FACUDADE DE TECNOLOGIA DA ZONA SUL – SÃO PAULO

“DOM PAULO EVARISTO ARNS”

AGUIRRE MATOS NOVAES

JEFFERSON DA ROCHA

LUCAS MARIANO DE PAULA

**DOLITTLE**

SISTEMA PARA GESTÃO DE HOSPITAL CLÍNICO VETERINÁRIO

**SÃO PAULO – SP**

**2025**

AGUIRRE MATOS NOVAES

JEFFERSON DA ROCHA

LUCAS MARIANO DE PAULA

**DOLITTLE**

SISTEMA PARA GESTÃO DE HOSPITAL CLÍNICO VETERINÁRIO

Trabalho de Conclusão de Curso, apresentado à Faculdade de Tecnologia Zona Sul – São Paulo, como exigência parcial para obtenção do título de Tecnnólogo em Análise e Desenvolvimento de Sistemas

**SÃO PAULO**

**2025**

**AGRADECIMENTOS**

**RESUMO**

Desenvolver um sistema de gerenciamento para clínicas veterinárias que atendam as necessidades gerais da atividade, tais como: Manter cadastro (animal, tutor), agendamentos, fichas de atendimento, gerenciamento de estoque e de serviços. O projeto visa viabilizar a implementação em qualquer tipo de clínica e se adequando a regras de negócio generalistas comuns a este ramo de atividade.

ABSTRACT

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

SUMÁRIO

INTRODUÇÃO

DESENVOLVIMENTO

**Paleta de Cores**

Forma

Descrição gerada automaticamente

**Logotipo**



**Teste de Cores Adobe Colors**

Cor Predominante:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Cores Secundárias:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**Teste HTML W3C**

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**Teste CSS W3C**

**Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente**

**Teste Acessibilidade AccessMonitorPlus**

**Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente**

**Requisitos Funcionais**

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Se refere | Ator |
| RF01 | O sistema deverá permitir manter a lista de opções de serviços disponibilizados. Os atributos são: Nome, Duração, Qtd. De Pessoas, Valor. | Serviços | Administrador |
| RF02 | O sistema deverá permitir manter funcionário. Os atributos são: Nome, Sobrenome, Data Nasc, CPF, Telefone, Função, CVI (Certificado Veterinário Internacional) e senha. | Funcionário | Administrador |
| RF03 | O sistema deverá permitir manter o estoque. Os atributos são: Nome, Validade, Fornecedor, Comentário. | Estoque | Administrador |
| RF04 | O sistema deve permitir o cadastro de tutor com os atributos: Nome, Endereço, CPF, Telefone, Email | Tutor | Recepcionista |
| RF05 | O sistema deve permitir o cadastro do animal com os atributos: Nome, Idade, Sexo, Castrado, Espécie, Raça, Peso, Porte | Paciente | Recepcionista |
| RF06 | O sistema deve permitir manter uma ficha de atendimento. Os atributos são: Nº da Ficha, Tutor, Paciente, Data de abertura, Data de alteração, Veterinário, Serviços e Observação. | Ficha | Recepcionista |
| RF07 | O sistema deverá permitir manter o controle de agendamento. Os atributos são: Nº da ficha, Data do agendamento, Horário, Veterinário. | Agendamento | Recepcionista |
| RF08 | O sistema deverá permitir enviar notificações dos agendamentos. | Notificação | Recepcionista |
| RF09 | O sistema deverá permitir o login de funcionários. Os atributos são: CPF e senha. | Login | Recepcionista |
| RF10 | O sistema deverá permitir o retorno dos atendimentos, cadastros, diagnósticos, pacientes e tutores relacionados. | Pesquisa | Recepcionista |
| RF11 | O sistema deverá permitir manter o diagnóstico na ficha. Os atributos são: Serviço (Nome, Duração, Qtd. De Pessoas) e Observação(diagnóstico). | Ficha | Veterinário |

**Requisitos Não Funcionais**

|  |  |
| --- | --- |
| Nº | Descrição |
| RNF01 | O sistema será desenvolvido utilizando a plataforma Java |
| RNF02 | Para persistência será utilizado o banco de dados SQL |
| RNF03 | A aplicação será desenvolvida inicialmente para o sistema operacional Windows, podendo ser portabilizada para demais SOs |
| RNF04 | O sistema de notificações utilizará API’s para envio de mensagens (sms, email, whatsapp) |
| RN05 | O usuário veterinário terá apenas privilégios para consultar e alterar um atendimento, incluir um diagnóstico em um atendimento, incluir serviço em um atendimento |

**Regras de Negócio**

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Se refere | Ator |
| RN01 | Apenas usuário administrador terá permissões para cadastro de tutor e animais | Tutor e Animal | Administrador |
| RN02 | Deverá haver apenas 1 instância do mesmo usuário logado no sistema | Login | Funcionário |
| RN03 | Deve haver apenas 1 agendamento por animal | Agendamento | Recepcionista |
| RN04 | Deve haver apenas 1 atendimento ativo por animal | Atendimento | Recepcionista |
| RNF05 | Apenas usuário administrador terá habilitado opções de alterar/excluir serviço de um atendimento | Atendimento/Serviço | Administrador |

**Caso de Uso**

Caso de uso: Login

Ator: Administrador/Recepcionista/Veterinário



Cenário ótimo:

1. O usuário insere suas credenciais de CPF e senha corretamente.
2. O usuário valida as credenciais no banco de dados.
3. O usuário tem acesso às funcionalidades respectivas ao seu cargo.

Cenário Alternativo:

1. O usuário insere o CPF incorreto
   1. O sistema apresenta uma mensagem descrevendo a inconsistência de CPF e solicita um CPF válido
2. O usuário insere uma senha inválida

1.1 O sistema apresenta uma mensagem descrevendo “senha incorreta” e solicita uma senha válido

Caso de Uso: Cadastro

Ator: Administrador / Recepcionista



Cenário ótimo:

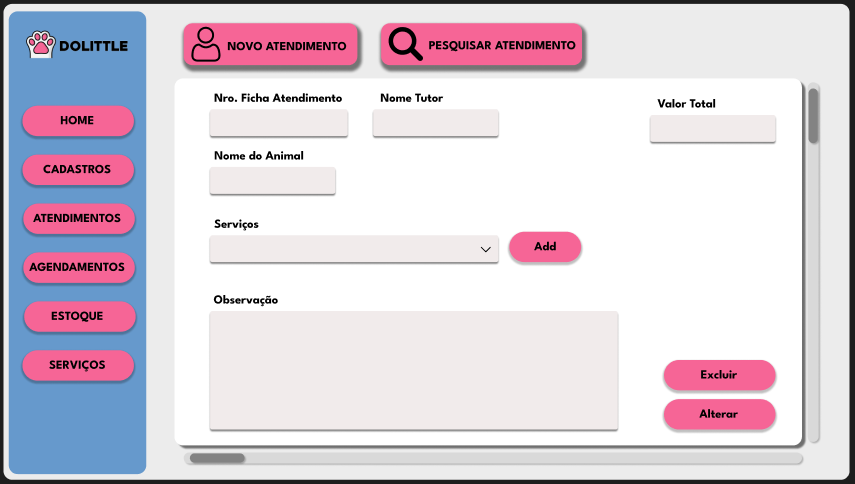
1. O usuário seleciona o botão cadastrar tutor caso seja sua primeira visita.
   1. Após efetivado o cadastro do tutor, ele seleciona o cadastro de um animal (poderá ser realizado cadastros de muitos animais para o mesmo tutor).
2. O usuário seleciona o botão buscar para listar especificamente um animal/tutor.
3. O usuário pode verificar uma lista de animais cadastrados no sistema e aplicar filtros para ordenar os resultados

Cenário Alternativo:

1. Caso o usuário insira dados incorretamente no formulário, o botão de cadastrar não ficará ativo, impossibilitando a efetivação do cadastro.
2. Caso haja algum dado inserido, será notificado no próprio campo na cor vermelha a informação incorreta ou o tipo de informação que deverá ser inserida.
3. Caso um cadastro seja realizado com inconsistência, o mesmo poderá ser alterado realizando uma busca do cadastro efetivado.

Caso de Uso: Atendimentos

Ator: Recepcionista/Veterinário



Cenário Ótimo:

1. O ator recepcionista abre um novo atendimento inserindo as informações do animal em um formulário, o atendimento é inserido no sistema após clicar no botão inserir. O sistema fornece automaticamente uma sugestão de horário, que pode ser alterada conforme as necessidades.

1.1 Caso necessário alteração de horário, clicando no ícone de procurar, será aberta uma janela sobreposta de agendamento com calendário.

1. Quando o atendimento é aberto pelo veterinário, ele fica associado a este usuário, e não mais aparece na lista de atendimentos em aberto. Para o veterinário possível apenas adicionar a opção de diagnostico (encerrar e reagendar somente recepcionista).

Cenário Alternativo:

1. Caso o animal já esteja inserido em um atendimento, ele não será listado na busca pelo nome.
2. Atendimentos com inconsistências só poderão ser modificados por um usuário com acesso administrador.

Caso de Uso: Agendamentos

Ator: Recepcionista



Cenário Ótimo:

1. O usuário registra um agendamento à partir da ficha de atendimento do paciente, selecionando a data no calendário e um horário.
2. Um usuário recepcionista/administrador pode alterar ou excluir agendamentos.
3. O usuário pode pesquisar uma lista de agendamentos ou verificar agendamentos no calendário interno da aplicação.

Cenário Alternativo:

1. Caso a data e/ou horário não esteja disponível, estará marcado em vermelho e desabilitado para seleção.

Caso de Uso: Estoque

Ator: Administrador/Veterinário



Cenário Ótimo:

1. O administrador cadastra um novo produto com os campos: nome, quantidade, fornecedor, validade do produto, e comentário.
2. Um usuário veterinário é capaz de dar baixa no estoque, associando produtos a atendimentos.

Cenário Alternativo:

1. Um produto tem sua quantidade zerada, impossibilitando a associação dele em atendimentos até um administrador atualizar este produto.

Caso de Uso: Serviços

Ator: Veterinário/Administrador

Cenário Ótimo:

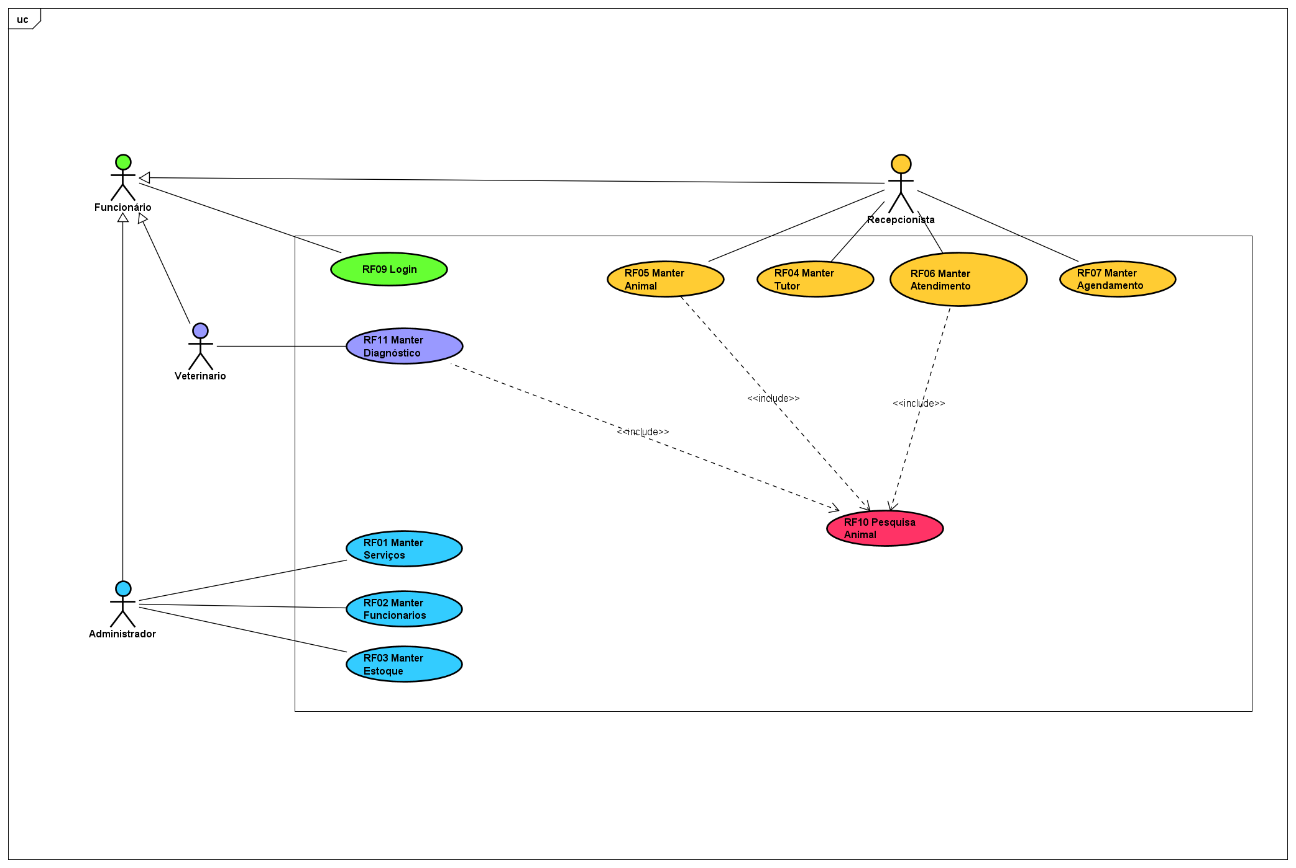
1. O usuário administrador é capaz de inserir novos tipos de serviços a serem prestados na clínica, habilitando essa nova opção em atendimentos.
2. Ao gerar um atendimento, o usuário veterinário associa todos os serviços a serem prestados na ficha de atendimento.

Cenário Alternativo:

1. Após um determinado serviço ser desabilitado pelo usuário administrador. Este serviço não aparecerá mais nas opções de serviços disponíveis em atendimentos para os usuários veterinários

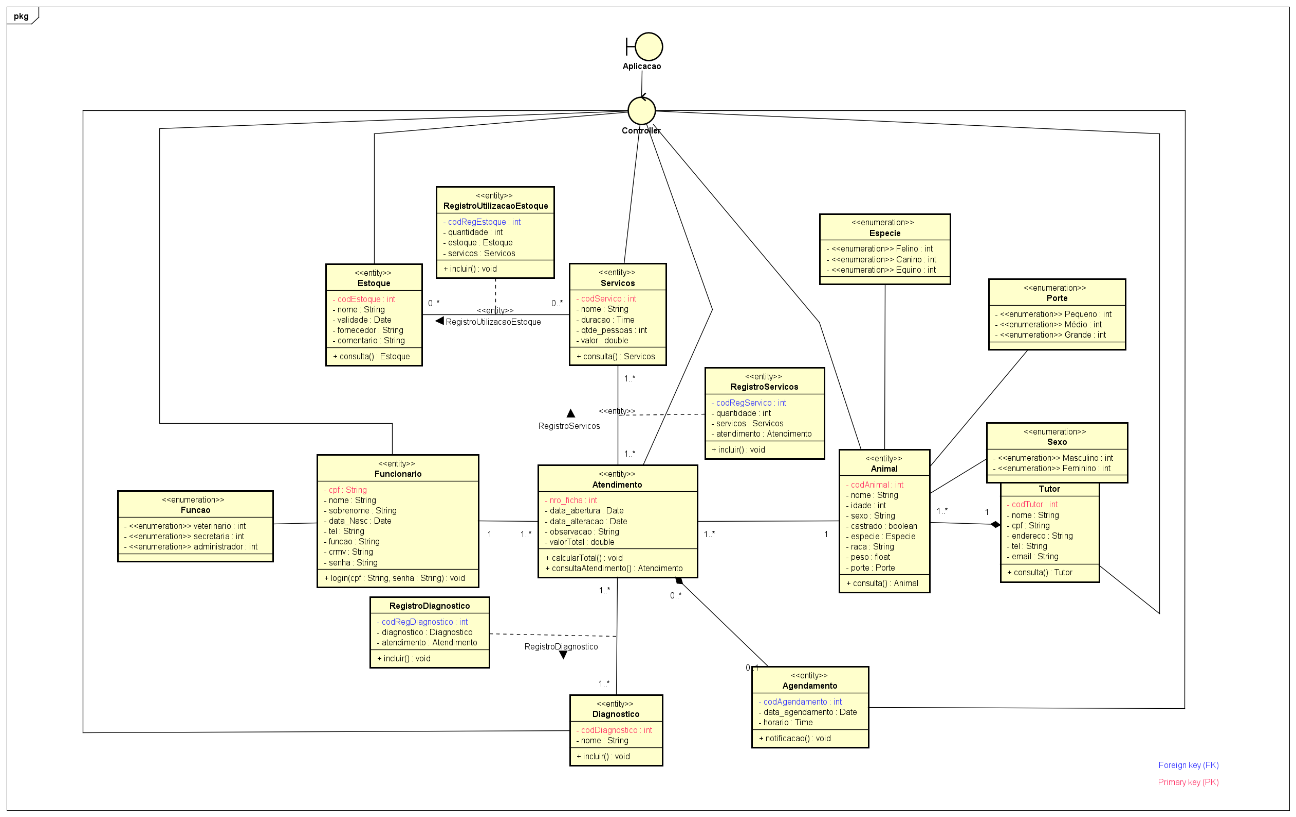
**Diagrama de Caso de Uso:**

O diagrama de caso de uso abaixo representa de maneira simplificada as funcionalidades que são executadas pela aplicação e seus respecitivos atores.



**Diagrama de Classes**

No diagrama de classes tempos uma abstração representando cada objeto da aplicação, suas instâncias, métodos e atributos.



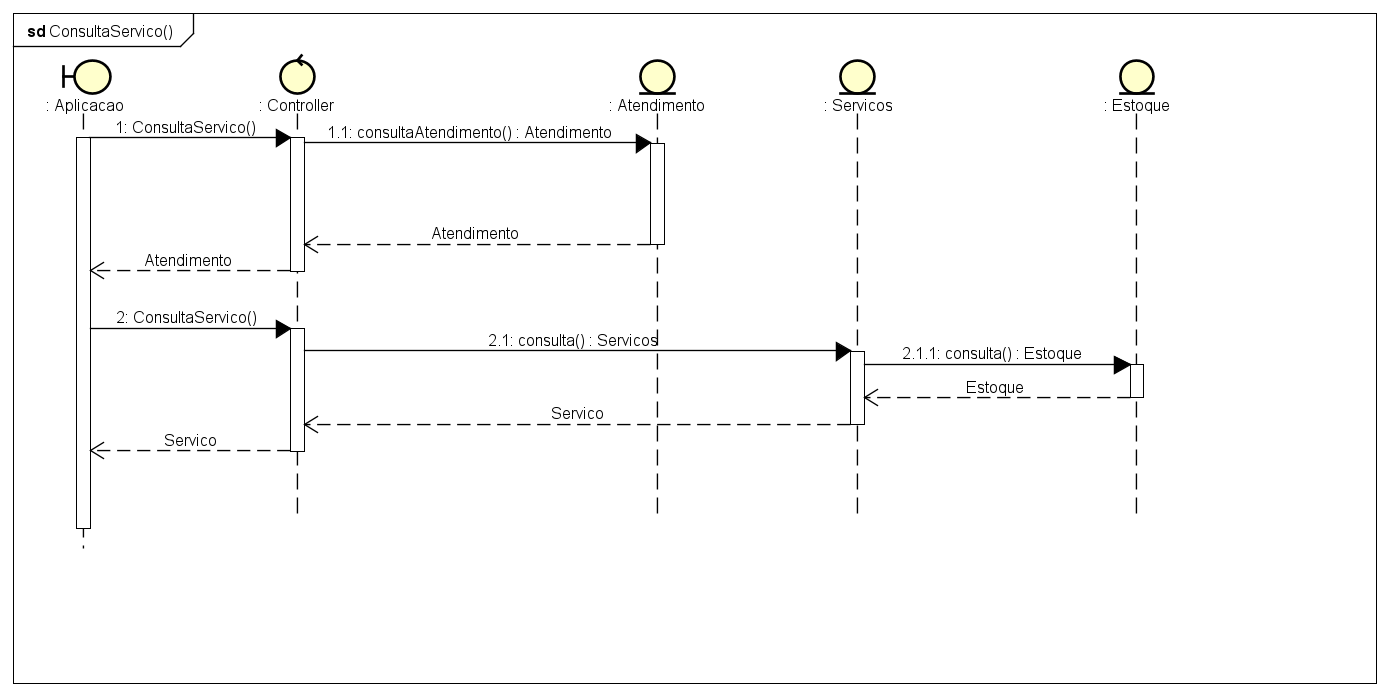
**Diagrama de Objetos e Diagrama de Sequência**

Os diagramas de objetos abaixo apresentam uma visão estática de um objeto instanciado em determinada operação da aplicação que será representada através do seu respectivo diagrama de sequência.

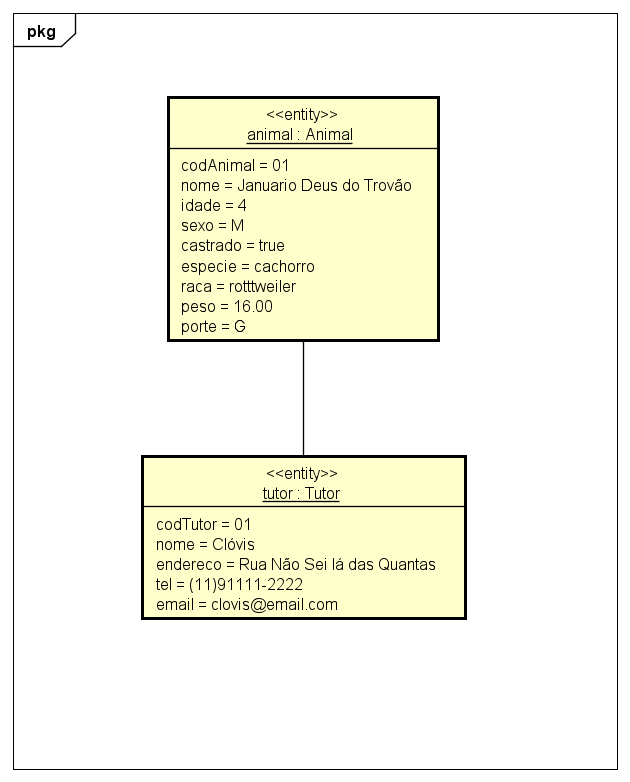
Consulta de um Serviço:

Uma imagem contendo Diagrama

Descrição gerada automaticamente

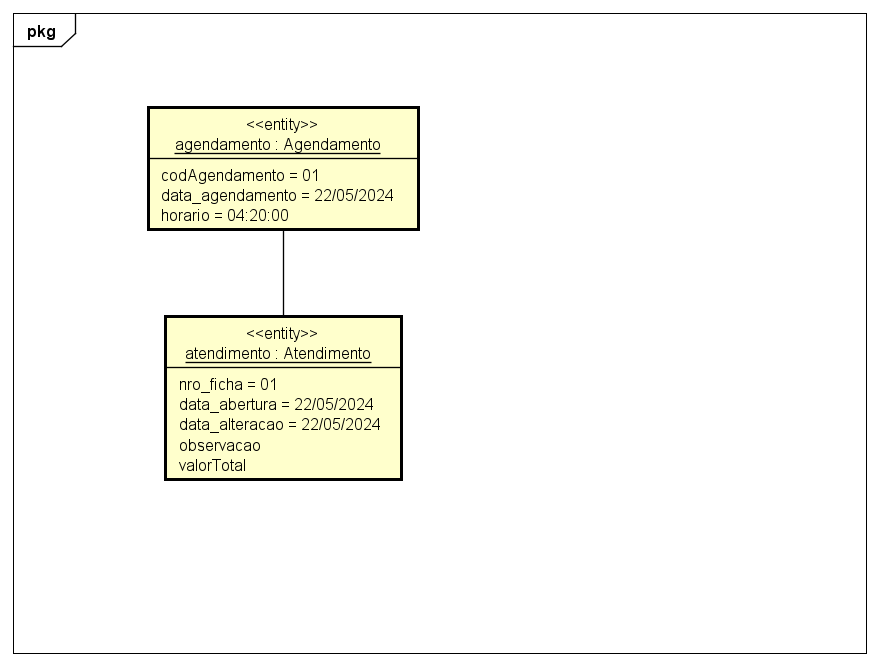
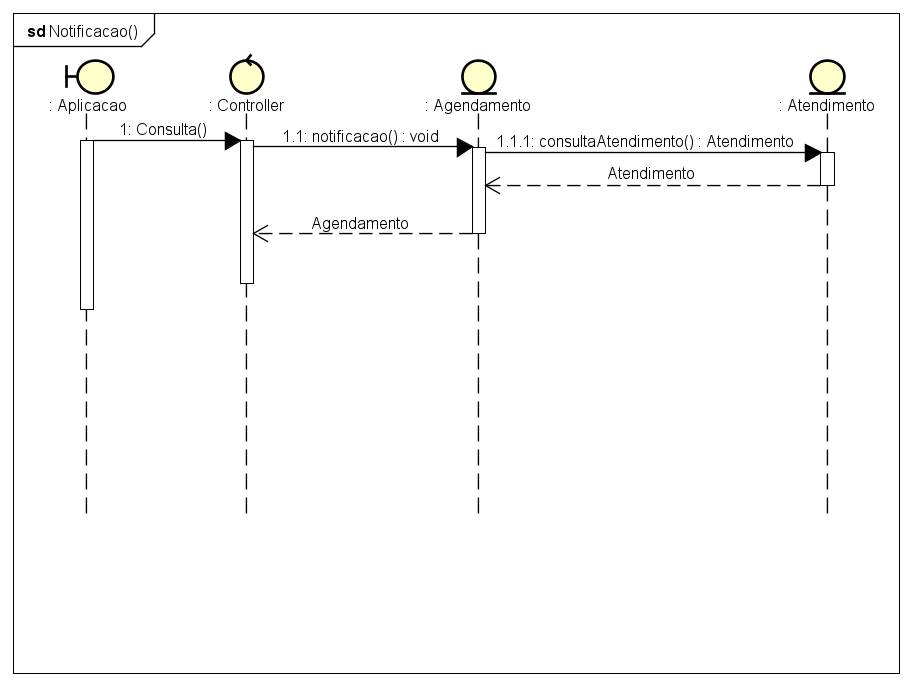


Consulta de um Animal:



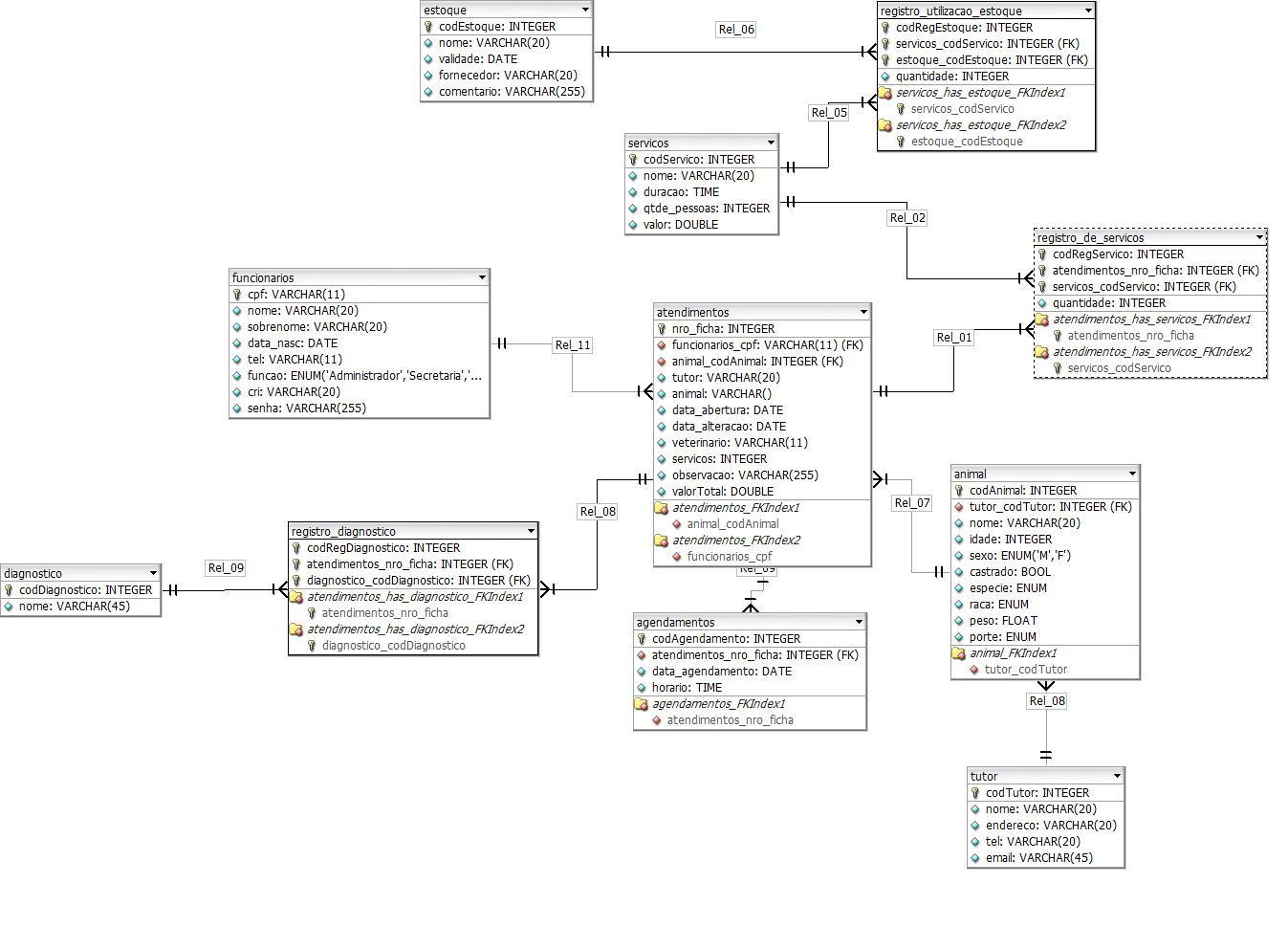
Diagrama

Descrição gerada automaticamente

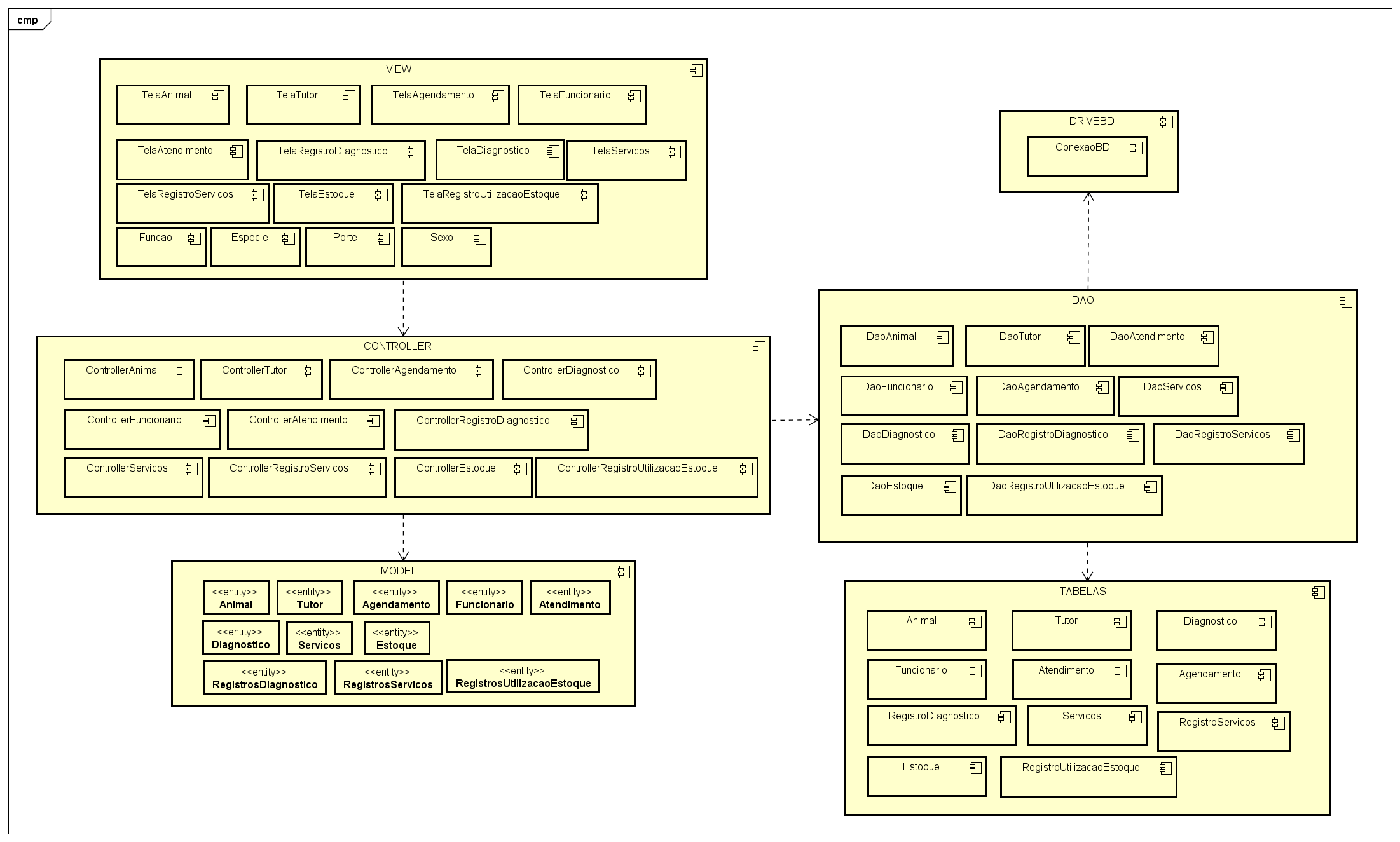
Envio de notificação de Atendimento:

**Modelagem de Dados**

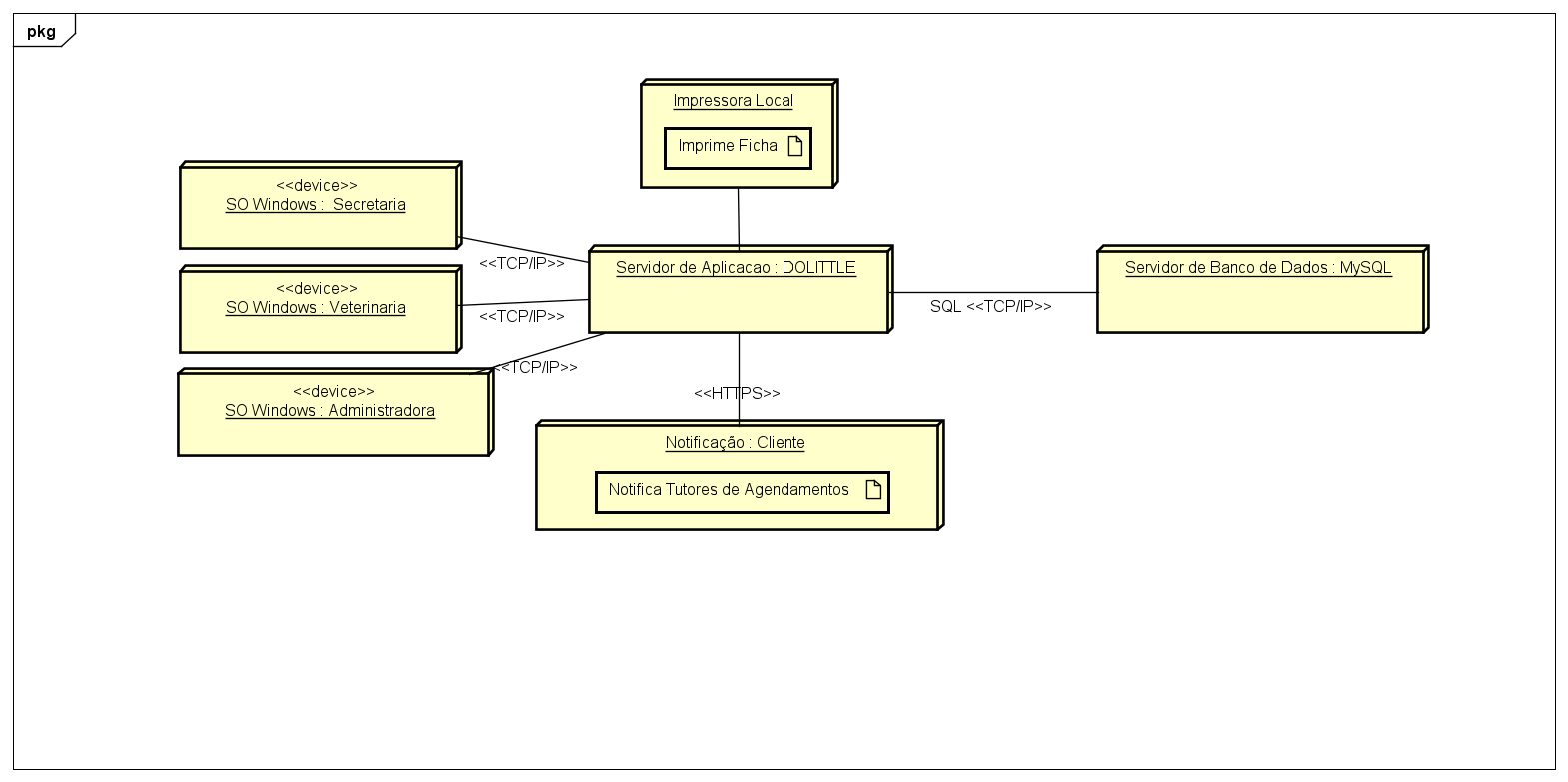
O modelo abaixo representa o conjunto de tabelas com atributos e relações de cada uma das instâncias da aplicação, junto com suas chaves identificadoras.



**Diagrama de Componentes**



**Diagrama de Implantação**



**REFERÊNCIAS BIBLIOGRÁFICAS**

Gilleanes T. A. Guedes. UML 2 Uma abordagem na prática. 2° Edição. Novatec.

Junho/2011.

Sommerville, Ian. (2011). Engenharia de Software, 9ª edição. São Paulo: Editora

Pearson.