

Universidade Federal do Amazonas - UFAM

Instituto de Ciências Exatas e Tecnologia - ICET

Programação Orientada a Objetos - ERE - 2020/01

ClinicVet

Equipe: RGS

Responsável: Alternei Brito

Ficha Técnica

Equipe Responsável pela Elaboração

Rafael Guedes da Silva

Versão 1.1 - Itacoatiara, dezembro de 2020

REGISTRO DE ALTERAÇÕES

Versão	Responsável	Data	Alterações
1.0	Rafael Guedes	04/12/2020	Desenvolvimento de introdução, objetivo.
1.1	Rafael Guedes	08/12/2020	Inserção de diagramas UML, modelo MER do banco e detalhes de implementação.

Dúvidas, críticas e sugestões devem ser encaminhadas por escrito para os seguintes endereços digitais: rafaelguedes@ufam.edu.br

Sumário

I.	Panorama.....	3
	Introdução.....	3
	Objetivos do sistema	3
	Objetivos do projeto	3
II.	Diagramas e Modelagem	4
	Modelo Entidade Relacionamento (MER)	4
	Modelo Lógico do banco.....	5
	Diagrama de Classes	5
III.	Detalhes de implementação	6
	Processo de desenvolvimento	6
	Integração com o banco de dados	6
	Interfaces Gráficas	6
IV.	Referências	11

Introdução

Este projeto visa desenvolver a versão beta de um sistema de clínica veterinária. Este é um trabalho para complemento de nota da disciplina Programação Orientada a Objetos, ministrada pelo docente Alternei de Souza Brito, da Universidade Federal do Amazonas, campus Instituto de Ciências Exatas e Tecnologia.

O projeto foi desenvolvido utilizando a linguagem Java, dentro dos princípios do paradigma de desenvolvimento orientado a objetos. A integração com o banco foi realizada utilizando MySQL.

Objetivos do sistema

O sistema tem objetivo de facilitar os processos dentro de uma clínica veterinária. E portanto oferecer funcionalidades como Cadastro de animal, marcação de consulta, execução de consulta e até marcação de exames.

Objetivos do projeto

- Implementar as classes que se fizerem necessárias.
- Desenvolver a interface gráfica para duas funcionalidades, “Cadastrar animal” e “Simular consulta”.
- Realizar integração do sistema com um servidor de banco de dados.

Diagramas e Modelagem

Modelo Entidade Relacionamento (MER)

A figura 1 representa o MER do banco de dados utilizado no sistema ClinicVet.

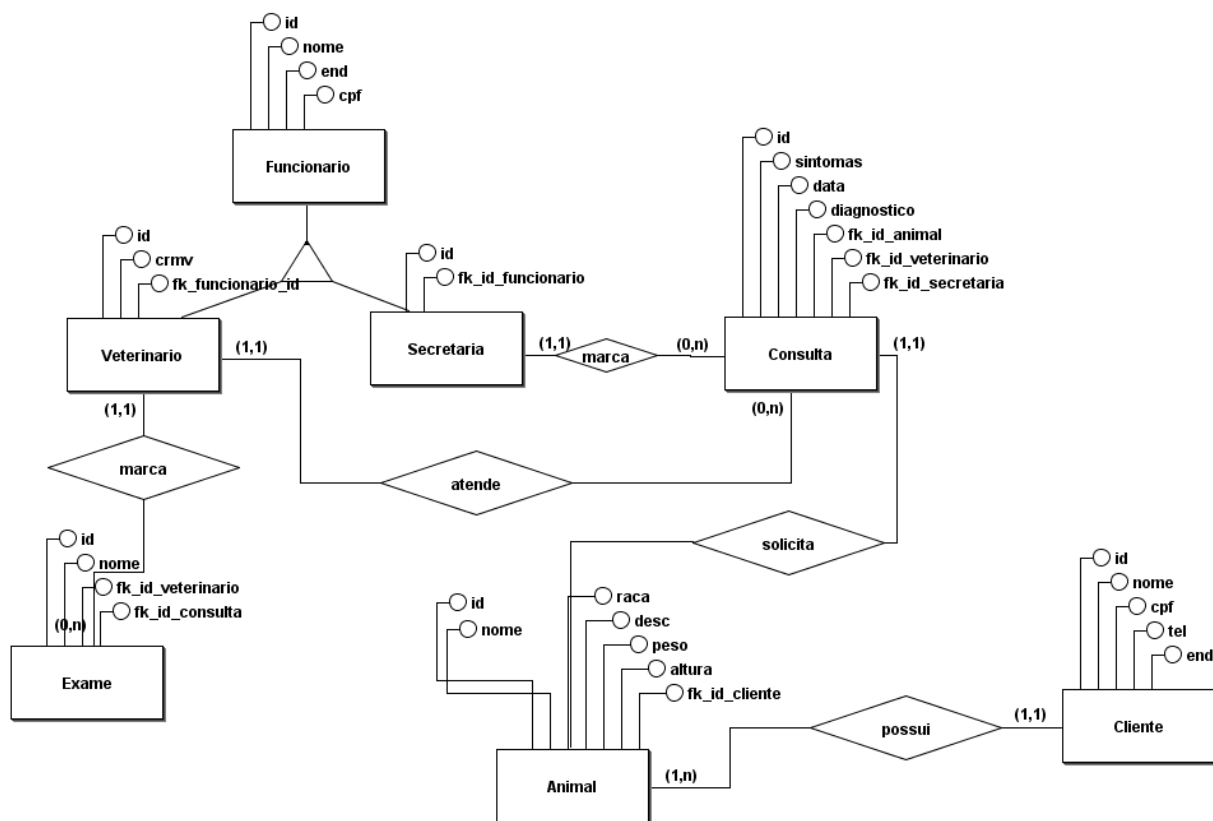


Figura 1 - Modelo Entidade Relacionamento

Modelo Lógico do banco

Na figura 2, se encontra o modelo lógico do banco de dados do sistema ClinicVet.

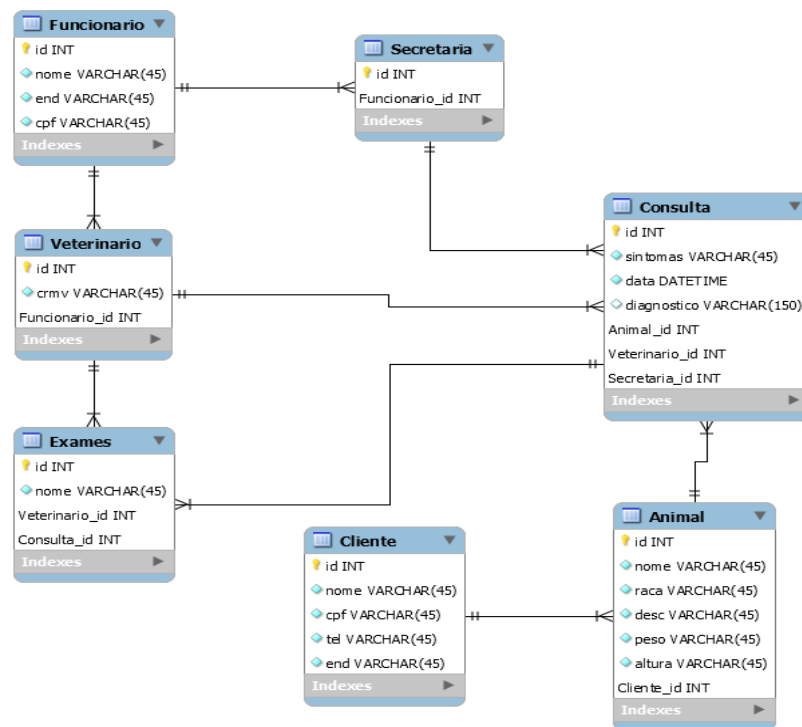


Figura 2 - Modelo Lógico

Diagrama de Classes

A figura 3 mostra o diagrama de classes do sistema, que representa as classes implementadas no decorrer do projeto e seus respectivos atributos e métodos disponíveis.

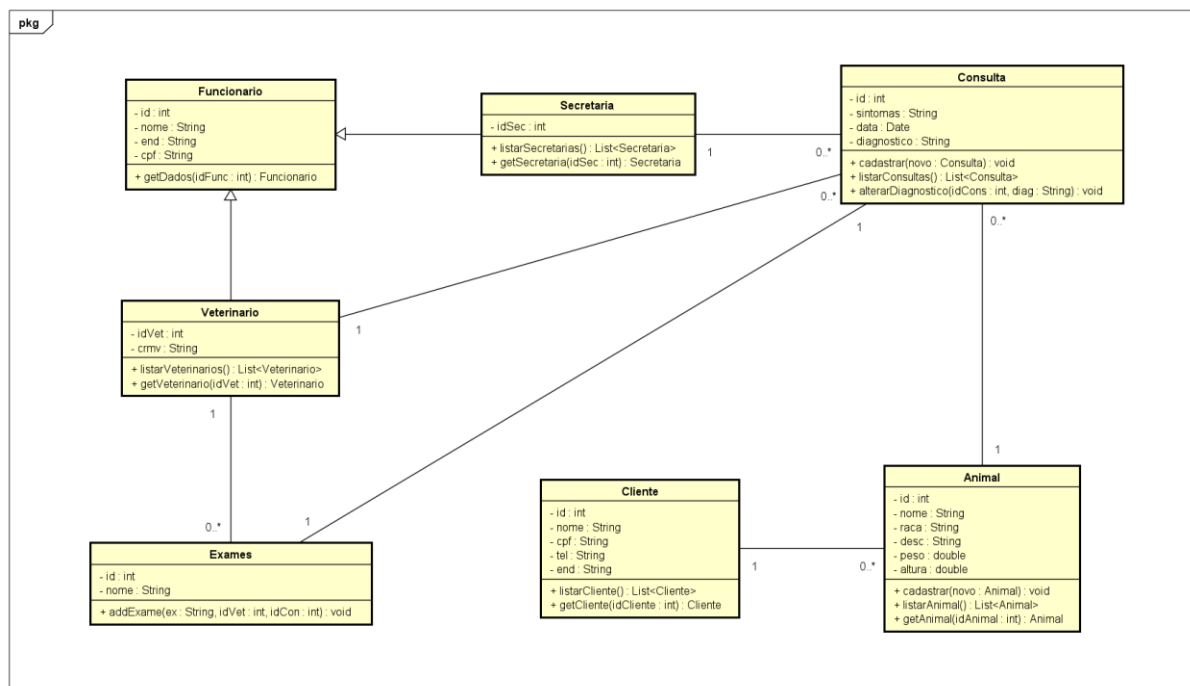


Figura 3 - Classes

Como o sistema foi desenvolvido em Java e os atributos são privados, já fica implícito o uso de Get e Set nas classes implementadas, e a fim de não causar problemas no entendimento do diagrama, os mesmos foram omitidos.

Detalhes de implementação

Processo de desenvolvimento

O ambiente de desenvolvimento escolhido para o projeto foi o NetBeans, IDE muito conhecida e utilizada no mercado para o desenvolvimento em Java.

Nesse projeto foram desenvolvidas as seguintes funcionalidades:

- ✓ Cadastrar animal, que salva os dados de um animal e vincula o mesmo a um cliente (dono).
- ✓ Marcar consulta, que registra data e hora para a consulta e vincula a mesma a um veterinário, um animal, e a secretaria que a marcou.
- ✓ E Simular consulta, onde o veterinário pode dar seu diagnóstico sobre uma consulta marcada, e se desejar pode marcar exames.

Integração com o banco de dados

A interligação do sistema com o banco foi realizada através do MySQL, e um plug-in, mysql-connector-java-5.1.49, instalado no NetBeans. Foram desenvolvidas classes de controle (DAO) para a implementação dos métodos de conexão com o banco de dados da aplicação, as mesmas foram instanciadas nas telas do sistema.

Interfaces Gráficas

As interfaces gráficas desenvolvidas para o projeto foram criadas utilizando a Paleta, ferramenta disponível no NetBeans. Ferramenta auxilia no uso e implementação de componentes do pacote Java Swing.

Foi desenvolvida uma logo para ilustrar a aplicação. O logo foi baseado nos objetivos do sistema ClinicVet. A mesma foi inserida em posição de destaque nas telas desenvolvidas. A figura 4, mostra a última versão da logo desenvolvida.



Figura 4 - Logotipo ClinicVet

Foi desenvolvido e implementado uma barra de menu, a fim de possibilitar a navegação do usuário através de todas as telas de funcionalidades disponíveis e dar a opção de sair do sistema. A figura 5, mostra como ficou a versão final do menu.

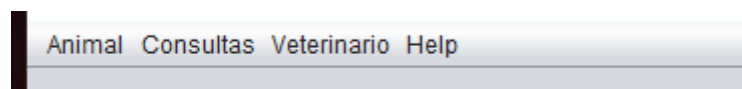


Figura 5 - Menu de navegação

A tela principal da aplicação e a primeira a ser desenvolvida foi a tela de Cadastro de animal. A mesma implementa um formulário para pegar os dados do animal e a qual cliente deve ser vinculado. A figura 6, traz um print da tela citada.

Cadastro de Animal

Animal Consultas Veterinario Help

ClinicVet

Nome

Descrição

Raça

Peso

Altura

Cliente

Hayden Yates

Cadastrar

Figura 6 - Tela 1 (Cadastro de Animal)

A segunda tela do sistema foi a de Marcar novas consultas, que possibilita ao usuário selecionar um veterinário, um animal, e a secretaria que serão vinculados a consulta, necessita que o usuário forneça os sintomas do animal e possibilita a escolha da data e do turno para a consulta. A figura 7, traz um print da tela citada.



Marcar consulta

Animal Consultas Veterinario Help

ClinicVet

Animal: Ada Lovelace

Veterinário: Aaron Lester

Atendente: Autumn Garrett

Sintomas

Data da Consulta: 

Turno: Matutino

Confirmar Voltar

Figura 7 - Tela 2 (Marcar consultas)

A última tela foi para a funcionalidade de Simular consulta, que possibilita a seleção entre uma das consultas marcadas que foram armazenadas no banco, e recebe o diagnóstico do veterinário referente a essa consulta. A figura 8, mostra a tela citada.

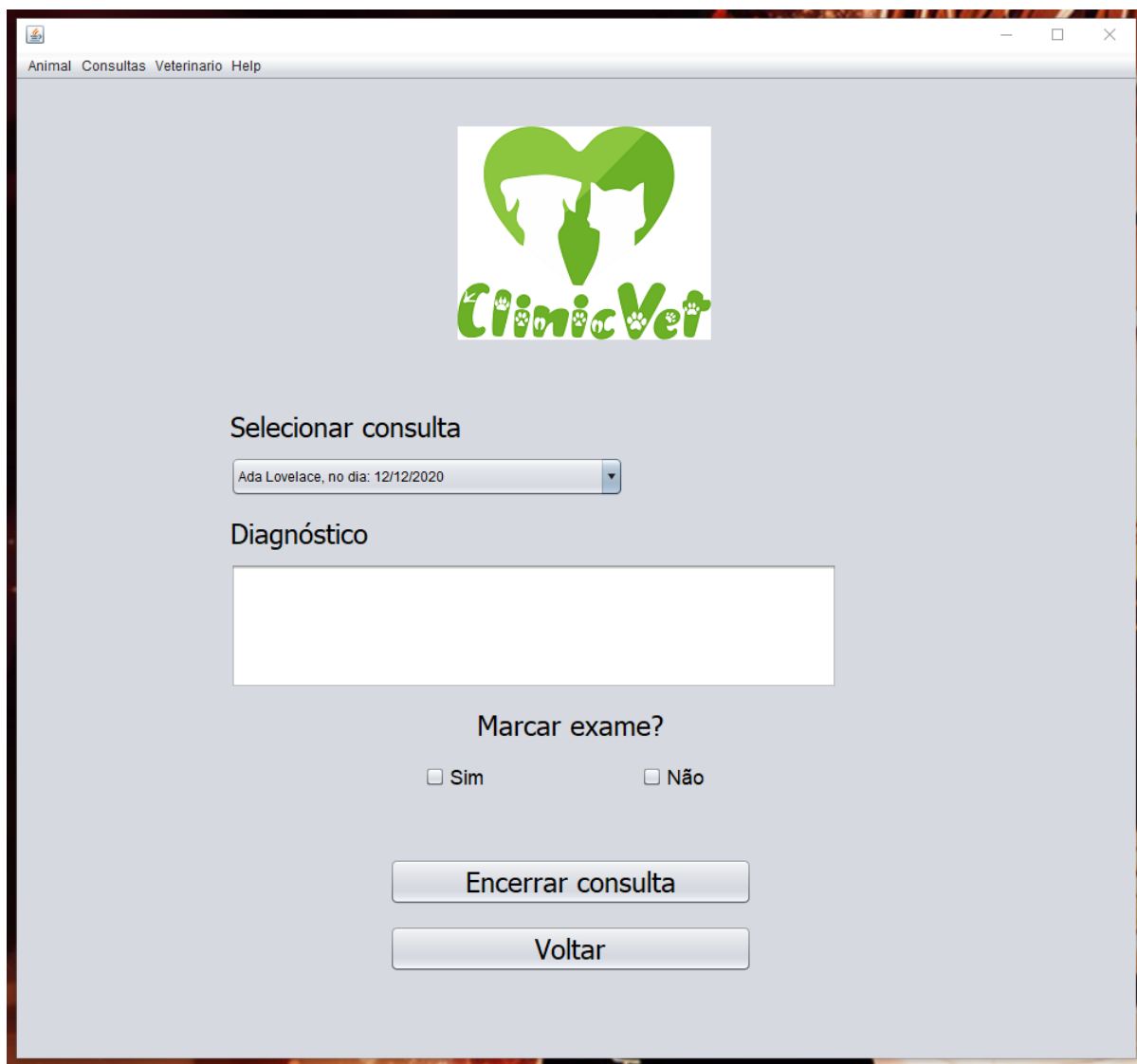


Figura 8 - Tela 3 (Simular consulta)

Ainda na última tela, o usuário precisa selecionar entre 'sim' ou 'não' para marcar exames, e se a seleção for 'sim', o sistema exibe uma caixa de entrada de texto solicitando o nome do exame, e vincula o mesmo a consulta selecionada. A figura 9, mostra a caixa de entrada.

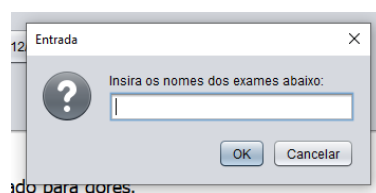


Figura 9 - Tela 3 (Caixa de entrada)

Todas as telas possuem caixas de diálogo para informação, confirmação de ação ou ocorrência de possíveis erros (Exceptions). Além disso, os campos de texto dos formulários podem apresentar dicas do que deve ser inserido no devido campo. A figura 10, dá exemplo de uma dessas dicas. E a figura 11, demonstra uma possível caixa de diálogo.

Figura 10 - Dicas de campo de formulário

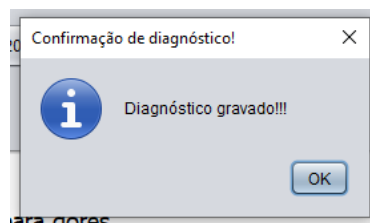


Figura 11 - Caixa de diálogo

Referências

SOMMERVILLE, Ian. Engenharia de Software. 9. ed. atual. São Paulo: Pearson Prentice Hall, 2011. 544 p. v. 3. ISBN 978-85-7936-108-1.

PRESSMAN, Roger S.; MAXIM, Bruce R.. Engenharia de software: uma abordagem profissional. 8 ed. Porto Alegre: AMGH, 2016.

DEITEL, Harvey M.; DEITEL, Paul J. Java: How to program. 11. ed. [S. l.]: Pearson, 2001. 1296 p. ISBN 9780134743356.

INTHURN, Cândida. Qualidade e Teste de Software. Florianópolis: Visual Books, 2001. 108 p.