

UNIVERSIDADE FEDERAL DO PARANÁ

EDUARDO ANTONIO PERIN RAUTA

DESENVOLVIMENTO DE UM SISTEMA PARA GERENCIAMENTO BOVINO

CURITIBA

2016

EDUARDO ANTONIO PERIN RAUTA

DESENVOLVIMENTO DE UM SISTEMA PARA GERENCIAMENTO BOVINO

Monografia apresentada como requisito parcial à obtenção do grau de Especialista em Engenharia de Software, no Curso de Pós-Graduação em Engenharia de Software, Setor de Educação Profissional e Tecnológica, da Universidade Federal do Paraná.

Orientador: Prof. Dr. Razer A. N. R. Montaña

CURITIBA

2016



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
Setor EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
Curso de Pós-Graduação ENGENHARIA DE SOFTWARE

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em ENGENHARIA DE SOFTWARE da Universidade Federal do Paraná foram convocados para realizar a arguição da Monografia de Especialização de **EDUARDO ANTONIO PERIN RAUTA** intitulada: **Desenvolvimento de um sistema para controle bovino.**, após terem inquirido o aluno e realizado a avaliação do trabalho, são de parecer pela sua APROVAÇÃO.

Curitiba, 17 de Dezembro de 2016.

RAZER ANTHOM NIZER ROJAS MONTAÑO
Presidente da Banca Examinadora (UFPR)

JAIME WOJCIECHOWSKI
Avaliador Externo (UFPR)

RESUMO

O controle reprodutivo é o acompanhamento de todas as ocorrências reprodutivas ao longo da vida do animal. Ter um acompanhamento disto é de extrema importância, pois é possível identificar problemas e ineficiências. Desta forma, destaca-se a necessidade de desenvolvimento de sistemas computacionais, visando ajudar os produtores no gerenciamento e na tomada de decisões, uma vez que, registros manuais estão mais sujeitos a erros, podendo haver inconsistência nos dados registrados. A proposta deste trabalho foi desenvolver um aplicativo em Java, para executar nos sistemas Android, visando auxiliar o proprietário no gerenciamento e monitoramento em alto nível de rebanhos. A solução é baseada em um sistema Android que é responsável por realizar todas as operações referentes ao controle reprodutivo do animal, e através de um *web service*, sincronizar os dados do aplicativo com a internet. Este documento descreve os métodos adotados para o desenvolvimento de nossa solução, recursos necessários e ferramentas utilizadas.

Palavras-chave: Android. Web service. Gerenciamento bovino.

ABSTRACT

Reproductive control is the monitoring of all reproductive events throughout the animal's life. Having a monitoring it is important, because it is possible to identify problems and inefficiencies. Thus, there is the need for development of computer systems in order to help producers in the management and decision-making, since manual records are more subject to error, and there may be inconsistencies in the recorded data. The purpose of this study was to develop a Java application, to run on Android systems in order to assist the owner in managing and monitoring high level of cattle. The solution is based on an Android system that is responsible for performing all related to reproductive control of the animal operations, and through a web service, synchronize application data with the Internet. This document describes the methods adopted for the development of our solution, necessary resources and tools used.

Key-words: Android. web service. Cattle manager.

Lista de Ilustrações

Figura 1 - Modelo de ficha individual para registro zootécnico.....	14
Figura 2 - Vendas mundiais de smartphones para usuários por sistema operacional no primeiro trimestre de 2016 (Milhares de Unidades).....	17
Figura 3 - Camadas MVC.....	18
Figura 4 - Representação de um Web Service.....	19
Figura 5 - Representação de um JSON.....	20
Figura 6 - WBS do planejamento das atividades.....	22
Figura 7 - Cronograma das Atividades.....	23

Lista de Tabelas

Tabela 1 - Riscos e Plano de Ação.....	23
--	----

Lista de Apêndices

DIAGRAMA DE CASOS DE USO.....	35
DIAGRAMA DE CLASSES.....	36
MODELO LÓGICO BANCO DE DADOS.....	37
REGRAS DE NEGÓCIO.....	38
INTERFACES.....	41
CASOS DE USO.....	48
DIAGRAMAS DE SEQUÊNCIA.....	70
PLANO DE TESTES.....	78

LISTA DE ABREVIATURAS E SIGLAS

- API - Application Programming Interface
- EJB - Enterprise Java Beans
- JSON - JavaScript Object Notation
- REST - Representational State Transfer
- SDK - Software Development Kit
- XML - eXtensible Markup Language

Sumário

1 INTRODUÇÃO.....	13
1.1 JUSTIFICATIVA.....	13
1.2 OBJETIVOS.....	14
1.2.1 Objetivo Geral.....	14
1.2.2 Objetivos Específicos.....	14
2 FUNDAMENTAÇÃO TEÓRICA.....	16
2.1 JAVA.....	16
2.1.1 Spring Framework.....	16
2.1.1.1 Spring Security.....	16
2.2 ANDROID.....	17
2.3 ARQUITETURA MVC.....	17
2.4 WEB SERVICES.....	18
2.4.1 REST.....	19
2.4.2 JSON.....	19
2.5 MYSQL.....	20
3 METODOLOGIA.....	21
3.1 ENGENHARIA DE SOFTWARE.....	21
3.1.1 Diagrama de Caso de Uso.....	21
3.1.2 Diagrama de Sequência.....	22
3.1.3 Especificação de Caso de Uso.....	22
3.2 PLANO DE ATIVIDADES.....	22
3.3 CRONOGRAMA.....	23
FONTE: O AUTOR (2016).....	23
3.4 PLANO DE RISCOS.....	23
4 APRESENTAÇÃO DO SOFTWARE.....	25
4.1 FUNCIONAMENTO DO SOFTWARE.....	25
4.1.1 Aplicativo para Android.....	25
4.1.2 Sistema Web.....	26
4.1.3 Webservice.....	26
4.2 VISÃO GERAL DO SISTEMA.....	26
4.2.1 Tela Login.....	26
4.2.2 Tela Cadastro.....	26

4.2.3 Tela Inicial.....	27
4.2.4 Gerenciar Animais.....	27
4.2.5 Perfil Animal.....	27
4.2.6 Editar Animal.....	27
4.2.7 Gerenciar Visitas.....	28
4.2.8 Editar Visita.....	28
4.2.9 Gerenciar Vacinas.....	28
4.2.10 Editar Vacina.....	29
4.2.11 Gerenciar Inseminações.....	29
4.2.12 Adicionar Inseminação.....	29
4.2.13 Editar Inseminação.....	29
5 CONSIDERAÇÕES FINAIS.....	31
REFERÊNCIAS.....	32
GLOSSÁRIO.....	33
APÊNDICES.....	35

1 INTRODUÇÃO

A propriedade agrícola é cada vez mais assunto de estudos, pois se encontra em um momento de transição para utilização de recursos tecnológicos na área de gerenciamento administrativo, substituindo as práticas tradicionais. Segundo LOPES (1998), para que um rebanho seja bem administrado, é necessário o conhecimento do maior número possível de fatos que ocorrem na propriedade, tais como as ocorrências reprodutivas, identificação de animais que repetem cios, conhecer o intervalo de partos das vacas, entre outros.

Tais informações são de extrema importância, pois, desta forma, problemas e defeitos poderão ser detectados e corrigidos. Sendo assim, destaca-se a necessidade de desenvolvimento de sistemas computacionais, visando ajudar os produtores no gerenciamento e na tomada de decisões, uma vez que, registros manuais estão mais sujeitos a erros, podendo haver inconsistência nos dados informados.

1.1 JUSTIFICATIVA

Segundo Lopes (LOPES, 2005), o controle reprodutivo é feito para evitar erros comuns no dia-a-dia, como, vacas vazias sendo consideradas como prenhes, vacas prenhes sendo considerada como vazias, previsões erradas da data de secagem e data de parto. Além de se ter um diagnóstico completo do seu rebanho, as anotações destas informações ajudam o proprietário na tomada de decisão. A Figura 1 demonstra um modelo de ficha individual para controle zootécnico do animal.

- a) Através de um levantamento bibliográfico, apresentar os principais temas sobre o manejo de rebanhos.
- b) Levantamento da documentação referente ao sistema proposto.
- c) Propor um sistema de manejo de rebanhos.

2 FUNDAMENTAÇÃO TEÓRICA

Para melhor compreensão da maneira como o sistema foi desenvolvido, se faz necessário abordar os principais conceitos e tecnologias utilizadas, além de, abordar o histórico de seus surgimentos.

2.1 JAVA

JAVA é uma linguagem de programação que não está vinculada a um sistema operacional específico, a qual pode ser acessada em qualquer computador independente do microprocessador e do sistema operacional específico utilizado.

O uso desta tecnologia se justifica pela sua portabilidade entre plataformas de hardware e software, ser gratuito, possuir recursos de processamento de banco de dados, tratamento de exceções, componentes de interfaces gráficas, e de redes cliente/servidor baseadas na Internet.

2.1.1 Spring Framework

Desenvolvido por Rod Johnson (JOHNSON, 2016) para diminuir os problemas encontrados ao utilizar a tecnologia de Enterprise Java Beans (EJB). O Spring Framework trabalha com o conceito de Injeção de Dependências que é uma forma de controle na qual um componente não instancia suas dependências mas o ambiente (contêiner) automaticamente as fornece.

A principal razão ao escolher o Spring Framework, é o fato de não interferir diretamente na modelagem da aplicação. Ele afeta a arquitetura e a forma como estes módulos interagem, do sistema, mas os objetos de negócio não sofrem muitos impactos.

2.1.1.1 Spring Security

Dentro da Spring existem vários projetos, entre eles, temos a ferramenta Spring Security, que possibilita uma autenticação forte, personalizável e ainda estabelece uma estrutura de controle de acesso que é o padrão para a segurança de aplicativos baseados na *framework* Spring (WINCH, 2012).

2.2 ANDROID

O Android é um ambiente aberto desenvolvido para dispositivos móveis, baseado no *kernel* Linux, e que permite a criação de aplicações por qualquer programador. O desenvolvimento dos aplicativos é realizado utilizando o Android SDK (GOOGLE, 2016), que possui as bibliotecas e ferramentas básicas para um aplicativo Android.

A Gartner (GARTNER, 2016), publicou uma nota sobre venda de *smartphones* durante o primeiro trimestre de 2016. O sistema operacional Android é responsável por abranger 84% do mercado, 8% a mais que o mesmo período de 2015.

Figura 2 - Vendas mundiais de smartphones para usuários por sistema operacional no primeiro trimestre de 2016 (Milhares de Unidades)

Operating System	1Q16 Units	1Q16 Market Share (%)	1Q15 Units	1Q15 Market Share (%)
Android	293,771.2	84.1	264,941.9	78.8
iOS	51,629.5	14.8	60,177.2	17.9
Windows	2,399.7	0.7	8,270.8	2.5
Blackberry	659.9	0.2	1,325.4	0.4
Others	791.1	0.2	1,582.5	0.5
Total	349,251.4	100.0	336,297.8	100.0

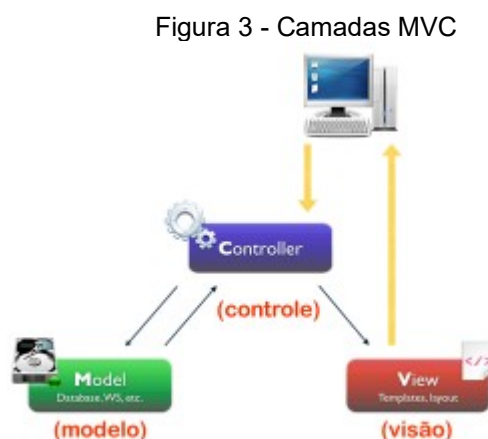
FONTE: Gartner (2016).

A escolha foi feita após uma análise de mercado e a plataforma do Google foi escolhida por apresentar algumas vantagens, sendo as principais: ser um sistema *opensource* e possuir uma grande comunidade de desenvolvedores, maior diversidade de dispositivos no mercado e existência de dispositivos de menor custo, desta forma, sendo possível abranger uma maior diversidade do mercado.

2.3 ARQUITETURA MVC

Com o objetivo de uma melhora na qualidade do desenvolvimento de software diversos padrões de desenvolvimento foram planejados e desenvolvidos. O

padrão MVC se compõe de 3 tipos de elementos arquiteturais (BUSCHMANN, 1996); a camada de Modelo, que representa o modelo do negócio; Controlador, que trata os eventos disparados pelos usuários na Visão; e Visão, que representa a interface com o usuário. A Figura 3 ilustra essas camadas e suas principais responsabilidades.



FONTE: <http://www.webartz.com.br/especiais/tecnologia-da-informacao/codeigniter-e-mvc-parte-01-mvc-conceito-e-funcionamento/>

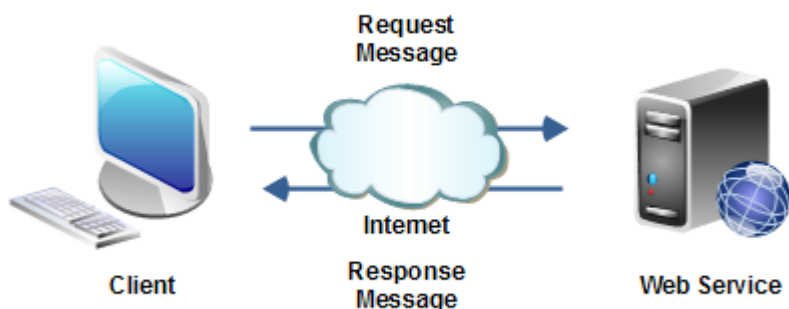
Um dos principais objetivos do padrão MVC é a organização do código de uma aplicação em camadas, realizando assim a separação física dos componentes do software. Com o aumento da complexidade das aplicações desenvolvidas torna-se fundamental a separação entre os dados, camada Model, e o layout, camada View. Desta forma, alterações feitas no layout não afetam a manipulação de dados, e estes poderão ser reorganizados sem alterar o layout.

2.4 WEB SERVICES

Web services são APIs web que podem ser acessadas através de uma rede. São utilizados na integração de sistemas distribuídos e na comunicação entre aplicações diferentes, independente da plataforma e linguagem utilizada (W3C, 2004).

A figura 4 ilustra o funcionamento de Web Service, em que um cliente envia uma mensagem de pedido (*Request*) a um Web Service, e recebe uma mensagem de resposta (*Response*).

Figura 4 - Representação de um Web Service



FONTE: http://www.criandobits.com.br/fs-programacao/fs_materias-webservice.php

Com esta tecnologia, torna-se possível que aplicações diferentes interajam entre si e sistemas desenvolvidos em plataformas diferentes se tornem compatíveis. Os Web services são componentes que permitem que aplicações enviem e recebam dados em formatos variados.

2.4.1 REST

Segundo Elkstein (ELKSTEIN, 2008), o REST é um estilo de arquitetura que em vez de usar mecanismos complexos para ligar máquinas entre si, usa o HTTP para fazer chamadas entre máquinas.

Os recursos em REST são representados principalmente em XML ou em JSON.

2.4.2 JSON

O JSON é um acrônimo para "*JavaScript Object Notation*", e é um formato de texto para a serialização de dados estruturados. O JSON pode representar quatro tipos primários (strings, números, booleanos e nulos) e dois tipos estruturados (objetos e vetores). Um objecto é uma coleção não ordenada de zero ou mais pares nome/valor, onde o nome é uma string e o valor é uma string, número, booleano, nulo, objeto ou vetor. Um vetor é uma sequência ordenada de zero ou mais valores (CROCKFORD, 2006). A Figura 5 mostra a representação da sintaxe de um JSON.

Figura 5 - Representação de um JSON

```
[  
  {  
    "id": 1,  
    "descricao": "Nelore"  
  },  
  {  
    "id": 2,  
    "descricao": "Wagyu"  
  },  
  {  
    "id": 3,  
    "descricao": "Angus"  
  }  
]
```

FONTE: O AUTOR (2016)

A escolha para utilização de JSON, ao invés de XML, se fez ao fato do JSON ser mais leve, possuir maior facilidade para manipular os objetos, tornando o código mais limpo e, desta forma, deixando a possibilidade de se dedicar apenas a estrutura dos dados.

2.5 MYSQL

MySQL é um sistema gerenciador de banco de dados objeto-relacional de código aberto mantido pela Oracle.(MYSQL, 2016). Ele utiliza SQL (Structured Query Language), a linguagem mais popular de manipulação de bancos de dados.

O MySQL foi escolhido para ser o Sistema de Gerenciamento de Banco de Dados (SGBD), por ser confiável, robusto, gratuito, estável, portátil para diferentes plataformas e oferecer um alto nível de segurança, e também por ser de fácil utilização em conjunto com o Java.

3 METODOLOGIA

A documentação é essencial para o desenvolvimento do projeto, pois contém as análises realizadas sobre o escopo do problema e a metodologia que deverá ser adotada para que o sistema atenda às respectivas necessidades. Neste capítulo são apresentadas as metodologias utilizadas no desenvolvimento do projeto, ferramentas e recursos, incluindo a documentação gerada em toda a fase do projeto, desde o levantamento de requisitos até o acompanhamento ao longo do projeto.

3.1 ENGENHARIA DE SOFTWARE

A UML (Linguagem de Modelagem Unificada) foi utilizada para realizar a modelagem, comportamentos e estruturas lógicas a partir de uma linguagem visual. A escolha foi feita por ela ser baseada no paradigma da Orientação a Objeto utilizado na programação do sistema.

Três de seus diagramas foram utilizados no desenvolvimento deste projeto que são:

- Diagrama de casos de uso;
- Diagrama de classes;
- Diagrama de sequência;

A documentação completa com os diagramas encontra-se nos apêndices, ao final deste documento.

3.1.1 Diagrama de Caso de Uso

O Diagrama de Caso de Uso (APÊNDICE A) auxiliou a identificação do usuário, chamado ator, as funcionalidades e os cenários, bem como o relacionamento entre eles, possibilitando visão geral das funcionalidades dentro do sistema. O sistema possui somente um ator principal – Usuário – onde o mesmo realiza todas as funcionalidades do sistema.

3.1.2 Diagrama de Sequência

O Diagrama de Sequência (APÊNDICE F) representa o fluxo de dados e eventos dentro do sistema, iniciando no ator até a inserção ou busca das informações no banco de dados. Para cada Caso de Uso foi elaborado um Diagrama de Sequência.

3.1.3 Especificação de Caso de Uso

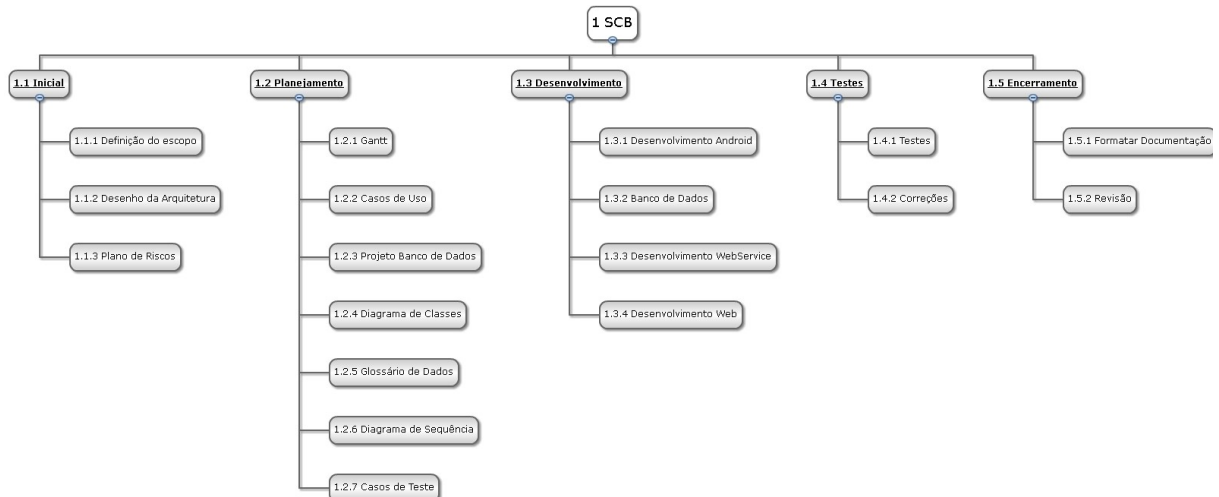
A Especificação de Casos de Uso (APÊNDICE E) fornece detalhes dos Casos de Uso através da descrição das telas, definindo o fluxo principal de ações entre o usuário e o sistema, bem como fluxos alternativos, exceções e regras de negócio.

3.2 PLANO DE ATIVIDADES

O plano de atividades é parte do levantamento do projeto e é fundamental para o planejamento e gerenciamento do mesmo, pois permite definir e distribuir tarefas, monitorar o projeto pelo cronograma e analisar os riscos que podem impactar o sistema e o projeto.

Apesar de estar ordenado as fases do plano não foram executados sequencialmente, e muitos dos artefatos foram ajustados durante todo o processo de desenvolvimento.

Figura 6 - WBS do planejamento das atividades



FONTE: O AUTOR(2016)

3.3 CRONOGRAMA

Figura 7 - Cronograma das Atividades

Nome	Duração	Início	Fim	Antecessores	Nomes dos Recursos
<input type="checkbox"/> SCB	13 dias?	01/06/16 08:00	17/06/16 17:00		
<input type="checkbox"/> Inicial	13 dias?	01/06/16 08:00	17/06/16 17:00		
Definição de Escopo	3 dias?	01/06/16 08:00	03/06/16 17:00		Eduardo Rauta
Desenho da Arquitetura	9 dias?	03/06/16 08:00	15/06/16 17:00		Eduardo Rauta
Plano de Riscos	2 dias?	16/06/16 08:00	17/06/16 17:00		Eduardo Rauta
<input type="checkbox"/> Planejamento	16 dias?	17/06/16 08:00	08/07/16 17:00		
Gantt	1 dia?	17/06/16 08:00	17/06/16 17:00		Eduardo Rauta
Casos de Uso	5 dias?	18/06/16 08:00	24/06/16 17:00		Eduardo Rauta
Projeto de Banco de Dados	2 dias?	27/06/16 08:00	28/06/16 17:00		Eduardo Rauta
Diagrama de Classes	4 dias?	28/06/16 08:00	01/07/16 17:00		Eduardo Rauta
Glossário de Dados	1 dia?	28/06/16 08:00	28/06/16 17:00		Eduardo Rauta
Diagrama de Sequência	7 dias?	29/06/16 08:00	07/07/16 17:00		Eduardo Rauta
Casos de Teste	1 dia?	08/07/16 08:00	08/07/16 17:00		Eduardo Rauta
<input type="checkbox"/> Desenvolvimento	58 dias?	11/07/16 08:00	28/09/16 17:00		
Desenvolvimento Android	45 dias?	11/07/16 08:00	09/09/16 17:00		Eduardo Rauta
Banco de Dados	3 dias?	15/08/16 08:00	17/08/16 17:00		Eduardo Rauta
Desenvolvimento WebServ	30 dias?	18/08/16 08:00	28/09/16 17:00		Eduardo Rauta
Desenvolvimento Web	7 dias?	01/09/16 08:00	09/09/16 17:00		Eduardo Rauta
<input type="checkbox"/> Testes	73 dias?	21/08/16 08:00	30/11/16 17:00		
Testes	45 dias?	21/08/16 08:00	21/10/16 17:00		Eduardo Rauta
Correções	65 dias?	01/09/16 08:00	30/11/16 17:00		Eduardo Rauta
<input type="checkbox"/> Encerramento	22 dias?	01/11/16 07:00	30/11/16 17:00		
Formatação Documento	15 dias?	01/11/16 07:00	21/11/16 17:00		Eduardo Rauta
Revisão	8 dias?	21/11/16 07:00	30/11/16 17:00		Eduardo Rauta

FONTE: O AUTOR (2016)

3.4 PLANO DE RISCOS

O plano de risco tem uma visão geral e com os principais riscos para o projeto, e se tratamento de poucos recursos, e apesar de alguns atrasos, houve uma facilidade no gerenciamento desses riscos mitigando-os ao máximo possível. A tabela a seguir (TABELA 1) indica os riscos identificados no projeto e a análise qualitativa dos mesmos.

Tabela 1 - Riscos e Plano de Ação

Item	Condição	Consequência	Ação	Probabilidade	Impacto
1	Problemas na documentação	Retrabalho no desenvolvimento	Revisar a documentação	médio	moderado
2	Não cumprir os prazos estabelecidos	Atraso na entrega do projeto	Compensar o tempo perdido	moderado	alto
3	Problemas técnicos com linguagens de	Atraso no cronograma	Dedicação e foco. Busca por cursos e	moderado	moderado

Item	Condição	Consequência	Ação	Probabilidade	Impacto
	desenvolvimen to.		demais canais de en-sino.		
4	Distrações externas	Atraso no cronograma	Buscar por locais adequados para o desenvolvimen to	alto	alto

4 APRESENTAÇÃO DO SOFTWARE

O software foi desenvolvido sobre a plataforma Android, utilizando a versão 4.4 *KitKat*, desta forma, não atendendo as plataformas anteriores. As rotinas foram desenvolvidas seguindo as normas de *design* propostas pelas Google, a fim de melhorar a interação e a experiência do usuário com o sistema, o que permite a operação do sistema por pessoas não especializadas em informática.

O sistema tem como objetivo auxiliar os proprietários de bovinos no gerenciamento de bovinos. Com este aplicativo, o proprietário dos animais possuirá uma visão gerencial de sua propriedade de uma forma simples, evitando perda de informações e facilitando a consulta dos dados da propriedade.

4.1 FUNCIONAMENTO DO SOFTWARE

O sistema é dividido em 3 partes, o aplicativo para sistema Android, o sistema WEB e o *webservice*, responsável pela sincronização dos dados do aplicativo com a web.

4.1.1 Aplicativo para Android

O aplicativo é responsável por todo o cadastramento de informações referentes ao sistema, entre esses cadastros estão: Entrada e Saída de Animais; Visitas Veterinárias; Registro Reprodutivo e Registro de Sanidade do animal.

Para realizar o Login ou o cadastro do usuário, é necessário que o aplicativo esteja conectado com a internet. Após o login, que é realizado somente uma vez, como os dados são gravados na memória do celular, é possível executar o aplicativo de forma *offline*, ou seja, não é necessário a conexão com a internet para executar o aplicativo.

O aplicativo possui a função de sincronização, que envia os dados do aplicativo para a WEB, para esta função, é necessário a conexão com a internet. De forma transparente para o usuário, o aplicativo enviará os dados para um servidor Web. Desta forma, os dados salvos não são perdidos juntos com o celular, sendo baixados, sempre que o login é realizado.

4.1.2 Sistema Web

O sistema WEB, após a realização do login, é responsável por gerar relatórios e mostrar os dados de forma mais dinâmica, não sendo possível fazer o cadastramento de dados.

4.1.3 Webservice

O Webservice é responsável por toda sincronização entre os dados do aplicativo com a WEB. Nele é feita toda validação entre os dados para que não haja conflito com os dados que estão no aplicativo, com os dados que estão sendo passados para a web. O Webservice também é responsável pela validação do Login e/ou criação do usuário no aplicativo.

4.2 VISÃO GERAL DO SISTEMA

4.2.1 Tela Login

A Tela Login (APÊNDICE E: Data View 8 - Login) é apresentada assim que o aplicativo é iniciado e apresenta os seguintes componentes:

- Formulário Login: composto por campos a serem preenchidos pelo usuário.
- Botão “Login” para realizar o Login e “Criar Novo Usuário” para carregar a tela de novo cadastro.

4.2.2 Tela Cadastro

A Tela de Cadastro (APÊNDICE E: Data View 11 - Cadastrar Novo Usuário) permite ao usuário criar um cadastro para uso do aplicativo:

- Formulário: composto por campos a serem preenchidos pelo usuário.
- Botão “Cadastrar” para realizar o cadastro e “Voltar” para voltar a tela anterior.

4.2.3 Tela Inicial

A Tela Inicial (APÊNDICE E: Data View 1 - Tela inicial) é visualizada assim que o login é realizado e apresenta os seguintes componentes:

- Menu: contém as opções disponíveis para navegação no aplicativo.
- Campo de informações: Campo com as informações de cadastro, próxima data de visita e o total de animais cadastrados no sistema.
- Botão Sincronizar responsável pelo envio dos dados para a web.

4.2.4 Gerenciar Animais

A tela Listar Usuários (APÊNDICE E: Data View 3 - Gerenciar Animais) apresenta a lista de animais. A tela é composta de:

- Lista: lista que contém os animais cadastrados no sistema. Ao clicar sobre o animal, o sistema o conduz a tela de perfil do animal. Se o clique for mais longo, é aberto a opção de remover o animal.
- Busca: campo de preenchimento do usuário para filtrar a lista conforme a busca desejada.
- Botão “Novo” é redirecionado para a tela de inserção de um novo animal.

4.2.5 Perfil Animal

Essa tela permite consiste em mostrar informações e opções do animal selecionado (APÊNDICE E: Data View 5 - Perfil Animal). A tela apresenta os seguintes componentes:

- Informações sobre o animal.
- Botão “Registro Reprodutivo” que direciona para a tela de Gerenciar Inseminações.
- Botão “Registro Sanidade” que direciona para a tela de Gerenciar Vacinas.
- Botão “Editar Dados” que direciona para a tela de editar os dados do animal.

4.2.6 Editar Animal

Essa tela permite ao administrador cadastrar/editar animais (APÊNDICE E: Data View 4 - Entrada de Animal). A tela apresenta os seguintes componentes:

- Formulário do Animal: composto por campos a serem preenchidos pelo usuário.

- Botão “Salvar” para realizar o cadastro e “Voltar” para voltar a tela anterior.

4.2.7 Gerenciar Visitas

A tela Gerenciar Visitas(APÊNDICE E: Data View 6 - Gerenciar Visitas) apresenta a lista de visitas veterinárias. A tela é composta de:

- Lista: lista que contém as visitas veterinárias cadastrados no sistema. Ao clicar sobre a visita, o sistema o conduz a tela de informações da visita. Se o clique for mais longo, é aberto a opção de remover a visita.

- Busca: campo de preenchimento do usuário para filtrar a lista conforme a busca desejada.

- Botão “Novo” é redirecionado para a tela de inserção de uma nova visita.

4.2.8 Editar Visita

Essa tela permite ao administrador cadastrar/editar visitas veterinárias(APÊNDICE E: Data View 7 - Cadastrar Visita). A tela apresenta os seguintes componentes:

- Formulário de Visita: composto por campos a serem preenchidos pelo usuário.

- Botão “Salvar” para realizar o cadastro e “Voltar” para voltar a tela anterior.

4.2.9 Gerenciar Vacinas

A tela Gerenciar Vacinas apresenta a lista de vacinas do animal. A tela é composta de:

- Lista: lista que contém as vacinas associadas ao animal. Ao clicar sobre a vacina, o sistema o conduz a tela de edição dos dados da vacina. Se o clique for mais longo, é aberto a opção de remover a vacina.

- Busca: campo de preenchimento do usuário para filtrar a lista conforme a busca desejada.

- Botão “Novo” é redirecionado para a tela de inserção de uma nova visita.

4.2.10 Editar Vacina

Essa tela permite ao administrador cadastrar/editar vacinas do animal(APÊNDICE E: Data View 12 - Cadastrar Registro Sanidade). A tela apresenta os seguintes componentes:

- Formulário de Vacina: composto por campos a serem preenchidos pelo usuário.
- Botão “Salvar” para realizar o cadastro e “Voltar” para voltar a tela anterior.

4.2.11 Gerenciar Inseminações

A tela Gerenciar Inseminações(APÊNDICE E: Data View 9 - Listar Registros Reprodutivos) apresenta a lista de registros reprodutivo do animal. A tela é composta de:

- Lista: lista que contém os registros reprodutivos do animal. Ao clicar sobre o registro, o sistema o conduz a tela de edição dos dados da inseminação. Se o clique for mais longo, é aberto a opção de remover o registro.
- Busca: campo de preenchimento do usuário para filtrar a lista conforme a busca desejada.
- Botão “Novo” é redirecionado para a tela de inserção de uma nova visita.

4.2.12 Adicionar Inseminação

Essa tela permite ao administrador cadastrar inseminações do animal(APÊNDICE E: Data View 13 - Cadastrar Registro Reprodutivo). A tela apresenta os seguintes componentes:

- Formulário de Inseminação: composto por campos a serem preenchidos pelo usuário.
- Botão “Salvar” para realizar o cadastro e “Cancelar” para voltar a tela anterior.

4.2.13 Editar Inseminação

Essa tela permite ao administrador editar inseminações do animal(APÊNDICE E: Data View 10 - Alterar Registro Reprodutivo). A tela apresenta os seguintes componentes:

- Formulário de Inseminação: composto por campos a serem preenchidos pelo usuário.

- Botão “Salvar” para realizar a edição e “Voltar” para voltar a tela anterior.

5 CONSIDERAÇÕES FINAIS

A informática é uma ferramenta vital para as mais diversas atividades empresariais, incluindo o setor agropecuário. Para que a tecnologia possa auxiliar os produtores rurais, é necessário que se disponham softwares destinados a atender estas necessidades. Esse aplicativo veio para tornar o gerenciamento de bovinos, principalmente na parte reprodutiva, mais prático e rápido, aumentando a confiabilidade das informações, evitando perda de informações e facilitando a consulta dos dados da propriedade a qualquer momento e lugar.

Como o sistema tem por objetivo a melhora contínua, mais relatórios podem ser adicionados ao sistema, facilitando a tomada de decisão do proprietário. Outro grande passo para a ampliação e melhora do sistema é o desenvolvimento de alguns módulos que podem ser adicionados, como controle leiteiro individual para cada animal e controle de peso para gado de corte.

REFERÊNCIAS

BUSCHMANN, Frank; MEUNIER, Regine; ROHNERT, Hans; SOMMERLAD, Peter; STAL, Michael. **Pattern-Oriented Software Architecture: A System of Patterns**, 1st ed., John Wiley & Sons. 1996

CROCKFORD, Douglas. **The application/json Media Type for JavaScript Object Notation (JSON)**. 2006. Disponível em: <<http://www.ietf.org/rfc/rfc4627.txt>>. Acesso em: 04 nov. 2016.

ELKSTEIN, M. **What is rest**. 2008. Disponível em: <<http://rest.elkstein.org/2008/02/what-is-rest.html>>. Acesso em: 04 nov. 2016.

GARTNER, Inc. **Gartner Says Worldwide Smartphone Sales Grew 3.9 Percent in First Quarter of 2016**. 2016. Disponível em: <<http://www.gartner.com/newsroom/id/3323017>>. Acesso em: 31 out. 2016 .

GOOGLE, Inc. **The Android Developer's Guide**. 2016. Disponível em: <<http://developer.android.com/guide/index.html>>. Acesso em: 31 out. 2016 .

JOHNSON, Rod. **The Spring Framework – Reference Documentation**. 2016. Disponível em: <<http://docs.spring.io/spring/docs/4.3.x/spring-framework-reference/pdf/spring-framework-reference.pdf>>. Acessado em: 04 nov. 2016.

LOPES, Marcos Aurélio; VIEIRA, P. de F. **Criação de bezerros leiteiros**. Jaboticabal: Funep, 1998.

LOPES, Marcos Aurélio; SANTOS, Glauber dos; ALBUQUERQUE, Flamarion Tenório. **Maneira Prática de Realizar Controle Reprodutivo em Gado Leiteiro em Propriedades com Economia Familiar**. Boletim Técnico-n. 74 – p.1-20, 2005. Disponível em: <<http://livraria.editora.ufla.br/upload/boletim/tecnico/boletim-tecnico-74.pdf>>. Acessado em: 07 out. 2016.

MYSQL. **About**. Disponível em <<http://www.mysql.com/>>. Acesso em: 04 nov. 2016.

WINCH Robert, MULARIEN Peter. **Spring Security 3.1**. 2016. Disponível em: <<http://projects.spring.io/spring-security/>> em Acesso em: 01 nov. 2016.

W3C , Working Group. **Web Services Architecture**. Note 11, 2004. Disponível em: <<https://www.w3.org/TR/ws-arch/>>. Acesso em: 04 nov. 2016.

GLOSSÁRIO

A

ANDROID – Sistema operacional do Google para dispositivos móveis baseado no Linux.

API – É um conjunto de rotinas e padrões de programação para acesso a um aplicativo de software ou plataforma baseado na Web. A sigla refere-se ao termo em inglês Application Programming Interface que significa em português Interface de Programação de Aplicativos.

E

EJB – Enterprise JavaBeans é um componente da plataforma Java Platform, Enterprise Edition. Seu principal objetivo consiste em fornecer um desenvolvimento rápido e simplificado de aplicações Java.

J

JAVA – Linguagem de programação e plataforma computacional lançada pela primeira vez pela Sun Microsystems em 1995.

JSON - JavaScript Object Notation, é um modelo para armazenamento e transmissão de informações no formato texto.

R

REST – A Representational State Transfer (REST), em português Transferência de Estado Representacional, é uma abstração da arquitetura da World Wide Web (Web), um estilo arquitetural que consiste de um conjunto de coordenado de restrições arquiteturais aplicadas a componentes, conectores e elementos de dados.

S

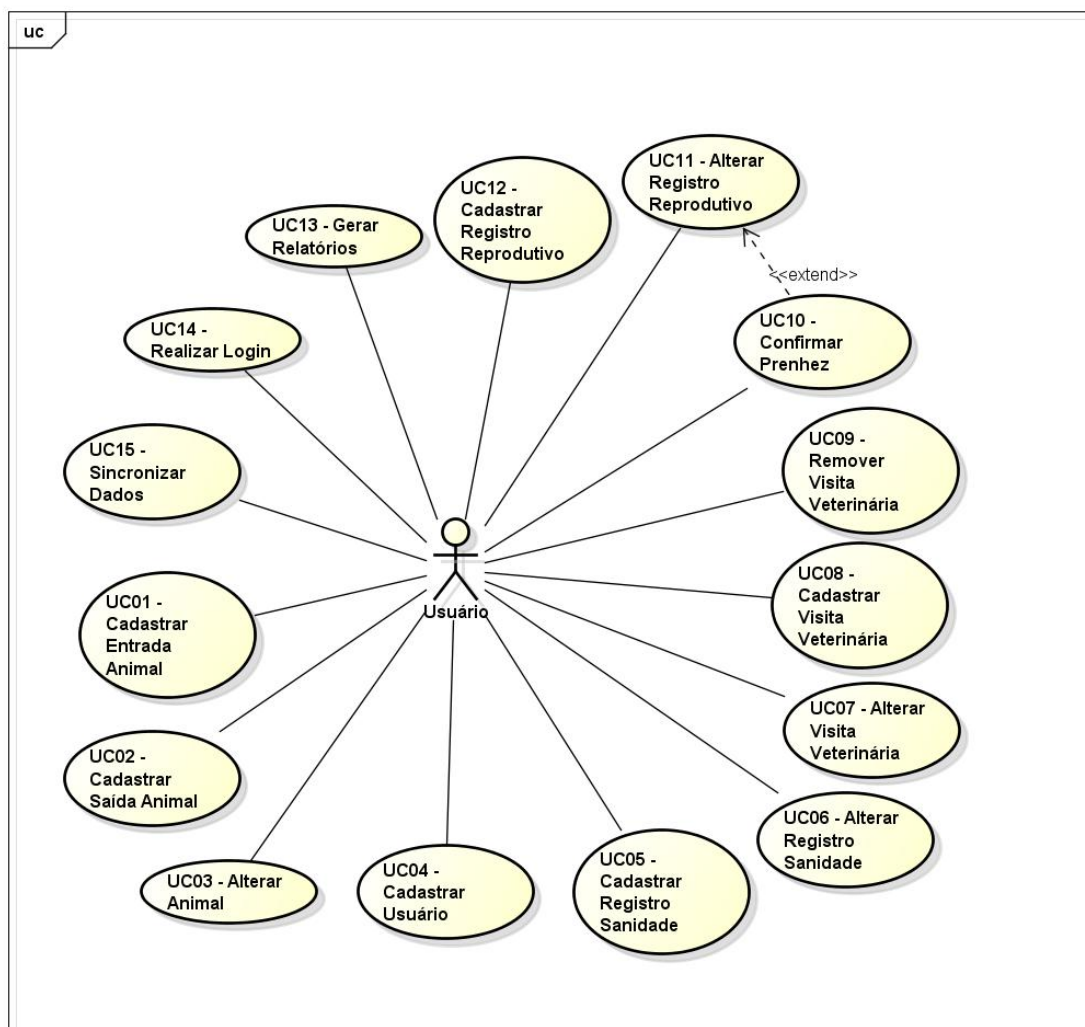
SDK – Software Development Kit ou Kit de Desenvolvimento de Software, é tipicamente um conjunto de ferramentas de desenvolvimento de software que permite a criação de aplicativos para um certo pacote de software.

X

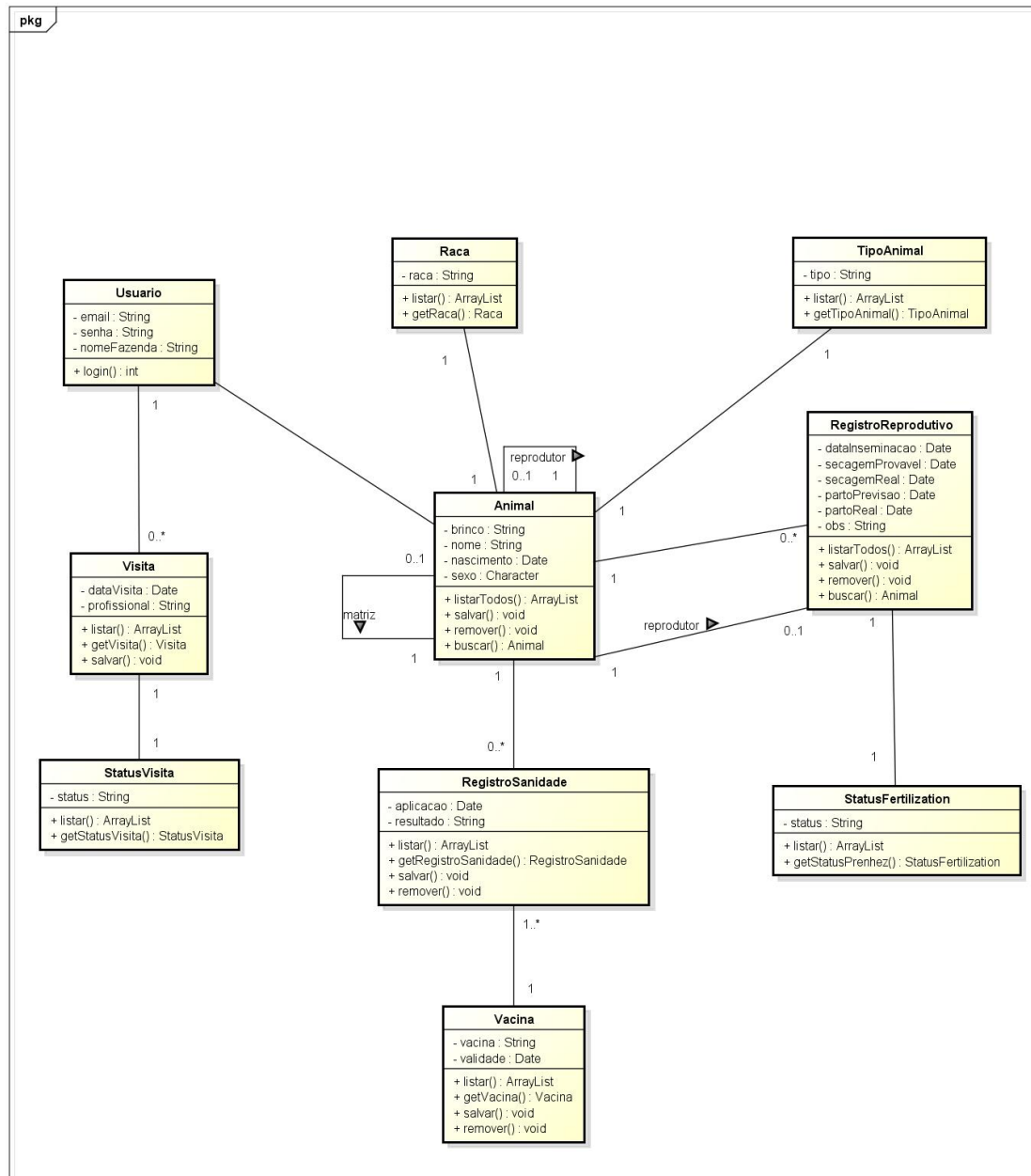
XML – eXtensible Markup Language, é uma linguagem de marcação para a criação de documentos com dados organizados hierarquicamente.

APÊNDICES

APÊNDICE A - DIAGRAMA DE CASOS DE USO



APÊNDICE B - DIAGRAMA DE CLASSES



APÊNDICE D - REGRAS DE NEGÓCIO

LOGAR NO SISTEMA

- **RN01: Acesso ao Sistema**
 - Para utilizar os módulos do sistema, o usuário deverá efetuar o login no aplicativo do sistema.

MANTER ANIMAL

- **RN02: Cadastrar Entrada Animal**
 - O usuário deverá preencher os campos obrigatórios para cadastrar entrada de animal.
- **RN03: Número Brinco Único**
 - Cada animal possuirá uma numeração de brinco única, não podendo ser repetida.
- **RN04: Alterar Animal**
 - O cadastro poderá ser atualizado quando houver alterações nos dados do animal. Todos os campos estarão habilitados para alteração, exceto o campo Brinco do Animal.
- **RN05: Cadastrar Saída Animal**
 - O animal não poderá ser excluído. Se houver motivos de desativação, o mesmo deverá receber o status "Inativo".

MANTER FAZENDA

- **RN06: Cadastrar Fazenda**
 - O usuário deverá preencher os campos obrigatórios para cadastrar fazenda.
- **RN07: Alterar Fazenda**
 - O cadastro poderá ser atualizado quando houver alterações nos dados da fazenda. Todos os campos estarão habilitados para alteração.

MANTER RELATÓRIOS

- **RN08: Acesso ao Sistema**
 - Para utilizar os módulos de geração de relatórios, o usuário deverá efetuar o login no sistema Web.

MANTER VISITAS VETERINÁRIO

- **RN09: Cadastrar Visita**
 - O usuário deverá preencher os campos obrigatórios para cadastrar a visita.

- **RN10: Alterar Visita**
 - O cadastro poderá ser atualizado quando houver alterações nos dados da visita, e a visita ainda não tenha sido realizada. Somente o campo de descrição da visita poderá ser alterado.
- **RN11: Remover Visita**
 - O cadastro poderá remover somente se a visita ainda não tenha sido realizada.
- **RN12: Criar evento na agenda**
 - Durante o cadastro, o usuário terá a opção de selecionar se deseja cadastrar a visita veterinária na agenda do celular, para receber as notificações no dia do evento.

MANTER REGISTRO REPRODUTIVO

- **RN13: Cadastrar Registro Reprodutivo**
 - O usuário deverá preencher os campos obrigatórios para cadastrar o Registro Reprodutivo.
- **RN14: Alterar Registro Reprodutivo**
 - O cadastro poderá ser atualizado quando houver alterações nos dados do Registro Reprodutivo. Todos os campos estarão habilitados para alteração.
- **RN15: Remover Registro Reprodutivo**
 - Não será possível realizar a exclusão de Registros Reprodutivos cujo status for “Prenha” ou “Repetir Toque”.
- **RN16: Confirmar Prenhez**
 - Deverá ser confirmada a prenhez do animal quando houver inseminação. Após isso, a previsão de secagem deverá fornecer a data automática 60 dias antes da data de parto do animal, ou seja, 230 dias após a inseminação. Sendo assim a previsão para a data de parto receberá o valor de 290 dias após a inseminação.

MANTER REGISTRO DE SANIDADE

- **RN17: Cadastrar Registro de Sanidade**
 - O usuário deverá preencher os campos obrigatórios para cadastrar o Registro de Sanidade.
- **RN18: Alterar Registro de Sanidade**
 - O cadastro poderá ser atualizado quando houver alterações nos dados do Registro de Sanidade. Todos os campos estarão habilitados para alteração.

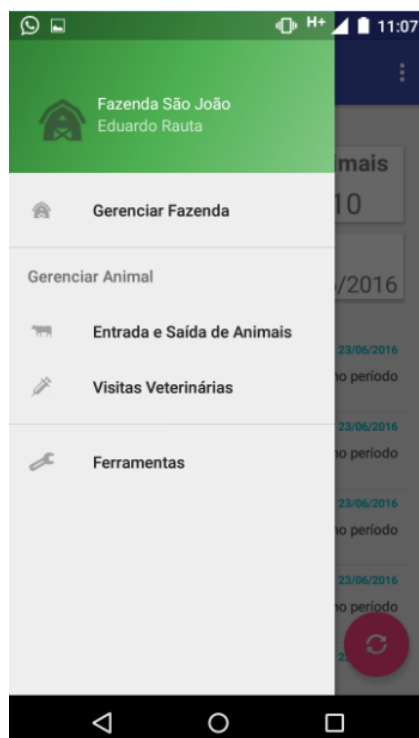
SINCRONIZAR DADOS

- **RN19: Sincronizar Dados**
 - O usuário deverá estar com o *smartphone* conectado a internet para realizar a sincronização dos dados

APÊNDICE E - INTERFACES



Data View 1 - Tela inicial



Data View 2 - Menu Lateral



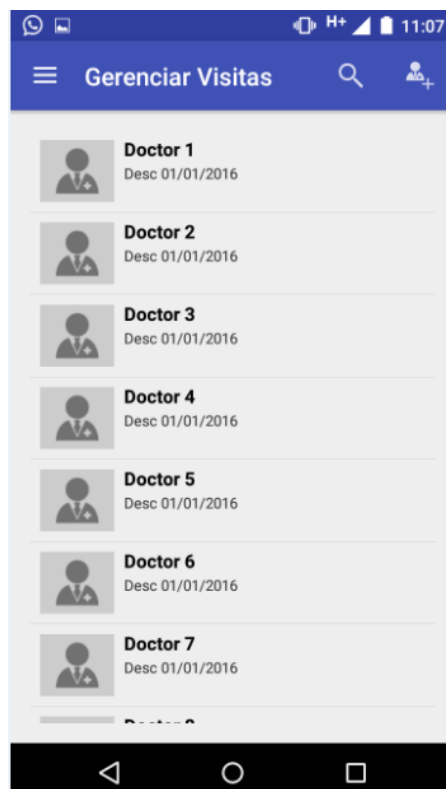
Data View 3 - Gerenciar Animais



Data View 4 - Entrada de Animal



Data View 5 - Perfil Animal



Data View 6 - Gerenciar Visitas

Veterinário, lugar, local...

Inicio HH:mm

Fim HH:mm

Criar evento na Agenda

Data View 7 - Cadastrar Visita

Image Coming Soon

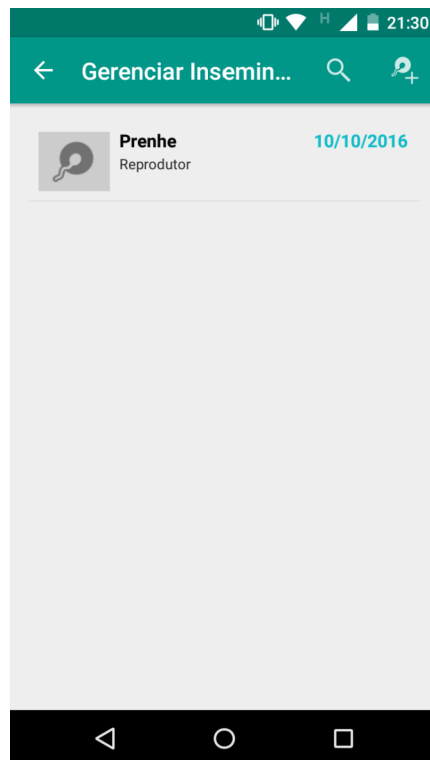
Usuario

Senha

Entrar

[Criar uma conta](#)

Data View 8 - Login




Data View 9 - Listar Registros Reprodutivos



Data View 10 - Alterar Registro Reprodutivo

SCB

6:00

 Cowntrol

Nome

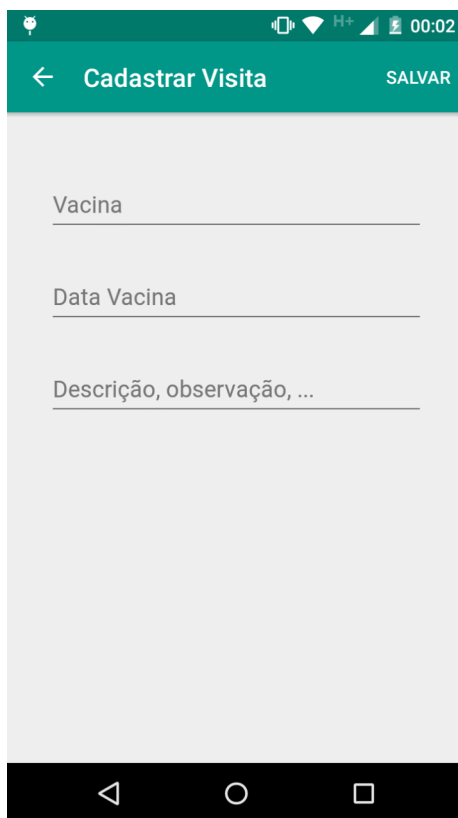
Email

Senha

Nome Fazenda

CRIAR CONTA

Data View 11 - Cadastrar Novo Usuário



The screenshot shows a mobile application interface for recording a visit. At the top, there is a green header bar with a back arrow on the left, the text "Cadastrar Visita" in the center, and "SALVAR" on the right. Below the header, the form consists of three input fields: "Vacina", "Data Vacina", and "Descrição, observação, ...". Each field has a horizontal line indicating the input area. At the bottom of the screen, there is a black navigation bar with three white icons: a triangle pointing left, a circle, and a square.

Data View 12 - Cadastrar Registro Sanidade



The screenshot shows a mobile application interface for managing insemination records. At the top, there is a green header bar with a back arrow on the left, the text "Gerenciar Insemin..." in the center, and search and add icons on the right. Below the header, there is a card for "Prenhe Reprodutor" with a date of "10/10/2016". A white dialog box is overlaid on the screen with the title "Adicionar Registro Reprodutivo". The dialog box contains a "Data Cobertura" input field, a "Brinco Reprodutor" label, and two buttons at the bottom: "CANCELAR" and "SALVAR". At the bottom of the screen, there is a black navigation bar with three white icons: a triangle pointing left, a circle, and a square.

Data View 13 - Cadastrar Registro Reprodutivo

APÊNDICE F - CASOS DE USO

Especificação de Caso de Uso **UC01 – Cadastrar Entrada Animal**

Descrição

Este caso de uso serve para efetuar o cadastro de entrada de um animal.

Data View

- **Data View 4** - Entrada de animal.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo os dados do animal.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV4)**
2. O usuário preenche todos os campos da tela.
3. O usuário pressiona o botão Salvar. **(A1)(E1)(E2)**
4. O sistema inclui o animal.
5. O sistema emite a mensagem “Entrada de Animal realizada com sucesso”.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1. Brinco já cadastrado:

1. O sistema consiste o Brinco. **(RN03)**
2. O sistema retorna a mensagem “Brinco já cadastrado”.
3. O Use Case é reiniciado.

E2. Dados obrigatórios:

1. O sistema consiste os dados preenchidos. **(RN02)**
2. O sistema retorna a mensagem “É necessário preencher todos os campos obrigatórios”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso **UC02 – Cadastrar Saída Animal**

Descrição

Este caso de uso serve para efetuar o cadastro da saída de um animal.

Data View

- **Data View 3** - Gerenciar animais.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login
2. Possuir o cadastro do animal para alteração.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo a saída do animal.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV3)**
2. O usuário seleciona o animal desejado. **(A1)**
3. O usuário pressiona o ícone responsável pela ação de saída de animal.
4. O sistema emite a mensagem “Deseja confirma saída do animal XXX?”.
5. O usuário seleciona a opção “Confirmar”. **(A2)**
6. O sistema registra a saída do animal. **(RN05)**
7. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Buscar” pressionado.

1. O usuário usa o filtro para buscar o animal desejado.
2. O usuário seleciona o animal desejado.

3. Retorna ao Fluxo Principal.

A2: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Especificação de Caso de Uso

UC03 – Alterar Animal

Descrição

Este caso de uso serve para efetuar a alteração no cadastro de um animal.

Data View

- **Data View 4** - Entrada de animal.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login
2. Possuir o cadastro do animal para alteração.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo os dados do animal.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV4)**
2. O usuário altera os campos da tela.
3. O usuário pressiona o botão Salvar. **(A1) (E1)**
4. O sistema inclui o animal.
5. O sistema emite a mensagem “Alteração realizada com sucesso”.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1. Dados obrigatórios:

1. O sistema consiste os dados preenchidos. **(RN04)**
2. O sistema retorna a mensagem “É necessário preencher todos os campos obrigatórios”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso

UC04 – Cadastrar Usuário

Descrição

Este caso de uso serve para efetuar o cadastro do usuário no sistema.

Data View

- **Data View 11** – Cadastrar Novo Usuário.

Pré-condições

Não existe pré-condição.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo o usuário.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV11)**
2. O usuário preenche todos os campos da tela.
3. O usuário pressiona o botão Criar Conta. **(A1)(E1)**
4. O sistema inclui o usuário.
5. O sistema emite a mensagem “Usuário cadastrado com sucesso”.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1. Dados Inválidos:

1. O sistema valida os campos.
2. O sistema retorna a mensagem “Dados Inválidos”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso

UC05 – Cadastrar Registro Sanidade

Descrição

Este caso de uso serve para efetuar o cadastro de registro de sanidade de um animal.

Data View

- **Data View 12** – Cadastrar Registro de Sanidade.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo os dados do registro de sanidade animal.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV12)**
2. O usuário preenche todos os campos da tela.
3. O usuário pressiona o botão Salvar. **(A1)**
4. O sistema inclui o registro de sanidade.
5. O sistema emite a mensagem “Registro cadastrado com sucesso”.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Especificação de Caso de Uso

UC07 – Alterar Visita Veterinária

Descrição

Este caso de uso serve para efetuar a alteração nos dados de uma visita veterinária.

Data View

- **Data View 7** - Cadastrar Visita.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo os dados da visita veterinária.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV7)**
2. O usuário preenche os dados da descrição.
3. O usuário seleciona a opção “Salvar”. **(A1) (E1)**
4. O sistema salva as informações.
5. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1: Dados obrigatórios:

1. O sistema consiste os dados preenchidos. **(RN10)**

2. O sistema retorna a mensagem “É necessário preencher todos os campos obrigatórios”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso
UC08 – Cadastrar Visita Veterinária

Descrição

Este caso de uso serve para efetuar o cadastro de uma visita veterinária.

Data View

- **Data View 7** - Cadastrar Visita.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo o cadastro da visita veterinária.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV7)**
2. O usuário preenche os dados.
3. O usuário seleciona a opção “Criar evento na agenda”. **(RN12) (A1)**
4. O usuário seleciona a opção “Salvar”. **(A2) (E1)**
5. O sistema emite a mensagem “Será criado um evento na agenda do seu celular para a visita veterinária”.
6. O usuário seleciona a opção “Confirmar”. **(A3)**
7. O sistema salva as informações e inclui um evento na agenda.
8. O caso de uso é finalizado.

Fluxos Alternativos

A1: Opção “Criar evento na agenda” não selecionada.

1. O usuário não seleciona a opção “Criar evento na agenda”.
2. O usuário seleciona a opção “Salvar”. **(A2)**

3. O sistema salva as informações.
4. O Caso de Uso é encerrado.

A2: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

A3: Botão “Cancelar” pressionado.

1. O Caso de Uso é reiniciado.

Fluxos de Exceção

E1. Dados obrigatórios:

1. O sistema consiste os dados preenchidos. (RN09)
2. O sistema retorna a mensagem “É necessário preencher todos os campos obrigatórios”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso

UC09 – Remover Visita Veterinária

Descrição

Este caso de uso serve para remover o registro de visita veterinária.

Data View

- **Data View 6** – Gerenciar Visitas.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter removido o registro da visita veterinária.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV6)**
2. O usuário seleciona o registro desejado. **(A1)**
3. O usuário pressiona o ícone responsável pela ação de remover a visita veterinária.
4. O sistema emite a mensagem “Deseja confirmar a exclusão da visita?”.
5. O usuário seleciona a opção “Confirmar”. **(A2) (RN11) (E1)**
6. O sistema registra a saída do animal.
7. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Buscar” pressionado.

1. O usuário usa o filtro para buscar a visita desejada.
2. O usuário seleciona a visita desejada.
3. Retorna ao Fluxo Principal.

A2: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1. Dados obrigatórios:

1. O sistema consiste os dados preenchidos. (RN11)
2. O sistema retorna a mensagem “Não é possível excluir, visita já foi realizada”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso
UC11 – Alterar Registro Reprodutivo

Descrição

Este caso de uso serve para efetuar a alteração do registro reprodutivo de um animal.

Data View

- **Data View 10** – Alterar Registro Reprodutivo.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo os dados do registro reprodutivo do animal.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV10)**
2. O usuário preenche todos os campos da tela.
3. O usuário pressiona o botão Salvar. **(A1) (E1) (E2) (E3)**
4. O sistema inclui o registro de sanidade.
5. O sistema emite a mensagem “Registro atualizado com sucesso”.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1. Data diagnóstico maior que data atual:

1. O sistema valida os campos.
2. O sistema retorna a mensagem “Data de diagnóstico maior que data atual”.
3. O Use Case é reiniciado.

E2. Data Secagem Real maior que data atual:

1. O sistema valida os campos.
2. O sistema retorna a mensagem “Data de secagem real maior que data atual”.
3. O Use Case é reiniciado.

E3. Data Parto Real maior que data atual:

1. O sistema valida os campos.
2. O sistema retorna a mensagem “Data de parto real maior que data atual”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso
UC12 – Cadastrar Registro Reprodutivo

Descrição

Este caso de uso serve para efetuar o cadastro de registro reprodutivo de um animal.

Data View

- **Data View 13** – Cadastrar Registro Reprodutivo.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo os dados do registro reprodutivo do animal.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV13)**
2. O usuário preenche todos os campos da tela.
3. O usuário pressiona o botão Salvar. **(A1) (E1)**
4. O sistema inclui o registro reprodutivo.
5. O sistema emite a mensagem “Registro cadastrado com sucesso”.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Voltar” pressionado.

1. Encerra o caso de uso.

Fluxos de Exceção

E1. Data cobertura maior que data atual:

1. O sistema valida os campos.
2. O sistema retorna a mensagem “Data de cobertura maior que data atual”.
3. O Use Case é reiniciado.

Especificação de Caso de Uso

UC14 – Realizar Login

Descrição

Este caso de uso serve para realizar login no sistema.

Data View

- **Data View 8** – Login.

Pré-condições

Este caso de uso pode iniciar somente se:

1. Deve possuir um usuário cadastrado no sistema.

Pós-condições

Após o fim normal deste caso de uso o usuário deve:

1. Ter acesso a todas as funcionalidades do sistema.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV8)**
2. O usuário preenche os campos “usuário” e “senha”.
3. O usuário seleciona a opção “Login”. **(A1) (RN01) (E1)**
4. O sistema consiste o login do usuário.
5. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Criar usuário” pressionado.

1. O usuário pressiona o botão “Criar Usuário”.
2. Encerra o caso de uso.

Fluxos de Exceção

E1. Login Inválido:

1. O sistema consiste os dados preenchidos. **(RN01)**
2. O sistema retorna a mensagem "Usuário e/ou senha inválidos".
3. O Use Case é reiniciado.

Especificação de Caso de Uso

UC15 – Sincronizar Dados

Descrição

Este caso de uso serve para sincronizar os dados do aplicativo com os dados da web.

Data View

- **Data View 1** – Tela Inicial.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC14 – Realizar Login.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter sincronizado os dados com a web.

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela. **(DV1)**
2. O usuário pressiona o ícone responsável pela ação de sincronizar os dados.
3. O sistema emite a mensagem “Deseja sincronizar os dados?”.
4. O usuário seleciona a opção “Confirmar”. **(A1) (RN19) (E1)**
5. O sistema sincroniza os dados com a web.
6. O caso de uso é finalizado.

Fluxos Alternativos

A1: Botão “Cancelar” pressionado.

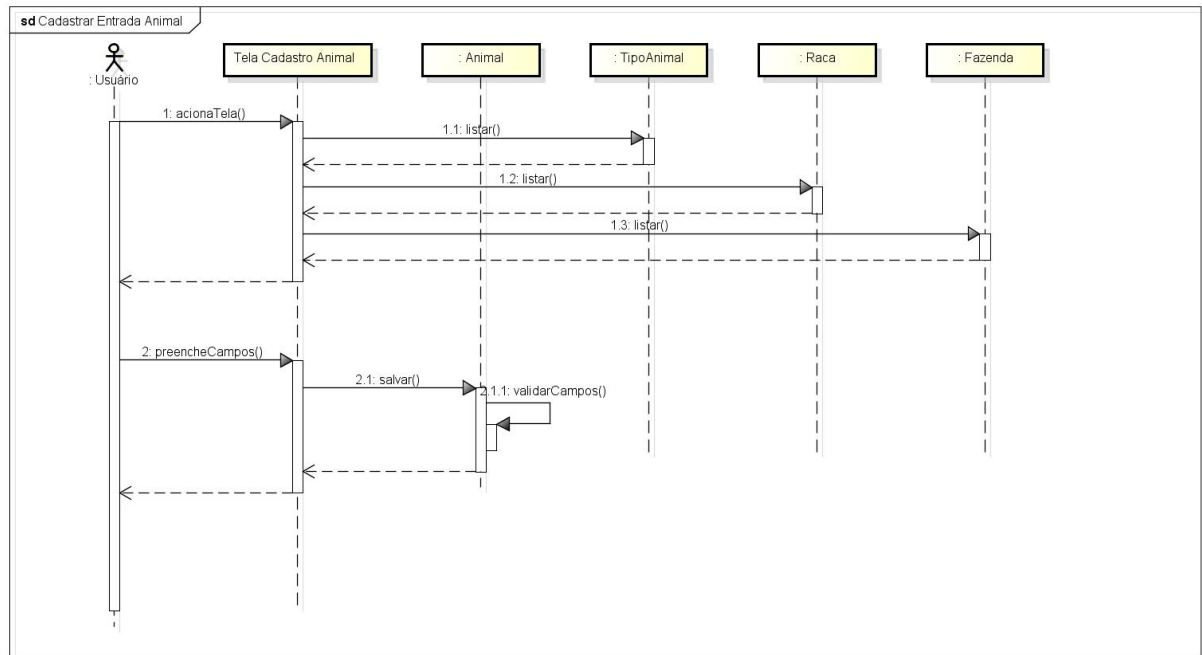
1. O usuário pressiona o botão “Cancelar”.
2. Encerra o caso de uso.

Fluxos de Exceção

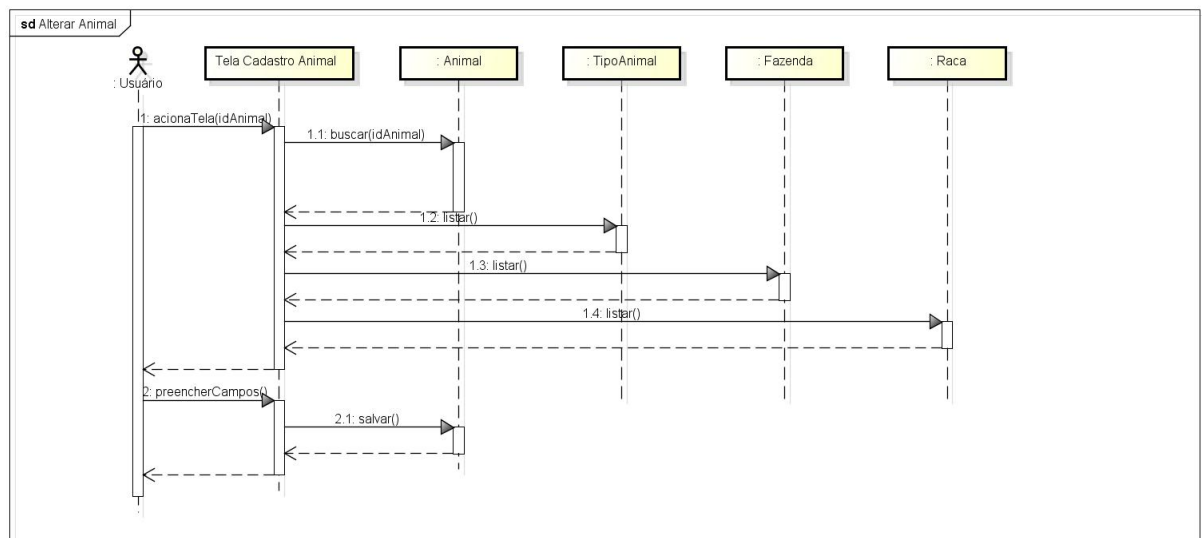
E1. Não conectado a internet:

1. O sistema verifica se possui conexão com a internet. **(RN19)**
2. O sistema retorna a mensagem “É preciso ter conexão com a internet para sincronizar os dados”.
3. O Use Case é reiniciado.

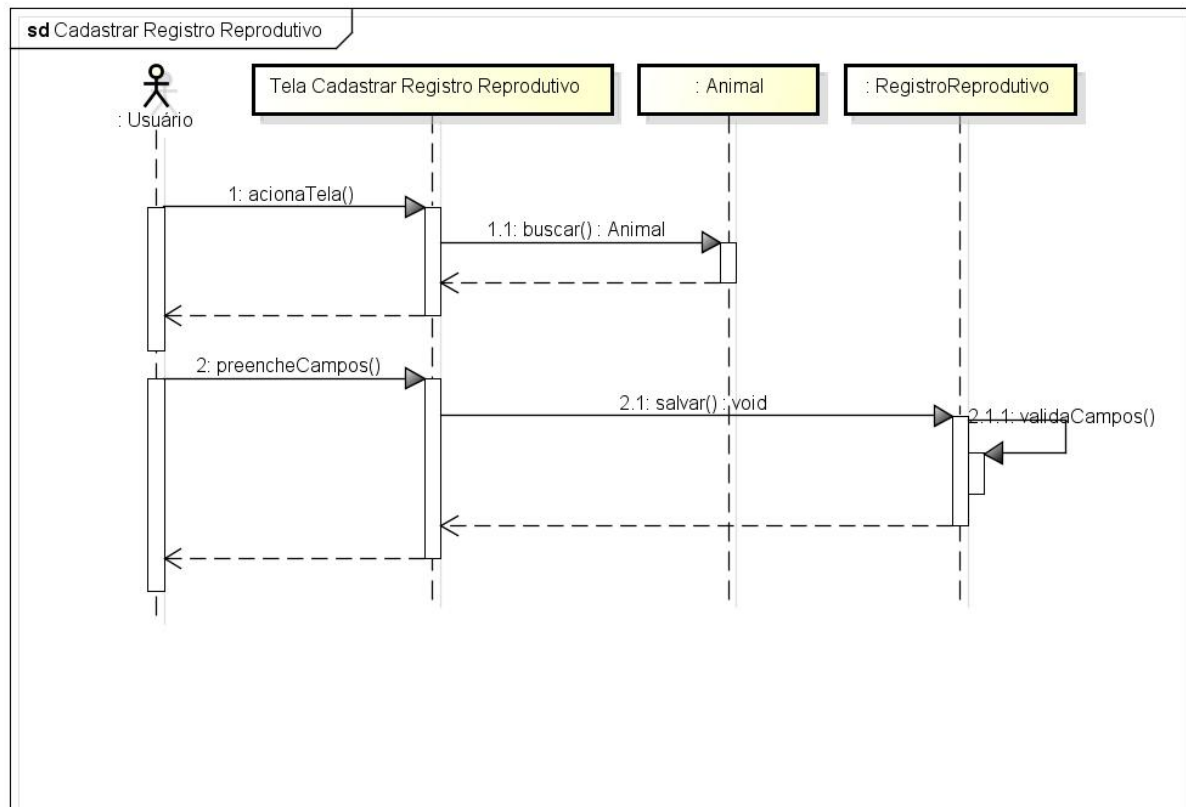
APÊNDICE G - DIAGRAMAS DE SEQUÊNCIA



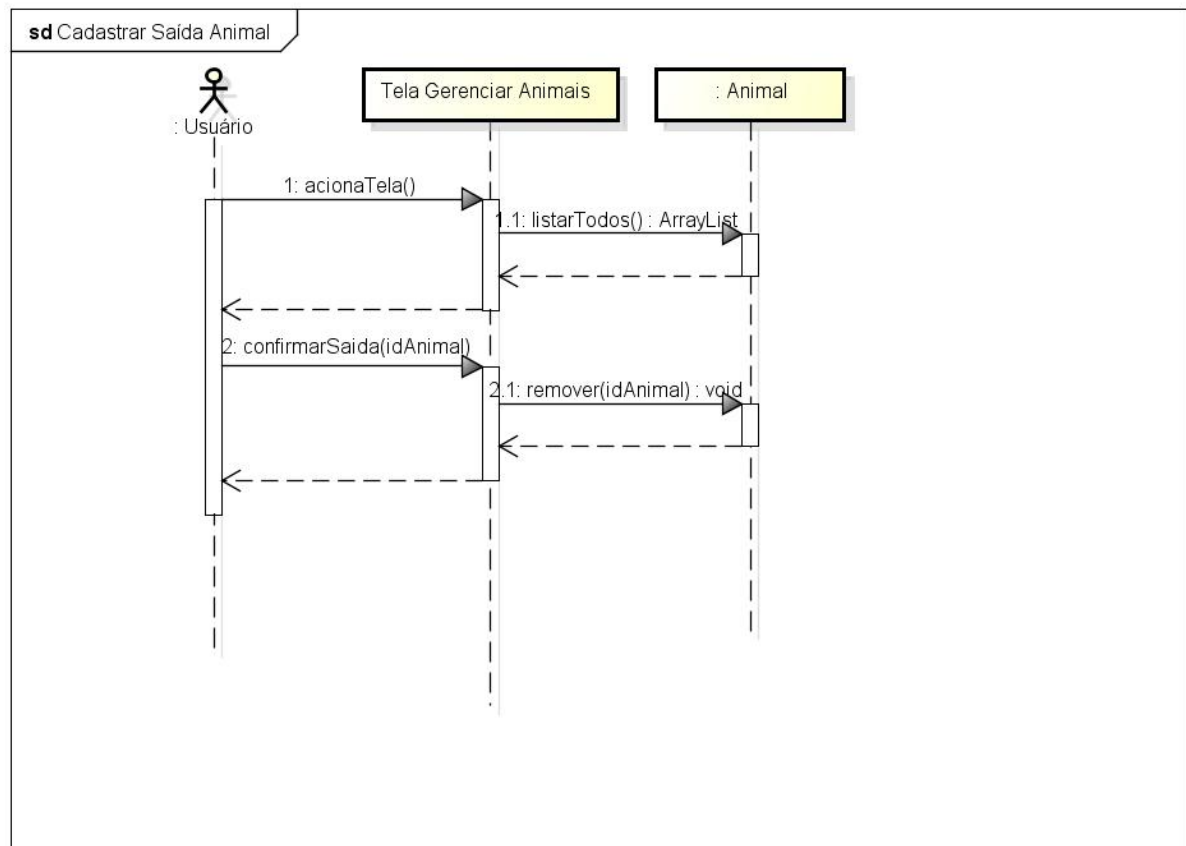
powered by Astah



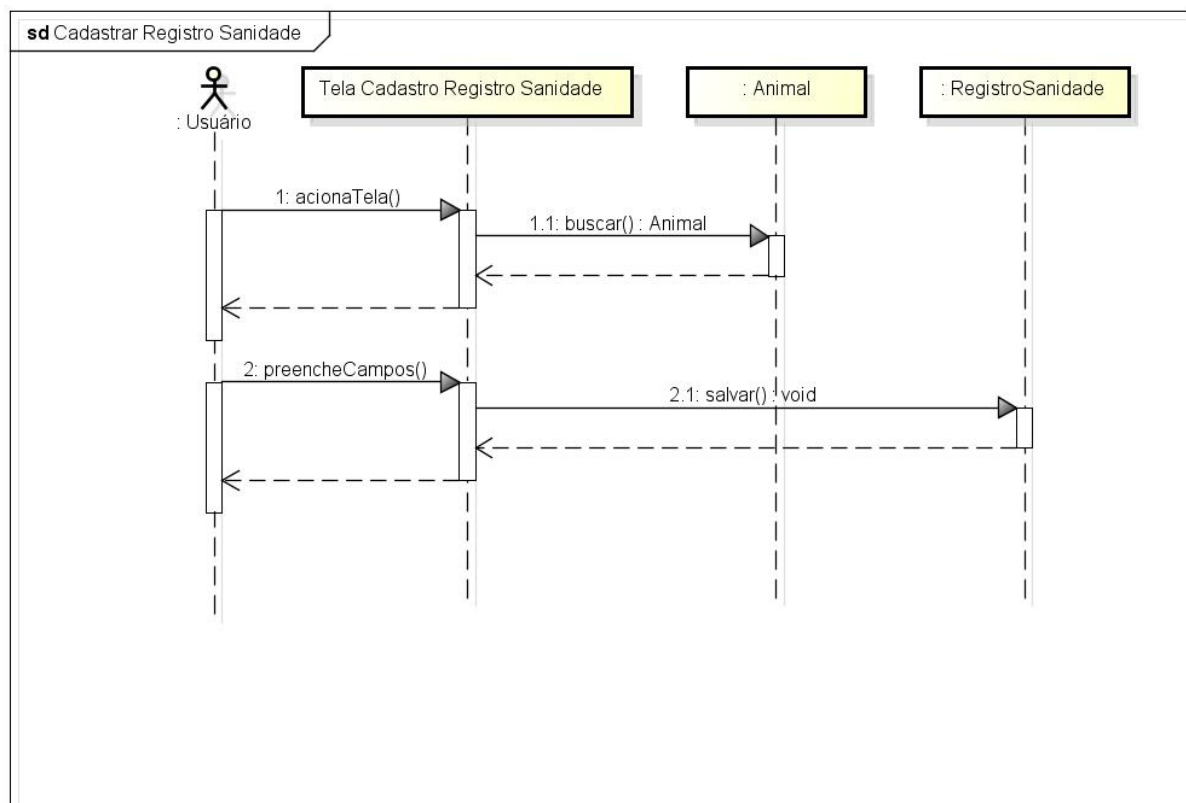
powered by Astah



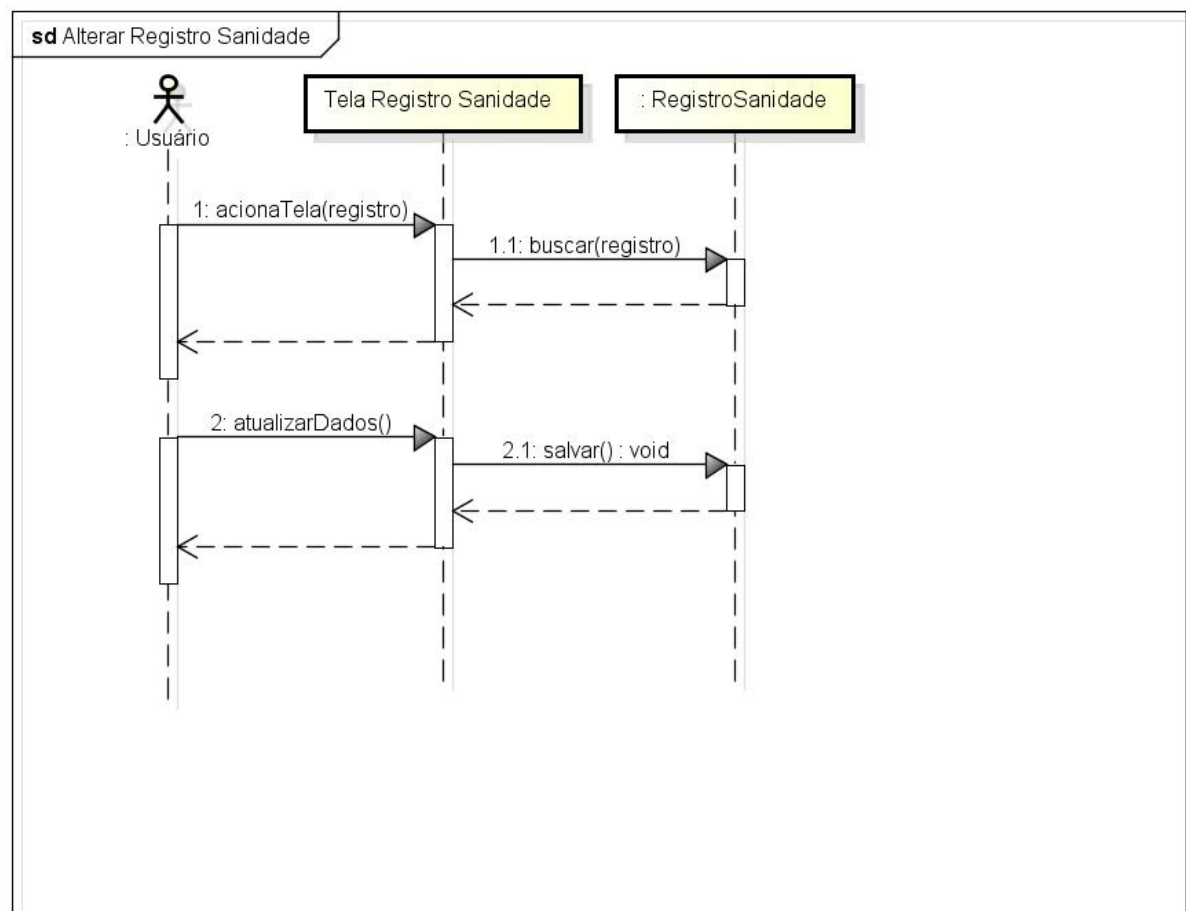
powered by Astah



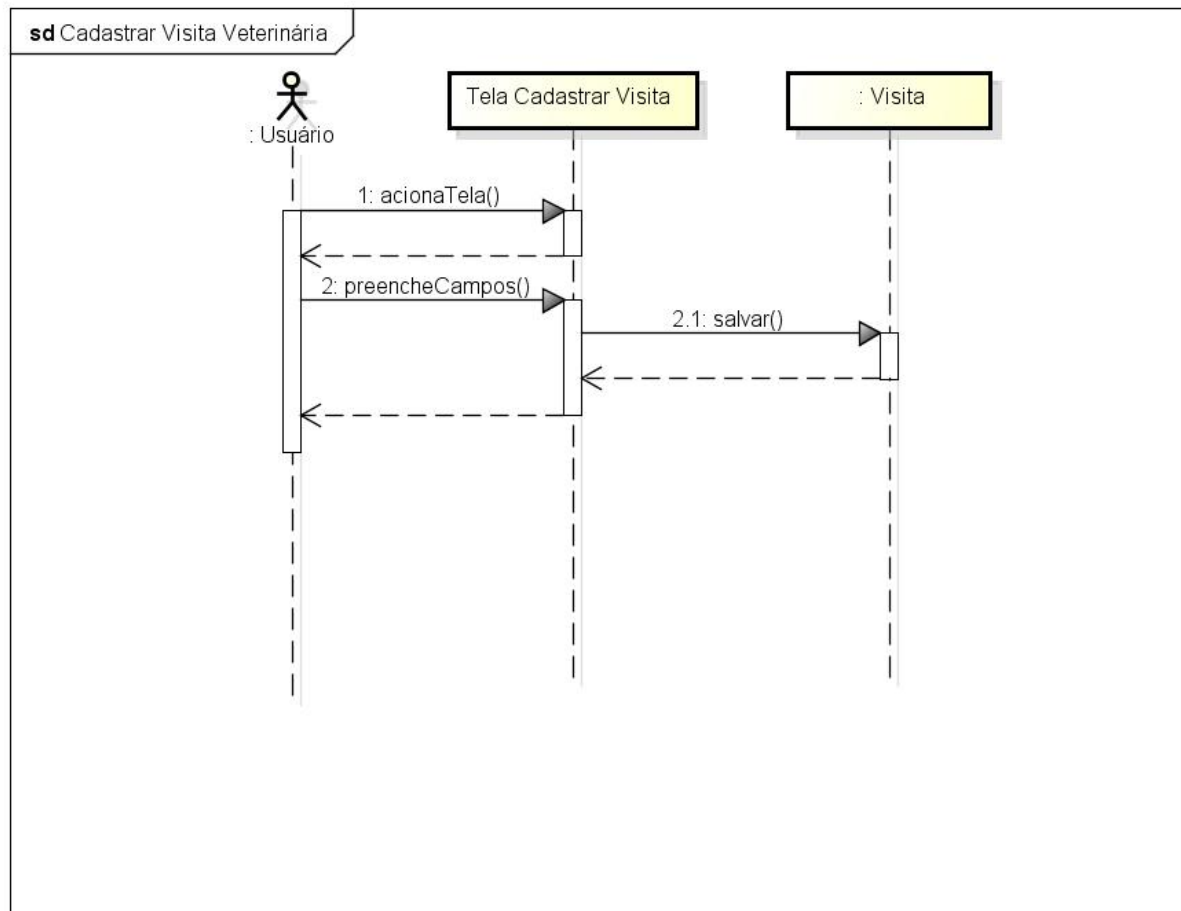
powered by Astah



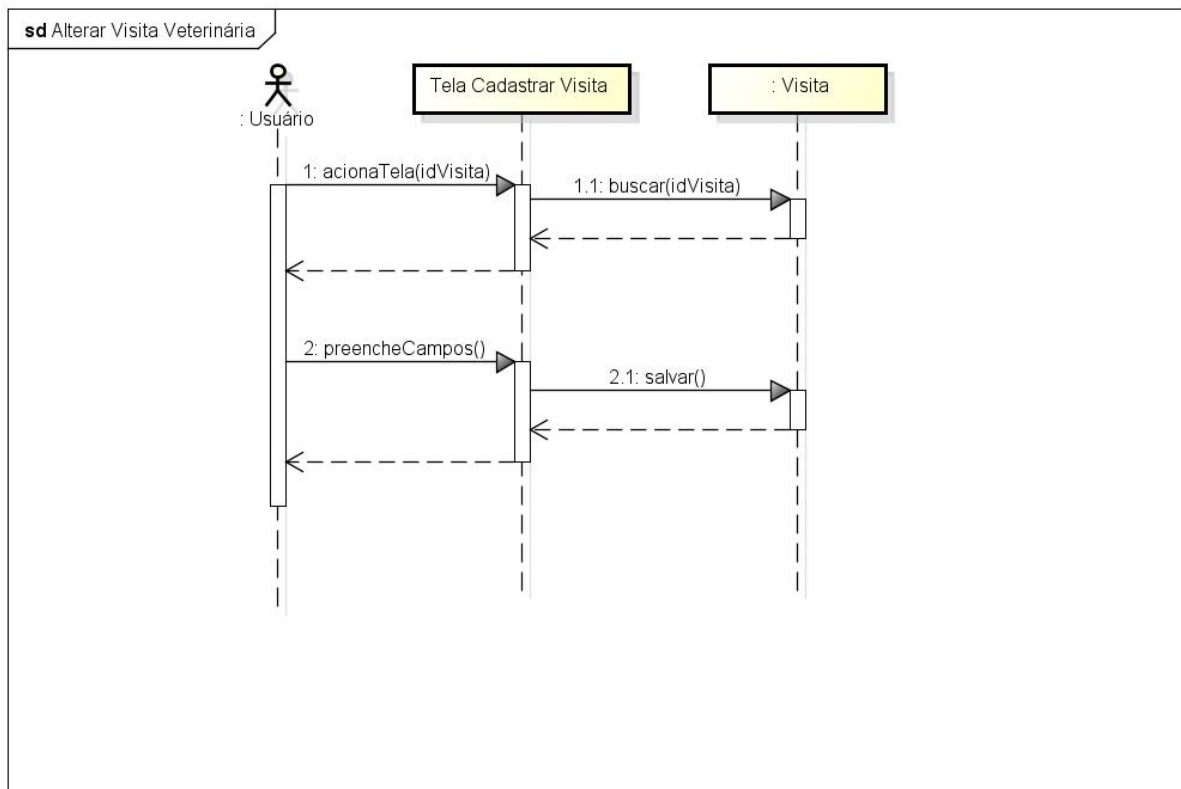
powered by Astah



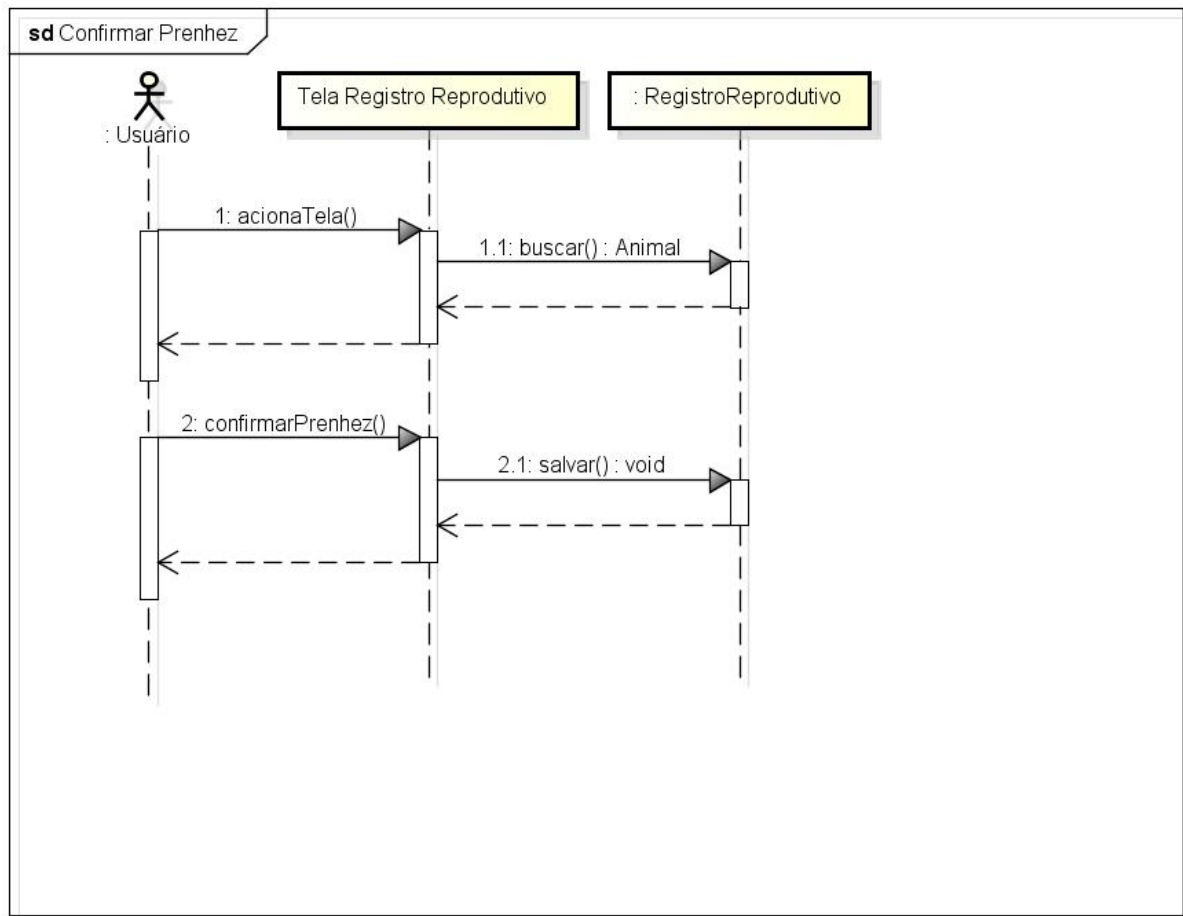
powered by Astah



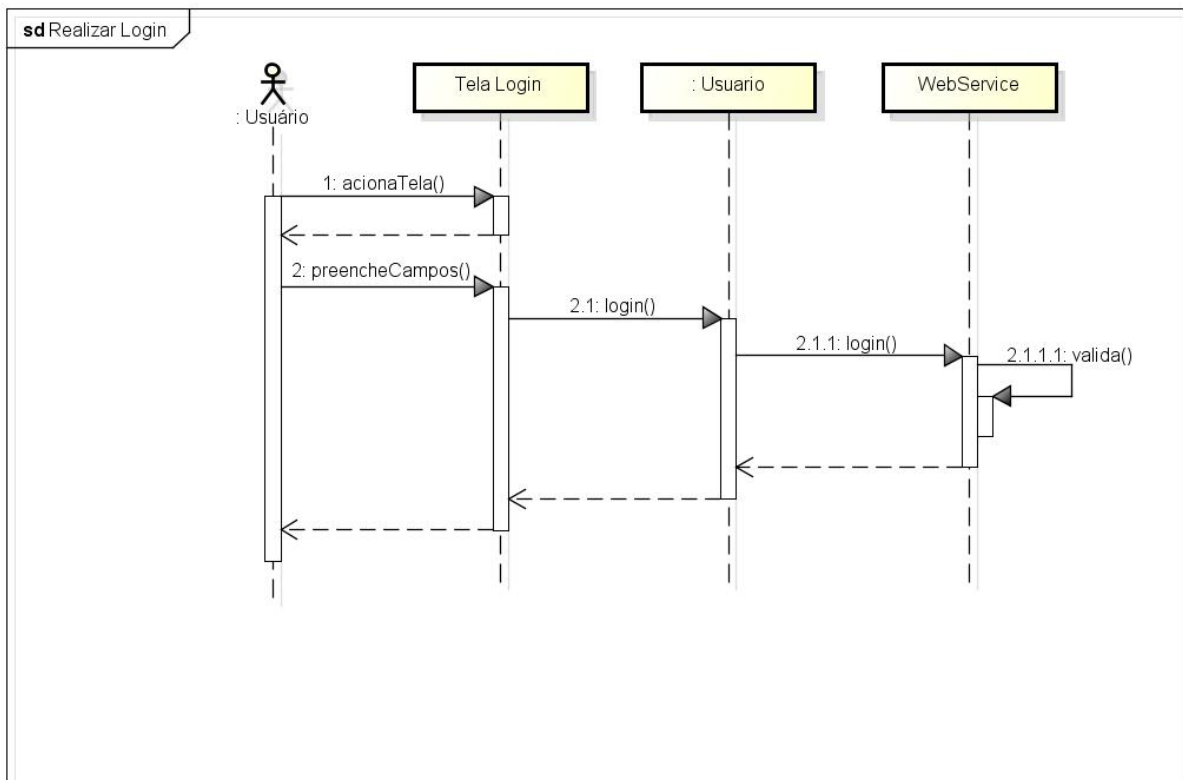
powered by Astah



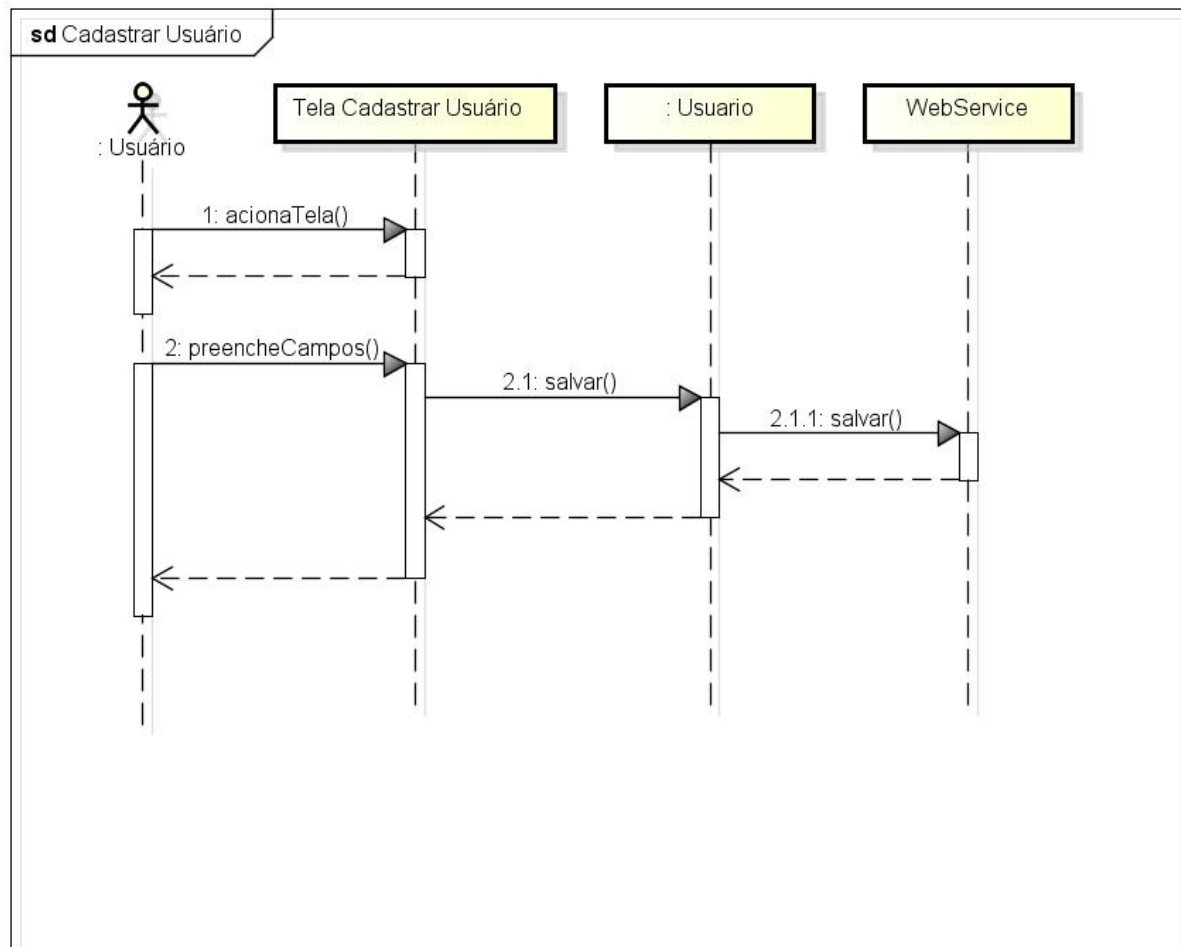
powered by Astah

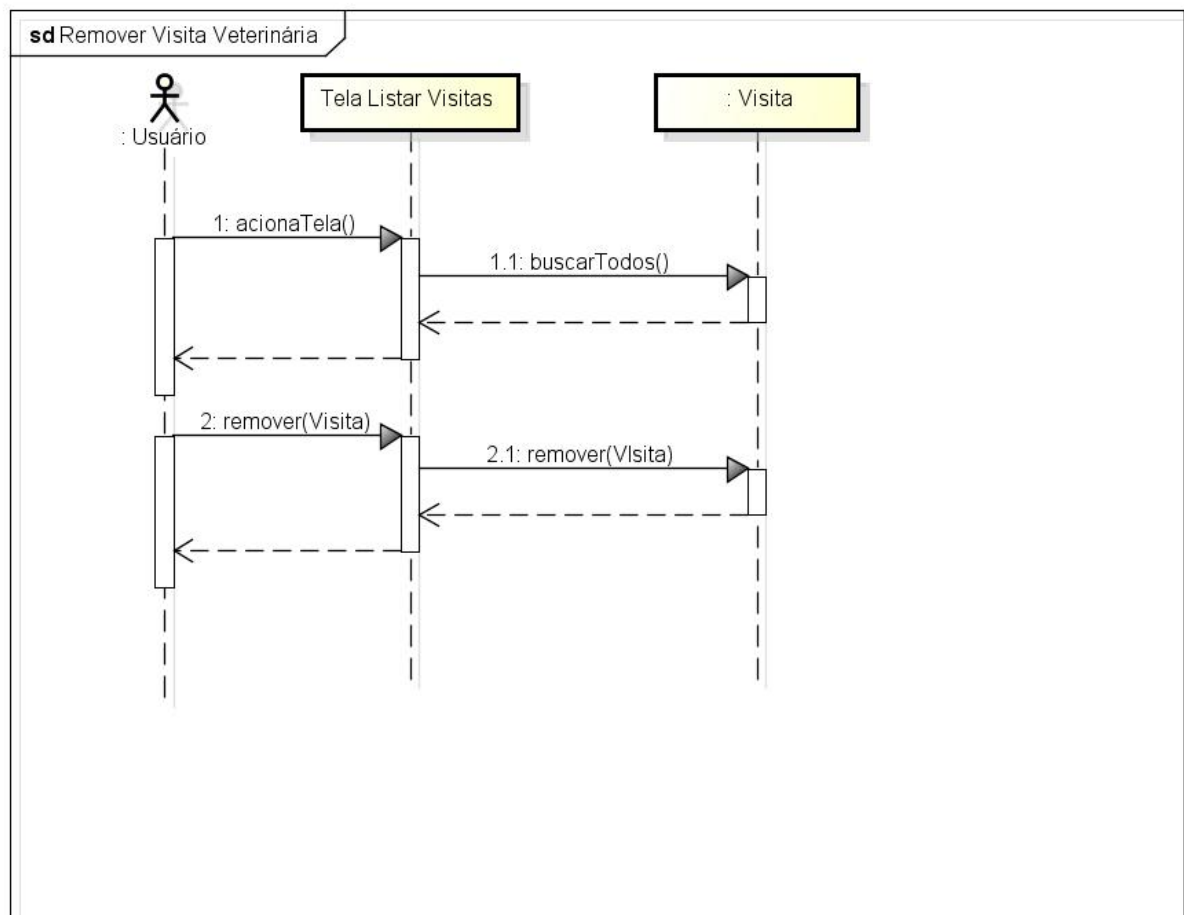


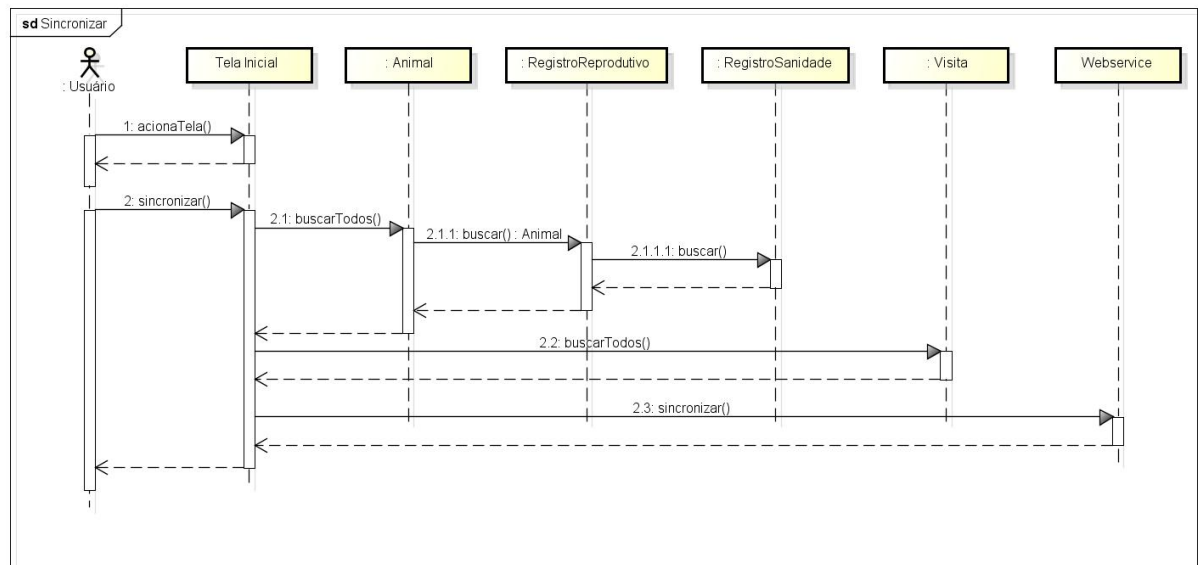
powered by Astah



powered by Astah







APÊNDICE H - PLANO DE TESTES

Caso de Uso / Requisito	Tipo de Teste	Técnica de teste	Nível de Teste	Caso de Teste	Ator
UC01 – Cadastrar Entrada Animal	Funcionalidade	Funcional	Teste de Sistema	CT01 – Preencher todos os campos e clicar no botão Salvar. CT02 – Não preencher o campo brinco e clicar no botão Salvar CT03 – Inserir um brinco que já existe e clicar no botão Salvar.	Usuário
UC02 – Cadastrar Saída Animal	Funcionalidade	Funcional	Teste de Sistema	CT04 – Selecionar o animal e confirmar a saída. CT05 – Selecionar o animal e cancelar a saída.	Usuário
UC03 – Alterar Animal	Funcionalidade	Funcional	Teste de Sistema	CT06 – Preencher todos os campos e clicar no botão Salvar. CT07 – Não preencher o campo data nascimento e clicar no botão Salvar. CT08 – Validar se o campo brinco está bloqueado para edição.	Usuário
UC04 – Cadastrar Usuário	Funcionalidade	Funcional	Teste de Sistema	CT09 - Preencher todos os campos e clicar no botão Salvar. CT10 – Não preencher um dos dados obrigatórios e clicar no botão Salvar.	Usuário
UC05 – Cadastrar Registro Sanidade	Funcionalidade	Funcional	Teste de Sistema	CT11 - Preencher todos os campos e clicar no botão Salvar. CT12 – Não preencher a data e clicar no botão Salvar.	Usuário
UC06 – Alterar Registro	Funcionalidade	Funcional	Teste de Sistema	CT13 - Preencher todos os campos e clicar no botão	Usuário

Caso de Uso / Requisito	Tipo de Teste	Técnica de teste	Nível de Teste	Caso de Teste	Ator
Sanidade				Salvar. CT14 – Não preencher um dos dados obrigatórios e clicar no botão Salvar.	
UC07 – Alterar Visita Veterinária	Funcionalidade	Funcional	Teste de Sistema	CT15 - Preencher todos os campos e clicar no botão Salvar. CT16 – Não preencher um dos dados obrigatórios e clicar no botão Salvar.	Usuário
UC08 – Cadastrar Visita Veterinária	Funcionalidade	Funcional	Teste de Sistema	CT17 - Preencher todos os campos e clicar no botão Salvar. CT18 – Não preencher um dos dados obrigatórios e clicar no botão Salvar.	Usuário
UC09 – Remover Visita Veterinária	Funcionalidade	Funcional	Teste de Sistema	CT19– Selecionar o animal e confirmar a exclusão. CT20 – Selecionar o animal e cancelar a exclusão.	Usuário
UC10 – Confirmar Prenhez	Funcionalidade	Funcional	Teste de Sistema	CT21 – Preencher o status e a data e clicar no botão Salvar. CT22 – Preencher com uma data superior ao dia atual.	Usuário
UC11 – Alterar Registro Reprodutivo	Funcionalidade	Funcional	Teste de Sistema	CT23 - Preencher todos os campos e clicar no botão Salvar. CT24 – Não preencher um dos dados obrigatórios e clicar no botão Salvar.	Usuário
UC12 – Cadastrar Registro Reprodutivo	Funcionalidade	Funcional	Teste de Sistema	CT25 - Preencher todos os campos e clicar no botão Salvar. CT26 – Não preencher um dos dados obrigatórios e clicar no botão Salvar.	Usuário

Caso de Uso / Requisito	Tipo de Teste	Técnica de teste	Nível de Teste	Caso de Teste	Ator
UC13 – Gerar Relatórios	Interface	Funcional	Teste de Sistema	CT27 – Validar geração da ficha do animal.	Usuário
UC14 – Realizar Login	Funcionalidade	Funcional	Teste de Sistema	CT28 - Preencher todos os campos e clicar no botão Login. CT29 – Não preencher um dos dados obrigatórios e clicar no botão Login.	Usuário
UC15 – Sincronizar Dados	Funcionalidade	Funcional	Teste de Sistema	CT30 – Validar a sincronização dos dados. CT31 – Validar a sincronização sem acesso a internet.	Usuário