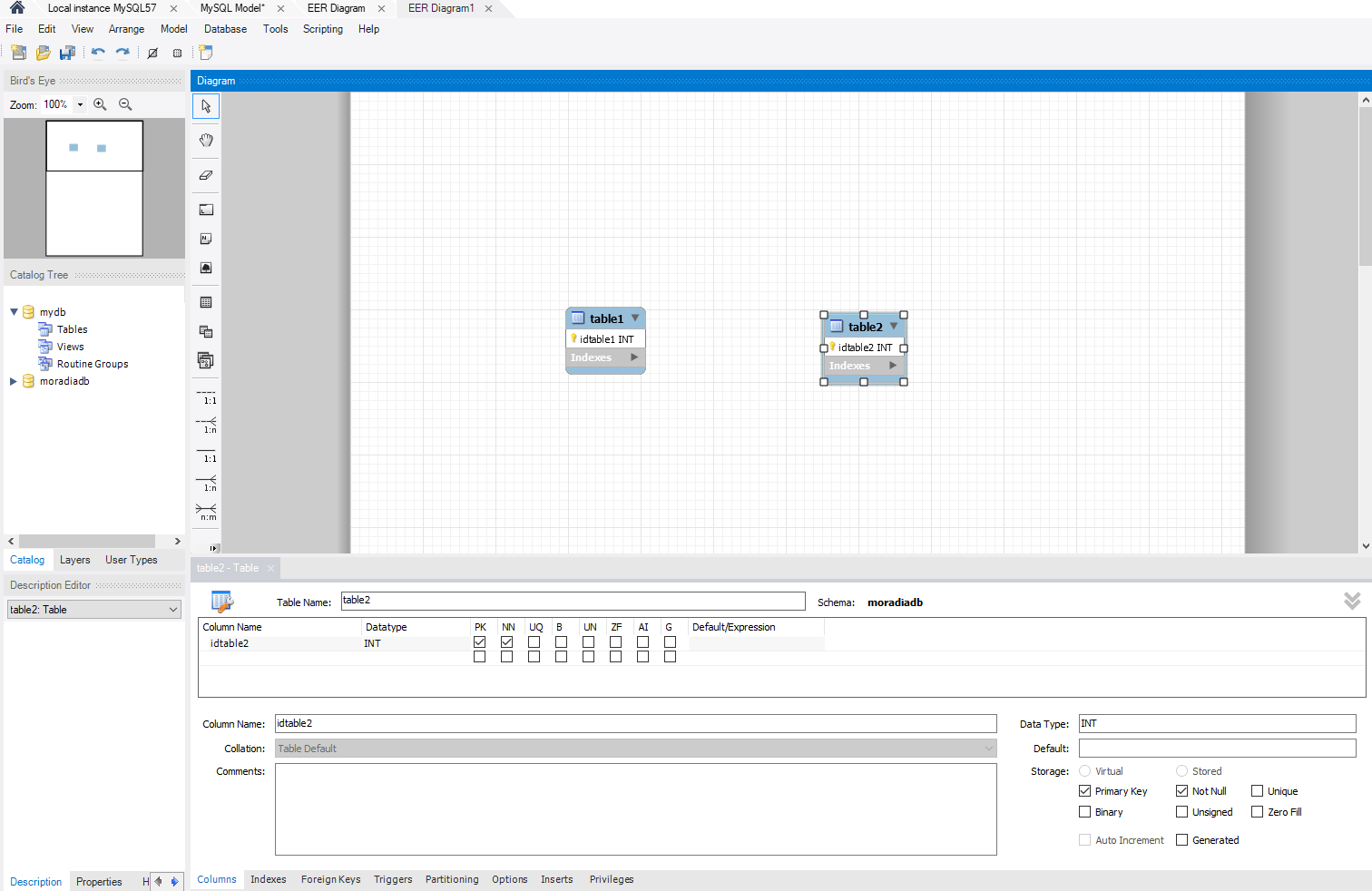
Nessa ferramenta podemos modelar nosso banco de uma forma visual, ficando mais prático e rápido para criar a estrutura de dados necessária para o *software*. Na Figura 9 é possível ver quais são as ferramentas (circuladas em vermelho) que foram usadas para gerar o diagrama anterior:

Figura 9 - Diagrama Mysql workbench (ferramentas)

 Fonte: Elaborado pelo autor (2017).

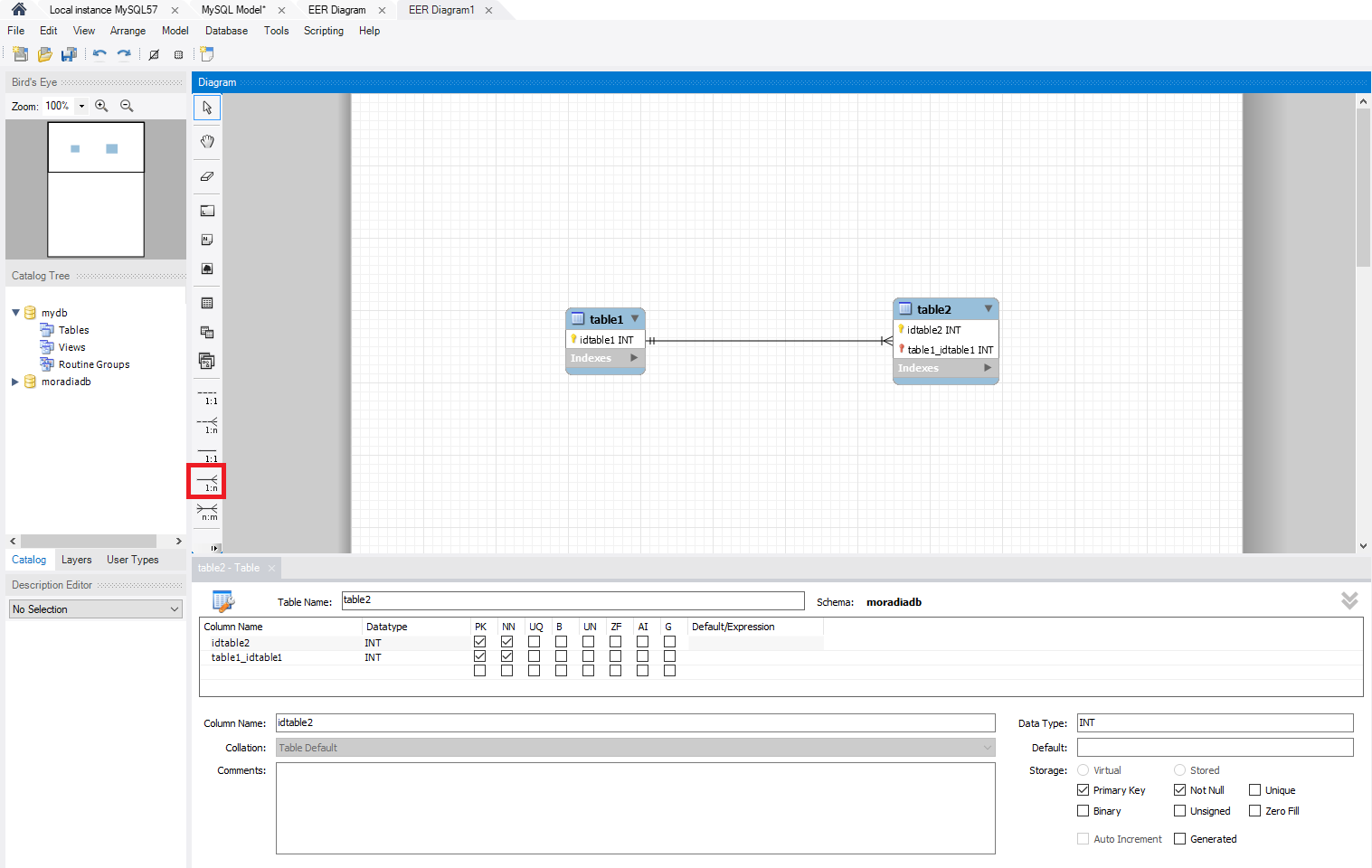
Após selecionarmos a ferramenta de tabelas é possível criar quantas forem necessárias e configurar os campos da tabela da seguinte forma:

Figura 10 - Diagrama Mysql workbench (criando tabelas e colunas)

 Fonte: Elaborado pelo autor (2017).

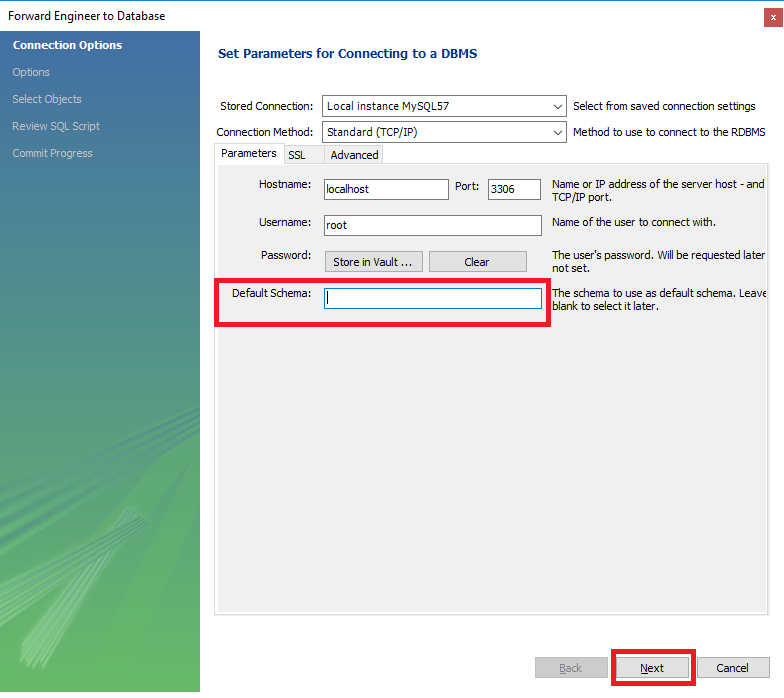
Após fazer o procedimento anterior com todas as tabelas do *software* é necessário escolher o tipo de relacionamento entre as tabelas, para o exemplo utilizaremos a ligação 1:n, onde a tabela 2 recebe a chave estrangeira. Basta programador selecionar o tipo de ligação, selecionar a tabela principal que irá receber a chave estrangeira e clicar na outra tabela que será relacionada, conforme pode ser visualizada na Figura 11.

Figura 11 - Diagrama Mysql workbench (relacionando tabelas)

 Fonte: Elaborado pelo autor (2017).

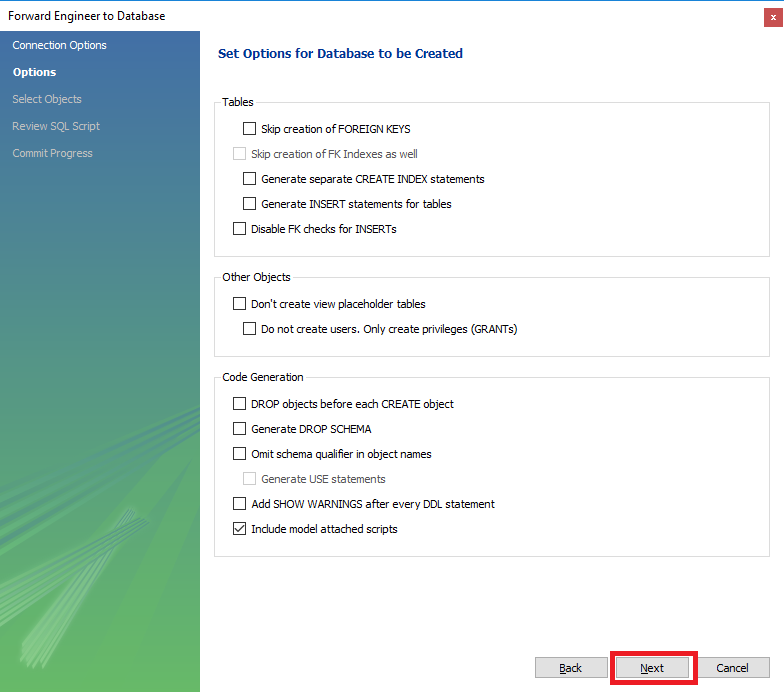
Depois que toda configuração do diagrama for finalizada, podemos criar as nossas tabelas no banco, para isso devemos ir *Database* > *Forward Engineer* (localizado no menu superior do *workbench* ou apertar o atalho “CTR+G “. Após executar um dos comando anteriores, aparecerá uma janela onde colocaremos o banco de dados que desejamos criar as tabelas:

Figura 12 - Mysql workbench foward engineer - conexão com bando de dados

 Fonte: Elaborado pelo autor (2017).

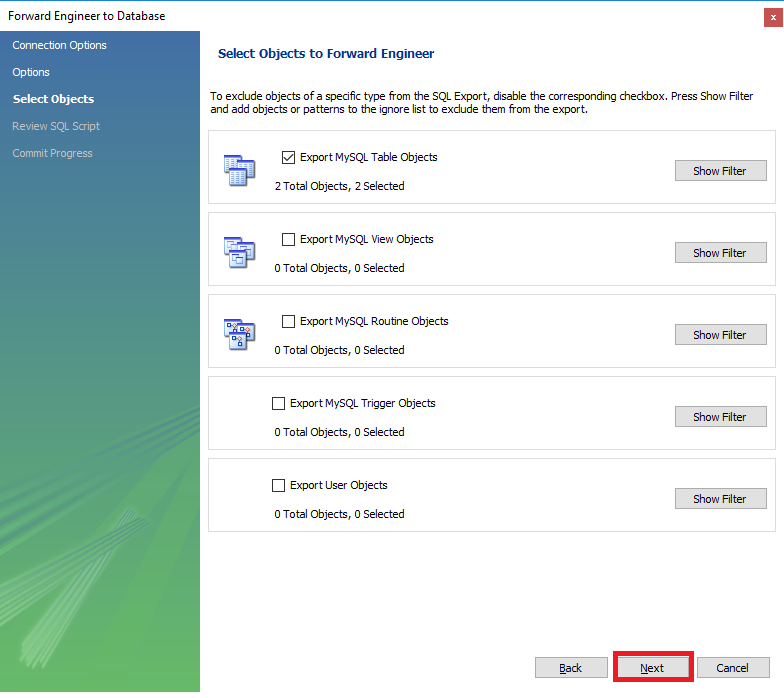
Depois de indicar o banco de dados, clicamos em *next* para escolher algumas opções para a criação das tabelas, selecione o que for ser usado e clieque em next novamente.

Figura 13 - Mysql workbench foward engineer - opções de criação de banco

 Fonte: Elaborado pelo autor (2017).

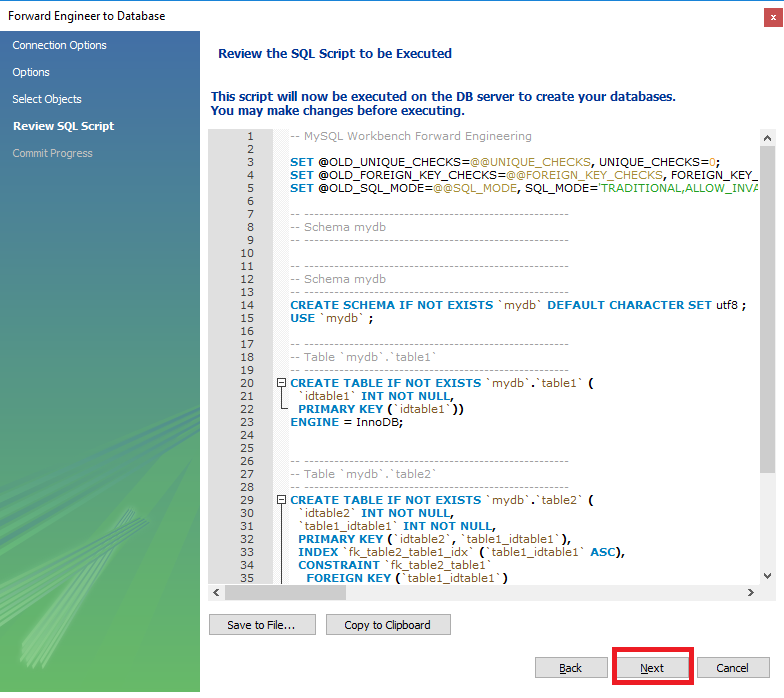
Agora selecionamos os objetos criados para serem salvos como tabela no banco e clicamos em *next*.

Figura 14 - Mysql workbench foward engineer - seleção de objetos

 Fonte: Elaborado pelo autor (2017).

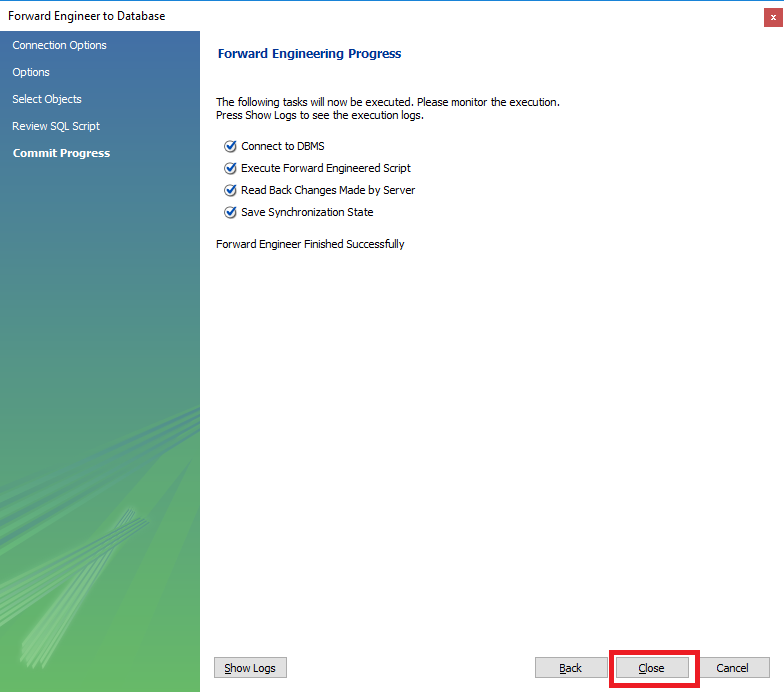
Nesta etapa podemos copiar o código executado pelo *mysql*, dessa forma não será necessário o uso do *workbench* para criar as tabelas do nosso sistema futuramente:

Figura 15 - Mysql workbench foward engineer - código sql

 Fonte: Elaborado pelo autor (2017).

No final, temos essa tela que indica o sucesso da criação de tabelas:

Figura 16 - Mysql workbench foward engineer – conclusão

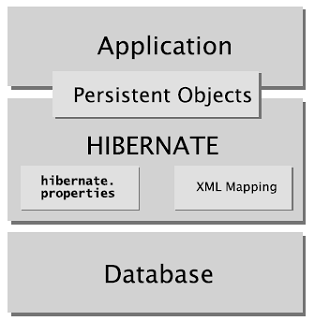
 Fonte: Elaborado pelo autor (2017).

## Hibernate

O *Hibernate* é um *framework* que faz o papel de um ORM (*object-relational mapping*), sendo que ORM é uma técnica para conversão de dados entre objetos de linguagens de programação orientadas a objetos e banco de dados relacionais. Em um banco de dados, as tabelas são representadas por tabelas e as colunas são os tipos das variáveis. Uma tabela pode se relacionar com outras criando diversos relacionamentos (FARIA, 2015).

O *Hibernate* é um *framework* de persistência de dados que auxilia o desenvolvedor a mapear tabelas no banco de dados e transforma-las em classes. As linhas da tabela se referem a um objeto, as colunas aos atributos e a chave estrangeira à associação. Dessa forma o desenvolvedor ganha agilidade por não fazer a conversão diretamente. Podemos verificar na Figura 17 como o *Hibernate* se comporta em uma aplicação:

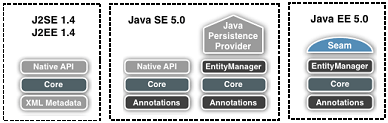
Figura 17 - Camadas hibernate



Fonte: Devmedia (2017).

O *Hibernate* surgiu em 2004, desenvolvido por Gavin King e utiliza arquitetura de pacotes Java, a Figura 18 apresenta a estrutura de pacotes do *Hibernate*.

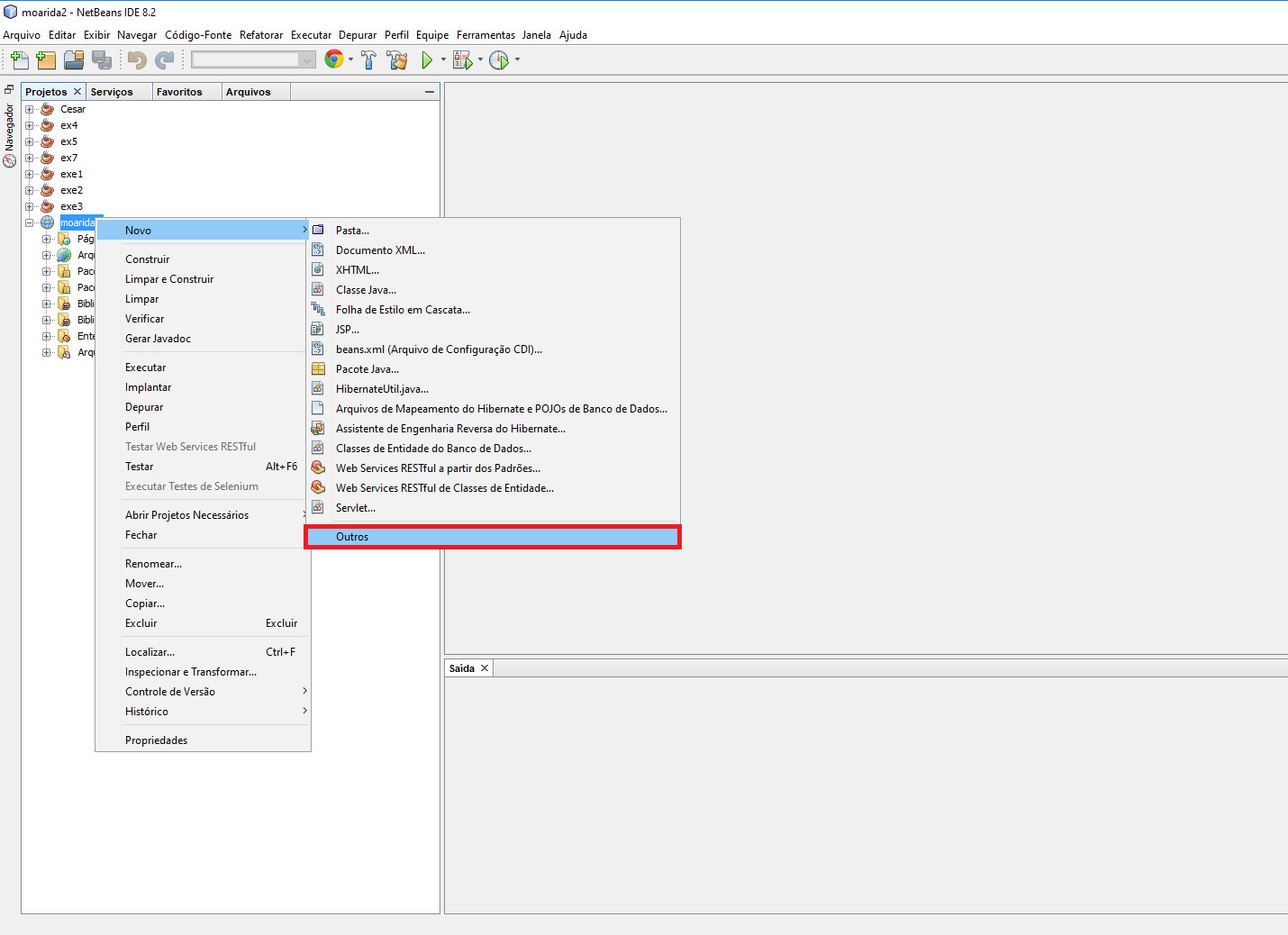
Figura 18 - Pacotes do Hibernate



Fonte: Devmedia (2017).

Para configurar o *Hibernate* no projeto devemos clicar com o botão direito em cima do projeto, selecionar a opção ‘novo’ e depois a opção ‘outros’

Figura 19 - Opções do projeto

 Fonte: Elaborado pelo autor (2017).

Após selecionar a opção ‘outros’ iremos para a parte de categorias e selecionaremos a pasta ‘*Hibernate’*. Após feito esses procedimentos, iremos usar as 4 opções selecionadas em vermelho (Figura 20).

A primeira opção é para criar o arquivo de configuração do *Hibernate*, nomeado de hibernate.cfg.xml. Neste arquivo (figura 22) podemos especificar o mapeamento de objetos e as propriedades da conexão com o banco de dados.

Na segunda opção, mapeamos as tabelas de um banco de dados relacional para fazer a engenharia reversa, que na IDE *Netbeans* é um mapeamento de *Pojo* (Plain Old Java Objects). Os *Pojos* são classes Java simples que não dependem de herança de interfaces ou de *frameworks*. Após o mapeamento, tudo é referenciado no arquivo hibernate.revenge.xml (figura 23).

Depois de criado esses 2 arquivos, selecionamos a terceira opção que criará as classes java (figura 25) e os mapeamentos (figura 24) necessário para o *Hibernate* funcionar, esses mapeamentos podem ser feitos manualmente seguindo o modelo da figura 24. Podemos visualizar os arquivos criados na figura 21.

Também podemos fazer o caminho inverso e criar as tabelas no banco de dados, para isso basta colocar o comando “<property name="hibernate.hbm2ddl.auto">create</property> “ no hibernate.cfg.xml (figura 22), os pré-requisitos para criar o banco dessa forma são as classes e o mapeamento de objetos.

Concluído os processos anteriores, basta criar a classe hibernateutil.java que é responsável por referenciar todos os nossos parâmetros criados para a comunicação com o banco de dados. Depois de criar a classe, podemos pegar o código gerado (figura 26) e colocar em uma classe genérica, que é o método utilizado nesse sistema.

Figura 20- Arquivos hibernate

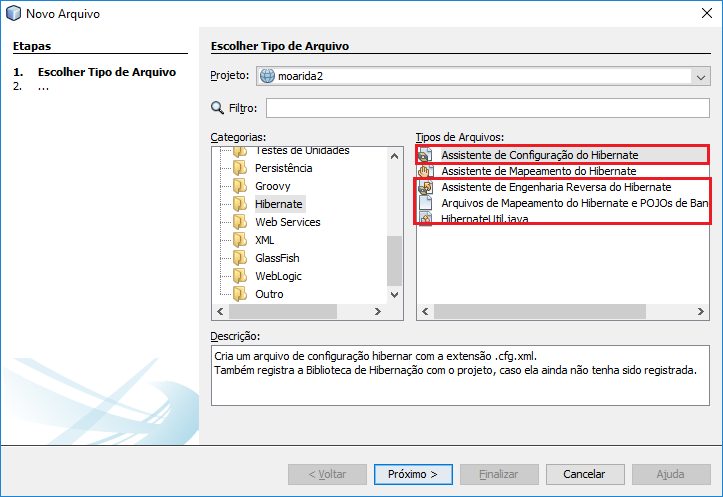
 Fonte: Elaborado pelo autor (2017).

Figura 21 - Arquivos criados Hibernate

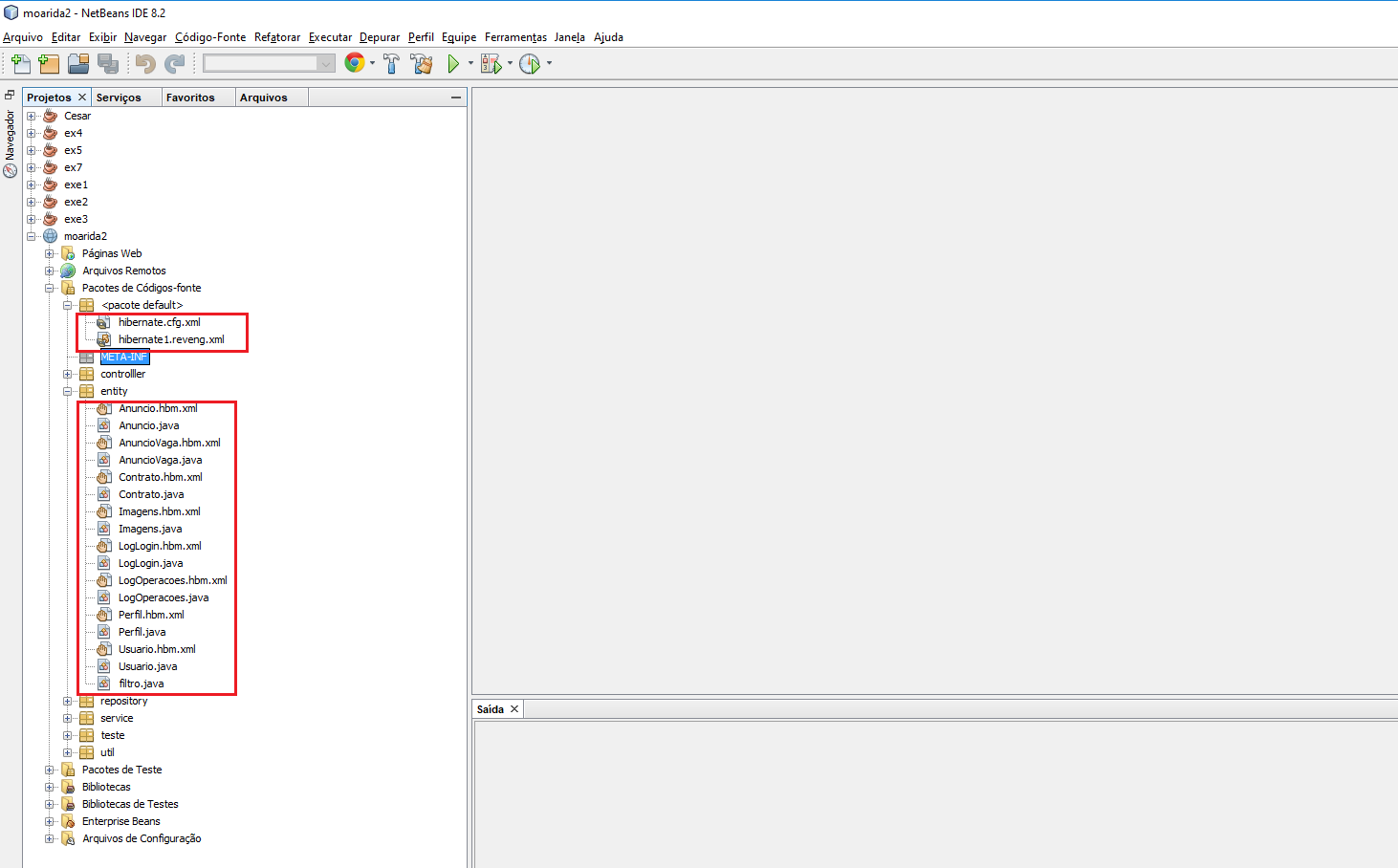
 Fonte: Elaborado pelo autor (2017).

Figura 22 - Hibernate.cfg.xml

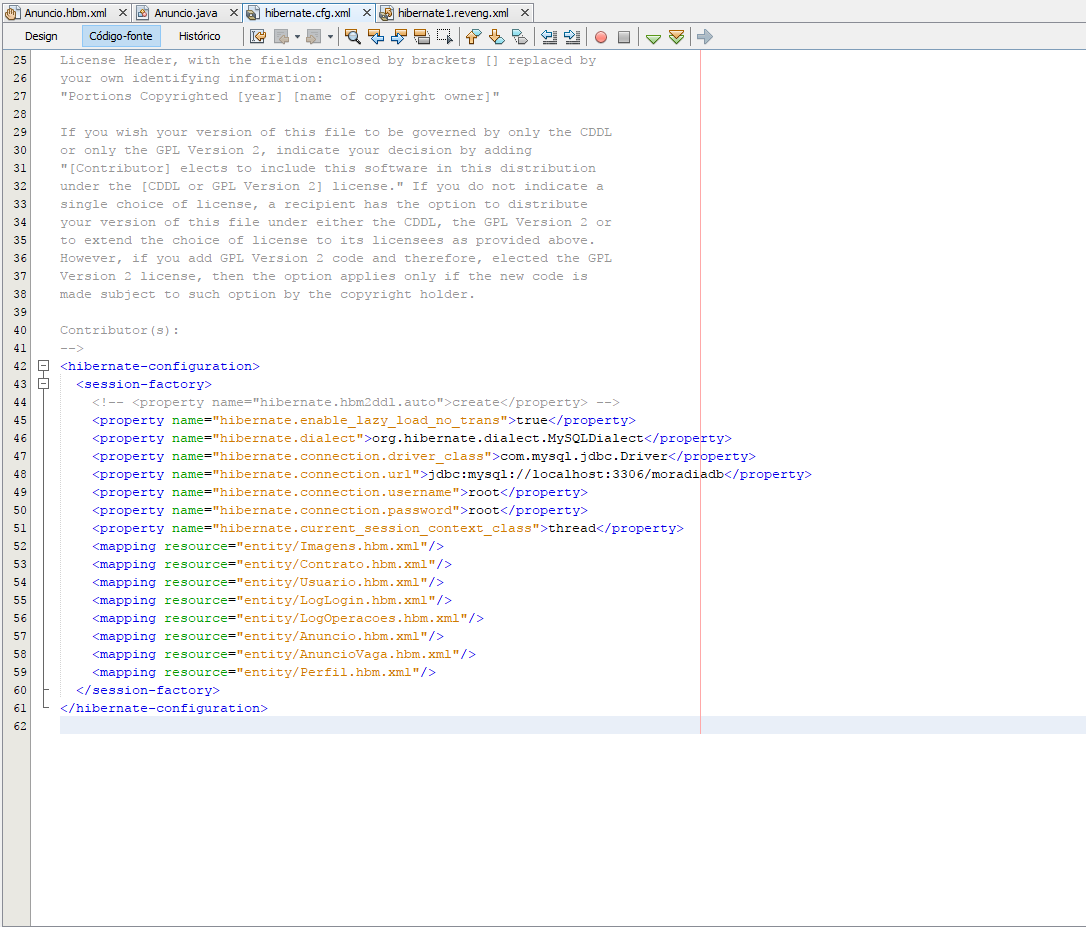
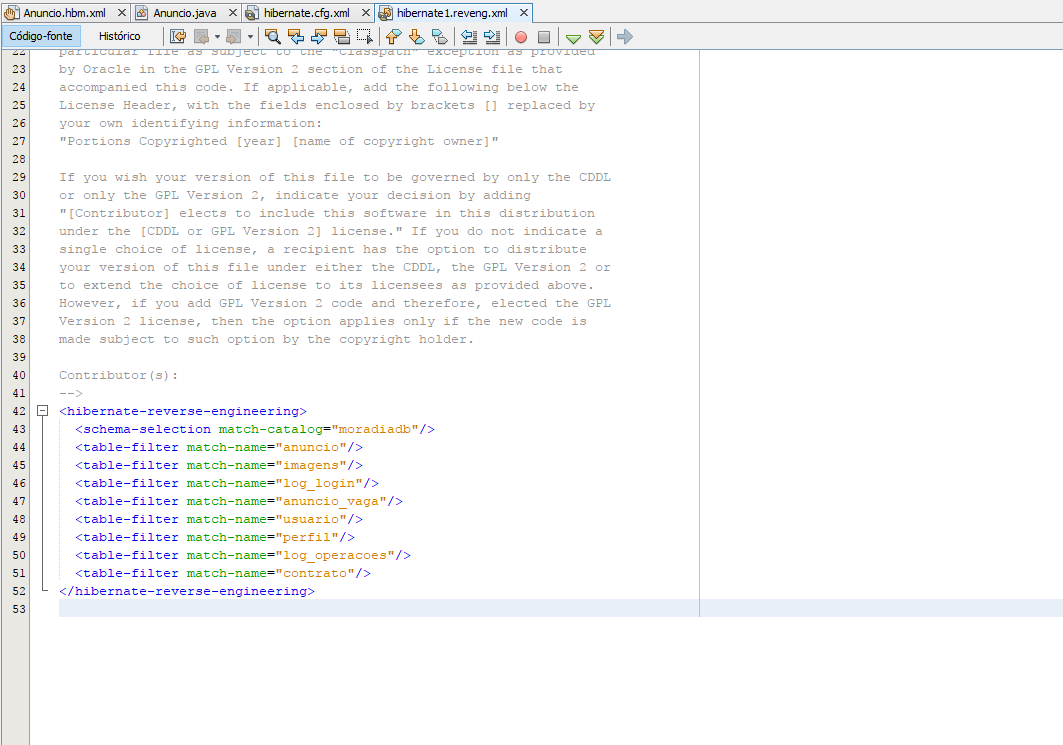
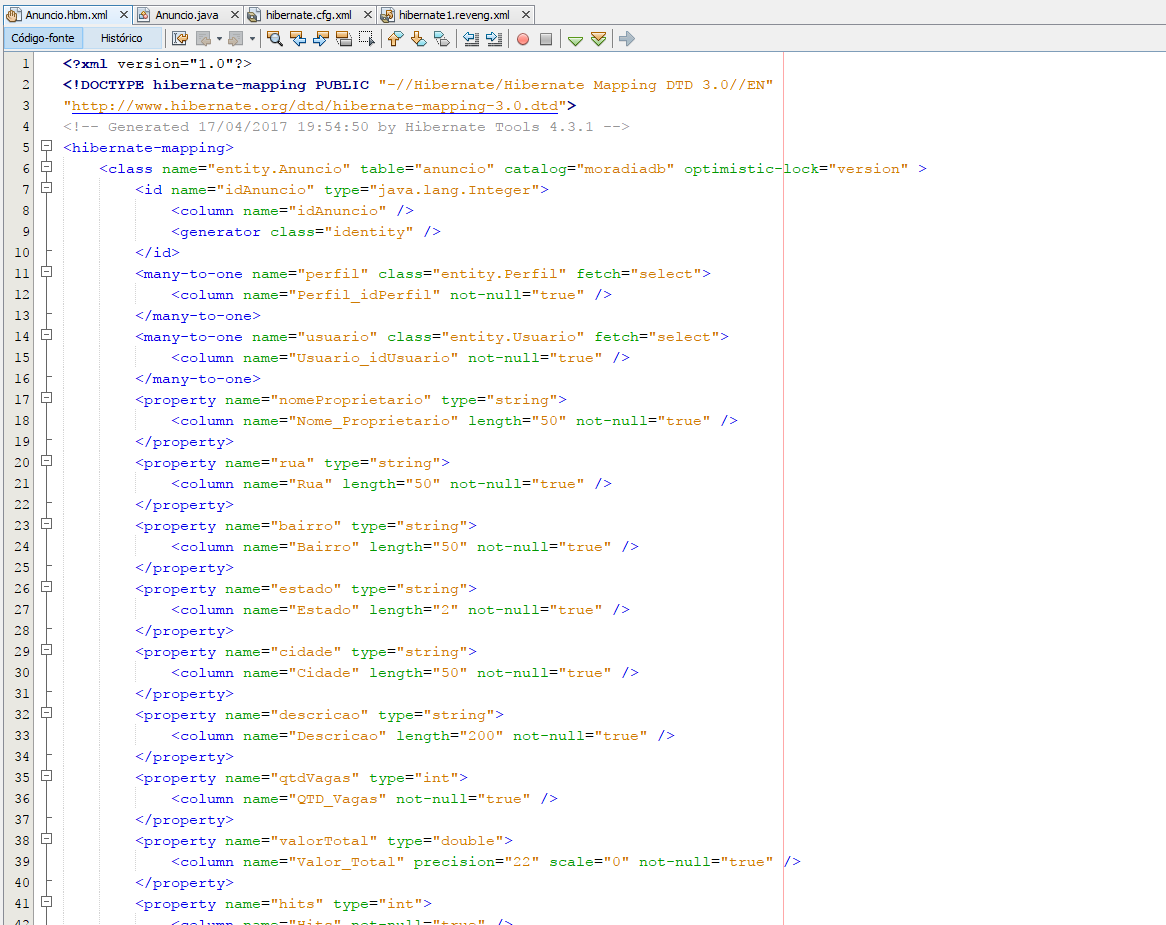
 Fonte: Elaborado pelo autor (2017).

Figura 23 - Hibernate.revenge.xml



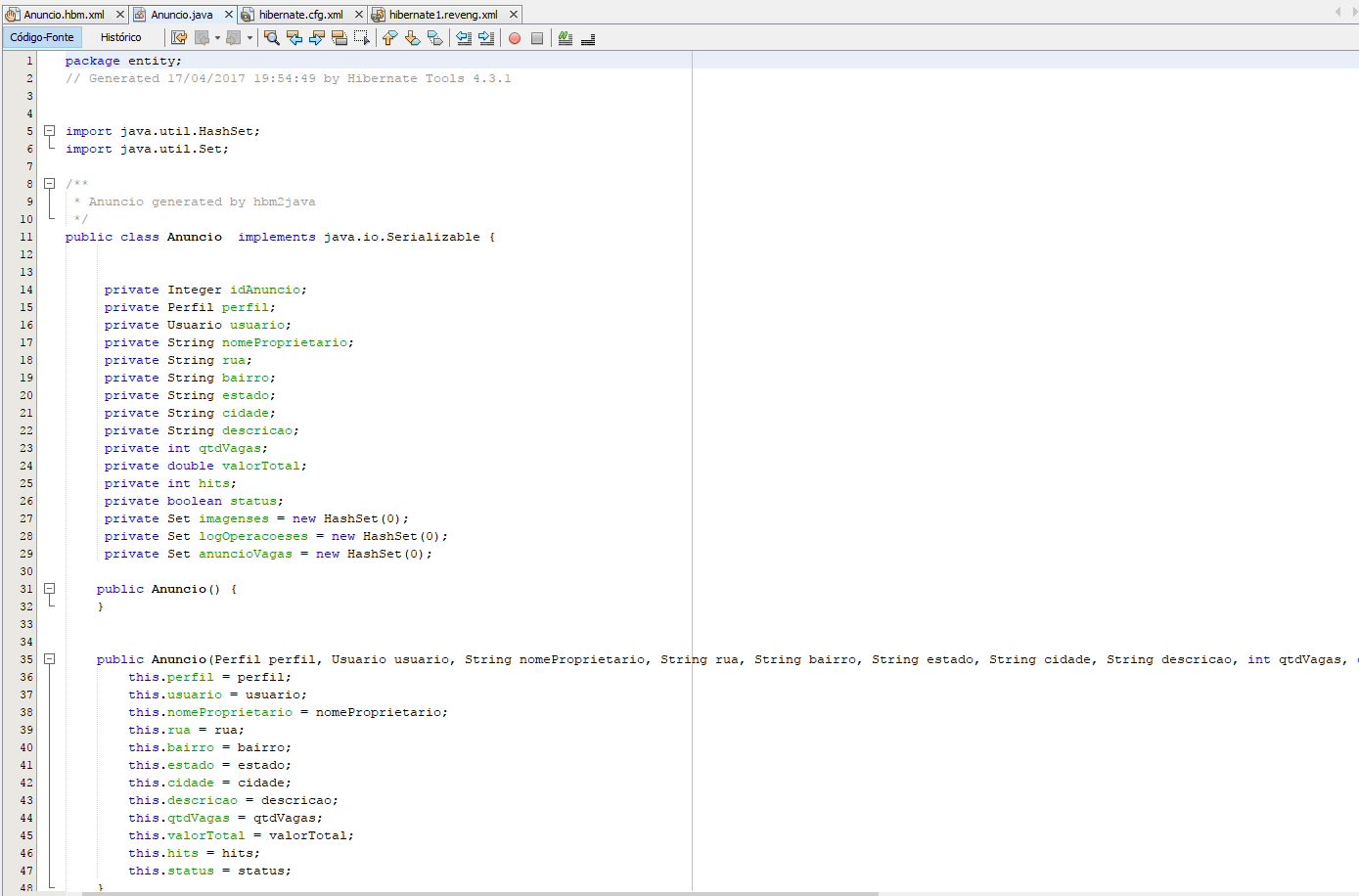
Fonte: Elaborado pelo autor (2017).

Figura 24 - Classe de mapeamento Hibernate (Anuncio)



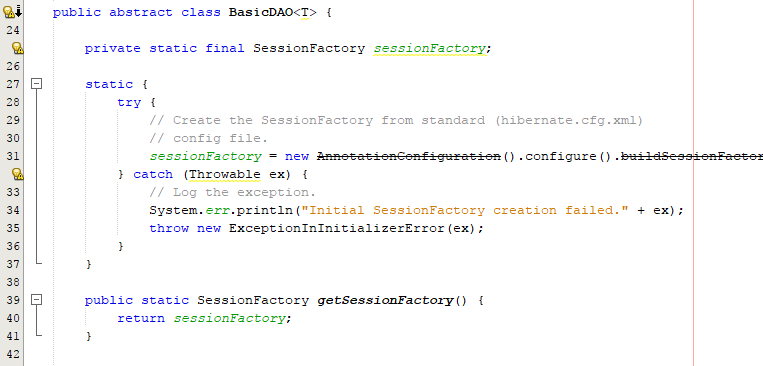
Fonte: Elaborado pelo autor (2017).

Figura 25 - Classe Java – Engenharia reversa Hibernate (Anuncio)



Fonte: Elaborado pelo autor (2017).

Figura 26 - Código hibernateutil



Fonte: Elaborado pelo autor (2017).

## JSF

JSF (*Java Server Faces*), é uma tecnologia de desenvolvimento *web* que utiliza um modelo de interfaces gráficas orientada a eventos. Definido pela JCP (*Java Community Process*) o JSF se torna um padrão de desenvolvimento que facilita os fornecedores de ferramentas que, ao criarem produtos personalizados, ofereçam produtividade no desenvolvimento de interfaces (FARIA,2013).

O JSF é baseado no padrão MVC (*Model View Controller*) que torna o desenvolvimento do *software* mais simples. O MVC divide o programa em 3 partes (modelo, visualização e controle). O modelo é a parte responsável por apresentar os objetos de negócio, manter o estado de aplicação e fornecer controle sobe os dados. A visualização é a parte responsável pela interface do usuário e define como os dados serão mostrados. A camada de visualização comunica as ações do usuário com o controle. A camada de controle é responsável por ligar o modelo e a visualização, onde interpreta as requisições do usuário para que as mudanças no sistema ocorram (FARIA,2013).

O JSF é um *framework* de componentes *web* que auxilia o programador na hora de ligar os objetos JAVA com a interface *web*. Além do *design* flexível, o JSF proporciona agilidade para os programadores na comunicação entre a interface e o controle da aplicação. Os componentes do JSF podem ser customizados de acordo com a necessidade do desenvolvedor.

## Primefaces

O Primefaces é uma biblioteca de componentes baseados em JSF. O portfólio do *framework* tem diversos botões, campos de entrada, gráficos entre outros componentes. (FARIA,2013).  
 Além de ter objetos JSF customizados, o Primefaces apresenta funcionalidades Ajax integrados que foram baseados na API (*Application Programming Interface*) Ajax do JSF. API são padrões e rotinas para acesso de um *software* ou plataforma que se baseia em *Web*.

O Ajax é uma forma de comunicação com o servidor, carregado pelo lado do cliente, que permite a atualização da página sem que ela seja carregada novamente. Dessa forma além de um desenvolvimento rápido, conseguimos fazer com que o cliente processe uma parcela da aplicação, economizando recursos do servidor que hospeda a aplicação.

## Bootstrap

O *Bootstrap* é um kit com vários componentes *web* prontos para desenvolver uma aplicação *web*/*mobile* de forma objetiva sem ter conhecimentos prévios em *Java script* e CSS. (SCHMITZ,2014)

Por sem desenvolvido em HTML puro, o *Bootstrap* não interfere em *frameworks* de componentes *web*. Dessa forma, ele é um grande aliado para os desenvolvedores que optem por alterar o *layout* do site sem alterar as funcionalidades dos botões, listas, etc. Este *framework* foi feito pelos desenvolvedores do *Twitter* e é disponibilizado gratuitamente.

# Projeto de dados

O projeto de dados tem o objetivo de mostrar o comportamento de uma classe de uma linguagem orientada a objetos, em um banco de dados relacional. As tabelas a seguir mostram como um relacionamento entre objetos é feito em um banco de dados relacional, além de especificar quais são os campos obrigatórios para fazer um registro no sistema.

Tabela 33 - Projeto de dados (anúncio)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de anúncio** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| idAnuncio | Integer | SIM | X |  |  | X |
| Usuario\_idUsuario | Integer | SIM |  |  |  |  |
| Nome\_Proprietario | Varchar (50) | SIM |  |  |  |  |
| Rua | Varchar (50) | SIM |  |  |  |  |
| Bairro | Varchar (50) | SIM |  |  |  |  |
| Estado | Varchar (2) | SIM |  |  |  |  |
| Cidade | Varchar (50) | SIM |  |  |  |  |
| CEP | Varchar (8) | SIM |  |  |  |  |
| Descricao | Varchar (200) | SIM |  |  |  |  |
| QTD\_Vagas | Integer | SIM |  |  |  |  |
| Valor\_Total | Double | SIM |  |  |  |  |
| Hits | Integer | NÃO |  |  |  |  |
| Status | Tinyint | SIM |  |  |  |  |
| Perfil\_idPerfil | Integer | SIM |  | perfil | idPerfil |  |

Fonte: Elaborado pelo autor (2017).

Tabela 34 - Projeto de dados (perfil)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de perfil** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| idPerfil | Integer | SIM | X |  |  | X |
| Fuma | Tinyint | SIM |  |  |  |  |
| Bebe | Tinyint | SIM |  |  |  |  |
| Chega\_madrugada | Tinyint | SIM |  |  |  |  |
| solteiro | Tinyint | SIM |  |  |  |  |
| Faz\_festa\_casa | Tinyint | SIM |  |  |  |  |
| Cozinha\_Limpa\_casa | Tinyint | SIM |  |  |  |  |

Fonte: Elaborado pelo autor (2017).

Tabela 35 - Projeto de dados (usuário)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de usuário** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| idUsuario | Integer | SIM | X |  |  | X |
| Nome | Varchar (50) | SIM |  |  |  |  |
| Email | Varchar (50) | SIM |  |  |  |  |
| Senha | Varchar (32) | SIM |  |  |  |  |
| Foto | Mediumblob | NÃO |  |  |  |  |
| Rg | Varchar (25) | NÃO |  |  |  |  |
| Cpf | Varchar (25) | SIM |  |  |  |  |
| Sexo | Varchar (1) | SIM |  |  |  |  |
| DataNascimento | Date | SIM |  |  |  |  |
| Celular | Varchar (11) | SIM |  |  |  |  |
| Telefone | Varchar (10) | NÃO |  |  |  |  |
| Permissao | Tinyint | SIM |  |  |  |  |
| Perfil\_idPerfil | Intefer | SIM |  | perfil | idPerfil |  |

Fonte: Elaborado pelo autor (2017).

Tabela 36 - Projeto de dados (anuncio\_vaga)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de anúncio\_vaga** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| idAnuncio\_Vaga | Integer | SIM | X |  |  | X |
| Aceite | Tinyint | NÃO |  |  |  |  |
| OBS\_Aceite | Varchar (200) | NÃO |  |  |  |  |
| Anuncio\_idAnuncio | Integer | SIM |  | Anuncio | idAnuncio |  |
| Usuario\_idUsuario | Integer | SIM |  | Usuário | idUsuario |  |

Fonte: Elaborado pelo autor (2017).

Tabela 37 - Projeto de dados (imagens)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de imagens** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| idImages | Integer | SIM | X |  |  | X |
| Imagem | Mediumblob | SIM |  |  |  |  |
| Descricao\_imagem | Varchar (200) | NÃO |  |  |  |  |
| Anuncio\_idAnuncio | Integer | SIM |  | Anuncio | idAnuncio |  |

Fonte: Elaborado pelo autor (2017).

Tabela 38 - Projeto de dados (log\_login)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de log\_login** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| idlog | Integer | SIM | X |  |  | X |
| Data\_loggout | DATE | SIM |  |  |  |  |
| Usuario\_idUsuario | Integer | SIM |  | Usuário | idUsuario |  |
| Data\_Login | DATE | SIM |  |  |  |  |

Fonte: Elaborado pelo autor (2017).

Tabela 39 - Projeto de dados (log\_operacoes)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela de log\_operacoes** | | | | | | |
| **Atributo** | **Tipo** | **Obrigatório** | **Chave Primária** | **Chave Estrangeira** | | **Auto incremento** |
| **Tabela** | **Campo** |
| id\_log\_operacoes | Integer | SIM | X |  |  | X |
| Tipo\_Operacao | Varchar(8) | SIM |  |  |  |  |
| Data\_Operacao | Date | SIM |  |  |  |  |
| Anuncio\_idAnuncio | Integer | SIM |  | Anuncio | idAnuncio |  |
| IdUsuarioOperacao | Integer |  |  | Usuário | idUsuario |  |

Fonte: Elaborado pelo autor (2017).

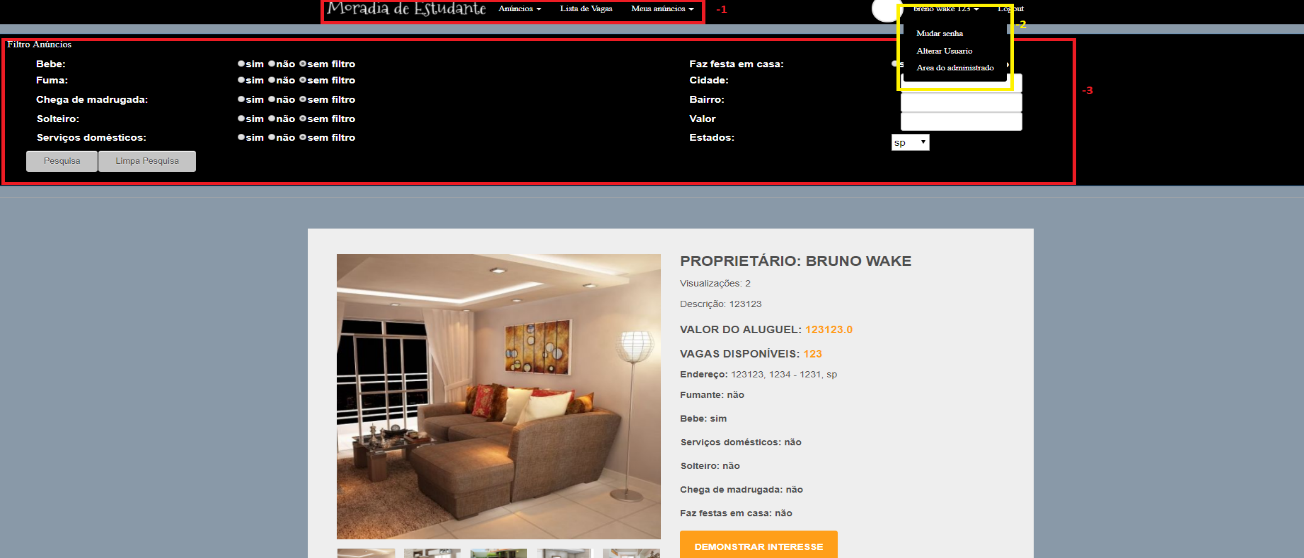
# Projeto de interface

O projeto de interface complementa os diagramas elaborados durante este trabalho. De uma forma visual, mostra as funcionalidades do *software* de uma forma diferente dos diagramas, mostrando diretamente a utilização na *web*. Este tópico irá abordar algumas das principais funções do *software*, que são:

* Inclusão e gerencia de imóveis.
* Demonstrar interesse.
* Alterar usuário e senha.
* Gerenciar interessados no imóvel e consultar telefone
* Utilizar a área administrativa

O menu do *software* estará presente em toda pagina do sistema, ficando em uma posição fixa, dessa forma, o *layout* fica mais limpo e ágil entre a troca de páginas. Podemos verificar o menu visualizando a figura 27, no retângulo de número 1. Continuando na mesma figura, vemos no retângulo 2 o menu para alterar senha, usuário, acesso à área administrativa e *logout*. Por fim, no retângulo de número 3, na figura 27, temos os filtros do sistema, onde poderão ser feitas buscas de anúncios de acordo com o perfil desejado. Para acessar a parte de filtros, devemos ir em ‘menu’ > ‘Anúncios’ > ‘lista’.

Figura 27 - Menu/Filtro

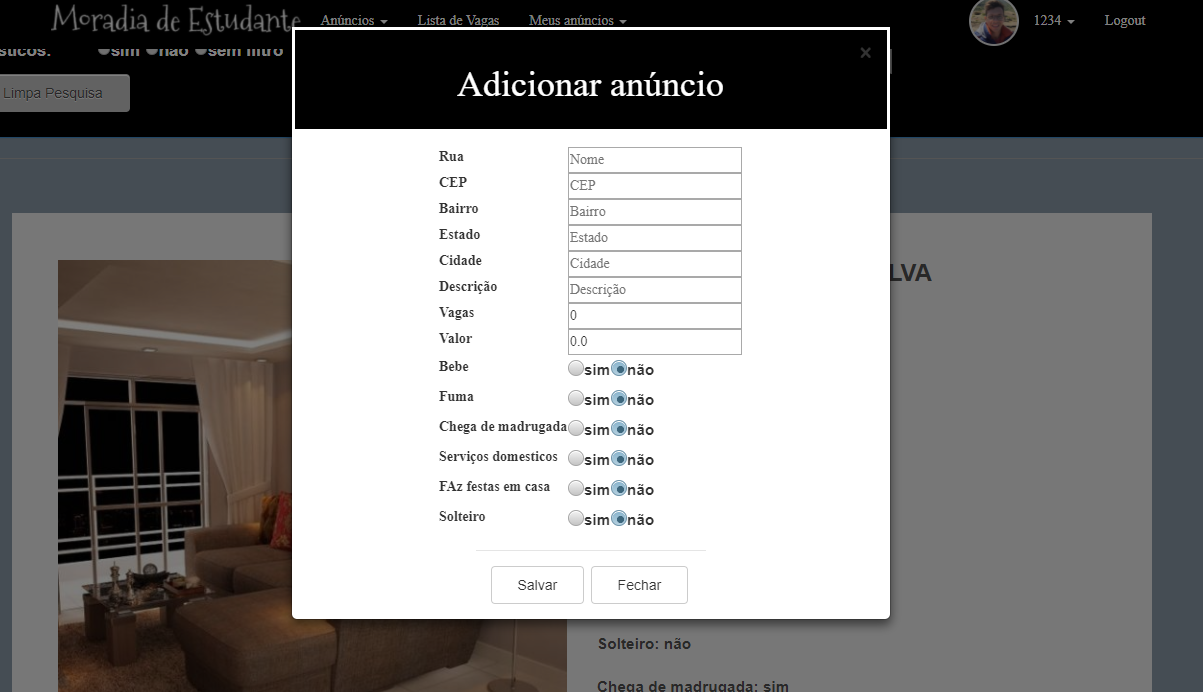


Fonte: Elaborado pelo autor (2017).

## Inclusão de anúncios e gerencia de imóveis

Após efetuar o *login*, o anunciante poderá inserir um anuncio através de um *dialog*. Essa opção é acessível em qualquer página do *site*, basta selecionar o botão referente em ‘menu’ > ‘Meus anúncios’ > ‘adicionar anúncio’ e aparece um cadastro conforme a Figura 28:

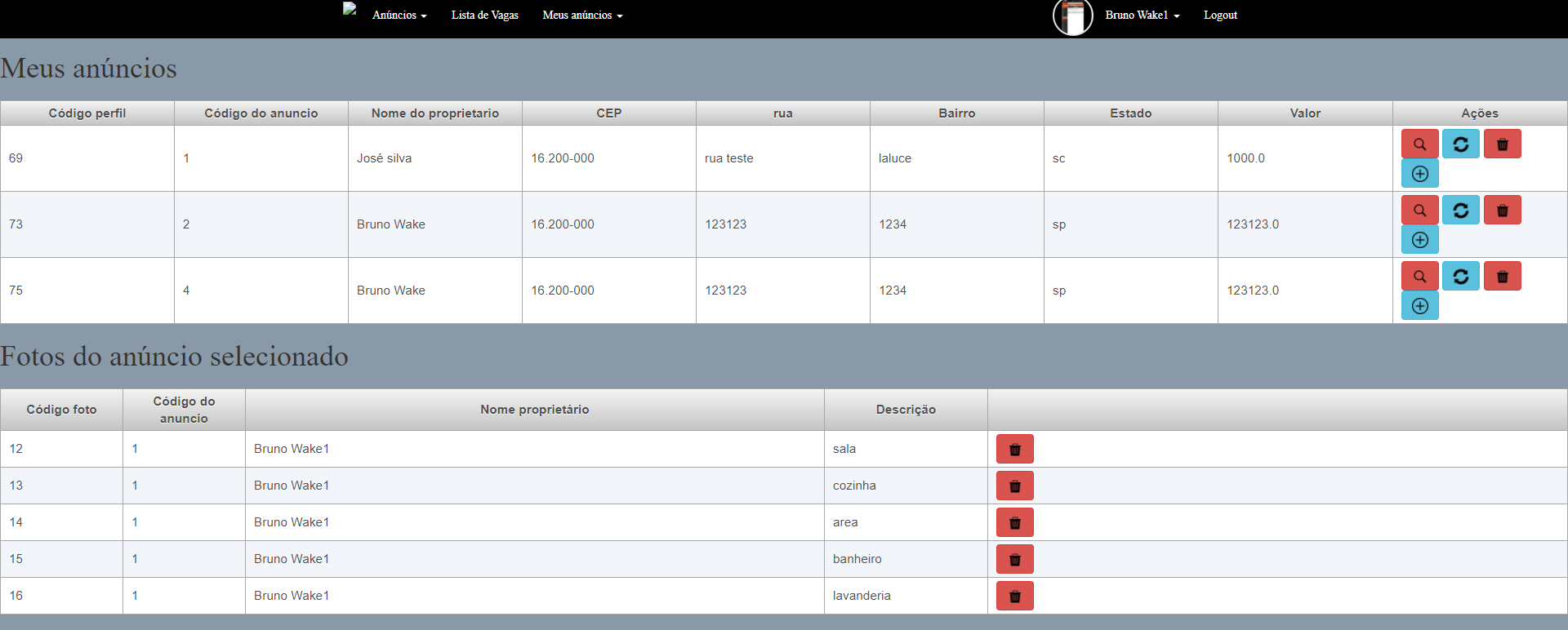
Figura 28 - Adicionar anúncio



Fonte: Elaborado pelo autor (2017).

Para adicionar as fotos aos anúncios, o anunciante deverá selecionar a opção ‘Meus anúncios’ localizado no menu, no item ‘Meus anúncios’. Após carregar a página, uma lista com os anúncios de quem fez o *login* irá aparecer e, nos botões circulados em vermelho, temos as funcionalidades. O primeiro botão da esquerda para a direita é para selecionar as fotos de um anúncio, quando acionado, a lista de baixo atualiza com as imagens referente a linha em que o botão se localiza. O segundo botão serve para alterar o anúncio e o perfil da linha selecionada. O terceiro botão serve para excluir o anuncio da linha selecionada. O quarto botão serve para adicionar uma imagem e o comentário da imagem. Na tabela de fotos, o botão presente serve para excluir a foto da linha selecionada. Todas as ações, com exceção do botão de selecionar fotos da primeira lista, é feito por *dialog*.

Figura 29 - Meus anúncios



Fonte: Elaborado pelo autor (2017).

## Demonstrar interesse:

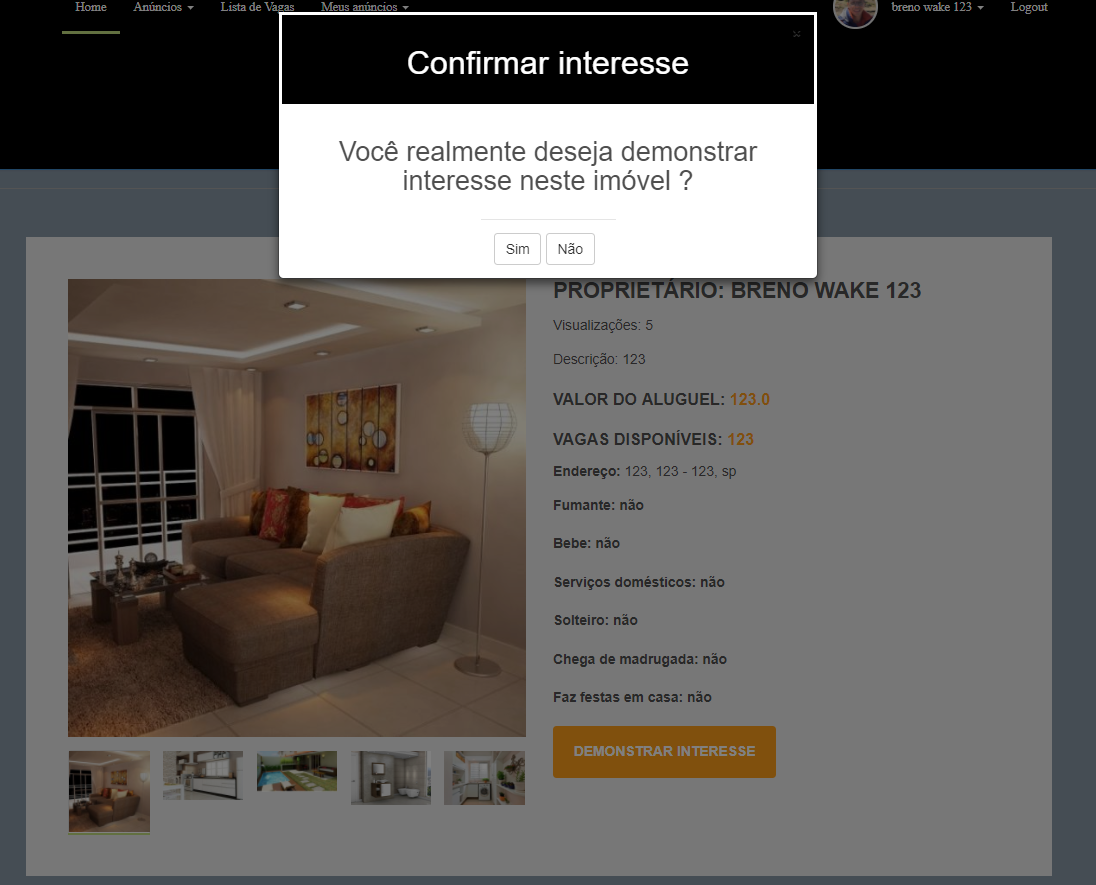
Para demonstrarmos interesse em um imóvel, devemos acessar a lista de imóveis, que fica localizada no menu, dentro do item ‘Anúncios’. Para demonstrar interesse em um imóvel, o usuário devera acessar o *login* no sistema e clicar no botão destacado em vermelho na figura 30, um *dialog* irá aparecer pedindo a confirmação da ação conforme a figura 31.

Figura 30 - Demonstrar interesse



Fonte: Elaborado pelo autor (2017).

Figura 31 - Confirmar interesse



Fonte: Elaborado pelo autor (2017).

## Alterar usuário e senha

Para alterar o usuário e a senha, os procedimentos também são feitos por *dialogs*. Os botões para essas funções estão localizados no retângulo número 2 da figura 27. Podemos ver o formulário de alteração de senha e usuário nas figuras 32 e 33, respectivamente.

Figura 32 - Alteração senha

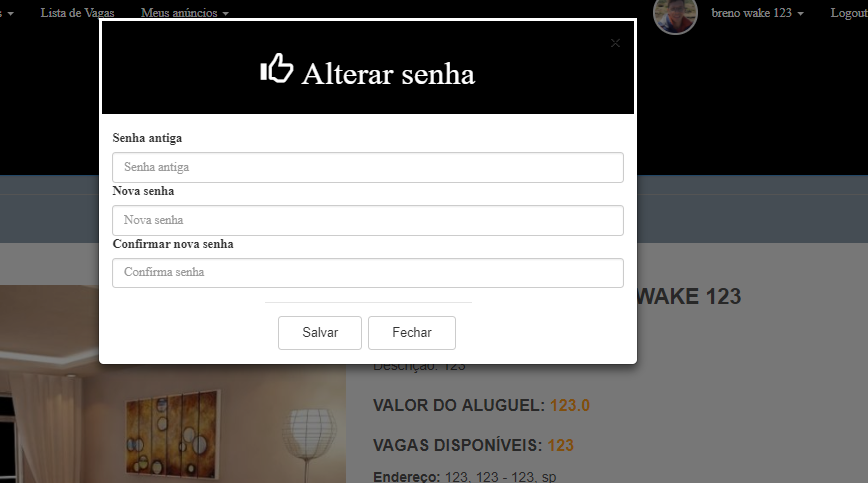
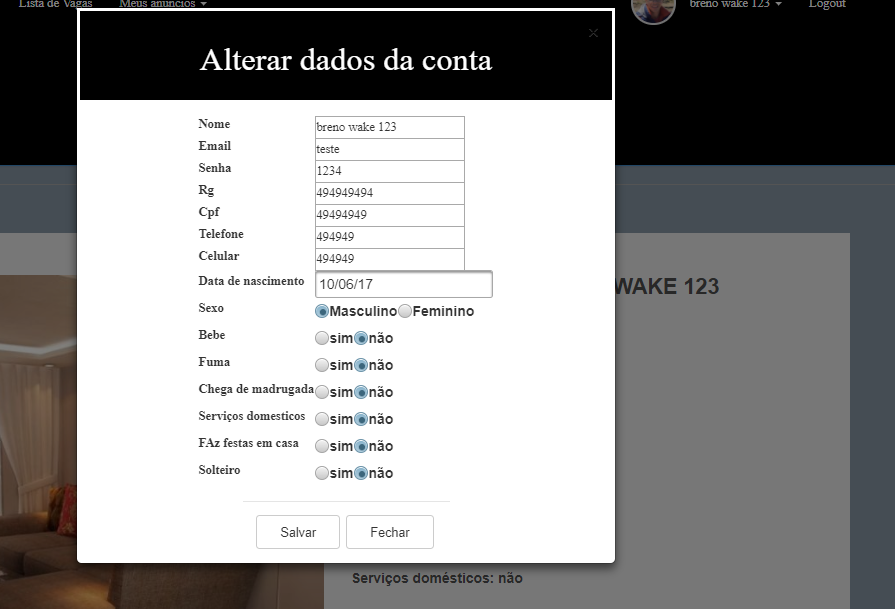
 Fonte: Elaborado pelo autor (2017).

Figura 33 - Alteração usuário



Fonte: Elaborado pelo autor (2017).

## Gerenciar interessados no imóvel e ver telefone

Após demonstrar interesse nos imóveis, é gerado um registro onde o proprietário do imóvel pode ver o perfil do interessado (figura 36), aceitar (figura 37) ou recusar o interesse, vemos essas opções na segunda lista da figura 34. Na opção aceitar ou recusar é possível colocar uma observação sobre o interesse. Também é gerado um registro onde quem demonstrou o interesse aguarda ser aceitou ou não pelo dono do imóvel. Quando aceito, aparecerá um botão na coluna ver telefone (figura 35) - podemos ver essa coluna na figura 34 - da primeira lista, e ao clicar, aparecerá um *dialog* informando o telefone para contato com o anunciante do imóvel. O *dialog* de aceitar é parecido com o recusar, mudando apenas o valor salvo no banco de dados e o texto no cabeçalho do *dialog*. A coluna aceite, na lista de anúncios que tenho interesse, só tem o *status* ‘aguardando resultado’ alterado quando o dono do imóvel aceita o interesse ou recusa o interesse. Quando aceito, o interessado tem o aceite do interesse alterado automaticamente para aceito, quando recusado, é necessário que o dono do imóvel coloque uma observação para que o *status* do aceite seja alterado para recusado.

Figura 34 - Confirmar interesse

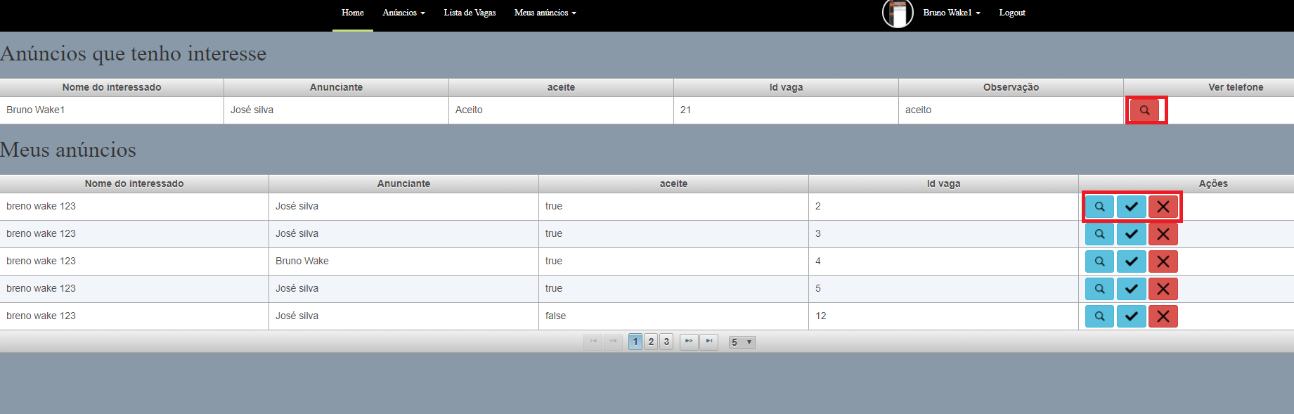
 Fonte: Elaborado pelo autor (2017).

Figura 35 - Dialog telefone

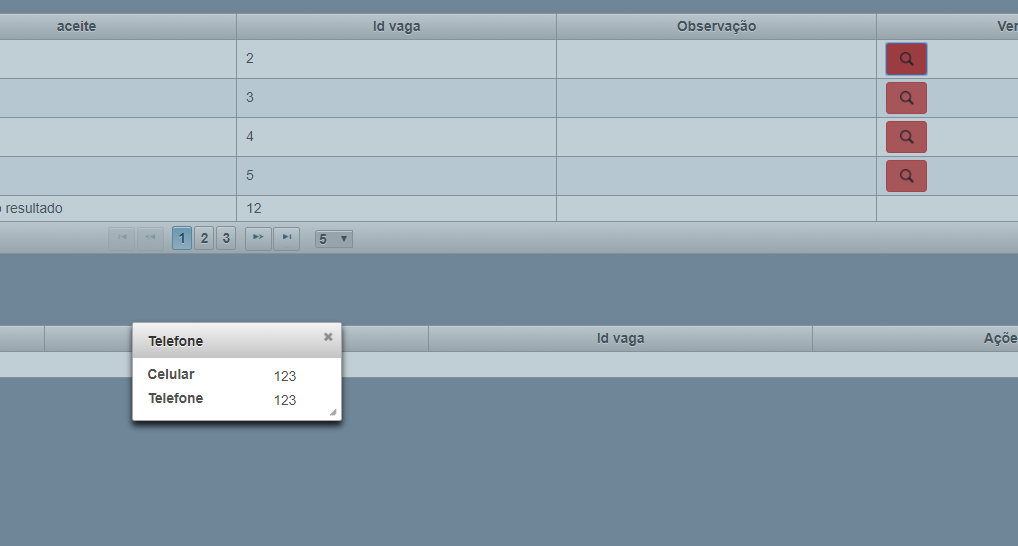
 Fonte: Elaborado pelo autor (2017).

Figura 36 - Dialog perfil interessado

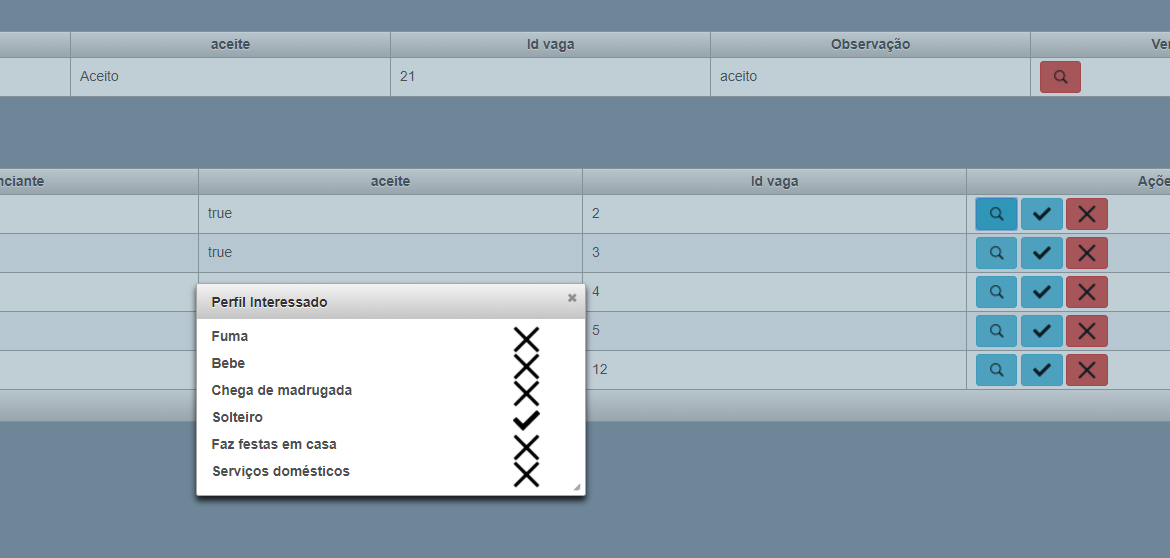
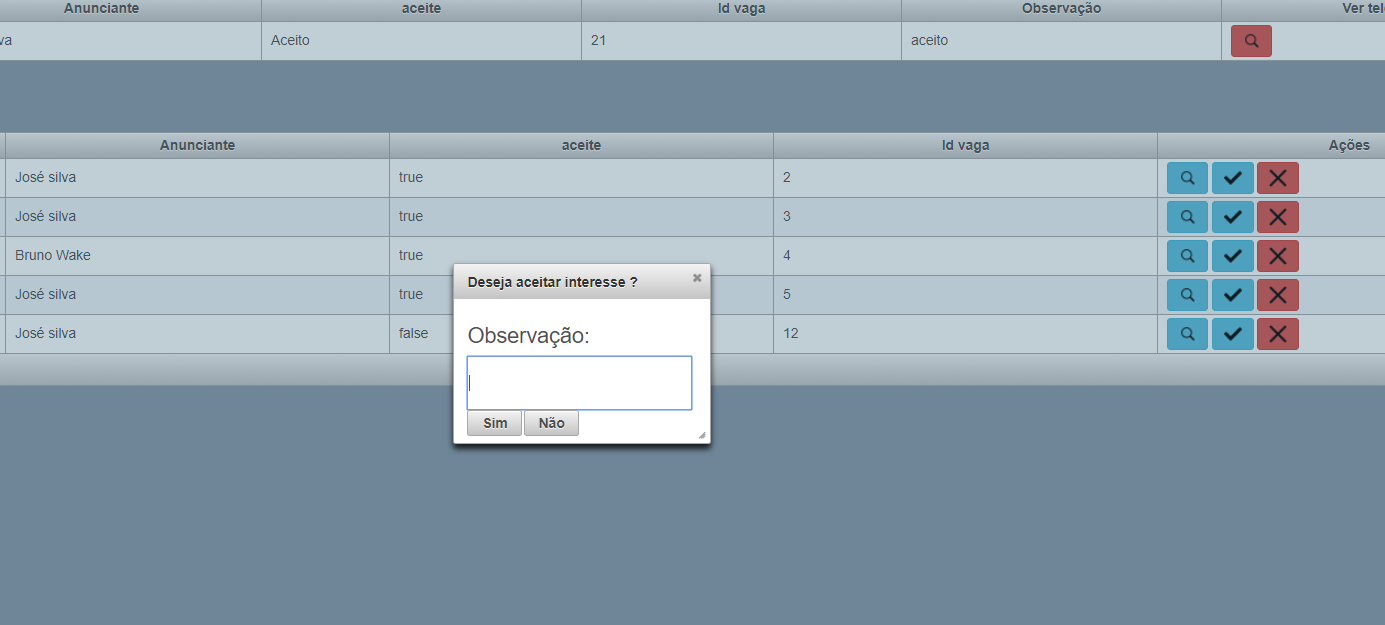
 Fonte: Elaborado pelo autor (2017).

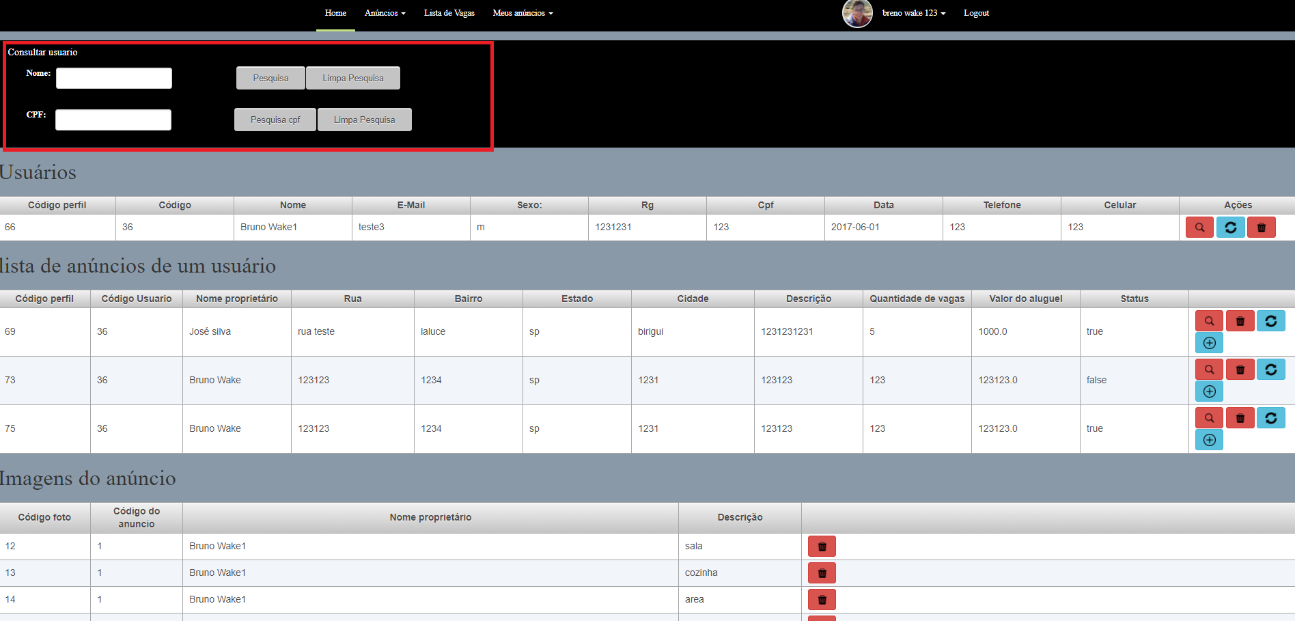
Figura 37 - Dialog aceitar

 Fonte: Elaborado pelo autor (2017).

## Utilizar à area administrativa

O administrador, ao fazer *login*, será direcionado diretamente para essa página. Ele poderá pesquisar o assinante por nome ou CPF inserindo os dados nos campos em destaque na figura 38. As ações executadas por esses filtros irão atualizar a primeira lista, que é referente a todos os usuarios cadastrados no *site*. Os botões na coluna ação, da figura 38 localizada na primeira lista, são de busca aos anúncios do usuário, alteração do usuário / perfil e exclusão do usuário, respectivamente. O primeiro botão, atualiza a segunda lista com os anúncios do usuário que foi selecionado. O segundo botão abre um *dialog* para alteração do anúncio, os campos para alterar são os mesmo para criar um anúncio. E o terceiro botão abre um *dialog* para confirmar a exclusão do usuário. A segunda e a terceira lista tem as mesma ações e funcionalidades da página ‘Meus anúncios’, citada no tópico inclusão de anúncios e gerencia de imagens. O diferencial dessas tabelas do administrador é que ele tem acesso as informações de todos os usuários, não somente dos que efetuaram o *login*.

Figura 38 - Filtro administrador

****** Fonte: Elaborado pelo autor (2017).

# Considerações Finais

Os resultados do estudo realizado possibilitaram a criação de um *software* simples e direto para aqueles que procuram um imóvel dentre o meio acadêmico. Com *layout* simples e mecanismo repetitivo, o usuário se adapta facilmente ao ambiente pois, apesar dos botões fazerem tarefas distintas, a localidade no *layout* e a forma de como os dados mostrados são parecidos. Essas funcionalidades permitem que o usuário do sistema ganhe tempo sem precisar alternar entre diferentes páginas para obter as informações desejadas. Baseado em situações do cotidiano presente em um meio acadêmico, o objetivo de facilitar e preservar os dados dos usuários foi concluída com sucesso. Facilidade na utilização, objetividade nos anúncios e clareza ao saber onde os seus hábitos serão bem-vindos, são as características incorporadas ao projeto que cumpre com os objetivos propostos de auxiliar o mercado emergente, economizar tempo, dinheiro e excessivos contatos com os anunciantes.

# Melhorias Futuras

Este tópico tem como objetivo registrar as melhorias futuras para o sistema. Dentre essas melhorias estão novas funcionalidades, melhorias de desempenho do *software* e alterações no *layout* com objetivo de melhorar o visual da aplicação. As futuras melhorias são:

* Inclusão de miniaturas das fotos da lista de imagem
* Implementação de contrato para diferenciar o inquilino do anunciante
* Implementar *Lazy Loading* no hibernate
* *Chat* para anunciante e inquilino conversarem

# Referências bibliográficas

CAELUM. FJ-21 Java para Desenvolvimento Web. Apostila Versão 14.9.1. CAELUM Ensino e Inovação, s/d.

CAELUM. FJ-31 Java EE avançado e Web Services. Apostila Versão 15.6.17. CAELUM Ensino e Inovação, s/d.

FOWLER, Martin. **UML Essencial:** Um breve guia para a linguagem-padrão de modelagem de objetos. 3. ed. Bookman, 2005.

FARIA, Thiago. **Java EE 7:**Com JSF, PrimeFaces e CDI. Algaworks, 2013.

FARIA, Thiago. **JPA E HIBERNATE**. Algaworks, 2015.

Governo do Brasil. **Total de estudantes formados no País dobrou em sete anos, segundo Censo da Educação.** Disponível em: <http://www.brasil.gov.br/educacao/2011/01/total-de-estudantes-formados-no-pais-dobrou-em-sete-anos-segundo-censo-da-educacao>. Acesso em: 28 jul. 2014.

Jornal do Comércio. **Estudantes aumentam a demanda pelos aluguéis. Disponível em**:<http://jcrs.uol.com.br/site/noticia.php?codn=187682> Acesso em: 11 de novembro de 2017.

LARMAN, Craig. **Applying UML and Patterns:** An Introduction to Object-Oriented Analysis and Design and the Unified Process. 2.ed. Prentice Hall, 2001.

SCHMITZ, Daniel. **Bootstrap:**framework front-end para aplicações web e mobile. 3. ed. Leanpub, 2014.

Utilizando o Hibernate em uma Aplicação Web. Disponível em: <https://netbeans.org/kb/docs/web/hibernate-webapp\_pt\_BR.html> Acesso em: 11 de novembro de 2017.