**Faculdade Vale do Salgado**

**Curso de Análise e Desenvolvimento de Sistemas**

**Disciplina:** Projeto Integrador III

**Professor:** Farley Duarte Gurgel

**Semestre:** 2015-1

Documento de Especificação de Projeto Integrador III

**Histórico de Revisão**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 01/03/2015  02/03/2015  04/03/2015  05/03/2015  05/03/2015  05/03/2015  06/03/2015  09/03/2015  10/03/2015  15/03/2015 | 1.0  1.1  1.2  1.3  1.4  1.5  1.6  1.7  1.8  1.9 | Criação inicial do documento  Tema do projeto  Objetivo do projeto  Método de trabalho  Título do projeto  Equipe  Regras de negócios  Requisitos Funcionais  Requisitos não funcionais  Documento de visão | Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva |
| 10/04/2015  10/04/2015  18/04/2015  19/04/2015  20/04/2015  28/04/2015  28/04/2015 | 1.9  2.0  2.1  2.2  2.3  2.4  2.5 | Adicionados Índices e Diagramas  Iniciando os diagramas de sequencia  Diagrama de sequencia  Iniciando o diagrama de classe  Diagrama de classe  Iniciando os diagramas de atividade  Diagrama de atividade | Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva |
| 05/04/2015  06/04/2015  06/05/2015  08/05/2015  20/05/2015 | 2.6  2.7  2.8  2.9  3.0 | Finalizando Diagramas  Iniciando os casos de testes  Caso de teste login  Finalizando os casos de testes  Caso de teste Gerenciamento de Registros | Yago Silva  Yago Silva  Yago Silva  Yago Silva  Yago Silva |

.

Índice Detalhado

[1 Descrição Geral 6](#_Toc428465959)

[1.1 Tema do Projeto 6](#_Toc428465960)

[1.2 Objetivo do Projeto 6](#_Toc428465961)

[1.3 Método de Trabalho 6](#_Toc428465962)

[1.4 Título Projeto e Nome do produto 6](#_Toc428465963)

[1.5 Equipe 6](#_Toc428465964)

[1.6 Regras de Negócio 7](#_Toc428465965)

[2 Plano de Projeto 8](#_Toc428465966)

[2.1 Requisitos Funcionais 8](#_Toc428465967)

[Cadastro Individual 11](#_Toc428465968)

[Cadastro fichas de visitas domiciliar 12](#_Toc428465969)

[Agenda 13](#_Toc428465970)

[2.2 Requisitos Não Funcionais 14](#_Toc428465971)

[2.3 Diagrama de Caso de Uso 16](#_Toc428465972)

[3 Documento de Visão 17](#_Toc428465973)

[3.1 Protótipo (Ícone do Sistema) 17](#_Toc428465974)

[3.2 Protótipo (Tela de Login) 18](#_Toc428465975)

[3.3 Protótipo (Tela de Cadastro de Usuário) 19](#_Toc428465976)

[3.4 Protótipo (Tela de Menu Principal) 20](#_Toc428465977)

[3.5 Protótipo (Tela de Cadastro Domiciliar) 21](#_Toc428465978)

[3.6 Protótipo (Tela de Cadastro Individual) 22](#_Toc428465979)

[3.7 Protótipo (Tela de Cadastro de Visitas) 23](#_Toc428465980)

[3.8 Protótipo (Tela de Agenda) 24](#_Toc428465981)

[3.9 Diagrama de Sequência 25](#_Toc428465982)

[3.10 Diagrama de Sequência Realizar Login 25](#_Toc428465983)

[3.11 Diagrama de Sequência Manter de Usuário 26](#_Toc428465984)

[3.12 Diagrama de Sequência Cadastros Domiciliar 27](#_Toc428465985)

[3.13 Diagrama de Sequência Cadastros Individual 28](#_Toc428465986)

[3.14 Diagrama de Sequência Cadastros Visitas 29](#_Toc428465987)

[3.15 Diagrama de Sequência Manter Agendamento 30](#_Toc428465988)

[3.16 Diagrama de Sequência Manter Agendamento Alterar 31](#_Toc428465989)

[3.17 Diagrama de Sequência Manter Agendamento Consultar 32](#_Toc428465990)

[3.18 Diagrama de Classes do Sistema 33](#_Toc428465991)

[3.19 Diagrama de Atividades Login 34](#_Toc428465992)

[3.20 Diagrama de Atividade Realizar Login 35](#_Toc428465993)

[3.21 Diagrama de Atividades Manter Usuário 36](#_Toc428465994)

[3.21.1 Diagrama de Atividade Realizar Cadastro Domiciliar 37](#_Toc428465995)

[3.21.2 Diagrama de Atividade Realizar Cadastro Individual 38](#_Toc428465997)

[3.21.3 Diagrama de Atividade Realizar Cadastro Visitas 39](#_Toc428465998)

[3.21.4 Diagrama de Atividades Manter Agendamento 40](#_Toc428465999)

[3.22 Diagrama de Atividades Consultar Agendamento Realizado 41](#_Toc428466000)

[3.23 Diagrama de Atividades Atualizar Agendamento Realizado 42](#_Toc428466001)

[3.24 Diagrama de Atividades Excluir Agendamento Realizado 43](#_Toc428466002)

[4 Planejamento de Testes 44](#_Toc428466003)

[4.1 Plano de Testes 44](#_Toc428466004)

[CT01-Autenticar Login (Positivo) 44](#_Toc428466005)

# Descrição Geral

## Tema do Projeto

A equipe analisou a facilidade de uso de se criar um aplicativo de coleta de dados que será desenvolvido para dispositivos moveis, a fim de gerar recomendações de projeto para fazer a sua interface amigável e fácil de se usar e adequada para os usuário.

## Objetivo do Projeto

Objetivo do projeto é desenvolver um software mobile que vai integrar as informações recolhidas pelo o agente comunitário de saúde (ACS) que serão repassadas para o órgão (prefeitura) responsável de enviar os dados para o Ministério da Saúde (MS). Com intuito de melhorias e de inovação nesta área, melhorando a iteração dos usuários com a tecnologia e seus recursos, minimizando papeladas e tempo de serviço e processo do agente.

## Método de Trabalho

O projeto será desenvolvido em metodologia ágeis, que é baseado na noção de desenvolvimento e entrega incremental de processo de software. A modelagem do sistema é em UML (orientada a objeto, estruturada, outras).

## Título Projeto e Nome do produto

Sipacs – Sistema Integrado de Pesquisa para Agente Comunitário de Saúde.

## Equipe

201317257- Yago Silva Cavalcante

20137232 - Jean Pereira Batista

## Regras de Negócio

|  |  |  |
| --- | --- | --- |
| Id | Nome | Descrição |
| RN01 | USUARIO | O USUARIO TERA QUE CADASTRAR UM USUARIO PARA ACESSAR O SISTEMA. |
| RN02 | SISTEMA E USUARIO | O USUARIO NÃO PODERÁ DA SUAS INFORMAÇÕES DE ACESSO PARA OUTRAS PESSOAS. |
| RN03 | INSTALAÇÃO, SISTEMA | O APK DO SISTEMA SO PODERÁ SER INSTALADO EM SMARTPHONES ANDROID E QUE SEJA DO AGENTE COMUNITARIO DE SAUDE. |
| RN04 | CADASTRO E USUARIO | O USUARIO DEVERÁ CADASTRAR TODOS OS CAMPOS DO SISTEMA SE NECESSARIO. |
| RN05 | TEMPO DO SISTEMA | O SISTEMA ESTARÁ DISPONIVEL AO USUARIO 24 HORAS POR DIAS PARA SEU TRABALHO. |
| RN06 | SENHA | A SENHA DO USUARIO SO PODERÁ TER NO MAXIMO 10 CARACTERES INCLUINDO A DO USUARIO POR FORMA DE SEGURANÇA. |
| RN07 | REDIGITAÇÃO DE CADASTRO | O USUARIO NÃO PODERÁ CRIAR DOIS CADASTROS IGUAIS, MAIS PODERÁ DELETAR O CADASTRO OU ATUALIZA-LO SEJA ELE QUAL FOR. |
| RN08 | GERAR CODIGO | USUARIO É OBRIGADO A GERAR UM CODIGO SEMPRE QUE FOR CADASTRA NOVAS INFORMAÇÕES NO SISTEMA. |

# Plano de Projeto

## Requisitos Funcionais

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação | Nome | |  |
| RF01 | Efetuar Login | | |
| Descrição | | | |
| O sistema deverá Logar o usuário autenticado para ter acesso as demais funcionalidades do sistema. | | | |
| Entradas e pré-condições | | | |
| O usuário deve possuir login e senha válidos. | | | |
| Saídas e pós-condições: | | | |
| O sistema mostra a tela com todas as funcionalidades. | | | |
| Fluxo principal | | | |
| 1. O usuário indica o login; 2. O usuário indica a senha; 3. O usuário clica em [Entrar] 4. O sistema verifica se o usuário existe; 5. O sistema verifica se a senha é válida; 6. O sistema mostra a tela de acesso às funcionalidades; | | | |
| Fluxos alternativos | | | |
| [FA 001] Login inexistente;  [FA 002] Senha incorreta; | | | |
| Prioridade: | | Essencial. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação | Nome | |  |
| RF02 | Cadastrar Usuário | | |
| Descrição | | | |
| O usuário deverá solicitar seu cadastro no sistema, para ter acesso as outras funcionalidades que o sistema oferece. | | | |
| Entradas e pré-condições | | | |
| O usuário deverá se cadastrar no sistema com usuário e senha. | | | |
| Saídas e pós-condições: | | | |
| O sistema mostra a tela de cadastro para o usuário. | | | |
| Fluxo principal | | | |
| 1. O usuário indica o login; 2. O usuário indica a senha; 3. O usuário clica em [Salvar] 4. O sistema verifica se o usuário existe; 5. O sistema verifica se a senha é válida; 6. O sistema retorna a tela de Login; | | | |
| Fluxos alternativos | | | |
| [FA 001] Cadastrado com sucesso. | | | |
| Prioridade: | | Essencial. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação | Nome | |  |
| RF03 | Cadastro domiciliar | | |
| Descrição | | | |
| O sistema provê meios de gerenciar o cadastro dos dados do domicilio. Há funções para cadastrar, editar os dados do domicilio, O sistema armazena os dados do domicilio para cada cadastro; | | | |
| Entradas e pré-condições | | | |
| O usuário deverá cadastrar todos os campos da tela de cadastro domiciliar corretamente, podendo corrigir campos, e armazenando o cadastro. | | | |
| Saídas e pós-condições: | | | |
| O sistema exibi a tela de Cadastro domiciliar com todos os campos para ser preenchidos. | | | |
| Fluxo principal | | | |
| 1. O usuário deverá cadastrar todos dados corretamente; 2. O usuário não poderá deixar campos vazios; 3. O usuário clica em [Salvar]; 4. O sistema informa uma mensagem ao usuário: “Cadastro realizado com sucesso!”; 5. O sistema verifica se os dados foram cadastro corretamente; 6. O usuário poderá editar o cadastro; 7. O usuário clica em [Atualizar]; 8. O usuário clica em [Listar]; 9. O sistema informa uma mensagem ao usuário: “Cadastro atualizado com sucesso!”; 10. O usuário poderá alterar campos do cadastro. | | | |
| Fluxos alternativos | | | |
| [FA 001] Salva.  [FA 002] Editar. | | | |
| Prioridade: | | Importante. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação | Nome | |  |
| RF03 | Cadastro Individual | | |
| Descrição | | | |
| O sistema provê meios de gerenciar o cadastro dos dados do individual. Há funções para cadastrar, editar os dados do individual, O sistema armazena os dados do individual para cada cadastro; | | | |
| Entradas e pré-condições | | | |
| O usuário deverá cadastrar todos os campos da tela de cadastro individual corretamente, podendo corrigir campos, e armazenando o cadastro. | | | |
| Saídas e pós-condições: | | | |
| O sistema exibi a tela de Cadastro individual com todos os campos para ser preenchidos. | | | |
| Fluxo principal | | | |
| 1. O usuário deverá cadastrar todos dados corretamente; 2. O usuário não poderá deixar campos vazios; 3. O usuário clica em [Salvar]; 4. O sistema informa uma mensagem ao usuário: “Cadastro realizado com sucesso!”; 5. O sistema verifica se os dados foram cadastro corretamente; 6. O usuário poderá editar o cadastro; 7. O usuário clica em [Editar]; 8. O sistema informa uma mensagem ao usuário: “Cadastro editado!”; 9. O usuário poderá alterar campos do cadastro. | | | |
| Fluxos alternativos | | | |
| [FA 001] Salva.  [FA 002] Editar. | | | |
| Prioridade: | | Importante. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação | Nome | |  |
| RF04 | Cadastro fichas de visitas domiciliar | | |
| Descrição | | | |
| O sistema provê meios de gerenciar o cadastro dos dados do visitas. Há funções para cadastrar, editar os dados do visitas, O sistema armazena os dados do visitas para cada cadastro; | | | |
| Entradas e pré-condições | | | |
| O usuário deverá cadastrar todos os campos da tela de cadastro visitas corretamente, podendo corrigir campos, e armazenando o cadastro. | | | |
| Saídas e pós-condições: | | | |
| O sistema exibi a tela de Cadastro visitas com todos os campos para ser preenchidos. | | | |
| Fluxo principal | | | |
| 1. O usuário deverá cadastrar todos dados corretamente; 2. O usuário não poderá deixar campos vazios; 3. O usuário clica em [Salvar]; 4. O sistema informa uma mensagem ao usuário: “Cadastro realizado com sucesso!”; 5. O sistema verifica se os dados foram cadastro corretamente; 6. O usuário poderá editar o cadastro; 7. O usuário clica em [Editar]; 8. O sistema informa uma mensagem ao usuário: “Cadastro editado!”; 9. O usuário poderá alterar campos do cadastro. | | | |
| Fluxos alternativos | | | |
| [FA 001] Salva  [FA 002] Editar | | | |
| Prioridade: | | Importante. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação | Nome | |  |
| RF05 | Agenda | | |
| Descrição | | | |
| O usuário poderá solicitar um agendamento de visitas se necessário, no sistema com campos para preenchimento como nome, endereço e número, podendo ver seus agendamento salvando, atualizando e excluindo. | | | |
| Entradas e pré-condições | | | |
| O usuário poderá agendar visitas domiciliar, salvando, atualizando e excluindo os agendamento armazenados. | | | |
| Saídas e pós-condições: | | | |
| O sistema exibirá a tela de agenda para preencher os campos e podendo ver os agendamento. | | | |
| Fluxo principal | | | |
| 1. O usuário deverá preencher os campos para agendar as visitas domiciliares; 2. O usuário não poderá deixar campos vazios; 3. O usuário deverá salvar o agendamento; 4. O usuário clica em [Salvar]; 5. O sistema informa uma mensagem ao usuário: “Agendamento salvo!” 6. O usuário poderá atualizar o agendamento; 7. O usuário clica em [Atualizar]; 8. O sistema informa uma mensagem ao usuário: “Agendamento atualizado!” 9. O usuário poderá excluir o agendamento; 10. O usuário clica em [Excluir]; 11. O sistema informa uma mensagem ao usuário: “Agendamento excluído!” 12. O usuário poderá entrar na tela de Agenda e ver seus agendamentos salvos; 13. O sistema exibir a tela de acesso ao agendamento ao usuário; | | | |
| Fluxos alternativos | | | |
| [FA 001] Salva  [FA 002] Atualizar  [FA 003] Excluir | | | |
| Prioridade: | | Essencial. | |

## Requisitos Não Funcionais

|  |  |
| --- | --- |
| [RNF 01] | Interface Amigável |
| Descrição | |
| Considerando que os operadores do sistema geralmente não possuem intimidade com aplicações, a interface gráfica deverá considerar aspectos de usabilidade. Desenvolveremos a GUI utilizando a API de Android com Java. O objetivo é utilizar aspectos que beneficiem a navegabilidade do usuário no sistema, como disposição intuitiva e hierárquica dos menus. Uma boa estimativa de tempo para o usuário aprender a utilizar o sistema é que seu aprendizado dure, no máximo, 25 minutos. | |
| Prioridade | Importante. |

|  |  |
| --- | --- |
| [RNF 03] | Linguagens de programação |
| Descrição | |
| O Sipacs utilizará Androidpara implementação dos códigos do Sistema. Essa tecnologia permitirá o desenvolvimento de um sistema simples, além de suportar a programação multi-plataforma. | |
| Prioridade | Desejável. |

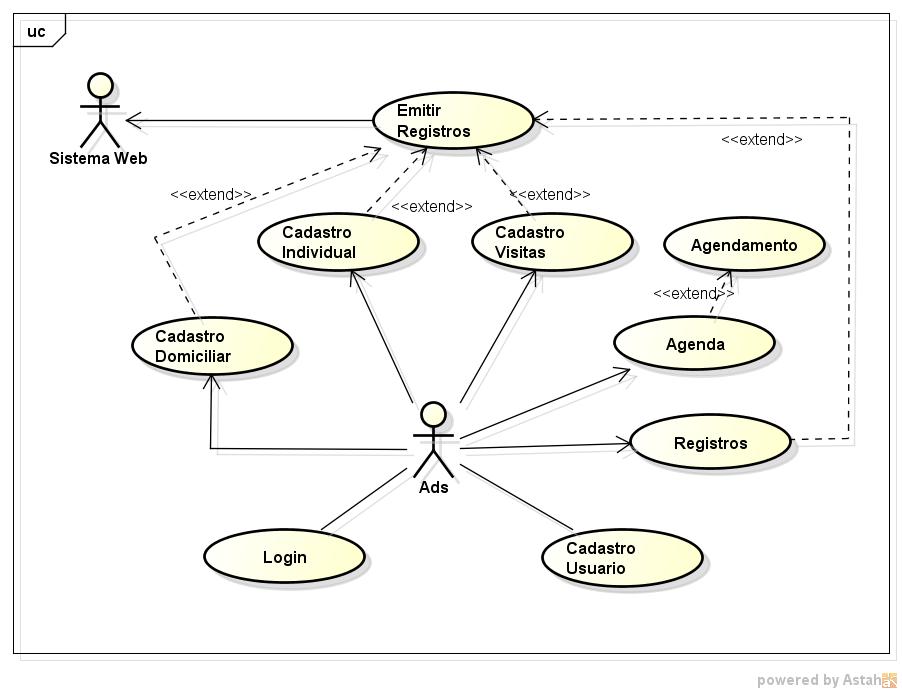
|  |  |
| --- | --- |
| [RNF 04] | Ferramentas |
| Descrição | |
| Android Studio: Um IDE para desenvolvimento de aplicações Android;  Astah Community: Ferramenta CASE UML utilizada para a modelagem dos casos de uso, de diagramas de classes e outros aspectos do projeto.  Microsoft Office: Sua utilização será como editor de material auxiliar, i. e. documentação,  Notepad++: No desenvolvimento do site do projeto; | |
| Prioridade | Desejável. |

|  |  |
| --- | --- |
| [RNF 05] | Flexibilidade do Sistema |
| Descrição | |
| O SIPACS deverá ser bem fácil de ser manipulado pelo o usuário. Facilidade de aprendizado, facilidade em memorização, maior produtividade, redução de erros e satisfação subjetiva do usuário. | |
| Prioridade | Importante. |

|  |  |
| --- | --- |
| [RNF 07] | Usabilidade |
| Descrição | |
| O sistema deverá alcançar objetivos específicos com efetividade, eficiência e satisfação num contexto especifico de uso. | |
| Prioridade | Importante. |

|  |  |
| --- | --- |
| [RNF 07] | **Segurança** |
| Descrição | |
| O sistema deverá estar disponível 24h por dia durante os 7 dias da semana. (Disponibilidade, Integridade – Precisão X Completude) e Confidencialidade. | |
| Prioridade | Importante. |

## Diagrama de Caso de Uso



1. Caso de uso Sipacs
2. O caso de uso que estar na figura a cima mostra os relacionamentos do autor com as classe que são essenciais para o sistema, e para servir para o diagrama de classe para o desenvolvimento dos diagramas seguintes.
3. Atores envolvidos usuário que é o agente comunitário de saúde e um sistema externo que vai receber o dados capturado pelo o sistema Sipacs.
4. A pré-condições é que o sistema Sipacs deverá estar acessível para o usuário 24 horas.
5. O fluxo principal do sistema são os dados colhidos pelo o usuário que vai operar o sistema.
6. Pós-condições o sistema deverá fornecer todas as funções e telas para o usuário operar.

# Documento de Visão

## Protótipo (Ícone do Sistema)



1. Objetivo da tela é mostrar as funcionalidades do ícone que dá acesso para o usuário no sistema Sipacs.
2. O ícone vai estar na tela de trabalho do dispositivo mobile que estará instalado em cada aparelho dos usuários.
3. O usuário vai clicar no ícone para ter acesso as funcionalidades do sistema Sipacs.
4. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.

## Protótipo (Tela de Login)



1. Objetivo da tela é mostrar as funcionalidades do login com o usuário e como essa funções vão se comportar se o usuário estar autenticado ou não autenticado, informando ao usuário que estar operar o sistema.
2. Estar tela do login é ativado quando iniciado o sistema, a próxima tela a ser chamada é a tela do meu principal com suas funcionalidades principais mas só se o usuário estiver autenticado no sistema.
3. O usuário é obrigado estar cadastrado no sistema para ter acesso as próximas funcionalidade.
4. O domínio do sistema na tela de login é dois campos um vai receber o usuário que vai ter tipo String, e o outro campo da senha vai possuir um tipo Password com botões.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de login vai possuir dois campos obrigatórios que são “Usuário” e “Senha”.

## Protótipo (Tela de Cadastro de Usuário)



1. Objetivo da tela cadastro de usuário é mostrar as funcionalidades do cadastro que vai da acesso na telas do login, para o usuário estar autenticado e ter acesso a tela de Menu.
2. Estar tela do cadastro de usuário é ativado quando o usuário não tem acesso ao sistema e ele deve se cadastrar para ele ter acesso a próxima tela que é a tela de menu principal, a próxima tela a ser chamada é tela de menu principal que o usuário terá acesso com suas funcionalidades.
3. O usuário é obrigado fazer o cadastro para ter acesso a tela de menu.
4. O domínio do sistema na tela de cadastro é autenticação para o acesso principal do sistema.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de cadastro de usuário vai ter dois campos principais para seu cadastro e autenticação que será realizada com sucesso voltando para a tela de login.

## Protótipo (Tela de Menu Principal)



1. Objetivo da tela é mostrar as funcionalidades do menu principal que vai da acesso as telas de cadastros, agenda e dos registro que receberá os registrados que foram cadastros pelo o usuário mostrando suas funções e operações essenciais.
2. Estar tela do menu é ativado quando o usuário tem acesso e quando ele estar autenticado no sistema, a próxima tela a ser chamada são telas de sequência, tela d de cadastros ou de agenda com suas funcionalidades principais de formulários.
3. O usuário é obrigado fazer o cadastro em sequência Ex: Cadastro Domiciliar, Cadastro Individual e Cadastro de Visitas.
4. O domínio do sistema na tela de menu são as opções de operação e funcionalidade com restrições que o usuário vai operar e tratar dessas informações.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de menu vai existir cinco opções que o usuário vai acessar e operar as pesquisas essências.

## Protótipo (Tela de Cadastro Domiciliar)



1. Objetivo da tela é mostrar as funcionalidades e a interação com usuário na tela de cadastro domiciliar que vai dá acesso ao um formulário de pesquisa, que serão salvo ou armazenado na tela de registro.
2. Estar tela de cadastro domiciliar é ativado quando o usuário tem acesso aos menus que ele deverá em sequência para cada cadastro, a próxima tela depois de ter preenchidos e armazenado nos registro é a tela de cadastro individual que o usuário vai ter acesso.
3. O usuário é obrigado fazer o cadastro em sequência Ex: Cadastro Domiciliar, Cadastro Individual e Cadastro de Visitas sem deixar campos vazios.
4. O domínio do sistema na tela de cadastro domiciliar são as informações que o sistema vai mostrar na tela para o usuário e funcionalidade com restrições que o usuário vai operar e tratar dessas informações.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de cadastro domiciliar vai ter opções para o usuário operar e interação do sistema com o usuário.

## Protótipo (Tela de Cadastro Individual)



1. Objetivo da tela de cadastro individual é mostrar as funcionalidades e a interação com usuário na tela de cadastro individual que vai dá acesso ao um formulário de pesquisa, que serão salvo ou armazenado na tela de registro.
2. Estar tela de cadastro individual é ativado quando o usuário tem acesso aos menus que ele deverá em sequência para cada cadastro, a próxima tela depois de ter preenchidos e armazenado nos registro é a tela de cadastro de visitas que o usuário vai ter acesso.
3. O usuário é obrigado fazer o cadastro em sequência Ex: Cadastro Domiciliar, Cadastro Individual e Cadastro de Visitas sem deixar campos vazios.
4. O domínio do sistema na tela de cadastro individual são as informações que o sistema vai mostrar na tela para o usuário e funcionalidade com restrições que o usuário vai operar e tratar dessas informações.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de cadastro individual vai ter opções para o usuário operar as suas funções e mostrar sua interação com o sistema.

## Protótipo (Tela de Cadastro de Visitas)



1. Objetivo da tela de cadastro de visitas é mostrar as funcionalidades e a interação com usuário na tela de cadastro de visitas que vai dá acesso ao um formulário de pesquisa, que serão salvo ou armazenado na tela de registro.
2. Estar tela de cadastro de visitas é ativado quando o usuário tem acesso aos menus que ele deverá em sequência para cada cadastro, a próxima tela depois de ter preenchidos e armazenado nos registro é a tela da Agenda que o usuário vai ter acesso para agendar outros visitas que faltaram para realizar.
3. O usuário é obrigado fazer o cadastro em sequência Ex: Cadastro Domiciliar, Cadastro Individual e Cadastro de Visitas sem deixar campos vazios.
4. O domínio do sistema na tela de cadastro de visitas são as informações que o sistema vai mostrar na tela para o usuário e funcionalidade com restrições que o usuário vai operar e tratar dessas informações.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de cadastro de visitas vai ter opções para o usuário operar suas funções principais e a interação com o sistema.

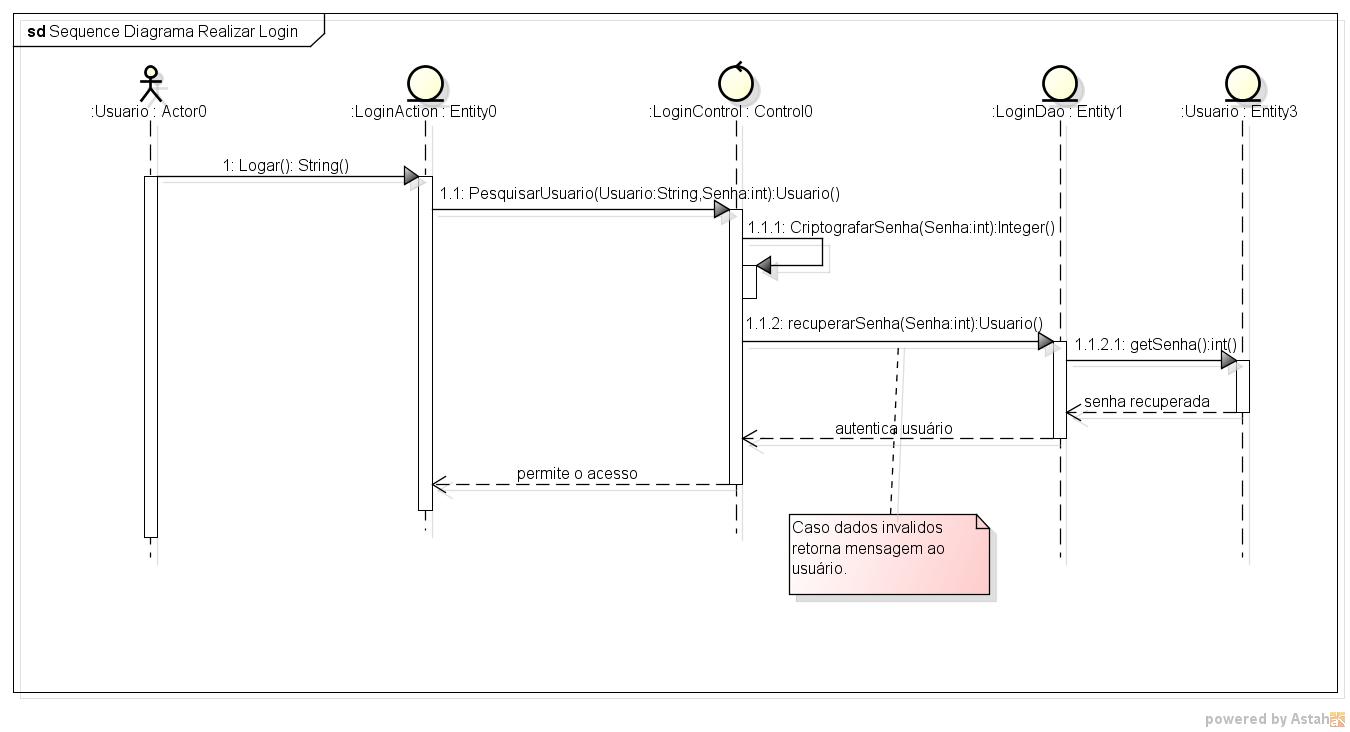
## Protótipo (Tela de Agenda)



1. Objetivo da tela de Agenda é mostrar as funcionalidades e a interação com usuário que vai poder agendar visitas que faltaram e usar outras funcionalidades do sistema, dando acesso a campos de cadastro simples, usar métodos de salvar, deletar, atualizar ou pesquisar.
2. Estar tela de agenda é ativada quando o usuário tem acesso ao menu e clicando na opção de agenda, a próxima tela depois de ter preenchidos e salvo os agendamentos é a tela de registros que o usuário vai ter acesso para manipular os dados de cada cadastro que foram salvos e realizados pelo o usuário.
3. O usuário só será obrigado a agendar visitas se for necessário sem deixar campos vazios e salvando sempre cada agendamento.
4. O domínio do sistema na tela de agenda são as informações que o sistema vai mostrar na tela para o usuário que vai operar e tratar dessas informações.
5. O tipos de usuário do sistema é único só tem acesso para o agente comunitário de saúde.
6. Na tela de agenda vai ter opções para o usuário operar e interagir com o sistema.

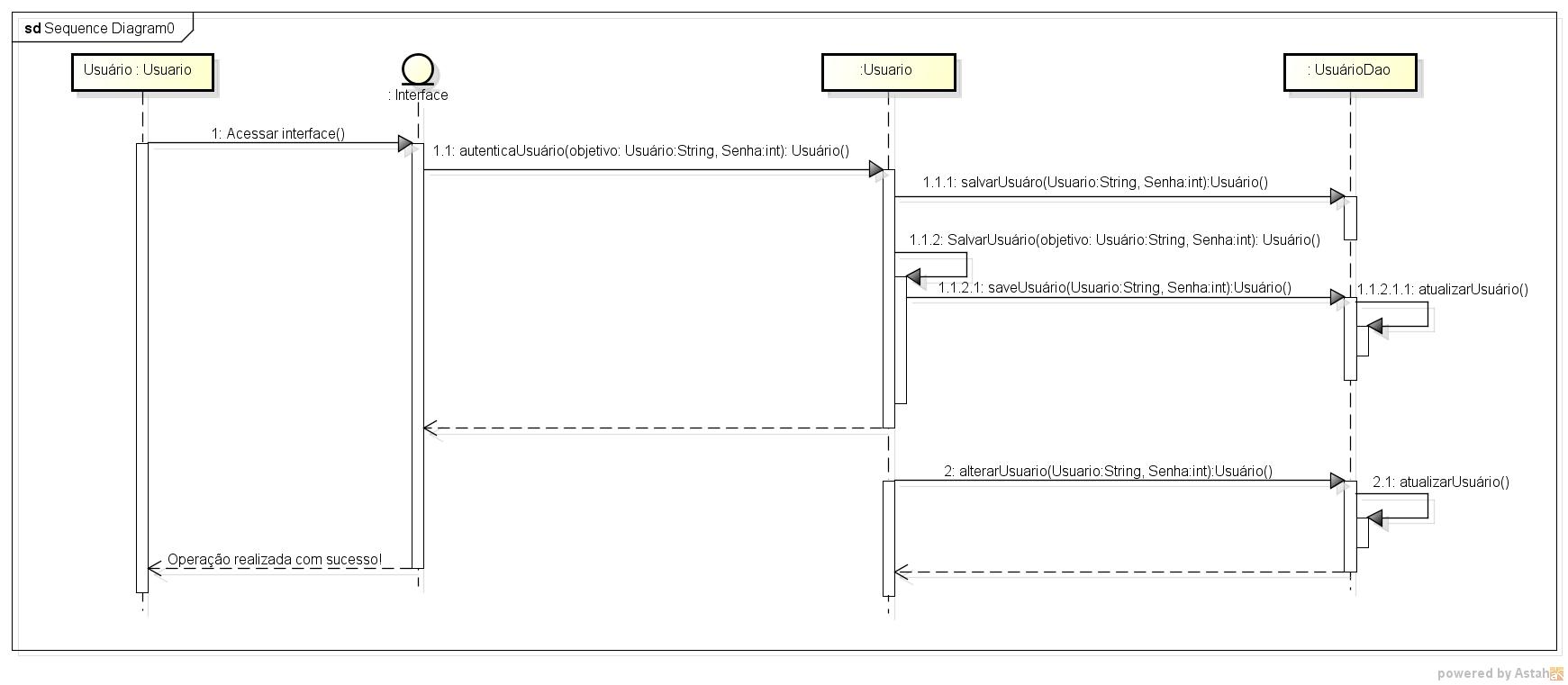
## Diagrama de Sequência

## Diagrama de Sequência Realizar Login



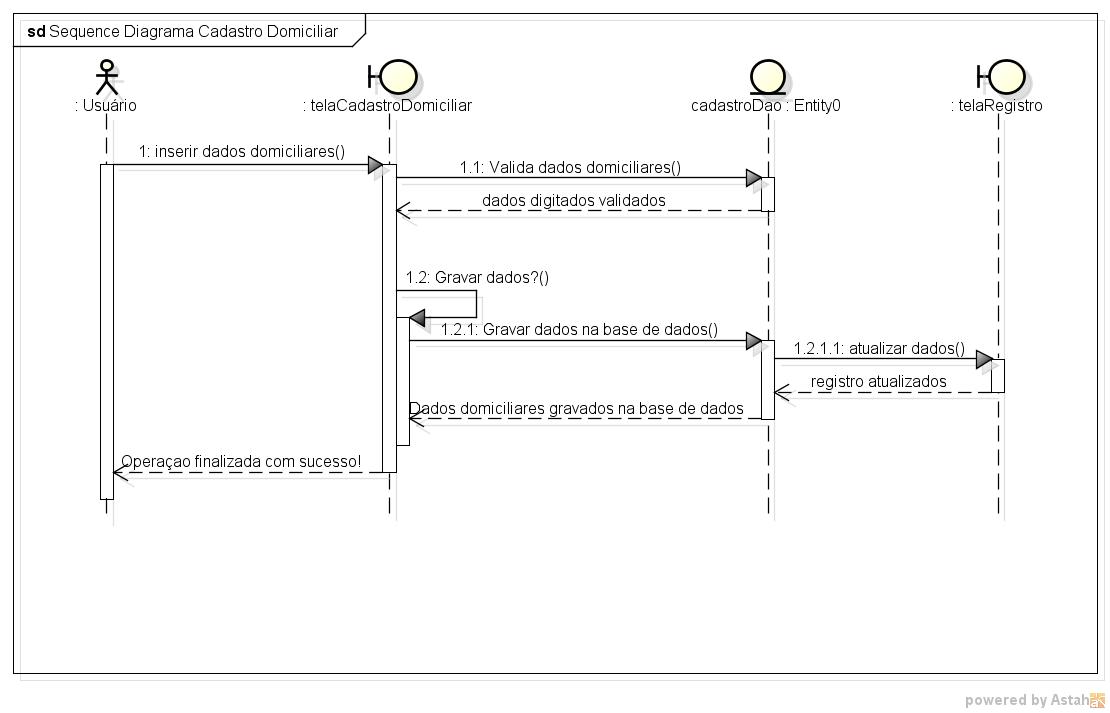
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para o login, demostra três **LifeLine**, a **LifeLine Usuário**, mostra que o usuário vai acessar a **LifeLine LoginAction** do sistema que nela o usuário vai preencher os campos que a login vai exibir para o usuário, sendo verificado e pesquisado pela a **LifeLine LoginControl** para informar se o usuário estar ou não cadastro no sistema, e se o usuário for valido no sistema, o sistema informa ao usuário com uma mensagem e dá acesso as funcionalidades do sistema. |

## Diagrama de Sequência Manter de Usuário



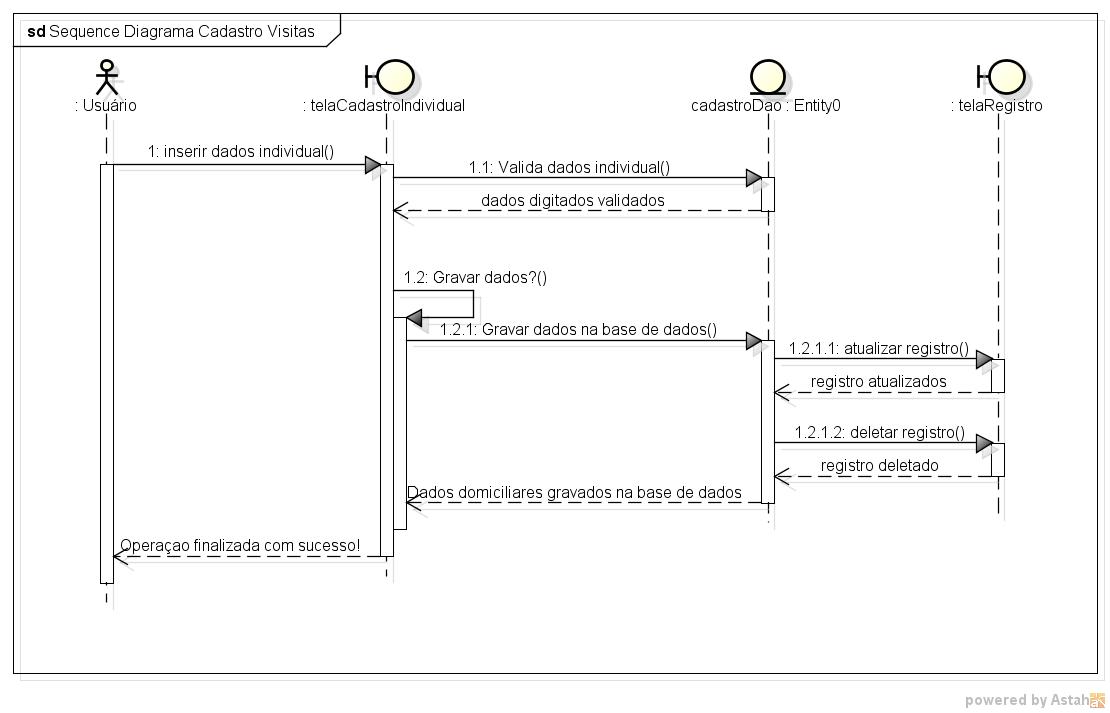
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para confirmar cadastro do usuário no sistema, demostrar quatro **LifeLine**, a **LifeLine** **Usuário**, mostra que o usuário vai se comportar parecido com o diagrama de sequência de login acessando a **LifeLine de Interface** do sistema que nela o usuário vai autenticar o usuário no cadastro exibindo para o usuário campos para ele preencher, em seguida a **LifeLine :Usuário** onde o usuário vai poder salvar os campos preenchidos no sistema, se autenticando e exibindo um mensagem de confirmação ao usuário, o sistema também dá ao usuário opção de se atualizar na **LifeLine :UsuárioDao**. |

## Diagrama de Sequência Cadastros Domiciliar



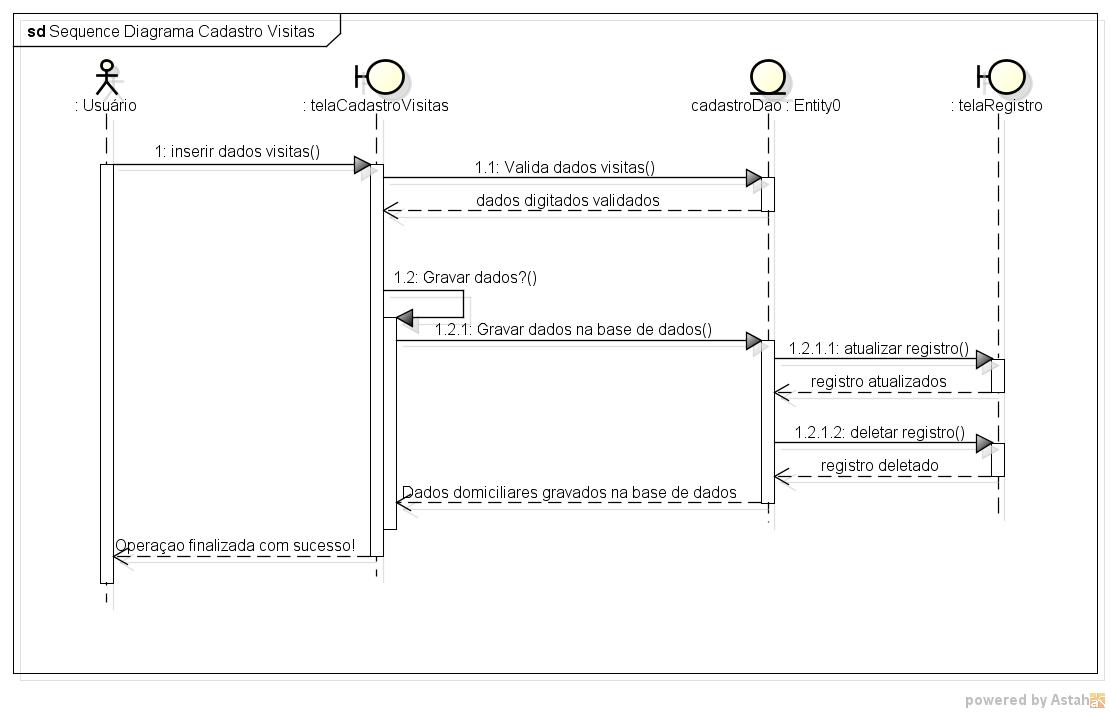
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para ilustrar como os cadastro de cada menu vai se comportar no sistema, demostrar com quatro LifeLine, a LifeLine usuário uso para demostrar como o usuário vai agir na hora de operar o sistema e o que o sistema vai realizar para o usuário, na primeira **LifeLine** mostra que o usuário acessar ao menu principal do sistema depois de estar cadastro, aonde ele vai ter uma interface que coloquei de nome Tela principal, nela vai exibir menu e as opções e os cadastro que usuário vai manipular, o usuário vai selecionar o cadastro após o sistema vai exibir os campos para ser feito a coleta de dados, ele vai preencher todos os campos sem deixar campos vazios, depois de ter realizado essa operação na **LifeLine da telaCadastroDomiciliar**, o usuário vai fazer a confirmação de todos os dados preenchidos depois de verificar se todos os campos estão preenchidos de forma correta passando na **LifeLine** de Cadastro vai salvar todos os dados e persistir os dados na **LifeLine** de **Dados** e **CadastroDao** para ficarem armazenados e podendo atualizar o cadastro na **LifeLine telaRegistro**. |

## Diagrama de Sequência Cadastros Individual



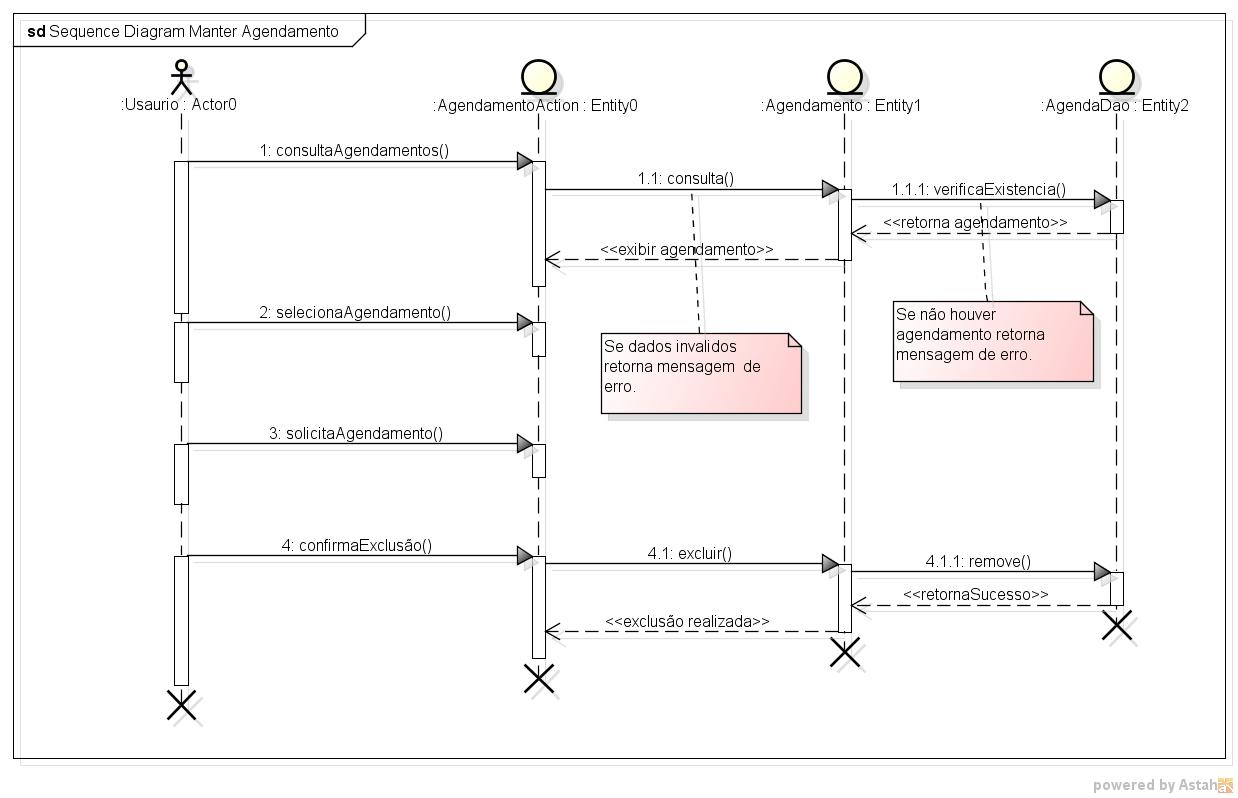
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para ilustrar como os cadastro de cada menu vai se comportar no sistema, demostrar com quatro LifeLine, a LifeLine usuário uso para demostrar como o usuário vai agir na hora de operar o sistema e o que o sistema vai realizar para o usuário, na primeira **LifeLine** mostra que o usuário acessar ao menu principal do sistema depois de estar cadastro, aonde ele vai ter uma interface que coloquei de nome Tela principal, nela vai exibir menu e as opções e os cadastro que usuário vai manipular, o usuário vai selecionar o cadastro após o sistema vai exibir os campos para ser feito a coleta de dados, ele vai preencher todos os campos sem deixar campos vazios, depois de ter realizado essa operação na **LifeLine da telaCadastroDomiciliar**, o usuário vai fazer a confirmação de todos os dados preenchidos depois de verificar se todos os campos estão preenchidos de forma correta passando na **LifeLine** de Cadastro vai salvar todos os dados e persistir os dados na **LifeLine** de **Dados** e **CadastroDao** para ficarem armazenados e podendo atualizar o cadastro na **LifeLine telaRegistro**. |

## Diagrama de Sequência Cadastros Visitas



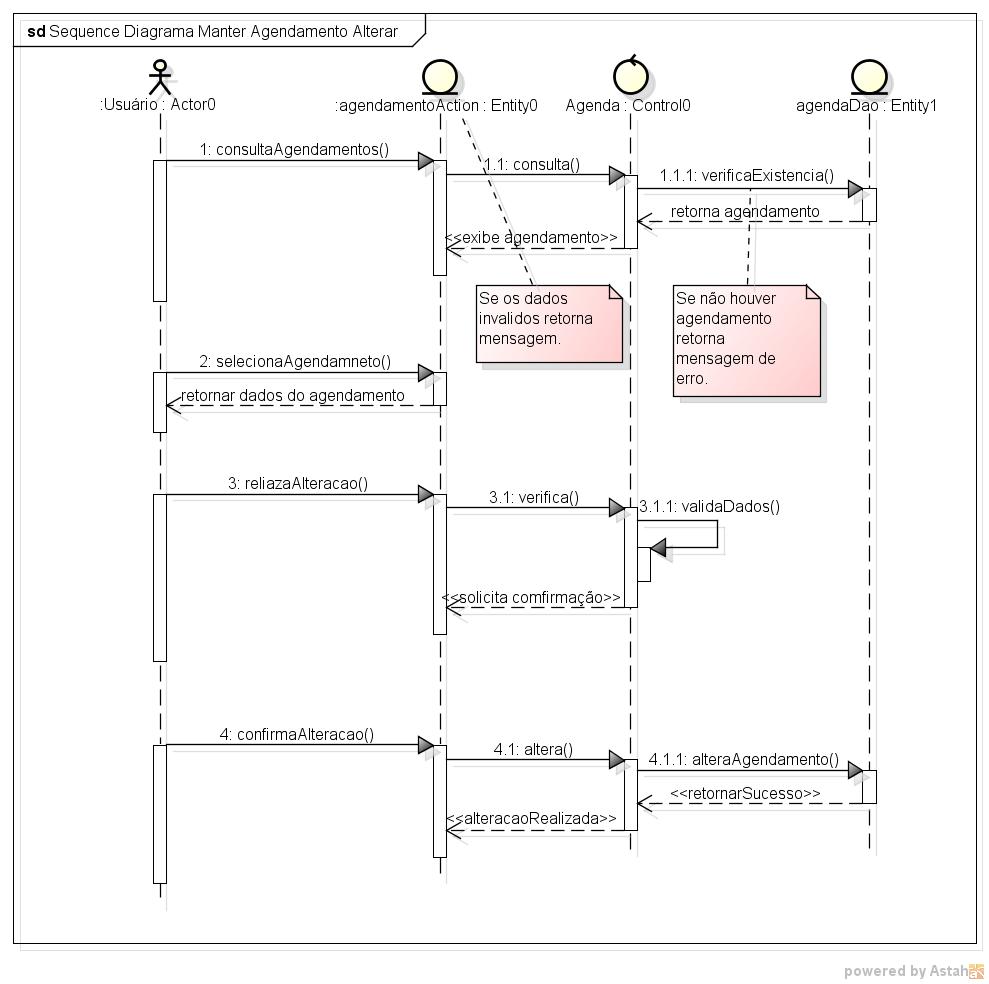
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para ilustrar como os cadastro de cada menu vai se comportar no sistema, demostrar com quatro LifeLine, a LifeLine usuário uso para demostrar como o usuário vai agir na hora de operar o sistema e o que o sistema vai realizar para o usuário, na primeira **LifeLine** mostra que o usuário acessar ao menu principal do sistema depois de estar cadastro, aonde ele vai ter uma interface que coloquei de nome Tela principal, nela vai exibir menu e as opções e os cadastro que usuário vai manipular, o usuário vai selecionar o cadastro após o sistema vai exibir os campos para ser feito a coleta de dados, ele vai preencher todos os campos sem deixar campos vazios, depois de ter realizado essa operação na **LifeLine da telaCadastroDomiciliar**, o usuário vai fazer a confirmação de todos os dados preenchidos depois de verificar se todos os campos estão preenchidos de forma correta passando na **LifeLine** de Cadastro vai salvar todos os dados e persistir os dados na **LifeLine** de **Dados** e **CadastroDao** para ficarem armazenados e podendo atualizar o cadastro na **LifeLine telaRegistro**. |

## Diagrama de Sequência Manter Agendamento



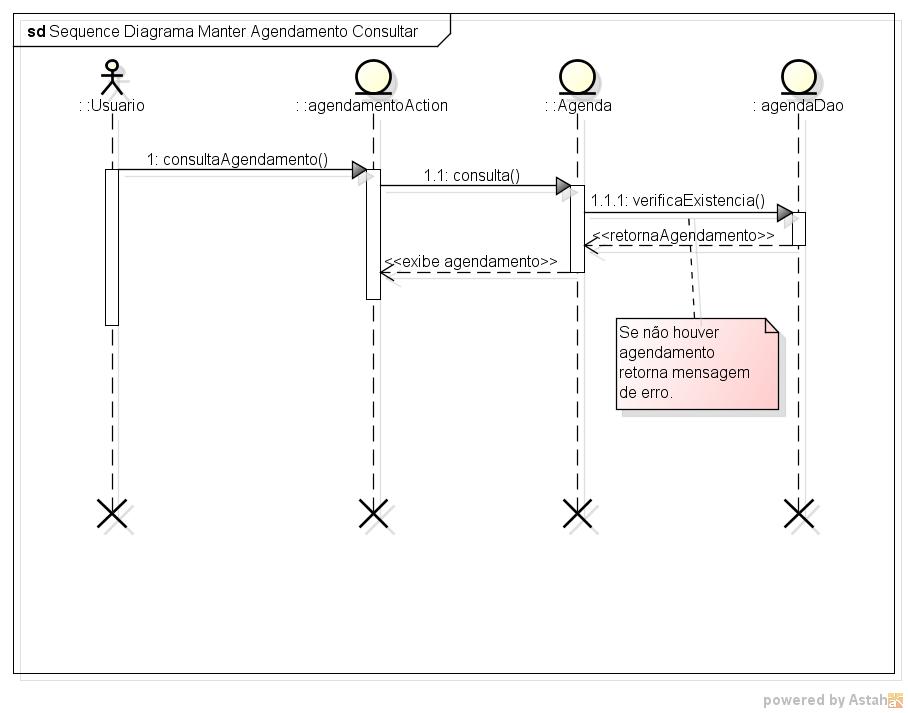
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para a Tela da Agenda, com objetivo de demostrar que inicia com três **LifeLine,** a primeira **LifeLine Usuário** mostra o usuário solicitando um agendamento na **LifeLine Agenda** aonde após sua solicitação a interface da agenda deverá exibir os campos de preenchimento ao usuário que o próprio informa os dados, salvando as informações preenchias pelo usuário, na **LifeLine AgendaDao** aonde ele tem opções como atualizar e excluir. |

## Diagrama de Sequência Manter Agendamento Alterar



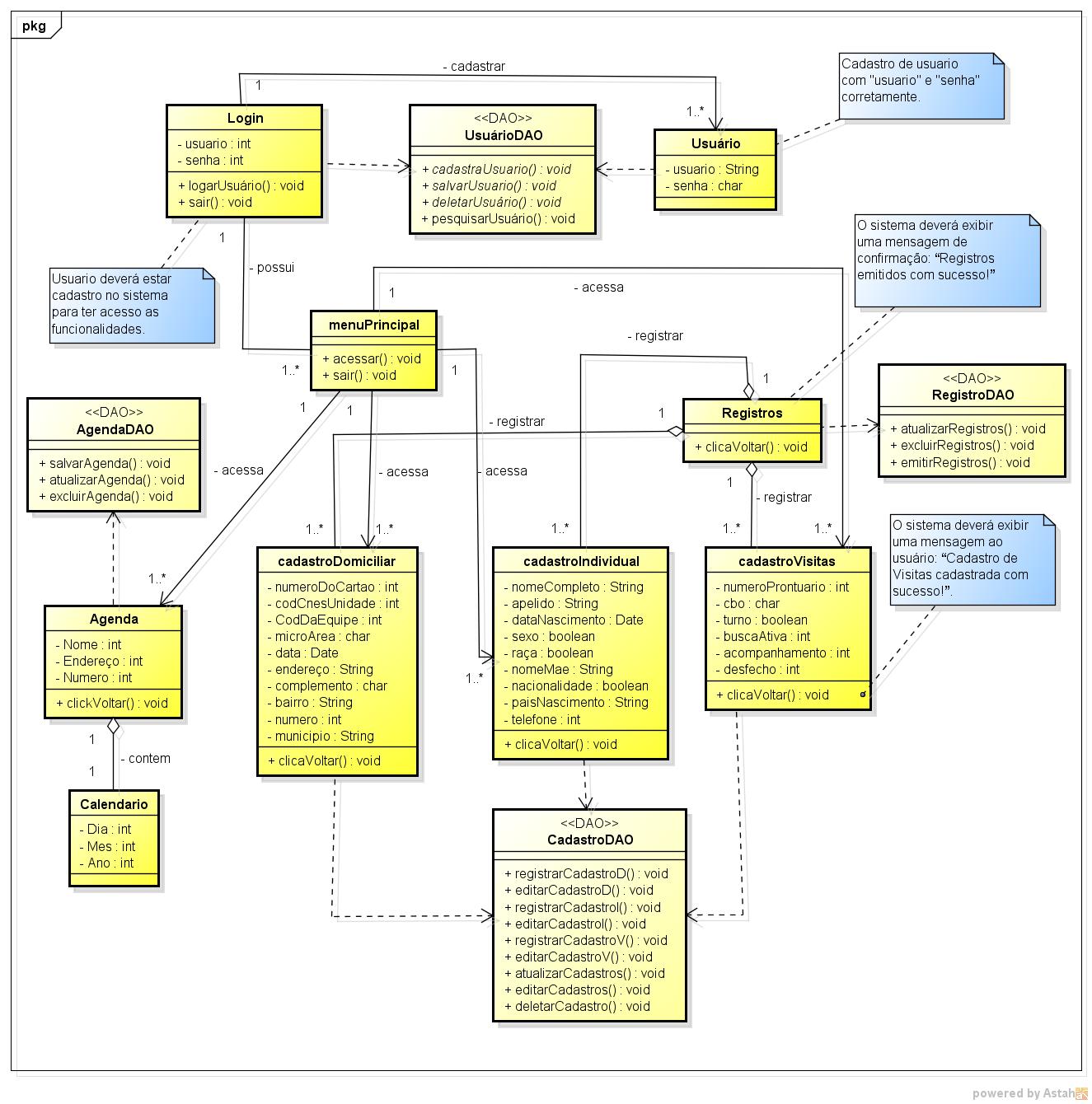
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para a Tela da Agenda, com objetivo de demostrar que inicia com três **LifeLine,** a primeira **LifeLine Usuário** mostra o usuário solicitando um agendamento na **LifeLine AgendamentoAction** aonde após sua solicitação a interface da agenda deverá exibir os campos de preenchimento ao usuário que o próprio informa os dados, salvando as informações preenchias pelo usuário, na **LifeLine AgendaDao** aonde ele tem opções como atualizar e excluir. |

## Diagrama de Sequência Manter Agendamento Consultar

****

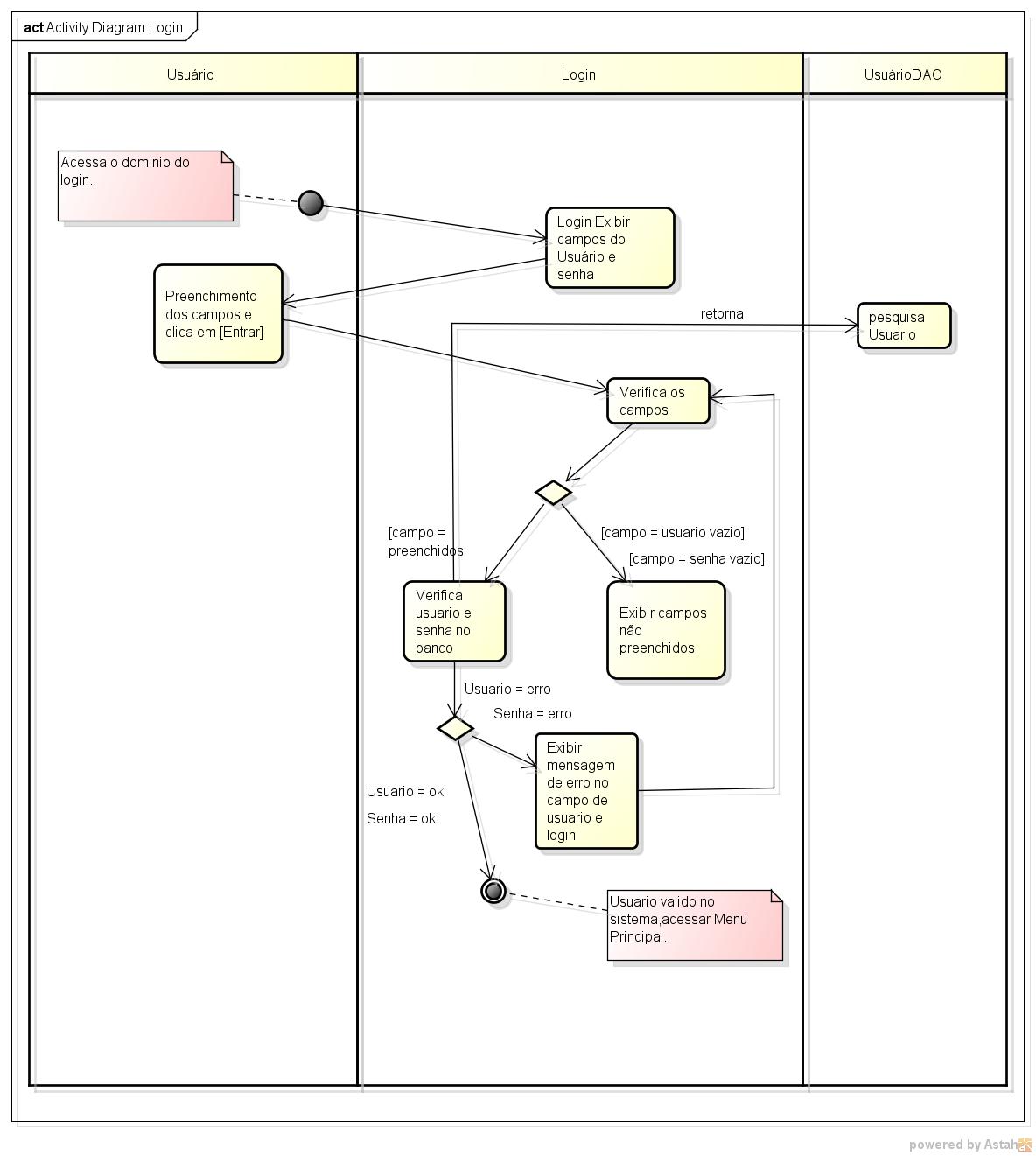
|  |
| --- |
| **Descrição:** |
| No diagrama de sequência que foi desenvolvido para a Tela da Agenda, com objetivo de demostrar que inicia com três **LifeLine,** a primeira **LifeLine Usuário** mostra o usuário solicitando um agendamento na **LifeLine AgendamentoAction** aonde após sua solicitação a interface da agenda deverá exibir os campos de preenchimento ao usuário que o próprio informa os dados, salvando as informações preenchias pelo usuário, na **LifeLine AgendaDao** aonde ele tem opções como atualizar e excluir. |

## Diagrama de Classes do Sistema



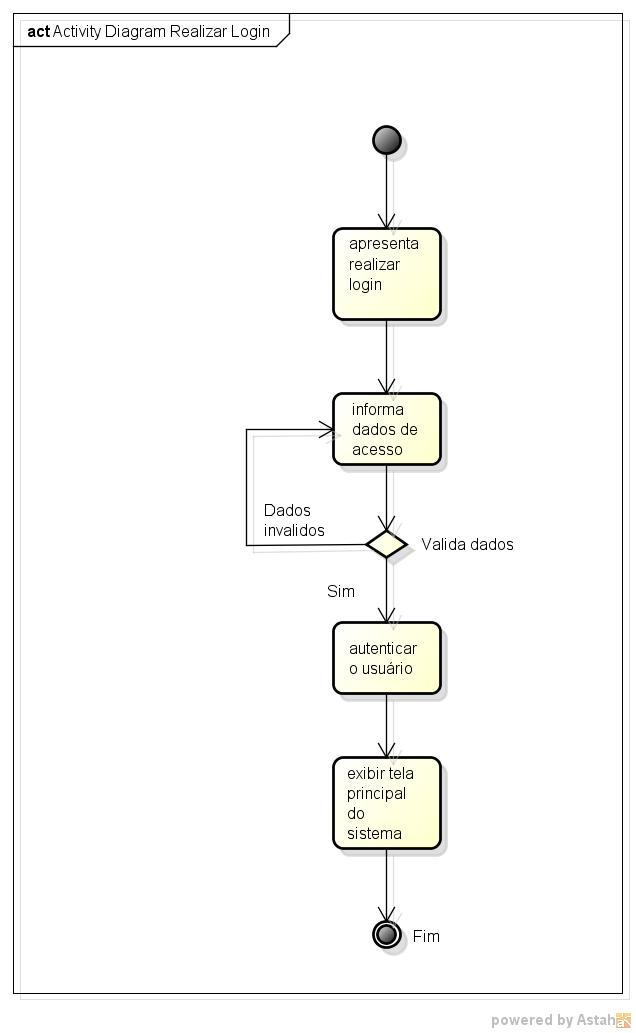
|  |
| --- |
| **Descrição:** |
| No diagrama de Classe do sistema, o próprio no já diz que são as classe do sistema estaticamente representadas, afim de auxiliar no desenvolvimento dos outro diagramas, então o diagrama de classe mostra como as classes se relacionam entre si quais são seus atributos, métodos e nome para cada classe. Mostrando também as classes **DAO** que são de alta importância para o desenvolvimento do sistema. |

## Diagrama de Atividades Login



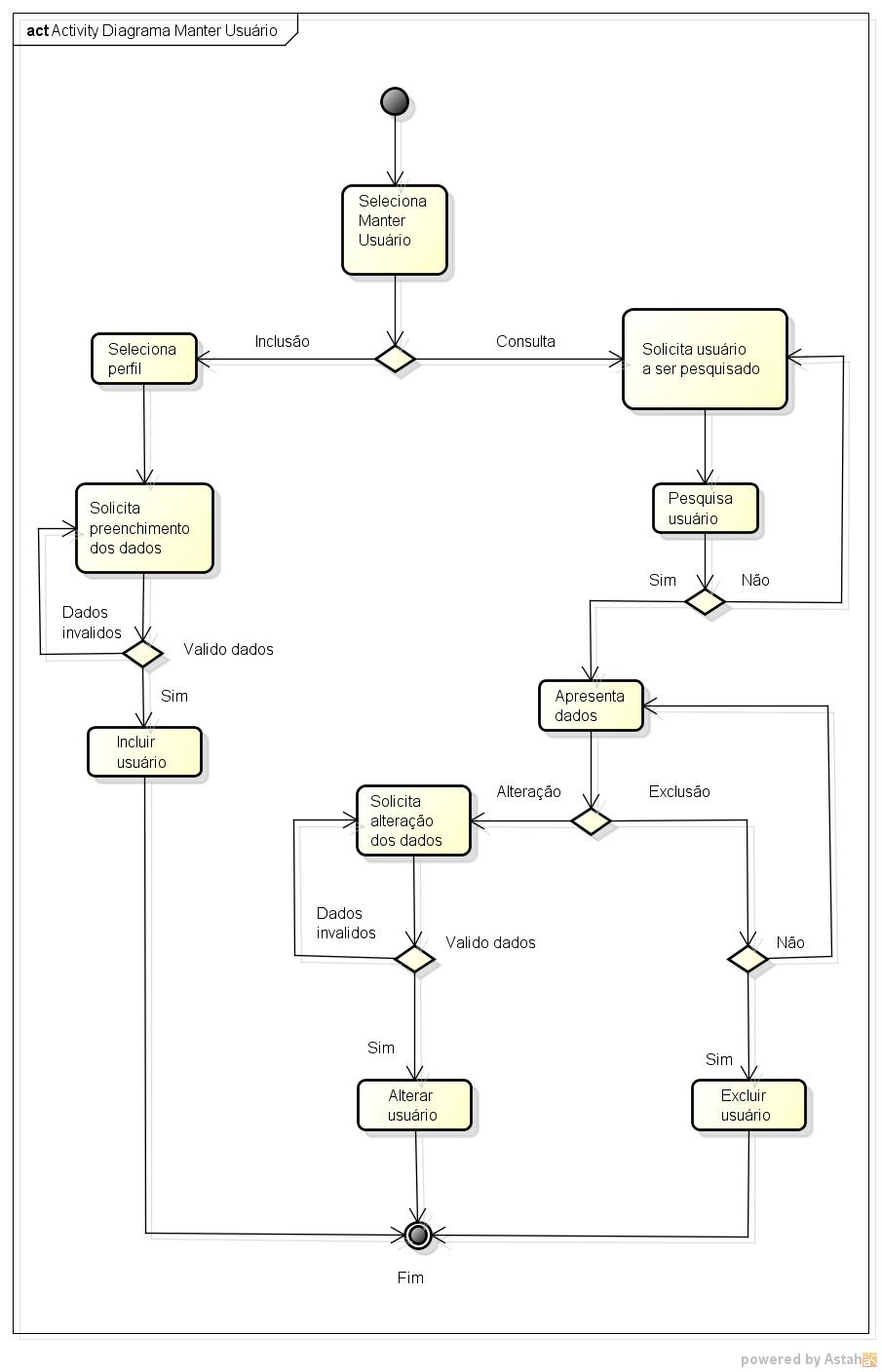
|  |
| --- |
| **Descrição:** |
| Iniciando os diagramas de Atividades, mostramos como os fluxos de informações e dados se comportam mudando suas ações para cada digrama diferente, neste diagrama usei **Partition[Vertical],** para organizar o diagrama e suas informações. O diagrama de Atividade do Login mostra o fluxo e aonde inicia e se terminar sua operação. |

## Diagrama de Atividade Realizar Login



|  |
| --- |
| **Descrição:** |
| Iniciando os diagramas de Atividades realizar login, mostramos como os fluxos de informações e dados se comportam mudando suas ações para cada digrama diferente, para organizar o diagrama e suas informações. O diagrama de Atividade do Login mostra o fluxo e aonde inicia e se terminar sua operação. |

## Diagrama de Atividades Manter Usuário



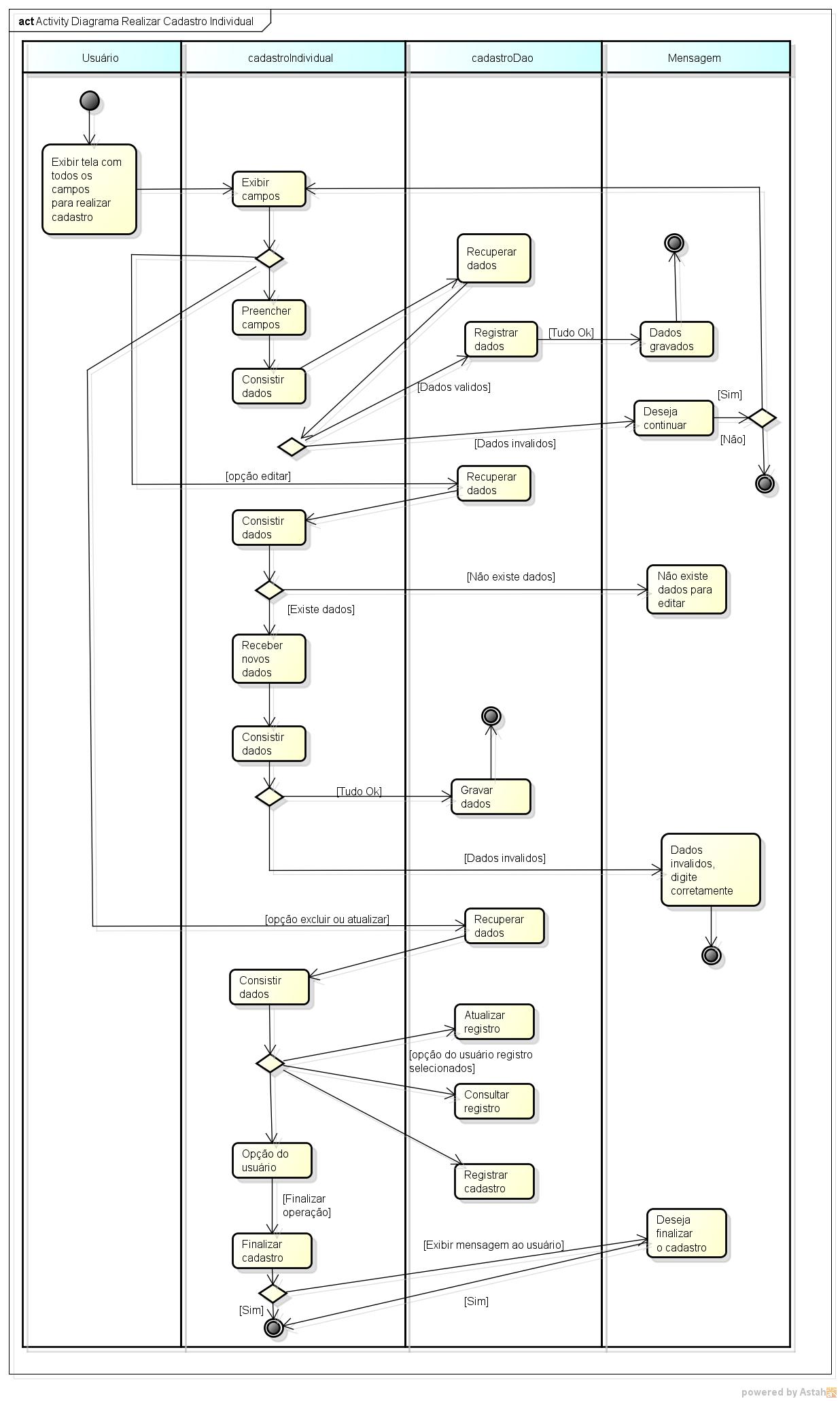
|  |
| --- |
| **Descrição:** |
| No diagrama de atividade para a interface do cadastro de usuário no sistema, usei **Partition[Vertical]** para organizar e iniciar **Partition** Usuário iniciando com um bola toda preta mostra o que o usuário deverá fazer em seguida vai para a segunda **Partition** chamada Cadastro de usuário mostrando cada face que usuário deverá passar para ser cadastro no sistema por fim finalizado o cadastro e se encerrando com um bola com bordas branca e no seu centro preta. |

### Diagrama de Atividade Realizar Cadastro Domiciliar

### C:\Users\Cavalcante\Desktop\SIPACS\DAIGRAMAS ATUALIZADOS\Activity Diagrama Realizar Cadastro Domiciliar.jpg

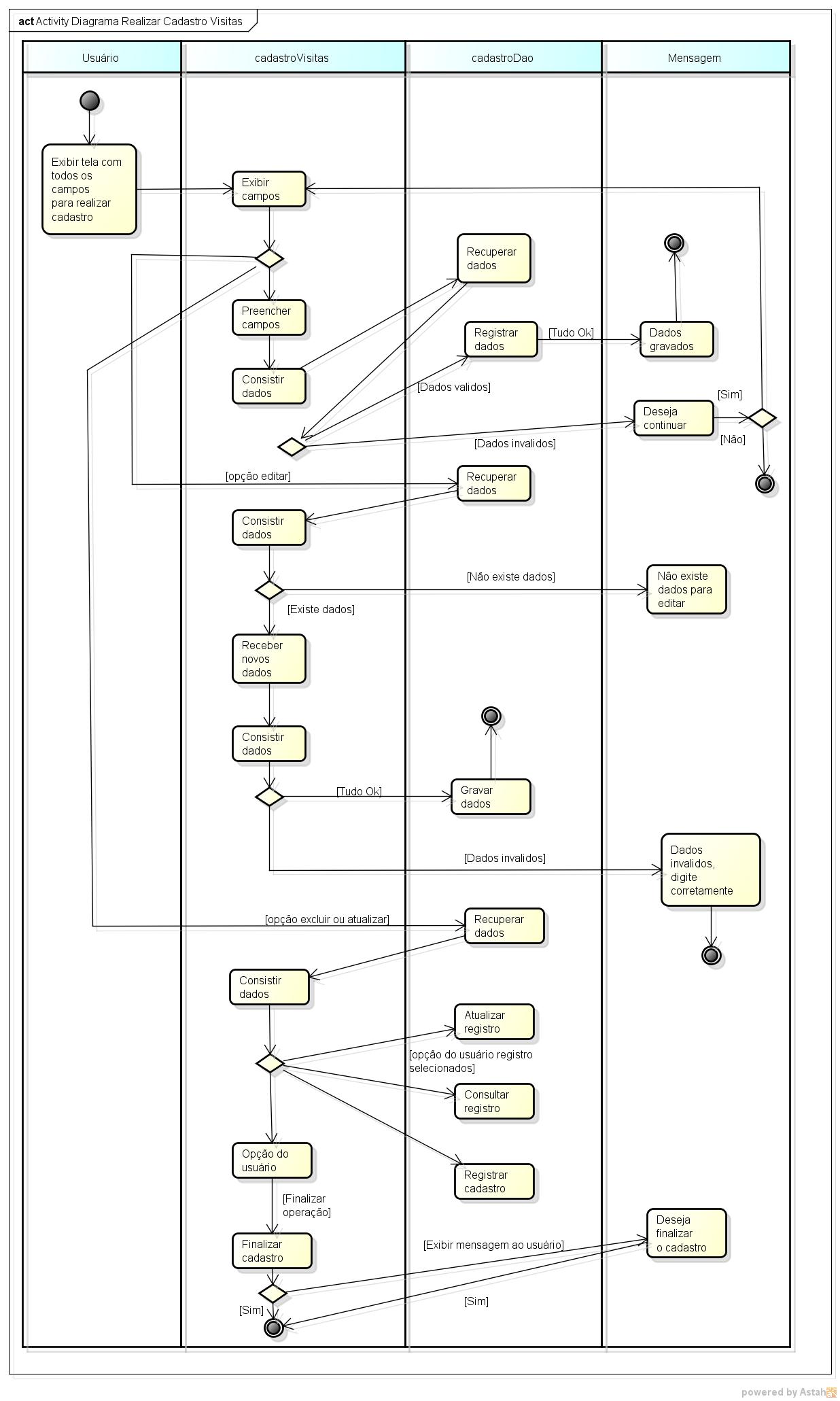
|  |
| --- |
| **Descrição:** |
| O diagrama de atividades Realizar Cadastro Domiciliar no sistema, usa três **InitialNode, ActivityFinal, Action e Decision Node** para deixar mais organizado o diagrama, definido na primeira **InitialNode** que dá início as outras ações do diagrama com cada evento que forma a atividade, **Action** demostra o que o sistema vai realizar em cada **Action,** usando **DecisionNode** para **escolher** caminhos dentro do sistema e por fim para finalização a ação da operação o **ActivityFinal** que representa a finalização da ação realizada pelo o usuário. |

### Diagrama de Atividade Realizar Cadastro Individual



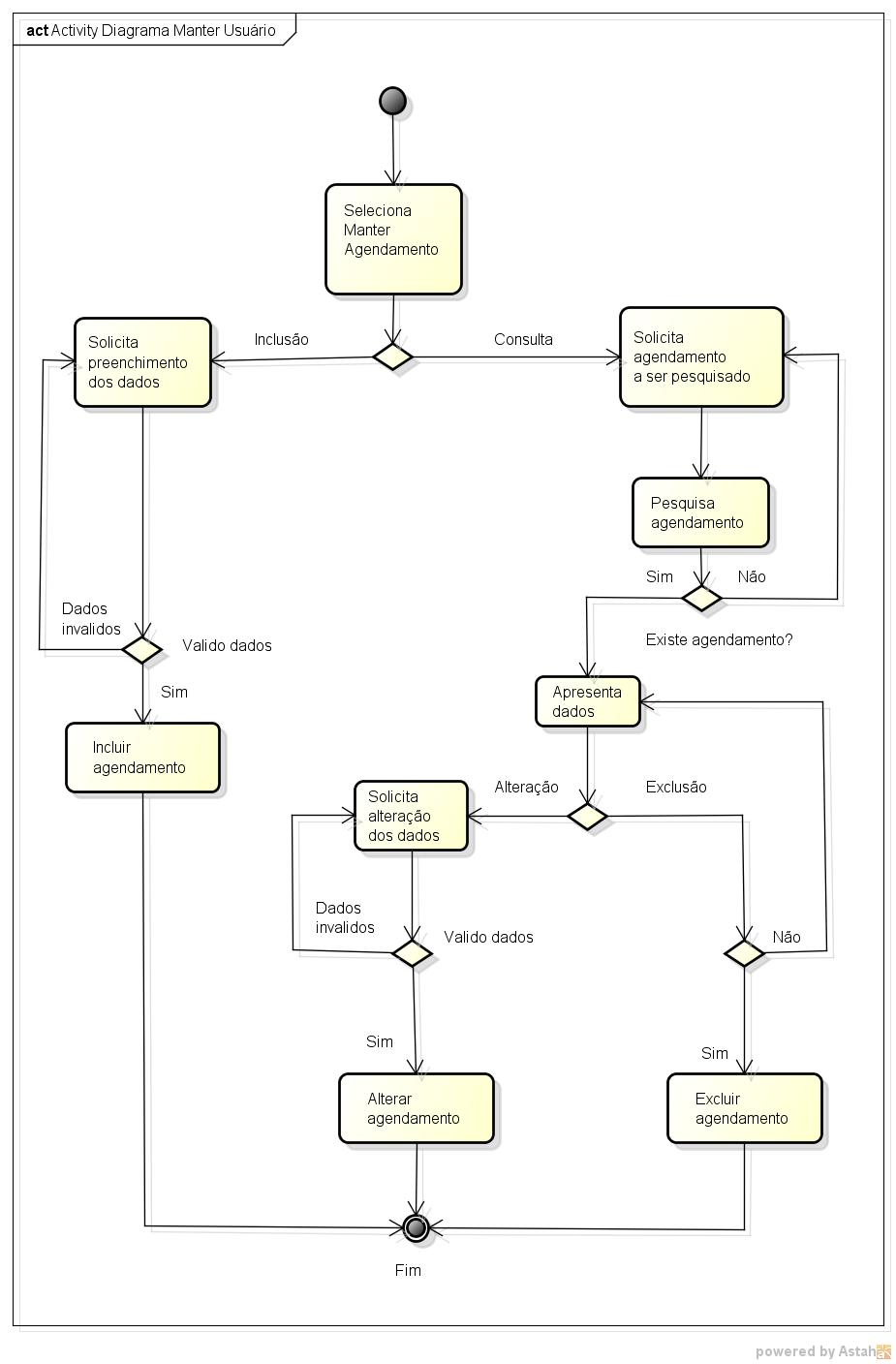
|  |
| --- |
| **Descrição:** |
| O diagrama de atividades Realizar Cadastro Individual no sistema, usa três **InitialNode, ActivityFinal, Action e Decision Node** para deixar mais organizado o diagrama, definido na primeira **InitialNode** que dá início as outras ações do diagrama com cada evento que forma a atividade, **Action** demostra o que o sistema vai realizar em cada **Action,** usando **DecisionNode** para **escolher** caminhos dentro do sistema e por fim para finalização a ação da operação o **ActivityFinal** que representa a finalização da ação realizada pelo o usuário. |

### Diagrama de Atividade Realizar Cadastro Visitas



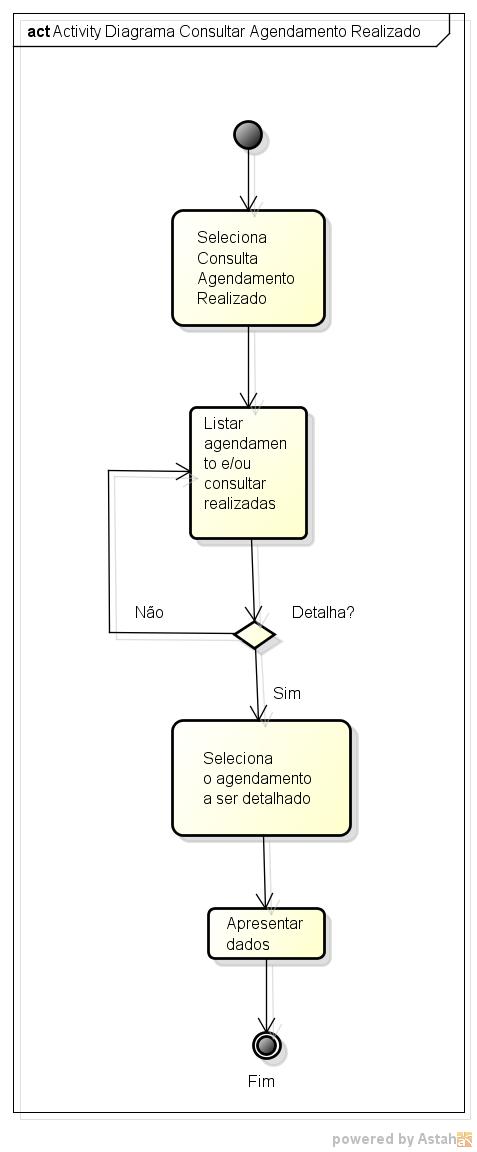
|  |
| --- |
| **Descrição:** |
| O diagrama de atividades Realizar Cadastro Visitas no sistema, usa três **InitialNode, ActivityFinal, Action e Decision Node** para deixar mais organizado o diagrama, definido na primeira **InitialNode** que dá início as outras ações do diagrama com cada evento que forma a atividade, **Action** demostra o que o sistema vai realizar em cada **Action,** usando **DecisionNode** para **escolher** caminhos dentro do sistema e por fim para finalização a ação da operação o **ActivityFinal** que representa a finalização da ação realizada pelo o usuário. |

### Diagrama de Atividades Manter Agendamento

****

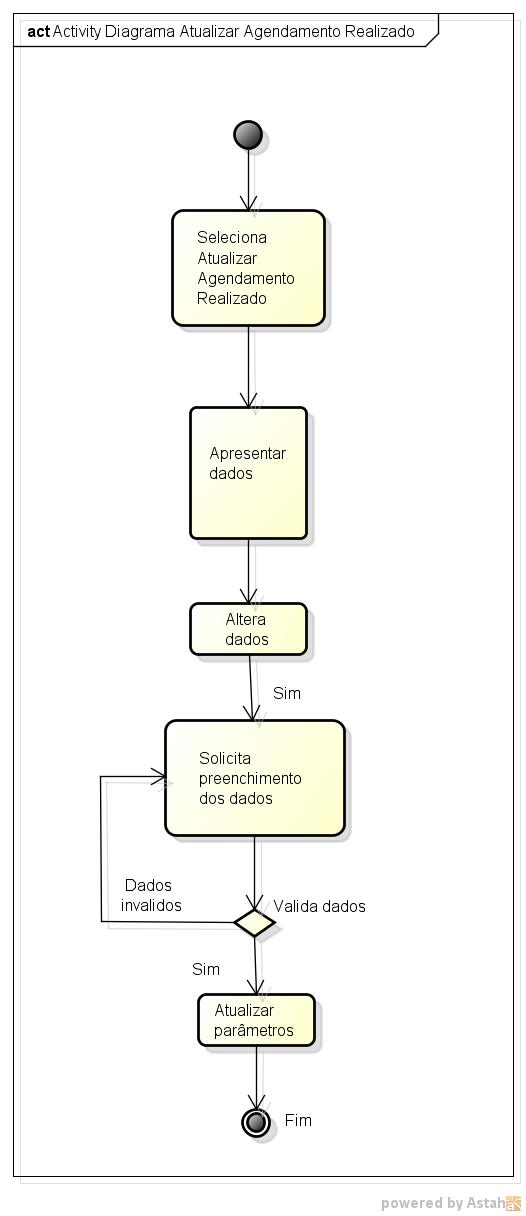
|  |
| --- |
| **Descrição:** |
| O diagrama de atividade Manter Agendamento no sistema, usei paraorganizar o diagrama e da entendimento a ação, definido na primeira **InitialNode** para iniciaroutras ações no diagrama com cada evento que forma a atividade, demostra como os caminhos vai realizar funções diferentes e que define a funcionalidade no sistema vai realizar dependendo do que o usuário vai fazer com os agendamentos. Representei com o **ActivityFinal**, para mostrar que quando a ação for realizada pelo o usuário finalizando no diagrama a fase selecionada. |

## Diagrama de Atividades Consultar Agendamento Realizado



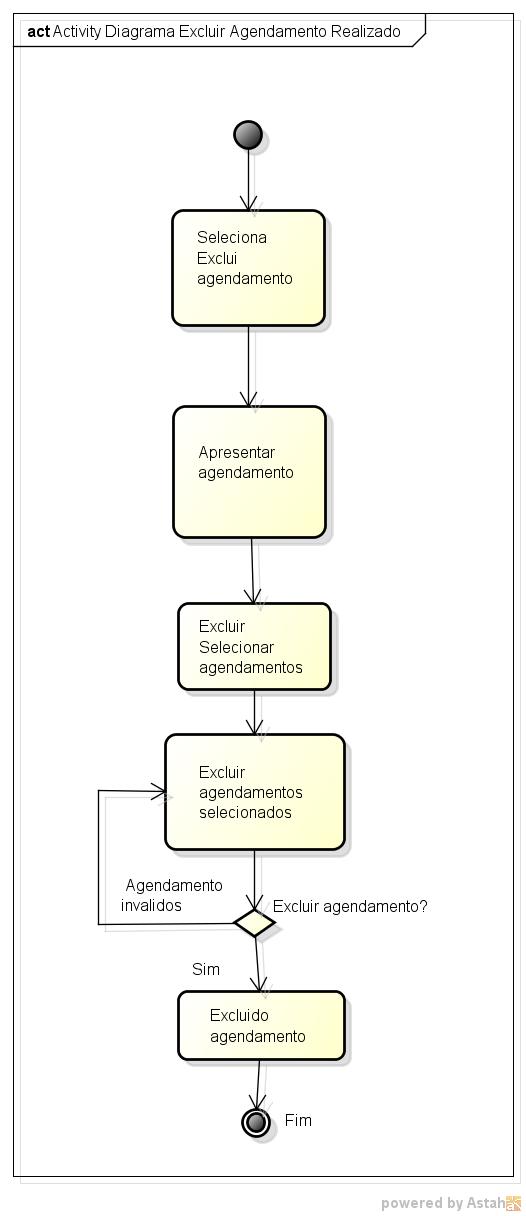
|  |
| --- |
| **Descrição:** |
| O diagrama de atividades Consulta Agendamento Realizado no sistema, usa três **InitialNode, ActivityFinal, Action e Decision Node** para deixar mais organizado o diagrama, definido na primeira **InitialNode** que dá início as outras ações do diagrama com cada evento que forma a atividade, **Action** demostra o que o sistema vai realizar em cada **Action,** usando **DecisionNode** para **escolher** caminhos dentro do sistema e por fim para finalização a ação da operação o **ActivityFinal** que representa a finalização da ação realizada pelo o usuário. |

## Diagrama de Atividades Atualizar Agendamento Realizado



|  |
| --- |
| **Descrição:** |
| O diagrama de atividades Atualizar Agendamento Realizado no sistema, usa três **InitialNode, ActivityFinal, Action e Decision Node** para deixar mais organizado o diagrama, definido na primeira **InitialNode** que dá início as outras ações do diagrama com cada evento que forma a atividade, **Action** demostra o que o sistema vai realizar em cada **Action,** usando **DecisionNode** para **escolher** caminhos dentro do sistema e por fim para finalização a ação da operação o **ActivityFinal** que representa a finalização da ação realizada pelo o usuário. |

## Diagrama de Atividades Excluir Agendamento Realizado



|  |
| --- |
| **Descrição:** |
| O diagrama de atividades Excluir Agendamento Realizado no sistema, usa três **InitialNode, ActivityFinal, Action e Decision Node** para deixar mais organizado o diagrama, definido na primeira **InitialNode** que dá início as outras ações do diagrama com cada evento que forma a atividade, **Action** demostra o que o sistema vai realizar em cada **Action,** usando **DecisionNode** para **escolher** caminhos dentro do sistema e por fim para finalização a ação da operação o **ActivityFinal** que representa a finalização da ação realizada pelo o usuário. |

# Planejamento de Testes

## Plano de Testes

### CT01-Autenticar Login (Positivo)

|  |  |
| --- | --- |
| Descrição: | Neste Caso de Teste, verificar se é possível efetuar o Login no sistema. |
| Pré-Condição: | Deverá existir um usuário João cadastrado no sistema. O sistema deve estar disponível e operante |
| Ações: | **Resultados Esperados:** Deverá ser apresentado um tela de Login, com os seguintes campos a serem preenchidos: “**Usuário**”, “**Senha**”. |
| 1. O usuário deverá executar o ícone Sipacs que se encontra na área de trabalho do dispositivo móvel. | 1. O sistema exibirá a tela de Login com os campos usuário, senha e logar. |
| 1. Digite <João> para no campo “Usuário”. <123456> no campo “Senha”. | 1. O sistema apresentara a tela Principal ao usuário. |
| 1. O usuário clique no Botão [Entrar]. | 1. O sistema dá acesso ao usuário autenticado. |
| Pós-Condições: | O usuário João terá acesso ao tela principal do sistema. |

**CT02-Autenticar Login (Negativo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo desse Caso de Teste é mostrar o requisito anormal para efetuar o Login no sistema. |
| Pré-Condição: | Deverá existir um usuário João cadastrado no sistema. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado um tela de Login, com os seguintes campos a serem preenchidos: “**Usuário**”, “**Senha**”. |
| 1. O usuário deverá executar o ícone Sipacs que se encontra na área de trabalho do dispositivo móvel. | 1. O sistema exibirá a tela de Login com os campos Usuário, Senha para logar no sistema. |
| 1. Digite <joão> para no campo “Usuário”. <123456> no campo “Senha”. E clique no Botão [Entrar]. | 1. O sistema apresentará uma mensagem: **“Usuário valido!”**. |
| 1. O usuário clica em [Ok]. | 1. O sistema carrega mais dá acesso ao usuário na tela principal. Retorna com os campos vazios. |
| Pós-Condições: | O sistema não dá acesso ao usuário João mesmo sendo autenticado e cadastro no sistema Sipacs. |

**CT01-Cadastrar Usuário (Positivo)**

|  |  |
| --- | --- |
| Descrição: | O Caso de Teste cadastra usuário tem como objetivo é verificar se é possível realizar o cadastro de usuário no sistema Sipacs. |
| Pré-Condição: | O usuário deverá se cadastrar no sistema para ter acesso a tela de menu principal. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado um tela de Cadastro de usuário, com os seguintes campos a serem preenchidos: “**Usuário**”, “**Senha**”. |
| 1. O usuário deverá executar o ícone Sipacs que se encontra na área de trabalho do dispositivo móvel exibindo a tela de login com dois botões Entra e Registra ou Cadastro. | 1. O sistema exibirá a tela de Cadastro de usuário com os campos **Usuário**, **Senha** e com um botão **Cadastrar**. |
| 1. Digite <João> para no campo “Usuário”. <123456> no campo “Senha”. E clique no Botão [Cadastrar]. | 1. O sistema informa ao usuário uma mensagem: “**Usuário cadastro com sucesso!**”. O sistema retorna para e tela de login depois do usuário autenticado no sistema. |
| Pós-Condições: | Com o usuário autenticado e registrado no sistema, poderá ter acesso a Tela de Login e após a Tela de menu principal do sistema Sipacs. |

**CT02 – Cadastrar Usuário (Negativo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo desse Caso de Teste é mostrar com o requisito pode ocorrer de forma anormal ao usuário. |
| Pré-Condição: | O usuário deverá se cadastrar no sistema para ter acesso as sua tela de menu principal. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado um tela de Cadastro de Usuário, com os seguintes campos a serem preenchidos: “**Usuário**”, “**Senha**” e com um botão Cadastrar. |
| 1. O usuário deverá executar o ícone Sipacs que se encontra na área de trabalho do dispositivo móvel exibindo a tela de login com dois botões Entra e Registra ou Cadastro. | 1. O sistema exibirá a tela de Cadastro de Usuário com os campos **Usuário**, **Senha** e **Cadastrar**. |
| 1. Digite <joão> para no campo “Usuário”. <123456> no campo “Senha”. E clique no Botão [Cadastrar]. | 1. Com o usuário cadastro e autenticado no sistema, de forma anormal o sistema poderá realizar o cadastro mais não reconhecer o usuário. |
| Pós-Condições: | O usuário João poderá não ser reconhecido pelo o sistema sendo uma forma anormal que o sistema poderá representar ao usuário. |

**CT01 – Cadastro Domiciliar (Positivo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo desse Caso de Teste é mostrar o fluxo das ações que seram realizadas no cadastrar domiciliar com os dados no sistema que seram salvos no sistema. |
| Pré-Condição: | O usuário João que já estar logado no sistema como um usuário registrado. Não deverá ser cadastro o mesmo domicilio em um único dia data e hora. |
| Ações: | **Resultados Esperados**: Deverá ser apresentado uma tela de cadastro domiciliar ao usuário João, com campos que serão exibidos para ser preenchidos: “**Nome**”, “**Numero**”, “**Endereço**” e “**Bairro**”. |
| 1. O usuário na Tela de Menu Principal deverá acessar o menu Cadastro Domiciliar para realizar suas pesquisa e coleta de dados. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<Maria> para no campo “Nome”. Digite<54> para no campo “Numero”. Digite<Rua Francisco> para no campo “Endereço”. Digite<Centro> para no campo “Bairro”. | 1. Deverá exibir uma mensagem: “**Cadastro domiciliar realizado com sucesso!”.** |
| 1. O usuário João, deverá clicar no botão[Ok] na mensagem. | 1. O usuário João, salva o cadastro domiciliar clicando em **[Salvar].** |
| 1. O usuário João, poderá atualizar o cadastro. | 1. O usuário João, poderá editar o cadastro. |
| 1. O usuário poderá clicar no botão [Atualizar] o cadastro | 1. O usuário poderá clicar no botão **[Editar]** o cadastro. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro atualizado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro editado!”. Usuário clica em **[Ok].** |
| Pós-Condições: | O sistema deverá armazenar os dados no sistema, e salvar os dados coletados na Tela de Registro aonde o usuário poderá manipular as informações coletadas. |

**CT02 – Cadastro Domiciliar (Negativo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo de CT é verificar a impossibilidade deletar cadastro domiciliar do sistema. |
| Pré-Condição: | O usuário deverá estar logado no sistema como um usuário.  Não deverá ser cadastro o mesmo domicilio em um único dia data e hora no sistema. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado uma tela de cadastro, com campos que serão preenchidos: “**Nome**”, “**Numero**”, “**Endereço**” e “**Bairro**”. |
| 1. O usuário na Tela Principal deverá acessar o menu Cadastro Domiciliar em sequência de cada cadastro. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<Maria> para no campo “Nome”. Digite<54> para no campo “Numero”. Digite<Rua Francisco> para no campo “Endereço”. Digite<Centro> para no campo “Bairro”. | 1. Deverá exibir uma mensagem: “**Cadastro não realizou a operação, digite novamente os campos corretamente!**”. |
| 1. O usuário deverá clicar no botão[Ok]. | 1. O usuário João, salva o cadastro domiciliar clicando em [**Salvar**]. |
| 1. O usuário João, poderá atualizar o cadastro. | 1. O usuário João, poderá editar o cadastro. |
| 1. O usuário poderá clicar no botão [Atualizar] o cadastro. | 1. O usuário poderá clicar no botão [**Editar**] o cadastro. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro atualizado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Cadastro editado!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O sistema poderá não realizar corretamente com o requisito dando um forma anormal nas mensagens ao usuário, podendo não aparecer na tela para o usuário visualizar. |

**CT01 – Cadastro Individual (Positivo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo de CT é verificar se é possível cadastrar todos os dados no sistema. |
| Pré-Condição: | O usuário deverá estar logado no sistema como um usuário.  Não deverá ser cadastro o mesmo individuo ou o mesmo cadastro individual em um único dia, data e hora. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado uma tela de cadastro, com campos que serão preenchidos: “**Nome** **Completo**”, “**Data** **de** **Nascimento**”, “**Apelido**” e “**Sexo**”. |
| 1. O usuário na Tela Principal deverá acessar o menu Cadastro Individual em sequência de cada cadastro. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<Maria da Silva> para no campo “Nome Completo”. Digite<12/04/1990> para no campo “Data de Nascimento”. Digite<Mari> para no campo “Apelido”. Digite<Feminina> para no campo “Sexo”. | 1. Deverá exibir uma mensagem: “**Cadastro Individual realizado com sucesso!**”. |
| 1. O usuário deverá clicar no botão[Ok]. | 1. O usuário João, salva o cadastro Individual clicando em [**Salvar**]. |
| 1. O usuário João, poderá atualizar o cadastro. | 1. O usuário João, poderá editar o cadastro. |
| 1. O usuário poderá clicar no botão [Atualizar] o cadastro. | 1. O usuário poderá clicar no botão [**Editar**] o cadastro. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro atualizado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Cadastro editado!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O sistema deverá armazenar os dados no sistema, e salvar os dados coletados na Tela de Registro aonde o usuário poderá manipular as informações coletadas. |

**CT02 – Cadastro Individual (Negativo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo desse Caso de Teste é mostrar um forma anormal que o requisito poderá acontecer com o sistema Sipacs |
| Pré-Condição: | O usuário João já vai estar logado no sistema como um usuário registrado.  Não deverá ser cadastro o mesmo domicilio em um único dia data e hora no sistema. |
| Ações: | **Resultados Esperados**: Deverá ser apresentado uma tela de cadastro, com campos que serão preenchidos: “**Nome** **Completo**”, “**Data** **de** **Nascimento**”, “**Apelido**” e “**Sexo**”. |
| 1. O usuário na Tela Principal deverá acessar o menu Cadastro Individual em sequência de cada cadastro. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<Maria> para no campo “Nome Completo”. Digite<12/04/1990> para no campo “Data de Nascimento”. Digite<Mari> para no campo “Apelido”. Digite<Feminina> para no campo “Sexo”. | 1. Deverá exibir uma mensagem: “**Cadastro não realizou a operação, digite novamente os campos corretamente!**”. |
| 1. O usuário João, deverá clicar no botão[Ok] na mensagem. | 1. O usuário João, salva o cadastro Individual clicando em [**Salvar**]. |
| 1. O usuário João, poderá atualizar o Cadastro. | 1. O usuário João, poderá editar o **cadastro**. |
| 1. O usuário poderá clicar no botão [Atualizar] o cadastro. | 1. O usuário poderá clicar no botão [**Editar**] o cadastro. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro atualizado com sucesso!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Cadastro editado!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O sistema poderá não poderá exibir mensagem de erro ao usuário, podendo não realizar corretamente o requisito no sistema. |

**CT01 – Cadastro de Visitas (Positivo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo da Caso de Teste cadastro de visitas é verificar se é possível cadastrar todos os dados no sistema para armazenar e salvar os dados no sistema Sipacs. |
| Pré-Condição: | O usuário João já estar logado no sistema como um usuário. |
| Ações: | Resultados Esperados: Deverá ser apresentado uma tela de cadastro, com campos que serão preenchidos: “**Código**”, “**Data da Visita**”, “**Visita realizada**”. |
| 1. No Cadastro Visitas que estar no menu principal que dá acesso ao usuário autenticado no sistema. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<001> para no campo “Código”. Digite<12/04/2015> para no campo “Data da Visita”. Digite<Sim> para no campo “Visita Realizada”. | 1. Deverá exibir uma mensagem: “**Cadastro de Visitas cadastrado com sucesso!**”. |
| 1. O usuário deverá clicar no botão[Ok]. | 1. O usuário João, salva o cadastro de visitas clicando em [**Salvar**]. |
| 1. O usuário João, poderá atualizar o Cadastro. | 1. O usuário João, poderá editar o **Cadastro.** |
| 1. O usuário poderá clicar no botão [Atualizar] o cadastro. | 1. O usuário poderá clicar no botão [**Editar**] o cadastro. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro atualizado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Cadastro editado!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O sistema deverá armazenar os dados no sistema, e salvar os dados coletados na Tela de Registro aonde o usuário poderá manipular as informações coletadas. |

**CT02 – Cadastro de Visitas (Negativo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo do Caso de Teste é demostrar da forma anormal que o requisito poderá apresentar na manipulação com o usuário. |
| Pré-Condição: | O usuário João já estar logado no sistema como um usuário. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado uma tela de cadastro, com campos que serão preenchidos: “**Código**”, “**Data da Visita**”, “**Visita realizada**”. |
| 1. No Cadastro Visitas que estar no menu principal que dá acesso ao usuário autenticado no sistema. Exibindo a tela de cadastro de visitas | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<001> para no campo “Código”. Digite<12/04/2015> para no campo “Data da Visita”. Digite<Sim> para no campo “Visita Realizada”. | 1. Deverá exibir uma mensagem: “**Cadastro não realizou a operação, digite novamente os campos corretamente!**”. |
| 1. O usuário deverá clicar no botão[Ok]. | 1. O usuário João, salva o cadastro domiciliar clicando em [**Salvar**]. |
| 1. O usuário João, poderá atualizar o Cadastro. | 1. O usuário João, poderá editar o **Cadastro.** |
| 1. O usuário poderá clicar no botão [Atualizar] o cadastro. | 1. O usuário poderá clicar no botão [**Editar**] o cadastro. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Cadastro atualizado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Cadastro editado!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O sistema poderá não poderá exibir mensagem de erro ao usuário, podendo não realizar corretamente o requisito no sistema. |

**CT01 – Agenda (Positivo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo de Caso de Teste Agenda é verificar se é possível Agendar todos os dados no sistema. |
| Pré-Condição: | O usuário João já estar logado no sistema como um usuário. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado uma **tela de Agenda**, com campos que serão preenchidos: “**Nome**”, “**Endereço**”, “**Numero**” e “**Data**”. |
| 1. O usuário na Tela Principal deverá acessar o menu Agenda em sequência de cada menu. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<Maria da Silva> para no campo “Nome”. Digite<Rua São Francisco> para no campo “Endereço”. Digite<35> para no campo “Numero”. Digite <22/01/2015> para no campo “Data”. | 1. Deverá exibir uma mensagem: “**Agendamento realizado com sucesso!**”. |
| 1. O usuário deverá clicar no botão[Ok]. | 1. O usuário João, salva o Agenda de visitas clicando em [**Salvar**]. |
| 1. O usuário João, poderá editar o agendamento. | 1. O usuário João, poderá excluir o **agendamento.** |
| 1. O usuário poderá clicar no botão [Editar] o Agendamento. | 1. O usuário poderá clicar no botão [**Excluir**] o Agendamento. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Agendamento editado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Agendamento excluído!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O usuário realizará o agendamento salvado as informações, depois podendo verificar os agendamento salvos no sistema e podendo manipular os agendamentos com excluir ou atualizar. |

**CT02 – Agenda (Negativo)**

|  |  |
| --- | --- |
| Descrição: | O objetivo de CT é verificar a impossibilidade deletar o agendamento do sistema. |
| Pré-Condição: | O usuário deverá estar logado no sistema como um usuário. |
| Ações: | **Resultados Esperados:** Deverá ser apresentado uma **tela de Agenda**, com campos que serão preenchidos: “**Nome**”, “**Endereço**”, “**Numero**” e “**Data**”. |
| 1. O usuário na Tela Principal deverá acessar o menu Agenda em sequência de cada menu. | 1. Deverá exibir uma tela com campos para o usuário preencher. |
| 1. Digite<Maria > para no campo “Nome”. Digite<Rua São Francisco> para no campo “Endereço”. Digite<35> para no campo “Numero”. Digite <22/01/2015> para no campo “Data”. | 1. Deverá exibir uma mensagem: “**Agendamento invalido, digite novamente os campos corretamente!**”. |
| 1. O usuário deverá clicar no botão[Ok]. | 1. O usuário João, salva o cadastro domiciliar clicando em [**Salvar**]. |
| 1. O usuário João, poderá editar o Agendamento. | 1. O usuário João, poderá excluir o **Agendamento.** |
| 1. O usuário poderá clicar no botão [Editar] o Agendamento. | 1. O usuário poderá clicar no botão [**Excluir**] o Agendamento. |
| 1. O sistema deverá exibir uma mensagem de confirmação: “Agendamento editado!”. Usuário clica em [Ok]. | 1. O sistema deverá exibir uma mensagem de confirmação: **“Agendamento excluído!**”. Usuário clica em [**Ok**]. |
| Pós-Condições: | O requisito poderá da falha na hora da ação, podendo não mostra a mensagem ao usuário ou armazenando as informações no sistema. |