

**UNIVERSIDADE NOVE DE JULHO – UNINOVE**  
**DIRETORIA DE INFORMÁTICA**



ELIZABETE DE OLIVEIRA DOS SANTOS	–	RA 9222
KAMILLA SAQUETO FORTUNATO	–	RA 9211
MONIQUE FARIAS ALVES	–	RA 421
RUAN CARLOS ALVES FONSECA	–	RA 4231
RENAN NOBRE DA SILVA	–	RA 9221
THIAGO HENRIQUE BENEDITO ARGOLO	–	RA 421

**PROJETO EM DESENVOLVIMENTO DE SISTEMAS**  
**EMPRESA – Pet House**

**SÃO PAULO**  
**2023**

**ELIZABETE DE OLIVEIRA DOS SANTOS – RA 9222**  
**KAMILLA SAQUETO FORTUNATO – RA 9211**  
**MONIQUE FARIAS ALVES – RA 421**  
**RUAN CARLOS ALVES FONSECA – RA 4231**  
**RENAN NOBRE DA SILVA – RA 9221**  
**THIAGO HENRIQUE BENEDITO ARGOLO – RA 421**

**PROJETO EM DESENVOLVIMENTO DE SISTEMAS**  
**EMPRESA – Pet House**

Trabalho apresentado à Universidade  
Nove de Julho, UNINOVE, em  
cumprimento parcial às exigências da  
disciplina de Projeto em  
Desenvolvimento de Sistemas, sob  
orientação do Prof. **Leandro Fernandes  
da Mota**

## Sumário

1. RESUMO.....	4
2. OBJETIVOS.....	5
3. MODELAGEM DE PROCESSOS DE NEGÓCIOS.....	6
4. PROCESSO DE DESENVOLVIMENTO DE SOFTWARE.....	8
5. MODELAGEM ORIENTADA A OBJETOS.....	12
5.1. DIAGRAMA DE CLASSES.....	12
6. DESENVOLVIMENTO ORIENTADO A OBJETOS.....	13
6.1. TELAS DO SISTEMA.....	13
6.2. BANCO DE DADOS.....	19
6.3. CÓDIGOS DO SISTEMA.....	21
7. CONCLUSÃO.....	23
8. REFERENCIAS.....	24

## **1. RESUMO**

Este trabalho trata-se de um sistema que será implementado em uma empresa no setor de pet shop, com o objetivo de facilitar seu atendimento, organizar seus serviços e melhorar a visibilidade da empresa. No sistema foi aplicado uma tela inicial, onde o usuário tem a visualização de algumas informações da empresa, após realizar seu cadastro e efetuar seu login o usuário terá acesso aos serviços, contato e localidade onde a empresa atua. O sistema foi produzido a partir do Java juntamente com um banco de dados para armazenar informações dos usuários.

## **ABSTRACT**

This work deals with a system that will be implemented in a company in the pet shop sector with the objective of facilitating its service, organizing its services and improving the company's visibility. An initial screen was applied to the system, where the user has the visualization of some company information, after registering and logging in, the user will have access to the services, contacts and location where the company operates. The system was produced from Java along with a database to store user information.

## **2. OBJETIVOS**

A nossa essência é acolher e cuidar dos nossos clientes e seus pets. Desde 2020, a Pet House tem como propósito transformar a vida de milhares de famílias promovendo acolhimento e aumentando a expectativa de vida dos pets. Com o objetivo de oferecer um ambiente completo e com o que há de melhor no mercado veterinário, sempre contando com os melhores profissionais. E aqui na Pet House você conta com um serviço de banho e tosa de qualidade e com muito amor todos os dias. Além de acomodar o seu pet caso queira descansar no final de semana. Tudo isso, aliado a uma estrutura moderna e completa para proporcionar toda comodidade num só lugar.

Nossa busca diária é conquistar a confiança dos nossos tutores, pois entendemos que cada animalzinho é o bem mais precioso de cada família. E essa confiança é o nosso maior presente.

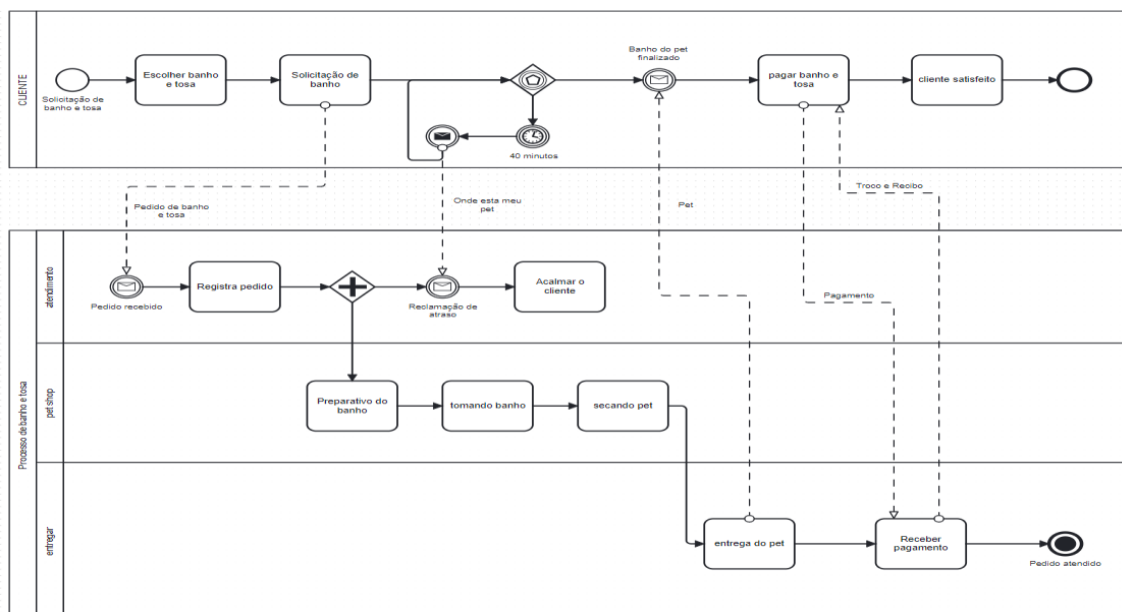
### 3. MODELAGEM DE PROCESSOS DE NEGÓCIOS

Business Process Modeling ou Modelagem de Processo de Negócio é uma atividade que representa os processos de uma empresa, sendo que o processo atual pode ser melhorado e analisado ou podendo impor um processo novo, tendo como objetivo melhorar a eficiência do processo e da qualidade.

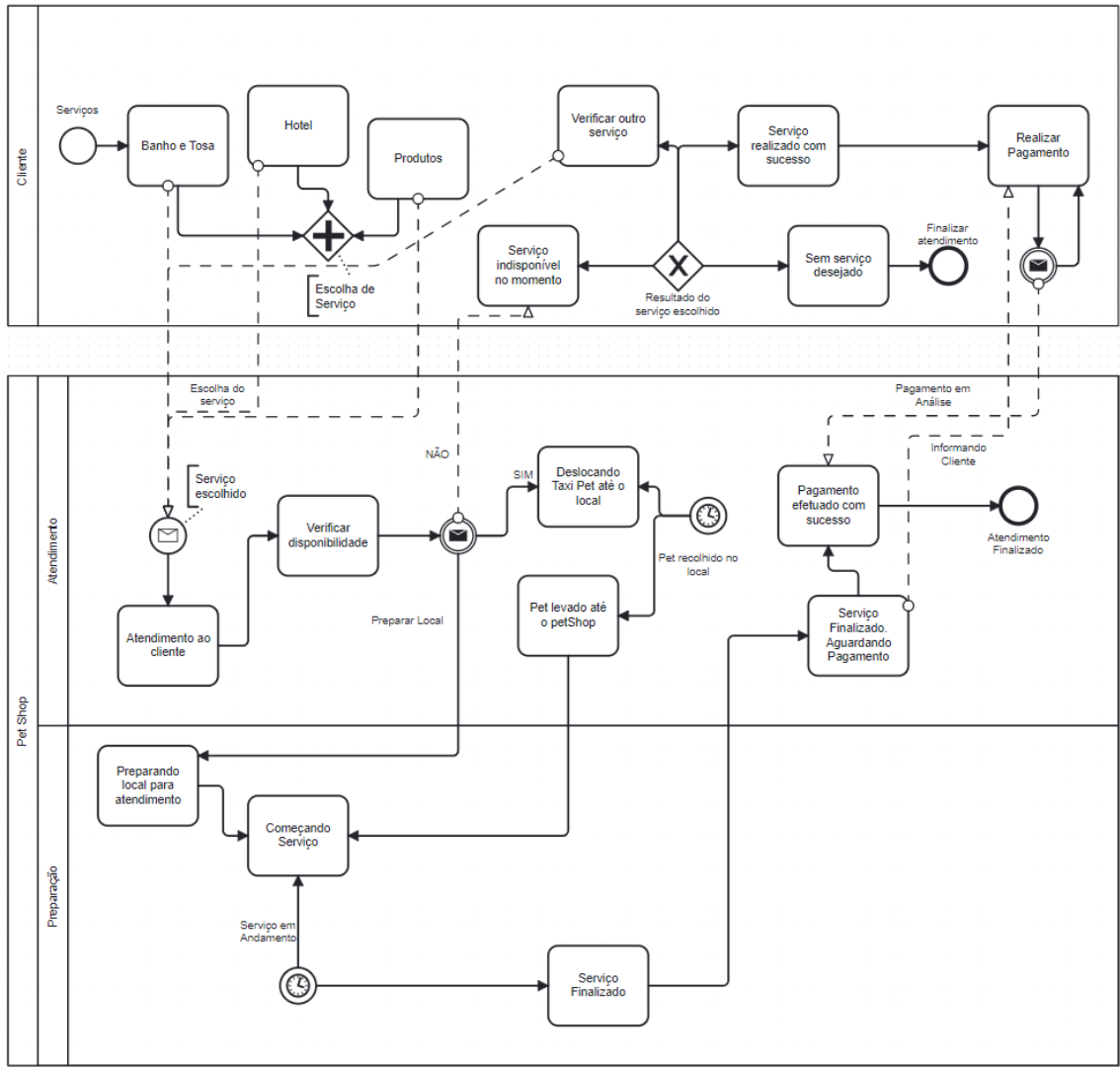
Representa o processo pelo modelo de notação BPMN (notação de gerenciamento de processos de negócios, utilizando desenho para demonstração do usuário.), são constituídos por fluxogramas da realização das atividades, devem ser descritos de forma que consigam entender cada fluxo que deve ser adotado (WILLIAM P. 2012).

Sendo assim o processo atual de uma organização são chamados de AS-IS e o processo de melhorias e propostas projetados para o futuro são chamados de TO-BE.

No Pet House tínhamos o processo AS-IS aonde o serviço principal era somente o banho e tosa, segue abaixo o fluxograma:



Conforme avaliamos que necessitava de um novo processo, pois adotamos serviços novos e para melhorar a eficiência do processo e da qualidade, adotamos o processo abaixo de melhorias:



#### 4. PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

Para desenvolver o sistema foi utilizado o modelo ágil Scrum deixando claro logo de início papéis e funções para o time se auto-organizar.

Scrum é um modelo ágil um conjunto de boas práticas, aplicado para gerenciar projetos complexos a onde não se sabe suas necessidades ou todas suas etapas. Com foco no time, o Scrum transforma os processos executados mais simples, mantendo o que for registrado visível no andamento de todas as etapas (AUGUSTO REIS, 2021).

##### Time Scrum

##### → Product Owner (Dono do produto)

Utilizando o método ágil do Scrum para gerenciar o projeto, com o papel de Product Owner, **Kamilla**, sendo a conexão com o cliente, e a responsável por maximizar o valor de negócios do projeto.

##### → Scrum Master

**Elizabete** como Scrum Master, tirando todos os impedimentos, ensinando as práticas do Scrum, garantindo que todos os processos estejam sendo realizados e facilitando todo o processo da equipe.

##### → Scrum Team (Time de desenvolvimento)

**Monique, Thiago, Renan e Ruan** sendo o time de desenvolvimento, no foco de desenvolver o Software e atingir os objetivos de cada sprint, identificando impedimentos que necessitam ser removidos do projeto.



## Reunião de planejamento da Sprint

Reunião de planejamento de Sprint em 4 semanas durante 8hrs.

O Product Owner realizou a explicação para o Time Scrum sobre prioridades e as atividades alvo. Na segunda parte, nosso time de desenvolvimento decidiu como concluir o que selecionou do Backlog do produto e suas funções para características do Software finalizado para cumprir a meta Sprint.

## Reuniões Diárias

Reuniões Diárias com duração de 15 minutos.

Nos reunimos para ver o andamento do projeto, discutimos as partes já realizadas e o que deve ser realizado antes da próxima.

## Revisão da Sprint

Nossa reunião Sprint durou aproximadamente de 4 horas.

Os participantes incluem o **Time Scrum** e os Stakeholders chave convidados pelo Product Owner; **O Product Owner** esclarece quais itens do Backlog do Produto foram “Prontos” e quais não foram “Prontos”; **O Time de Desenvolvimento** discute o que foi bem durante a Sprint, quais problemas ocorreram dentro da Sprint, e como estes problemas foram resolvidos, demonstra o trabalho que está “Pronto” e responde as questões sobre o incremento; **O Product Owner** discute o Backlog do Produto tal como está. Ele (ou ela) projetou os prováveis alvos e datas de entrega baseado no progresso até a data.

## Retrospectiva do Sprint

Nossa Retrospectiva do Sprint durou aproximadamente 4 horas, servindo para discutir e revisar o que teve no sprint anterior e fazendo melhorias para as próximas sprints. O Scrum Master encoraja o Time Scrum a melhorar o

processo de desenvolvimento e as práticas para torná-lo mais efetivo e agradável para a próxima Sprint.

Ao final da Retrospectiva da Sprint, nosso Time Scrum conseguiu identificar melhorias que serão implementadas na próxima Sprint ou, dependendo do caso, em um momento mais oportuno.

## **Backlog do produto**

O Software será um APP para Pet shop, facilitando o dia a dia de donos de pet e permitindo melhores resultados ao proprietário.

### **Criação das telas:**

**Tela login** → A tela inicial onde o cliente terá acesso ao app com sua conta ou ali mesmo informar se não possui se caso o cliente não possuir uma conta será direcionado a tela cadastrar onde poderá decidir entre se cadastrar informando nome completo, e-mail e senha ou continuar sem cadastro.

**Tela Home** → Será a tela principal, lá o cliente encontrará ícones com os recursos disponíveis:

- Home - História;

Uma maneira do cliente conhecer sobre a empresa. Esse ícone possui a nossa história, objetivos e nossos ideais.

- Serviços;

Apresentamos os nossos serviços disponíveis, juntamente com imagens para o cliente se sentir confiante, sendo assim, alguns deles são:

- Hotel;
- Banho e Tosa;
- Petshop;

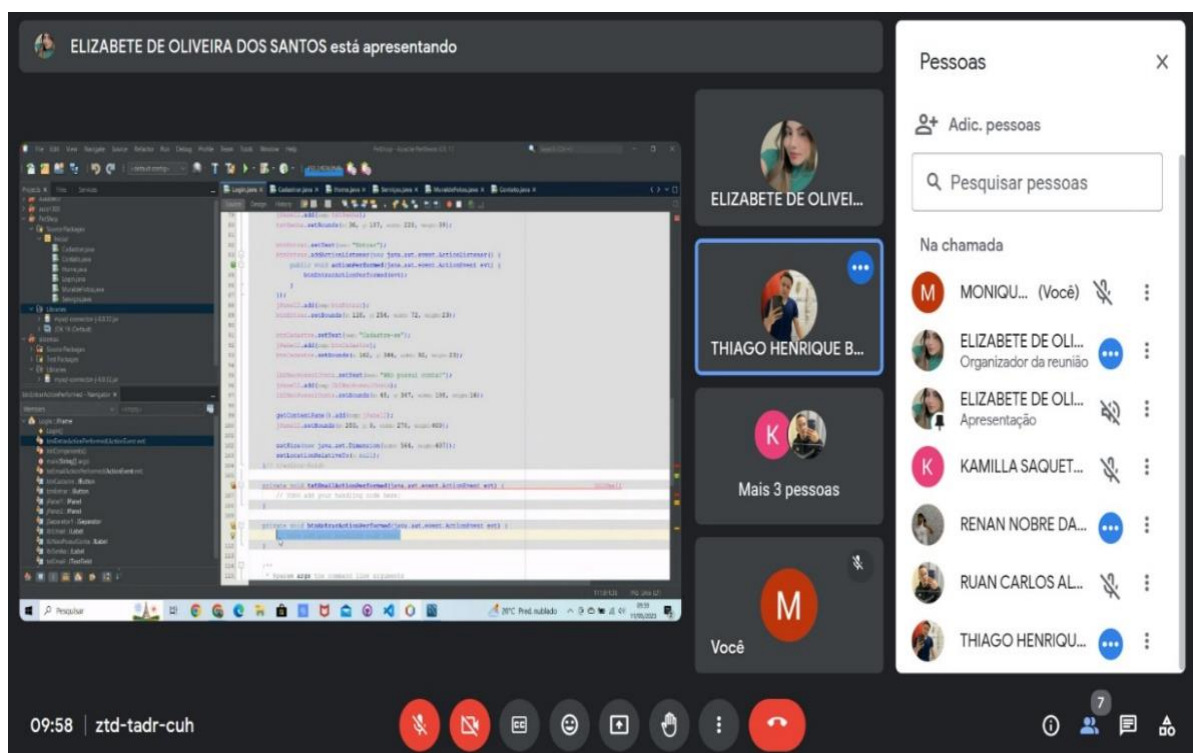
- Mural de Fotos;

Nesta tela mostramos alguns de nossos pets que vieram nos visitar com permissão da família para exposição, também com a opção de informar se gostou ou não da visita.

- Contato;

Informamos a nossa localização, juntamente com o mapa para verificação, também o nosso contato WhatsApp e telefone para mais informações.

## DAILY SCRUM



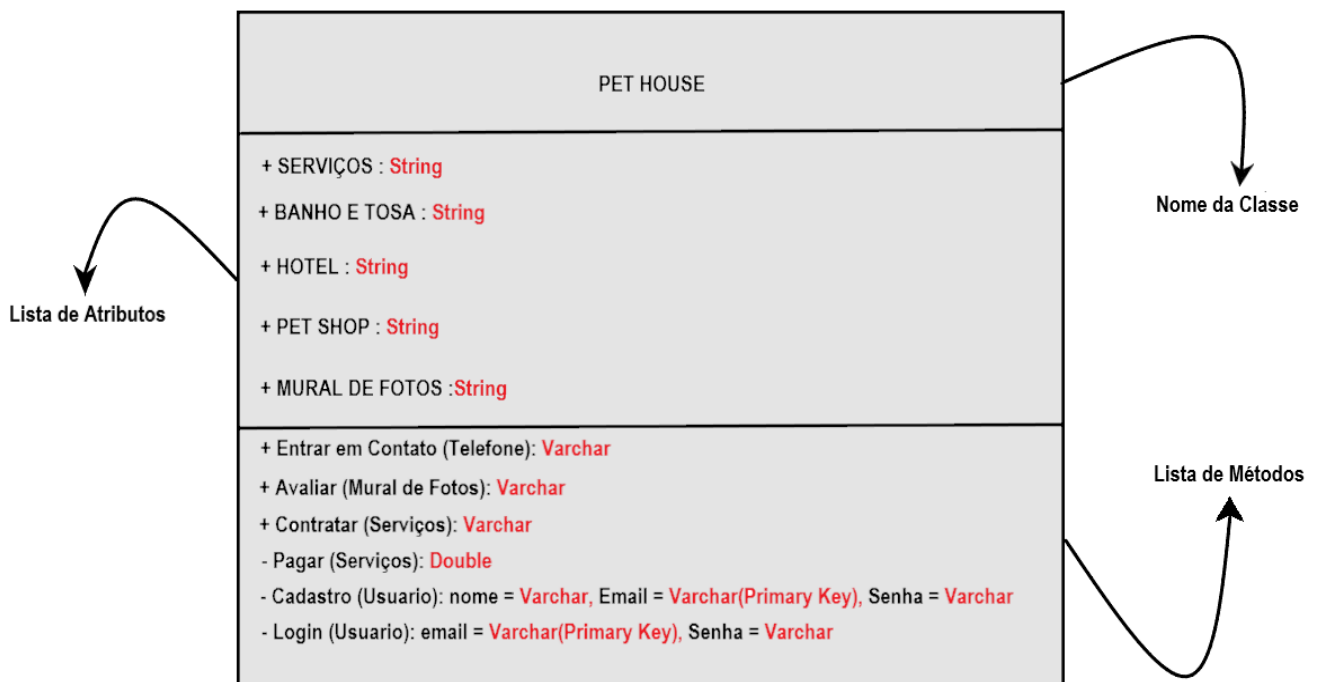
## 5. MODELAGEM ORIENTADA A OBJETOS

A modelagem orientada a objetos é aspecto-chave da engenharia de software com foco no projeto e implementação de sistemas usando princípios orientados a objetos. E usamos essas formas de abordagens para modelagem orientada a objetos:

Primeiro, entendemos o domínio do problema, depois identificamos objetos e classes e definimos os relacionamentos de classes. Após isso, é feita a definição das responsabilidades e métodos da classe, onde fazemos a implementação, teste e depuração junto com o refinamento e iteração, usando tais princípios citados acima, que são orientados a objetos.

### 5.1. DIAGRAMA DE CLASSES

Diagramas de classes são basicamente cópias de um sistema ou subsistema, pode ser utilizado para modelar os itens que são utilizados em um sistema, exibindo a relação com objetos e descrevendo o que os objetos realiza e serviços que fornecem (LEONARDO M. 2017).



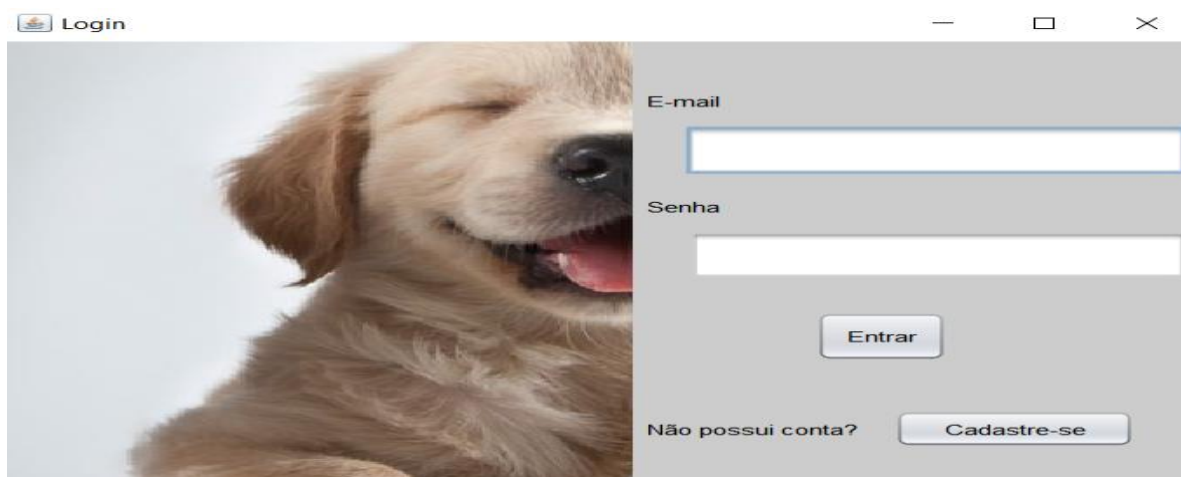
## 6. DESENVOLVIMENTO ORIENTADO A OBJETOS

Os tópicos de desenvolvimento orientados a objetos incluem:

Abstração (onde classes e interfaces definem comportamentos e atributos comuns compartilhados por classes relacionadas), Herança (promove a reutilização de código e permite a organização hierárquica das classe), Polimorfismo (permite vinculação dinâmica e flexibilidade nas interações de objetos), Encapsulamento (garante a integridade dos dados e fornece uma interface clara para interagir com os objetos), Modelagem (melhora a capacidade de manutenção, reutilização e compreensão da base de código), Padrões de Projeto (fornecem abordagens comprovadas para organizar e estruturar sistemas orientados a objetos de forma eficaz) e Princípios Sólidos (orientam o design e desenvolvimento de software para obter flexibilidade, capacidade de manutenção e extensibilidade). Seguindo uma abordagem sistemática e utilizando destes tópicos aqui citados, criamos modelos desenvolvidos e orientados a objetos, que são projetos de software robusto e de fácil manutenção e utilização para o usuário.

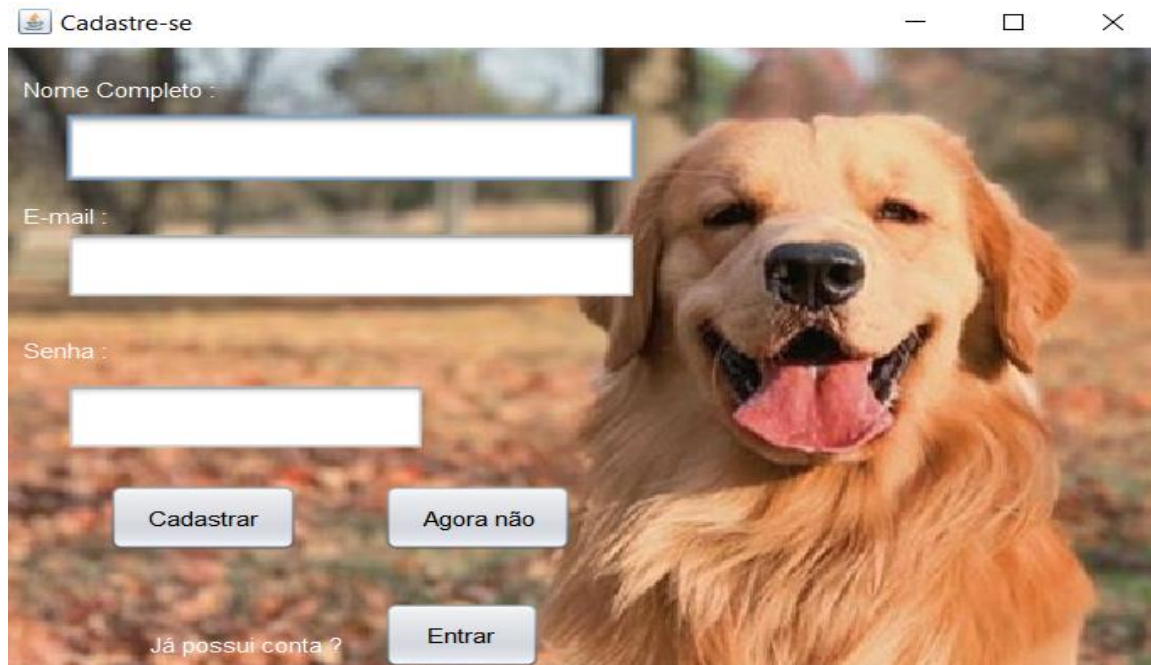
### 6.1. TELAS DO SISTEMA

As telas foram elaboradas de acordo com o tema da empresa, com um design simples, mas agradável e com informações de fácil leitura e acesso.



Nesta tela o usuário consegue estar realizando login ou cadastro, que irá

diretamente para o banco de dados, sendo assim, quando realiza o cadastro o sistema informa que foi cadastrado com sucesso, e segue o direcionamento de login para entrar, caso não queira se cadastrar tem a opção do botão agora não que irá diretamente para a outra tela.



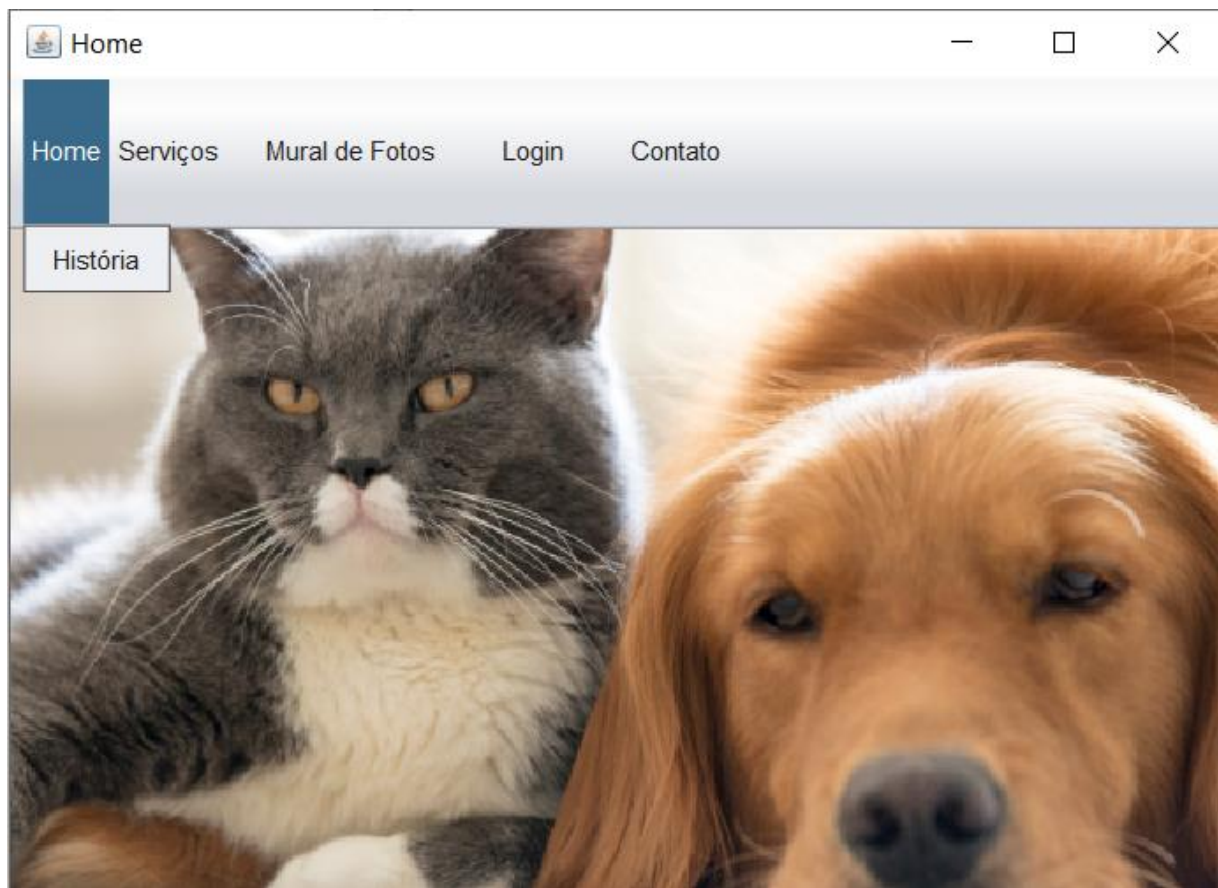
The screenshot shows a web browser window titled "Cadastre-se" (Register). The background of the registration form is a close-up image of a smiling golden retriever. The form contains three input fields: "Nome Completo:" (Full Name), "E-mail:", and "Senha:" (Password). Below the fields are two buttons: "Cadastrar" (Register) and "Agora não" (Not now). At the bottom left, there is a link "Já possui conta?" (Already have an account?), and at the bottom right, there is a button "Entrar" (Login).

Nesta tela segue sendo o principal direcionamento para outras telas, sendo a tela de Home.

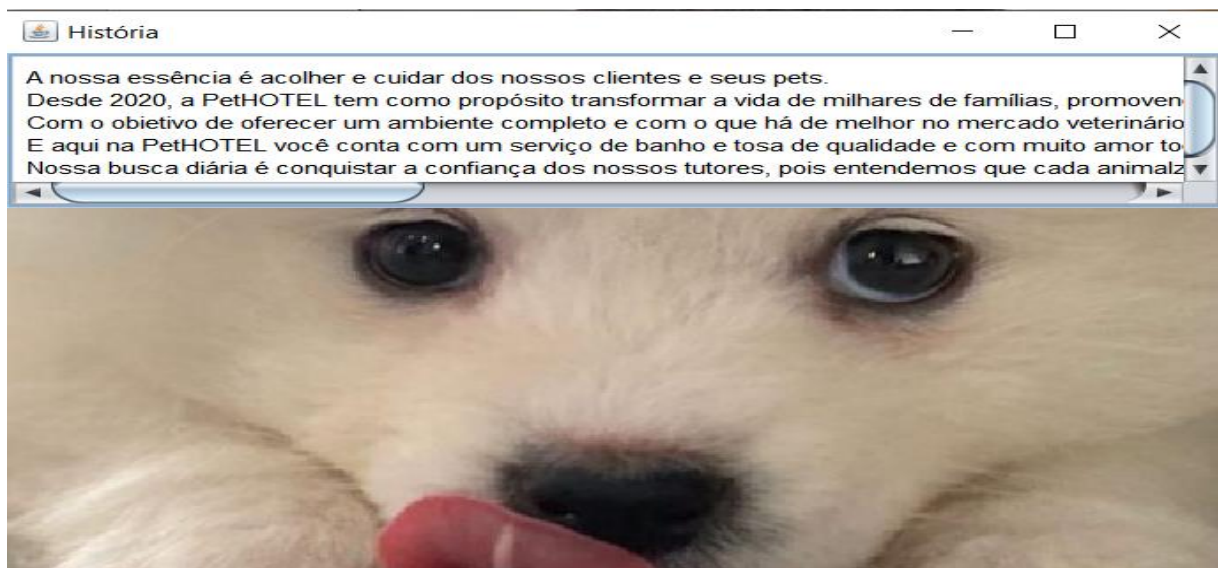




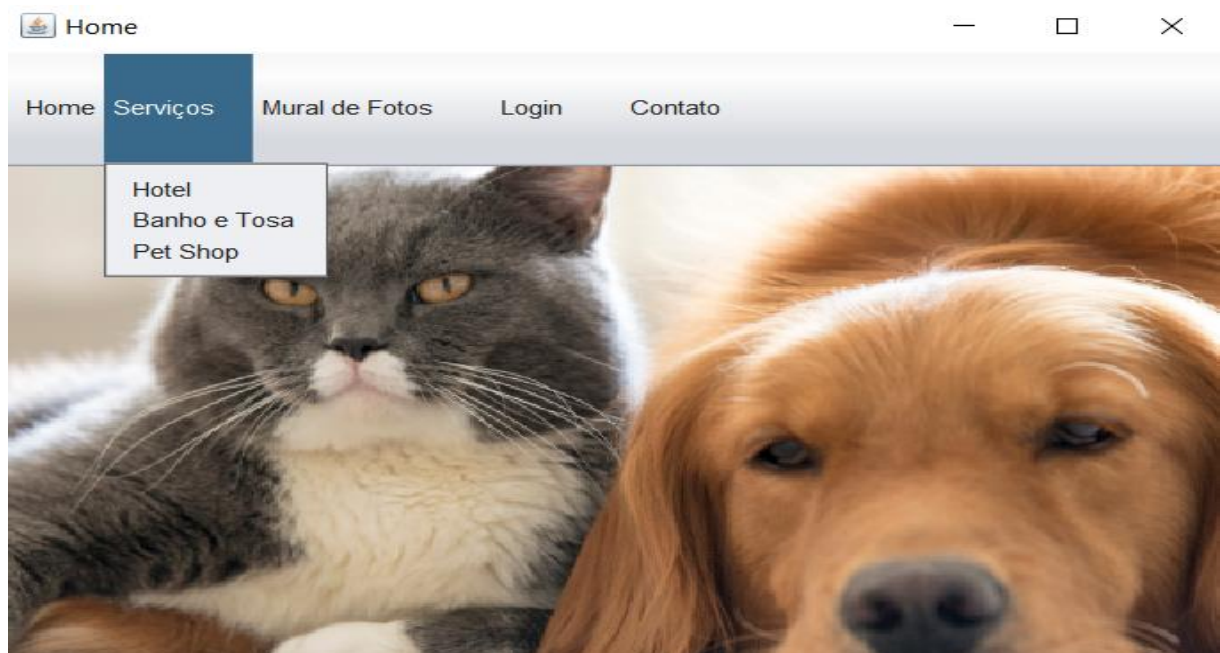
Abaixo de Home, tem a opção de história.



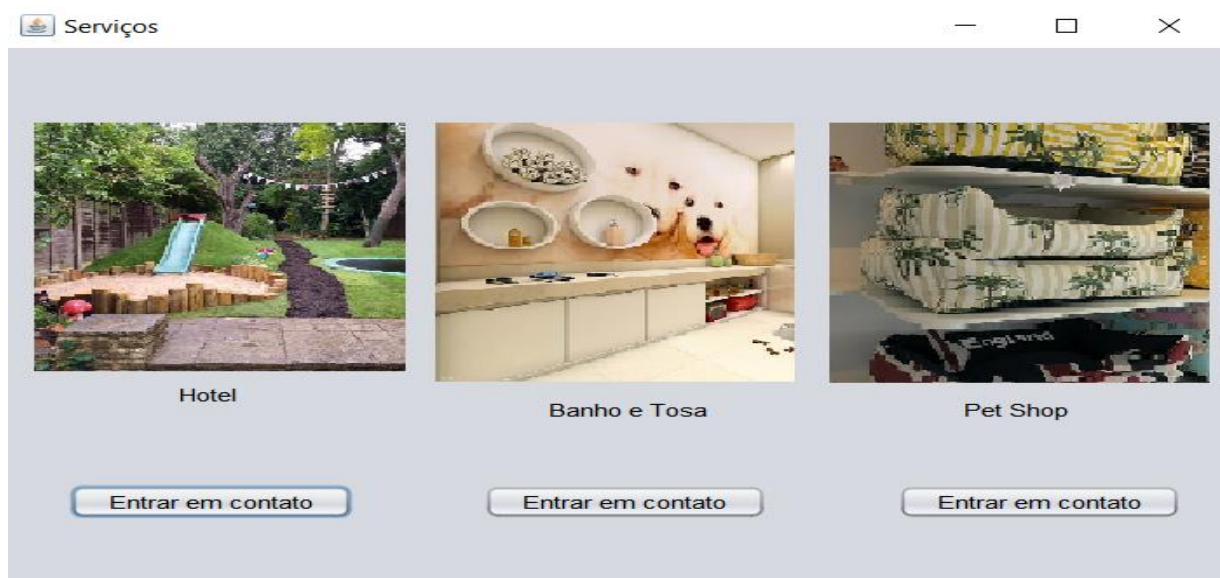
Clicando em história, abre a tela da nossa história do Pet House e nossos ideais.



Seguindo novamente na tela de Home, tem a opção de serviços.

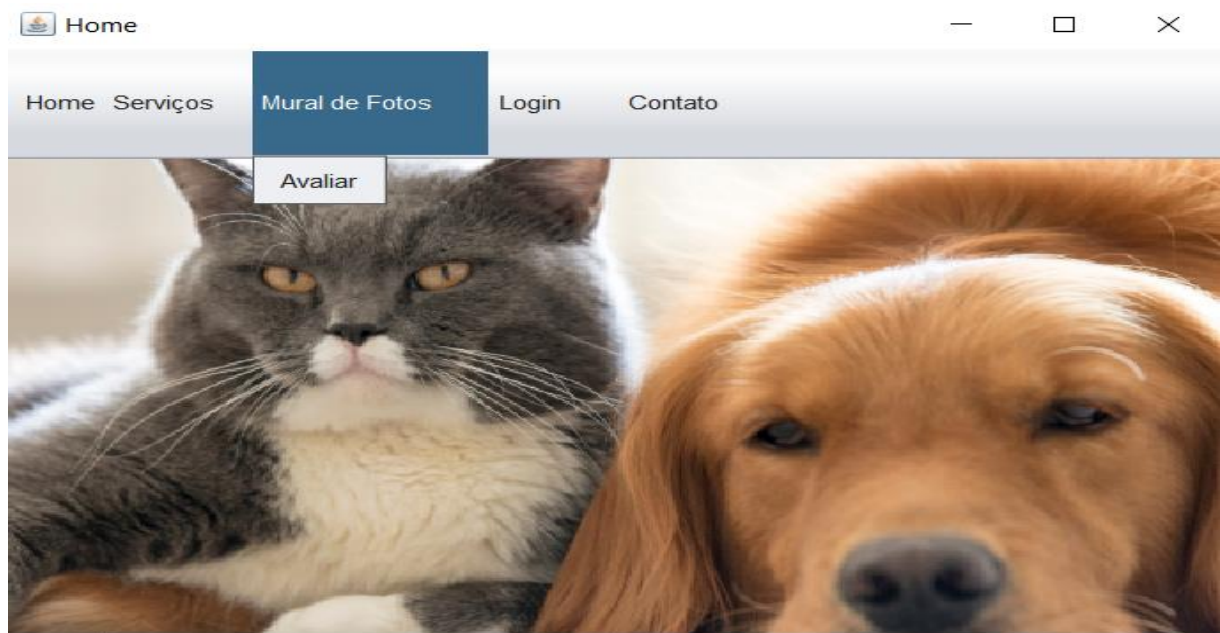


Ao clicar em qualquer opção de serviços, segue para a tela de Serviços, onde informamos o que temos a oferecer no nosso Pet House, ao clicar no botão de entrar em contato segue o direcionamento para a tela de Contato.





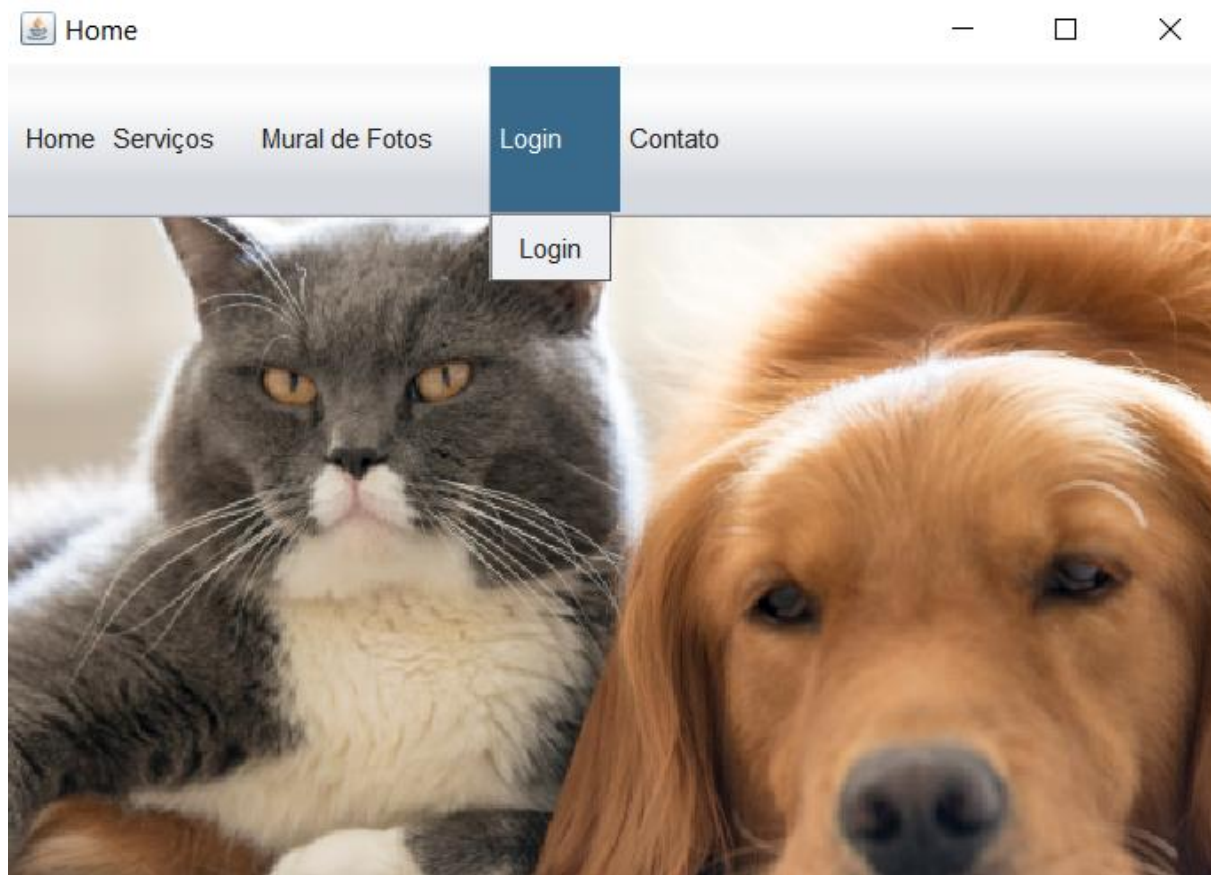
Seguindo novamente na tela de Home, tem a opção de Mural de Fotos.



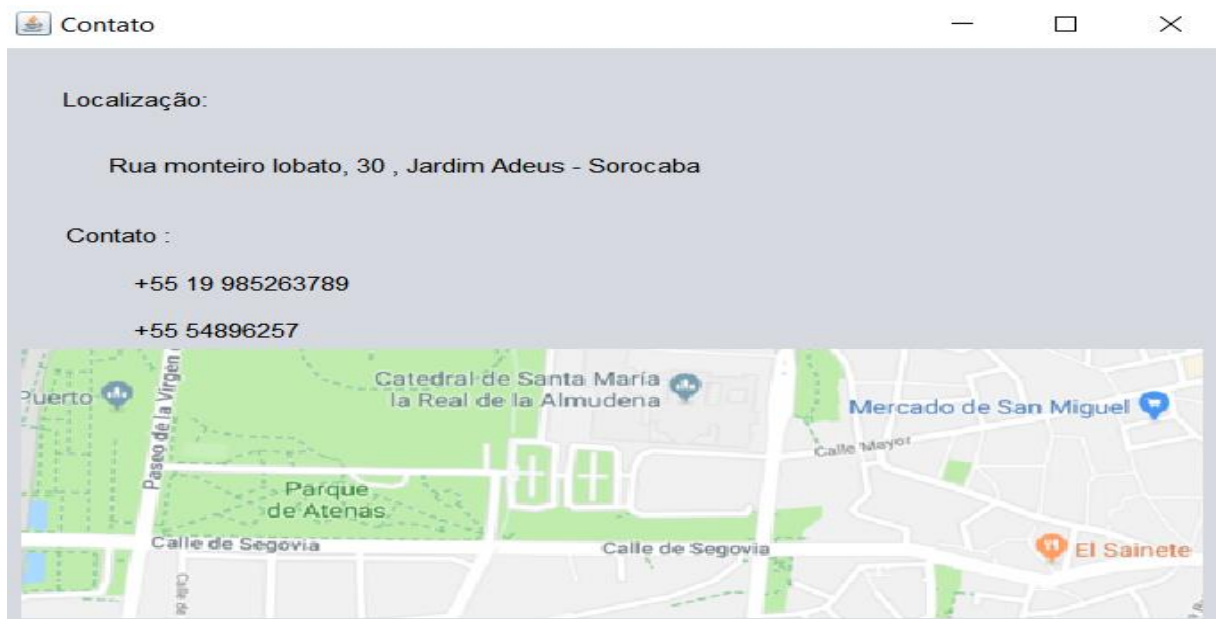
Ao clicarmos em avaliar, abre a tela de Mural de Fotos, com fotos de nossos pets que vieram nos visitar, e tem a opção de avaliar se gostou ou não gostou, que irá diretamente para o banco de dados registrando a informação e deixando armazenada.



Seguindo novamente na tela de Home, tem a opção de Login, aonde irá diretamente para a Tela de Login informado anteriormente.



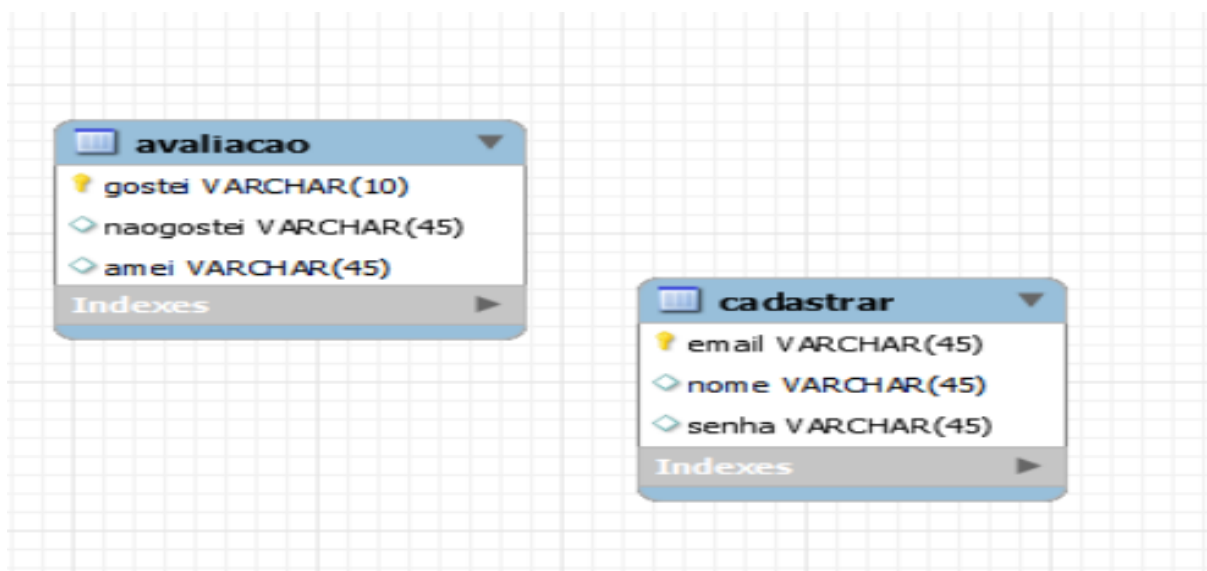
Na tela de Home, tem a opção de contato, ao clicarmos nela, conseguimos visualizar a Tela de Contato, onde informamos o local que estamos localizados e o nosso contato, juntamente com um mapa.




## 6.2. BANCO DE DADOS

No banco de dados MYSQL o nome para identificação da empresa foi colocado como petshop e as tabelas são avaliação e cadastrar.

Diagrama do banco de dados MYSQL:



Código do banco de dados MYSQL:

 Name:  Specify the name of the schema here. You can use any combination of ANSI letters, numbers and the underscore character

Refactor model, changing all references found in view, triggers, stored procedures and functions from the old schema name to the new one.

Charset/Collation:   The character set and its collation selected here will be used when no other charset/collation is set for a database object (it

1 • `SELECT * FROM petshop.avaliacao;`

gostei	naogostei	amei
Gostei	Gostei	Gostei
Não gostei	Não gostei	Não gostei
NULL	NULL	NULL

1 • `SELECT * FROM petshop.cadastrar;`

email	nome	senha
deiton@gmail.com	deiton	123
ellizabetessantosh@gmail.com	Elizabete	123
teste@teste.com	teste	teste
NULL	NULL	NULL

O banco de dados foi criado no modo Low Code, simplificando sua criação e minimizando o tempo gasto no projeto, ainda assim a eficácia dos dados armazenados e gerados nesse método de programação foi garantido.

### 6.3. CÓDIGOS DO SISTEMA

O sistema foi produzido no método low code, facilitando e reduzindo linhas complexas de código, ainda assim foram realizadas algumas linhas de códigos essenciais para o funcionamento do sistema, abaixo estará disponível alguns códigos do sistema. Obs: “não está listado o código de todas as telas pelo motivo da base ser a mesma e apenas mudar algumas variáveis”. Segue códigos abaixo:

Este código representa uma ação no botão Entrar da tela Login, conectado no banco de dados, o código avalia se há dados registrado do usuário no banco de dados, caso não haja dados registrado, gera uma mensagem “Usuário e/ou senha invalido”.

```
private void btnEntrarActionPerformed(java.awt.event.ActionEvent evt) {  
    try {  
        Class.forName(className:"com.mysql.cj.jdbc.Driver");  
        Connection conectado = DriverManager.getConnection(url:"jdbc:mysql://localhost:3306/petshop", user:"root", password:"elizabeth");  
        PreparedStatement st = conectado.prepareStatement(string:"SELECT * FROM cadastrar WHERE email = ? AND senha = ?");  
        st.setString(1, string:txtEmail.getText());  
        st.setString(2, string:txtSenha.getText());  
        ResultSet resultado = st.executeQuery();  
        if (resultado.next()){  
            dispose();  
            new Home().setVisible(b: true);  
        } else {  
            JOptionPane.showMessageDialog(parentComponent: null,message: "Usuário e/ou senha inválido");  
        }  
    } catch (ClassNotFoundException x) {  
        JOptionPane.showMessageDialog(parentComponent: null, "Driver JDBC não encontrado " + x.getMessage());  
    } catch (SQLException x) {  
        JOptionPane.showMessageDialog(parentComponent: null, "Erro na conexão com o banco de dados " + x.getMessage());  
    }  
}
```

Na imagem a seguir mostra o código do Cadastro de usuário que também é conectado com o banco de dados. Esse código juntamente com o banco de dados impede que um usuário faça 2 cadastros iguais, caso ocorra do usuário tentar cadastrar um Email já cadastrado, o sistema gera um catch de comunicação informando “Este Usuário já está cadastrado”.

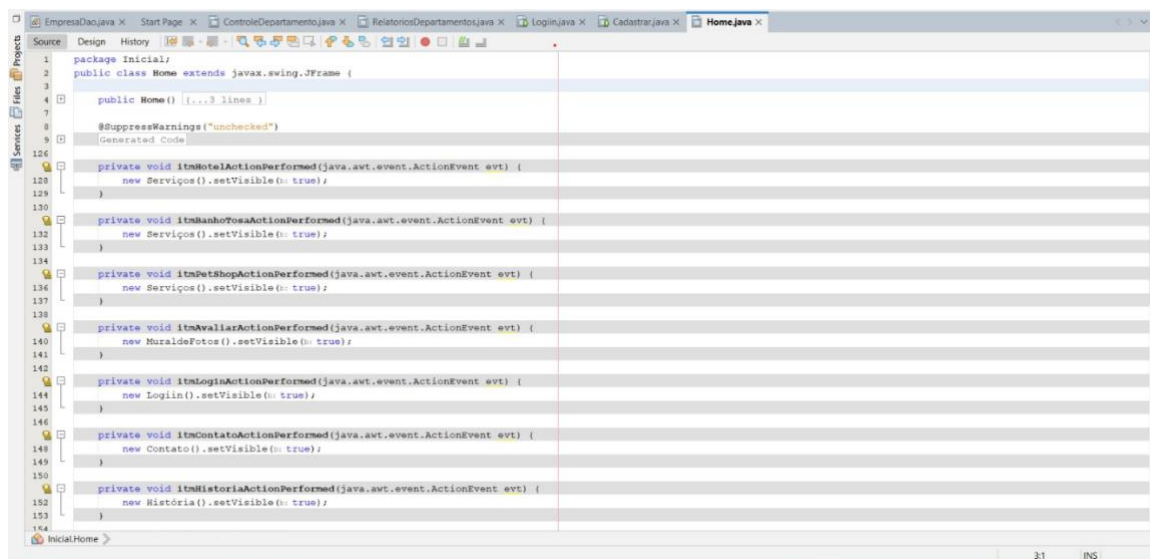


```

private void btnCadastrarActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        Class.forName(className: "com.mysql.cj.jdbc.Driver");
        Connection conectado = DriverManager.getConnection(url: "jdbc:mysql://localhost:3306/petshop", user: "root", password: "elizabeth");
        PreparedStatement st = conectado.prepareStatement(string: "INSERT INTO cadastrar VALUES(?, ?, ?)");
        st.setString(1: 1, (txtEmail.getText()));
        st.setString(2: 2, string: txtNome.getText());
        st.setString(3: 3, string: txtSenha.getText());
        st.executeUpdate(); //executa o insert
        JOptionPane.showMessageDialog(parentComponent: null, message: "Usuário cadastrado com sucesso");
        //Limpa os Campos
        //btnExcluir.setVisible(true);
        txtEmail.setText("");
        txtNome.setText("");
        txtSenha.setText("");
        //Posiciona o cursor dentro do campo usuário
        txtEmail.requestFocus();
    } catch (ClassNotFoundException x) {
        JOptionPane.showMessageDialog(parentComponent: null, "Driver JDBC não encontrado " + x.getMessage());
    } catch (SQLException x) {
        if (x.getMessage().contains(s: "Duplicate entry")) {
            JOptionPane.showMessageDialog(parentComponent: null, "Este Usuário " + txtEmail.getText() + " já está cadastrado");
        } else {
            JOptionPane.showMessageDialog(parentComponent: null, "Erro na conexão com o banco de dados " + x.getMessage());
        }
    }
}

```

Na tela a seguir mostra o código da tela Home que garante que todas as informações sejam visíveis e ao clicar em algum item menu ira te direcionar ao local desejado.



```

package Inicial;

public class Home extends javax.swing.JFrame {

    public Home() {
        //...3 lines...
    }

    @SuppressWarnings("unchecked")
    // Generated Code

    private void btnHotelActionPerformed(java.awt.event.ActionEvent evt) {
        new Serviços().setVisible(true);
    }

    private void btnBancoFotosActionPerformed(java.awt.event.ActionEvent evt) {
        new Serviços().setVisible(true);
    }

    private void btnPetshopActionPerformed(java.awt.event.ActionEvent evt) {
        new Serviços().setVisible(true);
    }

    private void btnAvallarActionPerformed(java.awt.event.ActionEvent evt) {
        new MuraldeFotos().setVisible(true);
    }

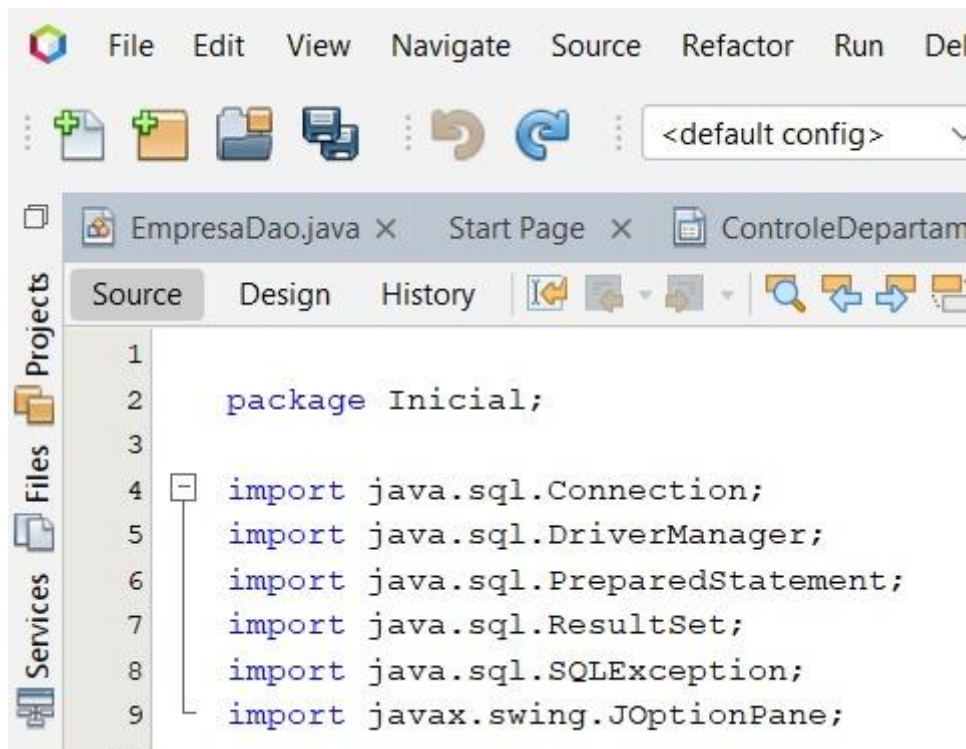
    private void btnLoginActionPerformed(java.awt.event.ActionEvent evt) {
        new Login().setVisible(true);
    }

    private void btnContatoActionPerformed(java.awt.event.ActionEvent evt) {
        new Contato().setVisible(true);
    }

    private void btnHistoriaActionPerformed(java.awt.event.ActionEvent evt) {
        new História().setVisible(true);
    }
}

```

A tela abaixo está relacionada aos importes necessários para funcionamento das ações do sistema, comunicação com o banco de dados etc.



## 7. CONCLUSÃO

Na elaboração do projeto foi criado uma empresa chamada Pet House no ramo de Pet shop. Foi definida a visão e missão da empresa. Também foi imposto um problema, e a solução foi o desenvolvimento de um sistema para interação do usuário e facilitar o atendimento de seus clientes. Na elaboração desta nova etapa da empresa, a modelagem de processos da empresa foi redesenhada solucionando os problemas anteriores. Para o desenvolvimento do sistema foi implementado a metodologia Scrum, organizando o projeto desde o início e distribuindo funções, com reuniões diárias mantendo as informações sempre bem atualizadas até o fim do desenvolvimento do sistema. O sistema foi vinculado juntamente com o banco de dados armazenando e salvando informações colocadas no sistema do Pet shop. A inclusão desse novo sistema e seus processos na empresa teve um impacto positivo, pois antes a empresa

não tinha outra forma para oferecer seus serviços, e seu atendimento era limitado somente de forma presencial, dificultando a solicitação de um serviço, depois que o sistema foi implementado os clientes teve a capacidade de saber se o serviço requerido estava disponível, através do sistema online ou através do contato.

## 8. REFERENCIAS

AUGUSTO R. Faculdade de Tecnologia de São Paulo. **Metodologia Híbrida com Scrum e Kanban Para Desenvolvimento de Software**. Disponível em: <ads\_2021\_2\_augustoalbuquerqueis\_scrumbanmetodologiahíbrida.pdf(cps.sp.gov.br)>. Acesso em 28 de maio de 2023.

LEONARDO M. Instituto de Computação. **Diagrama de Classes**. Disponível em:< http://www2.ic.uff.br/~leomurta/courses/es1/aula7.pdf>. Acesso em 28 de maio de 2023.

WILLIAM P. FATEC-SP. **A metodologia Business Process Management (BPM) e sua importância para as organizações**. Disponível em: <http://www.fatecsp.br/dti/tcc/tcc00084.pdf>. Acesso em 28 de maio de 2023.



