

Documento de Requisitos

Sistema de Gerenciamento de Haras

14/12/2023

Versão 00.03

Letícia Raquel de Souza Nascimento

Alice Tabita de Freitas Pereira

Histórico de Alterações

Data	Versão	Descrição	Autor
10/11/2023	00.01	Draft inicial do documento.	Letícia
13/11/2023	00.02	Suplementar as partes dos requisitos, incluindo os diagramas de caso de uso, classe, sequência	Alice e Letícia
22/11/2023	00.03	Organizar o arquivo e correções	Alice e Letícia

Conteúdo

INTRODUÇÃO	3
VISÃO GERAL DO DOCUMENTO.....	4
CONVENÇÕES, TERMOS E ABREVIACÕES.....	4
<i>Identificação dos requisitos.....</i>	4
<i>Prioridades dos requisitos.....</i>	5
DESCRIÇÃO GERAL DO SISTEMA	5
CLIENTE.....	5
USUÁRIOS.....	5
VISÃO GERAL DO SISTEMA.....	5
REQUISITOS FUNCIONAIS	6
[RF001] Realiza Login e Senha.....	6
[RF 002] Cadastramento Proprietário.....	6
[RF 003] CRUD Histórico Médico.....	6
[RF 004] Lista Histórico Médico.....	6
[RF 005] CRUD Anuncios.....	6
[RF 006] CRUD Cobrança.....	6
REQUISITOS NÃO FUNCIONAIS	6
USABILIDADE.....	7
[NF001] Interface Amigável.....	7
SOFTWARE.....	7
[NF002] Arquitetura MVC.....	7
[NF003] API JPA-Hibernate.....	7
[NF004] Linguagem Java.....	7
[NF005] Banco de Dados MySQL.....	7
DESEMPENHO.....	8
[NF006] Agilidade na Execução das Operações.....	8
DIAGRAMAS DE CASOS DE USO	10
DIAGRAMAS DE CLASSE	11
DIAGRAMAS DE SEQUÊNCIA	12
DETALHAMENTO DOS CASOS DE USO	13
[CDU001] Cadastramento.....	13
[CDU002] Realiza Login e Senha.....	14
[CDU003] Histórico Médico do Equino.....	15
[CDU004] Permissões.....	16
[CDU005] Venda de Equino.....	16
[CDU006] Compra de Equino.....	17

<i>[CDU007] Visualiza Histórico Médico.....</i>	<i>18</i>
<i>[CDU008] Pagamento.....</i>	<i>19</i>

INTRODUÇÃO

Este documento especifica os requisitos do projeto para um sistema de Haras, fornecendo as informações necessárias para o projeto e implementação, assim como para a realização dos testes.

VISÃO GERAL DO DOCUMENTO

Além desta seção introdutória, as seções seguintes estão organizadas como descrito abaixo.

- **Seção 2 – Descrição geral do sistema:** apresenta uma visão geral do sistema, caracterizando qual é o seu escopo e descrevendo seus usuários.
- **Seção 3 – Requisitos funcionais :** especificam todos os cenários do Sistema.
- **Seção 4 – Requisitos não-funcionais:** especifica todos os requisitos não funcionais do sistema, divididos em requisitos de usabilidade, confiabilidade, desempenho e software.
- **Seção 5 – Diagramas de Caso de Uso:** especifica os atores e cenários utilizando a notação de diagramas UML.
- **Seção 6 – Detalhamento de Casos de Uso:** especifica a prioridade, fluxo principal e alternativo dos diagramas de caso de uso e sua relação com os requisitos funcionais e não funcionais.
- **Seção 7 – Referências:** apresenta referências para outros documentos utilizados para a confecção deste documento.

CONVENÇÕES, TERMOS E ABREVIACÕES

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

IDENTIFICAÇÃO DOS REQUISITOS

Por convenção, a referência a requisitos é feita através do identificador do requisito, de acordo com a especificação a seguir:

[identificador do requisito]

Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RF001] para os requisitos funcionais e [NF001] para os não funcionais e prossegue sendo incrementada à medida que forem surgindo novos requisitos.

PRIORIDADES DOS REQUISITOS

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações “essencial”, “importante” e “desejável”.

- **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.
- **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.
- **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

DESCRIÇÃO GERAL DO SISTEMA

Esta seção descreve superficialmente o cliente, os futuros usuários e fornece uma visão geral do *Sistema de Gerenciamento de Haras*.

CLIENTE

Serão os usuários do sistema: Proprietário(Cliente), Proprietário do equino e o funcionário do Haras.

USUÁRIO

No Sistema Haras, temos o proprietário(cliente), funcionário e proprietário do equino.

PROPRIETÁRIO - CLIENTE

No Sistema de Gerenciamento de Haras, temos o Proprietário(Cliente) o qual poderá consultar eventos e competições, acessar o sistema, visualizar histórico médico e tratamento de um equino, associar equinos a um proprietário, cadastrar/atualizar informações de acesso no sistema e realizar emissão de faturas e pagamentos.

FUNCIONÁRIO

O funcionário, gerencia as permissões de acesso, poderá remover um proprietário do sistema, deletar informações do equino, cadastrar/excluir/atualizar informações de tratamento dos equinos, poderá também registrar os pagamentos recebidos, crud eventos e competições.

PROPRIETÁRIO DO EQUINO

O proprietário do equino, poderá vender o cavalo no sistema, assim como também comprar um cavalo, consultar equinos à venda.

VISÃO GERAL DO SISTEMA

O objetivo do sistema de Gerenciamento de Haras, é promover o gerenciamento do Haras e de equinos, a fim de que o proprietário(cliente) e o proprietário do equino, possam encontrar no sistema, suporte e todas as informações necessárias para seu equino, tais como: Suporte veterinário contendo o histórico médico, onde o

proprietário poderá acessar as informações do seu equino, como: O nome do veterinário, o tratamento, o tipo do tratamento, agendar consultas, eventos e competições.

O sistema busca gerenciar as informações relacionadas ao equino de forma eficiente, garantindo a satisfação do proprietário(cliente) e do proprietário do equino.

REQUISITOS FUNCIONAIS

[RF001] REALIZA LOGIN E SENHA

Descrição: O usuário deve poder realizar o acesso na primeira tela do sistema, fornecendo as informações necessárias para o acesso.

[RF 002] CADASTRAMENTO PROPRIETÁRIO

Descrição: O usuário deve poder realizar cadastramento no sistema fornecendo as informações tais como: Nome, E-mail, Senha, CPF/CNPJ e o token de acesso.

[RF 003] CRUD HISTÓRICO MÉDICO

Descrição: O Veterinário(funcionário) do Haras, poderá realizar o cadastramento, excluir, editar no histórico médico. Nele é preenchido: Nome, data da consulta, proprietário, veterinário, tratamento.

[RF 004] LISTA HISTÓRICO MÉDICO

Descrição: O proprietário, poderá visualizar as informações abastecidas pelo veterinário do Haras.

[RF 005] CRUD ANÚNCIOS

Descrição: O proprietário(cliente) e proprietário do equino, poderá realizar o anúncio de vendas do equino no sistema, onde é solicitado a data do anúncio, poderá finalizar ou cancelar e informar o status do anúncio.

[RF 006] CRUD COBRANÇA

Descrição: O proprietário(cliente) e proprietário do equino, poderá realizar uma cobrança ao comprador, onde é informado o nome do proprietário, a referência, o custo total da cobrança, valor a ser cobrado, forma de pagamento e emite as notificações.

REQUISITOS NÃO FUNCIONAIS

USABILIDADE

Esta seção descreve os requisitos não funcionais associados à facilidade de uso da interface com o usuário e *help on-line*.

[NF001] INTERFACE AMIGÁVEL

O sistema terá uma interface amigável ao usuário primário sem se tornar cansativa aos usuários mais experientes.

Prioridade:

☐ Essencial

☒ Importante

☐ Desejável

SOFTWARE

Esta seção descreve os requisitos não funcionais associados aos softwares que devem ser utilizados para o desenvolvimento do sistema.

[NF002] ARQUITETURA MVC

A arquitetura MVC é um padrão de design amplamente utilizado no desenvolvimento de software para separar e organizar o código em três componentes principais: Model, View e Controller.

Prioridade:

☐ Essencial

☒ Importante

☐ Desejável

[NF003] API JPA- HIBERNATE

Conjunto com framework Hibernate.

Prioridade:

☐ Essencial

☒ Importante

☐ Desejável

[NF004] LINGUAGEM JAVA

Para desenvolver um produto facilmente expansível, reutilizável e flexível, é recomendável adotar a linguagem Java. Seguindo as técnicas orientadas a objetos.

Prioridade:

☒ Essencial

☐ Importante

☐ Desejável

[NF005] BANCOS DE DADOS MYSQL

Prioridade:

☒ Essencial

☐ Importante

☐ Desejável

DESEMPENHO

Esta seção descreve os requisitos não funcionais associados à eficiência, uso de recursos e tempo de resposta do sistema.

[NF006] AGILIDADE NA EXECUÇÃO DAS OPERAÇÕES

O sistema deve executar as operações no menor tempo possível, visando dar uma maior agilidade ao processo.

Prioridade:

☐ Essencial

☒ Importante

☐ Desejável

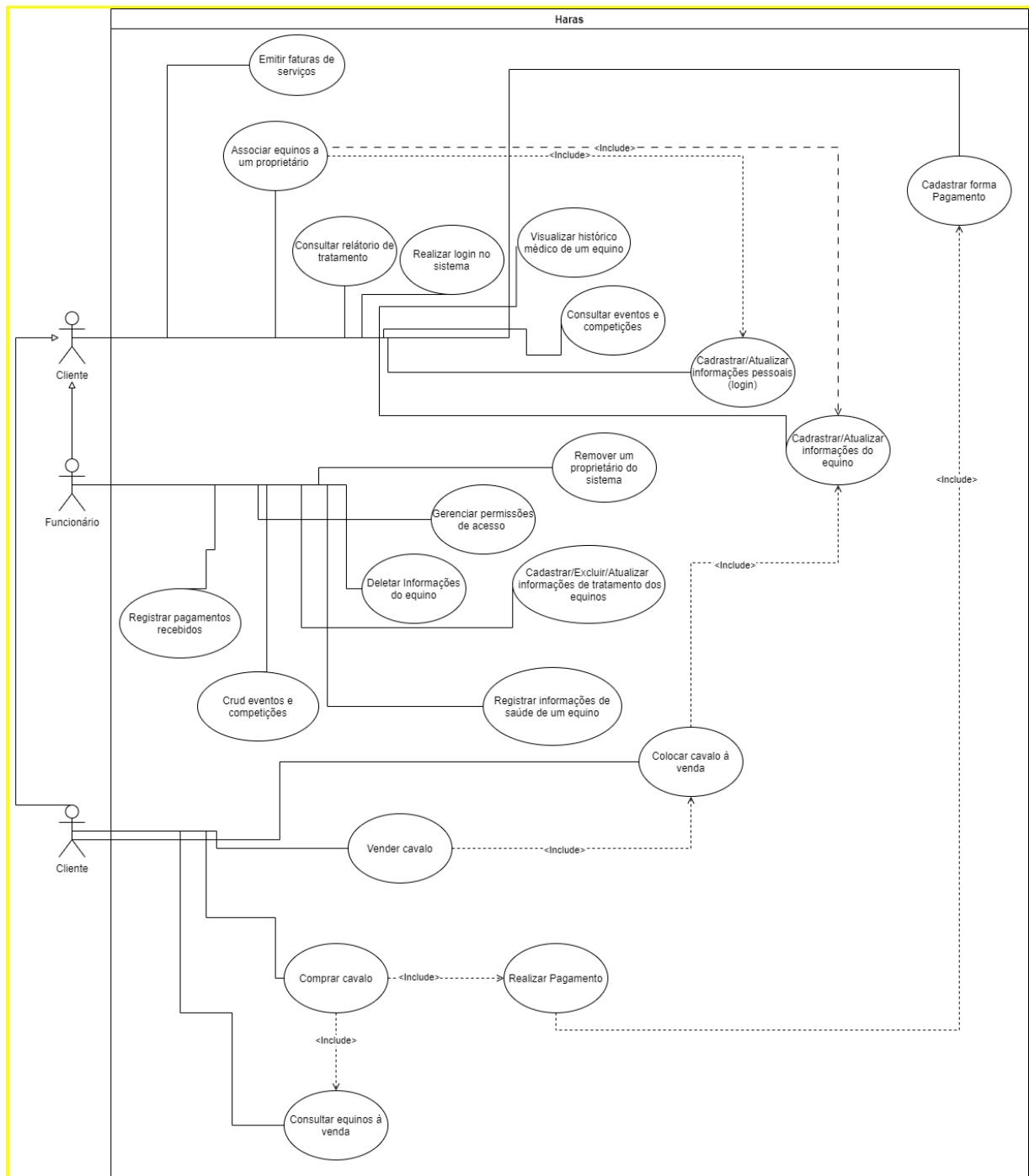


DIAGRAMA DE CLASSE

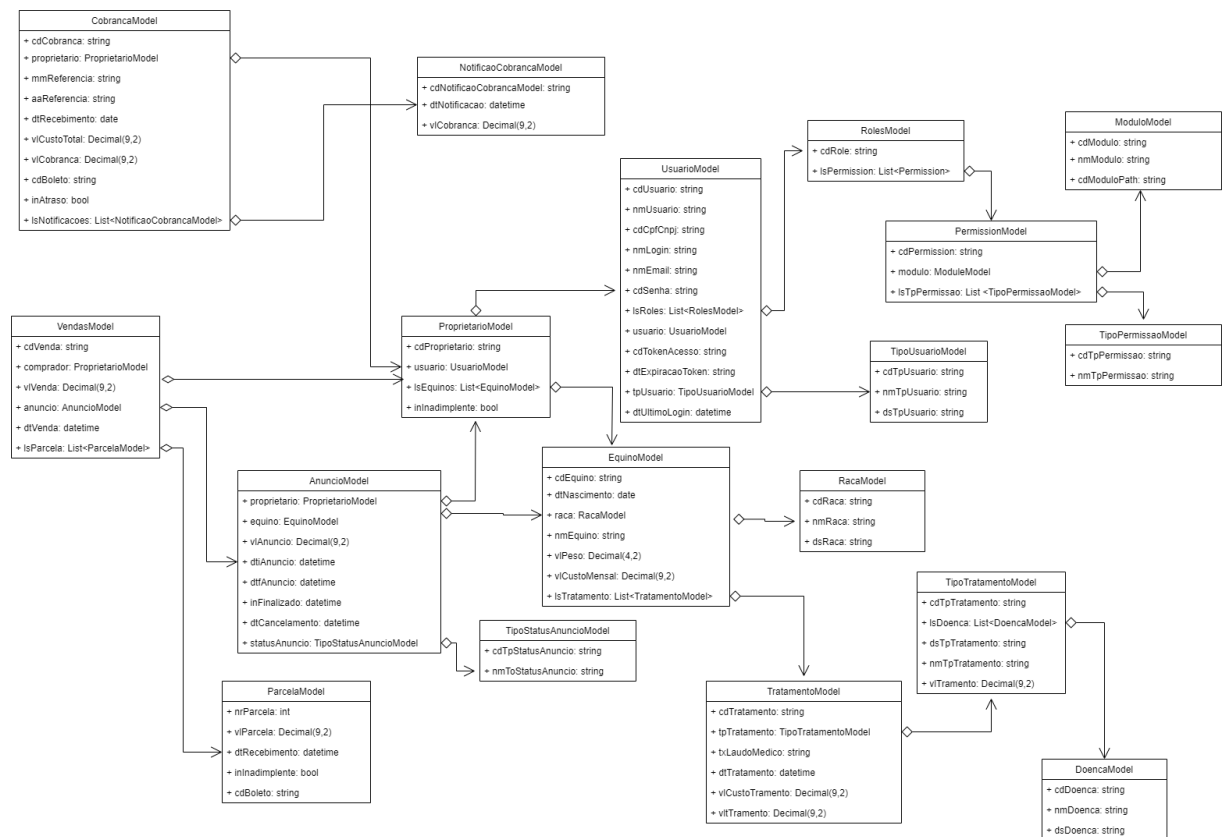
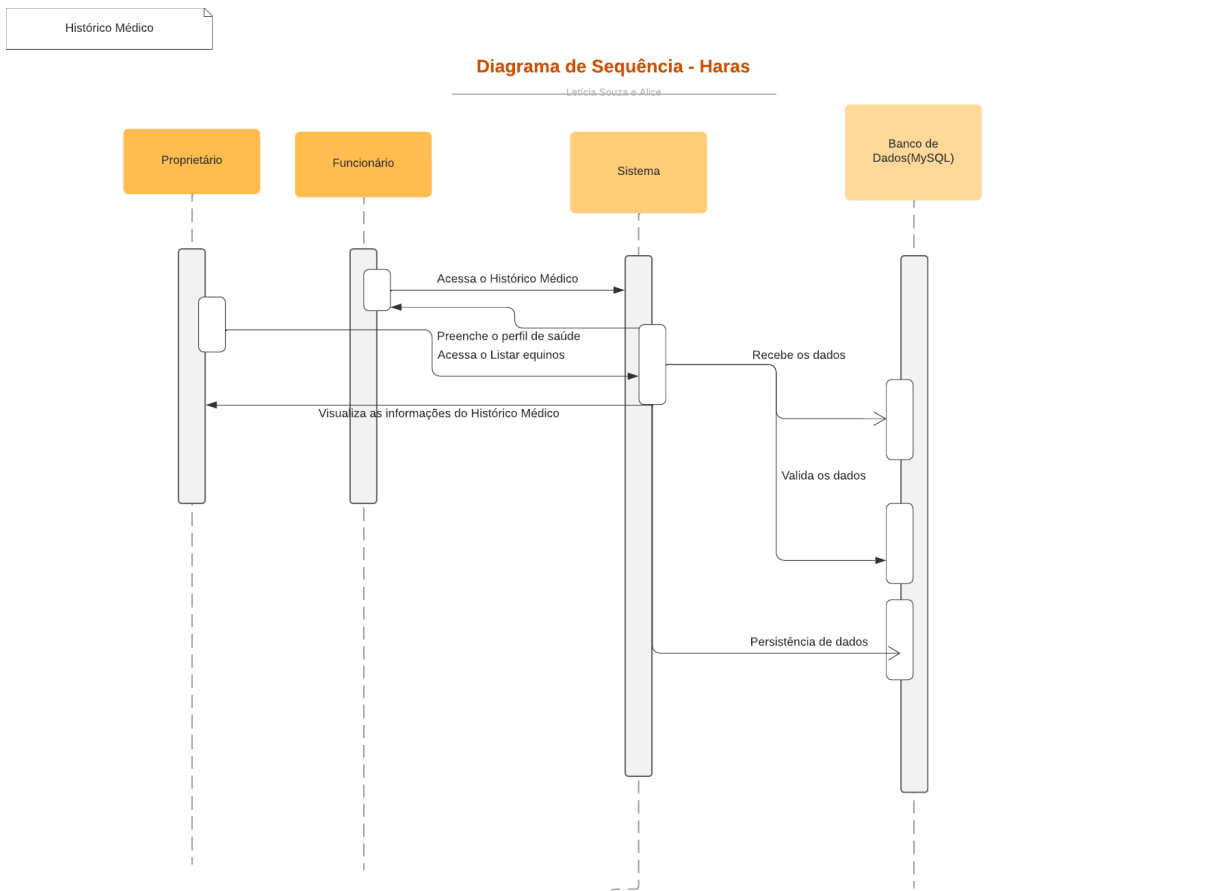
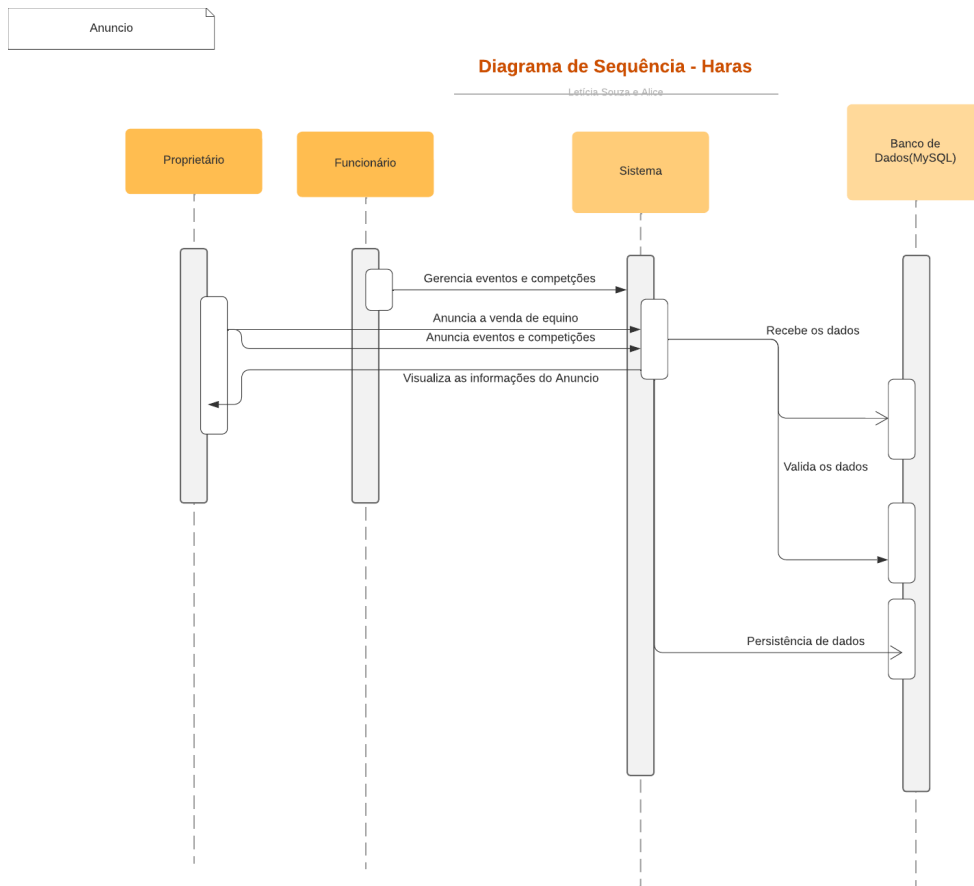


DIAGRAMA DE SEQUÊNCIA





DETALHAMENTO DOS CASOS DE USO

[CDU001] CADASTRAMENTO

Descrição do caso de uso: Os usuários realizaram o cadastro no sistema, para acessarem o sistema. É Solicitado: Nome, CPF/CNPJ, Email, Login e senha.

Ator: Proprietário (cliente), proprietário do equino.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições:

O usuário informa os dados no formulário de cadastro.

Saídas e pós-condição:

O sistema retorna uma mensagem informando o resultado da operação foi concluída com sucesso.

Fluxo de eventos principal

1. Usuário escolhe a ação cadastrar.
2. O sistema oferece o formulário de cadastro.
3. O usuário insere os dados essenciais para o cadastramento.
4. O sistema insere os dados submetidos no banco de dados.
5. O sistema informa o token para o cadastramento.
6. O usuário informa o código no sistema.
7. O sistema retorna para o usuário uma mensagem informando que a operação foi realizada com sucesso.

Fluxos secundários/exceção

1. Caso ocorra uma falha de comunicação com o banco de dados, uma mensagem de erro é retornada para o usuário e a operação é cancelada.

[CDU002] REALIZA LOGIN E SENHA

Descrição do caso de uso: Os usuários acessaram o sistema conforme o cadastramento no sistema, e utilizaram o cadastro de acesso.

Ator: Proprietário (cliente), proprietário do equino e funcionário

2.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

O proprietário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que o login foi concluído com sucesso.

Fluxo de eventos principal

1. O usuário fornece os dados de login e senha, o qual foi informado no cadastro.
2. O sistema analisa as informações.
3. O usuário, por fim entra no sistema.

Fluxos secundários/exceção

1. Caso ocorra um erro de login e senha, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro.

[CDU003] HISTÓRICO MÉDICO DO EQUINO

Descrição do caso de uso: O funcionário realiza a análise do equino, preenche o perfil de saúde e salva o perfil.

O funcionário pode: Excluir/Editar/Salvar o perfil de saúde do equino.

Ator: Funcionário

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

O funcionário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que o cadastro foi concluído com sucesso.

Fluxo de eventos principal

1. O usuário fornece os dados para o perfil de saúde do equino e salva.
2. O sistema analisa as informações.

Fluxos secundários/exceção

1. Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro .

[CDU004] PERMISSÕES

Descrição do caso de uso: O funcionário gerencia as permissões de acesso no sistema.

Ator: Funcionário

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

O funcionário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que o acesso foi liberado com sucesso.

Fluxo de eventos principal

1. O usuário verifica os dados de acesso.
2. O sistema analisa as informações.
3. O usuário libera a entrada.

Fluxos secundários/exceção

Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro e cancela a entrada.

[CDU005] VENDA DE EQUINO

Descrição do caso de uso: O proprietário do equino no sistema, entra na opção de Anuncio e informa o anúncio da venda do equino.

Ator: Proprietário do equino

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

O proprietário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que a venda foi concluída com sucesso.

Fluxo de eventos principal

1. O usuário acessa a aba de Anúncios.
2. O usuário informa os dados para a venda do equino.
3. O sistema analisa as informações e salva as informações.

Fluxos secundários/exceção

1. Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro e a venda é cancelada.

[CDU006] COMPRA DE EQUINO

Descrição do caso de uso: O proprietário(cliente) no sistema, entra na opção de Anúncio e verifica os equinos no sistema.

Ator: Proprietário(Cliente)

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

O proprietário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que a compra foi concluída com sucesso.

Fluxo de eventos principal

1. O usuário acessa a aba de Anúncios.
2. O usuário verifica os dados para a compra do equino.
3. O usuário solicita a compra.
4. O sistema analisa as informações e salva as informações.
5. O sistema emite uma notificação sobre o pagamento.

Fluxos secundários/exceção

1. Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro e a compra é cancelada.

[CDU007] VISUALIZA HISTÓRICO MÉDICO DO EQUINO

Descrição do caso de uso: Os usuários acessam a opção de Histórico de equinos, para visualizarem o histórico médico.

Ator: Proprietário(cliente) e proprietário do equino

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

Os proprietários devem ter acesso ao sistema.

Saídas e pós-condição:

O sistema informa a visualização do histórico médico.

Fluxo de eventos principal

1. O usuário acessa a aba de histórico de equinos.
2. O sistema exibe a tela.
3. O usuário acessa as informações.

Fluxos secundários/exceção

1. Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro.

[CDU008] PAGAMENTO

Descrição do caso de uso: O proprietário(cliente) entra no sistema, entra na opção de Anúncio e realiza o pagamento.

Ator: Proprietário(Cliente)

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições:

O proprietário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que o pagamento foi concluído com sucesso.

Fluxo de eventos principal

4. O usuário acessa a aba de anúncios..
5. O sistema informa sobre a notificação do boleto.
6. O usuário emite o boleto para o pagamento.
7. O usuário realiza o pagamento.
8. O sistema informa que a compra foi realizada com sucesso.

Fluxos secundários/exceção

2. Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro e cancela o pagamento.

[CDU009] PAGAMENTO

Descrição do caso de uso: O proprietário do equino informa no sistema o valor do equino e emite uma notificação.

Ator: Proprietário do equino

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições:

O proprietário deve ter acesso ao sistema.

Saídas e pós-condição:

O sistema retorna uma mensagem informando que a emissão do boleto foi concluída com sucesso.

Fluxo de eventos principal

9. O usuário cria a notificação do boleto com os dados.
10. O usuário envia para o cliente os dados para a compra do equino.
11. O sistema analisa as informações e salva as informações.
12. O usuário emite uma notificação para o cliente.
13. O usuário recebe a notificação para realizar o pagamento.

Fluxos secundários/exceção

3. Caso ocorra um erro, uma mensagem de erro é retornada para o usuário um alerta especificando qual a origem do erro.

