

Eduardo Henrique ferreira

Sistema Agrícola

Sistema Agrícola

Eduardo Henrique Ferreira

Trabalho de Conclusão de Curso apresentado ao
Instituto Municipal de Ensino Superior de Assis,
como requisito do Curso de Graduação, analisado
pela seguinte comissão examinadora:

Orientador: _____

Analisador (1): _____

Analisador (2): _____

Eduardo Henrique Ferreira

Sistema Agrícola

Trabalho de Conclusão de Curso apresentado ao
Instituto Municipal de Ensino Superior de Assis,
como requisito do Curso de Graduação, analisado
pela seguinte comissão examinadora:

Orientador: _____

Área de Concentração: _____

DEDICATÓRIA

Dedico este trabalho primeiramente a minha família que me deu suporte para que eu fizesse um curso de graduação, dedico também a minha namorada que esteve sempre ao meu lado durante toda esta fase da minha vida.

AGRADECIMENTOS

Agradeço a Deus por ter me dado saúde e sabedoria durante estes três anos de aprendizagem.

Ao Professor Dr. Almir Rogério Camolesi pela Orientação neste trabalho, pela amizade e paciência.

Ao Professor Marcelo Manfio pelas dicas de como dar procedência ao trabalho.

A todos os professores da FEMA que me ensinaram muito durante o decorrer do curso.

Agradeço muito também as pessoas especiais na minha vida, minha mãe Eliana que me incentivou na escolha do curso, meu padrasto Benedito Granado que é quem influenciou muito no meu caráter e a minha namorada que me aguentou e teve paciência nos meus momentos de dificuldade.

A todos os amigos de sala que estiveram juntos na mesma caminhada e com certeza me motivaram nas horas que necessitei.

E também aos meus amigos de trabalho que me ajudaram muito durante este curso.

RESUMO

O agronegócio é um setor que vem sofrendo muito nos últimos anos com os constantes aumentos de custos dos recursos para o cultivo de cereais (Soja e Milho), muitas vezes os lucros obtidos com a safra não cobrem os gastos do plantio. Daí surge a idéia de desenvolver um sistema para ajudar os agricultores a ter uma melhor gestão de sua propriedade.

O sistema possibilita o usuário ter um melhor controle das atividades desenvolvidas em sua propriedade. Para o desenvolvimento deste sistema foi utilizado o framework Adobe Flex que utiliza conceitos de RIA (Aplicações de Internet Rica) que possibilita uma melhor interação entre o usuário e a aplicação, e também a plataforma ASP.NET. Ambas tecnologias integradas pelo FluorineFX e SQLEXPRESS 2005 como banco de dados.

ABSTRACT

Agribusiness is one sector that has suffered in recent years with steady increases in resource costs for cereal crops (soybeans and corn), many times the profits earned from the crop do not cover the cost of planting. So appear the idea to develop a system to help farmers get a better management of their property.

The system allows the User to have better control of the activities developed on their property. To develop this system was used Adobe Flex framework that uses concepts RIA () that enables a better interaction between the User and application, and also the ASP.NET platform for both technologies integrated FluorineFx and SQLEXPRESS 2005 as the database.

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1: Diagrama de Caso de uso Geral..... | 16 |
| Figura 2: Caso de uso: UC01 - Manter Cadastro de Funcionário..... | 17 |
| Figura 3: Caso de uso: UC02 - Manter Cadastro de Fornecedor..... | 18 |
| Figura 4: Caso de uso: UC03 - Manter Cadastro de Compra..... | 19 |
| Figura 5: Caso de uso: UC04 - Manter Cadastro de Insumos..... | 20 |
| Figura 6: Caso de uso: UC05 - Manter Cadastro de Maquinários..... | 21 |
| Figura 7: Caso de uso: UC06 - Manter Cadastro de Manutenção de Maquinário..... | 22 |
| Figura 8: Caso de uso: UC07 - Manter Cadastro de Propriedades | 23 |
| Figura 9: Caso de uso: UC08 - Manter Cadastro de Atividade..... | 24 |
| Figura 10: Caso de uso: UC09 - Manter Cadastro de Zona e Talhão..... | 25 |
| Figura 11: Caso de uso: UC10 - Manter Cadastro de Operações | 26 |
| Figura 12: Caso de uso: UC11 - Manter Cadastro de Culturas | 27 |
| Figura 13: Caso de uso: UC12 - Manter Cadastro de Safra..... | 28 |
| Figura 14: Caso de uso: UC13 - Manter Cadastro de Doenças | 29 |
| Figura 15: Caso de uso: UC14 - Manter Cadastro de Vendas | 30 |
| Figura 16: Caso de uso: UC15 - Emitir Relatório de Produtividade por Safra. | 31 |
| Figura 17: Caso de uso: UC16 - Emitir Relatório de insumos gastos por Safra..... | 32 |
| Figura 18: Caso de uso: UC17 - Emitir Relatório de Insumos, Maquinário, Funcionário e Operação por Atividade | 33 |
| Figura 19: Caso de uso: UC18 - Emitir Relatório de Manutenção por Maquinário | 34 |
| Figura 20: Caso de uso: UC19 - Emitir Relatório de Fornecedor | 35 |
| Figura 21: Caso de uso: UC20 - Emitir Relatório de Fornecedor por Insumo . | 35 |
| Figura 22: Caso de uso: UC21 - Emitir Relatório de Funcionário..... | 36 |
| Figura 23: Caso de uso: UC22 - Emitir Relatório de Compras..... | 37 |
| Figura 24: Caso de uso: UC23 - Emitir Relatório de Doenças | 37 |
| Figura 25: Caso de uso: UC24 - Emitir Relatório de Safras | 38 |
| Figura 26: Caso de uso: UC25 - Emitir Relatório de Vendas por Safra | 39 |
| Figura 27: Caso de uso: UC26 - Emitir Relatório de Cultura por Zona e Talhão..... | 40 |
| Figura 28: Caso de uso: UC27 - Emitir Relatório de Contas a pagar por Período..... | 41 |
| Figura 29: Caso de uso: UC28 - Emitir Relatórios de Contas quitadas por Período..... | 42 |
| Figura 30: Caso de uso: UC29 - Emitir Relatório de Contas atrasadas por Período..... | 43 |

| | |
|--|----|
| Figura 31: - Caso de uso: UC29 - Emitir um Gráfico de Quantidade colhida por Safra..... | 44 |
| Figura 32: Diagrama de Seqüência: DS 1 – Cadastrar Atividade..... | 45 |
| Figura 33: Diagrama de Seqüência: DS 2 – Cadastrar Compra..... | 45 |
| Figura 34: Diagrama de Seqüência: DS 3 – Cadastrar Manutenção..... | 46 |
| Figura 35: Diagrama de Seqüência: DS 4 – Relatório de Insumos Maquinário Funcionário Operação por Atividade | 46 |
| Figura 36: Diagrama de Seqüência: DS 5 – Relatórios de Insumos gastos por safra | 47 |
| Figura 37: Diagrama de Deploy..... | 48 |
| Figura 38: Diagrama de Classes de Negócios | 49 |
| Figura 39: Organização das Camadas .net. | 50 |
| Figura 40: Camada MOR | 50 |
| Figura 41: Camada DAL..... | 51 |
| Figura 42: Camada BLL | 51 |
| Figura 43: Organização dos diretórios Flex..... | 52 |
| Figura 44: Referencia de Entidade..... | 52 |
| Figura 45: Modo Source | 53 |
| Figura 46: Modo design..... | 53 |
| Figura 47: FluorineFX ASP.NET Web Site..... | 54 |
| Figura 48: Referencia Agrícola | 54 |
| Figura 49: Servidor IIS | 55 |
| Figura 50: Pagina Home | 55 |
| Figura 51: Pagina de Cadastros..... | 56 |
| Figura 52: Pagina de Relatórios | 56 |
| Figura 53: Pagina de Visualização de Talhões..... | 57 |

SUMARIO

| | |
|---|-----------|
| Capitulo 1 : Introdução | 11 |
| Capitulo 2 : Tecnologias Utilizadas..... | 12 |
| 2.1 A UML | 12 |
| 2.2 A Tecnologia ASP.NET | 12 |
| 2.3 Adobe Flex..... | 13 |
| Capitulo 3 : Modelagem do Sistema Agrícola..... | 14 |
| 3.1 Análise de requisitos | 14 |
| 3.2 Lista de Eventos..... | 15 |
| 3.3 Diagrama de caso de uso geral | 16 |
| 3.4 Diagrama de caso de uso | 17 |
| 3.5 Diagrama de seqüência | 45 |
| 3.6 Diagrama de deploy | 48 |
| 3.7 Diagrama de Classes..... | 49 |
| Capitulo 4 : Implementação..... | 50 |
| 5.1 Organização do projeto na plataforma .net | 50 |
| 5.2 Organização do projeto no Flex | 52 |
| 5.3 Integração das tecnologias utilizando o FluorineFX..... | 54 |
| 5.4 Interfaces do Sistema | 55 |
| Capitulo 5 : Conclusão..... | 58 |
| Capitulo 6 : Referencias Bibliográficas | 59 |

1. Introdução

Nos últimos anos o uso de *softwares* de gerenciamento vem crescendo muito nas empresas, isso ocorre pela complexidade de manipular os dados que são extremamente importantes para a tomada de decisões.

Quem vem utilizando esse tipo de software tem obtido resultados muito bons, pois através de relatórios os diretores e gerentes podem ver mais claramente onde a empresa está errando podendo assim melhorar antes que o erro tome maiores proporções.

O agronegócio é um setor que vem sofrendo muito nos últimos anos com os constantes aumentos de custos dos recursos para o cultivo de cereais (Soja e Milho), muitas vezes os lucros obtidos com a safra não cobrem os gastos do plantio. A agricultura já conta com muitas tecnologias em maquinários e juntamente com a utilização de software o agricultor poderá administrar melhor os seus custos.

Os agricultores geralmente fazem o gerenciamento da sua propriedade manualmente, o grande problema disso é a vulnerabilidade dos dados, que tornam-se não muito confiáveis.

Daí surge a idéia de desenvolver um software de gerenciamento para a agricultura, pois também é uma área que depende muito de decisões acertadas para o seu sucesso.

O sistema visa ajudar o produtor a ter uma melhor gestão de sua propriedade, pois proporcionará dados mais precisos nos quais o usuário poderá confiar.

O software ajuda administrar os custos de produção desde o plantio, manejo até a colheita e conta com uma base de dados que permite ao usuário (agricultor) inserir todos os dados relacionados a sua atividade agrícola assim, mais tarde o mesmo contará com uma grande ferramenta para o auxílio nas sua tomada de decisões.

O trabalho foi dividido em capítulos visando uma melhor organização.

O capítulo 2 trata sobre as tecnologias utilizadas para o desenvolvimento do trabalho. No capítulo 3 foi desenvolvida a modelagem do sistema utilizando padrões UML.

No capítulo 4 será mostrada implementação do sistema com exemplos de como integra o Flex com o C#. No capítulo 5 será mostrada uma breve conclusão do trabalho, e no ultimo capítulo mostraremos as referencia bibliográficas utilizadas.

2. Tecnologias Utilizadas

2.1 A UML

A UML (*Unified Modeling Language*) é uma linguagem de modelagem de sistema para notação (principalmente gráfica) utilizada por métodos para expressar projetos, utilizando conceitos de orientação a objetos. Padronizada pela OMG (*Object Management Group*, uma entidade de padronização estabelecida pela indústria), a qual recebeu a aprovação de fato pela indústria (Larman,2000).

No final dos anos oitenta e início dos anos noventa o mercado de métodos de análise e projetos orientados a objetos estava bastante dividido e competitivo. Em 1994 *Jim Rumbaugh* deixou a General Eletric e se uniu a *Grady Booch* na *Rational Software*, com a intenção de unificar seus métodos populares - *Booch* e *OMT (Object Modeling Technique)*. No ano seguinte, a *Rational Software* comprou *Objectory* e *Ivar Jacobson* uniu-se a eles com seu método *OOSE (Object Oriented Software Engineering)*. Em 1996 ficaram conhecidos amplamente como "os três amigos" e seu método sob o nome de UML. Em 1997, a UML passou por um processo de padronização pelo OMG e tornou-se um padrão em engenharia de software (Larman,2000).

2.2 A Tecnologia ASP.NET

A tecnologia ASP.NET (*Active Server Pages*), sucessora do ASP 3.0, é uma plataforma da Microsoft que foi desenvolvida para facilitar o desenvolvimento de aplicações Web, e fornecer um novo modelo de programação, a arquitetura de três camadas (Shepherd,2007).

O ASP.NET é a primeira ferramenta RAD (*Rapid Application Design*) para o desenvolvimento de aplicativos específicos para a Web. Esses aplicativos rodam somente sob o Servidor Web da Microsoft IIS (*Information Internet Server*), porém as páginas são acessíveis em qualquer navegador e em qualquer Sistema Operacional, pois as aplicações dependem apenas do navegador (Shepherd,2007).

O ambiente de execução, o qual dá suporte a toda as linguagens de programação habilitadas para o .NET (Visual Basic .NET, C#, ASP .NET e Jscript.NET) é o CRL - *Common Language Runtime*, o que poupa tempo e agiliza o processo de desenvolvimento. E as aplicações criadas ao serem compiladas, geram um código intermediário conhecido como MSIL - *Microsoft Intermediate Language*. Esse código que é executado pelo CRL

(Shepherd,2007).

O desenvolvimento em ASP.NET é feito sobre formulários, o qual esta associado a um arquivo com extensão .aspx. Sobre os formulários são adicionados componentes que têm propriedades, métodos e eventos. Por isso podemos afirmar que esta tecnologia é orientada a objetos (Shepherd,2007).

O ASP.NET é parte integrante do Framework .NET, herdando todas as suas características, e desta maneira temos acesso a todos os benefícios da plataforma .NET. Com isso podemos acessar toda biblioteca de classes do *Framework .NET - .NET Framework Class Library* (Shepherd,2007).

Com ASP.NET podemos utilizar qualquer linguagem que esteja habilitado para o *Framework .NET* como, VB .NET, C# e JScript, as quais são compiladas, o que significa que toda página ASP .NET é compilada antes de ser executada. Além disso a versão compilada é mantidas em Cache, o que torna a carga da página muito mais rápida (Shepherd,2007).

Portanto com o ASP.NET temos acesso a uma série de facilidades que diminuem a quantidade de códigos que o programador precisa escrever, principalmente no tratamento de funções básicas. Isso faz com que o programador tenha que desenvolver apenas a lógica da aplicação, sem ter que se preocupar com aspectos básicos (Shepherd,2007).

2.3 Adobe Flex

Flex é um framework de desenvolvimento de aplicações RIA (Aplicações de Internet Rica) da Adobe, contendo uma poderosa linguagem de programação chamada Action Script 3.0 e uma linguagem para o desenvolvimento de interfaces chamada MXML (*Macromedia Flex Markup Language*) . Com este conjunto o flex torna-se uma ferramenta essencial para a criação de softwares para a web, utilizando conceitos de RIA. (ADOBE, 2007).

A metodologia de funcionamento do adobe flex é bastante semelhante ao adobe flash, que é uma ferramenta para animação que ganhou mercado no mundo todo. Hoje a maioria dos sites utiliza algum recursos em flash, e mais de 98% dos navegadores possuem o plug-in do flash para executarem suas animações. O adobe Flex trabalha de modo semelhante ao flash , sendo o aplicativo final um arquivo com a extensão SWF. (Schmitz, 2008).

Para a integração do adobe Flex com a plataforma .net é utilizado o fluorineFX que é uma biblioteca open source que permite trabalhar com objetos remotos entre o flex e a plataforma .net em tempo real. (Schmitz, 2008).

3. Modelagem do Sistema Agrícola

3.1 Análise de requisitos

Para o desenvolvimento desse sistema não foi realizada entrevista padrão com o usuário por realmente não ter um cliente em específico. No entanto todas as necessidades do sistema surgiram de conversas com o professor Marcelo Manfio que atua na área agrícola e tem uma experiência muito grande em softwares deste tipo e também conversas com meu orientador Professor Dr. Almir Rogério Camolesi que também já atuou na área e me deu todo o suporte para o desenvolvimento desta análise.

Nessas conversas foram levantados pontos importantes para o meu trabalho:

- Para o usuário ter uma melhor visão de sua atividade agrícola o sistema utilizará uma metodologia que irá dividir suas propriedades por Zona e talhão.
- Será desenvolvido um cadastro onde o usuário deverá cadastrar todas e quaisquer atividades que venham a ser executada dentro de sua propriedade. O cadastro deverá conter informações importantes como quais funcionários trabalharam na atividade, quais maquinários foram utilizados, qual o tipo e a quantidade de insumos foram gastos, dentre outras informações.
- Como o usuário pode não trabalhar somente com agricultura, como é o caso do professor Marcelo Manfio, o sistema será desenvolvido com padrões web e com uma interface onde terá as zonas e talhões representadas graficamente, e ao clicar com o mouse no talhão, o usuário terá acesso a ultima atividade desenvolvida em determinado talhão.

3.2 Lista de eventos

1. **Manter cadastro de funcionário**
2. **Manter cadastro de insumos**
3. **Manter cadastro de maquinários**
4. **Manter cadastro de propriedades**
5. **Manter cadastro de atividade**
6. **Manter cadastro de zona**
7. **Manter cadastro de talhão**
8. **Manter cadastro de operações**
9. **Manter cadastro de cultura**
10. **Manter cadastro de safra**
11. *Manter cadastro de manutenção de maquinário*
12. *Manter cadastro de fornecedor*
13. *Manter cadastro de compras*
14. *Manter cadastro de doenças*
15. *Manter cadastro de vendas*
16. **Emitir relatório de produtividade por safra**
17. **Emitir relatório de insumos gastos por safra**
18. **Emitir relatório de insumos, maquinário, funcionário, operação por Atividade**
19. **Emitir relato de funcionário**
20. **Emitir relatório de safra**
21. **Emitir relatório de cultura por talhão**
22. *Emitir relatório de manutenção por maquina*
23. *Emitir relatório de fornecedor*
24. *Emitir relatório de fornecedor por insumo*
25. *Emitir relatório de compra*
26. *Emitir relatório de venda por safra*
27. *Emitir relatório de contas a pagar por período*
28. *Emitir relatórios de contas quitadas por período*
29. *Emitir relatório de contas atrasadas por período*
30. *Emitir um gráfico de quantidade colhida por safra*

Obs: Os eventos em negrito são os que foram implementados nesta etapa do trabalho.

3.4 Diagrama de caso de uso

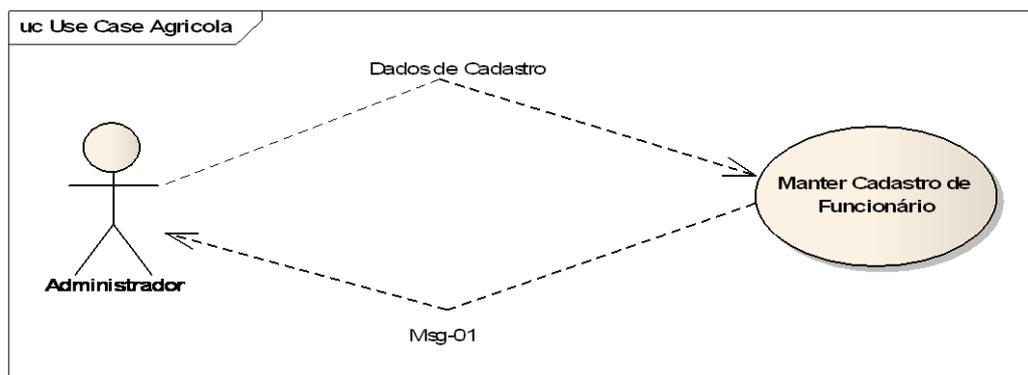


Figura 2 - Caso de uso: UC01- Manter Cadastro de Funcionário

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todos os funcionários no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar funcionário no menu principal e em seguida escolhe a opção cadastrar funcionário (A1) (A2).
- O sistema solicita os dados necessários para o cadastro do funcionário.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir funcionário.

- a) O sistema solicita o código do funcionário a ser excluído.

A2: O usuário escolhe a opção Alterar funcionário.

- a) O sistema solicita o código do funcionário a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. asos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

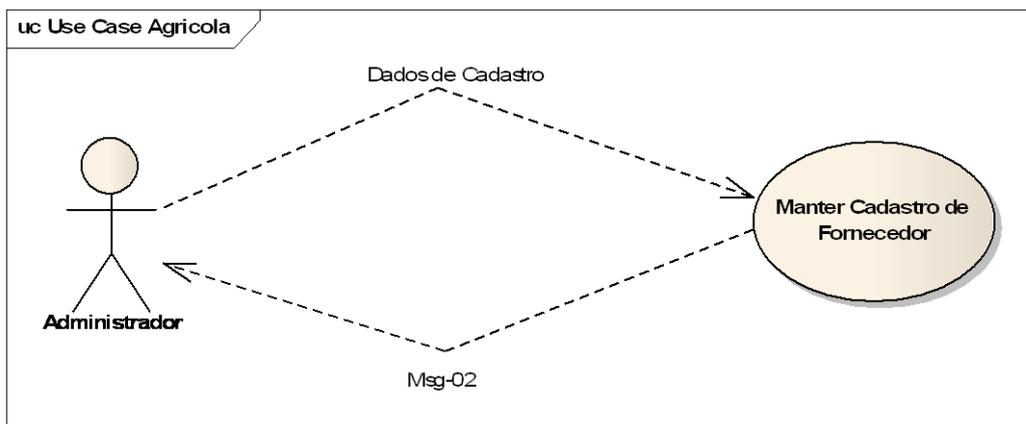


Figura 3 - Caso de uso: UC02 - Manter Cadastro de fornecedor

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todos os fornecedores no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar fornecedor no menu principal e em seguida escolhe a opção cadastrar fornecedor (A1) (A2).
- O sistema solicita os dados necessários para o cadastro do fornecedor.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir fornecedor.

- a) O sistema solicita o código do fornecedor a ser excluído.

A2: O usuário escolhe a opção Alterar fornecedor.

- a) O sistema solicita o código do fornecedor a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

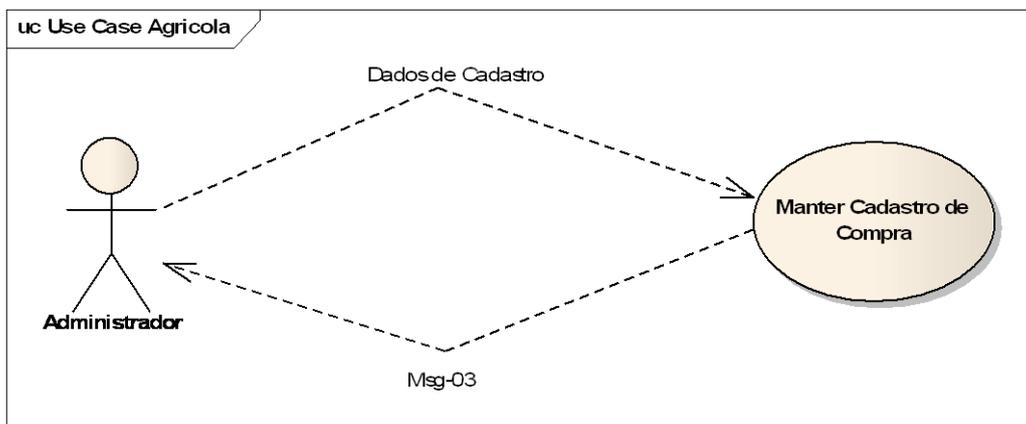


Figura 4 - Caso de uso: UC03 - Manter Cadastro de Compra

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar todas as compras no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar compra no menu principal
- O sistema solicita os dados necessários para o cadastro do compra.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

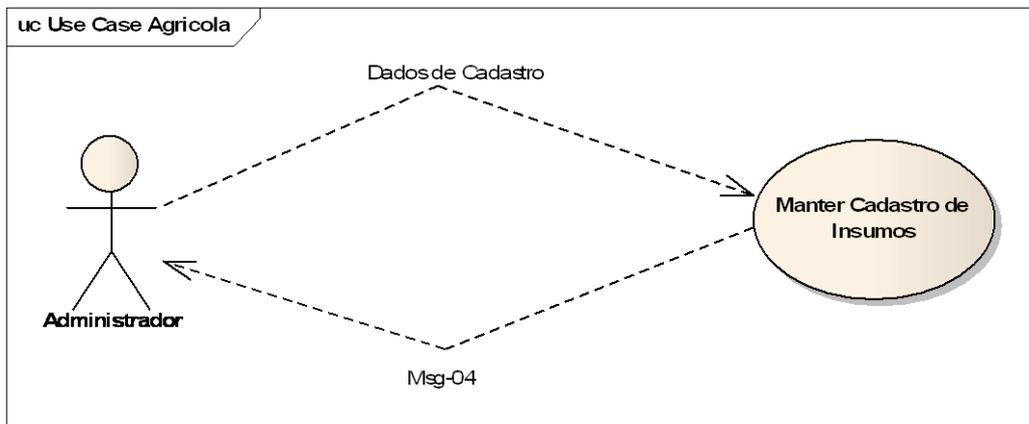


Figura 5 - Caso de uso: UC04 - Manter Cadastro de Insumos

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todos os insumos no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar insumos no menu principal e em seguida escolhe a opção cadastrar insumos (A1) (A2).
- O sistema solicita os dados necessários para o cadastro do insumos.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir insumos.

- a) O sistema solicita o código do insumos a ser excluído.

A2: O usuário escolhe a opção Alterar insumos.

- a) O sistema solicita o código do insumos a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

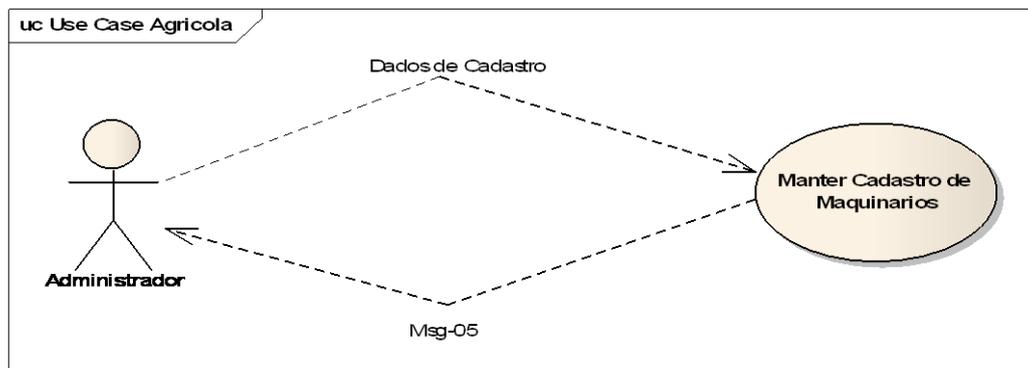


Figura 6 - Caso de uso: UC05 - Manter Cadastro de Maquinários

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todos os maquinários no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar maquinário no menu principal e em seguida escolhe a opção cadastrar maquinário (A1) (A2).
- O sistema solicita os dados necessários para o cadastro do maquinário.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir maquinário.

- a) O sistema solicita o código do maquinário a ser excluído.

A2: O usuário escolhe a opção Alterar maquinário.

- a) O sistema solicita o código do maquinário a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

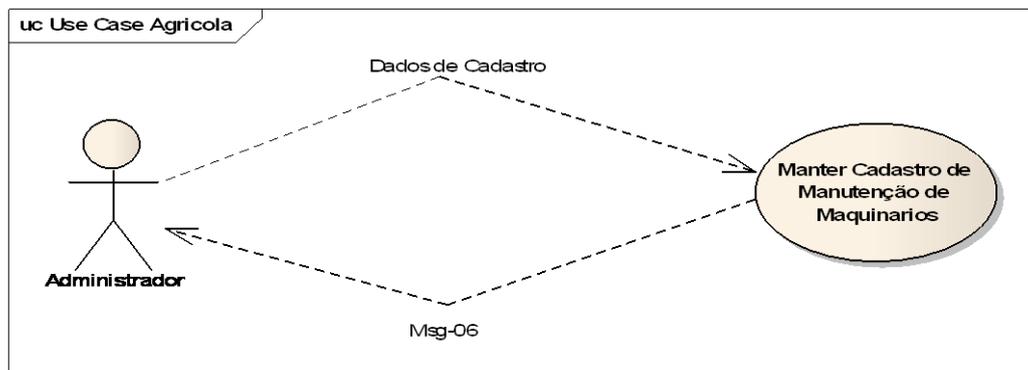


Figura 7 - Caso de uso: UC06 - Manter Cadastro de Manutenção de Maquinário

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as manutenções de maquinário no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar manutenção de maquinário no menu principal e em seguida escolhe a opção cadastrar manutenção (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da manutenção de maquinário.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir manutenção.

- a) O sistema solicita o código do maquinário a ser excluído.

A2: O usuário escolhe a opção Alterar manutenção.

- a) O sistema solicita o código do maquinário a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

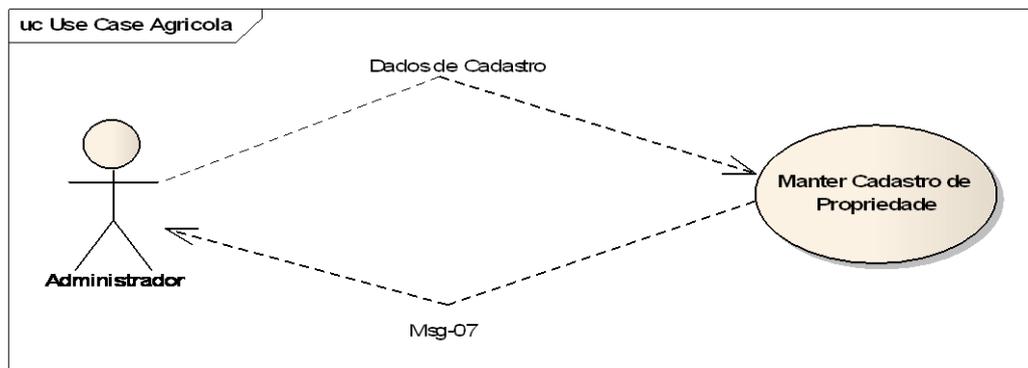


Figura 8 - Caso de uso: UC07 - Manter Cadastro de Propriedades

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as propriedades no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar propriedade no menu principal e em seguida escolhe a opção cadastrar propriedade (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da propriedade.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir propriedade.

- a) O sistema solicita o código da propriedade a ser excluído.

A2: O usuário escolhe a opção Alterar propriedade.

- a) O sistema solicita o código da propriedade a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

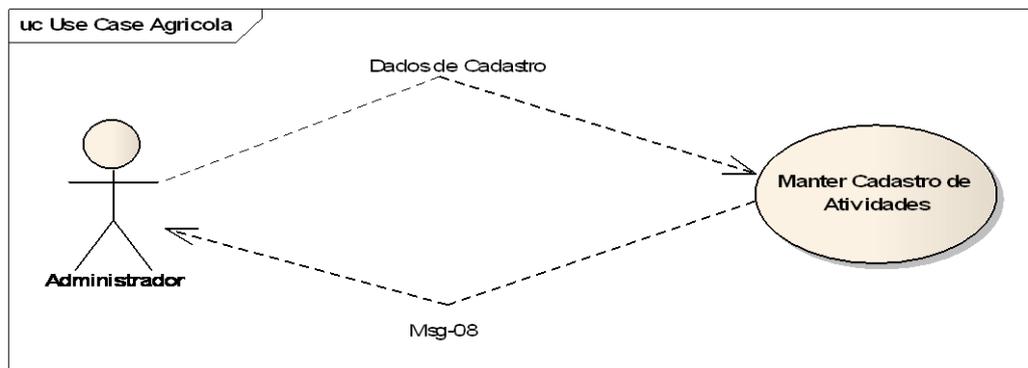


Figura 9 - Caso de uso: UC08 - Manter Cadastro de Atividade

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as Atividades desenvolvidas na propriedade no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar Atividade no menu principal e em seguida escolhe a opção cadastrar Atividade (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da Atividade.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir Atividade.

- a) O sistema solicita o código da Atividade a ser excluído.

A2: O usuário escolhe a opção Alterar Atividade.

- a) O sistema solicita o código da Atividade a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

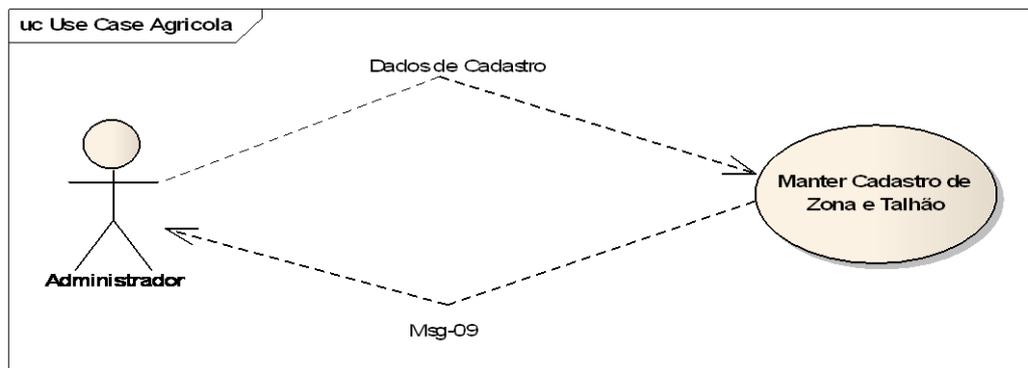


Figura 10 - Caso de uso: UC09 - Manter Cadastro de Zona e Talhão

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as Zonas e Talhões no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar Zona e Talhão no menu principal e em seguida escolhe a opção cadastrar Zona e Talhão (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da Zona e Talhão.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir Zona e Talhão.

- a) O sistema solicita o código da Zona e Talhão a ser excluído.

A2: O usuário escolhe a opção Alterar Zona e Talhão.

- a) O sistema solicita o código da Zona e Talhão a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

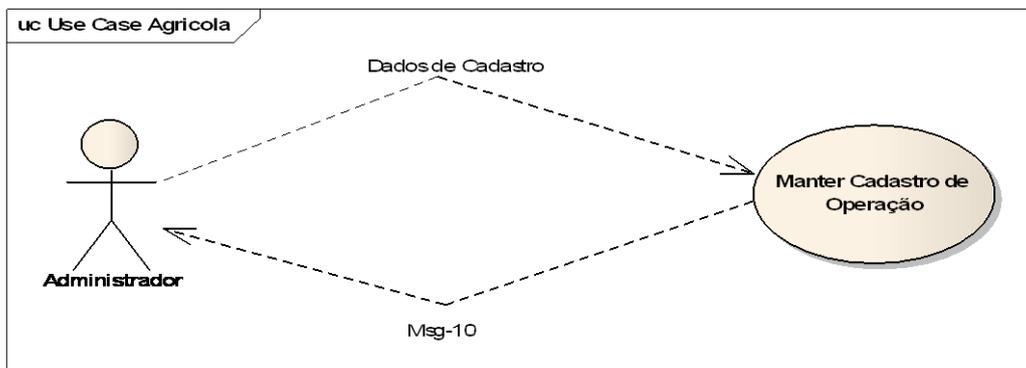


Figura 11 - Caso de uso: UC10 - Manter Cadastro de Operações

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as Operações no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar Operações no menu principal e em seguida escolhe a opção cadastrar Operação (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da Operação.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir Operação.

- a) O sistema solicita o código da Zona e Talhão a ser excluído.

A2: O usuário escolhe a opção Alterar Operação.

- a) O sistema solicita o código da Zona e Talhão a ser alterado

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

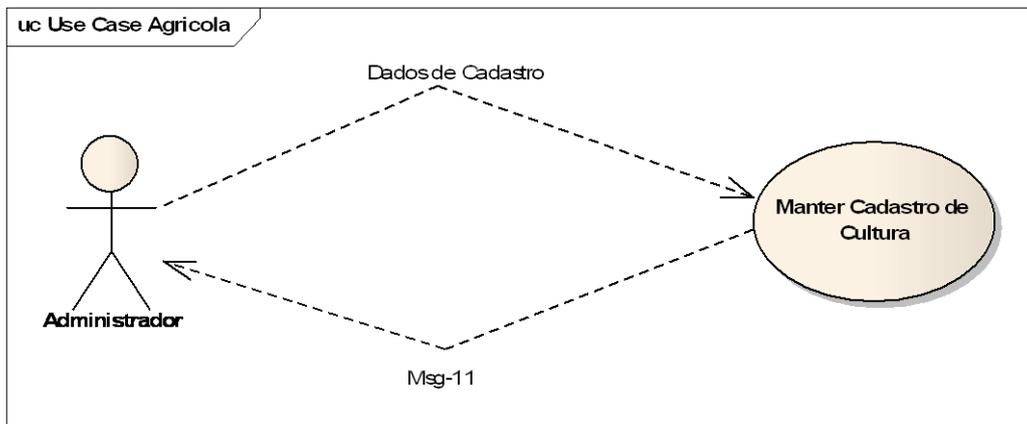


Figura 12 - Caso de uso: UC11 - Manter Cadastro de Culturas

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as culturas no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar culturas no menu principal e em seguida escolhe a opção cadastrar cultura (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da cultura.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir cultura.

- a) O sistema solicita o código da cultura a ser excluído.

A2: O usuário escolhe a opção Alterar cultura.

- a) O sistema solicita o código da cultura a ser alterado.

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

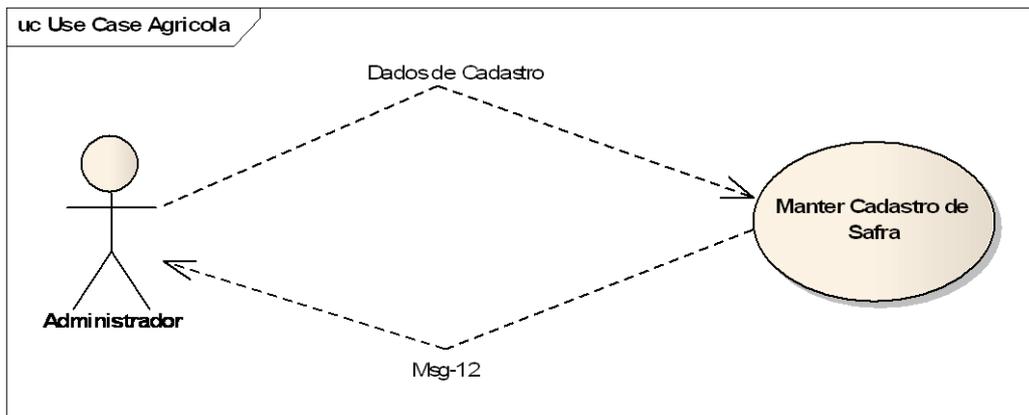


Figura 13 - Caso de uso: UC12 - Manter Cadastro de Safra

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as Safra no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar safra no menu principal e em seguida escolhe a opção cadastrar safra (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da safra.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir safra.

- a) O sistema solicita o código da safra a ser excluído.

A2: O usuário escolhe a opção Alterar safra.

- a) O sistema solicita o código da safra a ser alterado.

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

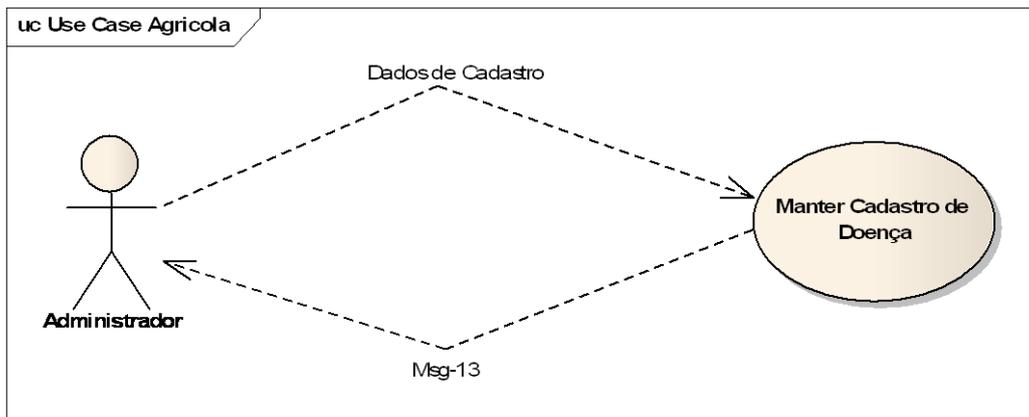


Figura 14 - Caso de uso: UC13 - Manter Cadastro de Doenças

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as Doenças no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar Doenças no menu principal e em seguida escolhe a opção cadastrar Doença (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da Doença.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir Doença.

- a) O sistema solicita o código da doença a ser excluído.

A2: O usuário escolhe a opção Alterar Doença.

- a) O sistema solicita o código da doença a ser alterado.

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

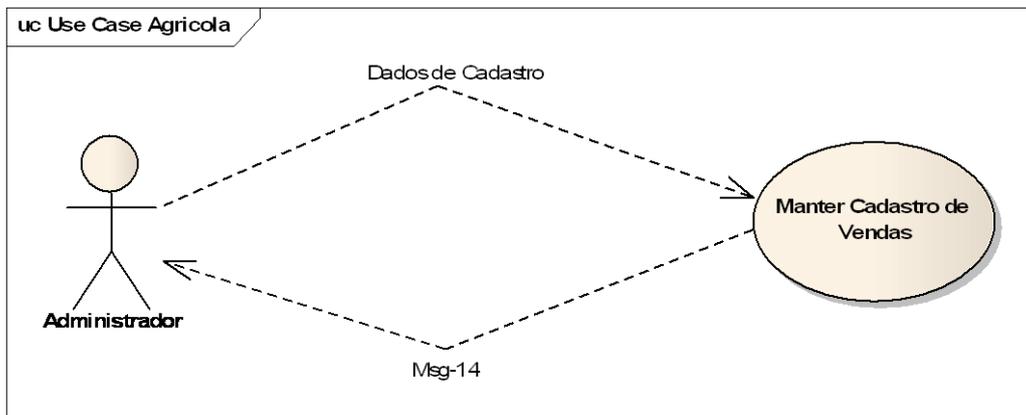


Figura 15- Caso de uso: UC14 - Manter Cadastro de Vendas

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) cadastrar e manter todas as vendas no sistema.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção cadastrar vendas no menu principal e em seguida escolhe a opção cadastrar venda (A1) (A2).
- O sistema solicita os dados necessários para o cadastro da venda.
- O usuário fornece os dados solicitados e confirma a operação (A3).
- O sistema verifica se todos os dados foram fornecidos e em seguida mostra uma mensagem de confirmação ao usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Excluir venda.

- a) O sistema solicita o código da venda a ser excluído.

A2: O usuário escolhe a opção Alterar venda.

- a) O sistema solicita o código da venda a ser alterado.

A3: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar domínios dos campos (Campos Numéricos, datas e etc...).

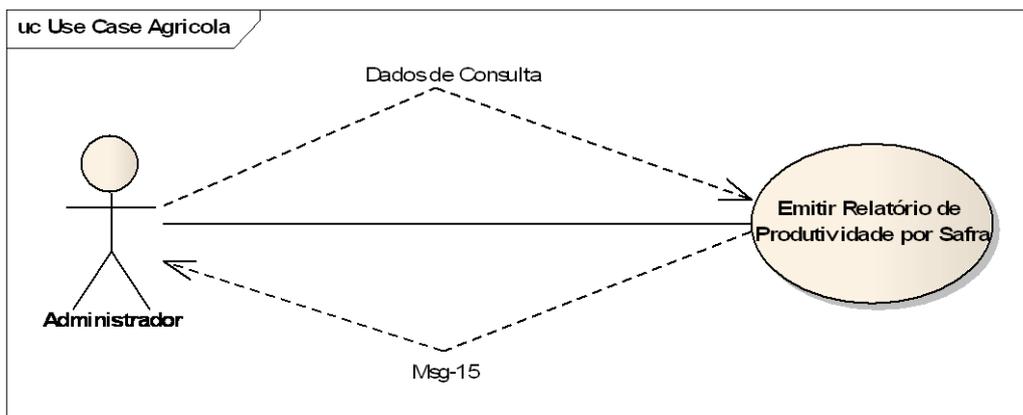


Figura 16 - Caso de uso: UC15 – Emitir Relatório de Produtividade por Safra

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de produtividade por safra.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios de produtividade por safra .
- O sistema solicita os dados da safra desejada.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se a safra informada existe.

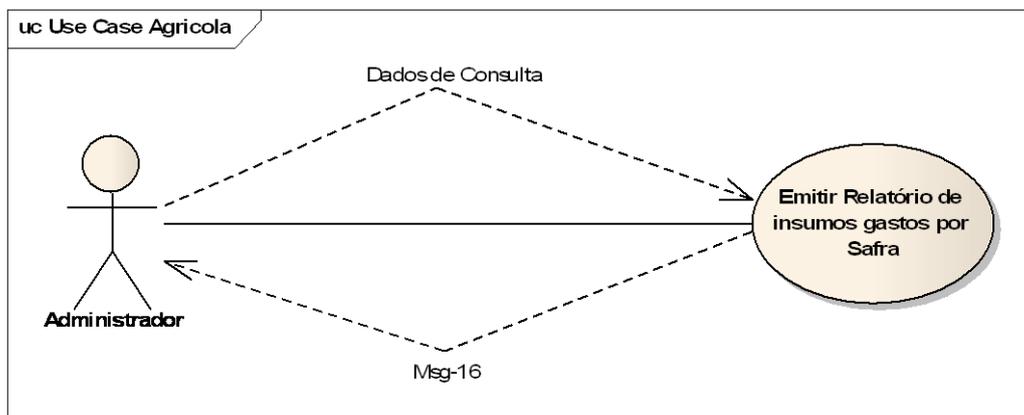


Figura 17 - Caso de uso: UC16 – Emitir Relatório de insumos gastos por Safra

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de insumos gastos por safra.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios de insumos gastos por safra .
- O sistema solicita os dados da safra desejada.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

- A1:** O usuário escolhe a opção Cancelar a operação.
- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se a safra informada existe.

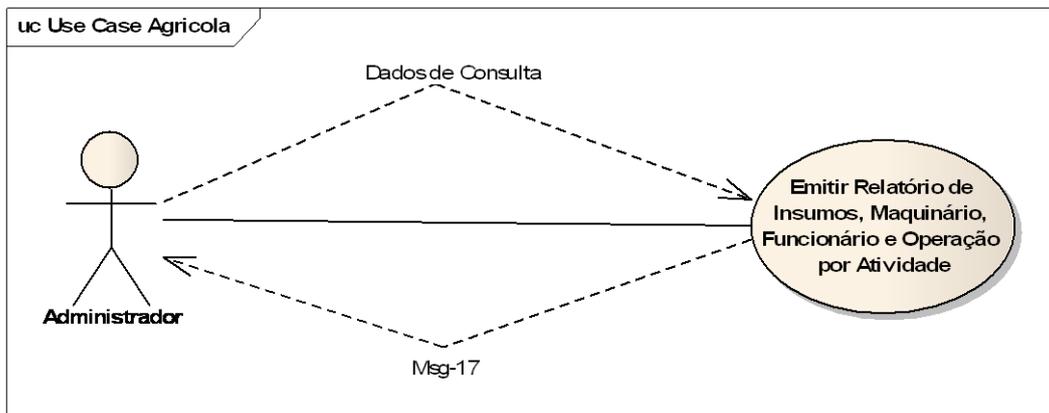


Figura 18 - Caso de uso: UC17 – Emitir Relatório de Insumos, Maquinário, Funcionário e Operação por Atividade

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de insumos, maquinário, funcionário e operação por atividade

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios de insumos, maquinário, funcionário e operação por atividade.
- O sistema solicita os dados da Atividade desejada.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

- A1:** O usuário escolhe a opção Cancelar a operação.
- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se a atividade informada existe.

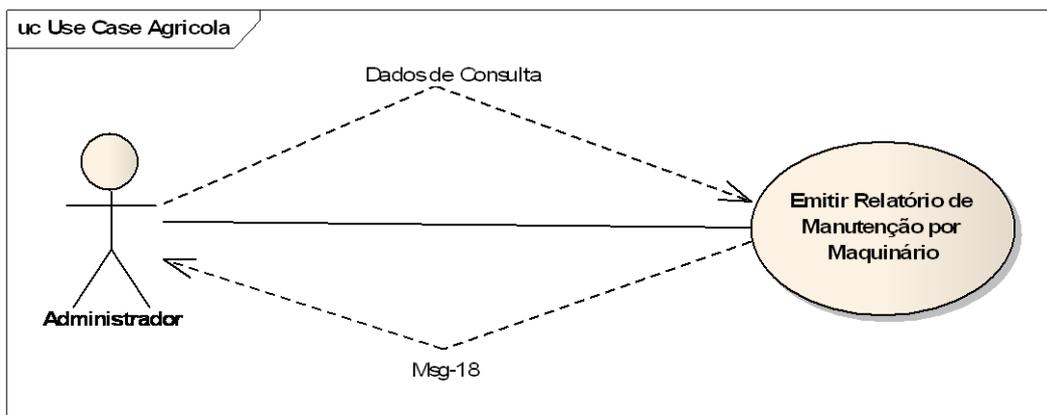


Figura 19 - Caso de uso: UC18 – Emitir Relatório de Manutenção por Maquinário

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de manutenções feitas por maquinários.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios manutenção por maquinário.
- O sistema solicita os dados do maquinário desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

- A1:** O usuário escolhe a opção Cancelar a operação.
- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se o maquinário informado existe.



Figura 20 - Caso de uso: UC19 – Emitir Relatório de Fornecedor

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios todos os fornecedores cadastrados.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios fornecedor.
- O sistema emite relatório na tela para o usuário.

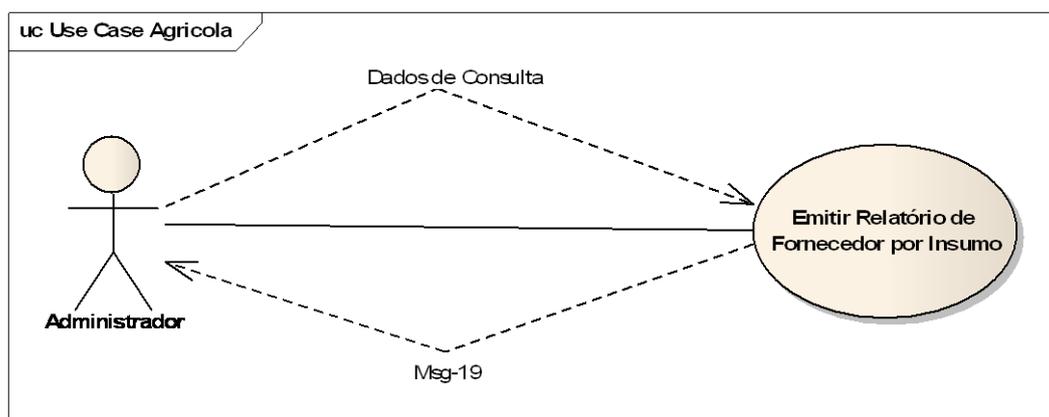


Figura 21 - Caso de uso: UC20 – Emitir Relatório de Fornecedor por Insumo

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de fornecedores por tipo de insumo.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios fornecedor por insumo.
- O sistema solicita os dados do insumo desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se o insumo informado existe.

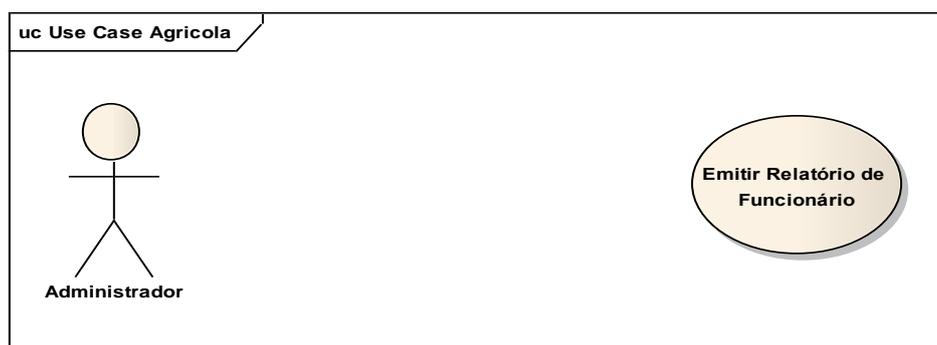


Figura 22 - Caso de uso: UC21 – Emitir Relatório de Funcionário

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios todos os funcionários cadastrados.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios funcionário.
- O sistema emite relatório na tela para o usuário.



Figura 23 - Caso de uso: UC22 – Emitir Relatório de Compras

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios todas as compras cadastradas.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios compras.
- O sistema emite relatório na tela para o usuário.

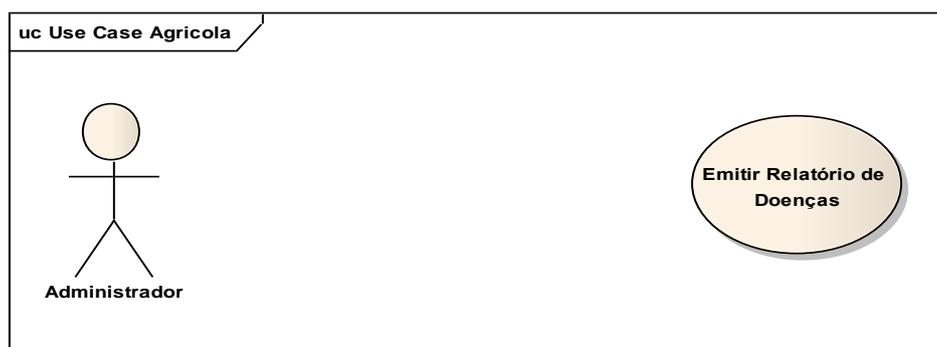


Figura 24 - Caso de uso: UC23 – Emitir Relatório de Doenças

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios todas as doenças cadastradas.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios doenças.
- O sistema emite relatório na tela para o usuário.

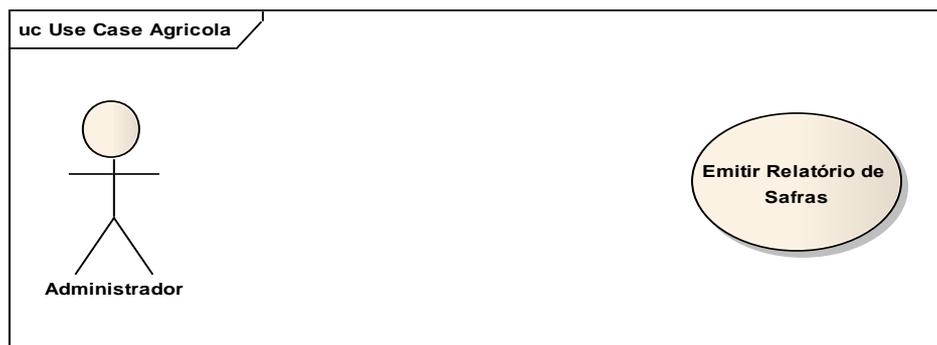


Figura 25 - Caso de uso: UC24 – Emitir Relatório de Safras

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios todas as safras cadastradas.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios safras.
- O sistema emite relatório na tela para o usuário.

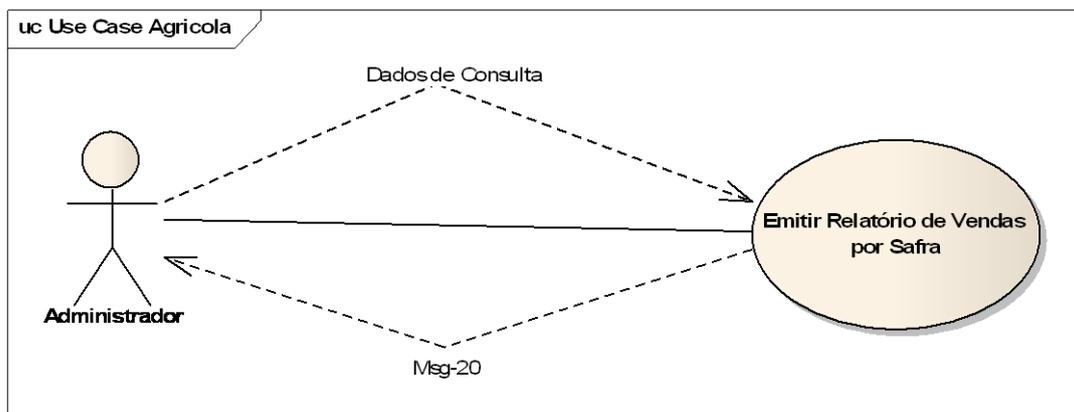


Figura 26 - Caso de uso: UC25 – Emitir Relatório de Vendas por Safra

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de vendas por safra.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios de Vendas por Safra.
- O sistema solicita os dados do safra desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se a safra informada existe.

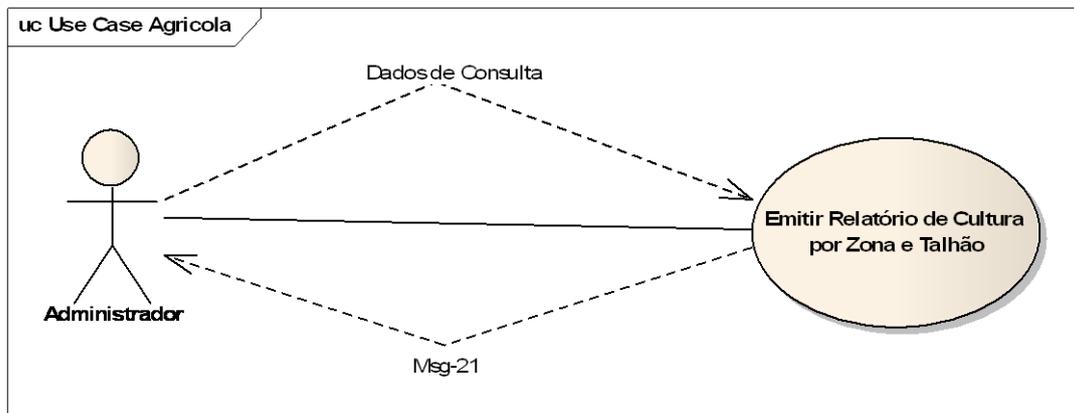


Figura 27 - Caso de uso: UC26 – Emitir Relatório de Cultura por Zona e Talhão

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de cultura por zona e talhão

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios de cultura por zona e talhão.
- O sistema solicita os dados da zona e talhão desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

- A1:** O usuário escolhe a opção Cancelar a operação.
- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se a zona e talhão informados existem.

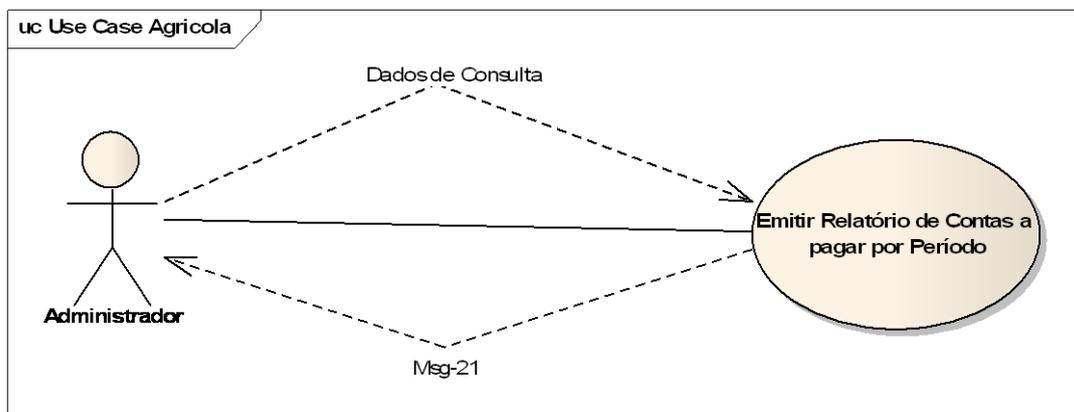


Figura 28 - Caso de uso: UC27 – Emitir Relatório de Contas a pagar por Período

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatório de contas a pagar por período

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatório de contas a pagar por período.
- O sistema solicita o período desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se o período informado existe.

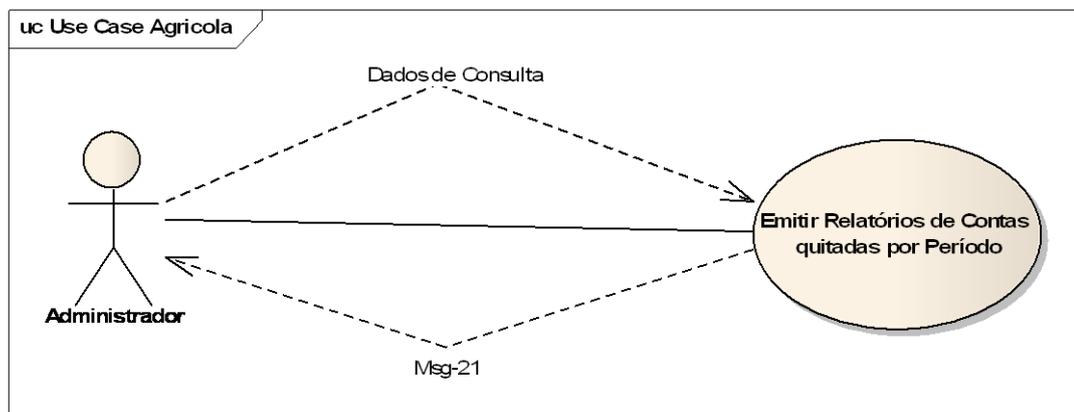


Figura 29 - Caso de uso: UC28 – Emitir Relatórios de Contas quitadas por Período

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatórios de contas quitadas por período

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatórios de contas quitadas por período.
- O sistema solicita o período desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

- A1:** O usuário escolhe a opção Cancelar a operação.
- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se o período informado existe.

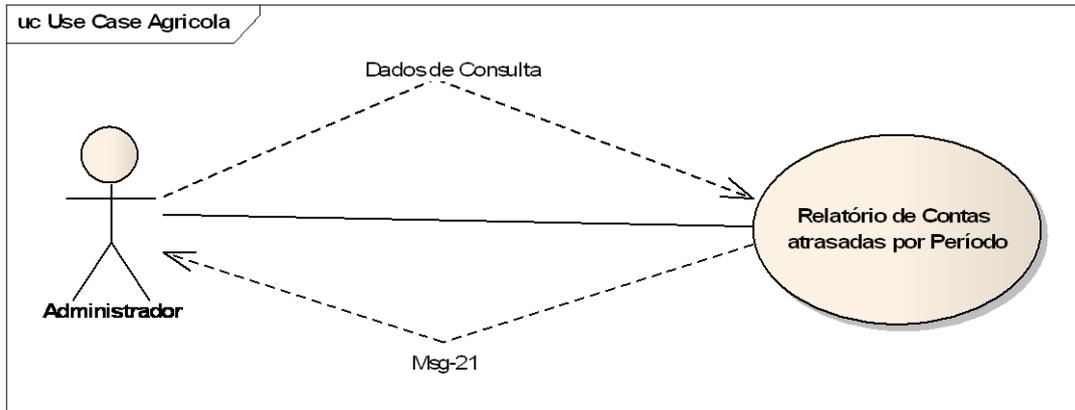


Figura 30 - Caso de uso: UC29 – Emitir Relatório de Contas atrasadas por Período

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar relatório de contas atrasadas por período.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção relatório de contas atrasadas por período.
- O sistema solicita o período desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

A1: O usuário escolhe a opção Cancelar a operação.

- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se o período informado existe.

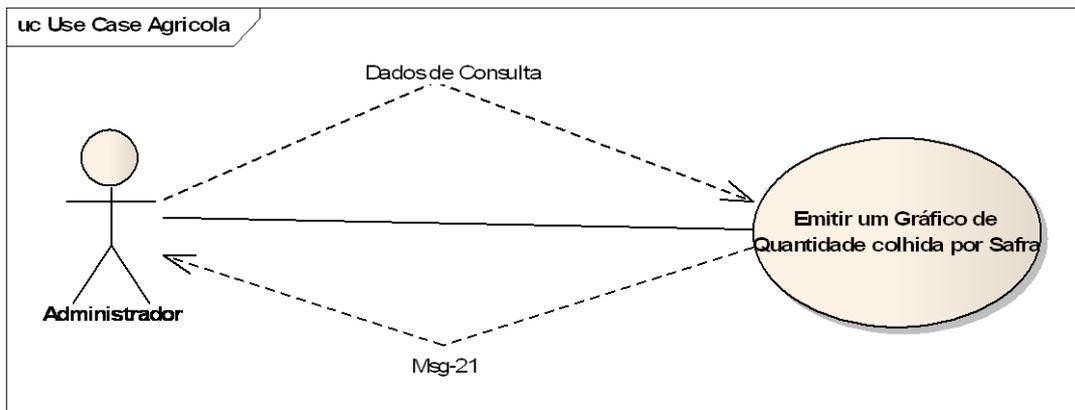


Figura 31 - Caso de uso: UC29 – Emitir um Gráfico de Quantidade colhida por Safra

1. Finalidade/Objetivo

- Permite ao usuário (Administrador) gerar um Gráfico de Quantidade colhida por Safra.

2. Ator(es)

- Administrador

3. Precondições

- O usuário deve ter efetuado o login como Administrador

4. Fluxo principal

- O administrador seleciona a opção gerar relatórios no menu principal e em seguida escolhe a opção Gráfico de Quantidade colhida por Safra.
- O sistema solicita a safra desejado.
- O usuário fornece os dados solicitados e confirma a operação (A1).
- O sistema emite relatório na tela para o usuário.

5. Fluxo Alternativo

- A1:** O usuário escolhe a opção Cancelar a operação.
- a) O sistema volta ao menu principal.

6. Casos de teste

- Verificar se a safra informada existe.

3.5 Diagrama de sequência

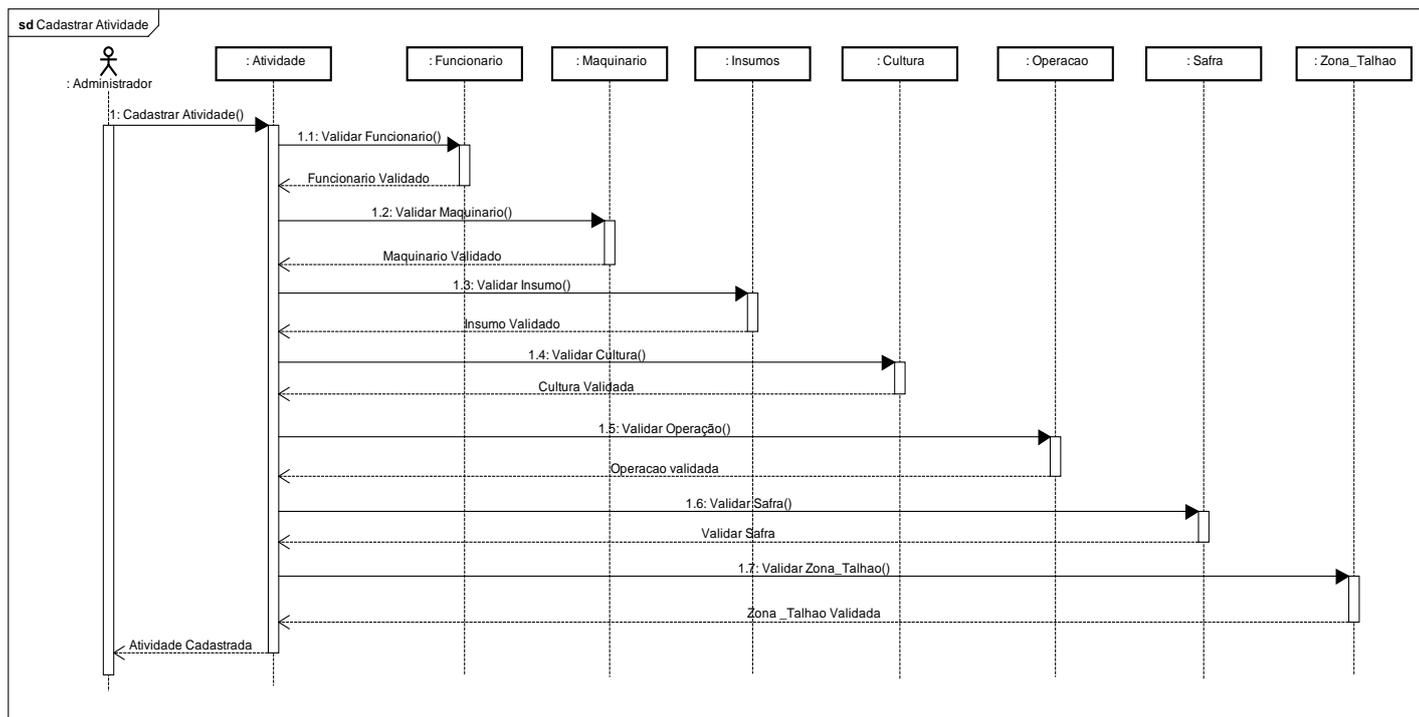


Figura 32 – Diagrama de Seqüência: DS 1 – Cadastrar Atividade

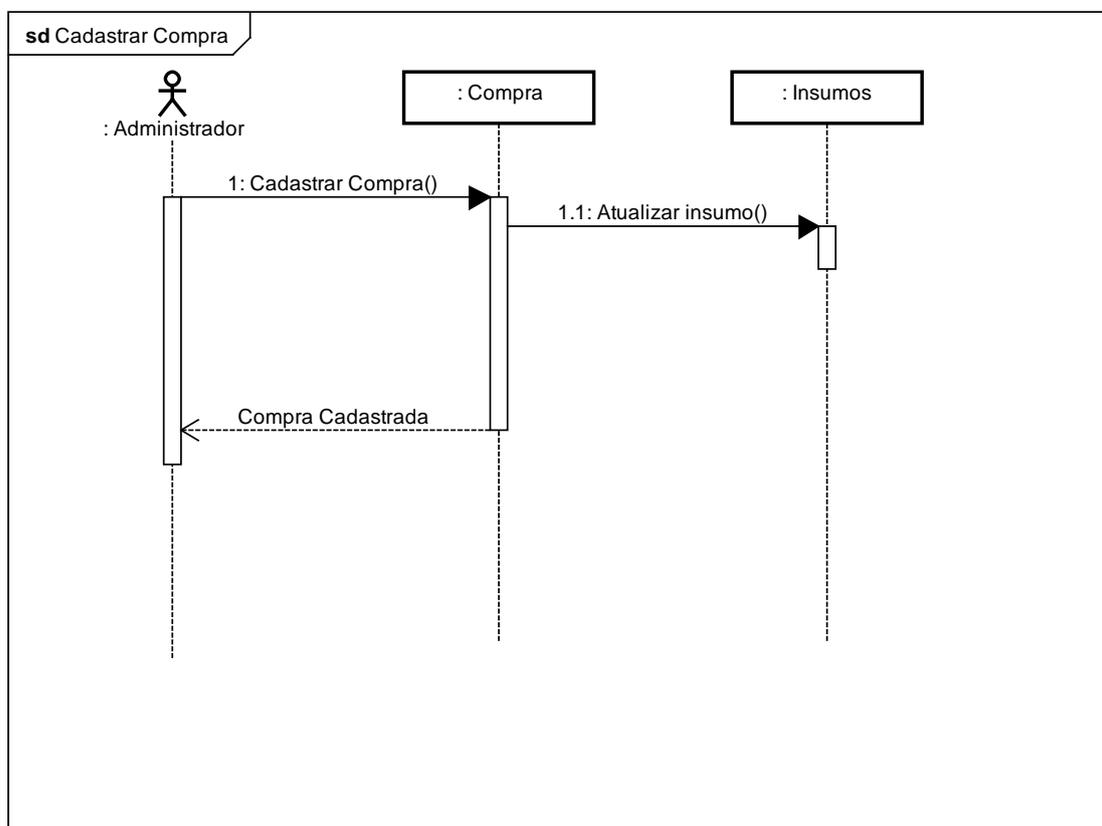


Figura 33 – Diagrama de Seqüência: DS 2 – Cadastrar Compra

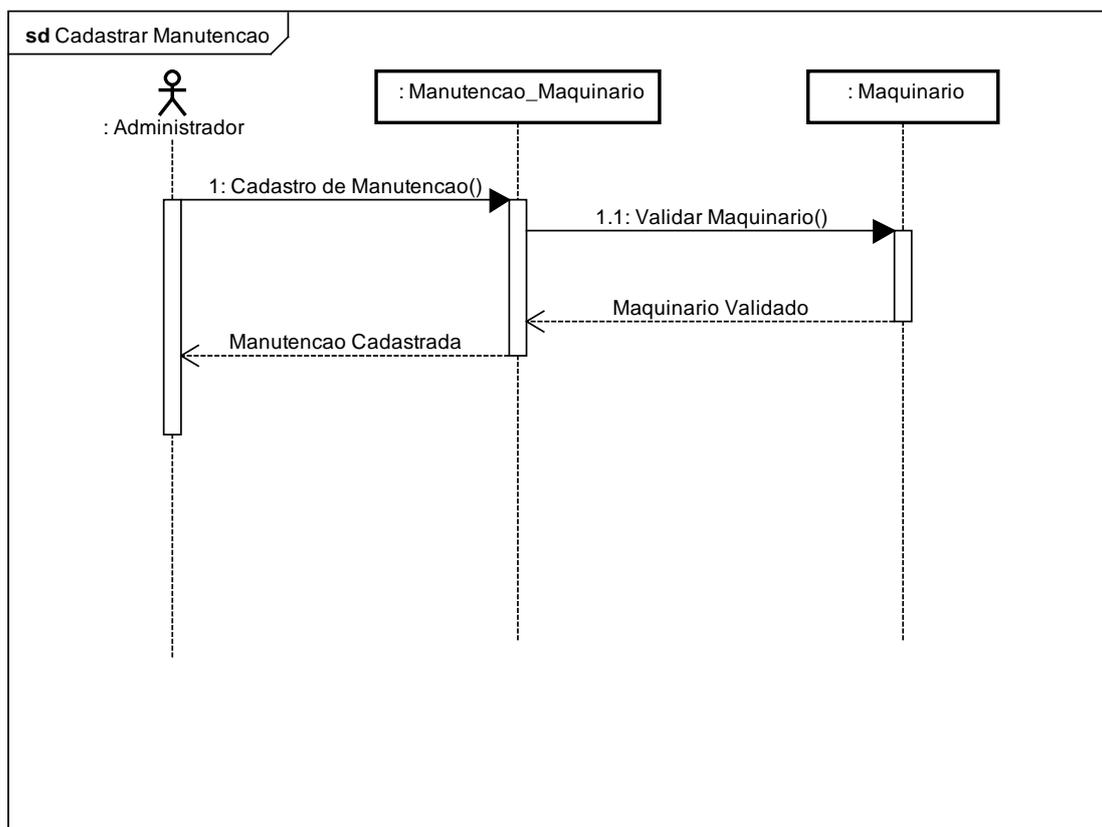


Figura 34 – Diagrama de Seqüência: DS 3 – Cadastrar Manutenção

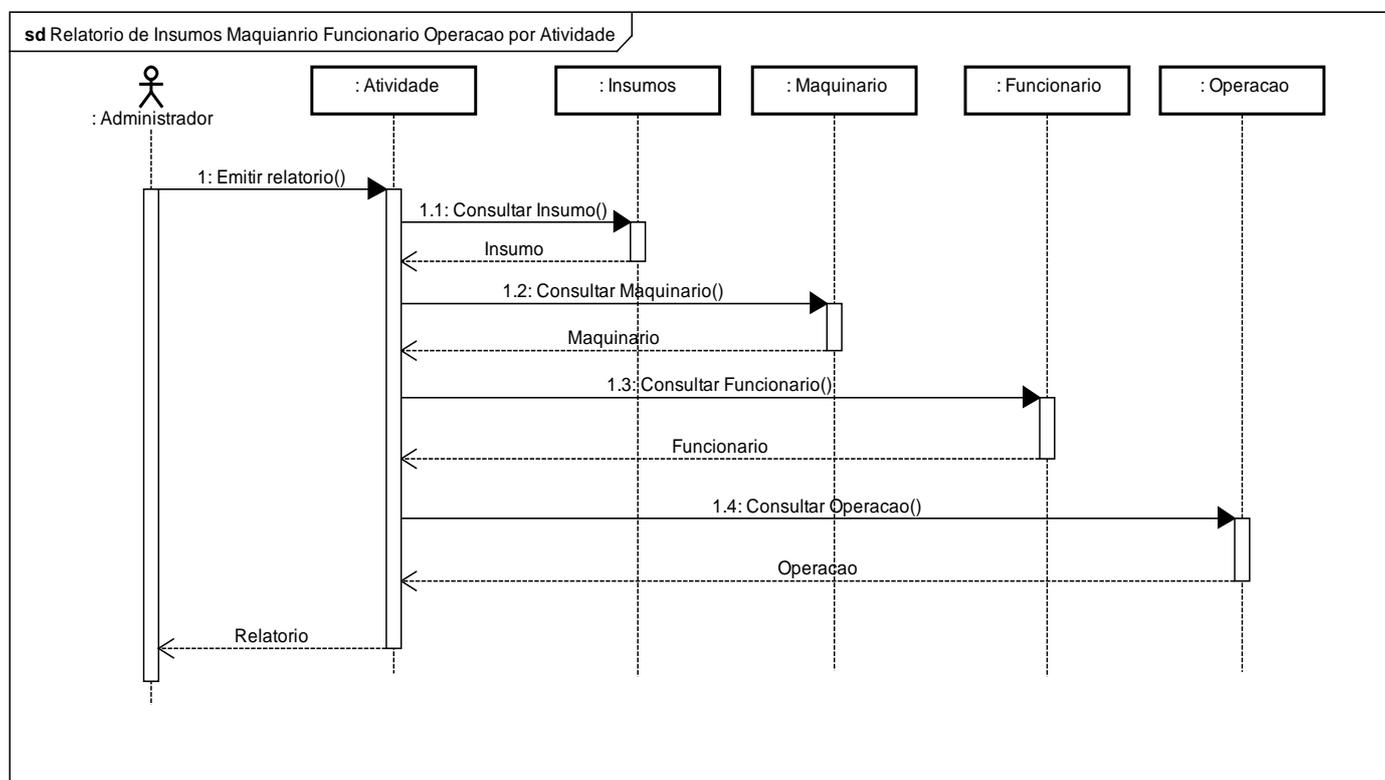


Figura 35 – Diagrama de Seqüência: DS 4 – Relatório de Insumos Maquinário Funcionário Operação por Atividade

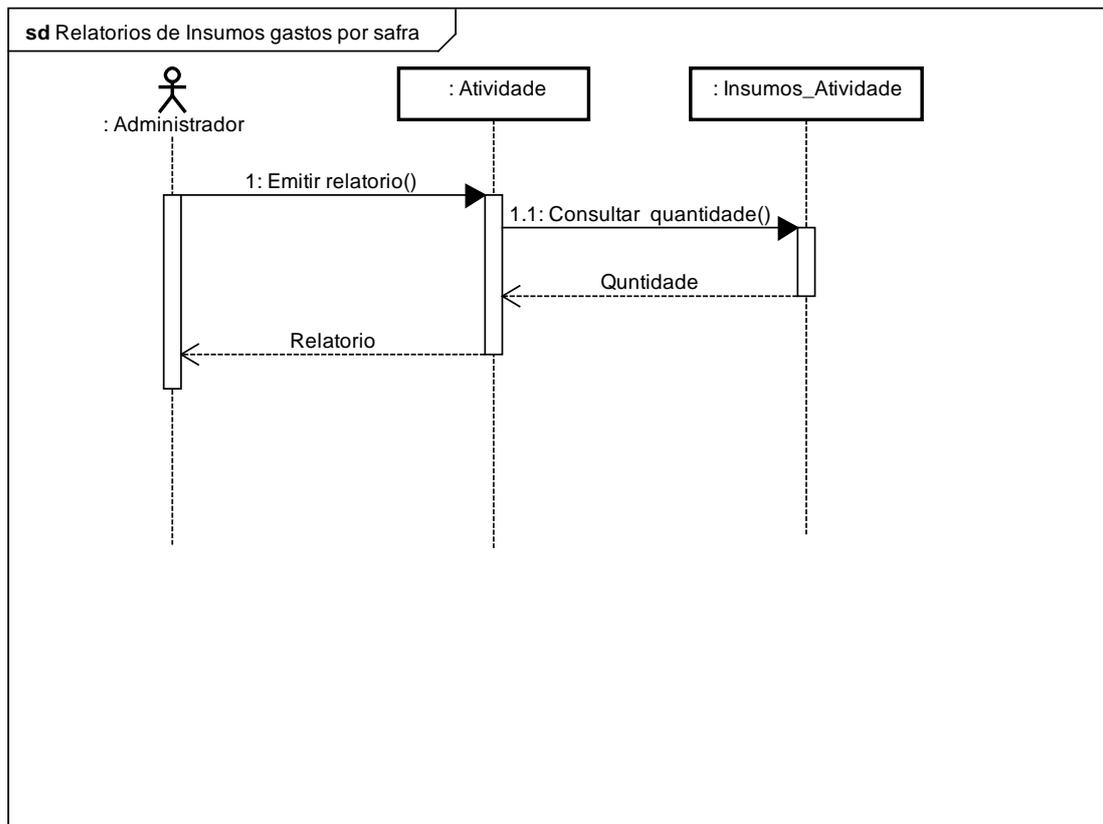


Figura 36 – Diagrama de Seqüência: DS 5 – Relatórios de Insumos gastos por safra

3.6 Diagrama de *Deploy*

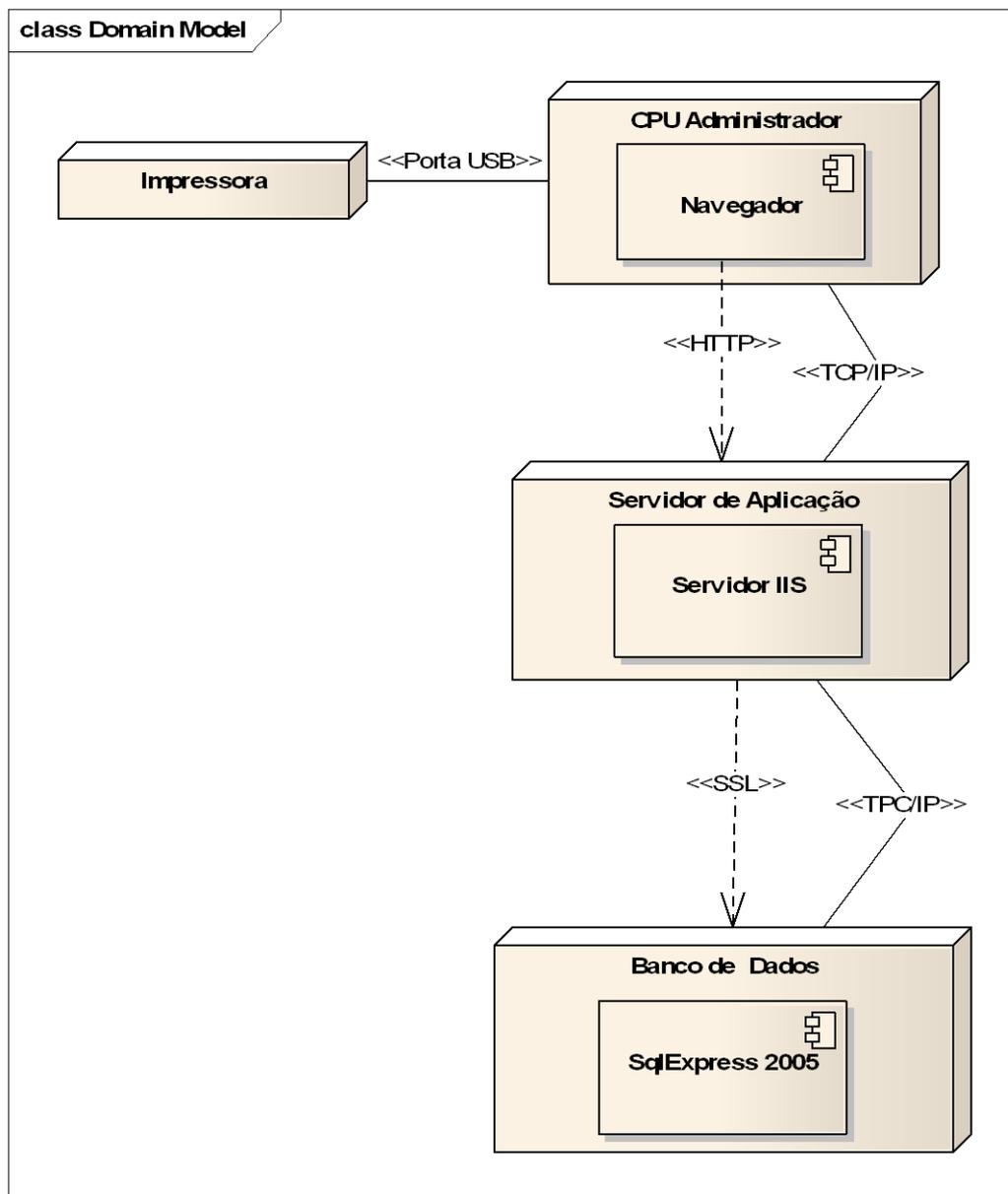


Figura 37 – Diagrama de Deploy.

3.7 Diagrama de Classes de Negócio

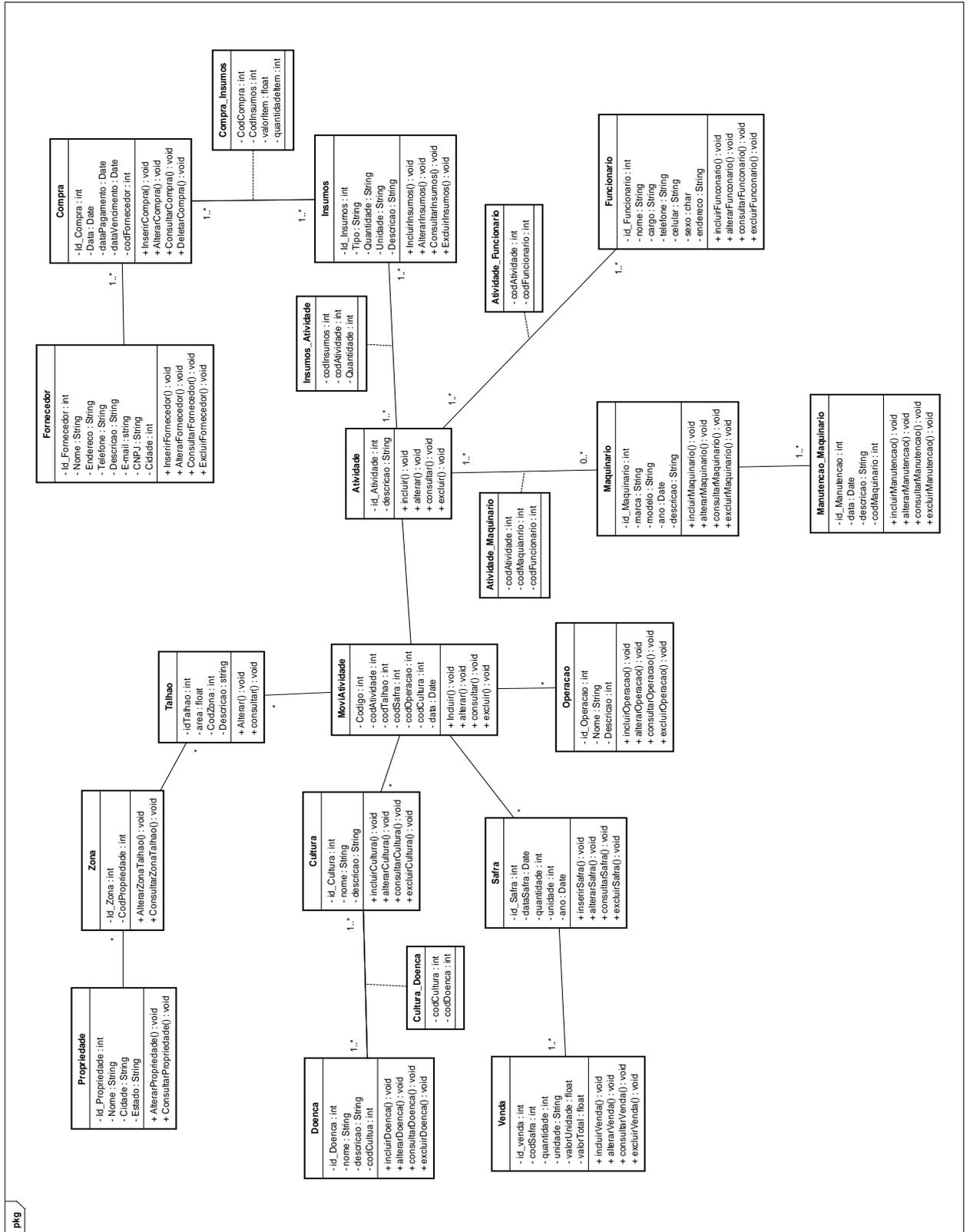


Figura 38 – Diagrama de Classes de Negócio

4. Implementação

Para a implementação da aplicação foi utilizado o framework adobe Flex com a linguagem *action script* e a plataforma .net com a linguagem *c#*. A biblioteca fluorinefx foi utilizada para a integração das duas tecnologias.

5.1 Organização do projeto na plataforma .net

Para o desenvolvimento na plataforma .net foi utilizada a IDE (*Integrated Development Environment*) VisualStudio 2008 ,e foi utilizada a técnica de desenvolvimento em três camadas: camada agrícola.MOR, camada agrícola.DAL e camada agrícola.BLL como mostra a figura abaixo:

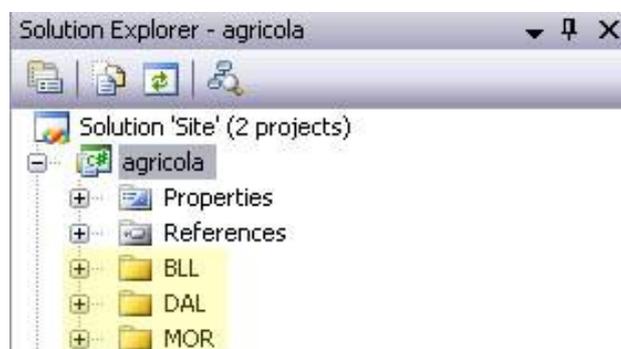


Figura 39 – Organização das Camadas .net.

Camada agrícola.MOR: é camada onde é criada as entidades que modelam o sistema.

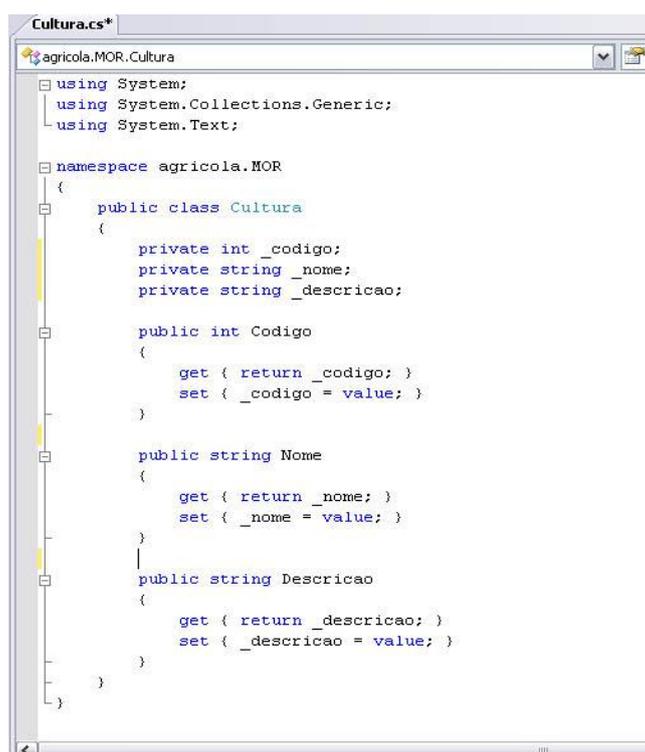


Figura 40 – Camada MOR.

Camada agrícola.DAL: é a camada onde são criados os métodos que fazem o relacionamento com banco de dados.



```

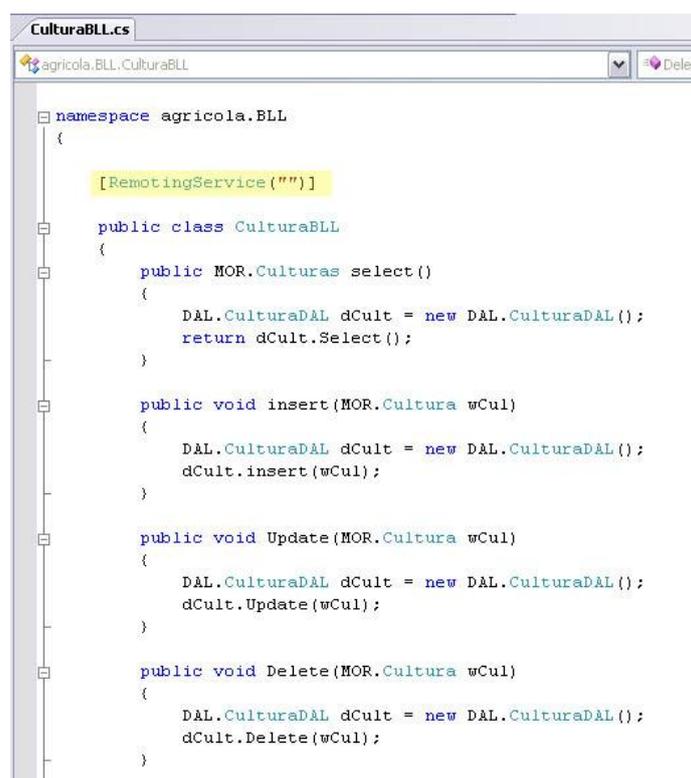
CulturaDAL.cs*
agricola.DAL.CulturaDAL
namespace agricola.DAL
{
    public class CulturaDAL
    {
        private string ConStr = @"Data Source=du;Initial Catalog=Agricola;User ID=sa;Password=123";

        public agricola.MOR.Culturas Select()
        {
            SqlConnection con = new SqlConnection(ConStr);
            con.Open();
            string SQL = "select *from Cultura";
            SqlCommand Cmd = new SqlCommand(SQL, con);
            SqlDataReader rd = Cmd.ExecuteReader(CommandBehavior.CloseConnection);
            MOR.Culturas wCultu = new MOR.Culturas();
            try
            {
                while (rd.Read())
                {
                    MOR.Cultura wCult = new MOR.Cultura();
                    wCult.Codigo = Convert.ToInt32(rd[0]);
                    wCult.Nome = rd[1].ToString();
                    wCult.Descricao = rd[2].ToString();
                    wCultu.Add(wCult);
                }
            }
            finally
            {
                rd.Close();
            }
            return wCultu;
        }
    }
}

```

Figura 41 – Camada DAL.

Camada agrícola.BLL: é a camada onde é criada a regra de negócio do sistema e é também a camada que deve ser referenciada como RemotingService para que possa ser acessada pelo objeto remoto do Flex:



```

CulturaBLL.cs
agricola.BLL.CulturaBLL
namespace agricola.BLL
{
    [RemotingService("")]
    public class CulturaBLL
    {
        public MOR.Culturas select()
        {
            DAL.CulturaDAL dCult = new DAL.CulturaDAL();
            return dCult.Select();
        }

        public void insert(MOR.Cultura wCul)
        {
            DAL.CulturaDAL dCult = new DAL.CulturaDAL();
            dCult.insert(wCul);
        }

        public void Update(MOR.Cultura wCul)
        {
            DAL.CulturaDAL dCult = new DAL.CulturaDAL();
            dCult.Update(wCul);
        }

        public void Delete(MOR.Cultura wCul)
        {
            DAL.CulturaDAL dCult = new DAL.CulturaDAL();
            dCult.Delete(wCul);
        }
    }
}

```

Figura 42 – Camada BLL.

5.2 Organização do projeto no Flex

Para o desenvolvimento em adobe Flex foi utilizada a IDE eclipse com o sdk do flex builder e foi organizado com os seguintes diretórios: src.agricola.mor, src.comp e src.img como mostra a Figura 43.

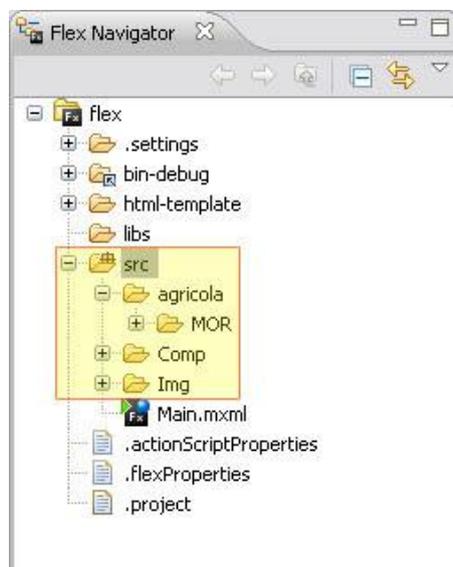


Figura 43 – Organização dos diretórios Flex

Diretório src.agricola.mor: é o diretório onde fica as entidades, que devem ser idênticas e referenciadas com as do c# como no exemplo abaixo.

```

1 package agricola.MOR
2 {
3     [RemoteClass(alias="agricola.MOR.Cultura")]
4     public class Cultura
5     {
6         private var _codigo:int;
7         private var _nome:String;
8         private var _descricao:String;
9
10        public function get Codigo():int { return _codigo; }
11        public function set Codigo(value:int):void { _codigo = value; }
12
13        public function get Nome():String { return _nome; }
14        public function set Nome(value:String):void { _nome = value; }
15
16        public function get Descricao():String { return _descricao; }
17        public function set Descricao(value:String):void { _descricao = value; }
18    }
19 }

```

Figura 44 – Referencia de Entidade.

Diretório src.comp: é o diretório onde foi criado os *MXML Component* que são dividido em dois modos, **Source** onde é feita a integração das plataformas através de *RemoteObjects*, que devem referenciar a camada superior do *c#* onde fica os métodos que vão ser acessados, para cada *RemoteObject* instanciado deve ser criado método *remoteResult* que irá trazer os resultados obtidos pelo método chamado no *c#* e um método *remoteFaut* que irá tratar as *Exceptions* se acaso der alguma falha no método chamado:

```

19     private var remote:RemoteObject;
20
21
22     private function create(event:FlexEvent):void
23     {
24         remote = new RemoteObject("fluorine");
25         remote.source = "agricola.BLL.CulturaBLL";
26         remote.addEventListener(ResultEvent.RESULT, remoteResult);
27         remote.addEventListener(FaultEvent.FAULT, remoteFaut);
28
29         btnInserir.addEventListener(MouseEvent.CLICK, clickIns);
30         btnUpdate.addEventListener(MouseEvent.CLICK, clickUp);
31         btnEdit.addEventListener(MouseEvent.CLICK, clickEdit);
32         btnDelete.addEventListener(MouseEvent.CLICK, clickDel);
33         btnNovo.addEventListener(MouseEvent.CLICK, clickNovo);
34         txtCodigo.enabled = false;
35         btnInserir.enabled = false;
36
37         var cultura:Cultura = new Cultura();
38
39         remote.select();
40     }
41

```

Figura 45 – Modo Source.

E também o modo **design** onde é montado layout do componente:

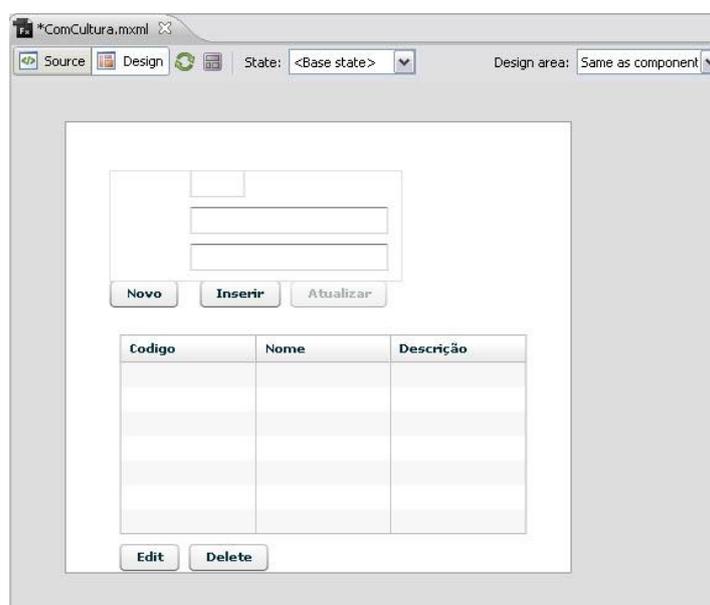


Figura 46– Modo design.

Diretório src.img: é o diretório onde fica as imagens utilizadas na aplicação.

5.3 Integração das tecnologias utilizando o FluorineFX

Inicialmente para integrar as duas tecnologias deve-se ter instalado corretamente o FluorineFX e o servidor IIS (Internet Information Services). Um novo web site do tipo FluorineFX ASP.NET Web Site deve ser criado setando o diretório onde está o projeto flex .

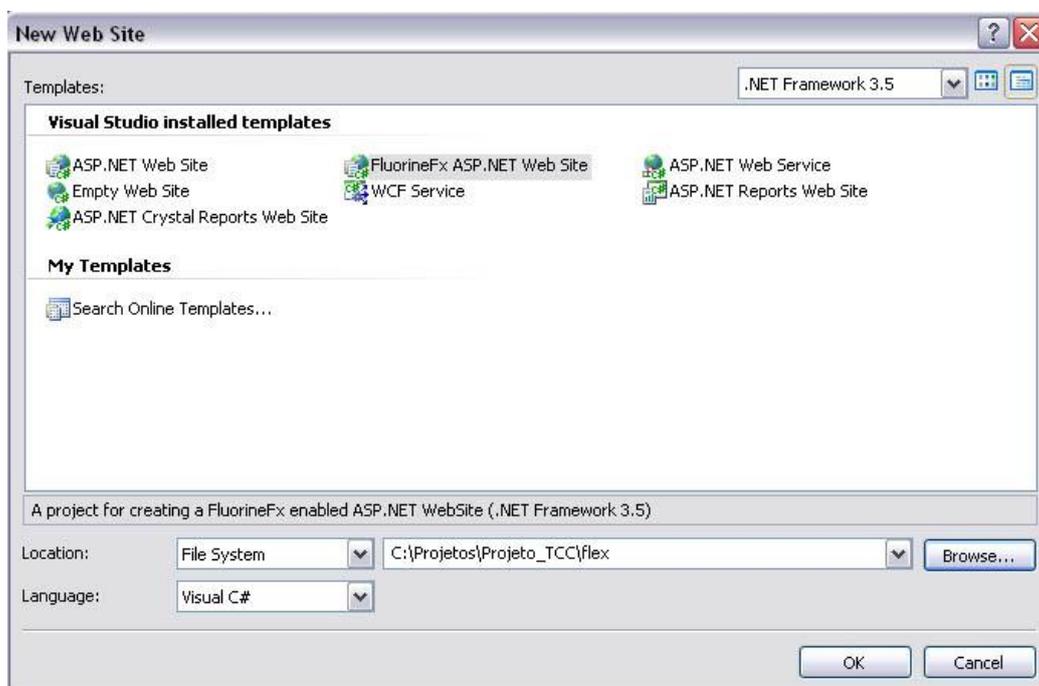


Figura 47– FluorineFX ASP.NET Web Site.

Na pasta BIN do projeto criado referencie a DLL do projeto agrícola mostrado anteriormente:

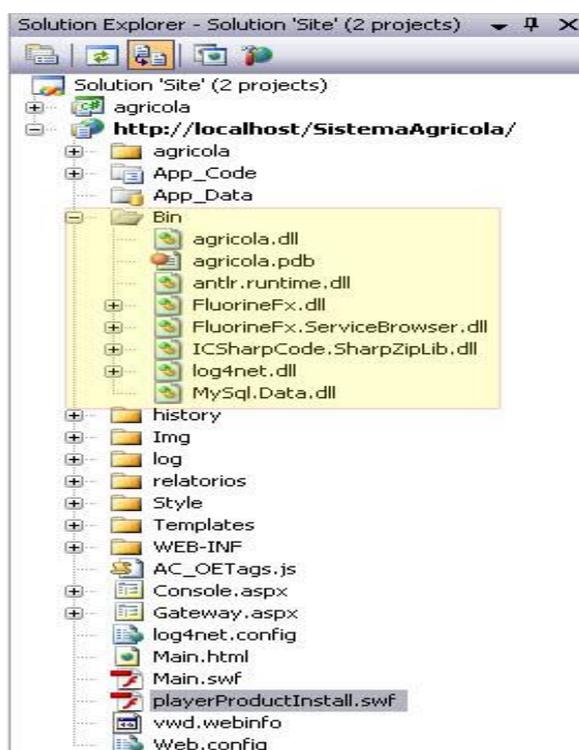


Figura 48– Referencia Agrícola.

E finalmente adicione o projeto no servidor IIS:

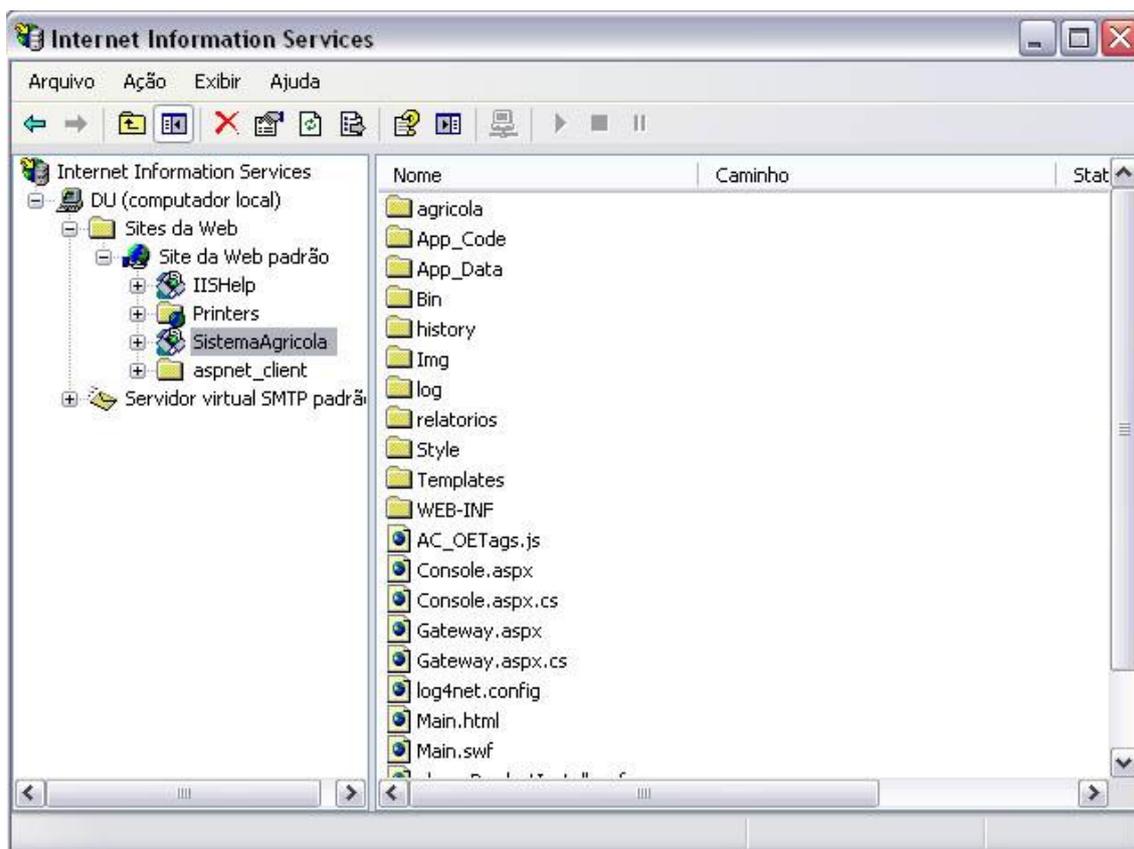


Figura 49– Servidor IIS.

Fazendo corretamente isto e as referencias citadas neste capitulo o FluorineFX conseguirá transitar dados através de objetos remotos entre as duas linguagem em tempo real.

4.3 Interfaces do sistema

Ao acessar o sistema abrirá a pagina home e o menu com quatro itens:



Figura 50– Pagina Home.

Ao seleccionar a opção cadastros do menu abrirá a pagina com os cadastros e um submenu com 10 itens onde cada item é relacionado a um cadastro:

Sistema de Gestão Agrícola

HOME | OPERAÇÃO | CULTURA | FUNCIONARIO | INSUMO | PROPRIEDADE | ZONA | TALHÃO | MAQUINARIO | SAFRA | ATIVIDADE

Cadastro de Atividade

Código

Descrição

Novo

Descrições | Maquinario | Funcionario | Insumo

Operação

Cultura

Safrá

Talhão

Data

Salvar

| Talhão | Safrá | Operação | Cultura | Data |
|--------|-------|-----------|---------|------------|
| 1 | 2000 | Plantio | Milho | 16/11/2009 |
| 2 | 2000 | Plantio | Soja | 16/11/2009 |
| 3 | 2000 | Colheitas | Milho | 16/11/2009 |
| 4 | 2000 | Colheitas | Milho | 16/11/2009 |

Deletar

Figura 51– Pagina de Cadastros.

Ao seleccionar a opção relatórios do menu abrirá a pagina com as opções de relatórios:

Sistema de Gestão Agrícola

HOME | OPERAÇÃO | CULTURA | FUNCIONARIO | INSUMO | PROPRIEDADE | ZONA | TALHÃO | MAQUINARIO | SAFRA | ATIVIDADE

HOME

CADASTRO

RELATÓRIOS

TALHÕES

Relatório de Funcionários

Relatório de Safrá

Relatório de Insumos Gastos por Safrá

Relatório Cultura por Talhão

1

Relatório

Relatório Produtividade Por safrá

2000

Relatório

Relatório Insumo Maquinario Fu...

24/11/2009

Relatório

Figura 52– Pagina de Relatórios.

E ao selecionar a opção talhões no menu abrirá a pagina de visualização que conterà vários pontos, onde cada um representa um talhão que ao ser selecionado abrirá um popup com as informações da ultima atividade cadastrada no determinado talhão:



The screenshot displays the 'Sistema de Gestão Agrícola' interface. On the left, a navigation menu includes 'HOME', 'CADASTRO', 'RELATÓRIOS', and 'TALHÕES'. The main area shows a satellite map of agricultural plots with several green pushpin markers. A popup window is open, showing the following information:

Código do talhão: 1
Operação: Plantio
Cultura: Soja
Safra: 2009
Data: 24/11/2009

Maquinarios

| |
|-------|
| tc 57 |
| tc 44 |

Funcionario

| |
|------------------|
| Eduardo Henrique |
| Maria Josefa |
| Josedei |

Insumo

| Insumo | Quantidade |
|---------------|------------|
| Veneno | 20 |
| Semnete Milho | 1 |

Figura 53– Pagina de Visualização de Talhões.

5. Conclusão

Embora o sistema não esteja finalizado, as funcionalidades desenvolvidas até o momento dão um diferencial comparado aos softwares disponíveis no mercado. O gerenciamento das atividades permite ao usuário uma visão ampla de tudo que vem sendo desenvolvido em sua propriedade. Uma outra característica importante no sistema é a interface gráfica que disponibiliza uma melhor interação entre o usuário e a aplicação, além de possibilitar uma visão de sua propriedade organizada em talhões.

A etapa de análise facilitou o entendimento do sistema. Por meio da lista de eventos e os casos de uso foi possível identificar as principais funcionalidades do sistema e as inter relações. O diagrama de sequência facilitou a compreensão do funcionamento do software enquanto que o diagrama de classes orientou o desenvolvimento da implementação.

No desenvolvimento deste projeto existiram muitas dificuldades, pelo fato de estar trabalhando com a integração de duas tecnologias (asp.net e adobe flex utilizando fluorineFX).

Grande parte do tempo de desenvolvimento deste projeto foi gasto no estudo das tecnologia utilizadas pois são recentes no mercado e não existe muita literatura voltada a integração destas duas linguagens. Por este motivo algumas partes do sistema não foram desenvolvidas, mas futuramente, em uma nova etapa dar-se-á continuidade ao software e a implementação de todas as necessidades que foram levantadas na etapa de análise.

O desenvolvimento deste trabalho foi de suma importância para o crescimento pessoal e profissional visto que as tecnologias utilizadas são as mesmas empregadas na empresa onde estou trabalhando atualmente.

6. Referencias Bibliográficas

DURÃES, Ramon. **Desenvolvimento para Web usando o Visual Studio 2008** / Rio de Janeiro : Brasport, 2008.

RUMBAUGH, James. **Modelagem e projetos baseados em objetos com UML2**; Tradução Daniel Vieira / Rio de Janeiro : Elsevier, 2006.

LARMAN, Craig. **Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos** / Craig Larman; trad. Luiz A. Meirelles Salgado. - Porto Alegre: Bookman, 2000.

SHEPHERD, George. **Microsoft ASP .NET 2.0: passo a passo** / George Shepherd; tradução Cláudio Belleza Dias. - Porto Alegre: Bookman, 2007.

SCHMITZ, Daniel Pace. **Adobe Flex Builder 3.0: Conceitos e Exemplos** / Daniel Pace Schmitz .- Rio de Janeiro: Brasport, 2008.

ADOBE. **Flex overview**. Visão geral do Flex: 30 mar. 2007. Disponível em: <<http://www.adobe.com/products/flex/overview/>>. Acesso em: 7 outubro de 2009.