

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

MIGRAÇÃO DO SISTEMA ARUANA-MALETA PARA
MULTIPLATAFORMA

JORGE ALEXANDRE HWIZDALECH FILHO

BLUMENAU
2015

2015/2-07

JORGE ALEXANDRE HWIZDALECH FILHO

**MIGRAÇÃO DO SISTEMA ARUANA-MALETA PARA
MULTIPLATAFORMA**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Sistemas de Informação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Prof. Matheus Carvalho Viana, Doutor - Orientador

**BLUMENAU
2015**

2015/2-07

MIGRAÇÃO DO SISTEMA ARUANA-MALETA PARA MULTIPLATAFORMA

Por

JORGE ALEXANDRE HWIZDALECH FILHO

Trabalho de Conclusão de Curso aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II pela banca examinadora formada por:

Presidente:

Prof. Matheus Carvalho Viana, Titulação – Orientador, FURB

Membro:

Prof. Francisco Adell Péricas, Me. – FURB

Membro:

Prof. Miguel Alexandre Wisintainer, Me. – FURB

Blumenau, 09 de dezembro de 2015.

Dedico este trabalho aos meus pais, Jorge Alexandre Hwizdalech (*in memoriam*) e Aparecida Lurdes Krauss Hwizdalech e a minha namorada Ana Paula Prim por todo apoio que me deram nesta minha trajetória.

AGRADECIMENTOS

Primeiramente aos meus pais, Jorge Alexandre Hwizdalech (*in memoriam*) e Aparecida Lurdes Krauss Hwizdalech, que sempre me incentivaram e me apoiaram no que foi preciso para conclusão desta etapa da minha vida.

Aos amigos que ganhei durante esta jornada, que sempre me ajudaram muito. Em especial ao Ramon Valentim da Silva que me ajudou com o desenvolvimento na ferramenta Delphi.

A minha namorada, Ana Paula Prim, que me apoiou e cobrou também o andamento do projeto.

Ao meu orientador, o Prof. Dr. Matheus Viana, que assumiu este compromisso de forma inesperada, mas que foi fundamental para a conclusão.

Ao Prof. Mauro Marcelo Mattos, que infelizmente não pode acompanhar o desenvolvimento do projeto, mas que foi o orientador da proposta e quem tornou este projeto possível.

RESUMO

Este trabalho teve por objetivo realizar a migração do sistema desktop de rastreabilidade bovina, intitulado Aruana-Maleta e desenvolvido pelo Laboratório de Desenvolvimento e Transferência de Tecnologia da FURB (LDTT), para um aplicativo *mobile* multiplataforma. Assim como na versão desktop, a aplicativo mobile permite registrar os dados dos animais de uma propriedade rural, bem como a compra, venda, nascimento e morte desses animais. Além disso, o aplicativo pode importar dados da versão desktop, ser utilizado em modo off-line e exportar novamente os dados atualizados para o sistema desktop. A principal vantagem é a praticidade proporcionada pelo dispositivo móvel para o acompanhamento dos animais dentro da propriedade rural. A versão mobile do aplicativo foi desenvolvida no ambiente Delphi 10 Seattle, que permite geração de uma versão do aplicativo para cada uma das principais plataformas de dispositivos móveis a partir de uma única implementação feita em uma linguagem própria do ambiente. Contudo, por questões de licença, somente a versão do aplicativo para a plataforma Android pôde ser gerada e testada.

Palavras-chave: Agronegócio. Rastreabilidade de animais. Desenvolvimento *mobile*. Multiplataforma.

ABSTRACT

This work has aimed to migrate desktop beef traceability system, entitled Aruana-Maleta and developed by the Development Laboratory and FURB Technology Transfer (LDTT), to a multiplatform mobile application. As the desktop version, the mobile application allows to record the data of animals in a rural property, as well as the purchase, sale, birth and death of these animals. In addition, the application can import data from the desktop version, be used in offline mode and re-export the updated data to the desktop system. The main advantage is the convenience provided by the mobile device to monitor the animals in the farm. The mobile version has been developed in the Delphi Seattle 10 environment, which allows generation of an application version to each of the major mobile platforms from a single implementation. However, due to licensing reasons, only the application version for the Android platform could be generated and tested.

Key-words: Agribusiness. Animal traceability. Mobile development. Multiplatform.

LISTA DE FIGURAS

Figura 1 - Diagrama de casos de uso do projeto Aruana.....	15
Figura 2 - Cadastro de animal do Aruana-Mobile.....	16
Figura 3 - Versão online - A3 Software para gestão pecuária.....	17
Figura 4 - Casos de Uso.....	19
Figura 5 - Modelo Entidade-Relacionamento - Simplificado	20
Figura 6 - Diagrama das classes do pacote form.....	21
Figura 7 - Diagrama das classes do pacote db.....	21
Figura 8 - Diagrama de classes do pacote modelo.....	22
Figura 9 - Delphi - Desenvolvimento interface.....	23
Figura 10 - Definições de interface para diferentes plataformas.....	24
Figura 11 - Validação da plataforma que está executando.....	24
Figura 12 - Sequência de telas iniciais	25
Figura 13 - Tela de cadastro de animal	26
Figura 14 - Telas para o cadastro de compra.....	26
Figura 15 - Cadastro de nascimento	27
Figura 16 - Registro de morte.....	27
Figura 17 - Venda de animais.....	28
Figura 18 - Transferir animais/lotos	28
Figura 19 - Pesagem de animais.....	29
Figura 20 - Mudança de Pasto	29
Figura 21 - Cadastro de desmame	30
Figura 22 - Descarte de animais	30
Figura 23 - Telas para importação e exportação de dados.	31
Figura 24 - Modelo Entidade-Relacionamento - Completo	50

LISTA DE QUADROS

Quadro 1 - Requisitos Não Funcionais.....	18
Quadro 2 - Requisitos Funcionais	19
Quadro 3 - Comparativo entre o projeto proposto e seus correlatos.	32
Quadro 4 - Descrição do Caso de Uso 01 - Efetuar login.....	35
Quadro 5 - Descrição do Caso de Uso 02 - Selecionar Retiro	35
Quadro 6 - Descrição do Caso de Uso 03 - Cadastrar Animal.....	36
Quadro 7 - Descrição do Caso de Uso 04 - Cadastrar Compra.....	37
Quadro 8 - Descrição do Caso de Uso 05 - Cadastrar Nascimento	38
Quadro 9 - Descrição do Caso de Uso 06 - Cadastrar Morte	39
Quadro 10 - Descrição do Caso de Uso 07 - Cadastrar Venda	40
Quadro 11 - Descrição do Caso de Uso 08 - Cadastrar Transferência.....	41
Quadro 12 - Descrição do Caso de Uso 09 - Cadastrar Pesagem	42
Quadro 13 - Descrição do Caso de Uso 10 - Cadastrar Mudança de Pasto.....	43
Quadro 14 - Descrição do Caso de Uso 11 - Cadastrar Desmame	44
Quadro 15 - Descrição do Caso de Uso 12 - Cadastrar Descarte.....	45
Quadro 16 - Descrição do Caso de Uso 13 - Realizar Importação.....	46
Quadro 17 - Descrição do Caso de Uso 14 - Realizar Exportação.....	46
Quadro 18 - Dicionário de dados do aplicativo.....	48

LISTA DE ABREVIATURAS E SIGLAS

FURB - Universidade Regional de Blumenau

IDC - *International Data Corporation*

LDTT - Laboratório de Desenvolvimento e Transferência de Tecnologia

MER - Modelo de Entidade-Relacionamento

NUPEX - Núcleo de Pesquisa e Extensão Universitária

PIB – Produto Interno Bruto

SISBOV - Serviço de Rastreabilidade da Cadeia Produtiva de Bovinos e Bubalinos

UML - *Unified Modeling Language*

XML - *eXtensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO.....	11
1.1 OBJETIVOS.....	11
1.2 ESTRUTURA.....	12
2 FUNDAMENTAÇÃO TEÓRICA.....	13
2.1 DESENVOLVIMENTO MOBILE	13
2.2 RASTREABILIDADE BOVINA	14
2.3 PROJETO ARUANA.....	14
2.4 TRABALHOS CORRELATOS.....	15
2.4.1 Migração do sistema Aruana-Maleta para a plataforma Android.....	15
2.4.2 A3: Sistema para gestão pecuária	16
3 DESENVOLVIMENTO.....	18
3.1 LEVANTAMENTO DE INFORMAÇÕES	18
3.2 ESPECIFICAÇÃO	18
3.2.1 Casos de Uso	19
3.2.2 Modelo de Entidade-Relacionamento	20
3.2.3 Diagrama de Classes	20
3.3 IMPLEMENTAÇÃO	23
3.3.1 Técnicas e ferramentas utilizadas.....	23
3.3.2 Operacionalidade da implementação	24
3.4 RESULTADOS E DISCUSSÕES.....	31
4 CONCLUSÕES.....	33
4.1 EXTENSÕES	33
REFERÊNCIAS	34
APÊNDICE A – DESCRIÇÃO DOS CASOS DE USO.....	35
APÊNDICE B – DICIONÁRIO DE DADOS.....	47
APÊNDICE C – MODELO ENTIDADE-RELACIONAMENTO	50

1 INTRODUÇÃO

O agronegócio brasileiro representa cerca de aproximadamente 23% do Produto Interno Bruto (PIB) Brasileiro, sendo, portanto, um mercado representativo na economia (BRASIL, 2013). Esse mercado está caminhando para a próxima década com foco em competitividade e modernidade. Seus recentes estudos apontam para um grande potencial, principalmente para os setores agropecuário e florestal, nos próximos anos (BRASIL, 2013).

Em um ritmo considerável, o mercado de dispositivos móveis também vem crescendo no Brasil, batendo recordes de vendas tanto para *smartphones* como *tablets*. A venda de *smartphones*, por exemplo, teve um aumento de 11% no terceiro bimestre 2014, em relação ao anterior. Esse crescimento é ainda maior se comparado com o mesmo período do ano de 2013, quando ocorreu um aumento de 49% nas vendas, conforme estudos da International Data Corporation (IDC) (IDC BRASIL, 2014).

Dispositivos móveis podem facilitar bastante a realização do controle da produção, dando portabilidade aos produtores que podem realizar o cadastro das informações já no campo. Por exemplo, a rastreabilidade bovina se tornou uma necessidade nos últimos anos, devido à exigência do mercado externo, principalmente da Europa, de ter acesso aos dados da produção (MENDES, 2006).

Em 2007, a Universidade Regional de Blumenau (FURB), através do Laboratório de Desenvolvimento e Transferência de Tecnologia (LDTT), desenvolveu um projeto para a gestão de rastreamento de rebanho bovino conjugado à pesquisa científica, inovação tecnológica e integração universidade/empresa. Esse projeto recebeu o nome de Aruana e seu escopo envolveu um conjunto de tecnologias para apoio à gestão pecuária, desde a produção dos brincos de identificação até o desenvolvimento de um sistema para gestão pecuária, intitulado Aruana-Maleta (DEMARCHI, 2012). Continuando o projeto, em 2012, foi implementada uma versão desse sistema para a plataforma Android, denominada Aruana-Mobile, desenvolvida no Trabalho de Conclusão de Curso (TCC) do estudante Felipe Demarchi, da FURB (DEMARCHI, 2012). Contudo, como existe uma diversidade de plataformas de dispositivos móveis, percebeu-se a necessidade de uma reimplementação do Aruana-Mobile para uma tecnologia multiplataforma.

1.1 OBJETIVOS

O objetivo geral deste trabalho é expandir o trabalho de Demarchi (2012) generalizando a versão Aruana-Mobile para uma tecnologia multiplataforma.

Os objetivos específicos do trabalho são:

- a) fazer uso das vantagens dos dispositivos móveis para auxiliar os pecuaristas no cadastramento dos dados dos animais em plena propriedade rural;
- b) migrar a estrutura do Aruana-Maleta para a nova arquitetura do Delphi 10 Seattle, que permite a geração do aplicativo para múltiplas plataformas;
- c) validar a aplicação por meio de estudos de caso com a versão original do Aruana-Maleta.

1.2 ESTRUTURA

A estrutura deste trabalho está dividida em quatro capítulos. No primeiro é apresentada a introdução ao projeto, os objetivos gerais e específicos do mesmo e a estrutura adotada no trabalho.

O segundo capítulo trata da fundamentação teórica utilizada para o desenvolvimento do trabalho, mais especificamente sobre desenvolvimento *mobile*, rastreabilidade bovina e o projeto Aruana. Além disso, são apresentados alguns trabalhos correlatos.

O terceiro capítulo contempla o desenvolvimento do trabalho, apresentando os requisitos do aplicativo, a especificação do aplicativo, as técnicas e ferramentas para a implementação e sua operacionalidade. No final deste capítulo, são apresentados os resultados e discussões a respeito do trabalho desenvolvido.

Para concluir o trabalho, o quarto capítulo aborda as conclusões e contribuições, bem como as sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos a serem apresentados nas seções a seguir, tais como desenvolvimento *mobile*, rastreabilidade bovina, projeto Aruana, além de trabalhos correlatos.

2.1 DESENVOLVIMENTO MOBILE

Desenvolvimento *mobile* refere-se ao processo de construção de aplicativos para dispositivos móveis, como *smartphones*, *tablets*, *folets* e outros dispositivos de bolso. Trazendo recursos e aplicações antes disponíveis apenas para os computadores *desktop* para perto do usuário, aumentando a portabilidade e comodidade (DEITEL et al., 2013).

Por causa das restrições de hardware dos dispositivos móveis, os aplicativos *mobile* possuem algumas características que os diferenciam dos aplicativos *desktop*. Primeiramente, devido ao tamanho de tela reduzido, a interface dos aplicativos *mobile* é deve ser mais simples e intuitiva. Há também restrições de memória e de bateria, o que obriga os ambientes de desenvolvimento de aplicativos *mobile* a serem capazes de gerar aplicações leves e eficientes. Outra característica é a limitação de espaço para armazenamento, o que faz com o que os bancos de dados também sejam reduzidos (DEITEL et al., 2013).

Outro problema para o desenvolvimento *mobile* é a existência de diferentes plataformas que mudam de versão frequentemente. Desenvolver uma versão do mesmo aplicativo para cada plataforma pode ser um processo inviável. Pensando nisso, alguns ambientes de desenvolvimento permitem o desenvolvimento em uma linguagem neutra, cujo código pode ser convertido para as principais plataformas (KAWATA, 2013).

Uma das principais ferramentas para desenvolvimento *mobile* é o ambiente de desenvolvimento Delphi. Esse ambiente vem sendo utilizado no desenvolvimento de aplicativos na plataforma Windows desde 1994. A partir da versão XE2, esse ambiente ganhou ferramentas para o desenvolvimento *mobile*, porém somente para a plataforma iOS. Mais tarde, já com a versão XE5, o Delphi passou a oferecer suporte para desenvolvimento de aplicativos para as plataformas iOS e Android, utilizando a mesma sintaxe, linguagem e componentes utilizados para o desenvolvimento e aplicações *desktop* (KAWATA, 2013). Na versão XE7, foram incluídos componentes de conectividade com a base de dados que podem ser utilizados em diversas plataformas *mobile* (KAWATA, 2015). Na versão mais nova, denominada Delphi 10 Seattle, o ambiente ficou mais leve, permitindo construir e modificar aplicações para múltiplas plataformas com maior eficiência (EMBARCADERO, 2015).

2.2 RASTREABILIDADE BOVINA

A rastreabilidade bovina se tornou uma necessidade para o Brasil depois que o mercado europeu, um dos principais consumidores dos produtos agropecuários brasileiros, passou a exigir conhecimento sobre o produto exportado (MENDES; OLIVEIRA; SANTOS, 2011). Para suprir a exigência do mercado europeu, foi implementado o Serviço de Rastreabilidade da Cadeia Produtiva de Bovinos e Bubalinos (SISBOV), que consiste na certificação de uma série de ações para garantir a origem, a sanidade e a segurança do alimento proveniente do gado brasileiro (SERPRO, 2008).

“O processo de rastreamento fornece além de uma quantidade relevante de informações ao processo de tomada de decisão, certificação de qualidade ao produto da propriedade. Com dados significativos do rebanho, dos processos produtivos, é possível assegurar a qualidade do produto vendido ao mercado, assim aumentando a confiança do cliente [...]” (CORÁ, 2009, p. 10)

Não só ao mercado externo, a rastreabilidade garante uma melhor qualidade e procedência para o produto comercializado no mercado nacional, chegando ao brasileiro um produto que segue os mesmos padrões de qualidades exigidos para exportação. A segurança alimentar vem ganhando importância nos últimos anos, em virtude da presença de um mercado consumidor mais exigente. Nesse contexto a rastreabilidade é fundamental para garantir a procedência do animal, porém, atualmente é exigida apenas a rastreabilidade dos últimos 90 dias de vida do animal e há também uma falta de padronização para a identificação dos animais, onde os brincos e códigos de barras não se mostram eficientes dando lugar ao chip, que se apresenta com uma das melhores soluções se considerar o reuso (MENDES; OLIVEIRA; SANTOS, 2011).

2.3 PROJETO ARUANA

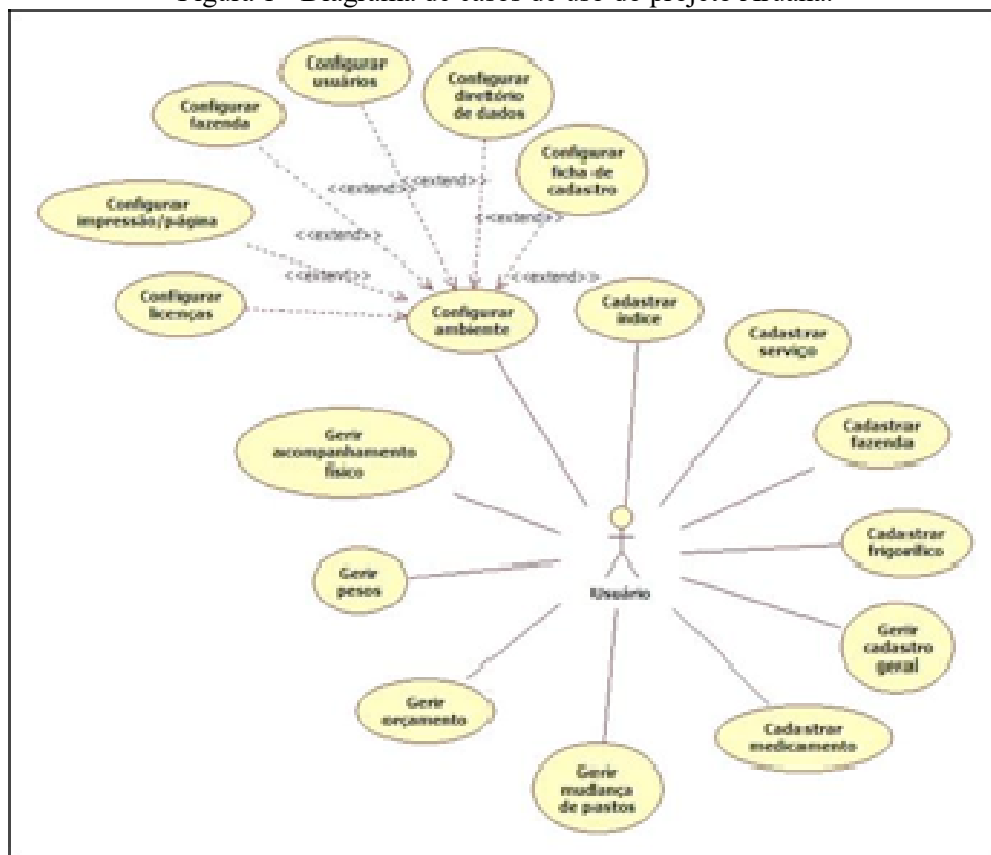
A FURB, em 2007, firmou um contrato de pesquisa e cooperação técnica com Ideatec Soluções em Tecnologia Agroindustrial Ltda., por meio do Núcleo de Pesquisa e Extensão Universitária (NUPEX). O objetivo era desenvolver um *software* especializado na gestão de rastreamento de rebanho bovino, conjugados a pesquisa científica, inovação tecnológica e integração universidade/empresa (EHLERT, 2009, p.21).

O sistema desenvolvido pretende atender o seguinte cenário: a leitura ocorre na fazenda, onde os dados serão então exportados para o sistema. O escritório, normalmente localizado na praça da fazenda, faz os ajustes e complementação dos dados. O proprietário, a

partir dos grandes centros, acessa via Internet os relatórios e gráficos do manejo de suas fazendas. (MATTOS, 2015).

O escopo do projeto envolveu um conjunto de tecnologias para apoio à gestão pecuária envolvendo o ciclo completo desde a produção dos brincos de identificação até o software livre para a gestão pecuária (DEMARCHI, 2012). O *software* para rastreabilidade animal desenvolvido no LDTT permite, rapidamente, o resgate histórico do acompanhamento de toda a vida do animal, impulsionando um produto de qualidade garantida e assegurada ao consumidor. Na Figura 1 é apresentado o diagrama de casos de uso do projeto Aruana.

Figura 1 - Diagrama de casos de uso do projeto Aruana.



Fonte: Ehlert (2009, p. 23).

2.4 TRABALHOS CORRELATOS

Nos tópicos a seguir estão descritos os trabalhos correlatos ao projeto proposto, de forma que combinam em características funcionais e também não funcionais:

2.4.1 Migração do sistema Aruana-Maleta para a plataforma Android

O trabalho realizado por Felipe Demarchi (2012) resultou no desenvolvimento de uma versão do sistema Aruana-Maleta exclusiva para a plataforma Android. Essa nova versão, denominada Aruana-Mobile, teve como vantagem a mobilidade proporcionada pelos

dispositivos móveis. Sendo, portanto, melhor adaptada para o cadastramento dos animais dentro das propriedades rurais.

O aplicativo Aruana-Mobile, apresentado na Figura 2, conta com as principais funcionalidades do sistema Aruana-Maleta. As principais diferenças entre as duas versões se devem às limitações dos dispositivos móveis. Por exemplo, como os dispositivos móveis não suportam bancos de dados tão robustos quanto os dos computadores, o Aruana-Mobile trabalha com um conjunto reduzido de informações em relação ao Aruana-Maleta, mantendo somente os dados indispensáveis para a operação dos cadastros dos animais. Além disso, as restrições de processamento e de tamanho de tela fazem com que a interface com o usuário do Aruana-Mobile fosse diferente da versão *desktop*.

Figura 2 - Cadastro de animal do Aruana-Mobile

The image shows a mobile application interface for animal registration. The title bar at the top reads 'Cadastro de Animais'. Below it, there are several input fields and dropdown menus:

- 'Código do Manejo': A text input field with a red border.
- 'Código do SISBOV': A text input field.
- 'Era': A dropdown menu with '1 - Bezerro' selected.
- 'Raça': A dropdown menu with 'AF - Africander' selected.
- 'Data de Nascimento': A text input field.
- 'Lote': A text input field.
- 'Seleção de Pasto': A dropdown menu with '2' selected.

 At the bottom of the screen, there are three buttons: 'Menu', 'Limpar', and 'Salvar'. The status bar at the top right shows the time as 13h39.

2.4.2 A3: Sistema para gestão pecuária

Diferente do trabalho anterior, esse não se trata de um trabalho acadêmico e, sim, de uma ferramenta comercial desenvolvida com objetivos similares aos do projeto Aruana-Maleta. O principal objetivo do A3 é facilitar o controle sobre o rebanho e também realizar a rastreabilidade do animal durante o processo dentro da propriedade (A3 PECUÁRIA, 2015).

O sistema A3 possui duas versões uma totalmente em ambiente *web*, o que lhe garante maior portabilidade, porém o torna inviável para clientes que não possuem acesso à Internet nas propriedades rurais, e a segunda versão seria em ambiente Windows. Na Figura 3 é

apresentada a tela do sistema A3 na sua versão *web* que mostra as informações gerais registradas.

As principais funções do sistema A3 são: controle de animais, em lote ou individual; pesagens de animais; medicação; cobertura e sêmen para acompanhamento da reprodução; controle financeiro; compras e vendas; estoque; e atualização automática de categorias pela idade do animal. Portanto, além de gerenciar os animais da fazenda, o sistema também contempla a gestão financeira e de estoque de materiais. Para auxiliar nessa gestão, o sistema conta com alguns relatórios como compras de animais, medicações aplicadas, pesagens, vendas e custo de produção (A3 PECUÁRIA, 2015).

Figura 3 - Versão online - A3 Software para gestão pecuária

The screenshot shows the A3 Software interface for farm management. At the top, there's a green header with the A3 logo and 'Software para Gestão Pecuária'. Below it, a navigation menu includes 'Home', 'Manejo', 'Estoque', 'Financeiro', 'Cadastros', 'Relatórios', 'Minha Conta', and 'Sair'. The main content area is divided into several sections:

- Welcome:** 'Bem Vindo, ADMINISTRADOR. Hoje é 08 Junho 2014, seu nível de acesso é ADMINISTRADOR.' A search bar for animals is present.
- Resumo Diário:** Lists various farm areas and animal counts, such as 'NUMERADOS 30 | S/NUMERAÇÃO 1036', 'PASTO NATURAL VERÃO | 2 animais', 'PASTO AVEIA/AZEVÉM SUL | 35 animais', etc.
- Calendário de Manejo Mensal (edit):** Shows monthly management events, including 'Junho | BOVINO | ESTAÇÃO DE MONTA' and 'Junho | BOVINO | RINOTRAQUEÍTE INFECCIOSA'.
- Informações Financeiras:** Displays financial data like 'Animais: - Valor compra | R\$ 58777.34', 'Custo alimentação | R\$ 0.00', and 'Total em produtos | R\$ 616.20'. It also lists recent revenues and expenses.
- Quick Links:** A grid of links for 'Animais', 'Controle de Estoque', 'Financeiro', and 'Relatórios', each with sub-links like 'Lote de Animais', 'Produtos', 'Receitas', etc.

Fonte: A3 Pecuária (2015).

3 DESENVOLVIMENTO

Nesta seção são abordadas as etapas envolvidas no desenvolvimento do aplicativo, organizadas da seguinte forma: a Seção 3.1 contém as informações levantadas para o desenvolvimento do aplicativo proposto; na Seção 3.2 é apresentada a especificação; na Seção 3.3 é descrita a implementação, abordando as técnicas e ferramentas utilizadas e a operacionalidade dessa implementação; e, por fim, na Seção 3.4 são listados os resultados e discussões do trabalho.

3.1 LEVANTAMENTO DE INFORMAÇÕES

O projeto realizou a migração do sistema atual, Aruana-Maleta para a ferramenta Delphi 10 Seattle, de tal maneira que seja possível gerar o aplicativo para execução em mais de uma plataforma, o mesmo foi validado para a plataforma Android. O aplicativo conta com os principais cadastros e movimentações da versão *desktop*, como o cadastro de novos animais, nascimento, morte, compra, venda e assim por diante, como também executa a importação e exportação de dados através de arquivos eXtensible Markup Language (XML) para serem utilizados no sistema Aruana-Maleta.

A principal fonte de informações para o desenvolvimento do projeto proposto foi a monografia do Felipe Demarchi (2012). Além disso, algumas informações a respeito do sistema Aruana-Maleta foram obtidas com o professor Mauro Marcelo Mattos, responsável pelo LDTT. Entretanto, houve dificuldade na obtenção de informações detalhadas, pois o projeto Aruana não está mais sendo desenvolvido no LDTT, de modo que não foi possível ter acesso ao sistema Aruana-Maleta.

3.2 ESPECIFICAÇÃO

O Quadro 1 apresenta os requisitos não funcionais definidos para o software.

Quadro 1 - Requisitos Não Funcionais

Requisitos Não Funcionais	
RNF01	O sistema deverá ser implementado na ferramenta e utilizando a linguagem do Delphi XE7
RNF02	Apresentar as mesmas características de interface com o usuário da versão Aruana-Maleta

O software deve atender os requisitos funcionais apresentados no Quadro 2, onde também são indicados os respectivos casos de uso para cada requisito.

Quadro 2 - Requisitos Funcionais

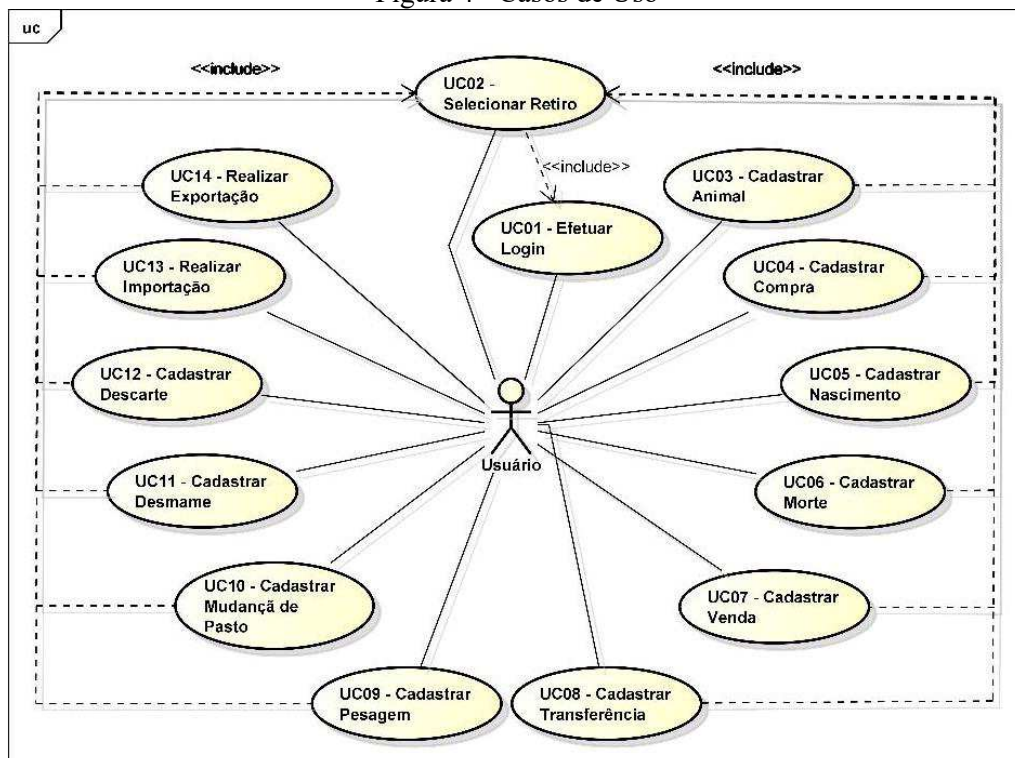
Requisitos Funcionais		Caso de Uso
RF01	Permitir o <i>login</i> do usuário cadastrado	UC01
RF02	Permitir o gerenciamento em nível de retiros na fazenda	UC02
RF03	Permitir o cadastro de novos animais	UC03
RF04	Permitir registrar a compra de animais	UC04
RF05	Permitir registrar o nascimento de animais	UC05
RF06	Permitir registrar a morte de animais	UC06
RF07	Permitir registrar a venda de animais	UC07
RF08	Permitir efetuar a transferência de animais	UC08
RF09	Permitir o registro da pesagem de animais	UC09
RF10	Permitir o registro da mudança de pastos entre os lotes	UC10
RF11	Permitir o registro de desmama	UC11
RF12	Permitir o registro de descarte de animais	UC12
RF13	Efetuar a importação e exportação de dados entre o aplicativo multiplataforma e o sistema Aruana-Maleta	UC13
RF14	Efetuar a transferência do arquivo de importação e exportação entre o aplicativo multiplataforma e o sistema Aruana-Maleta através de arquivo XML.	UC14

Após a especificação dos requisitos, foi feita a modelagem do sistema proposto com o uso da *Unified Modeling Language* (UML). A ferramenta Astah Community 6.9.0 foi a adotada para a criação dos modelos de casos de uso e de classes do sistema proposto.

3.2.1 Casos de Uso

Nesta seção será apresentado o diagrama de casos de uso, na Figura 4, que definem o aplicativo. As descrições estão dispostas Apêndice A.

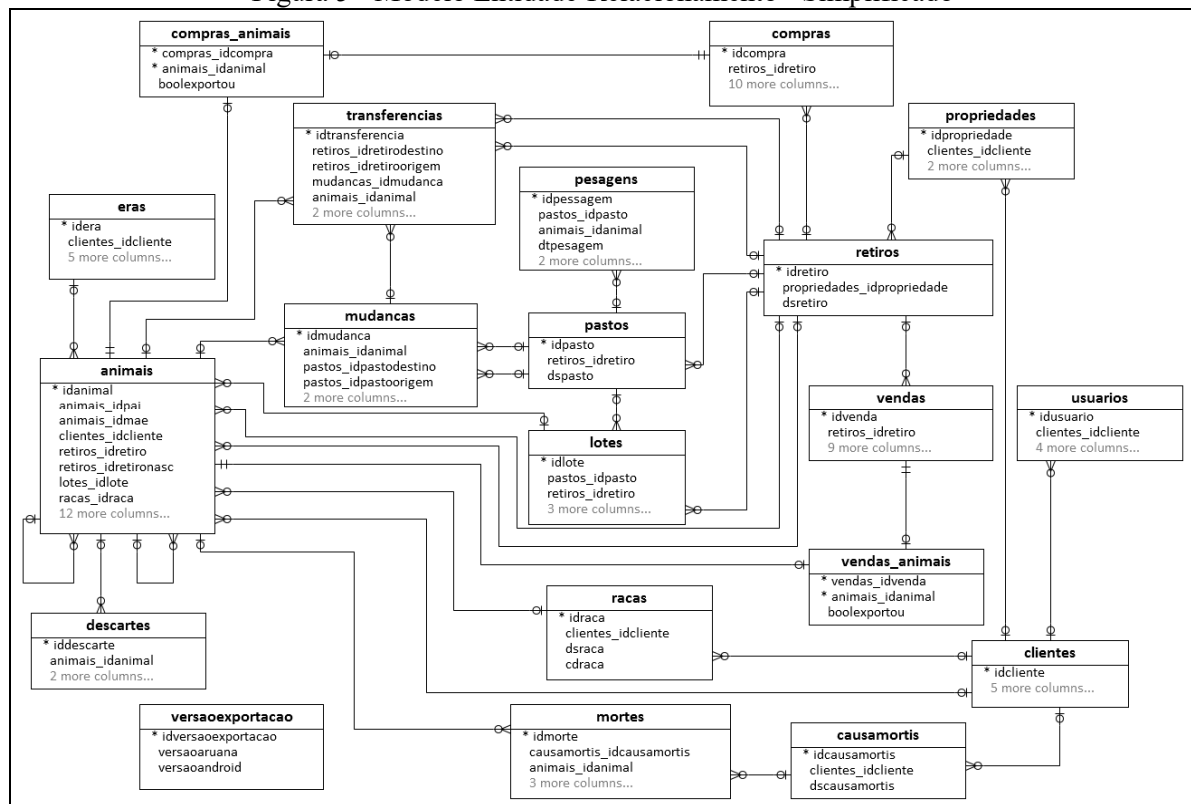
Figura 4 - Casos de Uso



3.2.2 Modelo de Entidade-Relacionamento

Nesta seção é apresentado o Modelo de Entidade-Relacionamento (MER) da base de dados utilizada pelo aplicativo e modelada para o SQLite, banco de dados utilizado nas plataformas Android e iOS. A Figura 5 apresenta o modelo de forma simplificada, permitindo uma percepção da estrutura da base de dados. O Modelo Entidade-Relacionamento completo encontra-se no Apêndice C.

Figura 5 - Modelo Entidade-Relacionamento - Simplificado



O aplicativo desenvolvido neste trabalho importa do sistema Aruana-Maleta algumas informações e estas são armazenadas nas tabelas *causamortis*, *clientes*, *eras*, *pastos*, *propriedades*, *racas*, *retiros* e *usuários*. Estas informações não são manipuladas pelo aplicativo. As demais tabelas armazenam as informações pertinentes aos animais e são manipuladas pelo aplicativo.

Para complementar as informações do MER, no Apêndice B está disponível o dicionário de dados.

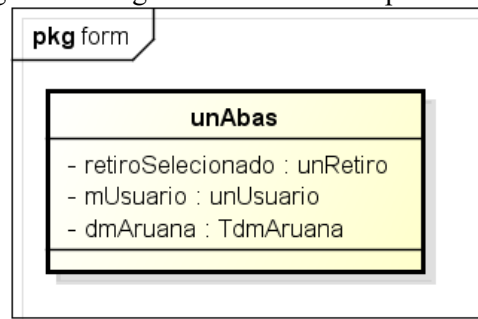
3.2.3 Diagrama de Classes

A arquitetura do aplicativo foi dividida em três camadas: formulários (interface gráfica), banco de dados e modelo. Cada camada foi implementada em um pacote específico, respectivamente, *form*, *db* e *modelo*. O aplicativo foi desenvolvido a partir da interface

passando pela estrutura da conexão com o banco de dados e por último a modelagem dos objetos da base de dados.

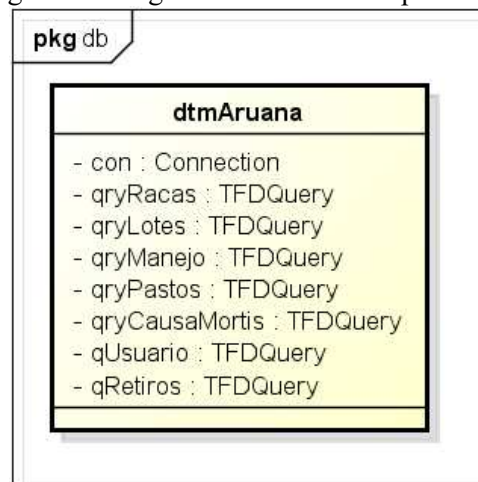
O pacote `form`, apresentado na Figura 6, é responsável pela interface com o usuário do aplicativo. Esse pacote contém a classe `unAbas`, que define o formulário principal da aplicação com as respectivas telas de cadastro definidas em abas. Esse formato com uma única janela e várias abas foi utilizado devido a limitações na estrutura estabelecida pela ferramenta de desenvolvimento, em que o Delphi apresentou problemas com a instância e destruição de formulários, havendo sobre carga da memória do dispositivo, pois os formulários não são corretamente removidos da memória.

Figura 6 - Diagrama das classes do pacote `form`.



O pacote `db`, conforme é mostrado no modelo da Figura 7, contém a classe `dtmAruana`, que é responsável pela conexão com a base de dados. Essa classe possui componentes do tipo `TFDQuery`, que fazem a representação e a comunicação com as tabelas da base de dados.

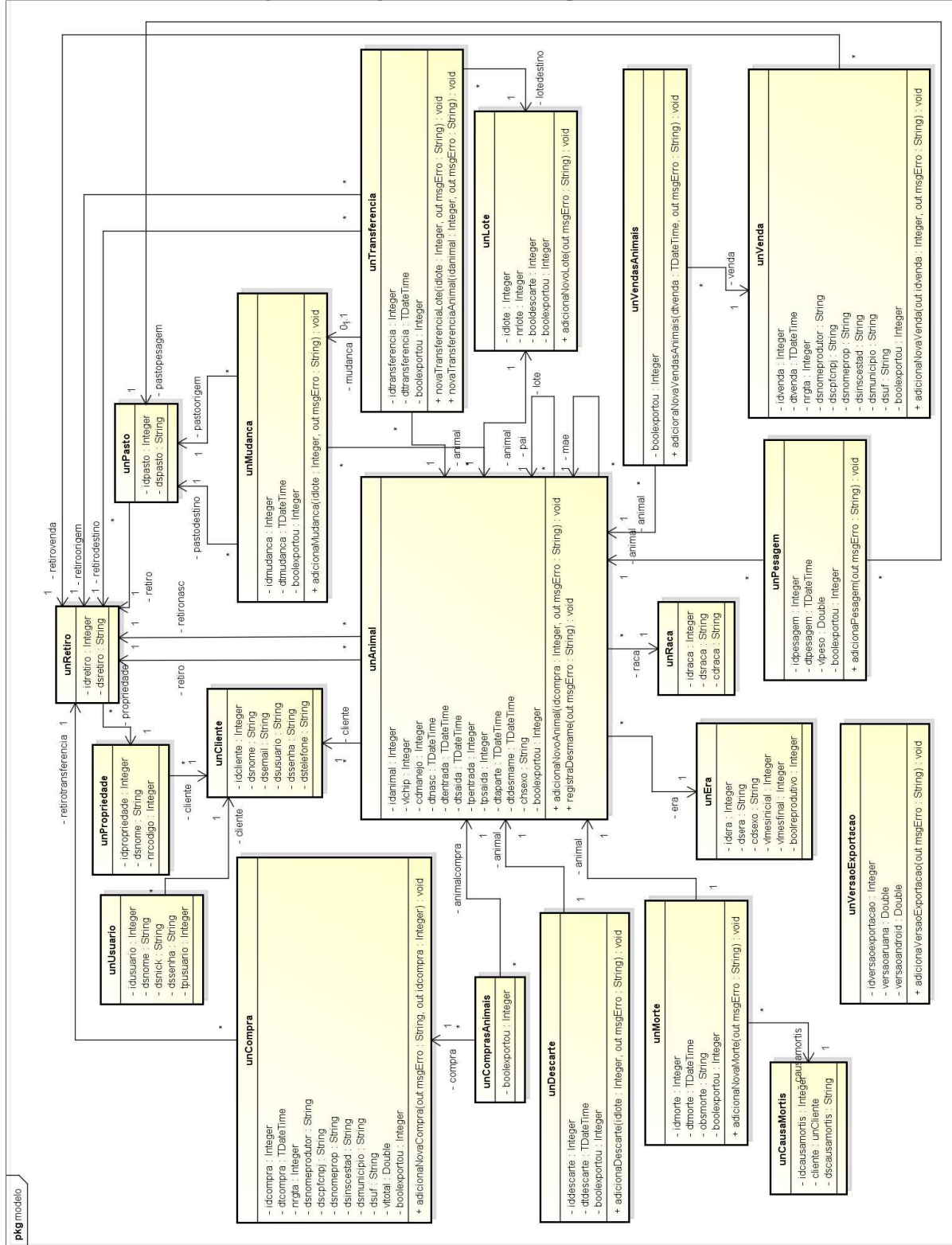
Figura 7 - Diagrama das classes do pacote `db`.



Na Figura 8 são apresentadas as classes que armazenam as informações do aplicativo e, por isso, estão dispostas no pacote `modelo`. Essas classes também contêm métodos utilizados para efetuar a ligação com o pacote `db` e realizar as alterações e inclusões no banco de dados. O aplicativo desenvolvido neste trabalho importa do sistema Aruana-Maleta as informações manipuladas pelas classes dessa camada para que possa funcionar em campo no

modo off-line, incluindo novos dados ou alterando os existentes. Contudo, os dados importados para as classes unCausaMortis, unCliente, unEra, unPasto, unPropriedade, unRaca, unRetiro e unUsuario são apenas para leitura.

Figura 8 - Diagrama de classes do pacote modelo.



3.3 IMPLEMENTAÇÃO

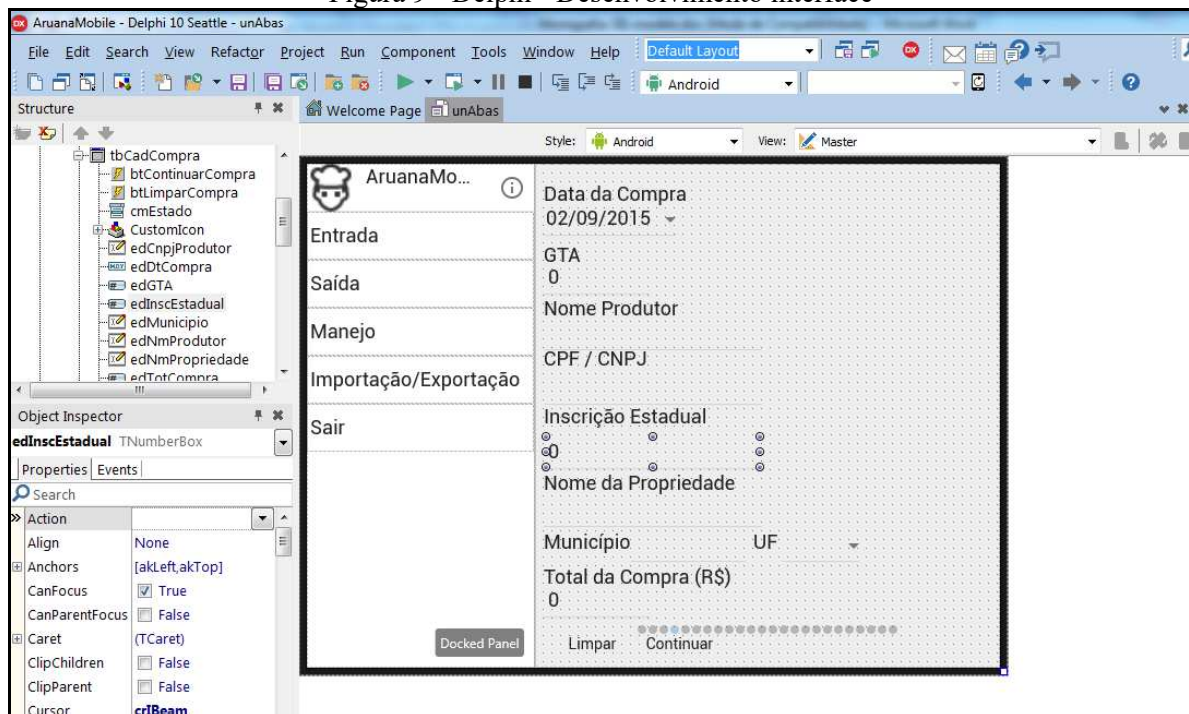
Nesta seção é apresentado o passo-a-passo para a implementação da ferramenta, comentando as técnicas e ferramentas utilizadas e a operacionalidade da implementação.

3.3.1 Técnicas e ferramentas utilizadas

O desenvolvimento do aplicativo foi realizado, inicialmente, com a ferramenta Delphi, versão XE7. Com essa ferramenta o desenvolvimento de aplicativos para dispositivos *mobile* é realizado com a mesma estrutura utilizada no desenvolvimento de aplicativos *desktop*. Contudo, durante o desenvolvimento do projeto, foi necessário migrar para a versão Delphi 10 Seattle devido às modificações nas licenças disponíveis nos servidores da FURB. Apesar disso, essa mudança de versão não causou modificações na forma de desenvolvimento e no código implementado com a versão anterior.

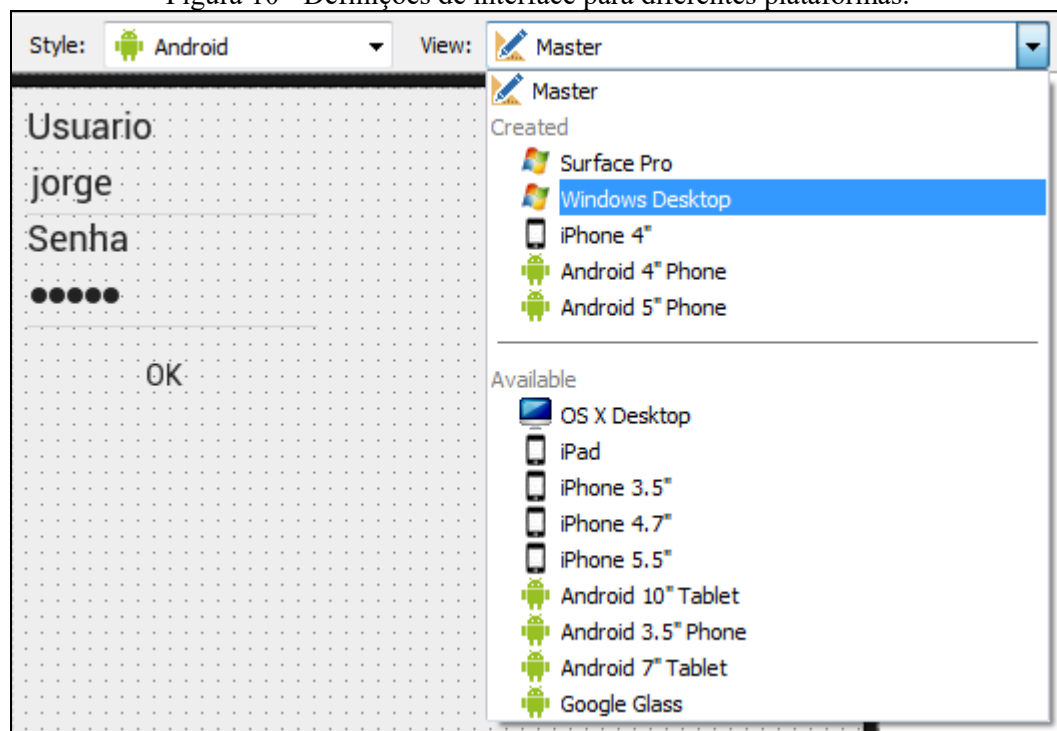
Para o desenvolvimento da interface gráfica, o Delphi disponibiliza um ambiente com componentes visuais, conforme é mostrado na Figura 9. Após definir visualmente a interface do aplicativo, o desenvolvedor necessita apenas implementar o código dos eventos utilizando a linguagem Object Pascal.

Figura 9 - Delphi - Desenvolvimento interface



A ferramenta permite salvar leiautes específicos para cada plataforma disponível como apresentado na Figura 10. Dessa maneira a usabilidade e portabilidade pode ser aprimorada para cada plataforma que se utilize o software desenvolvido. Para cada uma das *views*, pode ser definido desde o tamanho até a fonte utilizada para os componentes da interface.

Figura 10 - Definições de interface para diferentes plataformas.



O Delphi manteve a utilização do Object Pascal como linguagem, mesmo com componentes desenvolvidos para trabalharem nas diferentes plataformas. Contudo, há situações em que as plataformas exigem tratamentos diferenciados. Para resolver esse problema, é possível definir métodos específicos para cada plataforma, como apresentado na Figura 11, em que a linha 245 verifica se a plataforma for iOS ou Android, para em seguida o banco de dados ser criado de acordo com as especificações de cada plataforma.

Figura 11 - Validação da plataforma que está executando

```

- procedure TdmAruana.conBeforeConnect(Sender: TObject);
243 begin
-   {$IF DEFINED(IOS) or DEFINED(ANDROID)}
-   con.Params.Values['Database'] := TPath.GetDocumentsPath + PathDelim +
-     'aruanaMobile.s3db';
-   con.Params.Values['CreateDatabase'] :=
-     BoolToStr(not FileExists(TPath.GetDocumentsPath + PathDelim +
-     'aruanaMobile.s3db'), True);
250 {$ENDIF}
- end;

```

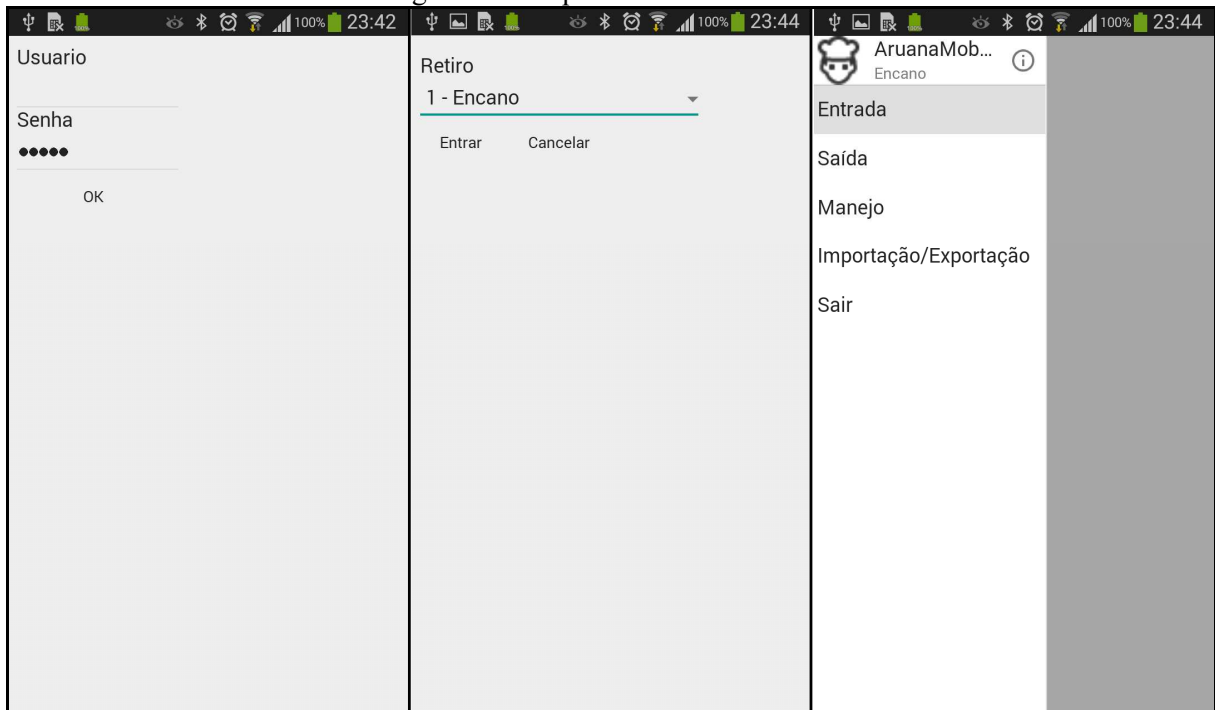
3.3.2 Operacionalidade da implementação

O aplicativo foi desenvolvido com um único formulário e dentro dele painéis que representam todas as telas, por recomendação dos desenvolvedores do LDTT. O motivo dessa recomendação é que os aplicativos desenvolvidos com o Delphi para a plataforma Android apresentam problemas na retirada dos formulários da memória do dispositivo, provocando travamentos no aplicativo.

A implementação do aplicativo seguiu a ordem dos casos de uso. Após a confecção da base de dados baseada nos modelos desenvolvidos, cada um deles foi desenvolvido nesta sequência: a interface com o usuário; a modelagem das classes do pacote modelo; criados os objetos responsáveis pela representação das tabelas e conectando com a conexão da base de dados; e, por último, adicionado os tratamentos de campos e botões de ação da interface.

Na inicialização do aplicativo aparece a tela de *login*. Após essa tela, o usuário deve selecionar o retiro em que estará movimentando os animais e, em seguida, deve utilizar o menu para selecionar qual a ação deseja realizar, a Figura 12 apresenta as três telas em sequência. A funcionalidade do aplicativo se divide em quatro grupos, conforme é apresentado no menu do aplicativo. Esses grupos são: *Entrada*, *Saída* e *Manejo de animais e Importação/Exportação* da base de dados para o Aruana-Maleta.

Figura 12 - Sequência de telas iniciais



No menu *Entrada*, podem ser efetivados os registros de cadastro, compra e nascimento de animais. No cadastro de animais, apresentado na Figura 13, o usuário preenche os dados do animal e clica em salvar para adicionar o animal à base. Nesse caso, existe a possibilidade de adicionar o animal cadastrado a um novo lote, sendo necessário informar o código do novo lote e selecionar o pasto à qual o lote irá pertencer.

Figura 13 - Tela de cadastro de animal

Codigo de Manejo
23

Codigo do SISBOV
45623

Era
1 - Bezerro

Raça
AN - Angus

Data de Nascimento
20/11/15

Lote
321

Seleção de Pasto

Limpar Salvar

O cadastro de compra por sua vez é responsável pelo registro dos animais adquiridos de um fornecedor externo. Esse processo é composto por duas telas, apresentadas na Figura 14. Na primeira tela o usuário fornece os dados da compra, como produtor, Guia de Trânsito Animal (GTA), entre outras. Já na segunda tela, é necessário informar os dados dos animais adquiridos e clicar em salvar para adicionar cada um deles na base de dados. A compra é finalizada quando se clica no botão Concluir.

Figura 14 - Telas para o cadastro de compra

Data da Compra
15/11/15

GTA
532

Nome Produtor
Nelson

CPF / CNPJ
56789987000135

Inscrição Estadual
234

Nome da Propriedade
Fazenda Corações

Município Blumenau UF SC

Total da Compra (R\$)
16523

Limpar Continuar

Codigo de Manejo
0

Codigo do SISBOV
0

Era

Raça

Data de Nascimento
20/11/15

Lote
0

Seleção de Pasto

Limpar Salvar Concluir

Para o registro do nascimento de novos animais, apresentado na Figura 15, o usuário deve informar desde o código de manejo do novo animal até o código de manejo da mãe do animal. Após o preenchimento desses dados, deve-se clicar no botão *Salvar* para concluir o cadastro.

Figura 15 - Cadastro de nascimento

Código do Manejo
0
Código do SISBOV
0
Raça
Sexo do Animal
Peso
Data de Nascimento
20/11/15
Código da Mãe do Animal
0
Limpar Salvar

No menu *Saída*, o usuário pode fazer o registro da morte de um animal, a venda e a transferência de animais para outro retiro. Para o registro de uma morte, o usuário deve informar o código do animal, a data do falecimento, selecionar a *causa mortis* e, se for pertinente, escrever alguma observação, conforme a tela apresentada na Figura 16.

Figura 16 - Registro de morte

Data da Morte
20/11/15
Código do Animal
0
Causa Mortis
Observação
Limpar Salvar

A venda de animal consiste em duas telas, visualizadas na Figura 17. Na primeira tela o usuário informa a data da venda e o lote à qual os animais pertenciam e, após clicar no botão *Continuar*, na segunda tela são informados os animais vendidos. Para concluir a venda o usuário utiliza o botão *Concluir*.

Figura 17 - Venda de animais

The figure displays two side-by-side screenshots of a mobile application interface for animal sales. Both screenshots show a status bar at the top with icons for signal, Wi-Fi, battery (89%), and time (02:04).
 The left screenshot shows a form with the following elements:
 - 'Data da Venda' (Sale Date) with a dropdown menu showing '10/10/15'.
 - 'Lote' (Lot) with a text input field containing '123'.
 - Two buttons at the bottom: 'Limpar' (Clear) and 'Continuar' (Continue).
 The right screenshot shows a form with the following elements:
 - 'Código do Animal' (Animal Code) with a text input field containing '0'.
 - Three buttons at the bottom: 'Limpar' (Clear), 'Continuar' (Continue), and 'Concluir' (Conclude).

A transferência de animais também formada por duas telas, dispostas na Figura 18. Na primeira o usuário deve informar se deseja realizar a transferência por animal ou por lote, bem como a data da transferência e o lote à qual os animais irão pertencer. Caso seja necessário criar um novo lote, deve-se informar o número desse novo lote e à qual pasto ele irá pertencer. Posteriormente, devem ser selecionados os animais ou lotes a serem transferidos. Para finalizar a transferência, o usuário clica em *Concluir*. Após isso, o aplicativo volta para a tela inicial do aplicativo.

Figura 18 - Transferir animais/lotes

The figure displays two side-by-side screenshots of a mobile application interface for animal transfers. Both screenshots show a status bar at the top with icons for signal, Wi-Fi, battery (86%), and time (02:31).
 The left screenshot shows a form with the following elements:
 - 'Transferir por' (Transfer by) with a dropdown menu showing 'Animal'.
 - 'Data da Transferência' (Transfer Date) with a dropdown menu showing '12/10/15'.
 - 'Retiro Destino' (Destination) with a dropdown menu.
 - 'Lote de Destino' (Destination Lot) with a text input field containing '0'.
 - 'Seleção de Pasto' (Pasture Selection) with a dropdown menu.
 - Two buttons at the bottom: 'Limpar' (Clear) and 'Continuar' (Continue).
 The right screenshot shows a form with the following elements:
 - 'Código do Animal' (Animal Code) with a text input field containing '2'.
 - Three buttons at the bottom: 'Limpar' (Clear), 'Continuar' (Continue), and 'Concluir' (Conclude).

O menu de Manejo permite ao usuário realizar o registro da pesagem dos animais, mudança de pasto, desmame e descarte. Para o cadastro de pesagem, o aplicativo atua com duas telas, conforme é mostrado na Figura 19. Na primeira, o usuário informa a data da pesagem e o lote que está sendo pesado. Em seguida, o usuário informa o código do animal e o peso atual do mesmo, utilizando o botão Continuar para o próximo animal. Para finalizar a pesagem, o usuário deve clicar no botão Concluir.

Figura 19 - Pesagem de animais

The image shows two side-by-side screenshots of a mobile application interface for animal weighing. The top status bar shows icons for signal, Wi-Fi, battery, and time (00:39). The left screen displays 'Data de Pesagem' with a dropdown menu showing '22/10/15' and 'Lote' with a text input field containing '0'. Below these are 'Limpar' and 'Continuar' buttons. The right screen displays 'Código do Animal' with a text input field containing '0' and 'Peso' with a text input field. Below these are 'Limpar', 'Continuar', and 'Concluir' buttons.

A mudança de pasto é realizada em uma única tela, apresentada na Figura 20. Nela, o usuário informa qual o lote que será mudado de pasto, para qual pasto o lote será enviado e a data da mudança. O usuário clica no botão Salvar e o sistema muda o pasto do lote selecionado e transfere todos os animais para o pasto selecionado, finalizando a mudança.

Figura 20 - Mudança de Pasto

The image shows a single screenshot of a mobile application interface for changing pastures. The top status bar shows icons for signal, Wi-Fi, battery, and time (01:11). The screen displays 'Lote' with a dropdown menu, 'Pasto de Destino' with a dropdown menu, and 'Data da Mudança' with a dropdown menu showing '22/11/15'. Below these are 'Limpar' and 'Salvar' buttons.

O cadastro de desmame também utiliza a estrutura de duas telas, conforme é apresentado na Figura 21. Na primeira, o usuário informa a data do desmame, o lote à qual os animais pertencem e para qual lote serão enviados. Novamente, caso esteja sendo criado um novo lote para estes animais, o usuário deve selecionar o pasto ao qual esse novo lote irá pertencer. Na segunda tela, o usuário informa o código dos animais, clicando no botão *Continuar* para o próximo animal. Ao clicar no botão *Concluir* é finalizado o desmame.

Figura 21 - Cadastro de desmame

The image shows two side-by-side screenshots of a mobile application interface. The left screenshot displays the 'Cadastro de desmame' form with the following fields: 'Data do Desmame' (02/11/15), 'Lote a Desmamar' (123), 'Lote de Destino' (123), and 'Seleção de Pasto' (dropdown menu). Below the fields are 'Limpar' and 'Continuar' buttons. The right screenshot shows the 'Código do Animal' field with the value '1' and 'Limpar', 'Continuar', and 'Concluir' buttons below it. Both screenshots show a status bar at the top with 94% battery and 11:26 time.

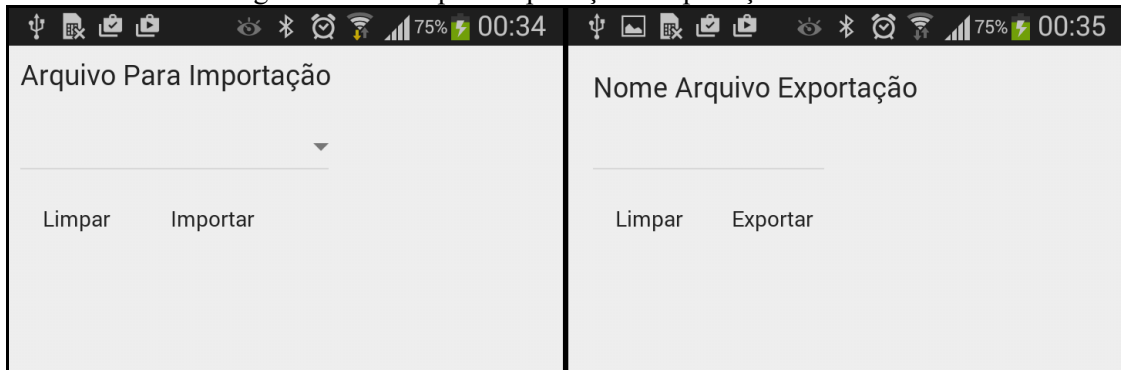
Para o descarte de animais, apresentado na Figura 22, o usuário deve informar primeiramente a data do descarte e o lote de destino, que deverá ser um lote de descarte. Um novo lote de descarte pode ser cadastrado nesse momento, bastando selecionar o pasto ao qual ele irá pertencer. Ao clicar no botão *Continuar*, o usuário é direcionado para a segunda tela, onde deve informar o código dos animais que serão descartados, finalizando o descarte utilizando o botão *Concluir*.

Figura 22 - Descarte de animais

The image shows two side-by-side screenshots of a mobile application interface. The left screenshot displays the 'Descarte de animais' form with the following fields: 'Data do Descarte' (29/10/15), 'Lote de Destino' (123), and 'Seleção de Pasto' (dropdown menu). Below the fields are 'Limpar' and 'Continuar' buttons. The right screenshot shows the 'Código do Animal' field with the value '0' and 'Limpar', 'Continuar', and 'Concluir' buttons below it. Both screenshots show a status bar at the top with 95% battery and 11:30 time.

A Figura 23 mostra as telas para importação e exportação de dados do aplicativo para o sistema Aruana-Maleta. Para realizar a importação, o usuário seleciona o arquivo desejado que se encontra dentro da pasta padrão do sistema. Já para realizar a exportação, o usuário deve informar o nome de arquivo e clicar na opção `Exportar`. Além do nome informado pelo usuário, o arquivo de exportação terá a data da exportação como prefixo. Esse arquivo também é armazenado na pasta padrão do sistema, assim como os arquivos importados.

Figura 23 - Telas para importação e exportação de dados.



3.4 RESULTADOS E DISCUSSÕES

O presente trabalho apresentou o desenvolvimento de uma nova versão *mobile* do sistema Aruana-Maleta. Isso facilita o trabalho dos pecuaristas, uma vez que podem realizar o cadastro dos animais em campo com um dispositivo mais fácil de carregar do que o um computador. Dessa forma, o primeiro objetivo específico do trabalho foi parcialmente alcançado.

Diferentemente do trabalho desenvolvido por Felipe Demarchi (2012), cujo aplicativo funciona apenas na plataforma Android, este trabalho foi desenvolvido no ambiente Delphi 10 Seattle, que é capaz de gerar versões do aplicativo para diferentes plataformas *mobile*. Contudo, apesar da geração ser garantida pelo ambiente, não foi possível executar a geração das versões do aplicativo para as plataformas Windows Phone e iOS, devido a restrições de licença exigidas por essas plataformas. Dessa forma, o segundo objetivo específico do trabalho foi parcialmente alcançado.

Quando o projeto deste trabalho foi proposto, estava prevista a ativação de um servidor com os dados do sistema Aruana-Maleta nas dependências da FURB. Esse servidor iria possibilitar a realização de testes do aplicativo desenvolvido com dados importados do sistema. Porém, durante o período de desenvolvimento do aplicativo, o projeto Aruana foi desvinculado do LDTT e o Prof. Mauro Marcelo Mattos, responsável pelo desenvolvimento da ferramenta original, perdeu acesso ao mesmo. Esse imprevisto impossibilitou a realização

de testes comparativos entre o aplicativo desenvolvido e o sistema Aruana-Maleta, conforme era previsto no terceiro objetivo específico do trabalho. Para possibilitar uma futura integração concreta com o sistema Aruana-Maleta, foi implementada no aplicativo a funcionalidade de exportação e importação de dados utilizando arquivos XML.

Em relação aos trabalhos correlatos o sistema tem uma principal vantagem que seria um código gerado que possibilita a compilação para múltiplas plataformas, conforme Quadro 3, que apresenta as funcionalidades dos sistemas correlatos e do projeto desenvolvido. Todos os três softwares apresentam portabilidade e fazem o gerenciamento dos animais, porém o sistema A3 não apresenta a possibilidade de trabalho sem conexão com a Internet para a versão portátil, que seria a sua versão *web*. Uma desvantagem do projeto proposto neste trabalho é o fato de que ele contempla somente a gestão dos animais, já o sistema A3 apresenta também módulo para a gestão financeira do negócio.

	Projeto	Aruana Android	A3
Plataforma	Multiplataforma	Android	WEB/Windows
Gestão de Animais	X	X	X
Portabilidade	X	X	X
Mobilidade Off-line	X	X	
Gestão de Financeira			X

Quadro 3 - Comparativo entre o projeto proposto e seus correlatos.

Recursos/Softwares	Projeto	Aruana Android	A3
Plataforma	Multiplataforma	Android	WEB/Windows
Gestão de Animais	X	X	X
Portabilidade	X	X	X
Mobilidade Off-line	X	X	
Gestão de Financeira			X

4 CONCLUSÕES

Este trabalho realizou a implementação das principais funções do sistema Aruana-Maleta para um aplicativo desenvolvido no ambiente Delphi 10 Seattle. Este trabalho resulta em uma extensão do trabalho de Demarchi (2012), que desenvolveu um aplicativo nos mesmo moldes, porém apenas para a plataforma Android.

O aplicativo desenvolvido manteve as principais funções do sistema Aruana-Maleta, como cadastro de animais, registro de nascimento de animais, morte, desmame, descarte, venda, compra, mudança de pastos e transferência. Para possibilitar uma integração futura do aplicativo com o sistema Aruana-Maleta foi mantida as funcionalidades de importação e exportação de dados, porém utilizando arquivos XML.

A ferramenta escolhida para o desenvolvimento, o Delphi 10 Seattle, trouxe praticidade para o desenvolvimento da interface, como já era para o ambiente Windows, porém, com limitações para as personalizações. Os componentes nativos apresentaram propriedades e eventos importantes para o desenvolvimento do aplicativo, tornando possível o desenvolvimento do aplicativo num curto período de tempo.

O ambiente Delphi 10 Seattle fornece recursos para o desenvolvimento de aplicativos para múltiplas plataformas, porém, a execução de testes e configuração da interface para as variadas plataformas que deve ser configurada de forma manual, conforme foi descrito na Seção 3.3.1. Um problema detectado é que, apesar do ambiente oferecer os recursos para desenvolvimento multiplataforma, restrições de licenciamento impostas por essas plataformas podem impedir que isso seja feito. Por exemplo, a geração da versão para a plataforma iOS é possível somente com a obtenção de uma licença da Apple Store e com o desenvolvimento sendo realizado em uma máquina da Apple.

4.1 EXTENSÕES

Como sugestões para trabalhos futuros:

- a) realizar a integração com a versão original do sistema Aruana-Maleta por meio de uma estrutura cliente-servidor, mantendo a possibilidade de trabalho *off-line*;
- b) realizar a geração das versões do aplicativo para as plataformas Windows Phone e iOS;
- c) implementação de um módulo para registro de informações financeiras e estoque de materiais.

REFERÊNCIAS

- A3 PECUÁRIA. **Software para gestão pecuária**. Porto Alegre. 2015. Disponível em: <<http://www.a3pecuaria.com.br/site/website/home/produto.htm>>. Acesso em 06 abril. 2015.
- BRASIL. Ministério da Agricultura, Pecuária e Abastecimento. **Projeções do Agronegócio: Brasil 2012/2013 a 2022/2023** / Ministério da Agricultura, Pecuária e Abastecimento. Assessoria de Gestão Estratégica. Brasília: Mapa/ACS, 2013. 96 p.
- CORÁ, Maria Fernanda Goulart Gutheil. **Rastreabilidade Bovina: Implantar ou não?** 2009. 21 f. Trabalho de Conclusão de Curso de Especialização (Especialização em Gestão Empresarial) – Universidade Federal do Rio Grande do Sul - Porto Alegre. Disponível em: <<https://www.lume.ufrgs.br/bitstream/handle/10183/26769/000748731.pdf?sequence=1>>. Acesso em: 25 março. 2015.
- DEITEL, H.; DEITEL, P.; DEITEL, A.; MORGANO, M. **Android para programadores: Uma abordagem baseada em aplicativos**. Bookman, 2013.
- DEMARCHI, Felipe. **Migração do Sistema Aruana-Maleta para a Plataforma Android**. 2012. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- EHLERT, Eduardo. **Módulo de avaliação do retorno de investimento em sistema de rastreabilidade animal**. 2009. 60 f. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- EMBARCADERO. **O que há de novo no RAD Studio 10 Seattle?** [S.1.], 2015. Disponível em: <<https://www.embarcadero.com/br/products/rad-studio/whats-new>>. Acesso em: 05 nov. 2015.
- IDC Brasil. **Estudo da IDC Brasil mostra recorde nas vendas de smartphones no terceiro trimestre de 2014**. São Paulo. 2014. Disponível em: <<http://br.idclatin.com/releases/news.aspx?id=1777>>. Acesso em: 20 mar. 2015.
- KAWATA, Fabricio Hissao. **Desenvolvimento Android no Delphi**. [S.1.], 2013. Disponível em: <<http://www.devmedia.com.br/desenvolvimento-android-no-delphi/29711>>. Acesso em: 01 abr. 2015.
- KAWATA, Fabricio Hissao. **Novidades do Delphi XE7**. [S.1.], 2015. Disponível em: <<http://www.devmedia.com.br/novidades-do-delphi-xe7/31702>>. Acesso em: 03 abr. 2015.
- MATTOS, Mauro. M. **Sistema de rastreabilidade bovina**. [11 de março, 2015]. Blumenau. Entrevista concedida a Jorge Alexandre Hwizdalech Filho.
- MENDES, Cássia I. C.; OLIVEIRA, Deise R. M. dos S.; SANTOS, Anderson R. **Estudo do mercado brasileiro de software para o agronegócio**. 1. ed. Campinas, São Paulo: Embrapa Informática Agropecuária, 2011. 184 p.
- MENDES, Ricardo E. O impacto financeiro da rastreabilidade em sistemas de produção de bovinos no estado de Santa Catarina, Brasil. **Ciência Rural**, Santa Maria, v. 36, n. 5, out. 2006. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-84782006000500028>. Acesso em: 18 mar. 2015.
- SERPRO. **SISBOV – Qualidade desde a Origem**. Brasil. [S.1.], 2008. Disponível em: <<http://www4.serpro.gov.br/imprensa/publicacoes/tema-1/antigas%20temas/tema-195/materias/sisbov/>>. Acessado em: 20 abr. 2015.

APÊNDICE A – Descrição dos Casos de Uso

Nesta seção da proposta apresenta a descrição dos casos de uso conforme previstos no diagrama apresentado na seção “3.2.1 Casos de Uso”.

No Quadro 4, apresenta-se o caso de uso *Efetuar Login*.

Quadro 4 - Descrição do Caso de Uso 01 - *Efetuar login*

UC01	<i>Efetuar Login</i>
Descrição	Este caso de uso tem por objetivo descrever os procedimentos necessários para a realização do <i>login</i> do usuário.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando.
Pós-condições	Usuário logado no sistema.
Cenário principal	1. O Usuário digita o seu nome de usuário; 2. O Usuário digita a sua senha de acesso; 3. O Usuário submete as informações pressionando o botão OK; 4. O aplicativo realiza o <i>login</i> do Usuário.
Exceção 01	No passo 3, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 3, se o aplicativo verificar que o nome de usuário e a senha não conferem com a base de dados, é apresentada uma mensagem informando que o nome de usuário e senha não conferem.

No Quadro 5, apresenta-se o caso de uso *Selecionar Retiro*.

Quadro 5 - Descrição do Caso de Uso 02 - *Selecionar Retiro*

UC02	<i>Selecionar Retiro</i>
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização da seleção do retiro que será administrado.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já realizado.
Pós-condições	Retiro a ser administrado selecionado.
Cenário principal	1. O Usuário seleciona o retiro que deseja administrar; 2. O Usuário submete a informação ao aplicativo pressionando o botão <i>Entrar</i> ; 3. O aplicativo atribui o retiro selecionado ao usuário; 4. O aplicativo exibe a tela contendo o menu da aplicação.

No Quadro 6, apresenta-se o caso de uso *Cadastrar Animal*.

Quadro 6 - Descrição do Caso de Uso 03 - Cadastrar Animal

UC03	Cadastrar Animal
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de um novo animal no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já executado.
Pós-condições	Cadastro de um novo animal na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Entrada; 3. O Usuário seleciona a opção Cadastrar Animal; 4. O aplicativo exibe o formulário para cadastro de um novo animal; 5. O Usuário informa o código de manejo; 6. O Usuário informa o código do SISBOV; 7. O Usuário seleciona a era do animal; 8. O Usuário seleciona a raça do animal; 9. O Usuário informa a data de nascimento do animal; 10. O Usuário informa o número do lote que irá pertencer ao animal; 11. O Usuário submete as informações ao aplicativo pressionando o botão Salvar; 12. O aplicativo realiza a inclusão do novo animal no banco de dados.
Fluxo alternativo	No passo 11, se o lote não existir, o Usuário deverá selecionar um pasto para o novo lote.
Exceção 01	No passo 11, se o aplicativo identificar que algum campo não foi preenchido, é apresentado uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 11, se o aplicativo verificar que o código de manejo já está cadastrado na base de dados, é apresentado uma mensagem informando que o código de manejo já está cadastrado.
Exceção 03	No passo 11, se o aplicativo verificar que a data de nascimento é posterior a data atual, é apresentada uma mensagem informando que a data de nascimento deve ser inferior ou igual a data atual.

No Quadro 7, apresenta-se o caso de uso Cadastrar Compra.

Quadro 7 - Descrição do Caso de Uso 04 - Cadastrar Compra

UC04	Cadastrar Compra
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de uma nova compra no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já executado.
Pós-condições	Cadastro de uma nova compra na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Entradas; 3. O Usuário seleciona a opção Comprar Animal; 4. O aplicativo exibe o formulário para cadastro de compra; 5. O Usuário informa a data da compra; 6. O Usuário informa o número do GTA; 7. O Usuário informa o nome do produtor; 8. O Usuário informa o CPF/CNPJ do produtor; 9. O Usuário informa a inscrição estadual do produtor; 10. O Usuário informa o nome da propriedade; 11. O Usuário informa o nome do município; 12. O Usuário informa o estado ao qual o município pertence; 13. O Usuário informa o valor total da compra; 14. O Usuário submete as informações ao aplicativo pressionando Continuar; 15. O aplicativo apresenta a tela para cadastro de animais que serão comprados. 16. O Usuário informa o código de manejo; 17. O Usuário informa o código do SISBOV; 18. O Usuário seleciona a era do animal; 19. O Usuário seleciona a raça do animal; 20. O Usuário informa a data de nascimento do animal; 21. O Usuário informa o número do lote que irá pertencer ao animal; 22. O Usuário submete as informações ao aplicativo pressionando Concluir; 23. O aplicativo realiza a inclusão da nova compra no banco de dados;
Fluxo alternativo	No passo 22, se o Usuário desejar adicionar mais animais para a compra, deverá pressionar o botão Continuar, que irá adicionar o animal para a compra e retornará ao passo 15.
Fluxo alternativo	No passo 22, na havendo lote, o Usuário seleciona um pasto para o novo lote.
Exceção 01	No passo 14, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 14, se o aplicativo identificar que a data da compra é posterior a data atual, é apresentada uma mensagem informando que a data da compra deve ser inferior ou igual a data atual.
Exceção 03	No passo 22, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 04	No passo 22, se o aplicativo verificar que o código de manejo não está cadastrado na base de dados, é apresentada uma mensagem informando que o código de manejo não existe.
Exceção 05	No passo 22, se o aplicativo verificar que a data de nascimento é posterior a data atual, é apresentada uma mensagem informando que a data de nascimento deve ser inferior ou igual a data atual.

No Quadro 8, apresenta-se o caso de uso Cadastrar Nascimento.

Quadro 8 - Descrição do Caso de Uso 05 - Cadastrar Nascimento

UC05	Cadastrar Nascimento
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de um nascimento no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando, com o <i>login</i> já executado e possuir um animal que seja da era denominada Vaca.
Pós-condições	Cadastro de nascimento de animal na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Entradas; 3. O Usuário seleciona a opção Cadastrar Nascimento; 4. O aplicativo exibe o formulário para cadastro de um nascimento; 5. O Usuário informa o código de manejo; 6. O Usuário informa o código do SISBOV; 7. O Usuário seleciona a raça do animal; 8. O Usuário seleciona o sexo do animal; 9. O Usuário informa o peso do animal; 10. O Usuário informa a data de nascimento do animal; 11. O Usuário informa o código da mãe do animal; 12. O Usuário submete as informações ao aplicativo pressionando Salvar; 13. O aplicativo realiza a inclusão do nascimento no banco de dados;
Exceção 01	No passo 12, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 12, se o aplicativo verificar que o código de manejo já está cadastrado na base de dados, é apresentada uma mensagem informando que o código de manejo já está cadastrado.
Exceção 03	No passo 12, se o aplicativo verificar que a data de nascimento é posterior a data atual, é apresentada uma mensagem informando que a data de nascimento deve ser inferior ou igual à data atual.
Exceção 04	No passo 12, se o aplicativo verificar que o código informado para a mãe do animal não está cadastrado na base de dados, é apresentada uma mensagem informando que o código da mãe não existe.

No Quadro 9, apresenta-se o caso de uso Cadastrar Morte.

Quadro 9 - Descrição do Caso de Uso 06 - Cadastrar Morte

UC06	Cadastrar Morte
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de uma morte de animal no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando, com o <i>login</i> já executado e com pelo menos um animal já cadastrado no sistema.
Pós-condições	Cadastro de uma morte de animal na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Saída; 3. O Usuário seleciona a opção Morte de Animal; 4. O aplicativo exibe o formulário para cadastro de uma nova morte; 5. O Usuário informa a data da morte; 6. O Usuário informa o código do animal; 7. O Usuário seleciona a causa da morte; 8. O Usuário informa uma observação sobre a morte do animal; 9. O Usuário submete as informações ao aplicativo pressionando o botão Salvar; 10. O aplicativo realiza a inclusão da nova morte de animal no banco de dados;
Exceção 01	No passo 9, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 9, se o aplicativo verificar que a data da morte é posterior a data atual, é apresentada uma mensagem informando que a data da morte deve ser inferior ou igual a data atual.
Exceção 03	No passo 9, se o aplicativo verificar que o código do animal não está cadastrado na base de dados, é apresentada uma mensagem informando que o código do animal é inexistente.

No Quadro 10, apresenta-se o caso de uso Cadastrar Venda.

Quadro 10 - Descrição do Caso de Uso 07 - Cadastrar Venda

UC07	Cadastrar Venda
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de uma nova venda no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando, com o <i>login</i> já executado e possuir pelo menos um animal cadastrado na base de dados.
Pós-condições	Cadastro de uma nova venda na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Saída; 3. O Usuário seleciona a opção Vender Animal; 4. O aplicativo exibe o formulário para cadastro de venda; 5. O Usuário informa a data da venda; 6. O Usuário informa o número do lote de onde os animais serão vendidos; 7. O Usuário submete as informações ao aplicativo pressionando o botão Continuar; 8. O aplicativo continua apresentando a tela para seleção dos animais que serão vendidos. 9. O Usuário informa o código do animal; 10. O Usuário submete as informações ao aplicativo pressionando o botão Concluir; 11. O aplicativo realiza a inclusão da nova compra no banco de dados;
Fluxo alternativo	No passo 10, se o Usuário desejar selecionar mais animais para a venda, deverá pressionar o botão Continuar, que irá adicionar o animal para a venda e retornará ao passo 8.
Exceção 01	No passo 7, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 7, se o aplicativo identificar que a data da venda é posterior a data atual, é apresentada uma mensagem informando que a data da venda deve ser inferior ou igual a data atual.
Exceção 03	No passo 7, se o aplicativo identificar que o lote é inexistente, é apresentada uma mensagem informando que o lote inexistente.
Exceção 04	No passo 10, se o aplicativo identificar que o código do animal não foi preenchido, é apresentada uma mensagem informando que o código do animal é obrigatório.
Exceção 05	No passo 10, se o aplicativo verificar que o código do animal não está cadastrado na base de dados, é apresentada uma mensagem informando que o código do animal não existe.

No Quadro 11, apresenta-se o caso de uso Cadastrar Transferência.

Quadro 11 - Descrição do Caso de Uso 08 - Cadastrar Transferência

UC08	Cadastrar Transferência
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de uma nova transferência no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando, com o <i>login</i> já executado e possuir pelo menos um animal cadastrado na base de dados.
Pós-condições	Cadastro de uma nova transferência na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Saída; 3. O Usuário seleciona a opção Transferir Animal; 4. O aplicativo exibe o formulário para cadastro de transferência; 5. O Usuário seleciona o tipo de transferência; 6. O Usuário informa a data da transferência; 7. O Usuário seleciona o retiro de destino; 8. O Usuário informa o lote de destino; 9. O Usuário submete as informações ao aplicativo pressionando o botão Continuar; 10. O aplicativo continua apresentando a tela para seleção dos animais ou lotes que serão transferidos. 11. O Usuário informa o código do animal ou o número do lote; 12. O Usuário submete as informações ao aplicativo clicando Concluir; 13. O aplicativo realiza a inclusão da nova transferência no banco de dados;
Fluxo alternativo	No passo 9, se o lote não existir, o Usuário deverá selecionar um pasto para o novo lote.
Fluxo alternativo	No passo 12, se o Usuário desejar selecionar mais animais ou lotes para a transferência, deverá pressionar o botão Continuar, que irá adicionar o animal ou lote para a transferência e retornará ao passo 10.
Exceção 01	No passo 9, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 9, se o aplicativo identificar que a data da transferência é posterior a data atual, é apresentada uma mensagem informando que a data da transferência deve ser inferior ou igual a data atual.
Exceção 03	No passo 12, se o aplicativo identificar que o código do animal ou o número do lote não foi preenchido, é apresentado uma mensagem informando que o campo é obrigatório.
Exceção 04	No passo 12, se o aplicativo verificar que o código do animal ou o número do lote não está cadastrado na base de dados, é apresentada uma mensagem informando que o código não existe.

No Quadro 12, apresenta-se o caso de uso Cadastrar Pesagem.

Quadro 12 - Descrição do Caso de Uso 09 - Cadastrar Pesagem

UC09	Cadastrar Pesagem
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de uma nova pesagem no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando, com o <i>login</i> já executado e possuir pelo menos um animal cadastrado na base de dados.
Pós-condições	Cadastro de uma nova pesagem na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Manejo; 3. O Usuário seleciona a opção Pesagem; 4. O aplicativo exibe o formulário para cadastro de pesagem; 5. O Usuário informa a data da pesagem; 6. O Usuário informa o lote da pesagem; 7. O Usuário submete as informações ao aplicativo clicando Continuar; 8. O aplicativo continua apresentando a tela para seleção dos animais que serão pesados. 9. O Usuário informa o código do animal; 10. O Usuário informa o valor do peso do animal; 11. O Usuário submete as informações ao aplicativo clicando Concluir; 12. O aplicativo realiza a inclusão da nova pesagem no banco de dados.
Fluxo alternativo	No passo 11, se o Usuário desejar realizar mais pesagens, deverá clicar no botão Continuar, que irá adicionar a pesagem e retornará ao passo 8.
Exceção 01	No passo 7, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 7, se o aplicativo identificar que a data da pesagem é posterior a data atual, é apresentada uma mensagem informando que a data da pesagem deve ser inferior ou igual a data atual.
Exceção 03	No passo 7, se o aplicativo identificar que o número do lote é inexistente, é apresentada uma mensagem informando que o lote é inexistente.
Exceção 04	No passo 11, se o aplicativo identificar que o código do animal não foi preenchido, é apresentada uma mensagem informando que o código do animal é obrigatório.
Exceção 05	No passo 11, se o aplicativo verificar que o código do animal não está cadastrado na base de dados, é apresentada uma mensagem informando que o código do animal não existe.
Exceção 06	No passo 11, se o aplicativo verificar que o valor digitado para o peso do animal é inválido, é apresentada uma mensagem informando que o valor do peso é inválido.

No Quadro 13, apresenta-se o caso de uso Cadastrar Mudança de Pasto.

Quadro 13 - Descrição do Caso de Uso 10 - Cadastrar Mudança de Pasto

UC10	Cadastrar Mudança de Pasto
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização do cadastro de uma mudança de pasto no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já executado.
Pós-condições	Cadastro de uma nova venda na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Manejo; 3. O Usuário seleciona a opção Mudar Pasto; 4. O aplicativo exibe o formulário para realizar a mudança de pasto; 5. O Usuário seleciona o lote que sofrerá a mudança de pasto; 6. O Usuário seleciona o pasto para o qual o lote será transferido; 7. O Usuário informa a data da mudança; 8. O Usuário submete as informações ao aplicativo pressionando o botão Salvar; 9. O aplicativo realiza a mudança de pasto no banco de dados;
Exceção 01	No passo 8, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 8, se o aplicativo identificar que a data da mudança é posterior a data atual, é apresentada uma mensagem informando que a data da mudança deve ser inferior ou igual a data atual.

No Quadro 14, apresenta-se o caso de uso Cadastrar Desmame.

Quadro 14 - Descrição do Caso de Uso 11 - Cadastrar Desmame

UC11	Cadastrar Desmame
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização de cadastro de desmame no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando, com o <i>login</i> já executado e possuir animais disponíveis para o desmame.
Pós-condições	Cadastro de um desmame na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Manejo; 3. O Usuário seleciona a opção Desmame; 4. O aplicativo exibe o formulário para cadastro de desmame; 5. O Usuário informa a data de desmame; 6. O Usuário informa o lote a ser desmamado; 7. O Usuário informa o lote de destino; 8. O Usuário submete as informações ao aplicativo pressionando o botão Continuar; 9. O aplicativo continua apresentando a tela para seleção dos animais que serão desmamados. 10. O Usuário informa o código do animal; 11. O Usuário submete as informações ao aplicativo pressionando o botão Concluir; 12. O aplicativo realiza a inclusão do desmame no banco de dados.
Fluxo alternativo	No passo 8, se o lote não existir, o Usuário deverá selecionar um pasto para o novo lote.
Fluxo alternativo	No passo 11, se o Usuário desejar realizar mais desmames, deverá pressionar o botão Continuar, que irá adicionar o desmame e retornará ao passo 9.
Exceção 01	No passo 8, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 8, se o aplicativo identificar que a data do desmame é posterior a data atual, é apresentada uma mensagem informando que a data do desmame deve ser inferior ou igual a data atual.
Exceção 03	No passo 8, se o aplicativo identificar que o número do lote a desmamar é inexistente, é apresentada uma mensagem informando que o lote é inexistente.
Exceção 04	No passo 8, se o aplicativo identificar que o lote a desmamar não possui animais para o desmame, é apresentada uma mensagem informando que o lote não possui animais para o desmame.
Exceção 05	No passo 11, se o aplicativo identificar que o código do animal não foi preenchido, é apresentado uma mensagem informando que o código do animal é obrigatório.
Exceção 06	No passo 11, se o aplicativo verificar que o código do animal não está cadastrado na base de dados, é apresentado uma mensagem informando que o código do animal não existe.

No Quadro 15, apresenta-se o caso de uso Cadastrar Descarte.

Quadro 15 - Descrição do Caso de Uso 12 - Cadastrar Descarte

UC12	Cadastrar Descarte
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização de cadastro de descarte no sistema.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já executado.
Pós-condições	Cadastro de um descarte na base de dados.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Manejos; 3. O Usuário seleciona a opção Descarte; 4. O aplicativo exibe o formulário para cadastro de descarte; 5. O Usuário informa a data de descarte; 6. O Usuário informa o lote de destino; 7. O Usuário submete as informações ao aplicativo pressionando o botão Continuar; 8. O aplicativo continua apresentando a tela para seleção dos animais que serão descartados. 9. O Usuário informa o código do animal; 10. O Usuário submete as informações ao aplicativo pressionando o botão Concluir; 11. O aplicativo realiza a inclusão do descarte no banco de dados;
Fluxo alternativo	No passo 7, se o lote não existir, o Usuário deverá selecionar um pasto para o novo lote.
Fluxo alternativo	No passo 10, se o Usuário desejar realizar mais descartes, deverá pressionar o botão Continuar, que irá adicionar o descarte e retornará ao passo 8.
Exceção 01	No passo 7, se o aplicativo identificar que algum campo não foi preenchido, é apresentada uma mensagem informando que determinado campo é obrigatório.
Exceção 02	No passo 7, se o aplicativo identificar que a data do descarte é posterior a data atual, é apresentada uma mensagem informando que a data do descarte deve ser inferior ou igual a data atual.
Exceção 03	No passo 7, se o aplicativo identificar que o lote de destino não é de descarte, é apresentada uma mensagem informando que o lote de destino não está marcado como descarte.
Exceção 04	No passo 10, se o aplicativo identificar que o código do animal não foi preenchido, é apresentada uma mensagem informando que o código do animal é obrigatório.
Exceção 05	No passo 10, se o aplicativo verificar que o código do animal não está cadastrado na base de dados, é apresentada uma mensagem informando que o código do animal não existe.

No Quadro 16, apresenta-se o caso de uso Realizar Importação.

Quadro 16 - Descrição do Caso de Uso 13 - Realizar Importação

UC13	Realizar Importação
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização da importação de dados para o aplicativo mobile.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já executado.
Pós-condições	Importar os dados para o aplicativo mobile.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Importação/Exportação; 3. O Usuário seleciona a opção Importar; 4. O aplicativo exibe a tela para realização da importação; 5. O Usuário seleciona o arquivo disponível na pasta do aplicativo para importação; 6. O Usuário submete as informações ao aplicativo pressionando o botão Importar; 8. O aplicativo realiza a importação dos dados para a base de dados do aplicativo mobile;

No Quadro 17, apresenta-se o caso de uso Realizar Exportação.

Quadro 17 - Descrição do Caso de Uso 14 - Realizar Exportação

UC14	Realizar Exportação
Descrição	Este caso de uso tem por objetivo descrever os procedimentos para a realização da exportação de dados do sistema mobile para um arquivo XML.
Ator	Usuário
Pré-condições	Estar com o aplicativo Aruana executando e com o <i>login</i> já executado.
Pós-condições	Exportar os dados para o arquivo XML.
Cenário principal	<ol style="list-style-type: none"> 1. O Usuário acessa o menu do aplicativo; 2. O Usuário seleciona a opção Importação/Exportação; 3. O Usuário seleciona a opção Exportar; 4. O aplicativo exibe a tela para realização da exportação; 5. O Usuário informa o nome do arquivo, que será precedido da data da exportação; 6. O Usuário submete as informações ao aplicativo pressionando o botão Exportar; 7. O aplicativo gera o arquivo XML com os dados da base de dados do aplicativo. 8. O aplicativo salva o arquivo XML no diretório do aplicativo.

APÊNDICE B – Dicionário de Dados

No

Quadro 18 - Dicionário de dados do aplicativo.

é apresentado o dicionário de dados do aplicativo.

Quadro 18 - Dicionário de dados do aplicativo.

Nome do Campo	Tipo de Dados	Descrição
booldescarte	INTEGER	Identifica se o lote é de animais descartados (0 - Não; 1 - Sim)
boolexportou	TINYINT	Identifica se o registro foi exportado
boolreprodutivo	CHAR(1)	Identificar se a era representa um animal reprodutivo (0 - Não; 1 - Sim)
cdmanejo	VARCHAR	Código de manejo atribuído ao animal pelo proprietário
cdraca	CHAR(2)	Código de identificação da raça do animal
cdsexo	CHAR(1)	Sexo do animal (M - Masculino; F - Feminino)
chsexo	CHAR(1)	Sexo do animal (M - Masculino; F - Feminino)
dscausamortis	VARCHAR	Descrição da Causa Mortis
dscpfcnpj	VARCHAR	CPF/CNPJ do produtor dos animais comprados
dsemail	VARCHAR	Email do cliente
dsera	VARCHAR	Descrição da era do animal (fase da vida)
dsinscstad	VARCHAR	Inscrição estadual do produtor dos animais comprados
dsmunicipio	VARCHAR	Município de origem dos animais comprados
dsnck	VARCHAR	Identificador de login do usuário
dsnome	VARCHAR	Nome do cliente
dsnome (propriedades)	VARCHAR	Nome da propriedade
dsnome (usuarios)	VARCHAR	Nome do usuário
dsnomeprodutor	VARCHAR	Nome do produtor dos animais comprados
dsnomeprop	VARCHAR	Nome do proprietário dos animais comprados
dspasto	VARCHAR	Descrição do pasto
dsraca	VARCHAR	Descrição da raça do animal
dsretiro	VARCHAR	Descrição do retiro
dssenha	VARCHAR	Senha principal do cliente
dssenha (usuarios)	VARCHAR	Senha do usuário
dstelefone	VARCHAR	Telefone do cliente
dsuf	CHAR(2)	UF do estado de origem dos animais comprados
dsusuario	VARCHAR	Usuário principal do cliente
dtaparte	VARCHAR	Data do parto do animal
dtcompra	VARCHAR	Data da compra
dtdescarte	VARCHAR	Data da realização do descarte
dtdesmame	VARCHAR	Data do desmame do animal
dtentrada	VARCHAR	Data de entrada do animal ao rebanho
dtmorte	VARCHAR	Data da morte do animal
dtmudanca	VARCHAR	Data da mudança de pasto
dtnasc	VARCHAR	Data de nascimento
dtpesagem	VARCHAR	Data da pesagem do animal
dtsaida	VARCHAR	Data em que o animal sai do rebanho
dttransferencia	VARCHAR	Data da transferência do animal
idanimal	INTEGER	ID do animal na base
idcausamortis	INTEGER	ID da Causa Mortis na base
idcliente	INTEGER	ID do cliente na base

idcompra	VARCHAR	ID da compra na base
iddescarte	INTEGER	ID do descarte na base
idera	INTEGER	ID da era na base
idlote	INTEGER	ID do lote na base
idmorte	INTEGER	ID do registro de morte na base
idmudanca	INTEGER	ID da mudança de pasto na base
idpasto	INTEGER	ID do pasto na base
idpesagem	INTEGER	ID da pesagem no sistema
idpropriedade	INTEGER	ID da propriedade na base
idraca	INTEGER	ID da raça na base
idretiro	INTEGER	ID do retiro na base
idtransferencia	INTEGER	ID da transferência na base
idusuario	INTEGER	ID do usuário na base
idvenda	INTEGER	ID da venda do animal na base
idversaoexportacao	INTEGER	ID da versão de exportação na base
nrcodigo	INTEGER	Código da propriedade presente na mesma (identificador)
nrgta	VARCHAR	Número da Guia de Trânsito Animal (GTA)
nrlote	INTEGER	Identificador do lote no pasto
obsmorte	VARCHAR	Observações em relação à morte do animal
tpentrada	TINYINT	Identifica o método de entrada do animal ao rebanho (0 - Registro; 1 - Nascimento; 2 - Compra)
tpsaida	TINYINT	Identifica o método de saída do animal ao rebanho (0 - Não Saiu; 1 - Venda; 2 - Morte; 3 - Descarte)
tpusuario	VARCHAR	Tipo do usuário e permissões
versaoandroid	DOUBLE	Versão do aplicativo Aruana-Mobile quando realizada a exportação
versaoaruana	DOUBLE	Versão do sistema Aruana quando realizada a exportação
vlchip	BIGINT	Código do chip do animal no SISBOV
vlmesfinal	INTEGER	Idade final, em meses, da vigência da era do animal
vlmesinicial	INTEGER	Idade inicial, em meses, da vigência da era do animal
vlpeso	DOUBLE	Peso do animal na pesagem
vltotal	DOUBLE	Valor total da compra

APÊNDICE C – Modelo Entidade-Relacionamento

Na Figura 24 é apresentado o Modelo Entidade-Relacionamento completo da base de dados criada para o aplicativo.

Figura 24 - Modelo Entidade-Relacionamento - Completo

