

**UNIVERSIDADE SALGADO DE OLIVEIRA**  
**PRÓ-REITORIA ACADÊMICA**  
**CURSO DE ANÁLISE DE SISTEMAS**

RAFAEL GOUVÊA MARTINS MALATESTA  
TALES ARAÚJO MENDONÇA

**SGCE – SISTEMA GERENCIAL CASA ESPIRITA**

Juiz de Fora  
2011

RAFAEL GOUVÊA MARTINS MALATESTA

TALES ARAÚJO MENDONÇA

## **SGCE – SISTEMA GERENCIAL CASA ESPIRITA**

Projeto de *Software* apresentado ao Curso de Análise de Sistemas da Universidade Salgado de Oliveira – UNIVERSO, como parte dos requisitos para conclusão do curso.

Juiz de Fora

2011

RAFAEL GOUVÊA MARTINS MALATESTA

TALES ARAÚJO MENDONÇA

**SGCE – SISTEMA GERENCIAL CASA ESPIRITA**

Projeto de *Software* apresentado ao Curso de Análise de Sistemas da Universidade Salgado de Oliveira – UNIVERSO, como parte dos requisitos para conclusão do curso.

---

RAFAEL GOUVÊA MARTINS MALATESTA

---

TALES ARAÚJO MENDONÇA

Juiz de Fora

2011

Se os fatos não se encaixam na teoria, modifique os fatos.

**Albert Einstein**

## ACOMPANHAMENTO DAS VERSÕES

Durante o desenvolvimento do projeto, com o intuito de aperfeiçoar o trabalho, foram criadas versões para que acompanham o ciclo de desenvolvimento do projeto, o qual é mostrado no quadro a seguir.

Data	Versão	Descrição	Autores
12/02/2011	1.0	Levantamento preliminar de requisitos	Rafael Malatesta e Tales Araújo
14/02/2011	1.1	Contextualização	Rafael Malatesta e Tales Araújo
11/03/2011	2.0	Planejamento	Rafael Malatesta e Tales Araújo
27/03/2011	2.1	Ajustes de ciclo de vida, cronograma e planos de monitoramento e controle	Rafael Malatesta e Tales Araújo
02/04/2011	2.2	Especificação de Requisitos	Rafael Malatesta e Tales Araújo
03/04/2011	2.3	Primeiro Monitoramento e Controle	Rafael Malatesta e Tales Araújo
23/04/2011	2.4	Ajustes dos Pontos de Função e cronograma	Rafael Malatesta e Tales Araújo
20/05/2011	3.0	Segundo Monitoramento e Controle, Fechamento Projetos 1	Rafael Malatesta e Tales Araújo
27/05/2011	3.1	Diagramas de Tabela Relacional, Diagrama de Classe e Diagrama de Sequência	Rafael Malatesta e Tales Araújo
06/06/2011	4.0	Início do desenvolvimento do sistema	Rafael Malatesta e Tales Araújo
16/06/2011	4.1	Testes preliminares	Rafael Malatesta e Tales Araújo

Quadro 1 – Acompanhamento de versões

## LISTA DE FIGURAS

Figura 1 – Modelo Clássico ou cascata.....	24
Figura 2 – Estrutura Analítica do Projeto.....	27
Figura 3 – Plano organizacional.....	37
Figura 4 – Cronograma de atividades.....	41
Figura 5 – Gráfico de Gantt.....	43
Figura 6 – Mapa de atores.....	65
Figura 7 – Diagrama Usuário.....	66
Figura 8 – Diagrama Usuário (Perfil Coordenador).....	67
Figura 9 – Diagrama Usuário (Perfil Estoquista).....	67
Figura 10 – Caso de uso família.....	68
Figura 11 – Caso de uso mantimento.....	69
Figura 12 – Caso de uso Usuário.....	69
Figura 13 – Caso de uso voluntário.....	70
Figura 14 – Caso de uso Log.....	71
Figura 15 – Moledo conceitual de dados (DER).....	112
Figura 16 – Novo cronograma de atividades.....	127
Figura 17 – Novo Gráfico de Gantt.....	129
Figura 18 – Diagrama de Classe.....	135
Figura 19 – Diagrama de Sequencia: Cadastrar Usuário.....	137
Figura 20 – Diagrama de Sequencia: Alterar Usuário.....	138
Figura 21 – Diagrama de Sequencia: Excluir Usuário.....	139
Figura 22 – Diagrama de Sequencia: Consultar Usuário.....	140
Figura 23 – Diagrama de Sequencia: Incluir Estoque.....	141
Figura 24 – Diagrama de Sequencia: Alterar Estoque.....	142
Figura 25 – Diagrama de Sequencia: Excluir Estoque.....	143
Figura 26 – Diagrama de Sequencia: Consultar Estoque.....	144
Figura 27 – Diagrama de Sequencia: Relatório de Vencimentos Estoque.....	145
Figura 28 – Diagrama de Sequencia: Relatório de Cestas Disponíveis.....	146
Figura 29 – Diagrama de Sequencia: Relatório de Itens Pendentes na Cesta.....	147
Figura 30 – Diagrama de Sequencia: Cadastro Definição Cesta.....	148
Figura 31 – Diagrama de Sequencia: Alterar Definição Cesta.....	149
Figura 32 – Diagrama de Sequencia: Excluir Definição Cesta.....	150
Figura 33 – Diagrama de Sequencia: Consultar Definição Cesta.....	151

Figura 34 – Diagrama de Sequencia: Cadastrar Família.....	152
Figura 35 – Diagrama de Sequencia: Cadastrar Frequência de Família.....	153
Figura 36 – Diagrama de Sequencia: Alterar Família.....	154
Figura 37 – Diagrama de Sequencia: Alterar Frequência Família.....	155
Figura 38 – Diagrama de Sequencia: Excluir Família.....	156
Figura 39 – Diagrama de Sequencia: Consultar Família.....	157
Figura 40 – Diagrama de Sequencia: Consultar Frequência Família.....	158
Figura 41 – Diagrama de Sequencia: Relatório Vencimento Matrícula.....	159
Figura 42 – Diagrama de Sequencia: Relatório Família Apta.....	160
Figura 43 – Diagrama de Sequencia: Efetuar Login.....	161
Figura 44 – Diagrama de Sequencia: Recuperar Senha.....	162
Figura 45 – Diagrama de Sequencia: Armazenar Log Sistema.....	163
Figura 46 – Diagrama de Sequencia: Consultar Voluntário.....	164
Figura 47 – Diagrama de Sequencia: Cadastrar Voluntário.....	165
Figura 48 – Diagrama de Sequencia: Alterar Voluntário.....	166
Figura 49 – Diagrama de Sequencia: Excluir Voluntário.....	167
Figura 50 – Diagrama de Sequencia: Consultar Voluntário.....	168
Figura 51 – Diagrama de Sequencia: Relatório Voluntário.....	169
Figura 52 – DTR.....	171
Figura 53 – Requisição básica de MVC.....	174

## LISTA DE QUADROS

Quadro 1 – Acompanhamento de versões.....	5
Quadro 2 – Entradas Externas.....	30
Quadro 3 – Saídas Externas.....	31
Quadro 4 – Arquivos Lógicos Internos.....	31
Quadro 5 – Consultas Externas.....	32
Quadro 6 – Cálculo do FPA não ajustado.....	33
Quadro 7 – Somatório dos níveis de Influência.....	34
Quadro 8 – Estimativa de esforço.....	36
Quadro 9 – Pontos de Monitoramento e Controle.....	39
Quadro 10 – Funções e responsabilidades.....	44
Quadro 11 – Custo de software.....	46
Quadro 12 – Custo de Hardware.....	47
Quadro 13 – Custo de Mão de Obra.....	47
Quadro 14 – Custo com outras despesas.....	49
Quadro 15 – Custo total.....	49
Quadro 16 – Histórico do Controle de Versão.....	56
Quadro 17 – Descrição dos Atores.....	64
Quadro 18 – Perfis dos Usuários do Sistema.....	68
Quadro 19 – Atributos do Usuário.....	112
Quadro 20 – Atributos do Voluntário.....	113
Quadro 21 – Atributos da Família.....	114
Quadro 22 – Atributos de Definição de Cesta.....	120
Quadro 23 – Atributos de Estoque.....	120
Quadro 24 – Atributos de Frequência.....	121
Quadro 25 – Atributos da Cesta.....	121
Quadro 26 – Atributos de log.....	122
Quadro 27 – Alterações nos pontos de função.....	124
Quadro 28 – Novo cálculo FPA não-ajustado.....	125
Quadro 29 – Nova estimativa de esforço.....	126
Quadro 30 – Modelo de Classe UML.....	133
Quadro 31 – Simbologia de explicação do diagrama de sequencia.....	136
Quadro 32 – estrutura de dados da tabela logs.....	175
Quadro 33 – estrutura de dados da tabela usuarios.....	176



Quadro 34 – estrutura de dados da tabela voluntarios.....	177
Quadro 35 – estrutura de dados da tabela definicoescestas.....	177
Quadro 36 – estrutura de dados da tabela estoques.....	178
Quadro 37 – estrutura de dados da tabela cestas_estoques.....	178
Quadro 38 – estrutura de dados da tabela cestas.....	179
Quadro 39 – estrutura de dados da tabela frequencias.....	179
Quadro 40 – estrutura de dados da tabela familias.....	179
Quadro 41 – estrutura de dados da tabela pessoas.....	180
Quadro 42 – estrutura de dados da tabela respostas.....	181
Quadro 43 – estrutura de dados da tabela questionarios.....	181
Quadro 44 – Funcionalidades implementadas.....	193
Quadro 45 – Programas desenvolvidos.....	194
Quadro 46 – Prazo do Primeiro Monitoramento e Controle.....	198
Quadro 47 – Custo do Primeiro Monitoramento e Controle.....	199
Quadro 48 – Prazo do Segundo Monitoramento e Controle.....	201
Quadro 49 – Custo do Segundo Monitoramento e Controle.....	202
Quadro 50 – Prazo do Segundo Monitoramento e Controle.....	203
Quadro 51 – Custo do Segundo Monitoramento e Controle.....	204
Quadro 52 – Tabela de Complexidade de Entrada.....	209
Quadro 53 – Tabela de Complexidade de Saída.....	209
Quadro 54 – Tabela de Complexidade ALI.....	209
Quadro 55 – Tabela de Complexidade AIE.....	210
Quadro 56 – Tabela de Complexidade de Consulta.....	210
Quadro 57 – Tabela de pesos para FPA.....	210

## **LISTA DE ABREVIATURAS E SIGLAS**

DAS – Departamento de Assistência Social

FPA – Function Points Analysis

SEJA – Sociedade Espirita Joanna de Ângelis

SGCE – Sistema Gerencial Casa Espirita

WBS – Work Breakdown Structure

UML – Unified Modeling Language

DER – Diagrama de Entidades Relacionais

DTR – Diagrama de Tabelas Relacionais

MVC – Model View and Controller

## SUMÁRIO

1 CONTEXTUALIZAÇÃO.....	16
1.1 Introdução.....	16
1.2 Objetivo.....	17
1.3 Motivação.....	18
1.4 Justificativa.....	18
1.5 Levantamento preliminar de requisitos.....	19
1.5.1 Cadastramento das famílias assistidas.....	19
1.5.2 Cadastramento dos voluntários cadastrados.....	20
1.5.3 Gerenciamento do controle de estoque.....	20
1.5.4 Funcionalidades gerais do Sistema.....	21
1.6 Outras questões do projeto.....	21
2 PLANEJAMENTO DO PROJETO.....	22
2.1 Declaração do escopo.....	22
2.2 Plano do processo de desenvolvimento.....	23
2.3 Metodologia de desenvolvimento.....	25
2.4 Estrutura analítica do projeto.....	25
2.5 Estimativa de tamanho, esforço, prazo.....	28
2.5.1 Identificação das funções da aplicação.....	28
2.5.1.1 Entradas Externas.....	28
2.5.1.2 Saídas Externas.....	29
2.5.1.3 Arquivos Lógicos Internos.....	29
2.5.1.4 Arquivos de Interface Externa.....	29
2.5.1.5 Consulta Externa.....	30
2.5.2 Definição da complexidade das funcionalidades.....	30
2.5.2.1 Entradas Externas.....	30
2.5.2.2 Saídas Externas.....	31
2.5.2.3 Arquivos Lógicos Internos.....	31
2.5.2.4 Consultas Externas.....	32
2.5.3 Cálculo dos pesos (FPA Não ajustados).....	33
2.5.4 Cálculo do Fator de ajuste e FPA ajustado.....	33
2.5.5 Estimativas de Esforço e Prazo.....	36
2.6 Plano de Organização.....	37
2.7 Plano de monitoramento e controle.....	38
2.7.1 Introdução.....	38
2.7.2 Custo.....	39

2.7.3 Prazo.....	39
2.7.4 Produção.....	40
2.7.5 Risco.....	40
2.8 Cronograma.....	40
2.8.1 Gráfico de Gantt .....	41
2.9 Plano de recursos humanos.....	44
2.10 Plano de recursos gerais.....	44
2.10.1 Hardware.....	44
2.10.2 Software.....	45
2.10.3 Recursos adicionais.....	46
2.11 Plano de custos.....	46
2.11.2 Custos de Hardware.....	47
2.11.3 Custos de Mão de Obra.....	47
2.11.4 Custos com outras despesas.....	48
2.11.5 Total Geral das Despesas do projeto.....	49
2.12 Plano de gerencia de dados.....	50
2.13 Plano de medição e análise.....	50
2.14 Plano de gerencia de configuração.....	50
2.15 Plano de gerencia de riscos.....	50
2.16 Plano de garantia da qualidade.....	50
2.17 Plano de verificação.....	51
2.18 Plano de validação.....	51
2.19 Plano de testes.....	51
2.20 Plano de treinamento.....	52
2.21 Plano de implantação.....	52
2.21.1 Introdução.....	52
2.21.2 Especificação do hardware e do software.....	53
2.21.2.1 Hardware.....	53
2.21.2.2 Software.....	54
2.21.3 Guia da instalação.....	54
2.22 Observações complementares.....	55
3 ESPECIFICAÇÃO DE REQUISITOS.....	55
3.1 Requisitos do cliente.....	56
3.1.1 Descrição geral da aplicação.....	56
3.1.1.1 Descrição da necessidade.....	56
3.1.1.2 Objetivo.....	57

3.1.1.3 Escopo da Aplicação.....	57
3.1.2 Descrição geral do cliente.....	58
3.1.3 Lista de requisitos do cliente.....	58
3.2 Requisitos do software.....	59
3.2.1 - Fronteiras do software.....	59
3.2.2 Itens de software.....	59
3.2.3 Lista de requisitos não funcionais.....	59
3.2.4 Requisitos funcionais.....	61
3.2.4.1 Lista dos requisitos funcionais.....	62
3.2.4.2 Descrição dos atores.....	64
3.2.4.3 Descrição de casos de uso.....	68
3.2.5.1 Lista de requisitos de Dados.....	109
3.2.5.2 Modelo Conceitual de Dados.....	112
3.2.5.3 Entidades e Atributos.....	112
3.2.6 Melhoramentos Previstos.....	122
3.3 Revisão de estimativas.....	123
3.3.1 Considerações preliminares.....	123
3.3.2 Estimativa de tamanho de Software.....	123
3.3.3 Estimativa de esforço .....	123
3.3.4 Estimativa de prazo.....	123
3.3.5 Cronograma revisado.....	128
3.3.7 Considerações finais sobre a revisão de estimativas .....	130
4. MODELAGEM DE ANÁLISE.....	130
4.1 Considerações preliminares.....	130
4.2 Metodologia adotada.....	131
4.3 Diagrama de caso de uso.....	131
4.4 Diagrama de classe.....	132
4.4.1 Classe .....	132
4.4.2 Diagrama de classes .....	133
4.4 Diagrama de sequência.....	136
4.5 Diagrama de tabelas relacionais (DTR) .....	170
5 MODELAGEM DE PROJETO.....	172
5.1 Considerações Iniciais.....	172
5.2 Arquitetura do Software.....	172
5.2.1 Framework CakePHP.....	173
5.2.2 MVC.....	174
5.2.3 Ajax.....	174

5.3 Especificação de Interface.....	175
5.4 Estrutura de Dados.....	175
5.4.1 Modelagem Física de Banco de Dados .....	175
5.4.2 Estrutura do Banco de Dados.....	182
5.5 Detalhes Procedimentais.....	191
5.5.1 Login.....	191
5.5.2 Máscaras e validações .....	191
5.6 Observações Complementares.....	192
6 PROCESSO DE TESTES.....	192
7 PRELIMINARES DA IMPLEMENTAÇÃO.....	192
7.1 Funcionalidades Implementadas.....	193
7.2 Programas Desenvolvidos.....	194
7.3 Resultado dos Testes.....	195
7.3.1 Caso de uso UC01 – Cadastrar Usuário .....	195
7.3.2 Caso de uso UC02 – Alterar Usuário .....	195
7.3.3 Caso de uso UC03 – Excluir Usuário .....	195
7.3.4 Caso de uso UC04 – Consulta Usuário .....	195
7.3.5 Caso de uso UC8 – Consultar estoque .....	196
7.3.6 Caso de uso UC13 – Cadastrar Definição de Cesta .....	196
7.3.7 Caso de uso UC14 – Alterar Definição de Cesta .....	196
7.3.8 Caso de uso UC15 – Excluir Definição de Cesta .....	196
7.3.9 Caso de uso UC26 – Efetuar Login .....	197
7.3.10 Caso de uso UC28 – Armazenar log .....	197
7.3.11 Caso de uso UC29 – Consultar log do Sistema .....	197
7.4 Observações Complementares.....	197
8 MONITORAMENTO E CONTROLE.....	198
8.1 Considerações Preliminares.....	198
8.2 Primeiro monitoramento e controle.....	198
8.2.1 Prazo.....	198
8.2.2 Produção.....	199
8.2.3 Custos.....	199
8.2.4 Riscos .....	200
8.2.5 Fechamento do Primeiro Monitoramento e Controle .....	200
8.3 Segundo monitoramento e controle.....	200
8.3.1 Prazo.....	200

8.3.2 Produção.....	202
8.3.3 Custos.....	202
8.3.4 Riscos .....	202
8.3.5 Fechamento do Segundo Monitoramento e Controle .....	203
8.4 Terceiro monitoramento e controle.....	203
8.4.1 Prazo.....	203
8.4.2 Produção.....	204
8.4.3 Custos.....	204
8.4.4 Riscos .....	205
8.4.5 Fechamento do Terceiro Monitoramento e Controle .....	205
9 CONSIDERAÇÕES FINAIS.....	205
BIBLIOGRAFIA.....	208
Anexo I – Tabelas Relativas ao FPA.....	209

## 1 CONTEXTUALIZAÇÃO

### 1.1 Introdução

O projeto, de caráter assistencial, objetiva-se a ser um complemento para a conclusão do curso de Análise de Sistemas da Universidade Salgado de Oliveira em Juiz de Fora, onde abordará todas as etapas em que o projeto será inserido, desde a contextualização do projeto, passando pelo seu planejamento, abordando as especificações de requisitos, modelagem de análise, monitoramento e controle, até a sua finalização com as considerações finais.

O desenvolvimento do SGCE – Sistema Gerencial Casa Espírita –, nome dado ao *software* que irá gerenciar o DAS – Departamento de Assistência Social –, monitorando o controle das famílias assistidas cadastradas, bem como a frequência de suas participações perante as palestras realizadas, o aconselhamento as famílias, o controle das doações em mantimentos e controle do estoque de mantimentos; também irá controlar o cadastro dos voluntários que se propuser a colaborar com a associação, contribuindo com dinheiro em espécie, trabalho voluntário ou doação de roupas e mantimentos.

A SEJA é uma instituição sem fins lucrativos, fundada por diversos irmãos de ideal espírita em 24 de janeiro de 1986, em Juiz de Fora- MG, situada na rua Severino Belfort, no bairro Bairro; possuindo diversos núcleos pelo país. Sua missão é promover o estudo metódico do espiritismo nos aspectos filosófico-científico e religioso, como foi codificado por Allan Kardec; promover a difusão das obras da codificação e os livros subsidiários fiéis a ela; promover a prática da mediunidade segundo a orientação kardequiana; promover atividade de assistência social espírita, assegurando suas características filantrópicas, conjugando a ajuda material e espiritual, com orientação evangélico-doutrinária; promover a formação do homem, em todas as faixas etárias.

Há uma proposta de criação de uma oficina de informática, para atender as famílias carentes que abstêm de um computador para estar conhecendo, pelo mesmo, o básico de seu funcionamento.

Com a implantação deste projeto, os voluntários responsáveis pelo DAS terão



mais tempo para estar contribuindo em outros projetos e estar a criar novas oficinas com intuito de oferecer mais opções de profissionalização para as entidades que carecem de obter tais proveitos de maneiras convencionais e sem ajuda dos demais.

## 1.2 Objetivo

O SGCE – Sistema de Gerenciamento Centro Espírita – nome referente ao *software* desenvolvido para a instituição filantrópica, visa automatizar todos os procedimentos que atualmente são realizados de forma mecânica e não informatizada.

O DAS é responsável pela orientação e pela coordenação do serviço de assistência e promoção social espírita no âmbito de ação da SEJA. Suas finalidades são propor estratégias, projetos, programas e diretrizes operacionais para implantação, desenvolvimento e manutenção do serviço de assistência e promoção social espírita, consoante orientações da Federação Espírita Brasileira e os normativos legais que disciplinam as atividades de assistência social, e assegurar as características beneficentes, preventiva e promocional do serviço de assistência e promoção social espírita, fazendo com que esse serviço se desenvolva concomitantemente com o atendimento às necessidades de evangelização.

São realizadas entrevistas, triagens, fichas individuais das famílias e de frequência, controle de estoque e cadastro de voluntários.

A proposta desse *software* é tornar informatizado todos os serviços oferecidos pelo DAS, de modo que haja um controle rígido das doações que chegam na casa até a distribuição das mesmas; das famílias em conformidade para que possam receber as doações; do controle de estoque para que não falte mantimentos para a criação das cestas básicas, nem que haja desperdício do mesmo por vencimento de validade.

O *software* proposto será desenvolvido de forma a agregar facilidade – para qualquer voluntário que necessite trabalhar e de forma simples – gerando conhecimento para que com isso, as informações fornecidas atendam em uma melhor tomada de decisão.

### **1.3 Motivação**

A Universidade Salgado de Oliveira proporciona ao aluno de Análise de sistemas colocar em prática todo o conteúdo programático que é visto durante o curso, para a fim de desenvolver um projeto único que apresenta como objetivo a colaboração de um programa de computador que será atendido a determinado público filantrópico ou mesmo a própria Universidade. Esta particularidade já serve como motivação para que o aluno possa comprometer-se com o desenvolvimento de um projeto para ajudar aqueles que não visam lucro e estão a dispor de outros para poderem ajudá-los.

Visando esse comprometimento, nos é favorecido a busca pela aprendizagem e conhecimento, em troca ajudaremos a quem precisa ser ajudado para colaborar com o próximo. Tudo isso é um cliço de parceria que deveria ser levado para toda uma vida e não deixando apenas na faculdade. Não temos apenas o prazer em realizar tal feito, mas o prazer de ver o quão gratificados e felizes nossos clientes estão.

### **1.4 Justificativa**

A SEJA é uma instituição que não visa lucro, visa melhorar a condição de vida das pessoas que não possuem recursos ou possuem poucos recursos para sobreviver.

Essa sociedade espírita busca não só o melhoramento das famílias que nela estão cadastradas, mas visa também ajudar psicologicamente e espiritualmente a esses necessitados, para que possam obter uma evolução de conhecimento, de vida.

Para todas as famílias carentes que estão estabelecidas no centro espírita são ofertadas oficinas com atividades diversificadas para que, os que procuram um trabalho para se auto sustentar, possam ser agradecidos pelo tempo que voluntários doam para o melhoramento da vida dos menos favorecidos.

Atualmente o centro dispõe de alguns computadores que foram doados, mas

pela falta de alguém com conhecimento para remanejar e colocar os computadores para funcionar, os mesmos encontram-se guardados, aguardando serem montados e postos para utilização.

Conversamos com nosso cliente e ficamos responsáveis, também, por não só colocar os computadores parados em funcionamento como também conseguir mais doações para montar uma possível sala de informática para pessoas carentes.

Consolidando a concretização do *software* para esta instituição, os voluntários que ali residem poderão dispor de mais tempo, tanto com seus familiares como para a realização de outras tarefas no centro, como a criação de novas oficinas.

## **1.5 Levantamento preliminar de requisitos**

Ao iniciar este projeto, a equipe de desenvolvimento tem a necessidade de fazer um levantamento preliminar das necessidades que o *software* irá atender. Este levantamento deu-se através de reuniões realizadas com os representantes e alguns colaboradores da SEJA em Juiz de Fora em uma reunião pautada e nas conversas, foram definidos os seguintes requisitos preliminares:

### ***1.5.1 Cadastramento das famílias assistidas***

- Será realizada uma visita à família para verificar se a mesma possa estar apta ou não para receber a doação de cesta básica mensal. Para isso é feito um cadastro completo referente a família, contendo a identificação, composição familiar e situação de moradia;
- Cada família cadastrada possui um número único de matrícula que terá prazo de validade de até 1 ano a contar da data do cadastro. Vencido o período de 1 ano, a família irá ganhar um status de pendencia, aguardando uma nova visita domiciliar para atualização dos dados cadastrais;
- A família para receber o status de apta ou não apta precisa preencher alguns

requisitos preestabelecidos, como: frequência continua nas atividades realizadas aos sábados. No caso de ausência, levam um documento explicando a falta, a fim de que a mesma seja justificada. Após três ausências sem justificativa, perdem o vínculo.

#### *1.5.2 Cadastramento dos voluntários cadastrados*

- Será feito um cadastro simples do voluntário, que irá conter o seu nome, RG, CPF, endereço, telefone; se deseja ou não ser colaborador com mantimento, lanche, vestuário ou financeiramente. Tendo o aceite, o mesmo terá que assinar um termo referente a lei do voluntário, nº 9.608 de 18 de fevereiro de 1998, mediante a assinatura de duas testemunhas;

#### *1.5.3 Gerenciamento do controle de estoque*

- O mantimento é recebido pela instituição e cadastrado no sistema assim que chega, o qual é enviado para o estoque. No último sábado do mês é feita a montagem das cestas para que sejam distribuídas para as famílias carentes cadastradas no sistema e aptas a receber;
- A cesta básica é composta de mantimentos necessários para atender as famílias com os pré-requisitos cadastrados. Caso falte algum item necessário no estoque para a montagem de uma cesta, o sistema deverá informar qual item e a quantidade que precisa para que possa ser atingido a base para atender a todas as famílias cadastradas e aptas;
- O sistema mostrará quantas cestas são possíveis ser montadas com a quantidade de mantimento disponível no estoque;
- O sistema deverá informar os alimentos que estarão com o prazo de validade

vencendo em um mês a partir da data de doação;

#### **1.5.4 Funcionalidades gerais do Sistema**

- Manutenção do cadastro de usuários onde serão restritas algumas funcionalidades dependendo do usuário que estiver acessando o sistema com um login e uma senha única, o(os) usuário(os) administrador(es) poderá(ão) incluir, excluir, alterar, consultar os diversos usuários cadastrados, além de poder utilizar todas as demais funcionalidades;
- O sistema será capaz de armazenar um histórico das transações realizadas no *software*;
- O sistema será capaz de realizar um backup automático do banco de dados que pode ser enviado para um e-mail cadastrado e configurado;
- Geração de relatórios tais como:
  - a) Famílias cadastradas com matrícula, situação atual, data da próxima visita domiciliar.
  - b) Voluntários cadastrados com nome, telefone e o tipo de colaboração.
  - c) Quantidade de cestas básicas completas no estoque.
  - d) Vencimento dos alimentos.

#### **1.6 Outras questões do projeto**

Já existe uma proposta de, após a conclusão deste projeto, criar um sistema completo voltado para todos os centros espíritas e disponibilizar de forma gratuita, tanto o *software* quanto o seu código fonte. Visto que a maioria dos centros espíritas possuem uma padronização baseado na Federação Brasileira Espírita. Dessa forma, este começo de projeto é apenas a ponta de um grande iceberg que irá se concomitar.

## **2 PLANEJAMENTO DO PROJETO**

### **2.1 Declaração do escopo**

O projeto SGCE – Sistema Gerencial Casa Espírita – visa gerenciar o controle de estoque, bem como as famílias assistidas pelo DAS – Departamento de Assistência Social –, gerindo um controle para que o usuário saiba o que entrou e o que saiu de mantimento; o controle da validade dos mantimentos; as famílias que estão aptas a participar da associação e estão em dia com as suas responsabilidades.

O sistema será construído de forma personalizada para atender ao SEJA, seguindo os requisitos preliminares do item 1.5, referente ao capítulo 1. A responsável para que o levantamento de requisitos fosse realizado é a Adriane Gonçalves Guedes, responsável pelo departamento do DAS. Foram realizadas reuniões para recolher informações de maneira mais clara com o objetivo de colaborar com a instituição na construção de um sistema que se adequasse ao funcionamento da mesma.

Para o funcionamento do sistema, serão realizados três tipos de cadastros com informações para movimentação do sistema e um cadastro referente aos usuários que irão utilizar o sistema. O primeiro cadastro é referente as famílias assistidas que terão direito de atendimento pela associação SEJA, como também de uma cesta básica de mantimentos por mês; o segundo cadastro será referente a todos os mantimentos recebidos de doações para a formação da cesta básica; haverá também um controle de validade dos produtos a serem doados; o terceiro cadastro irá conter todos os voluntários que queiram contribuir com a associação da maneira que for necessário para a mesma, ou seja, no setor que estiverem precisando de colaboradores; o cadastro dos utilizadores será gerenciado por um administrador que ficará responsável por atribuir permissões de acesso para novos contribuidores e para funcionários responsáveis pelo centro espírita.

O SGCE irá monitorar o controle das famílias assistidas cadastradas, bem como a frequência de suas participações perante as palestras realizadas – essas

com o intuito de contribuir de forma psicologicamente com as famílias, com aconselhamento das mesmas; irá, também, monitorar o controle das doações em mantimentos e a gerência dos mesmos; também controlará o cadastro dos voluntários que se propuserem a colaborar com a associação, seja da maneira que melhor lhe for conveniente, contribuindo de forma monetária, trabalho voluntário ou doação de roupas e mantimentos.

Como a gestão das informações trabalhadas para gerir a manutenção das famílias assistidas possui um volume moderado e que tais informações são controladas mediante apenas em papeis, faz-se necessário a aplicação para a implantação desse sistema que irá tornar simples a tarefa de manutenção e controle da SEJA.

## **2.2 Plano do processo de desenvolvimento**

Devido a baixa complexidade do projeto e os modelos estudados para alcançar o objetivo do mesmo, entrou-se num consenso de que o modelo clássico, linear ou cascata será de melhor adesão. Por se tratar de um projeto pequeno, a escolha deverá atender bem e de maneira íntegra aos processos de desenvolvimento do *software*, conhecidos também como ciclo de vida.

Um ciclo de vida pode ser entendido como um roteiro de atividades a serem feitos, em geral, de grandes etapas com objetos funcionais na construção do *software*. (TONSIG, 2003).

O plano de desenvolvimento da disciplina está dividida em duas partes, sendo a primeira etapa realizada no primeiro semestre e a segunda etapa realizada no segundo semestre. Por sugestão e decorrência do curso, o modelo clássico ou cascata – como é mais conhecido – se dispõe a atender melhor o nosso projeto e facilitar no fluxo que é seguindo no decorrer das complementações das atividades.

O modelo cascata descreve um método de desenvolvimento que é linear e sequencial, cada fase de desenvolvimento que é completada dá abertura para que uma nova fase comece, e termine. E assim dê início a outra que por sua vez começa e termina, sempre seguindo uma linha contínua de desenvolvimento aonde não há

retorno, todo o projeto tende apenas a seguir em diante, para frente. Cada fase de desenvolvimento prossegue em uma ordem estrita, sem qualquer sobreposição ou passos iterativos. (PRESSMAN, 2006) .

Por se tratar de um projeto que segue a ideia de um modelo sequencial, há uma utilização de retroalimentação para que o projeto seja mais conciso. Averiguando que o cliente nem sempre consegue fornecer todas as informações de imediato, este processo serve para fazer alguns eventuais reparos em inconsistência descritas nas etapas anteriores.

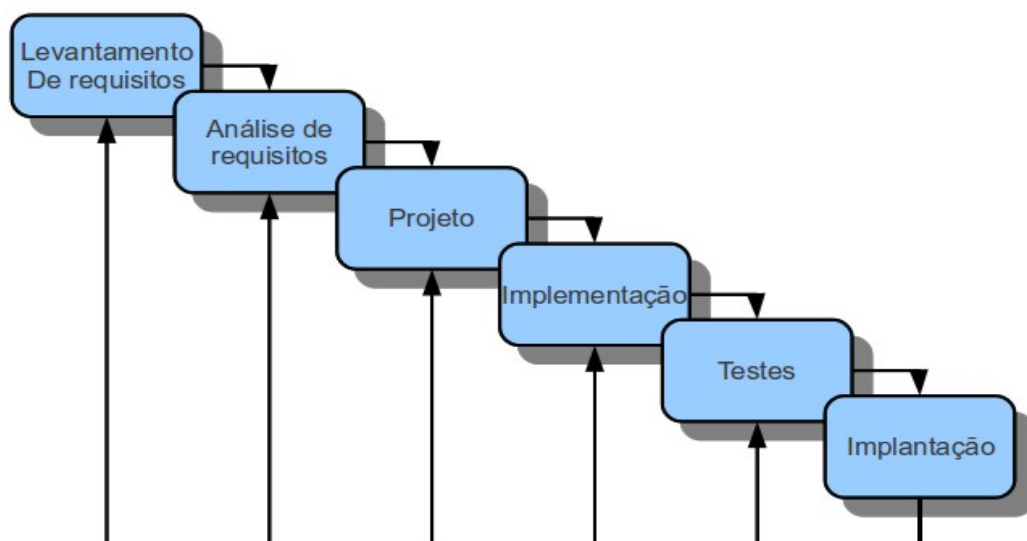


Figura 1 – Modelo Clássico ou cascata

- Levantamento de requisitos: Essa é a etapa que provê recolher informações para que haja um entendimento para a construção do *software*, como, também as necessidades estabelecidas pelos clientes.
- Análise de requisitos: Essa etapa destina-se a construir modelos, detalhando os requisitos coletados e à representação abrangente do projeto para o *software* a ser construído. (PRESSMAN, 2006).
- Projeto: Nessa etapa é realizada a identificação e a descrição das abstrações principais do sistema e suas relações. Nela será descrito como o projeto será implementado, ou seja, a interação entre os módulos do *software* com suas funcionalidades específicas.



- Implementação: Nessa etapa decorre a codificação, criação do código fonte, do *software* que é construído em cima dos modelos criados em etapas anteriores.
- Teste: Após a criação do código fonte que foi realizado na etapa anterior, inicia-se a fase de testes. Para garantir que os requisitos foram atingidos, é realizado testes lógicos internos e aspectos funcionais externos. Finalizando esse procedimento, o programa deverá estar acessível para que seja implantado.
- Implantação: Na implantação, o produto final, terminado, será entregue ao cliente pronto para ser utilizado.

### 2.3 Metodologia de desenvolvimento

Uma metodologia é uma explicação detalhada, minuciosa e exata de toda a ação desenvolvida no método (FACHIN, 2003); é um conjunto de passos e processos bem definidos para que possa desenvolver um sistema.

Para este projeto foi escolhido um método de desenvolvimento baseado na programação orientada a objetos, que caracteriza uma maneira mais próxima de desenvolver um *software* com características do nosso cotidiano, tornando mais confortável de se gerar e reaproveitar o código fonte.

### 2.4 Estrutura analítica do projeto

Uma Estrutura Analítica de Projetos (EAP) ou do inglês *Work Breakdown Structure* (WBS), é uma estrutura hierárquica orientada de forma a cumprir as etapas de cada objeto relacionado na árvore, para que seja possível completar o projeto. Segundo PMBOK (2004), uma WBS é utilizada para segmentar o projeto em pequenos pacotes, chamados pacotes de trabalho, sendo organizada hierarquicamente em formato de árvore. Essa estrutura analítica servirá de base para o acompanhamento de plano de todo o projeto.

É uma ferramenta não utilizada apenas pelo o Gerente de Projetos, mas por toda a equipe de desenvolvimento do sistema, bem como clientes e fornecedores. As pessoas envolvidas nos projetos devem sempre estar cientes para manter a WBS atualizada caso alguma informação seja excluída ou acrescida durante o desenvolvimento do projeto, devendo essas, sempre estar presente na criação das estruturas detalhadas.

PMBOK (2004), relata ainda que, desenvolver uma boa WBS é um dos fundamentos básicos para o gerenciamento do projeto.



Figura 2 – Estrutura Analítica do Projeto.

## 2.5 Estimativa de tamanho, esforço, prazo

Esta é uma fase essencial para um bom planejamento do projeto. Apesar de existirem outras métricas, neste projeto utilizaremos o FPA. De acordo com Longstreet (2008), FPA (Function Point Analysis) ou Ponto de Função, desenvolvido pela IBM em 1979, por Allan Albrecht, é uma unidade de medida de *software* utilizada para estimar o tamanho de um sistema de informação baseando nas funcionalidades percebidas pelo usuário do sistema, independentemente da tecnologia a ser utilizada para implementá-lo.

### 2.5.1 Identificação das funções da aplicação

Os valores necessários para a realização da contagem são as entradas externas, saídas externas, consultas externas, arquivos lógicos internos e arquivos de interface externa.

#### 2.5.1.1 Entradas Externas

Processa as informações vindas de fora do escopo da aplicação.

- Incluir família;
- Excluir família;
- Alterar família;
- Incluir voluntário;
- Excluir voluntário;
- Alterar voluntário;
- Incluir mantimento;
- Excluir mantimento;
- Alterar mantimento;
- Incluir usuário;
- Excluir usuário;

- Alterar usuário;
- Incluir frequência de famílias;
- Excluir frequência de famílias;
- Alterar frequência de famílias;

#### 2.5.1.2 Saídas Externas

Gerar dados ou informação de controle que saem da fronteira do sistema.

- Relatório de vencimento matrícula das famílias;
- Relatório quantidade de cestas básicas prontas;
- Relatório de itens pendentes das cestas básicas;

#### 2.5.1.3 Arquivos Lógicos Internos

Grupo lógico de dados do ponto de vista do usuário, cuja manutenção é feita internamente na aplicação.

- Cadastro das famílias;
- Cadastro dos voluntários;
- Cadastro dos mantimentos;
- Cadastro dos usuários;
- Cadastro de frequência das famílias;

#### 2.5.1.4 Arquivos de Interface Externa

Grupo lógico de dados que passa de uma aplicação para outra, cuja manutenção pertence a outra aplicação. Dentre os requisitos levantados, não inclui arquivos de interface externa.

### 2.5.1.5 Consulta Externa

Processo elementar que envia dados ou informação de controle para fora da fronteira da aplicação.

- Consulta situação das famílias;
- Consulta voluntários;
- Consulta voluntários colaboradores;
- Consulta estoque;
- Consulta usuários do sistema;
- Efetuar login;
- Consulta vencimento dos alimentos;
- Consulta frequência das famílias;

### 2.5.2 Definição da complexidade das funcionalidades

Para realizar os cálculos de grau das funções deste item, foi utilizado os quadros 15,16,17,18 e 19 no Anexo I da página 49.

#### 2.5.2.1 Entradas Externas

Entradas Externas			
Descrição Função	Qtde Arquivos	Qtde Itens	Grau da Função
incluir família	2	35	complexo
excluir família	1	2	simples
alterar família	2	35	complexo
incluir voluntário	1	11	simples
alterar voluntário	1	11	simples
excluir voluntário	1	2	simples
incluir mantimento	1	4	simples
alterar mantimento	1	4	simples
excluir mantimento	1	2	simples
incluir usuário	1	4	simples
alterar usuário	1	4	simples

excluir usuário	1	3	simples
incluir frequência	1	4	simples
alterar frequência	1	4	simples
excluir frequência	1	2	simples
Total formulário Entradas Externas			15
Total Simples			13
Total Médio			0
Total Complexo			2

Quadro 2 – Entradas Externas

### 2.5.2.2 Saídas Externas

Saídas Externas			
Descrição Função	Qtde Arquivos	Qtde Itens	Grau da Função
Relatório vencimento de matrícula famílias	2	5	simples
Relatório parcial das cestas básicas	1	3	simples
Relatório de quantidade de cestas básicas	1	1	simples
Total formulário Saídas Externas			3
Total Simples			3
Total Médio			0
Total Complexo			0

Quadro 3 – Saídas Externas

### 2.5.2.3 Arquivos Lógicos Internos

Arquivos Lógicos Internos			
Descrição Função	Qtde Registros	Qtde Campos	Grau da Função
cadastro família	2	35	médio

cadastro voluntário	1	11	simples
cadastro mantimento	1	4	simples
cadastro usuário	1	4	simples
cadastro frequência família	1	4	simples
Total formulário ALI			5
Total Simples			4
Total Médio			1
Total Complexo			0

Quadro 4 – Arquivos Lógicos Internos

#### 2.5.2.4 Consultas Externas

Consultas Externas			
Descrição Função	Qtde Arquivos	Qtde Campos	Grau da Função
Consulta situação família	2	5	simples
Consulta voluntário	1	3	simples
Consulta voluntário Colaborador	1	4	simples
Consulta mantimento	1	4	simples
Consulta usuário	1	3	simples
Consulta vencimento mantimentos	1	3	simples
Consulta frequência famílias	2	4	simples
Efetuar login	1	2	simples
Total formulário Consultas Externas			8
Total Simples			8
Total Médio			0
Total Complexo			0

Quadro 5 – Consultas Externas



### 2.5.3 Cálculo dos pesos (FPA Não ajustados)

Para os cálculos utilizados neste item, utilizou-se o quadro 20 do Anexo I da página 50.

Pontos de Função Não Ajustados				
Tipo de Função	Complexidade Funcional	Quantidade	Total Complexidade	Total Tipo
Arquivo Lógico Interno	Simple	4	* 7 = 0	28
	Média	1	* 10 = 0	10
	Complexa	0	* 15 = 0	0
Interface Externa	Simple	0	* 5 = 0	0
	Média	0	* 7 = 0	0
	Complexa	0	* 10 = 0	0
Entrada Externa	Simple	13	* 3 =	39
	Média	0	* 4 =	0
	Complexa	2	* 6 =	12
Saída Externa	Simple	3	* 4 =	12
	Média	0	* 5 =	0
	Complexa	0	* 7 =	0
Consulta Externa	Simple	8	* 3 =	24
	Média	0	* 4 =	0
	Complexa	0	* 6 =	0
<b>Total Ponto de Função Não Ajustado</b>				<b>125</b>

Quadro 6 – Cálculo do FPA não ajustado

### 2.5.4 Cálculo do Fator de ajuste e FPA ajustado

Para obter um maior grau de precisão são utilizados fatores de ajustes de valores correspondentes a perguntas cujos valores variam de zero a cinco, onde zero indica não importância e cinco indica que é essencial (PRESSMAN, 2006).

O nível de influência de cada característica é dado por uma escala de 0 a 5:

0 = Não existe nenhuma influência

1 = Pouca influência

2 = Influência moderada

3 = Influência média

4 = Influência significativa

5 = Grande influência

<b>Cálculo do Fator de Ajuste</b>	
<b>Características Gerais das Aplicações</b>	<b>Nível de Influência</b>
Comunicação	4
Funções distribuídas	4
Desempenho	0
Configuração de equipamentos	1
Volume de transações	0
Entrada de dados	5
Interfaces com o usuário	3
Atualizações on-line	3
Processamento complexo	0
Reutilização	1
Facilidade de implantação	1
Facilidade operacional	1
Múltiplos locais	3
Facilidade de mudanças (flexibilidade)	1
<b>Total geral</b>	<b>27</b>

Quadro 7 – Somatório dos níveis de Influência

Comunicação – grau 4: Aplicação é mais do que uma entrada de dados on-line, mas suporta apenas um tipo de protocolo de comunicação.

Funções distribuídas – grau 4: Processamento distribuído e a transferência de dados são on-line e em ambas as direções .

Desempenho – grau 0: Nenhum requerimento especial de performance foi

solicitado pelo usuário.

Configurações de equipamentos – grau 1: Existem restrições operacionais leves. Não é necessário esforço especial para resolver as restrições.

Volume de transações – grau 0: Não estão previstos períodos de picos de volume de transação.

Entrada de dados – grau 5: Mais de 30% das transações são entradas de dados *on-line*.

Interfaces com usuário – grau 3: O sistema possui menus, utilização de *mouse*, *scrolling* vertical, *drop down list*, janelas *pop-up*, *menos numero de telas para executar funções*.

Atualizações *on-line* – grau 3: *Atualização On-Line da maioria dos Arquivos Lógicos Internos*.

Processamento complexo – grau 0: Não há processamento caracterizado como complexo.

Reutilização – grau 1: Código reutilizável é usado somente na aplicação.

Facilidade de implantação – grau 1: Nenhuma consideração especial foi estabelecida pelo usuário, mas procedimentos especiais são necessários na implantação, neste caso disponibilidade de hospedagem na *web e configuração de banco de dados*.

Facilidade operacional – grau 1: Processo de backup e recuperação com a intervenção do operador.

Múltiplos locais – grau 3: A necessidade de múltiplos locais (navegadores) foi considerada no projeto, assim como diferentes *softwares e hardwares*.

Facilidade de mudanças – grau 1: Estão disponíveis facilidades como consulta e relatórios flexíveis para atender necessidades simples.

O fator de ajuste é dado a partir da fórmula pré definida:

Fator Ajuste (FA) =  $0,65 + (0,01 * \Sigma(NI))$ , onde NI = Nível de influência calculado anteriormente no quadro 6.

Aplicando a formula temos:

$$FA = 0.65 + (0,01 * 27)$$

$$FA = 0,92$$

O cálculo dos pontos de função ajustados é o produto do fator de ajuste e dos pontos de função brutos

$$FPA = FPNA * FVA$$

Onde FPA é ponto de função ajustado, FPNA é ponto de função não ajustado

e FVA é fator de valor de ajuste.

O resultado geral será fornecido através do cálculo

$$\text{FPA} = 125 * 0,92$$

$$\text{FPA} = 115$$

cujo resultado é 115, após arredondamento do produto realizado no passo anterior.

### 2.5.5 Estimativas de Esforço e Prazo

<b>Estimativa de Esforço</b>	
Fator hh/pf (Utilização da Linguagem Orientada a Objetos (PHP5) e considerando a produtividade baixa).	7,5
Esforço total em hh(hora/homem) $7,5 * 115 =$	862,5
(Considerando aproximadamente: 3 horas por dia, 3 dias por semana e 4,5 semanas por mês)	
Esforço total em hm (homem/mês) $862,5 / 40,5 =$	21,2962
<b>Estimativa de Prazo</b>	
Equipe composta por 2 integrantes: Rafael Malatesta e Tales Araújo Mendonça	2
Tamanho da Equipe = $((H / M) / \text{Prazo})$	
Prazo = $((H / M) / \text{Prazo}) = 21,2962 / 2 =$	10,64 meses
Prazo em meses (considerado 22 dias por mês)	
hm: Homem-mês - hh: Homem-hora - pf: Ponto-de-função	10 meses e 14 dias

Quadro 8 – Estimativa de esforço

## 2.6 Plano de Organização

O plano de organização é um modelo pela qual a empresa inserida, se organiza. Dividindo responsabilidades, relação do trabalho, através de autoridades.

Conforme o conceito anunciado, foi definido uma estrutura organizacional de desenvolvimento do projeto, através do organograma – Figura 2 –, aonde é detalho o envolvimento pelos envolvido do projeto.

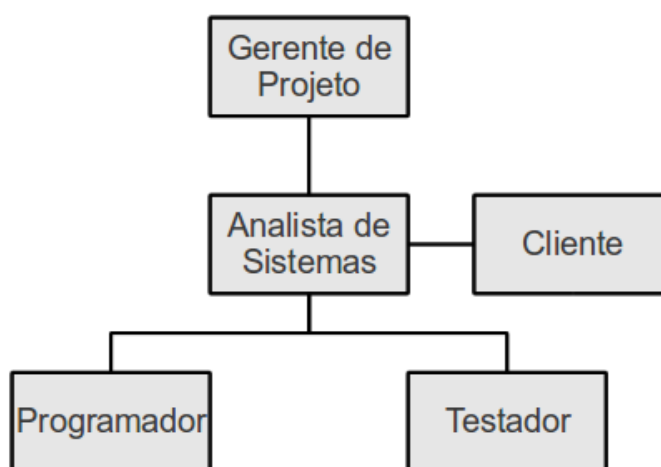


Figura 3 – Plano organizacional

- Gerente de projeto: É responsável por coordenar as interações com os demais níveis abaixo, para conduzir o planejamento do projeto, mantendo a equipe focada para alcançar os objetivos do projeto.
- Cliente: Atribui os dados necessários para que o sistema seja construído. Informando ao analista de sistemas todos os requisitos que irão compor o sistema; valida as propostas ofertadas pelo analista e aprova a construção do *software*.
- Analista de sistemas: Possui um papel essencial de comunicação com o cliente para obter os requisitos necessários para a construção do sistema; responsável por executar as atividades previstas no cronograma, pelas modelagens do sistema, assim como descrever o que será realizado em cada etapa, bem como a interatividade entre usuário e sistema.
- Programador: Codifica o sistema de acordo com as especificações

feitas pelo analista, objetivando a coesão de um sistema funcional de acordo com o que foi especificado.

- Testador: Responsável por verificar a integridade do sistema a procura de eventuais erros, tentando se a mesma está coesa com o funcionamento do sistema. Utiliza-se de técnicas que ajudam a garantir uma detecção por falhas de forma mais efetiva, permitindo, assim, entregar para o cliente um produto superior de qualidade.

## **2.7 Plano de monitoramento e controle**

### *2.7.1 Introdução*

Segundo Pmbok ( 2004), as etapas decorridas do monitoramento e controle do projeto relacionam-se em monitorar os processos do projeto associados com suas fases. Ações preventivas e corretivas são tomadas, para que haja um controle no desempenho do projeto. Essa parte do projeto torna-se de suma importância por coletar, medir e gerar informações a respeito do desempenho do projeto e sobre os dados extraídos pelas medições, com o intuito de fazer melhorias em torno do processo.

O gerente de projetos possui um papel crucial para estar analisando as medições e tomando as devidas providências, de acordo com o que foi planejado, para que todo o processo siga em frente e obtenha um término como previsto. Deste modo, aumentando a probabilidade de ocorrências positivas e diminuindo os eventos negativos que estarão incidindo sobre o projeto.

O monitoramento e controle será concebido de acordo com o cronograma, item 2.8, seguindo todas as estimativas nele descrito. Caso haja alguma ocorrência que possa ser erronia referente a estimativa inicial, a mesma será controlada para que o cronograma não fuja do objetivo proposto do projeto.

Estão previstos três planos de monitoramento e controle, que são os seguintes:

Ordem	Ponto
1º Marco	Final da fase da especificação de requisitos
2º Marco	Final da fase de análise e modelagem
3º Marco	Final da fase de processo de teste
4º Marco	Final da fase da implantação pré-operação
5º Marco	Final do fechamento

Quadro 9 – Pontos de Monitoramento e Controle

### 2.7.2 Custo

O cliente, uma instituição filantrópica, terá um custo para o desenvolvimento que do projeto essencialmente reduzido no que se refere ao *software* utilizado, pois, em sua grande parte, o *software* livre estará à frente. Outro detalhe relevante é que os poucos *softwares* proprietários utilizados são licenciados pela instituição de ensino. Todas as etapas que compõem o ciclo de vida do projeto foram consideradas a fim de minimizar o custo de desenvolvimento: do levantamento dos requisitos à implantação do sistema.

A estimativa do custo é baseada nos pontos de função, de acordo com o levantamento de requisitos realizado junto ao cliente.

### 2.7.3 Prazo

A realização do monitoramento e controle do prazo de desenvolvimento do *software* foi definido a fim de serem utilizados diagramas e tabelas baseadas na estrutura analítica do projeto, item 2.4. O monitoramento será realizado no final de cada fase de implementação do projeto; ao final, pois, de cada ciclo de desenvolvimento. Caso ocorra atraso, em quaisquer das fases do desenvolvimento, realizar-se á uma análise para que seja identificado o motivo; e, conseqüentemente, possa ser realizado um reajuste no cronograma.

#### 2.7.4 Produção

"Por decisão do Colegiado do curso, a apresentação do **plano de monitoramento e controle de produção** é opcional, e por este motivo não será elaborado neste projeto."

#### 2.7.5 Risco

"Por decisão do Colegiado do curso, a apresentação do **plano de monitoramento e controle de risco** é opcional, e por este motivo não será elaborado neste projeto."

### 2.8 Cronograma

Para o cumprimento dos prazos e das atividades exercidas no projeto, é de suma importância que exista um cronograma para que possa controlar e cumprir os prazos que serão estabelecidos. É nessa parte que são estabelecidas as datas inicial e final do projeto, com base na estrutura analítica do projeto (Figura 2).

Para as informações de construção do projeto utilizou-se, como definido no escopo, recursos contendo duas pessoas na equipe a trabalhar pela disponibilidade de cada integrante em meio ao período letivo.

A definição de cada data, com seus respectivos prazos, foi definida no cronograma de atividades (Figura 4), com observações decorridas em projetos passados que utilizaram o ciclo de vida cascata.

De acordo com Pressman (2006), a criação de um cronograma para o projeto que será construído é uma atividade no qual o esforço das atividades são distribuídas pela duração planejada do projeto.












		Nome	Duração	Início	Término
1		<b>Levantamento de Requisitos</b>	<b>3 dias</b>	<b>12/02/11</b>	<b>16/02/11</b>
2		Reunião com cliente	1 dia	12/02/11	14/02/11
3		Levantamento preliminar de requisitos	1 dia	14/02/11	14/02/11
4		Especificação de requisitos	2 dias	15/02/11	16/02/11
5		<b>Planejamento do Projeto</b>	<b>19 dias</b>	<b>17/02/11</b>	<b>15/03/11</b>
6		Escopo	1 dia	17/02/11	17/02/11
7		Plano Processo de desenvolvimento	1 dia	18/02/11	18/02/11
8		Metodologia de desenvolvimento	1 dia	18/02/11	18/02/11
9		Estrutura analítica	1 dia	21/02/11	21/02/11
10		Estimativas	2 dias	22/02/11	23/02/11
11		Plano Organização	1 dia	24/02/11	24/02/11
12		Monitoramento e Controle	1 dia	25/02/11	25/02/11
13		Cronograma	3 dias	28/02/11	02/03/11
14		Recursos Humanos	1 dia	03/03/11	03/03/11
15		Recursos Gerais	1 dia	04/03/11	04/03/11
16		Plano de Custo	2 dias	07/03/11	08/03/11
17		Plano de Teste	1 dia	09/03/11	09/03/11
18		Plano de Treinamento	2 dias	10/03/11	11/03/11
19		Plano de Implantação	2 dias	14/03/11	15/03/11
20		<b>Análise e modelagem</b>	<b>30 dias</b>	<b>16/03/11</b>	<b>26/04/11</b>
21		Especificação de requisitos	3 dias	16/03/11	18/03/11
22		Modelagem do Banco de Dados	27 dias	21/03/11	26/04/11
23		<b>Implementação e testes</b>	<b>107 dias</b>	<b>27/04/11</b>	<b>22/09/11</b>
24		Desenvolvimento do sistema	107 dias	27/04/11	22/09/11
25		Testes	63 dias	16/06/11	13/09/11
26		<b>Implantação</b>	<b>14 dias</b>	<b>23/09/11</b>	<b>12/10/11</b>
27		Plano de implantação	11 dias	23/09/11	07/10/11
28		Plano de treinamento	3 dias	10/10/11	12/10/11

Figura 4 – Cronograma de atividades

### 2.8.1 Gráfico de Gantt

O gráfico de Gantt faz um resumo do monitoramento das atividades que serão exercidas durante o projeto, permitindo, assim, a construção do projeto, possibilitando a visualização gráfica do planejamento; é, portanto, a junção de todo esforço a envolver técnicas de custo, desenvolvimento da programação, aquisição de recursos e gerenciamento de risco.

A produtividade das tarefas exercidas são exibidas no gráfico de Gantt, que permite uma visualização rápida e clara de todo o projeto, que por sua vez pode ser visualizada por toda equipe que está a desenvolver o projeto.



## 2.9 Plano de recursos humanos

Conforme exemplificado no item 2.6, a necessidade dos profissionais específicos responsáveis por cada área será listado no quadro 10.

Plano de Recursos Humanos		
Função	Responsabilidade	Nome
Gerente do Projeto	Gerenciar todas etapas do projeto	Rafael Malatesta e Tales Araújo Mendonça
Analista de Sistemas	Fazer o levantamento de requisitos, modelagem e a projeção do sistema.	Rafael Malatesta e Tales Araújo Mendonça
Cliente	Fornecer informações de como o sistema deve funcionar.	Adriane Gonçalves Guedes
Programador	Codificar e implementar o sistema.	Rafael Malatesta e Tales Araújo Mendonça
Testador	Testar o sistema e recolher resultados.	Rafael Malatesta e Tales Araújo Mendonça

Quadro 10 – Funções e responsabilidades

## 2.10 Plano de recursos gerais

Para a elaboração do projeto será necessária a utilização de alguns recursos, como mostrado nos itens a seguir:

### 2.10.1 Hardware

- 1 Notebook Pentium Dual-Core com 4GB de memória RAM, 320GB de HD;
- 1 Microcomputador pessoal Intel Core2 Duo com 512GB de memória RAM, 200GB de HD;
- 1 Impressora Multifuncional;
- 1 Impressora a laser;

- 2 Roteadores;

### 2.10.2 Software

- Ubuntu 10.10 Desktop Edition - *Software Livre*;
- Microsoft Windows 7 Home Basic - Licenciado;
- OpenOffice 3.2 - *Software Livre*;
- Google Chrome 9.0 - *Software Livre*;
- Mozilla Firefox 3.6 - *Software Livre*;
- Kdesvn 1.5.4 - *Software Livre*;
- Tortoise SVN 1.6.12 - Gratuito;
- StarUml 5.0.2 - *Software Livre*;
- OpenProj 1.42 - *Software Livre*;
- Wbs Tool Web 0.9 beta - Gratuito;
- brModelo 2.0 - *Software Livre*;
- Vi IMproved 7.2 - *Software Livre*;
- Geany 0.19 - *Software Livre*;
- CakePHP 1.3.2 - *Software Livre*;
- PHP 5.3.3 - *Software Livre*;
- Mysql 5.0 - *Software Livre*;
- Dia 0.97.1 - *Software Livre*;
- Apache 2.2.16 - *Software Livre*;
- Windows Live Messenger - Gratuito;
- Empathy 2.32 - *Software Livre*;
- Gimp 2.6 – *Software Livre*;
- Inkscape 0.48 - *Software Livre*;
- MySQL Workbench v5.2.34 – *Software Livre*;

### 2.10.3 Recursos adicionais

- Energia elétrica;
- Telefone;
- Internet;
- Encadernações;
- Transporte;
- Alimentação;

## 2.11 Plano de custos

O plano de custos deste projeto conforme todo planejamento, será apresentado pelos quadros a seguir.

### 2.11.1 Custos de Software

Custos de Software			
ITEM	Custo (em R\$)	Licença	Total (em R\$)
Windows 7 Home Basic	16,50	1	16,50
<b>Total</b>			<b>16,50</b>

Quadro 11 – Custo de software

Visando uma maior flexibilidade, foi optado por utilizar o maior número de *softwares* gratuitos e/ou livres para minimizar o custo total deste projeto. Para os *softwares* adquiridos, espera-se poder utilizá-los em mais quatro projetos consecutivos, o que faz com que o seu valor seja dividido, além de estar calculado a vida útil do *software* que foi estimado em quatro anos representado no quadro 11.

O valor do *software* windows 7 home basic foi baseados em um orçamento *ON-LINE* visando um melhor custo/benefício.

### 2.11.2 Custos de Hardware

Custos de Hardware			
Descrição	Custo unitário	Quantidade	Valor total (em R\$)
Notebook Dual Core 4Gb RAM, 320Gb de HD	R\$ 1.500,00	1	$(1500 / 4) = 375$
Microcomputador pessoal Core 2 Duo, 512GB RAM, 200Gb HD	R\$ 700,00	1	$(700 / 4) = 175$
Impressora laser	R\$ 150,00	1	$(150 / 4) = 37,5$
Impressora Multifuncional	R\$ 200,00	1	$(200 / 4) = 50$
Roteador	R\$ 100,00	2	$(100 / 4) = 25$
<b>TOTAL</b>			<b><math>(662,5 / 5 \text{ projetos}) = 132,5</math></b>

Quadro 12 – Custo de Hardware

Com base nos equipamentos, calcula-se conseguir trabalhar com esse hardware em mais cinco projetos, devido ao desgaste dos mesmos e depreciação. Portanto, os valores apresentados no quadro 12, acima, mostram os valores já distribuídos entre os prováveis projetos e divididos pela sua vida útil, o qual será utilizado durante quatro anos.

Os valores encontrados foram baseados em orçamentos *ON-LINE* visando um melhor custo/benefício total dos equipamentos.

### 2.11.3 Custos de Mão de Obra

Custos de Mão de Obra					
Profissional	Salário(mês) em R\$	Salário (44h/Mês) em R\$	Meses trabalhados	Quantidade de profissionais	Total

Gerente de projeto	700	175	10	2	3.500
Analista	500	125	10	2	2.500
Testador	900	225	3	2	1.350
Programador	1000	250	5	2	2.500
<b>TOTAL</b>					<b>9.850,00</b>

Quadro 13 – Custo de Mão de Obra

Para o cálculo dos valores aproximados foi utilizada uma pesquisa de mercado feita no site (<http://www.ceviu.com.br>) a pegar os menores valores, isso devido a pouca experiência dos profissionais envolvidos no projeto. Neste cálculo ficou definido que os cargos de Gerencia de projeto e Analista estarão envolvidos em mais dois projetos além deste, o que faz com que o valor seja distribuído entre os projetos. Os valores estão referenciados no quadro 13.

Os cargos de Testador e Programador foram alocados excepcionalmente para este projeto conforme os valores do quadro 13.

A coluna Salário indica o valor referente a um mês de cada cargo neste projeto, mas como foi estimado no item 2.5.5 deste projeto que a jornada de trabalho seria de 3 horas por dia durante 3 dias na semana, a coluna Salário (44h/mês) informa os valores referentes a esse tempo trabalhado.

A coluna meses trabalhados indica o período trabalhado por cada cargo alocado no projeto.

Cada envolvido no projeto, que são dois integrantes, estarão a participar de todas as etapas. Cada integrante irá assumir o papel de gerente de projetos, analista, testador e programador, assumindo assim, todos os cargos contidos no custo de mão de obra, quadro 13, a ter dois integrantes para cada função estabelecida.

#### 2.11.4 Custos com outras despesas



<b>Custos com outras despesas</b>			
	<b>Custo (mensal)</b>	<b>Quantidade (meses)</b>	<b>Total (R\$)</b>
Energia	R\$ 100,00	10	1.000,00
Telefone	R\$ 6,60	10	66,00
Internet	R\$ 43,30	10	443,00
Encadernação	R\$ 4,50 (cada)	10 (vezes)	45,00
Transporte	R\$ 31,00	10	310,00
Alimentação	R\$ 180,00	10	1.800,00
<b>Total</b>			<b>R\$ 3.664,00</b>

Quadro 14 – Custo com outras despesas

Levando em consideração o tempo total gasto no projeto conforme o quadro 8 do item 2.5.5, a tabela especifica os valores dos custos mensais assim como a sua totalidade.

Os itens energia, telefone e internet foram estimados com base nos equipamentos eletro eletrônicos usados neste projeto, além de mais dois consecutivos, fazendo com que o valor mensal seja dividido entre eles. Os demais itens foram estimados para apenas este projeto.

#### 2.11.5 Total Geral das Despesas do projeto

<b>Recurso</b>	<b>Valores (em R\$)</b>
Software	16,50
Hardware	132,5
Mão de Obra	9.850,00
Despesas Gerais	3.664,00
<b>TOTAL</b>	<b>R\$ 13.663,00</b>

Quadro 15 – Custo total

O quadro 15 resume o total gasto com este projeto desde o seu início até a implantação do mesmo, valendo a ressalva que todos os valores são estimados

podendo sofrer alterações a medida que o projeto avança.

## **2.12 Plano de gerencia de dados**

"Por decisão do Colegiado do curso, a apresentação do **plano de gerencia de dados** é opcional, e por este motivo não será elaborado neste projeto."

## **2.13 Plano de medição e análise**

"Devido o grau de maturidade do processo de desenvolvimento utilizado, o **Plano de medição e análise** não será elaborado neste projeto."

## **2.14 Plano de gerencia de configuração**

"Por decisão do Colegiado do curso, a apresentação do **plano de gerencia de configuração** é opcional, e por este motivo não será elaborado neste projeto."

## **2.15 Plano de gerencia de riscos**

"Por decisão do Colegiado do curso, a apresentação do **plano de gerencia de risco** é opcional, e por este motivo não será elaborado neste projeto."

## **2.16 Plano de garantia da qualidade**

"Por decisão do Colegiado do curso, a apresentação do **plano de garantia da qualidade** é opcional, e por este motivo não será elaborado neste projeto."

### 2.17 Plano de verificação

"Devido o grau de maturidade do processo de desenvolvimento utilizado, o **Plano de verificação** não será elaborado neste projeto."

### 2.18 Plano de validação

"Devido o grau de maturidade do processo de desenvolvimento utilizado, o **Plano de validação** não será elaborado neste projeto."

### 2.19 Plano de testes

O processo de execução de um produto para averiguar se ele atingiu as especificações e funcionou corretamente para o ambiente o qual foi projetado, é dito teste de *software*. Segundo Pressman (2006), um bom teste é o que possui grande probabilidade de encontrar falhas. O objetivo de testar o *software* é tentar minimizar ao máximo o aparecimento de defeitos, para que assim não surjam erros nem falhas no *software*. Os erros podem ocorrer por diversos fatores, como a inexperiência do desenvolvedor ou do analista a analisar os requisitos e os modelos.

Os testes devem ser analisados e planejados com antecedência, levando em consideração os requisitos do cliente, a fim de encontrar falhas e corrigi-las, na fase de codificação e testes de desenvolvimento do *software*.

Este projeto adotará a técnica caixa-preta, que consiste em um componente de *software* ser testado como se fosse uma caixa-preta, ou seja, não a considerar o comportamento interno do mesmo. Dados de entrada são fornecidos, o teste é executado e o resultado obtido é comparado a um resultado esperado previamente conhecido. Haverá sucesso no teste se o resultado obtido for igual ao resultado esperado. O componente de *software* a ser testado pode ser um método, uma função interna, um programa, um conjunto de programas e/ou componentes ou mesmo uma funcionalidade. A técnica de teste funcional é aplicável a todos os níveis de teste (PRESSMAN, 2006).

## **2.20 Plano de treinamento**

O objetivo de realizar um plano de treinamento é poder preparar o utilizador do sistema, usuário, para que possa desfrutar de todas suas funcionalidades de modo a agregar as tarefas rotineiras ao sistema, podendo assim atender os objetivos propostos.

Todo o treinamento será realizado, na instituição, de maneira formal, através de palestras que serão agendadas junto aos responsáveis da SEJA. Este treinamento objetiva-se a escolher pessoas chaves para que fique responsável pela administração do sistema e que assim possa estar a realizar o controle dos usuários no sistema; também será realizado um treinamento voltado para cada responsável do DAS, ficando estes incumbidos de repassarem as informações para os novatos que irão ocupar tais cargos e aos demais substitutos que precisarem utilizar o sistema.

Será realizado um feedback com o cliente para esclarecer possíveis dúvidas sobre a utilização do sistema no decorrer da implantação.

Todo treinamento será realizado de acordo com o item 2.8 estabelecido no cronograma.

## **2.21 Plano de implantação**

### *2.21.1 Introdução*

Para o plano de implantação do sistema, é necessário um conjunto de atividades ou tarefas a serem seguidas, a fim de colocar em funcionamento o produto desenvolvido, de forma que o mesmo esteja pronto para ser utilizado pelo cliente. Como atualmente a instituição advém de um sistema informatizado, não haverá necessidade de estar realizando algum tipo de migração, apenas a inserção dos dados atualmente utilizados.

Apesar do sistema desenvolvido ser voltado para web, serão configuradas duas estações para que o sistema possa ser acessado localmente; o sistema também estará disponível para ser acessado através da internet, necessitando apenas de um login e senha do usuário que esteja cadastrado. Devido à amplitude do porte da instituição ser pequeno, não haverá necessidade da utilização de uma máquina potente para atender a todos os utilizadores.

Para a implantação do sistema serão utilizados os seguintes recursos de hardware e *software*, seguindo do guia de instalação dos mesmos.

### *2.21.2 Especificação do hardware e do software*

#### *2.21.2.1 Hardware*

- Servidor
  - Processador com mínimo de 1GHz;
  - Memória RAM com mínimo de 384MB;
  - Disco rígido com mínimo de 2,5GB;
  - Placa de rede 10/100;
  - Monitor com resolução mínima de 800x600;
- Estações
  - Processador com mínimo de 600MHz;
  - Memória com mínimo de 128MB;
  - Disco rígido com mínimo de 2,5GB;
  - Placa de rede 10/100;
  - Monitor com resolução mínima de 800x600;
- Outras especificações
  - Cabo de rede;
  - Roteador

### 2.21.2.2 Software

- Servidor
  - Sistema Operacional GNU/Linux;
  - Servidor Apache2;
  - PHP5;
  - MySQL 5;
  - Crontab;
  - Mailx;
  - Ssmtp;
  - Mpack;
  - Bunzip2;
  - Mozilla Firefox 3.x ou Chrome/Chromium 9.x, ou superior;
- Estação
  - Sistema Operacional GNU/Linux;
  - Mozilla Firefox 3.x ou Chrome/Chromium 9.x, ou superior;

### 2.21.3 *Guia da instalação*

- Servidor
  - Passo 1: Instalar a distribuição GNU/Linux definida;
  - Passo 2: Configurar a rede no servidor com IP fixo;
  - Passo 3: Instalar os programas na seguinte ordem: apache, php5, mysql, crontab, mailx, ssmtp, mpack, bunzip2, firefox ou chrome/chromium;
  - Passo 4: Configurar o servidor apache;
  - Passo 5: Criar o diretório /var/www/sgce e copiar todo o sistema para tal lugar;
  - Passo 6: Executar o browser que foi instalado e acessar o endereço <http://localhost/sgce> e fazer a instalação do sistema de acordo com

o guia que será exibido.

- Estação
  - Passo 1: Instalar a distribuição GNU/LINUX definida;
  - Passo 2: Configurar a rede como DHCP;
  - Passo 3: Instalar o navegador firefox ou chrome/chromium.
- Outras especificações
  - Passo 1: Crimpar e instalar todos os cabos de rede;
  - Passo 2: Configurar o roteador.

## 2.22 Observações complementares

Em detrimento em optar pela utilização de 99% dos *softwares* neste projeto serem classificados como livres, *software* livre, ou gratuitos, foi possível obter um grande abatimento nos valores das licenças de *softwares* que poderiam ser adquiridos para, assim, poder distribuir a renda referente aos mesmos para outras atividades do projeto.

O objetivo não está referente apenas na redução de valores, mas em mostrar que é possível construir um bom sistema utilizando ferramentas não proprietárias, mostrar que apesar das ferramentas livres não possuem um custo para o utilizador, as mesmas são, em grande parte, desenvolvidas por grandes empresas sérias que visam a qualidade de seus produtos.

## 3 ESPECIFICAÇÃO DE REQUISITOS

De acordo com o item 1.5 do capítulo 1, foi realizado um levantamento preliminar dos requisitos, para que esses pudessem ser utilizados para calcular o tempo de desenvolvimento do projeto, bem como os custos que serão gastos. No presente momento dos levantamentos preliminares não havia uma maturidade para um levantamento de requisitos completos, o que se dará neste capítulo, onde todos os requisitos possíveis serão descritos e esmiuçados para uma melhor compreensão

e clareza do projeto que irá se formar.

Data	Versão	Descrição	Autor
19/03/2011	1.0	Levantamento de requisitos	Rafael Malatesta e Tales Araújo
21/03/2011	1.1	Especificação de caso de uso	Rafael Malatesta e Tales Araújo
20/04/2011	1.2	Término da especificação do caso de uso.	Rafael Malatesta e Tales Araújo

Quadro 16 – Histórico do Controle de Versão

A especificação de requisitos é uma parte primordial do sistema, creio que uma das mais importantes, se não a mais importante, pois é baseado nessa etapa que o *software* será construído e sem ela fica impossível construir qualquer tipo de *software*. Essa é a parte que o cliente dialoga tudo que precisa e o analista esmiúça as informações coletadas para se adentrarem no *software* que irá se formar.

### 3.1 Requisitos do cliente

#### 3.1.1 Descrição geral da aplicação

##### 3.1.1.1 Descrição da necessidade

A SEJA, atualmente, não possui um controle informatizado do DAS - Departamento de Assistência Social - que é controlado através de uma planilha impressa, onde fica a cuidado dos voluntários responsáveis manter os dados atualizados e íntegros, tanto das famílias assistidas quanto dos voluntários cadastrados e do controle de alimentos do estoque.

Visando um melhoramento, há necessidade de disponibilizar para o cliente um sistema web que irá trabalhar simultâneo com um número de até dez usuários, podendo esse número ser ampliado de acordo com as necessidades da instituição.



O sistema deverá atender as seguintes necessidades do cliente: realizar e manter o cadastro dos voluntários, bem como das famílias assistidas e dos alimentos que irão compor as cestas básicas; permitir um gerenciamento do estoque, controlando o prazo de validade dos alimentos e avisando sobre os alimentos que estarão com um prazo de 15 dias para vencer; emitir relatórios sobre as famílias que estão com vencimento de matrícula, quantidade de cestas básicas prontas, item pendente em cada cesta, todos os alimentos que irão vencer dentro de um mês, voluntários disponíveis.

Os benefícios esperados com essa solução de *software* está em manter, primeiramente, um controle de tudo que acontece no setor do DAS, melhorando o atendimento as famílias assistidas, oferecendo mais qualidade, tanto em atendimento quanto nos produtos doados; agilizar todo esse gerenciamento de maneira a reduzir o tempo que é gasto atualmente, podendo ser aproveitado em outras atividades existentes na instituição.

#### 3.1.1.2 Objetivo

O Objetivo do *software* SGCE é possuir uma base de dados de voluntários, famílias assistidas e alimentos, para que possam ser controlados e monitorados. Sendo os voluntários remanejados a medida que haja necessidade; a controlar as famílias assistidas para que estejam aptas a receber as doações de cestas básicas; e manter um controle do estoque de alimentos, avisando com antecedência sobre o vencimento bem como a falta de algum item que compõe a cesta, antes que ela possa ser montada e distribuída para as famílias carentes.

#### 3.1.1.3 Escopo da Aplicação

As necessidades sobre o que estará disponível para o cliente for definido no escopo da aplicação, a objetivar uma solução que transmita facilidade de gerenciamento de todo o setor do DAS, facilitando o controle das famílias assistidas

que fazem parte da SEJA, bem como controlar o estoque de alimentos. Informações complementares sobre o escopo pode ser conferida no item 2.1.

### *3.1.2 Descrição geral do cliente*

Conforme descrito anteriormente a SEJA localizada no bairro Bairu em Juiz de Fora e tem como principal fornecedora de dados a Adriane Gonçalves Guedes além dos demais colaboradores das demais áreas da instituição.

### *3.1.3 Lista de requisitos do cliente*

As necessidades que o cliente apresenta são ditos requisitos funcionais. discernidas em uma linguagem de fácil absorção, tem como objetivo expressar aquilo que o sistema deverá realizar, a proporcionar uma solução ideal para o cliente.

Abaixo serão descritos os requisitos do cliente em conformidade com o projeto proposto:

- RC01: Os usuários do sistema deverão ser autenticados;
- RC02: O sistema deverá permitir que o administrador do sistema modere outros usuários com um controle de permissão;
- RC03: O sistema deverá realizar o *backup* do banco de dados de forma automatizada, podendo esse ser enviado para um *e-mail* cadastrado;
- RC04: O sistema deverá possuir um histórico das transições realizadas;
- RC05: O sistema deverá manter, de forma íntegra, todos os dados inseridos na base de dados;

## 3.2 Requisitos do software

### 3.2.1 - Fronteiras do software

As fronteiras do *software* descritas abaixo descrevem as funcionalidades que não entrarão no escopo deste projeto a ser desenvolvido. Essas são funcionalidades que não apreciam o sistema em si.

- O SGCE não usará interfaces de programas de terceiros e nem disponibilizará que terceiros utilize sua interface;
- O SGCE não compartilhará o Banco de dados com outros sistemas;
- Os itens listados no item 3.2.2 deste projeto descrevem os limites da fronteira do SGCE.

### 3.2.2 Itens de software

O SGCE possuirá os seguintes itens de *software*:

- Inclusão (família, mantimento, usuário, frequência);
- Exclusão (família, mantimento, usuário);
- Alteração (família, mantimento, usuário, frequência);
- Consulta (família, mantimento, usuário, frequência);
- Controle do estoque de mantimentos(criação de cestas básicas);
- Controle das famílias assistidas (ativação/inativação);
- Monitoramento da validade dos alimentos.

### 3.2.3 Lista de requisitos não funcionais

Requisitos não funcionais estão relacionados ao uso da aplicação em termos de usabilidade, segurança, interface, acessibilidade, robustez e tecnologias

envolvidas. Geralmente os requisitos não funcionais são características mínimas de um *software* de qualidade, deixando que o desenvolvedor implemente ou não esses requisitos no *software* que será desenvolvido.

#### Usabilidade

- RNF01: O sistema deverá ser de fácil manuseio e intuitivo para que pessoas com pouca experiência consigam mexer no sistema, evitando futuros treinamentos específicos para possíveis novos usuários;

#### Segurança

- RNF02: Somente terá acesso ao sistema e suas funcionalidades as pessoas devidamente cadastradas no sistema e autenticadas por meio de usuário e senha
- RNF03: Somente usuários do tipo administrador poderão realizar exclusões no sistema;
- RNF04: Usuários Voluntários poderão realizar as operações de inclusão, alteração e consulta somente na sua área de atuação;

#### Interface

- RNF05: O SGCE será um sistema voltado para a internet e terá uma quantidade reduzida de imagens ou algo que possa retardar o uso do mesmo, gerando uma página bem leve e de rápido carregamento;
- RNF06: Os botões e ações possíveis sempre estarão em um local específico facilitando para o usuário no uso do *software*;
- RNF07: Os campos para preenchimento deverão conter máscara quando necessário além de informar com uma dica em cada campo de preenchimento de texto ou números;
- RNF08: Cada usuário terá o tempo de 1 min de ociosidade para que o sistema faça *logoff* de seu usuário.
- RNF09: Todas as interações do sistema com o usuário deverão ser padronizadas. As mensagens exibidas, assim como elementos de tela (botões, tabelas e menus), seguirão um padrão, tornando ainda mais

intuitivo o uso do sistema.

#### Acessibilidade

- RNF10: O SGCE será implementado com fontes de tamanho adequado e cores que não cansem a visão do usuário permitindo um uso rápido e prático;

#### Robustez

- RNF11: O sistema deverá prever o uso simultâneo de usuários além de tratar a concorrência de acesso em partes específicas do mesmo com mensagens na tela;

#### Tecnológico

- RNF12: O sistema será feito voltado para a Web. A linguagem de desenvolvimento utilizada será PHP, o sistema de banco de dados será MySQL.
- RNF13: O sistema será feito para rodar no sistema Linux local e posteriormente com a instalação da internet poderá ser acessado via web;
- O *software* deve ser compatível com os browsers Chrome e Mozilla Firefox;

### 3.2.4 Requisitos funcionais

Os requisitos funcionais descrevem o funcionamento do sistema, suas ações para cada entrada. É tudo aquilo que descreve o que deve ser realizado pelo sistema. É a parte principal do sistema, já que descrevem as funcionalidades que o sistema deve dispor.

### 3.2.4.1 Lista dos requisitos funcionais

#### 3.2.4.1.1 Inclusões

- RF01: Família;
- RF02: Estoque
- RF03: Definição Cesta;
- RF04: Usuário;
- RF05: Frequência da Família;
- RF06: Voluntário;

#### 3.2.4.1.2 Exclusão

- RF07: Família;
- RF08: Estoque;
- RF09: Definição Cesta;
- RF10: Usuário;
- RF11: Voluntário;

#### 3.2.4.1.3 Alterações

- RF12: Família;
- RF13: Estoque;
- RF14: Definição Cesta;
- RF15: Usuário;
- RF16: Frequência da família;
- RF17: Voluntário;

#### 3.2.4.1.4 Consultas

- RF18: Estoque;

- RF19: Definição Cesta;
- RF20: Usuário;
- RF21: Voluntário
- RF22: Família;
- RF23: Frequência Família;
- RF24: *Log* do sistema;

#### 3.2.4.1.5 Relatórios

- RF25: Relatório de vencimento anual matricula das famílias;
- RF26: Relatórios das famílias aptas a receber a cesta básica;
- RF27: Itens pendentes para completar cestas básicas;
- RF28: Relatório da quantidade de cestas disponíveis atuais;
- RF29: Relatório de vencimento dos itens do estoque;
- RF30: Relatório dos voluntários da instituição;

#### 3.2.4.1.7 Outras funcionalidades

- RF31: O *software* deve permitir que o administrador inclua, altere ou exclua perfis de acesso;
- RF32: O *software* deve identificar todos os usuários que desejam acessá-lo, identificando seu perfil;
- RF33: Recuperar senha.
- RF34: O sistema deve permitir ao administrador gerir usuários.
- RF35: O sistema deve permitir ao administrador gerir categoria dos usuários (consultor, coordenador, estoquista).
- RF36: O sistema deve permitir ao administrador alterar permissões de categorias.
- RF37: O sistema deve permitir ao administrador associar usuário às categorias. Um usuário pode ser associado a uma categoria e uma categoria pode se associar com vários usuários.
- RF38: Deverá existir um controle das datas de validade dos alimentos

em estoque;

- RF39: O sistema deverá controlar uma lista de presença das famílias aptas a receber a cesta;
- RF40: Nas entregas de cestas básicas, o usuário informará ao sistema o nome do representante da família e o sistema dará baixa no estoque dos mantimentos contidos na cesta.

### 3.2.4.2 Descrição dos atores

Nome	Descrição
Administrador	É o único responsável por realizar a manutenção dos Usuários além de poder realizar as demais funcionalidades do sistema abrangente aos outros Atores.
Coordenador	É um usuário que poderá realizar as funcionalidades de manutenção das famílias assistidas, das frequências das famílias, além de poder realizar as <u>consultas</u> previstas no item 3.2.4.1.4 deste projeto, além dos <u>relatórios</u> previstos no item 3.2.4.1.6. Não irá realizar as funções específicas do administrador.
Estoquista	É um usuário que somente poderá realizar a manutenção do mantimento, realizar as <u>consultas</u> previstas no item 3.2.4.1.4 deste projeto, além dos <u>relatórios</u> previstos no item 3.2.4.1.6.
Consultor	São usuários restritos do sistema e só poderão realizar as <u>consultas</u> referentes ao item 3.2.4.1.4 deste projeto e os <u>relatórios</u> referentes ao item 3.2.4.1.6.

Quadro 17 – Descrição dos Atores



### 3.2.4.2.1 Esquema dos atores

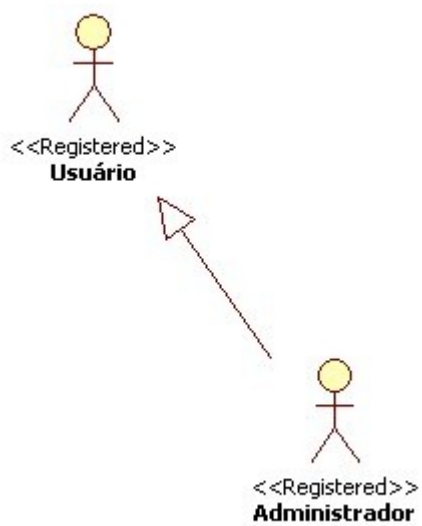


Figura 6 – Mapa de atores



Figura 7 - Diagrama Usuário

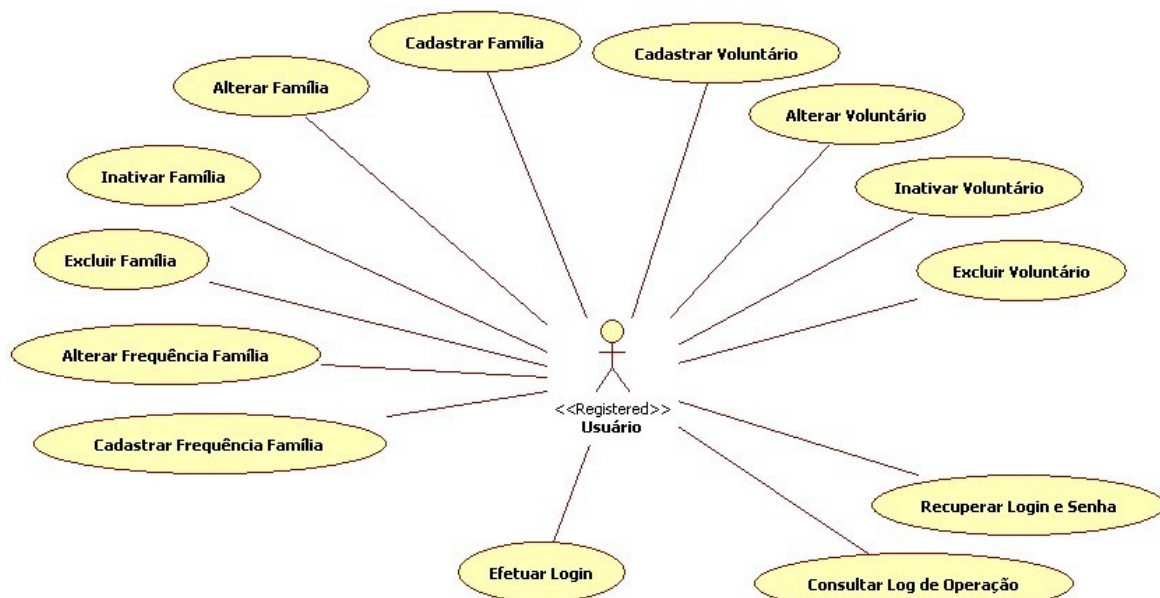


Figura 8 – Diagrama Usuário (Perfil Coordenador)

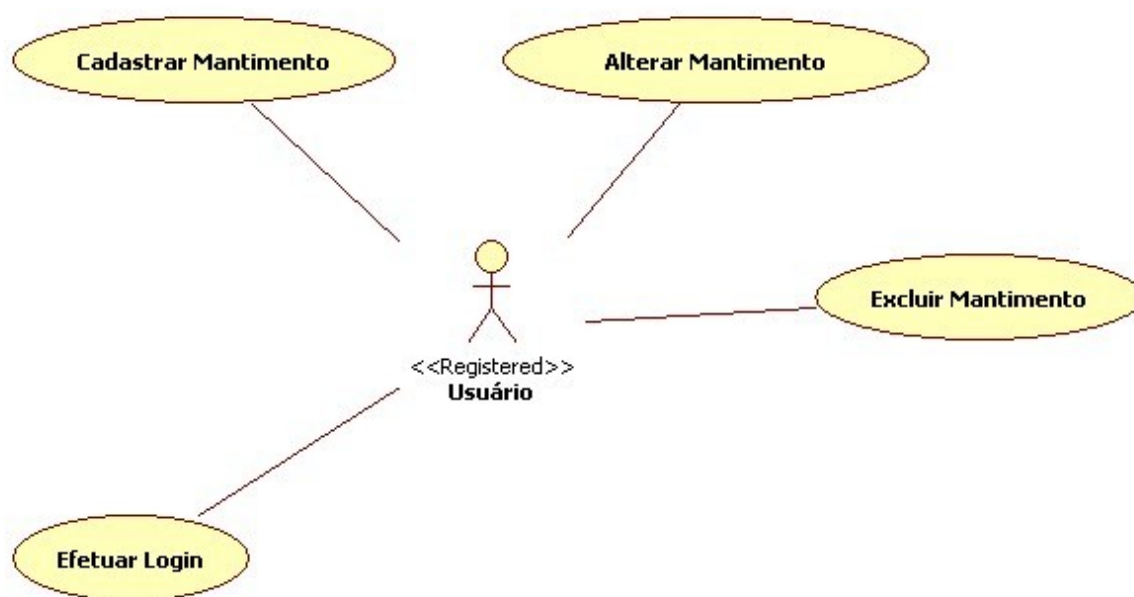


Figura 9 – Diagrama Usuário (Perfil Estoquista)

Nome	Descrição
Administrador	Responsável por administrar os usuários.
Consultor	Permite apenas realizar consultas no sistema.
Estoquista	Responsável por gerenciar o estoque, bem como realizar consultas no sistema.
Coordenador	Responsável por coordenar, administrar, as famílias, bem como realizar consultas no sistema.

Quadro 18 – Perfis dos Usuários do Sistema

### 3.2.4.3 Descrição de casos de uso

#### 3.2.4.3.1 Modelos de casos de uso

Casos de uso são representações gráficas das ligações dos atores com as funcionalidades do sistema.

Os diagramas abaixo descritos foram criados no StarUML conforme descrito no item 2.10.2 deste projeto.

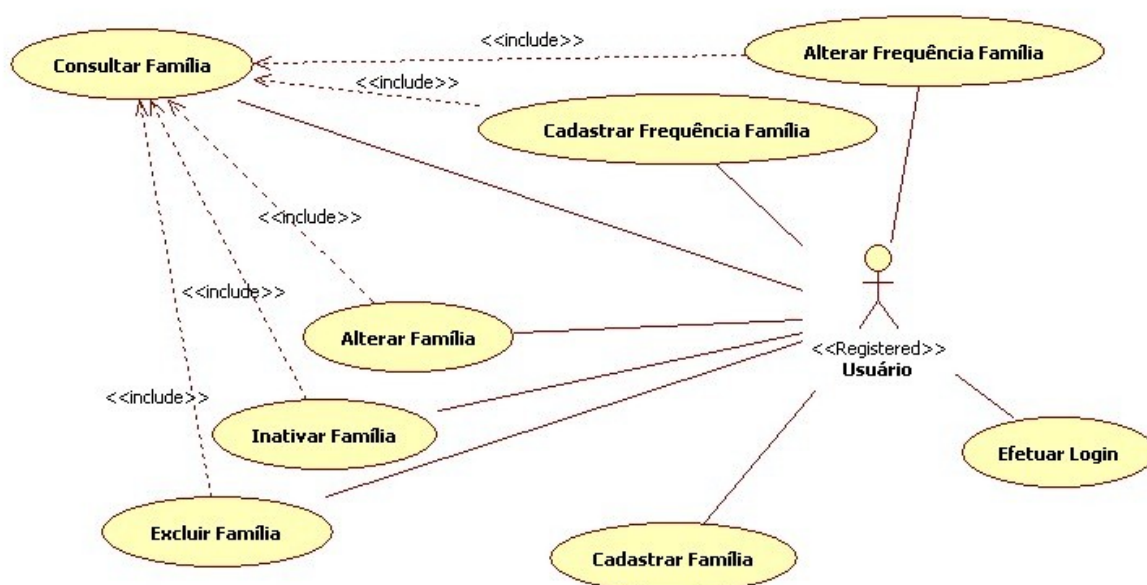


Figura 10 – Caso de uso família

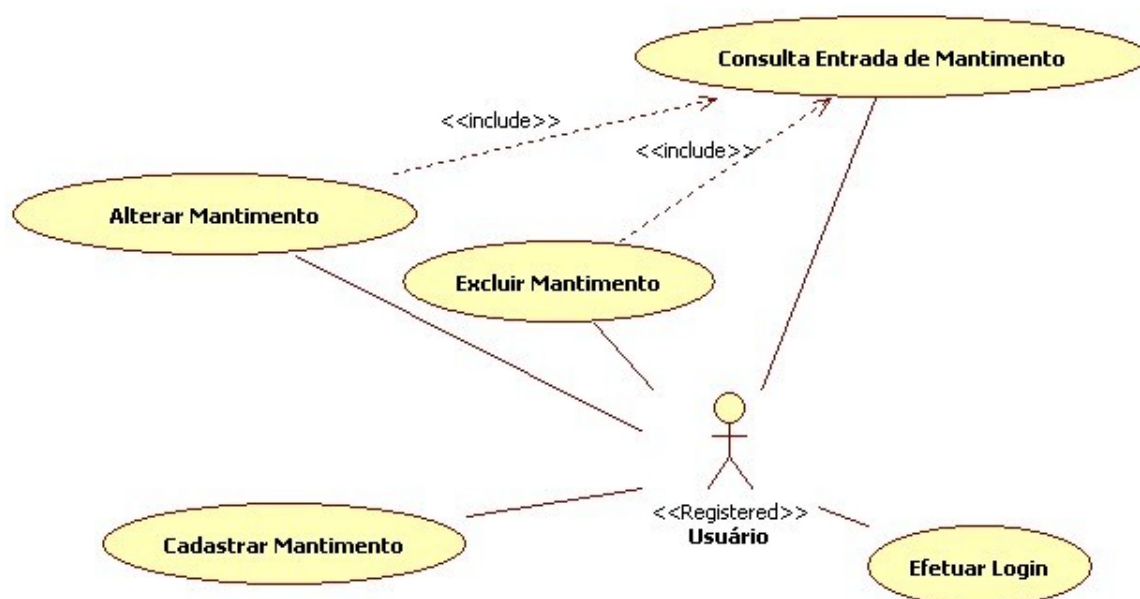


Figura 11 – Caso de uso mantimento

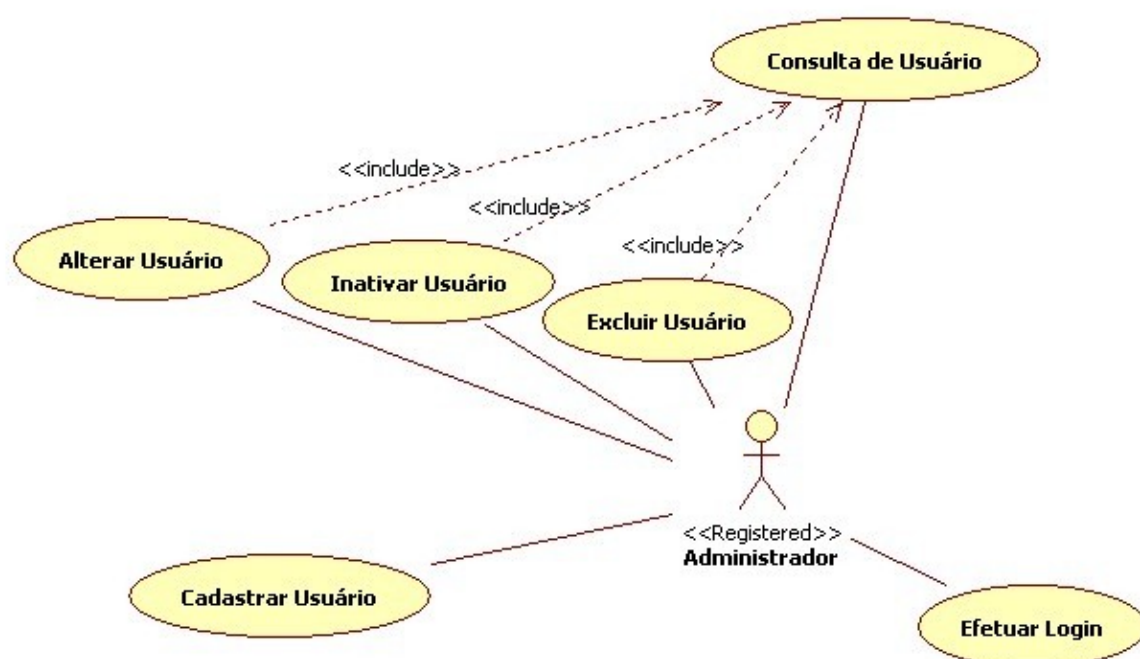


Figura 12 – Caso de uso Usuário

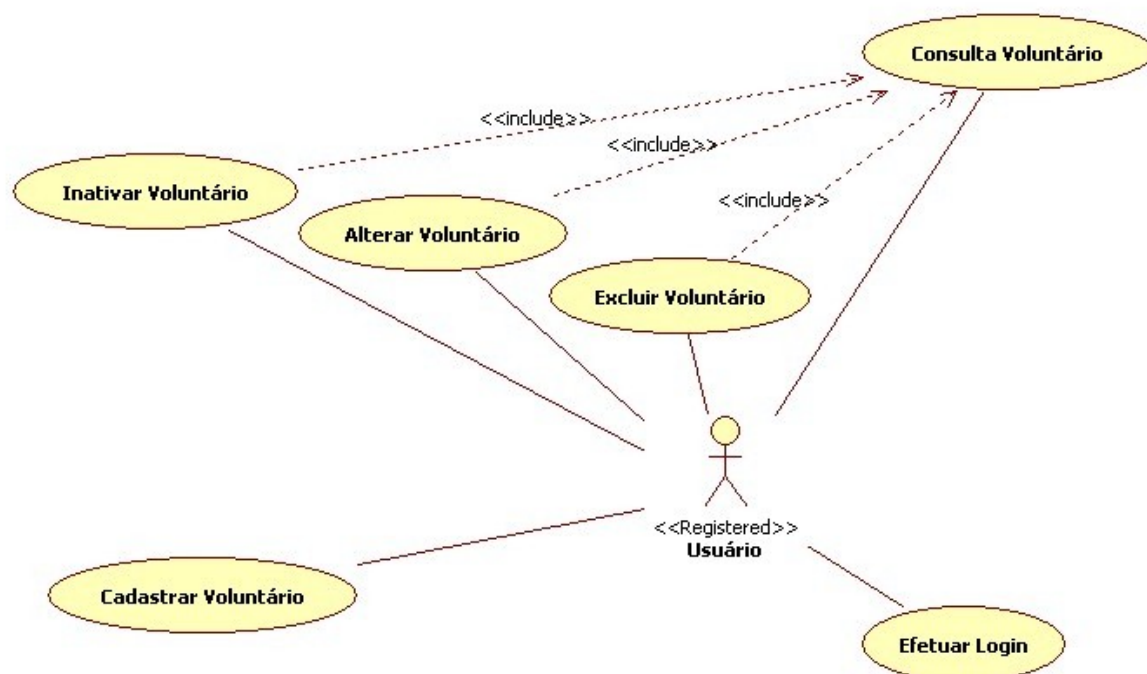


Figura 13 – Caso de uso Voluntário

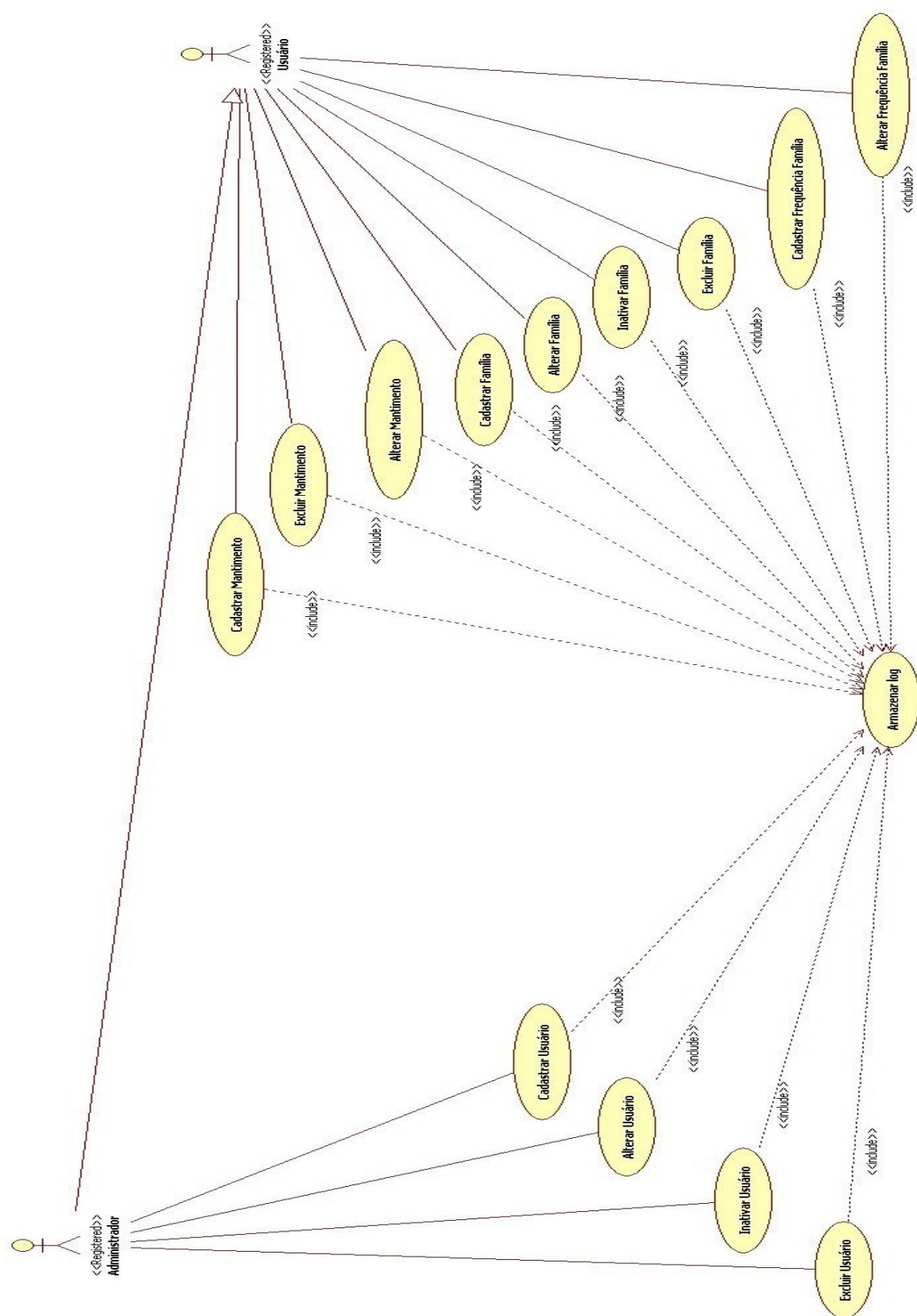


Figura 14 – Caso de uso Log

### 3.2.4.3.2 Especificação de caso de uso

Especificar um caso de uso é detalhar como deverá ocorrer uma interseção entre o sistema e os atores. A especificação de caso de uso pretende orientar o desenvolvimento do *software* a validar os requisitos antes de começar sua parte de programação do *software*.

As especificações demonstram cada ação referente ao ator e sua resposta esperada, a apresentar informações abstraídas a interface gráfica, não abordando aspectos ligados a arquitetura ou codificação, apenas a demonstrar o fluxo de funcionamento de cada caso de uso.

Serão demonstrados a seguir os casos de uso referente ao sistema. Para identificá-los, será utilizado a sigla UC, do inglês *Use Case*, seguido de um número que enumera a sequência, que por sua vez será seguido do nome do caso de uso.

#### 3.2.4.3.2.1 Caso de uso UC01 - Cadastrar Usuário

##### **Sumário**

Inicia-se quando um usuário solicita a interface de cadastro de usuário. O administrador do sistema será o responsável por adicionar os dados e definir as permissões de cada usuário.

##### **Ator**

Administrador

##### **Pré-condições**

Administrador cadastrado e logado no sistema com permissão para realizar a operação de alteração de usuário.

##### **Curso Normal**

1. Administrador solicita ao sistema a interface de cadastro de usuário;
2. Sistema autentica o usuário;



3. Sistema exibe a interface solicitada;
4. Administrador insere os dados (RD05);
5. Sistema verifica se os campos obrigatórios: status, perfil, prestação serviço, nome, *e-mail*, *senha* e CPF foram digitados.
6. Sistema valida *e-mail*;
7. Sistema valida senha;
8. Administrador confirma operação de inclusão de usuário;
9. Sistema autentica usuário;
10. Sistema verifica se não existe *e-mail* já cadastrado para outro usuário ativo;
11. Sistema verifica se não existe *e-mail* já cadastrado para outro usuário inativo;
12. Sistema gera um código único para o usuário cadastrado;
13. Sistema salva as informações no Banco de Dados;
14. Caso de uso UC28 - Armazenar *Log*;
15. É exibida uma mensagem de usuário cadastrado. Caso de uso termina.

### **Curso Alternativo**

- 1 a 8: O administrador pode cancelar a operação. Caso de uso termina.
- 11.1: *E-mail* já está cadastrado no sistema para outro usuário inativo;
- 11.2: Sistema solicita confirmação de ativação de usuário;
- 11.3: Administrador confirma a solicitação;
- 11.4: Sistema autentica administrador;
- 11.2 a 11.3: A operação pode ser cancelada. Caso de uso termina.

### **Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 4.1: O tipo de usuário não foi definido, é assumido o nível mais baixo como padrão, consultor;
- 5.1: Os campos obrigatórios não foram preenchidos. Sistema informa ao usuário;
- 6.1: O *e-mail* não está no padrão correto. Sistema informa ao usuário;
- 9.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 10.1: *E-mail* já está cadastrado no sistema para outro usuário ativo. Sistema informa ao usuário. Caso de uso termina.

11.4.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

### **Pós-Condições**

Usuário cadastrado no banco de dados do sistema.

#### 3.2.4.3.2.2 Caso de uso UC02 - Alterar Usuário

### **Sumário**

Inicia-se quando o Administrador solicita a interface de consulta de usuário por motivo de erro ou outro afim de manter os dados atualizados no sistema.

### **Ator**

Administrador

### **Pré-condições**

Administrador cadastrado e logado no sistema com permissão para realizar a operação de alteração de usuário;

Usuário cadastrado no sistema.

### **Curso Normal**

1. Caso de uso UC04 - Consultar usuário;
2. Administrador seleciona usuário e escolhe a opção "editar";
3. Administrador altera os dados desejados;
4. Sistema verifica se os campos obrigatórios (status, perfil, prestação serviço, nome, *e-mail*, *senha* e CPF) foram preenchidos;
5. Sistema valida *e-mail*;
6. Administrador confirma operação de alteração de usuário;
7. Sistema autentica Administrador;
8. Sistema verifica se não existe nome e *e-mail* já existentes para outro usuário ativo;
9. Sistema verifica se não existe nome e *e-mail* já existentes para outro usuário

inativo;

10. Sistema salva as informações no banco de dados;

11. Caso de uso UC28 - Armazenar *log* no sistema;

12. É exibida uma mensagem de usuário salvo. Caso de uso termina.

### **Curso Alternativo**

1 a 6: A operação pode ser cancelada. Caso de uso termina;

9.1: Nome e *e-mail* já existentes no sistema para usuário inativo;

9.2: Sistema solicita confirmação de ativação do usuário;

9.3: Administrador confirma operação;

9.4: Sistema autentica administrador;

9.1 e 9.3: A operação pode ser cancelada. Caso de uso termina.

### **Curso de Exceção**

4.1: Os campos obrigatórios não foram preenchidos. Sistema informa ao Administrador;

5.1: O *e-mail* não está no formato correto (RN01). Sistema informa ao usuário;

6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

8.1: Nome ou *e-mail* já existente no sistema para outro usuário ativo. Sistema exibe mensagem ao usuário. Caso de uso termina;

9.4.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

### **Pós-Condições**

Usuário alterado no banco de dados do sistema.

#### **3.2.4.3.2.3 Caso de uso UC03 - Excluir Usuário**

### **Sumário**

Inicia-se quando o administrador solicita ao sistema a interface de consulta de usuário afim de não mais permitir que o usuário acesse o sistema por completo.

**Ator**

Administrador

**Pré-condições**

Administrador cadastrado no sistema;

Usuário cadastrado no sistema.

**Curso Normal**

1. Caso de uso UC04 - Consultar Usuário;
2. Administrador seleciona usuário desejado;
3. Administrador escolhe operação de exclusão;
4. Sistema informa ao usuário mensagem de confirmação de exclusão;
5. Administrador confirma operação de exclusão;
6. Sistema autentica Administrador;
7. Sistema exclui informações do banco de dados do sistema;
8. Caso de uso UC28 - Armazenar *log* no sistema;
9. Sistema exibe mensagem de usuário excluído. Caso de uso termina.

**Curso Alternativo**

1 ao 5: Operação pode ser cancelada pelo administrador. Caso de uso termina.

**Curso de Exceção**

6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Usuário excluído do banco de dados do sistema.

3.2.4.3.2.4 Caso de uso UC04 - Consulta Usuário

**Sumário**

Inicia-se quando o administrador solicita a interface de consulta de usuário. É

utilizada para ver a listagem dos usuários cadastrados além de saber quais estão ativos e inativos.

**Ator**

Administrador

**Pré-condições**

Administrador cadastrado e logado no sistema;

Usuário cadastrado no sistema.

**Curso Normal**

1. Administrador solicita ao sistema interface de consulta de usuário;
2. Sistema apresenta na tela uma listagem disponível no banco de dados. Caso de uso termina.

**Curso Alternativo**

- 2: Operação pode ser cancelada pelo administrador. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sistema não encontra resultados a partir do filtro escolhido. Sistema exibe mensagem na tela. Caso de uso termina.

**Pós-Condições**

Usuário consultado poderá ser visualizado, alterado ou excluído do banco de dados do sistema.

### 3.2.4.3.2.5 Caso de uso UC05 - Incluir Estoque

**Sumário**

Inicia-se quando o usuário solicita a interface de inclusão de mantimento. O usuário com permissão para realizar o comando poderá incluir qualquer tipo de mantimento

no sistema.

### **Ator**

Usuário

### **Pré-condições**

Usuário cadastrado e logado com permissão para realizar a operação.

Cesta base pré-cadastrada no sistema.

### **Curso Normal**

1. Usuário solicita ao sistema a interface de cadastro de estoque;
2. Sistema autentica usuário;
3. Sistema exibe a interface solicitada;
4. Usuário insere dados (RD04) para incluir um novo mantimento;
5. Sistema verifica se os campos obrigatórios (mantimento, medida, complemento, quantidade, data entrada, validade) foram preenchidos;
6. Usuário confirma inclusão do item;
7. Sistema autentica usuário;
8. Sistema salva as informações no banco de dados;
9. Caso de uso UC28 - Armazenar *log* no sistema;
10. É exibida uma mensagem de estoque cadastrado. Caso de uso termina.

### **Curso Alternativo**

1 a 6: A operação pode ser cancelada. Caso de uso termina.

### **Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.
- 5.1: Campo obrigatório não foi preenchido. Sistema informa ao usuário;
- 7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

### **Pós-Condições**

Mantimento de estoque cadastrado no banco de dados do sistema.

### 3.2.4.3.2.6 Caso de uso UC06 - Alterar Estoque

#### **Sumário**

Inicia-se quando o usuário solicita a interface de consulta de estoque por motivo de erro ou outro afim de manter os dados atualizados no sistema.

#### **Ator**

Usuário

#### **Pré-condições**

Usuário cadastrado e logado com permissão para realizar a operação;  
Mantimento de estoque cadastrado no sistema.

#### **Curso Normal**

1. UC08 - Consultar estoque;
2. Usuário seleciona mantimento desejado;
3. Usuário escolhe a opção “editar” e altera os dados desejados;
4. Usuário salva alteração;
5. Sistema verifica se os campos obrigatórios foram preenchidos;
6. Sistema autentica usuário;
7. Sistema salva as informações no banco de dados;
8. Caso de uso UC28 - Armazenar log no sistema;
9. É exibida a mensagem de mantimento salvo. Caso de uso termina.

#### **Curso Alternativo**

1 a 4: Operação pode ser cancelada. Caso de uso termina.

#### **Curso de Exceção**

- 5.1: Campos obrigatórios não foram preenchidos. Sistema informa ao usuário;
- 6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Mantimento do estoque alterado no banco de dados do sistema.

**3.2.4.3.2.7 Caso de uso UC7 - Excluir Estoque****Sumário**

Inicia-se quando o usuário deseja excluir um mantimento do estoque por erro no cadastro ou motivo específico, excluindo o mantimento de estoque do banco de dados do sistema.

**Ator**

Usuário

**Pré-condições**

Usuário cadastrado e logado no sistema;

Mantimento de estoque cadastrado no bando de dados do sistema.

**Curso Normal**

1. Caso de uso UC08 - Consultar Estoque;
2. Usuário seleciona mantimento desejado;
3. Usuário escolhe operação de exclusão;
4. Sistema informa ao usuário mensagem de confirmação de exclusão;
5. Usuário confirma operação de exclusão;
6. Sistema autentica usuário;
7. Sistema exclui as informações no banco de dados do sistema;
8. Caso de uso UC28 - Armazenar *log* no sistema;
9. Sistema exibe mensagem de mantimento removido. Caso de uso termina.

**Curso Alternativo**

- 1 ao 5: Operação pode ser cancelada pelo usuário. Caso de uso termina.



**Curso de Exceção**

6.1: Sessão expirou. Sistema exhibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Mantimento do estoque excluído do banco de dados do sistema.

**3.2.4.3.2.8 Caso de uso UC8 - Consultar estoque****Sumário**

Inicia-se quando o usuário solicita ao sistema a interface de consulta de estoque. Permite que seja visualizado todos os itens cadastrados no estoque da instituição.

**Ator**

Usuário

**Pré-condições**

Usuário logado no sistema com permissão para realizar a operação.

**Curso Normal**

1. Usuário solicita ao sistema a interface de consulta do estoque;
2. Sistema autentica usuário;
3. Sistema apresenta na tela listagem disponível no banco de dados. Caso de uso termina.

**Curso Alternativo**

- 3: Operação pode ser cancelada. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exhibe interface de *login*. Caso de uso termina.
- 3.1: Sistema não encontra resultados. Sistema exhibe mensagem na tela. Caso de

uso termina.

### **Pós-Condições**

Resultados da consulta são exibidos na tela que poderá ser visualizado, alterado, ou excluído do banco de dados do sistema.

#### **3.2.4.3.2.9 Caso de uso UC9 - Relatório Vencimentos Estoque (FALTA FAZER)**

### **Sumário**

Inicia-se quando um usuário solicita ao sistema a emissão de um relatório de vencimento dos mantimentos. Os usuários com permissão poderão visualizar as informações do relatório solicitado.

### **Ator**

Usuário

### **Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Mantimento cadastrado no sistema.

### **Curso Normal:**

1. Usuário solicita ao sistema a interface de emissão de relatório de vencimento mantimento;
2. Sistema autentica usuário;
3. Sistema exibe interface solicitada;
4. Usuário insere o filtro desejado (nome, tipo, vencimento) do mantimento;
5. Usuário confirma a emissão do relatório;
6. Sistema autentica usuário;
7. Sistema exibe os resultados na tela. Caso de uso termina.

### **Curso Alternativo**

1 a 5: A operação pode ser cancelada. Caso de uso termina;

### **Curso de Exceção**

2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

### **Pós-Condições**

Relatório é exibido na tela para visualização podendo ser impresso posteriormente.

3.2.4.3.2.10 Caso de uso UC10 - Relatório cestas disponíveis (FALTA FAZER)

### **Sumário**

Inicia-se quando um usuário solicita ao sistema a emissão de um relatório de cestas disponíveis. Os usuários com permissão poderão visualizar as informações do relatório solicitado.

### **Ator**

Usuário

### **Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;

Mantimento cadastrado no sistema;

### **Curso Normal**

1. Usuário solicita ao sistema a interface de emissão de relatório de cestas disponíveis;

2. Sistema autentica usuário;

3. Sistema exibe interface solicitada;

4. Usuário confirma a emissão do relatório;

5. Sistema autentica usuário;

6. Sistema exibe os resultados na tela. Caso de uso termina.

**Curso Alternativo**

1 a 4: A operação poder ser cancelada. Caso de uso termina;

**Curso de Exceção**

2.1: Sessão expirou. Sistema exhibe interface de *login*. Caso de uso termina;

5.1: Sessão expirou. Sistema exhibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Relatório é exibido na tela para visualização podendo ser impresso posteriormente.

3.2.4.3.2.11 Caso de uso UC11 - Relatório itens pendentes das cestas (FALTA FAZER)

**Sumário**

Inicia-se quando um usuário solicita ao sistema a emissão de um relatório de itens pendentes das cestas básicas. Os usuários com permissão poderão visualizar as informações do relatório solicitado.

**Ator**

Usuário

**Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Mantimento cadastrado no sistema;

**Curso Normal**

1. Usuário solicita ao sistema a interface de emissão de relatório dos itens pendentes na cesta;
2. Sistema autentica usuário;
3. Sistema exhibe interface solicitada;

4. Usuário insere o filtro desejado (nome, tipo) do mantimento;
5. Usuário confirma a emissão do relatório;
6. Sistema autentica usuário;
7. Sistema exibe os resultados na tela. Caso de uso termina.

**Curso Alternativo:**

- 1 a 5: A operação poder ser cancelada. Caso de uso termina;

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

**Pós-Condições**

Relatório é exibido na tela para visualização podendo ser impresso posteriormente.

**3.2.4.3.2.12 Caso de uso UC12 - Cadastrar Definição de Cesta****Sumário**

Inicia-se quando o usuário solicita a interface de cadastro da definição de cesta básica. O usuário com permissão para realizar o comando poderá incluir qualquer tipo e quantidade de mantimento no sistema.

**Ator**

Usuário

**Pré-condições**

Usuário cadastrado e logado com permissão para realizar a operação.

**Curso Normal**

1. Usuário solicita ao sistema a interface de cadastro de definição da cesta;
2. Sistema autentica usuário;

3. Sistema exibe a interface solicitada;
4. Usuário insere dados (nome, tipo, quantidade mínima para cesta, medida) para incluir uma nova definição da cesta;
5. Sistema verifica se os campos obrigatórios (nome e quantidade mínima para cesta) foram preenchidos;
6. Usuário confirma cadastro do item;
7. Sistema autentica usuário;
8. Sistema salva as informações no banco de dados;
9. Caso de uso UC28 - Armazenar *log* no sistema;
10. É exibida uma mensagem de mantimento cadastrado. Caso de uso termina.

### **Curso Alternativo**

1 a 6: A operação pode ser cancelada. Caso de uso termina.

### **Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.
- 5.1: Campo obrigatório não foi preenchido. Sistema informa ao usuário;
- 7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

### **Pós-Condições**

Definição da cesta cadastrado no banco de dados do sistema.

#### **3.2.4.3.2.13 Caso de uso UC13 - Alterar Definição de Cesta**

### **Sumário**

Inicia-se quando o usuário solicita a interface de consulta da definição de cesta básica para alterar um item de por motivo de erro ou outro afim de manter os dados atualizados no sistema.

### **Ator**

Usuário

**Pré-condições**

Usuário cadastrado e logado com permissão para realizar a operação;  
Item de definição de cesta cadastrado no sistema.

**Curso Normal**

1. UC15 - Consultar Definição de Cesta;
2. Usuário seleciona mantimento desejado;
3. Usuário escolhe a opção “editar” e altera os dados desejados;
4. Usuário salva alteração;
5. Sistema verifica se os campos obrigatórios foram preenchidos;
6. Sistema autentica usuário;
7. Sistema salva as informações no banco de dados;
8. Caso de uso UC28 - Armazenar log no sistema;
9. É exibida a mensagem de mantimento salvo. Caso de uso termina.

**Curso Alternativo**

- 1 a 4: Operação pode ser cancelada. Caso de uso termina.

**Curso de Exceção**

- 5.1: Campos obrigatórios não foram preenchidos. Sistema informa ao usuário;  
6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Definição da cesta alterado no banco de dados do sistema.

**3.2.4.3.2.14 Caso de uso UC14 - Excluir Definição de Cesta****Sumário**

Inicia-se quando o usuário deseja excluir um item de definição de cesta básica por erro no cadastro ou motivo específico, excluindo o mantimento base do banco de

dados do sistema.

### **Ator**

Usuário

### **Pré-condições**

Usuário cadastrado e logado no sistema;

Item de definição de cesta cadastrado no bando de dados do sistema.

### **Curso Normal**

1. Caso de uso UC15 - Consultar Definição de Cesta;
2. Usuário seleciona mantimento desejado;
3. Usuário escolhe operação de exclusão;
4. Sistema informa ao usuário mensagem de confirmação de exclusão;
5. Usuário confirma operação de exclusão;
6. Sistema autentica usuário;
7. Sistema exclui as informações no banco de dados do sistema;
8. Caso de uso UC28 - Armazenar *log* no sistema;
9. Sistema exibe mensagem de mantimento removido. Caso de uso termina.

### **Curso Alternativo**

1 ao 5: Operação pode ser cancelada pelo usuário. Caso de uso termina.

### **Curso de Exceção**

6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

### **Pós-Condições**

Item de definição da cesta excluído do banco de dados do sistema.

3.2.4.3.2.15 Caso de uso UC15 - Consultar Definição de Cesta



**Sumário**

Inicia-se quando o usuário solicita ao sistema a interface de consulta da definição de cesta básica. Permite que seja visualizado todos os itens cadastrados que compõe uma cesta básica.

**Ator**

Usuário

**Pré-condições**

Usuário logado no sistema com permissão para realizar a operação.

**Curso Normal**

1. Usuário solicita ao sistema a interface de consulta de definição de cesta;
2. Sistema autentica usuário;
3. Sistema apresenta na tela a listagem disponível no banco de dados. Caso de uso termina.

**Curso Alternativo**

- 3: Operação pode ser cancelada. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exhibe interface de *login*. Caso de uso termina.
- 3.1: Sistema não encontra resultados. Sistema exhibe mensagem na tela. Caso de uso termina.

**Pós-Condições**

Resultados da consulta são exibidos na tela que poderá ser visualizado, alterado, ou excluído do banco de dados do sistema.

## Sumário

Inicia-se quando o usuário solicita a interface de cadastro de família.

## Ator

Usuário

## Pré-condições

Usuário cadastrado e logado no sistema com permissão para realizar a operação.

## Curso Normal

1. Usuário solicita ao sistema a interface de cadastro de família;
2. Sistema autentica o usuário;
3. Sistema exibe a interface solicitada;
4. Usuário insere dados (RD01, RD02 e RD03) para cadastrar uma nova família;
5. Sistema verifica se os campos obrigatórios: status, nome, data de nascimento, endereço, número, bairro e cidade foram digitados.
6. Usuário confirma operação de inclusão da família;
7. Sistema autentica usuário;
8. Sistema salva as informações no Banco de Dados;
9. Caso de uso UC28 - Armazenar *log*;
10. É exibida uma mensagem de família cadastrada. Caso de uso termina.

## Curso Alternativo

1 a 6: O usuário pode cancelar a operação. Caso de uso termina;

## Curso de Exceção

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 5.1: Os campos obrigatórios não foram preenchidos. Sistema informa ao usuário;
- 7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

## Pós-Condições

Família cadastrada no banco de dados do sistema.

#### 3.2.4.3.2.17 Caso de uso UC17 - Cadastrar Frequência Família(revisar)

##### **Sumário**

Inicia-se quando um usuário solicita a interface de consulta de encontro. Permite a inserção de presenças para as famílias, para o controle do interesse em receber os auxílios, além de servir como análise para tornar a família apta ou não.

##### **Ator**

Usuário

##### **Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no banco de dados do sistema.

##### **Curso Normal**

1. Usuário solicita ao sistema a interface de consulta de encontro;
2. Caso de uso UC22 - Consultar Frequência Família;
3. Sistema autentica o usuário;
4. Sistema exibe a interface solicitada;
5. Usuário insere a presença para o dia correspondente;
6. Sistema verifica se o campo foi preenchido;
7. Usuário confirma operação de inclusão da frequência;
8. Sistema autentica usuário;
9. Sistema salva as informações no banco de dados;
10. Caso de uso UC28 - Armazenar *Log*;
11. É exibida uma mensagem de operação realizada com sucesso. Caso de uso termina.

##### **Curso Alternativo**

- 1 a 7: O usuário pode cancelar a operação. Caso de uso termina.

**Curso de Exceção**

3.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

6.1: Usuário não preencheu o dia da presença. Sistema informa ao usuário;

8.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Frequência da família cadastrada no banco de dados do sistema.

**3.2.4.3.2.18 Caso de uso UC18 - Alterar Família****Sumário**

Inicia-se quando um usuário com permissão para tal solicita a interface de consulta de família devido a erros ou afim de atualizações.

**Ator**

Usuário

**Pré-condições:**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no sistema.

**Curso Normal**

1. Caso de uso UC21 - Consultar Família;
2. Sistema exibe listagem com os nomes das famílias na tela;
3. Usuário seleciona a família a ser alterada e escolhe “editar”;
4. Usuário altera os dados desejados;
5. Sistema verifica se os campos obrigatórios: status, nome, data de nascimento, endereço, número, bairro e cidade foram preenchidos;
6. Usuário confirma alteração de família;
7. Sistema autentica usuário;

8. Sistema salva as informações no banco de dados;
9. Caso de uso UC28 - Armazenar *log* no sistema;
10. É exibida uma mensagem de operação realizada com sucesso. Caso de uso termina.

### **Curso Alternativo**

1 a 6: Operação pode ser cancelada. Caso de uso termina;

### **Curso de Exceção**

- 5.1: O campo obrigatório não foi preenchido. Sistema informa ao usuário;
- 7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

### **Pós-Condições**

Família alterada no banco de dados do sistema.

#### 3.2.4.3.2.19 Caso de uso UC19 - Alterar Frequência Família(revisar)

### **Sumário**

Inicia-se quando o usuário solicita a interface de consulta de encontro por motivo de erro, afim de manter os dados atualizados no sistema.

### **Ator**

Usuário

### **Pré-condições:**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no sistema;  
frequência de família cadastrada no sistema;

### **Curso Normal**

1. Caso de uso UC22 - Consultar Frequência Família;

2. Usuário altera os dados desejados;
3. Usuário confirma operação de alteração da frequência da família;
4. Sistema autentica usuário;
5. Sistema salva as informações no banco de dados;
6. Caso de uso UC28 - Armazenar *log* no sistema;
7. É exibida uma mensagem de operação realizada com sucesso. Caso de uso termina.

### **Curso Alternativo**

1 a 3: A operação pode ser cancelada. Caso de uso termina;

### **Curso de Exceção**

4.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

### **Pós-Condições**

Frequência de família alterada no banco de dados do sistema.

#### 3.2.4.3.2.20 Caso de uso UC20 - Excluir Família

### **Sumário**

Inicia-se quando o usuário solicita ao sistema a interface de consulta de família afim de o mesmo não mais necessitar de auxílio ou por motivo de mudança de endereço ou motivo maior.

### **Ator**

Usuário

### **Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no sistema .

**Curso Normal**

1. Caso de uso UC21 - Consultar Família;
2. Sistema exibe listagem das famílias cadastradas na tela;
3. Usuário seleciona família desejada;
4. Usuário seleciona operação de exclusão;
5. Sistema informa ao usuário mensagem de confirmação de exclusão;
6. Usuário confirma operação de exclusão;
7. Sistema autentica usuário;
8. Sistema exclui informações do banco de dados do sistema;
9. Caso de uso UC28 - Armazenar *log* no sistema;
10. Sistema exibe mensagem de operação realizada com sucesso. Caso de uso termina.

**Curso Alternativo**

1 ao 6: Operação pode ser cancelada pelo usuário. Caso de uso termina.

**Curso de Exceção**

7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.

**Pós-Condições**

Família excluída do banco de dados do sistema.

**3.2.4.3.2.21 Caso de uso UC21 - Consulta Família****Sumário**

Inicia-se quando o usuário solicita a interface de consulta de família. É utilizada para ver a listagem das famílias cadastradas além de saber quais estão aptas e não aptas, situações de moradia, e outros.

**Ator**

Usuário

**Pré-condições:**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
 Família cadastrada no sistema.

**Curso Normal**

1. Usuário solicita ao sistema interface de consulta de família;
2. Sistema autentica usuário;
3. Sistema apresenta na tela a listagem disponível no banco de dados. Caso de uso termina.

**Curso Alternativo:**

- 3: Operação pode ser cancelada. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 3.1: Sistema não encontra resultados a partir do filtro escolhido. Sistema exibe mensagem na tela. Caso de uso termina.

**Pós-Condições**

Família consultada poderá ser visualizada, alterada ou excluída do banco de dados do sistema;

### 3.2.4.3.2.22 Caso de uso UC22 - Consulta Frequência Família

**Sumário**

Inicia-se quando o usuário solicita a interface de consulta frequência família. Utilizada para ver se a família está participando ou não das reuniões afim de um estudo maior para saber se a família esta apta ou não para receber a cesta básica.

**Ator**



Usuário

### **Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no sistema.

### **Curso Normal**

1. Caso de uso UC21 - Consultar Família;
2. Sistema autentica usuário;
3. Sistema apresenta na tela listagem disponível no banco de dados. Caso de uso termina.

### **Curso Alternativo**

- 3: Operação pode ser cancelada. Caso de uso termina.

### **Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 3.1: Sistema não encontra resultados a partir do filtro escolhido. Sistema exibe mensagem na tela. Caso de uso termina.

### **Pós-Condições**

Frequência de família consultada poderá ser visualizada, alterada e excluída.

3.2.4.3.2.23 Caso de uso UC23 - Relatório Vencimento Matrícula(FALTA REVISAR)

### **Sumário**

Inicia-se quando o usuário solicita ao sistema a emissão de um relatório de vencimento de matrícula. Os usuários com permissão poderão visualizar as informações do relatório solicitado.

**Ator**

Usuário

**Pré-condições**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no sistema.

**Curso Normal**

1. Usuário solicita ao sistema a interface de emissão de relatório de vencimento de matrícula;
2. Sistema autentica usuário;
3. Sistema exibe interface solicitada;
4. Usuário confirma a emissão do relatório;
5. Sistema autentica usuário;
6. Sistema exibe os resultados na tela. Caso de uso termina.

**Curso Alternativo**

1 a 4: A operação poder ser cancelada. Caso de uso termina;

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 5.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

**Pós-Condições**

Relatório é exibido na tela para visualização podendo ser impresso posteriormente.

3.2.4.3.2.24 Caso de uso UC24 - Relatório Famílias aptas(FALTA REVISAR)

**Sumário**

Inicia-se quando um usuário solicita ao sistema a emissão de um relatório das famílias aptas. Os usuários com permissão poderão visualizar as informações do

relatório solicitado, afim de saber quais famílias podem ou não receber a cesta básica.

**Ator**

Usuário

**Pré-condições:**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Família cadastrada no sistema.

**Curso Normal**

1. Usuário solicita ao sistema a interface de emissão de relatório de famílias aptas;
2. Sistema autentica usuário;
3. Sistema exibe interface solicitada;
4. Usuário confirma a emissão do relatório;
5. Sistema autentica usuário;
6. Sistema exibe os resultados na tela. Caso de uso termina.

**Curso Alternativo**

1 a 4: A operação poder ser cancelada. Caso de uso termina;

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 5.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

**Pós-Condições**

Relatório é exibido na tela para visualização podendo ser impresso posteriormente.

3.2.4.3.2.25 Caso de uso UC25 - Efetuar Login

**Sumário**

Inicia-se quando um usuário solicitar acesso ao sistema. Através do *login* somente os usuários previamente cadastrados e ativos no sistema terão acesso as funcionalidades disponíveis

### **Atores**

Administrador e Usuário

### **Pré-condições**

Usuário cadastrado no sistema.

### **Curso Normal**

1. Usuário informa URL do sistema no navegador;
2. Sistema exibe a página inicial de *login* e senha;
3. Usuário preenche os campos: *e-mail* e senha;
4. Sistema verifica se os campos foram digitados;
5. Usuário confirma a operação de *login*;
6. Sistema valida o *e-mail* e senha;
7. Sistema autentica o usuário com as devidas permissões. Caso de uso termina.

### **Curso Alternativo**

1 a 5: Operação pode ser cancelada. Caso de uso termina.

### **Curso de Exceção**

- 4.1: *E-mail* ou senha não foram preenchidos. Sistema informa ao usuário;
- 6.1: *E-mail* ou senha incorretos. Sistema informa ao usuário. Caso de uso termina.

### **Pós-Condições**

Usuário autenticado no sistema com as devidas permissões.

3.2.4.3.2.26 Caso de uso UC26 - Recuperar Senha

**Sumário**

O caso de uso inicia quando um usuário não autenticado ou um usuário que esqueceu sua senha, solicita a recuperação da senha para o sistema que será enviado a um *e-mail* previamente cadastrado.

**Atores**

Administrador e Usuário

**Pré-condições:**

Usuário cadastrado no sistema com *e-mail* válido.

**Curso Normal**

1. Usuário solicita interface de recuperação de senha;
2. Sistema exibe interface solicitada;
3. Usuário informa *e-mail* cadastrado no sistema;
4. Sistema verifica o campo preenchido;
5. Sistema valida o campo;
6. Usuário confirma a operação da solicitação;
7. Sistema envia *e-mail* para o endereço cadastrado juntamente com uma nova senha para o usuário;
8. Sistema exibe mensagem de envio de *e-mail*. Caso de uso termina.

**Curso Alternativo**

1 ao 5 : A operação pode ser cancelada. Caso de uso termina.

**Curso de Exceção**

4.1: O campo obrigatório não foi preenchido. Sistema informa ao usuário;

**Pós-Condições**

Senha recuperada e enviada por *e-mail* para que o usuário possa logar no sistema.

#### 3.2.4.3.2.27 Caso de uso UC27 - Armazenar *log*

##### **Sumário**

Inicia-se quando as operações de consulta, alteração, e exclusão são realizadas.

##### **Ator**

Sistema

##### **Pré-condições**

Operação de inclusão, alteração, exclusão.

##### **Curso Normal**

1. Sistema busca nome do usuário que realizou a operação;
2. Sistema busca hora e data atual;
3. Sistema armazena data e hora, nome do usuário, tipo de operação, tabela e registro modificados no banco de dados. Caso de uso termina.

##### **Pós-condições**

Informações referentes às modificações gravadas no banco de dados.

#### 3.2.4.3.2.28 Caso de uso UC28 - Consultar *log* do Sistema

##### **Sumário**

Inicia quando o usuário deseja visualizar as modificações feitas no sistema através do *log*. O administrador poderá visualizá-lo permitindo realizar estudos sobre o uso do sistema.

##### **Ator**

Administrador

##### **Pré-condições**

Usuário cadastrado no sistema e logado com permissão para realizar a operação;  
*Log* cadastrado no sistema.

**Curso Normal**

1. Usuário solicita interface de consulta de *log* do sistema;
2. Sistema autentica usuário;
3. Sistema exibe os resultados da busca na tela. Caso de uso termina.

**Curso Alternativo:**

- 3: A operação pode ser cancelada pelo usuário. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina.
- 3.1: Sistema não encontra registros para serem exibidos. Sistema informa ao usuário. Caso de uso termina.

**Pós-condições**

Informações buscadas são exibidas na tela.

**3.2.4.3.2.29 Caso de uso UC29 - Cadastrar Voluntário****Sumário**

Inicia-se quando um usuário solicita a interface de cadastro de voluntário. O administrador do sistema será o responsável por adicionar.

**Ator**

Administrador

**Pré-condições**

Usuário logado no sistema e com permissão para realizar a operação.

**Curso Normal**

1. Administrador solicita ao sistema a interface de cadastro de voluntário;
2. Sistema autentica administrador;
3. Sistema exibe a interface solicitada;
4. Administrador insere todos os dados (RD07);
5. Sistema verifica se os campos obrigatórios (nome e prestação de serviço) foram digitados;
6. Administrador confirma operação de inclusão de cadastro;
7. Sistema autentica usuário;
8. Sistema salva as informações no banco de dados;
9. Caso de uso UC28 - Armazenar *log*;
10. É exibida uma mensagem de operação realizada com sucesso. Caso de uso termina.

**Curso Alternativo**

- 1 a 6: O Administrador pode cancelar a operação. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 5.1: Os campos obrigatórios não foram preenchidos. Sistema informa ao usuário;
- 7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

**Pós-Condições**

Voluntário cadastrado no banco de dados do sistema.

**3.2.4.3.2.30 Caso de uso UC30 - Alterar Voluntário****Sumário**

Inicia-se quando o administrador solicita a interface de consulta de voluntário. Possibilita ao administrador alterar os dados do voluntário por algum motivo, mantendo o cadastro sempre atualizado.



**Ator**

Administrador

**Pré-condições**

Administrador cadastrado e logado no sistema com permissão para realizar a operação;

Voluntário cadastrado no sistema.

**Curso Normal**

1. Caso de uso UC32 - Consultar voluntário;
2. Sistema exibe listagem dos voluntários na tela;
3. Administrador seleciona voluntário desejado e escolhe “editar”;
4. Administrador altera os dados desejados;
5. Sistema verifica se os campos obrigatórios (nome e prestação de serviço) foram preenchidos;
6. Administrador confirma a operação de alteração;
7. Sistema autentica administrador;
8. Sistema salva as informações no banco de dados;
9. Caso de uso UC28 - Armazenar *log* no sistema;
10. É exibida uma mensagem Voluntário salvo. Caso de uso termina.

**Curso Alternativo**

- 1 a 8: Operação pode ser cancelada. Caso de uso termina;

**Curso de Exceção**

- 5.1: O campo obrigatório não foi preenchido. Sistema informa ao usuário;
- 7.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

**Pós-Condições**

Voluntário alterado no banco de dados do sistema.

### 3.2.4.3.2.31 Caso de uso UC31 - Excluir Voluntário

#### **Sumário**

Inicia-se quando o Administrador solicita ao sistema a interface de consulta de voluntário afim de excluir o mesmo do sistema.

#### **Ator**

Administrador

#### **Pré-condições**

Administrador cadastrado e logado no sistema;

Voluntário cadastrado no sistema.

#### **Curso Normal**

1. Caso de uso UC32 - Consultar Voluntário;
2. Sistema exibe listagem dos voluntários na tela;
3. Administrador seleciona voluntário desejado e escolhe “excluir”;
4. Sistema informa ao administrador mensagem de confirmação de exclusão;
5. Administrador confirma operação de exclusão;
6. Sistema autentica administrador;
7. Sistema exclui informações do banco de dados do sistema;
8. Caso de uso UC28 - Armazenar *log* no sistema;
9. Sistema exibe mensagem de operação realizada com sucesso. Caso de uso termina.

#### **Curso Alternativo**

- 1 ao 5: Operação pode ser cancelada pelo administrador. Caso de uso termina.

#### **Curso de Exceção**

- 6.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

**Pós-Condições**

Voluntário excluído do banco de dados do sistema.

**3.2.4.3.2.32 Caso de uso UC32 - Consultar Voluntário****Sumário**

Inicia-se quando o administrador solicita a interface de consulta de voluntário. É utilizada para ver a listagem dos voluntários cadastrados além de saber quais estão ativos e inativos.

**Ator**

Administrador

**Pré-condições**

Administrador cadastrado e logado no sistema;

Voluntário cadastrado no sistema.

**Curso Normal:**

1. Administrador solicita ao sistema interface de consulta de voluntário;
2. Sistema autentica administrador;
3. Sistema apresenta na tela listagem disponível no banco de dados. Caso de uso termina.

**Curso Alternativo**

- 3: Operação pode ser cancelada. Caso de uso termina.

**Curso de Exceção**

- 2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;
- 3.1: Sistema não encontra resultados a partir do filtro escolhido. Sistema exibe mensagem na tela. Caso de uso termina.

**Pós-Condições**

Voluntário consultado poderá ser visualizado, alterado ou excluído do banco de dados do sistema;

**3.2.4.3.2.33 Caso de uso UC33 - Relatório Voluntário****Sumário**

Inicia-se quando um usuário solicita ao sistema a emissão de um relatório de voluntários. Os usuários com permissão poderão visualizar as informações do relatório solicitado.

**Atores**

Administrador e Usuário

**Pré-condições:**

Usuário cadastrado e logado no sistema com permissão para realizar a operação;  
Voluntário cadastrado no sistema.

**Curso Normal**

1. Usuário solicita ao sistema a interface de emissão de relatório de voluntário;
2. Sistema autentica usuário;
3. Sistema exibe interface solicitada;
4. Usuário confirma a emissão do relatório;
5. Sistema autentica usuário;
6. Sistema exibe os resultados na tela. Caso de uso termina.

**Curso Alternativo**

1 a 4: A operação poder ser cancelada. Caso de uso termina;

**Curso de Exceção**

2.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

5.1: Sessão expirou. Sistema exibe interface de *login*. Caso de uso termina;

### **Pós-Condições**

Relatório é exibido na tela para visualização podendo ser impresso posteriormente.

### **3.2.5 Requisitos de dados**

#### **3.2.5.1 Lista de requisitos de Dados**

RD01: O sistema terá um cadastro dos responsáveis pelas famílias contendo os seguintes campos que correspondem aos seguintes dados: Nome, CPF, Data de Nascimento, Endereço, Número, Complemento, Bairro, Cidade, Referência, Telefone, Escolaridade, Profissão, Ocupação, Está trabalhando (Sim/Não), Situação conjugal - companheiro(a) (Sim/Não)

RD01.1: A situação conjugal pode assumir os seguintes campos: Escolaridade, Profissão, Ocupação, Está trabalhando (Sim/Não), Benefícios(Sim/Não), Possui dependente (Sim/Não), O(a) companheiro(a) é o pai/mãe da(s) criança(s) (Sim/Não);

RD02: O sistema terá um cadastro da composição familiar contendo os seguintes campos: Nome, Data de nascimento, Parentesco, Escolaridade, Escola, Manequim/Calçando, Ocupação, Renda, Renda Familiar, Renda per capita;

RD03: O sistema terá um cadastro sobre a situação de moradia das famílias contendo os seguintes campos que corresponde aos seguintes dados:

RD03.1: Condição habitacional pode assumir os seguintes valores: aluguel, casa própria, cedida;

RD03.2: O Tipo de construção pode assumir os seguintes valores: alvenaria, madeira, outro;

RD03.3: Dados complementares sobre a situação de moradia: Número de cômodos, Banheiro (Sim/Não, outro), Possui água e esgoto encanado (Sim/Não), Possui energia elétrica (Sim/Não), Casa ou terreno em risco (Sim/Não), Usa filtro de

água (Sim/Não), Tem horta caseira (Sim/Não);

RD03.4: Questão de vulnerabilidade pode assumir os seguintes valores: Conselho tutelar (a relatar), violência (a relatar), presidiário (a relatar), vivência de rua (a relatar), abrigo (a relatar);

RD03.5: Dados referentes a documentação pessoal:

RD03.5.1: A carteira de identidade, carteira de trabalho e CPF, para maiores de idade pode assumir os seguintes valores: Todos possuem; Em caso negativo, nomes daqueles que não possuem: Data de encaminhamento ao órgão necessário;

RD03.5.2: A certidão de nascimento para crianças pode assumir os seguintes valores: Todas possuem; Em caso negativo, nomes daqueles que não possuem: Data de encaminhamento ao órgão necessário;

RD03.5.2.1: A carteira de vacinação pode assumir os seguintes valores: Em dia; Em caso negativo, nome das crianças com vacinas atrasadas; Data de encaminhamento ao posto de saúde;

RD03.5.3: O sistema terá um cadastro referente ao ensino, que pode assumir os seguintes valores: Todas as crianças e jovens estão frequentando as aulas; Em caso negativo, nome daqueles que não estão frequentando as aulas: Motivo; Data do contato com a escola visando reinserção do aluno;

RD03.6: Dados referente a saúde:

RD03.6.1: O sistema terá um cadastro referente a situação nutricional das crianças até 7 anos, que pode assumir os seguintes valores: Nome, idade, peso, altura, situação normal (Sim/Não);

RD03.6.2: O sistema terá um cadastro referente a doenças na família, que pode assumir os seguintes valores: Doenças encontradas (Obesidade, pressão alta, Diabetes, Alcoolismo/Dependências químicas, Deficiência Mental, Deficiência de locomoção), Nome, idade, Em tratamento (Sim/Não), Encaminhamentos necessários;

RD03.7: O sistema terá um cadastro referente as condições de higiene da atual residência, que pode assumir os seguintes valores: Limpeza da casa (bom, regular, péssimo), Animais domésticos (Cachorro, gato, galinha, porco, cabrito, cavalo), Cuidados (Bem/Mal), Acondicionamento do lixo (Bom/Ruim), Providências necessárias;

RD03.8: O sistema terá um cadastro referente ao planejamento familiar, que pode assumir os seguintes valores: Gravidez na adolescência, Uso de método contraceptivo (qual), Necessidade de orientação médica para o planejamento de gravidez (Data do encaminhamento);

RD03.9: O sistema terá um cadastro referente a situação de risco familiar, que pode assumir os seguintes valores: Necessidade de encaminhamento do serviço de psicologia (nome, motivo, data do encaminhamento), Necessidade de encaminhamento ao conselho da criança e adolescente (nome, motivo, data do encaminhamento);

RD03.10: O sistema terá um cadastro referente a capacitação para o trabalho, que pode assumir os seguintes valores: Curso técnico/profissionalizante já realizado, Gostaria de fazer algum (Sim/Não, Qual);

RD03.11: O sistema terá um cadastro referente a necessidade de equipamentos sociais públicos, que pode assumir os seguintes valores: saúde, lazer, atenção p/ adolescentes, ensino (qual), creche, atenção p/ idosos;

RD03.12: O sistema terá um cadastro referente a organização comunitária, que pode assumir os seguintes valores: Existente no local (nome), Participa (Sim/Não), Qual tipo; O que é mais importante para sua família neste momento? O que pretende fazer a respeito (a relatar), O que você gostaria que fosse oferecido pela instituição (Cursos, materiais, serviços, outro), Está escrito no programa bolsa família (Sim/Não), recebe ajuda de outra instituição (Não, Sim (qual), Observações, Já se cadastrou na EMCASA para compra de lote ou casa (Sim/Não), Data Visita fraterna, apta, não apta, pendente;

RD04: O sistema terá um cadastro de definição da cesta contendo os seguintes campos: tipo, nome, quantidade mínima para cesta e medida;

RD05: O sistema terá um cadastro de usuários contendo os seguintes campos: Situação, perfil, prestação de serviço, nome, *e-mail*, senha, CPF, RG, endereço, número, complemento, bairro, cidade, UF, CEP, telefone, celular;

RD06: O sistema terá um cadastro de frequência de famílias contendo os seguintes campos: Código e data da presença;

RD07: O sistema terá um cadastro de voluntários contendo os seguintes campos: Nome, e-mail, endereço, complemento, n°, bairro, cidade, UF, CEP,

telefone, celular e prestação de serviço.

### 3.2.5.2 Modelo Conceitual de Dados

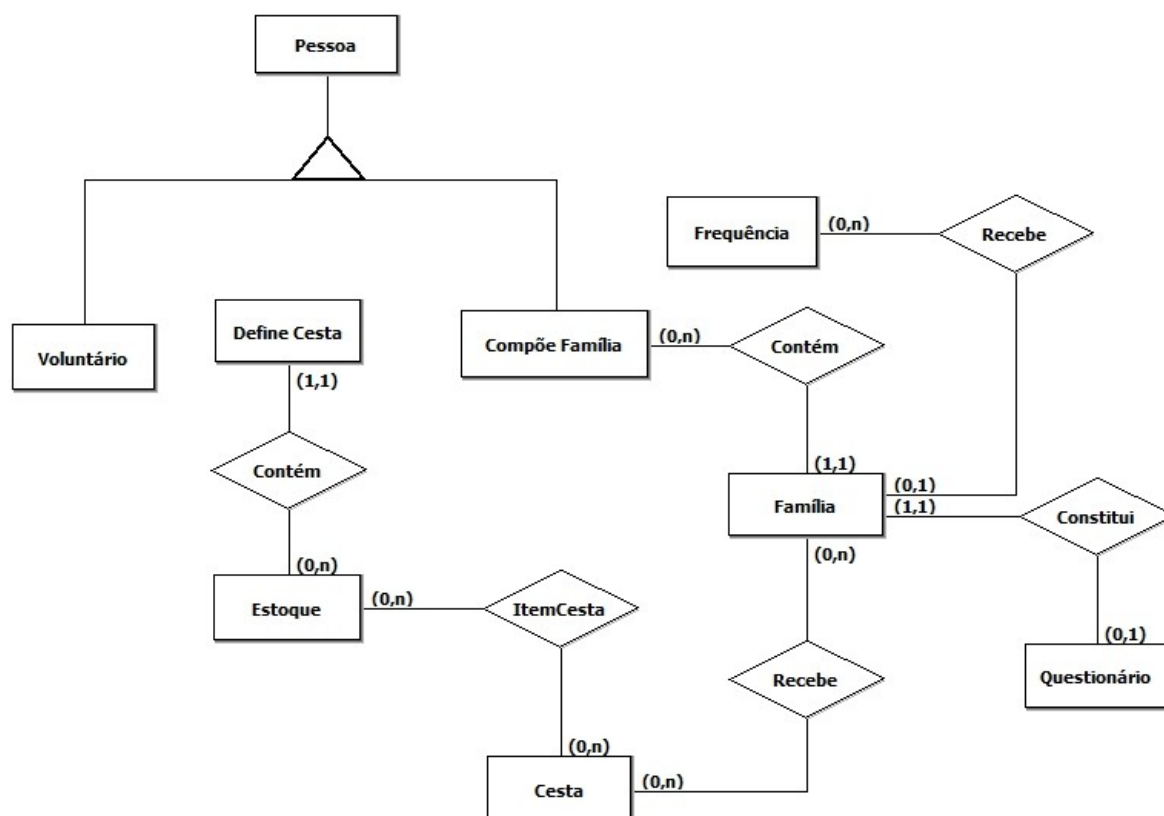


Figura 15 – Moledo conceitual de dados (DER)

### 3.2.5.3 Entidades e Atributos

#### 3.2.5.3.1 Entidade Usuário

Denominação: Usuário

Descrição: Representa todas as pessoas que assumem perfis no *software*.

Entidade Usuário		
Atributo	Tipo	Descrição



Identificação	número inteiro	Identificação do registro
Status	booleano	Usuário está ativo ou inativo?
Perfil	texto	Tipo do perfil que o usuário pode assumir.
Prestação de Serviço	texto	Tipo de serviço que o usuário irá oferecer.
Nome	texto	Nome do usuário.
E-mail	texto	<i>E-mail</i> do usuário
Senha	texto	Senha do usuário.
CPF	texto	CPF do usuário.
RG	texto	RG do usuário.
Endereço	texto	Endereço do usuário.
Número	texto	Número da residência do usuário.
Complemento	texto	Complemento da residência, caso exista.
Bairro	texto	Bairro que o usuário reside.
Cidade	texto	Cidade em que o usuário reside.
UF	texto	Estado que o usuário encontra-se.
CEP	texto	CEP referente a residência do usuário
Telefone	texto	Número do telefone fixo.
Celular	texto	Número do celular.

Quadro 19 – Atributos do Usuário

### 3.2.5.3.2 Entidade Voluntário

Denominação: Voluntário

Descrição: Representa terceiros que contribuem com a instituição de alguma forma, seja com doações ou prestação de serviço.

**Entidade Voluntário**

Atributo	Tipo	Descrição
Identificação	número inteiro	Identificação do registro
Status	booleano	Voluntário está ativo ou inativo?
Prestação de Serviço	texto	Tipo de serviço que o Voluntário irá oferecer.
Nome	texto	Nome do voluntário.
E-mail	texto	E-mail do voluntário.
Endereço	texto	Endereço do voluntário.
Número	texto	Número da residência do voluntário.
Complemento	texto	Complemento da residência, caso exista.
Bairro	texto	Bairro que o voluntário reside.
Cidade	texto	Cidade em que o voluntário reside.
UF	texto	Estado que o voluntário encontra-se.
CEP	texto	CEP referente a residência do voluntário.
Telefone	texto	Número do telefone fixo.
Celular	texto	Número do celular.

Quadro 20 – Atributos do Voluntário

### 3.2.5.3.3 Entidade Família

Denominação: Família

Descrição: Armazena informações das famílias assistidas.

Entidade Família		
Atributo	Tipo	Descrição
Identificação	número inteiro	Identificação do registro.
Status	booleano	Família está ativa ou inativa?

Nome	texto	Nome do representante da família.
CPF	texto	CPF do representante familiar.
Data de nascimento	data	Data de nascimento do representante familiar.
Endereço	texto	Rua, avenida, praça,.. do representante familiar.
Número	texto	Número da residência do representante familiar.
Complemento	texto	Número complementar da residência.
Bairro	texto	Bairro em que o representante mora.
Referência	texto	Uma referência da residência do representante familiar.
Telefone	texto	Telefone do responsável.
Escolaridade	texto	Nível de escolaridade dos familiares, representante da família, companheiro e dependentes.
Ocupação	texto	Qual a ocupação do representante familiar.
Trabalha	booleano	Está trabalhando?
Local de trabalho	texto	Local aonde o representante familiar trabalha.
Companheiro	booleano	Possui companheiro(a)?
Nome do Companheiro	texto	Nome do companheiro(a).
Escolaridade do companheiro	texto	Escolaridade do companheiro(a).
Profissão do companheiro	texto	Profissão do companheiro(a).
Ocupação do Companheiro	texto	Ocupação do companheiro(a).
Trabalho do companheiro	booleano	O companheiro(a) está trabalhando?

Local de trabalho companheiro	texto	Local aonde o companheiro(a) está trabalhando.
Dependentes	booleano	O companheiro(a) ou o representante familiar possui dependentes?
Pai ou mãe	booleano	O companheiro(a) é o pai ou mãe da(s) criança(s)?
Nome do dependente	texto	Nome do dependente
Data de nascimento do dependente	data	Data de nascimento do dependente.
Parentesco	texto	Possui algum grau de parentesco com o representante familiar.
Estuda	booleano	O dependente estuda?
Escolaridade do dependente	texto	Escolaridade do dependente.
Nome da Escola	texto	Qual escola frequenta.
Manequim	texto	Qual o manequim e calçado do dependente.
Ocupação do dependente	texto	Possui alguma ocupação.
Renda	número	O dependente, se tiver mais que 16 anos, possui alguma renda?
Renda familiar	número	Valor total da renda familiar.
Renda percapta	número	Valor total da renda familiar dividido pelo número de pessoas.
Habitacional	texto	Condição habitacional da família.
Construção	texto	Tipo de construção da casa.
Cômodos	número inteiro	Número de cômodos da casa.
Banheiro	booleano	Possui banheiro?
Esgoto	booleano	Possui água e esgoto encanado.

Energia	booleano	Possui energia elétrica.
Risco do terreno	booleano	Casa ou terreno em risco.
Filtro de água	booleano	Usa filtro de água.
Horta	booleano	Possui horta caseira.
Vulnerabilidade	texto	Questão de vulnerabilidade.
Possui documentos	booleano	Todos os maiores de idade possuem carteira de identidade, carteira de trabalho e CPF.
Nome não possui documento	texto	Nome dos que não possuem algum documento (Carteira de identidade, carteira de trabalho e CPF).
Certidão de nascimento	booleano	Todas crianças possuem certidão de nascimento.
Nome não possui certidão	texto	Nome das crianças que não possuem certidão de nascimento.
Vacinação	booleano	Carteira de vacinação das crianças em dia.
Nome vacinação	texto	Nome das crianças com carteira de vacinação atrasada.
Data posto de saúde	data	Data de encaminhamento ao posto de saúde.
Frequência aula	booleano	Todas as crianças e jovens estão frequentando as aulas.
Motivo falta aulas	texto	Qual o motivo para que as crianças e jovens não estejam frequentando as aulas.
Nome da criança	texto	Situação nutricional. Nome da criança com idade até 7 anos.
Idade da criança	número inteiro	Situação nutricional. Idade da criança.
Peso da criança	número	Situação nutricional. Peso da criança.
Altura da criança	número	Situação nutricional. Altura da criança.
Situação nutricional	booleano	Situação nutricional. Situação da criança está normal ou não.
Doença	multivalorado	Nome da pessoa que encontra-se com

		algum tipo de doenças.
Idade do doente	número inteiro	Idade da pessoa que possui algum tipo de doença.
Tratamento	booleano	A pessoa que possui alguma doença, está tendo algum tipo de tratamento.
Limpeza	texto	Como é a limpeza da mordida.
Animais	multivalorado	Possui animais de estimação.
Cuidados dos animais	booleano	Os animais são bem ou mal cuidados.
Providências com animais	texto	Providências que devem ser tomadas referente aos animais de estimação.
Gravidez na adolescência	booleano	Possui alguma gravidez na adolescência.
Contraceptivo	booleano	Usa algum método contraceptivo.
Tipo de contraceptivo	texto	Qual método contraceptivo é utilizado.
Orientação médica	booleano	Necessidade de orientação médica para planejamento de de gravidez.
Acompanhament o médico	data	Data que foi encaminhado para o acompanhamento médico para planejamento de gravidez.
Psicologia	booleano	Necessidade de encaminhamento ao serviço de psicologia.
Nome do necessitado	texto	Nome do necessitado para atendimento psicológico.
Motivo psicologia	texto	Motivo que levaram a necessidade de um psicólogo.
Data psicologia	data	Data para o encaminhamento psicológico.
Concelho	booleano	Necessidade de encaminhamento ao concelho da criança e adolescente.
Nome do aconselhado	texto	Nome do necessitado para ser encaminhado ao concelho da criança e adolescente.

Motivo aconselhamento	texto	Motivo que levou a criança ou adolescente a ser encaminhado.
Curso técnico	texto	Possui curso técnico já realizado, qual?
Fazer curso técnico	texto	Gostaria de fazer algum curso técnico, qual?
Equipamentos sociais	multivalorado	Necessidade de equipamentos sociais públicos.
Organização comunitária	texto	Existe no local alguma organização comunitária.
Tipo de organização	texto	Participa de alguma organização comunitária, qual o tipo?
Importância familiar	texto	O que é mais importante para a família neste momento? O que pretende fazer a respeito?
Oferecer	texto	O que gostaria que fosse oferecido pela instituição?
Bolsa família	booleano	Está inscrito no programa bolsa família?
Ajuda de instituição	booleano	Recebe ajuda de outra instituição?
Qual instituição	texto	Qual instituição que recebe algum tipo de ajuda.
Observações	texto	Observações complementares que não foram supridas nas perguntas anteriores.
EMCASA	booleano	Possui cadastro na EMCASA para a compra de um lote ou casa.

Quadro 21 – Atributos da Família

### 3.2.5.3.4 Entidade Definição de Cesta

Denominação: Definição de Cesta

Descrição: Representa informações referente aos mantimentos que serão cadastrados para compor, definir, uma cesta básica.

Entidade Definição de Cesta		
Atributo	Tipo	Descrição
Identificação	número inteiro	Identificação do registro
Tipo	texto	Tipo de mantimento que será armazenado, sendo classificado em tipo 1(mantimentos que compõe a cesta básica) ou tipo 2(mantimentos extras que não fazem parte da cesta).
Nome	texto	Nome referente ao mantimento.
Quantidade	número inteiro	Quantidade de cada mantimento cadastrado.
Medida	texto	Descreve a unidade de medida relacionada ao mantimento, como kilo, litro, lata.

Quadro 22 – Atributos de Definição de Cesta

### 3.2.5.3.5 Entidade Estoque

Denominação: Estoque

Descrição: Representa informações referente aos mantimentos que serão armazenados no estoque com, bem como a validade do mantimento e sua saída do estoque.

Entidade Estoque		
Atributo	Tipo	Descrição
Identificação	número inteiro	Identificação do registro
Complemento quantidade	número inteiro	Define se o mantimento que será cadastrado é de 1, 2, 3.. 5 kilos ou litros.
Data de entrada	texto	Data referente a entrada do mantimento no estoque.
Data de vencimento	texto	Data em que o mantimento irá vencer.



Data de saída	data	Data em que o mantimento irá para a cesta.
---------------	------	--

Quadro 23 – Atributos de Estoque

## 3.2.5.3.6 Entidade Frequência

Denominação: Frequência

Descrição: Armazena informações sobre a frequência das famílias nas reuniões.

Entidade Frequência		
Atributo	Tipo	Descrição
identificação	número inteiro	Identificação do registro.
Código	texto	Identifica se o representante da família está presente, ausente nas reuniões.
Data	data	Data referente a matrícula da família.

Quadro 24 – Atributos de Frequência

## 3.2.5.3.7 Entidade Cesta

Denominação: Cesta

Descrição: Armazena informações sobre a quantidade de cestas disponíveis para serem doadas para as famílias aptas.

Entidade Cesta		
Atributo	Tipo	Descrição
identificação	número inteiro	Identificação do registro.
Data de geração	data	Data em que a cesta foi gerada.
Saída de saída	data	Dia em que a cesta foi doada para a família.

Quadro 25 – Atributos da Cesta

3.2.5.3.8 Entidade *Log*

Denominação: *Log*

Descrição: Armazena informações sobre movimentações ocorridas no *software*.

Entidade <i>Log</i>		
Atributo	Tipo	Descrição
identificação	número inteiro	Identificação do registro.
Data de operação	data	Data de cadastro ou alteração do registro.
Descrição	texto	Descreve a operação realizada.

Quadro 26 – Atributos de *log*

## 3.2.6 Melhoramentos Previstos

Durante conversas com o cliente, ficou decidido que vários melhoramentos incorporarão o projeto principal. A instituição, que estará recebendo a contribuição do *software* SGCE, possui mais alguns departamentos além do DAS, como: departamento financeiro, departamento de evangelização, biblioteca. O cliente gostaria que todos os departamentos da instituição fossem informatizados, mas devido ao tempo ser estreito para inclusão de todos, decidiu-se que seria conveniente informatizar no primeiro momento o Departamento de Assistência Social e, que após a conclusão e entrega do mesmo, estaremos dispostos a concretizar e informatizar os departamentos restantes.

### **3.3 Revisão de estimativas**

#### *3.3.1 Considerações preliminares*

No capítulo 2, a revisão de estimativa ocorre, pois a que foi descrita teve como base uma especificação preliminar de requisitos presentes no capítulo 1. A partir da especificação completa de requisitos, obtida neste capítulo, torna-se possível uma revisão das estimativas com mais precisão do que as anteriores.

#### *3.3.2 Estimativa de tamanho de Software*

Em relação à estimativa de tamanho, algumas alterações ocorreram, pois no capítulo 2 a adequação de funcionalidades não foram explicitadas inteiramente; com isso o cálculo do FPA foi alterado em relação à primeira estimativa.

#### *3.3.3 Estimativa de esforço*

Devido à alteração realizada nas funções referenciadas do FPA, fez-se necessário realizar um novo cálculo da estimativa de esforço para adequá-la ao cronograma previsto no capítulo 2.

Os itens alterados na estimativa de esforço foram dois: os pontos de função ajustados devido à alteração no total de pontos não ajustáveis, já que houve modificações nas funções referenciadas de FPA; e também o fator de esforço de trabalho, pois a princípio utilizamos um fator de produtividade baixo, não levando em consideração o treinamento especificado no cronograma (detalhado no capítulo 2).

#### *3.3.4 Estimativa de prazo*

Considerando que no capítulo 2 utilizamos pontos de função, ou

simplesmente FPA, para estatísticas de estimativa, alguns itens necessários para definição desses prazos ficaram incompletos devido à extração preliminar de requisitos com o cliente.

Esses itens foram incluídos no capítulo 2, no cálculo dos pontos de função, e esse novo cálculo foi executado visando à correção das estimativas previstas no capítulo anterior.

A seguir estão relacionadas as alterações feitas no cálculo de pontos de função.

Cabe ressaltar ainda que, devido à falta de experiência, por parte dos envolvidos no sistema, houve essa grande alteração no que tange às tabelas do banco de dados.

Revisão de estimativas e prazos		
Item	Incluído	Excluído
<b>Entrada Externa</b>	Incluir estoque	Excluir frequência família
	Alterar estoque	Incluir mantimento
	Excluir estoque	Alterar mantimento
	Incluir definição cesta	Excluir mantimento
	Alterar definição cesta	
	Excluir definição cesta	
	Incluir log operações	
<b>Saídas Externas</b>	Relatório log operações	
	Relatório famílias aptas	
	Relatório vencimentos estoque	
	Relatório voluntários	
<b>Arquivos Lógicos Internos</b>	Cadastro estoque	Cadastro mantimento
	Cadastro definição cesta	
	Cadastro log operações	
<b>Consultas Externas</b>	Consulta família	Consulta situação família
	Consulta definição cesta	Consulta volunt. colaboradores
	Consulta estoque	Consulta mantimentos
	Consulta log operações	
	Recuperar senha	

Quadro 27 – Alterações nos pontos de função

Pontos de Função Não Ajustados				
Tipo de Função	Complexidade Funcional	Quantidade	Total Complexidade	Total Tipo
Arquivo Lógico Interno	Simples	6	* 7 = 0	42
	Média	0	* 10 = 0	0
	Complexa	1	* 15 = 0	15
Interface Externa	Simples	0	* 5 = 0	0
	Média	0	* 7 = 0	0
	Complexa	0	* 10 = 0	0
Entrada Externa	Simples	12	* 3 =	36
	Média	4	* 4 =	16
	Complexa	2	* 6 =	12
Saída Externa	Simples	6	* 4 =	24
	Média	0	* 5 =	0
	Complexa	0	* 7 =	0
Consulta Externa	Simples	8	* 3 =	24
	Média	0	* 4 =	0
	Complexa	1	* 6 =	6
<b>Total Ponto de Função Não Ajustado</b>				<b>175</b>

Quadro 28 – Novo cálculo FPA não-ajustado

FPA AJUSTADO = 0,92 conforme item 2.5.4 do capítulo 2. O valor permanece inalterado pois não foi preciso alterar nada nesse ponto do cálculo.

O cálculo dos pontos de função ajustados é o produto do fator de ajuste e dos pontos de função brutos

$$FPA = FPNA * FVA$$

Onde FPA é ponto de função ajustado, FPNA é ponto de função não ajustado e FVA é fator de valor de ajuste.

O resultado geral será fornecido através do cálculo

$$FPA = 175 * 0,92$$

$$\mathbf{FPA = 161}$$

cujo resultado é 161 após arredondamento do produto realizado no passo anterior. Com isso o quadro abaixo explicará o esforço e o prazo.

<b>Estimativa de Esforço</b>	
Fator hh/pf (Utilização da Linguagem Orientada a Objetos (PHP5) e considerando a produtividade baixa).	7,5
Esforço total em hh(hora/homem) $7,5 * 161 =$	1207,5
(Considerando aproximadamente: 3,2 horas por dia, 4 dias por semana e 4,5 semanas por mês)	20,9635
Esforço total em hm (homem/mês): $(1207,5) / (57,6) =$	
<b>Estimativa de Prazo</b>	
Equipe composta por 2 integrantes: Rafael Malatesta e Tales Araújo Mendonça	2
Tamanho da Equipe = $((H / M) / \text{Prazo})$	
Prazo = $((H / M) / \text{Prazo}) = 20,9635 / 2 =$	10,4817 meses
Prazo em meses (considerado 22 dias por mês)	10 meses e 10 dias
hm: Homem-mês - hh: Homem-hora - pf: Ponto-de-função	

Quadro 29 – Nova estimativa de esforço

		Nome	Duração	Início	Término
1		<b>Levantamento de Requisitos</b>	<b>3 dias</b>	<b>12/02/11</b>	<b>16/02/11</b>
2		Reunião com cliente	1 dia	12/02/11	14/02/11
3		Levantamento preliminar de requisitos	1 dia	14/02/11	14/02/11
4		Especificação de requisitos	2 dias	15/02/11	16/02/11
5		<b>Planejamento do Projeto</b>	<b>19 dias</b>	<b>17/02/11</b>	<b>15/03/11</b>
6		Escopo	1 dia	17/02/11	17/02/11
7		Plano Processo de desenvolvimento	1 dia	18/02/11	18/02/11
8		Metodologia de desenvolvimento	1 dia	18/02/11	18/02/11
9		Estrutura analítica	1 dia	21/02/11	21/02/11
10		Estimativas	2 dias	22/02/11	23/02/11
11		Plano Organização	1 dia	24/02/11	24/02/11
12		Monitoramento e Controle	1 dia	25/02/11	25/02/11
13		Cronograma	3 dias	28/02/11	02/03/11
14		Recursos Humanos	1 dia	03/03/11	03/03/11
15		Recursos Gerais	1 dia	04/03/11	04/03/11
16		Plano de Custo	2 dias	07/03/11	08/03/11
17		Plano de Teste	1 dia	09/03/11	09/03/11
18		Plano de Treinamento	2 dias	10/03/11	11/03/11
19		Plano de Implantação	2 dias	14/03/11	15/03/11
20		<b>Análise e modelagem</b>	<b>58 dias</b>	<b>16/03/11</b>	<b>03/06/11</b>
21		Especificação de requisitos	28 dias	16/03/11	22/04/11
22		Modelagem do Bando de Dados	30 dias	23/04/11	03/06/11
23		<b>Implementação e testes</b>	<b>128 dias</b>	<b>06/06/11</b>	<b>30/11/11</b>
24		Desenvolvimento do sistema	128 dias	06/06/11	30/11/11
25		Testes	63 dias	16/06/11	13/09/11
26		<b>Implantação</b>	<b>14 dias</b>	<b>01/12/11</b>	<b>20/12/11</b>
27		Plano de implantação	11 dias	01/12/11	15/12/11
28		Plano de treinamento	3 dias	16/12/11	20/12/11

Figura 16 – Novo cronograma de atividades

### 3.3.5 Cronograma revisado

As alterações referentes ao cronograma foram realizadas de acordo com o acompanhamento cronológico de cada atividade realizada, bem como a sua adequação, cronológica, com a estimativa de tamanho utilizada (FPA).

Com base nas tarefas na Figura 17, foi elaborado o Gráfico de Gantt, ilustrado na Figura 18.

Conforme descrito anteriormente, o gráfico de Gantt possui a finalidade de demonstrar cada ação que será desempenhada, bem como explicitar a interação entre as fases do projeto.



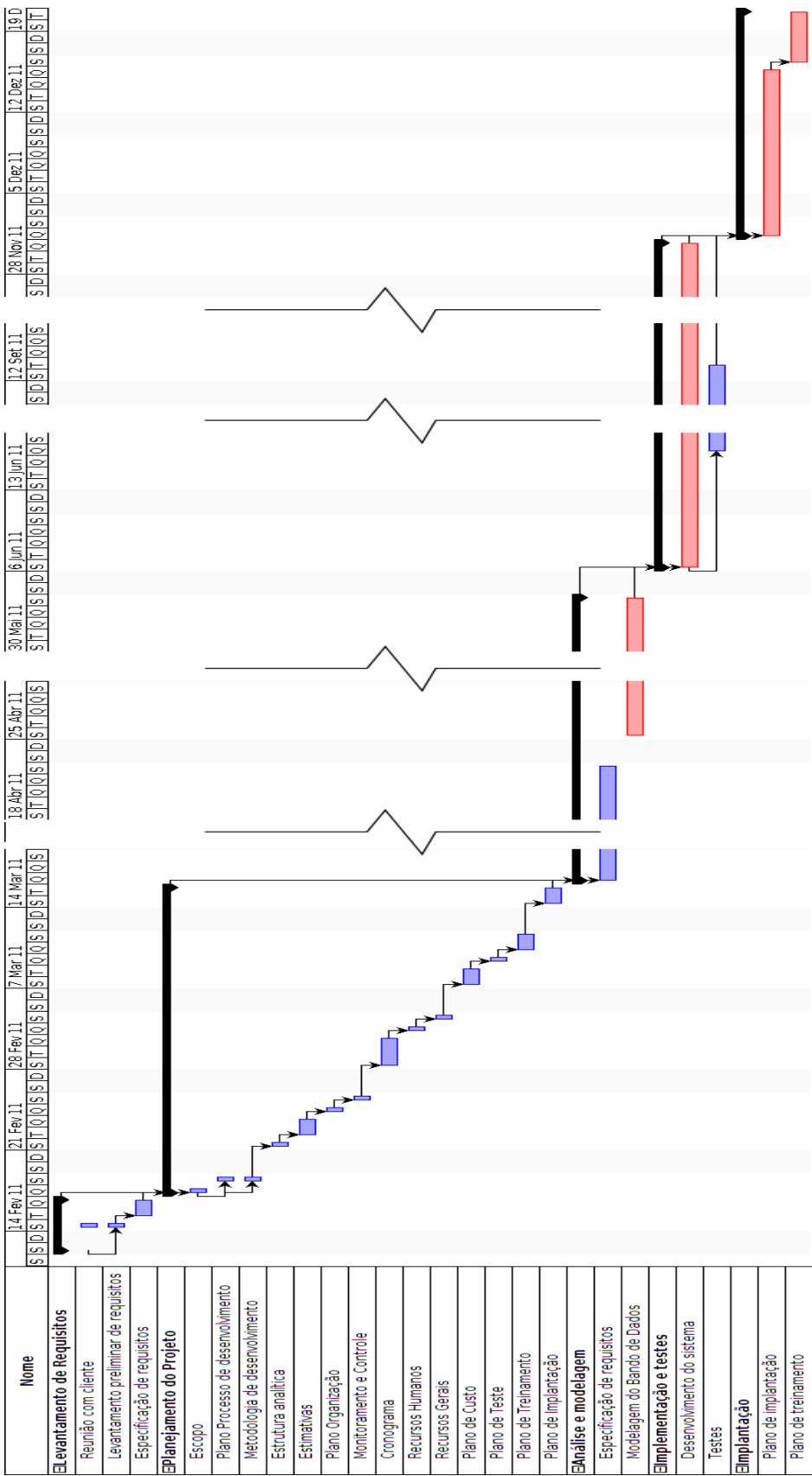


Figura 17 – Novo Gráfico de Gantt

### *3.3.6 Revisão dos custos estimados*

A estimativa de custos descrita no capítulo 2 foi baseada na especificação preliminar de requisitos.

Após a conclusão da fase de requisitos neste capítulo, foi observado que a estimativa de custos descrita no item 2.11 não se alterou, tendo em vista a pouca variação no cronograma não havendo nada que alterasse os custos do projeto que foram calculados com margem de segurança.

### *3.3.7 Considerações finais sobre a revisão de estimativas*

Após uma detalhada revisão das estimativas realizada neste capítulo, pode-se obter uma visão mais detalhada e realista do projeto. Essa visão, agora melhor, ocorreu devido as novas estimativas obtidas, essas, serem baseadas no levantamento de requisitos, mais completos, e não referente aos requisitos preliminares compostos no capítulo 1.

Partindo agora de uma nova estimativa e mais adequada, real, ao projeto, a equipe de desenvolvimento irá trabalhar com um cronograma mais adequado, agora estabelecido, visando cumprir os prazos estabelecidos, e a estabilidade do projeto no decorrer do seu ciclo de vida.

## **4. MODELAGEM DE ANÁLISE**

### **4.1 Considerações preliminares**

Programas desenvolvidos por empresas sérias que presam pela qualidade do software e de como mantê-lo atualizado, seguro e robusto, necessitam de uma boa documentação do projeto para que uma ponderação ocorra e assim, de maneira mais simples e organizada, consiga tirar proveito das documentações que consiste o

projeto. Dessa forma, a modelagem de análise busca objetivar o software de modo que esse esteja apto a ser desenvolvido posteriormente.

As definições ocorridas durante a modelagem e análise compreende os modelos necessários, como um todo, para que o projeto possa ser realizado. Engenheiros de software fazem uso de descrições textuais e gráficas para analisar e explicar a ideia em formação (PRESSMAN, 2006).

Os diagramas que modelam o sistema são representados neste momento, especificando o que precisa ser realizado e não como deve ser realizado. As representações gráficas tornam-se mais claras de serem visualizadas pela equipe responsável.

## **4.2 Metodologia adotada**

A linguagem de modelagem unificada ou *Unified Modeling Language* (UML) é uma linguagem – não proprietária de terceira geração – visual utilizada para criar modelos de programas, a qual foi adotada para utilização deste projeto, auxiliando na visualização do desenho e na comunicação entre objetos.

Devido à escolha de criar um projeto utilizando uma modelagem orientada a objetos, optou-se por utilizar a UML, que também é utilizada em várias modalidades de análise. A UML é a notação diagramática padrão para desenhar ou apresentar figuras – com algum texto – relacionadas a um software (LARMAN, 2007).

A utilização da UML promove a equipe um grau de abstração maior do que será desenvolvido, mostrando uma visão mais próxima do real, à nível conceitual, promovendo, assim, um melhor entendimento do que o software se propõe a realizar.

Através de ferramentas cases para a criação de modelos UML, consegue-se identificar uma melhor percepção do software que será criado, deste modo, fica mais claro encontrar divergências entre os requisitos e os modelos criados.

## **4.3 Diagrama de caso de uso**

O diagrama de caso de uso possui a finalidade de representar as interações dos atores com as funcionalidades ligadas ao sistema. Esse, por sua vez, apresenta um alto nível de abstração dentre os diagramas da UML. Através dele observa-se, com um alto nível de clareza, o que cada ator pode fazer no sistema, a explicar os requisitos capturados no início do trabalho.

De acordo com Tonsig (2003), o diagrama de caso de uso é utilizado para descrever o que esse novo sistema deverá realizar ou para um já existente, demonstrando o que poderá ocorrer em várias situações no decorrer de sua execução. Afirma também, Tonsig, qual tal diagrama deve prever todas as operações que o sistema tornar-se-á disponíveis.

Os diagramas de caso de uso foram modelados e apresentados no capítulo 3, no item 3.2.4.3.1.

## **4.4 Diagrama de classe**

### **4.4.1 Classe**

Uma classe é composta por uma estrutura que são representados por um conjunto de objetos, possuindo assim, características similares. Esses objetos serão criados, instanciados, a partir das classes para serem utilizados no sistema. Dessa forma, uma classe define o comportamento dos seus objetos através de métodos e os estados que estes podem assumir através de atributos.

A classe possui vários elementos que a diferencia das demais, como o seu nome, os atributos – definem as propriedades das classes –, e métodos – representam as funcionalidades que podem ser realizadas pelo objeto –. É demonstrado a seguir, um modelo de classe de acordo com a UML:

NomeDaClasse
-atributo a: tipo -atributo b: tipo -atributo c: tipo
+metodo x(): tipoRetorno +metodo y(): tipoRetorno +metodo z(): tipoRetorno

Quadro 30 – Modelo de Classe UML

#### 4.4.2 Diagrama de classes

O diagrama de classe consiste em representar as associações, existentes, entre classes e as interfaces que serão criadas para o projeto, exemplificando quais serão os relacionamentos entre os objetos instanciados desta classe, que será descrita graficamente.

Segundo Larman (2007), as associações representam uma interação que ocorre entre uma classe e outra, devendo esta ser armazenada na memória, independente do tempo que seja gasto.

Vários tipos de classes são comuns em muitos projetos. Percebendo isso, alguns frameworks possuem um conjunto de componentes com funcionalidades genéricas pré-implementadas em uma linguagem específica. Esses componentes são a instância de um objeto que representa, na prática, esse conjunto de classes, com regras de negócio comuns e acesso a banco de dados. Dessa maneira o desenvolvedor precisa apenas saber o que o componente faz e inseri-lo no projeto, de acordo com as especificações do framework utilizado. Com isso o desenvolvedor ficará isento de implementar classes com funcionalidades comuns, voltando sua atenção para a configuração dos componentes, para que assim consiga obter uma funcionalidade específica.

Apesar desse diagrama ter sido elaborado, com o intuito mais elevado de abstração das classes que o sistema terá, o projeto implementará de acordo com as especificações do framework (CakePHP) utilizado, seguindo suas métricas e padrões sugeridos.

Com objetivo de clareza e não poluição visual do seguinte documento, optou-se em não implementar os métodos das classes existentes no diagrama de classes. A implementação dos métodos pode ser visualizada no diagrama de sequência no item 4.4.

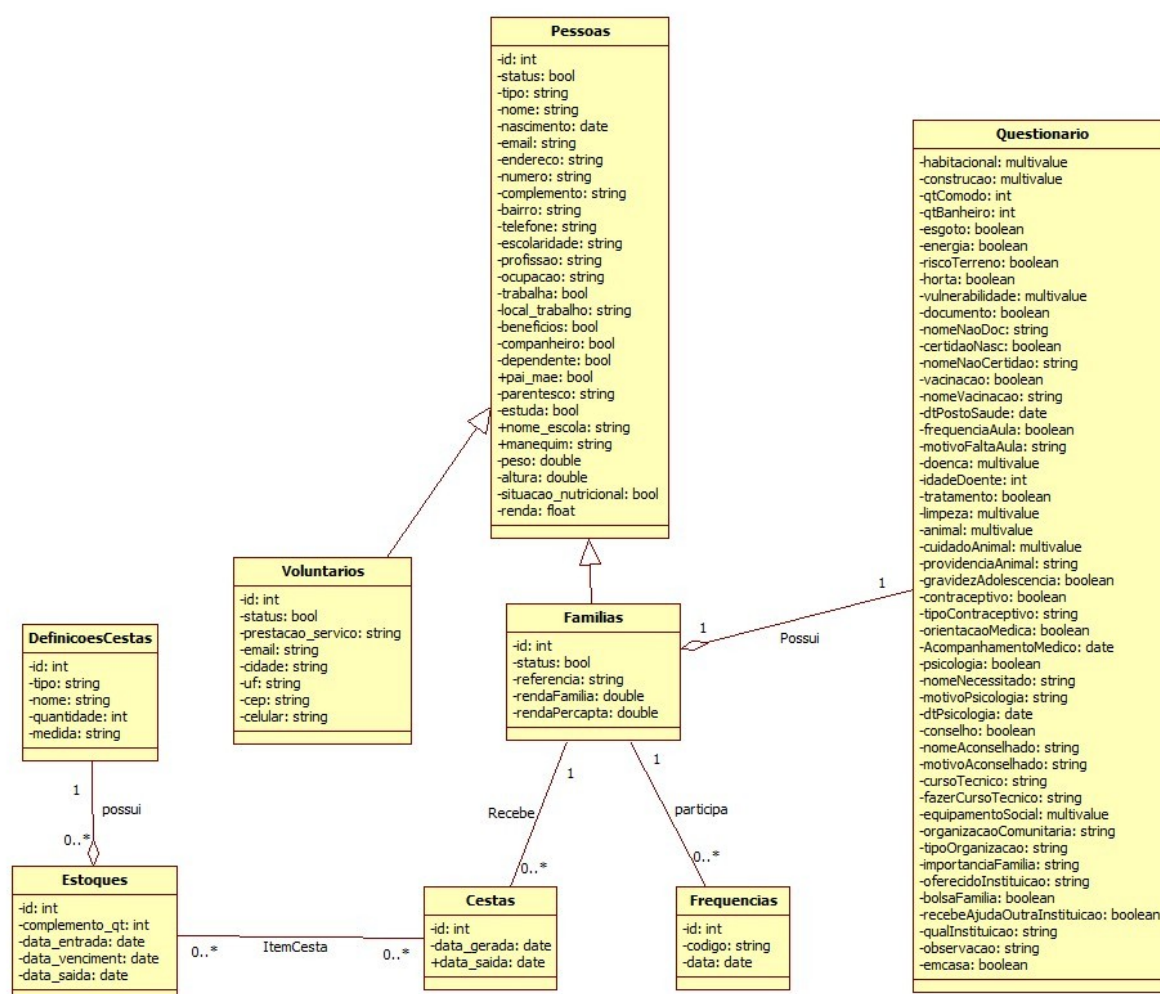


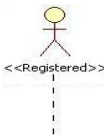
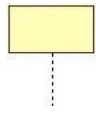
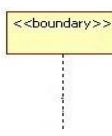
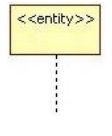
Figura 18 – Diagrama de Classe

#### 4.4 Diagrama de sequência

Devendo esse, diagrama de sequência, ser o mais importante dos diagramas – comunicação, visão e temporal – de interação. Possui a finalidade de demonstrar como ocorre a troca de mensagens dentro de uma linha sequencial de uma determinada operação a ser executada pelo sistema. Descreve como será realizada a operação de um caso de uso, bem como a sequência de passos que o sistema e o usuário irá interagir-se para realizar uma determinada operação, tudo isso de forma gráfica.

Larman (2007), afirma que diagramas de sequência são artefatos criados de forma rápida e fácil, a ilustrar os eventos de entrada e saída relacionadas ao sistema.

Os diagramas a seguir de acordo com a tabela abaixo, representam um cenário detalhado, das especificações de caso de uso, a demonstrar sua ocorrência no fluxo principal.

Simbolo	Nome do Simbolo	Descrição
	Actor (Ator)	Mostra qual ator está atuando em cada diagrama.
	Object (Objeto)	Mostra os objetos do sistema assim como suas interações com os outros objetos do sistema.
	Boundary (Fronteira)	Mostra a interação do objeto com o ator ou outros objetos do sistema.
	Entity (Entidade)	Mostra as entidades do banco de dados onde os objetos se interagem.

Quadro 31 – Simbologia de explicação do diagrama de sequencia



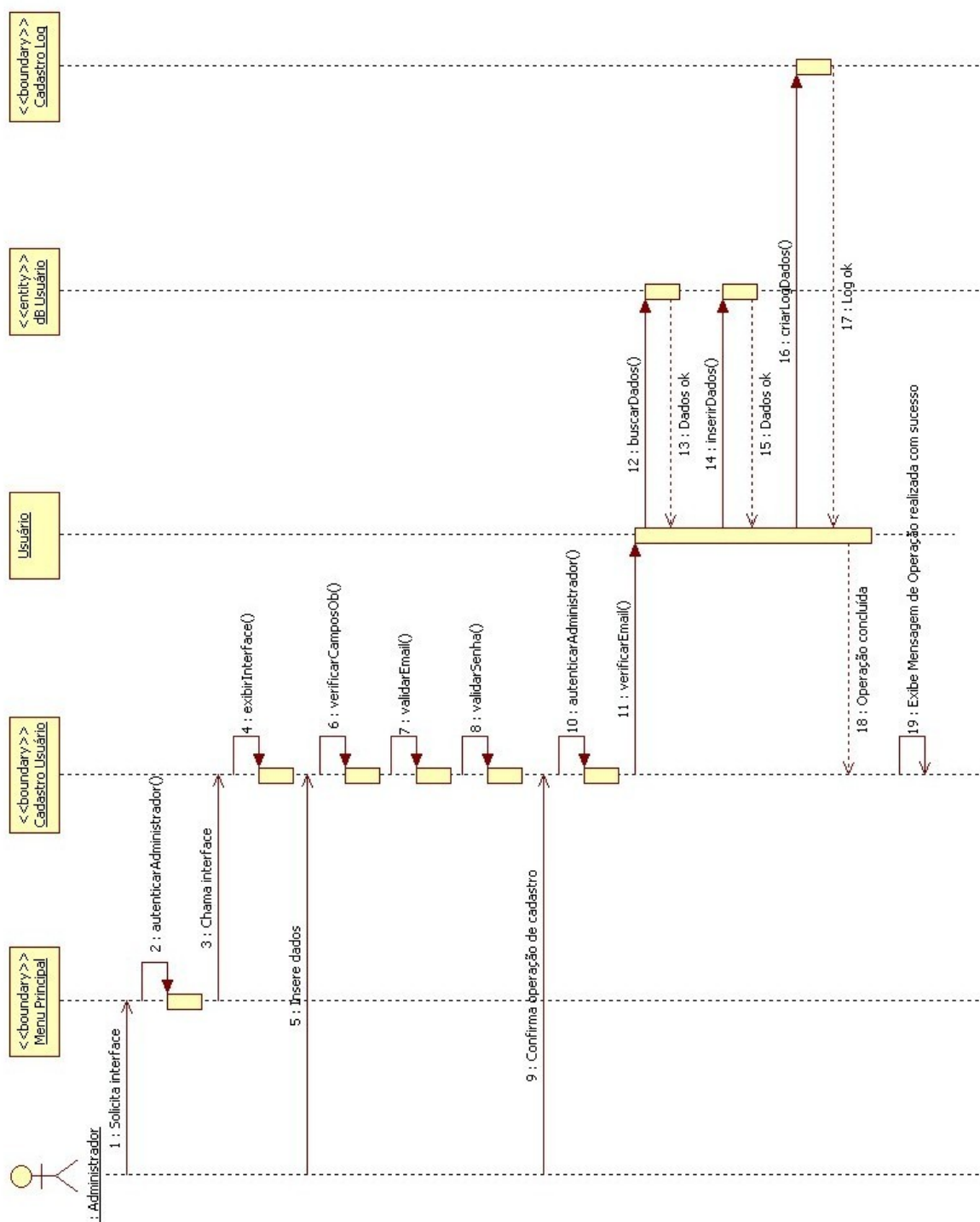


Figura 19 – Diagrama de Sequencia: Cadastrar Usuário

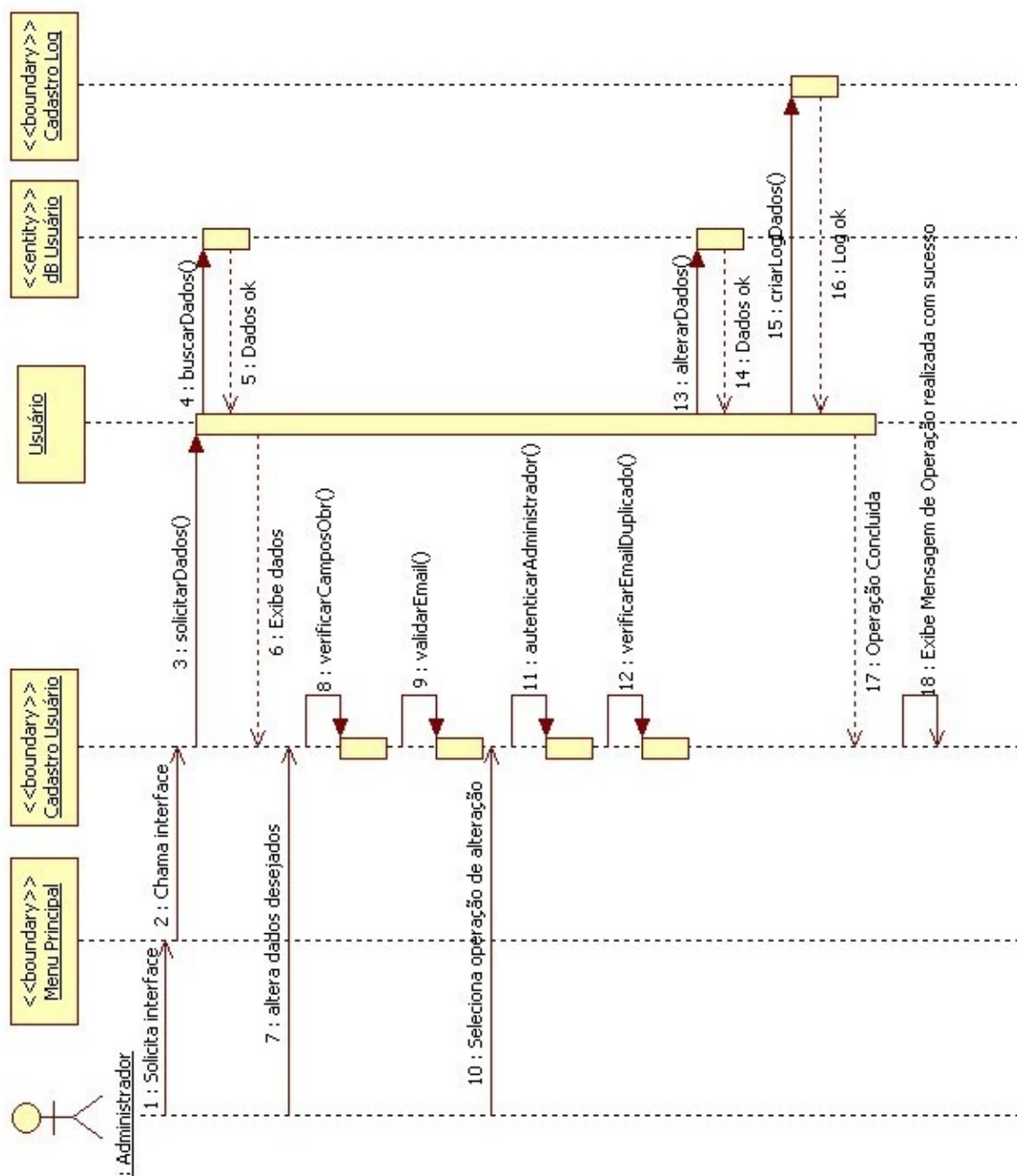


Figura 20 – Diagrama de Sequencia: Alterar Usuário

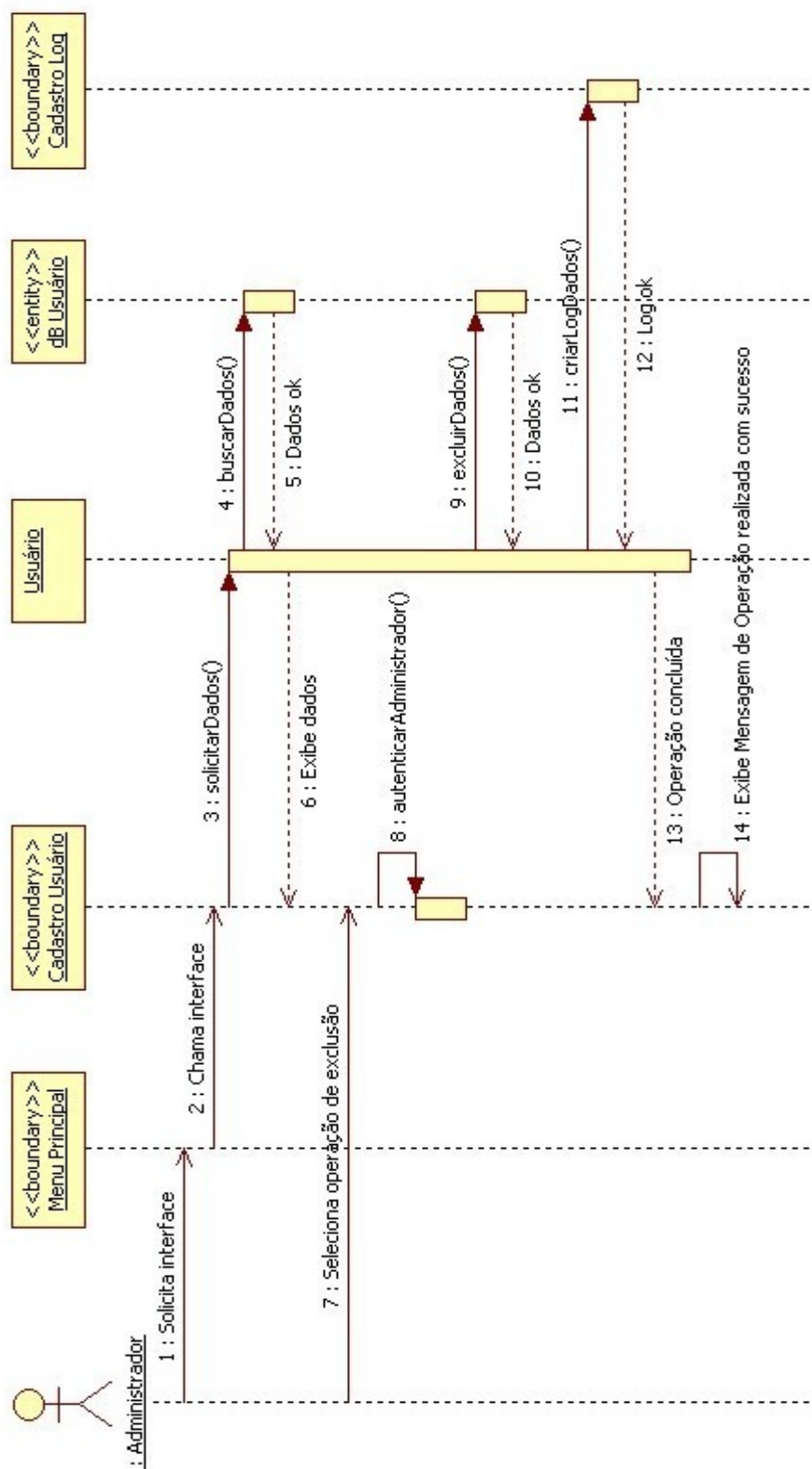


Figura 21 – Diagrama de Sequencia: Excluir Usuário

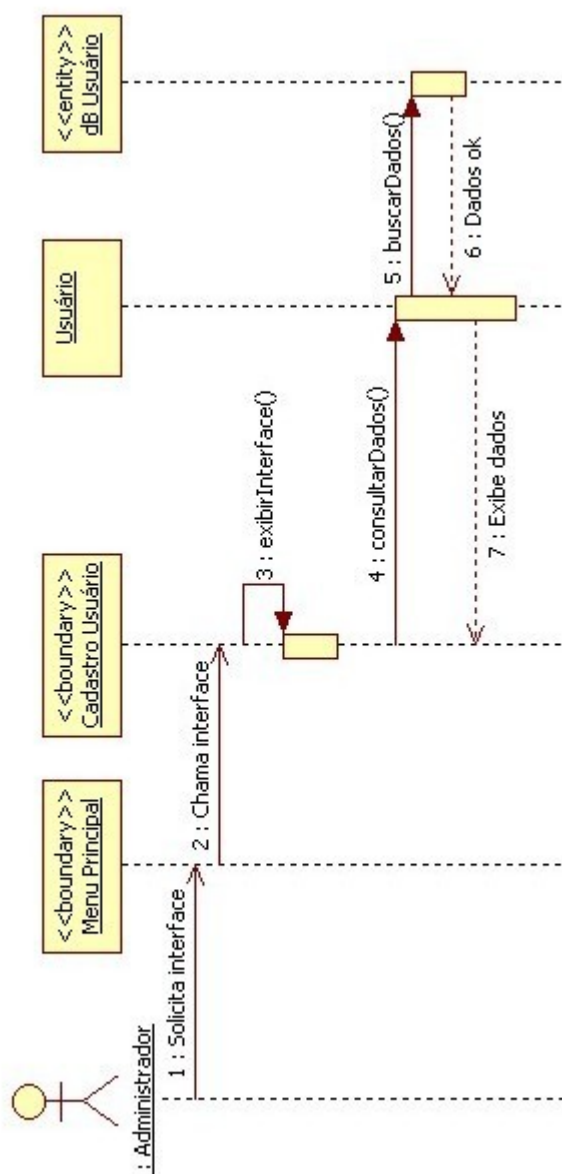


Figura 22 – Diagrama de Sequencia: Consultar Usuário

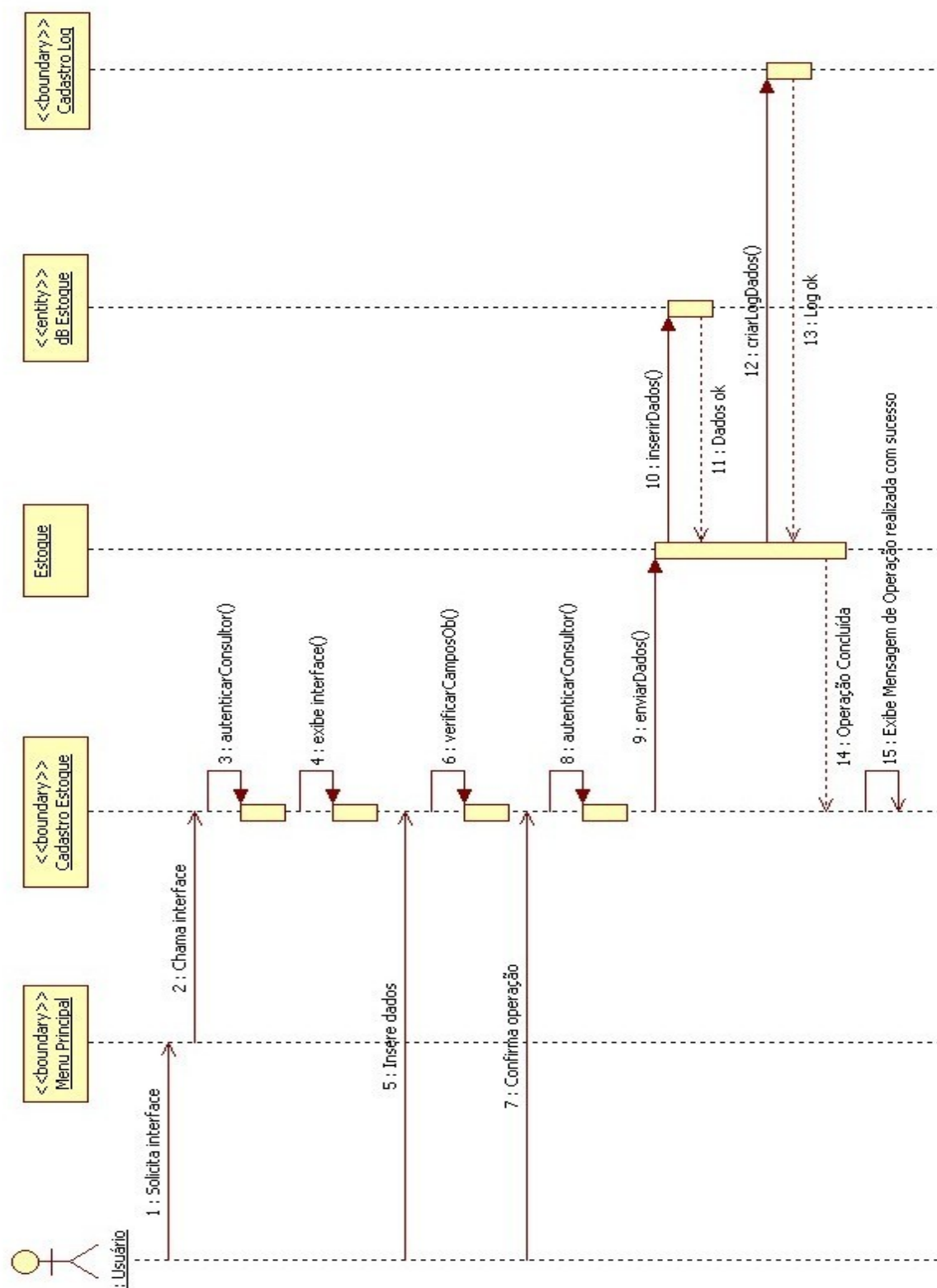


Figura 23 – Diagrama de Sequencia: Incluir Estoque

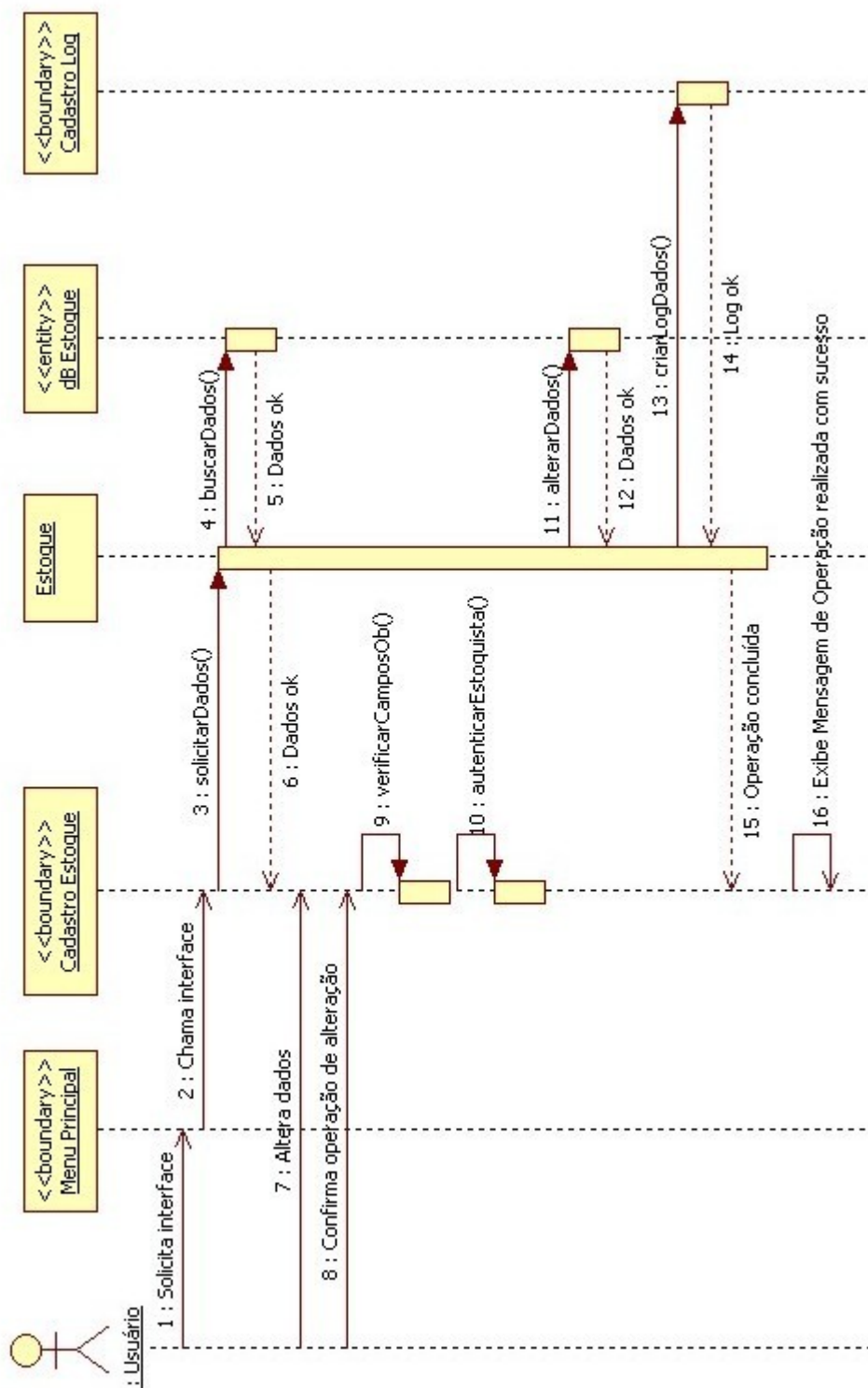


Figura 24 – Diagrama de Sequencia: Alterar Estoque

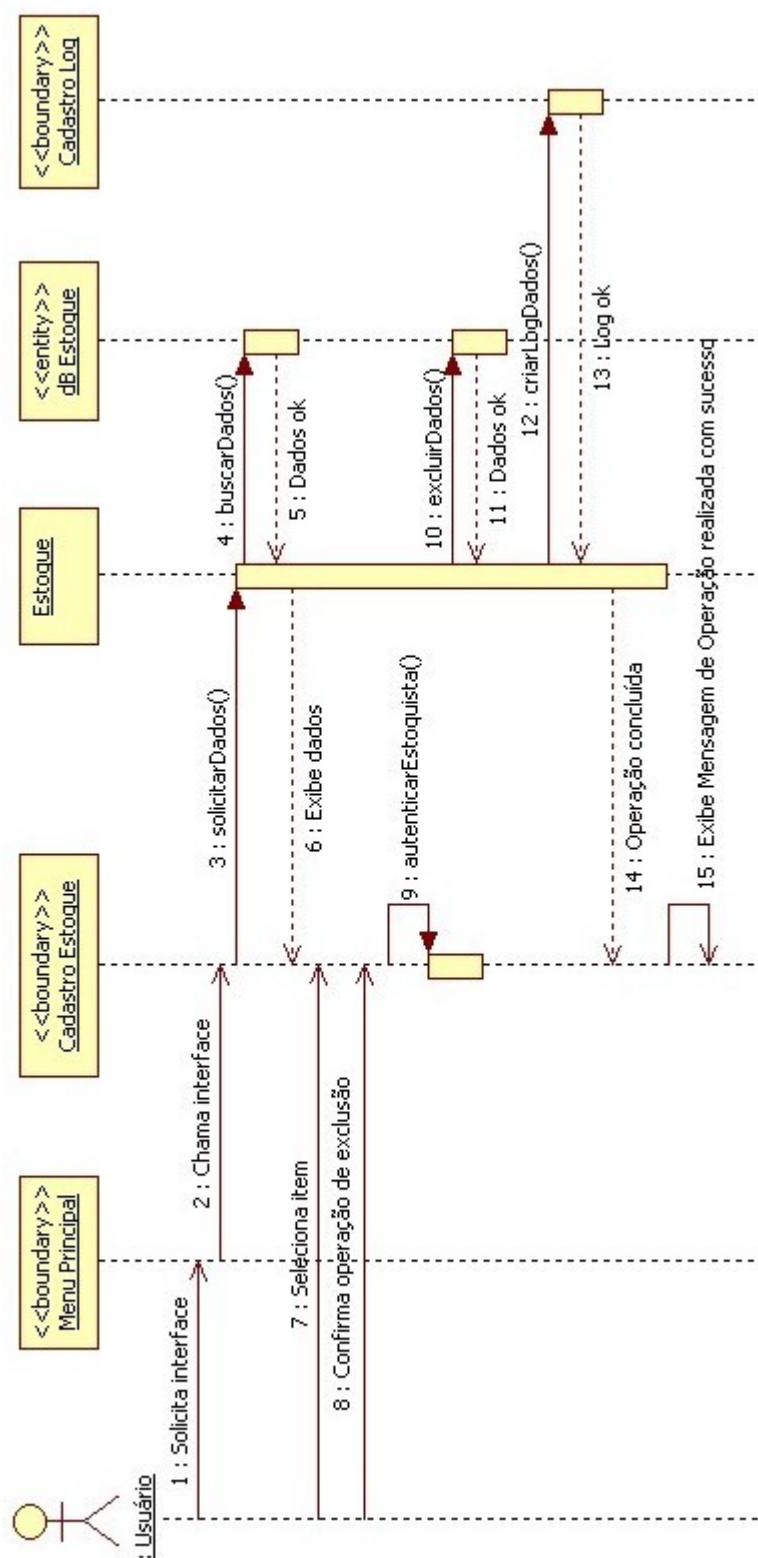


Figura 25 – Diagrama de Sequencia: Excluir Estoque

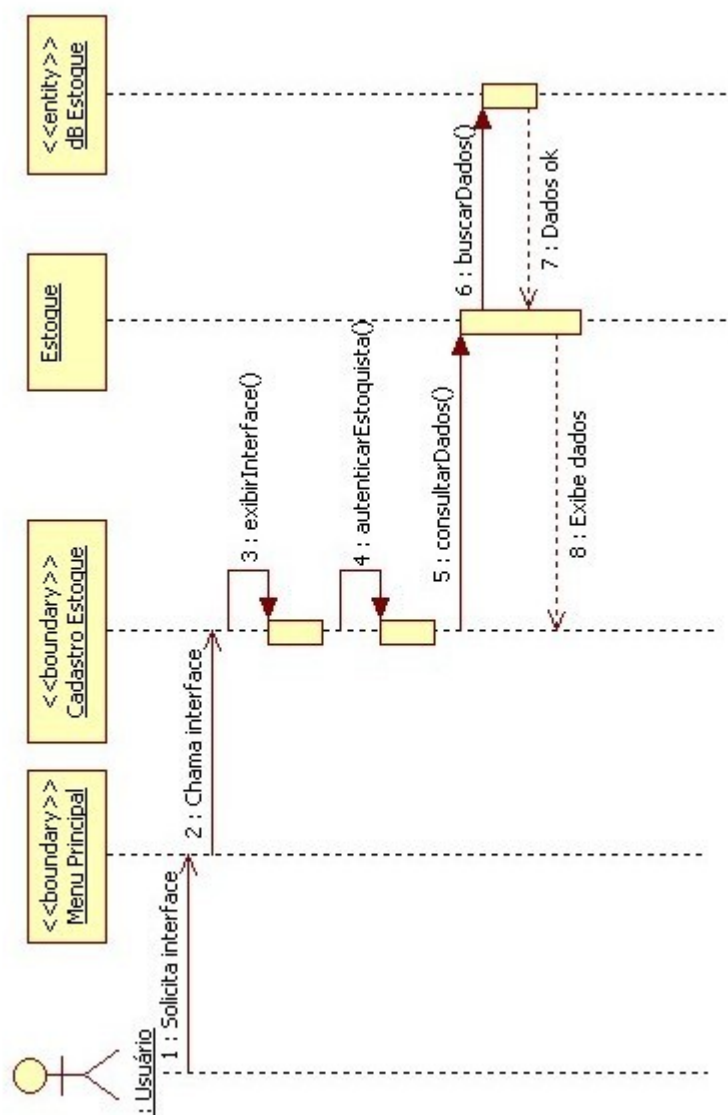


Figura 26 – Diagrama de Sequencia: Consultar Estoque



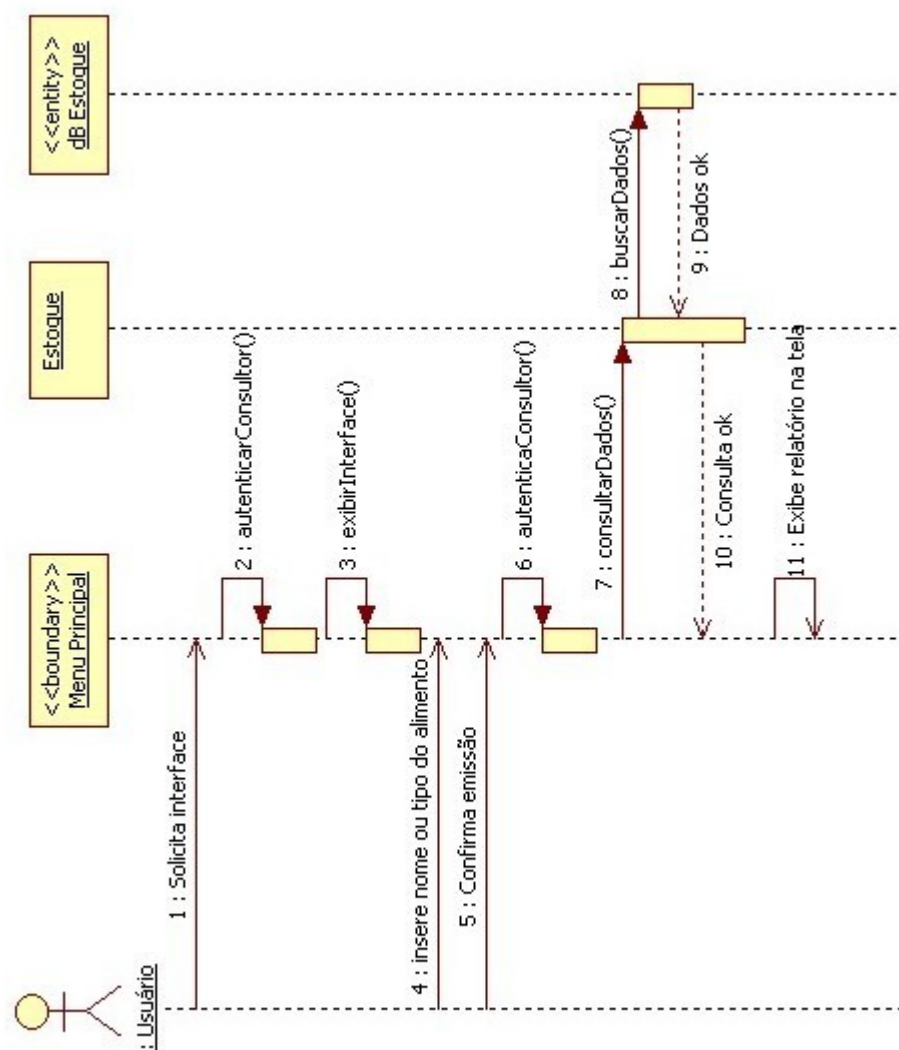


Figura 27 – Diagrama de Sequencia: Relatório de Vencimentos Estoque

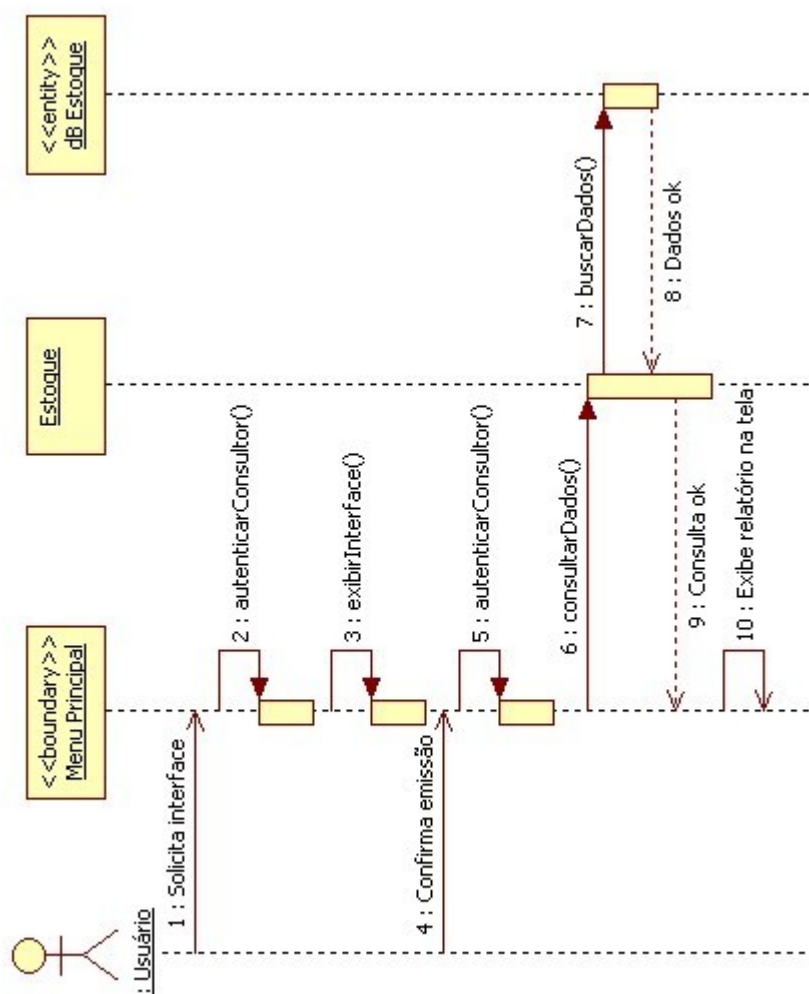


Figura 28 – Diagrama de Sequencia: Relatório de Cestas Disponíveis

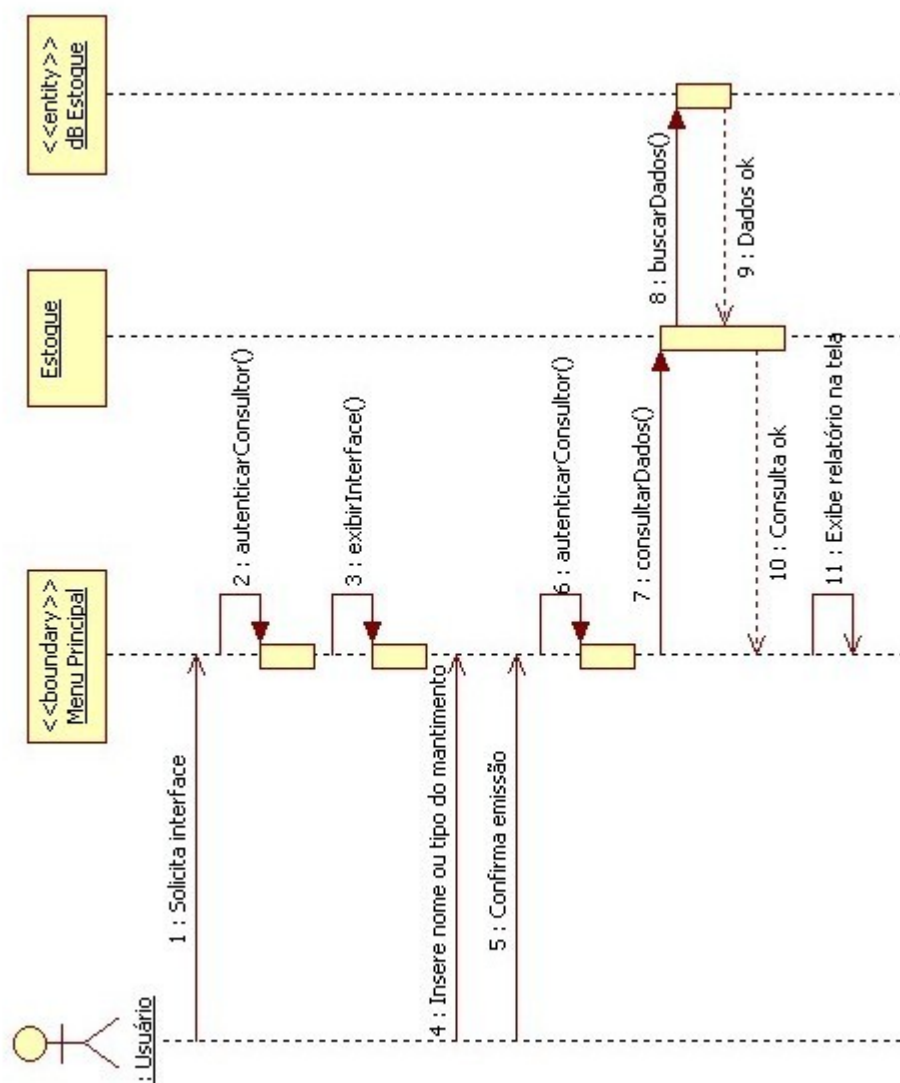


Figura 29 – Diagrama de Sequencia: Relatório de Itens Pendentes na Cesta

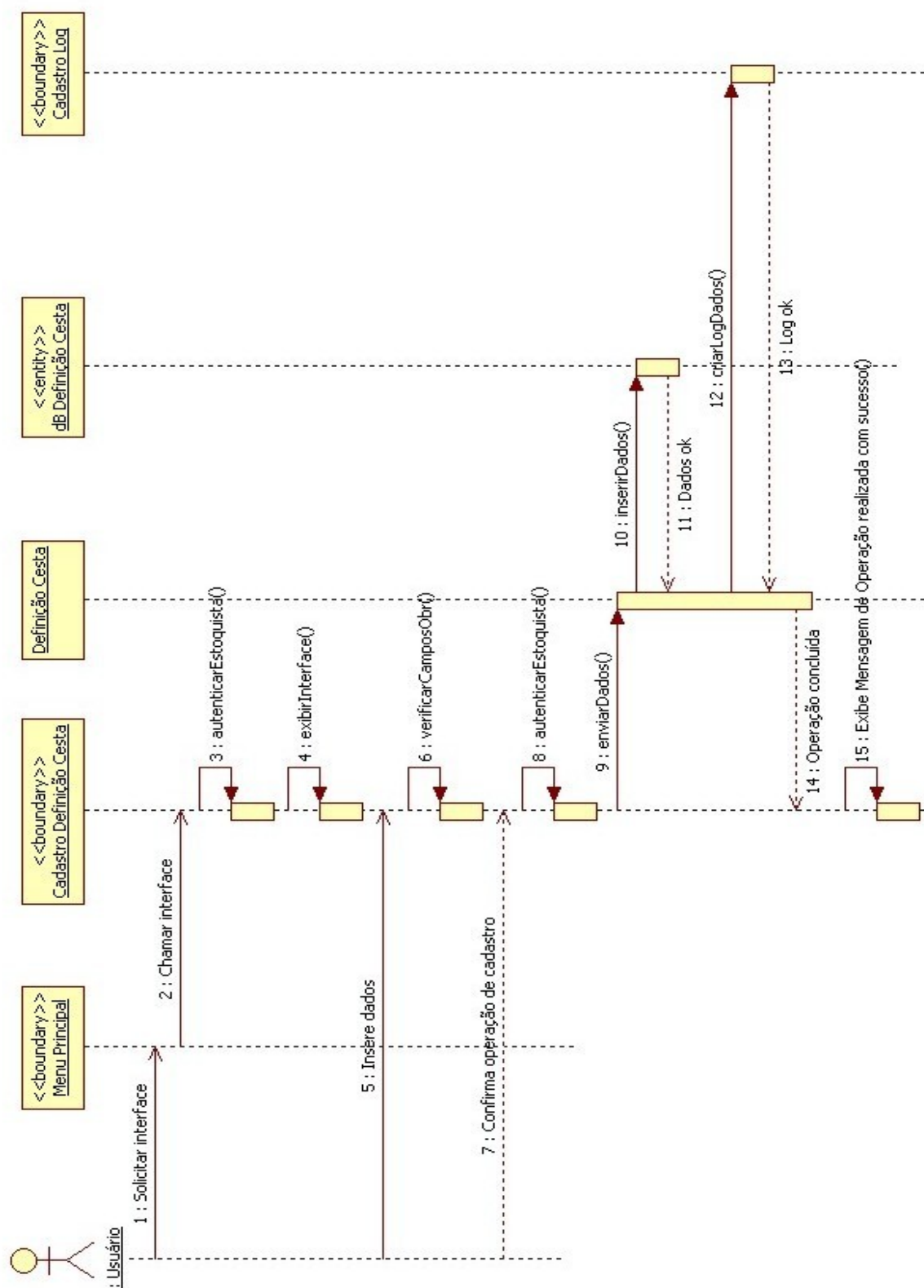


Figura 30 – Diagrama de Sequencia: Cadastro Definição Cesta

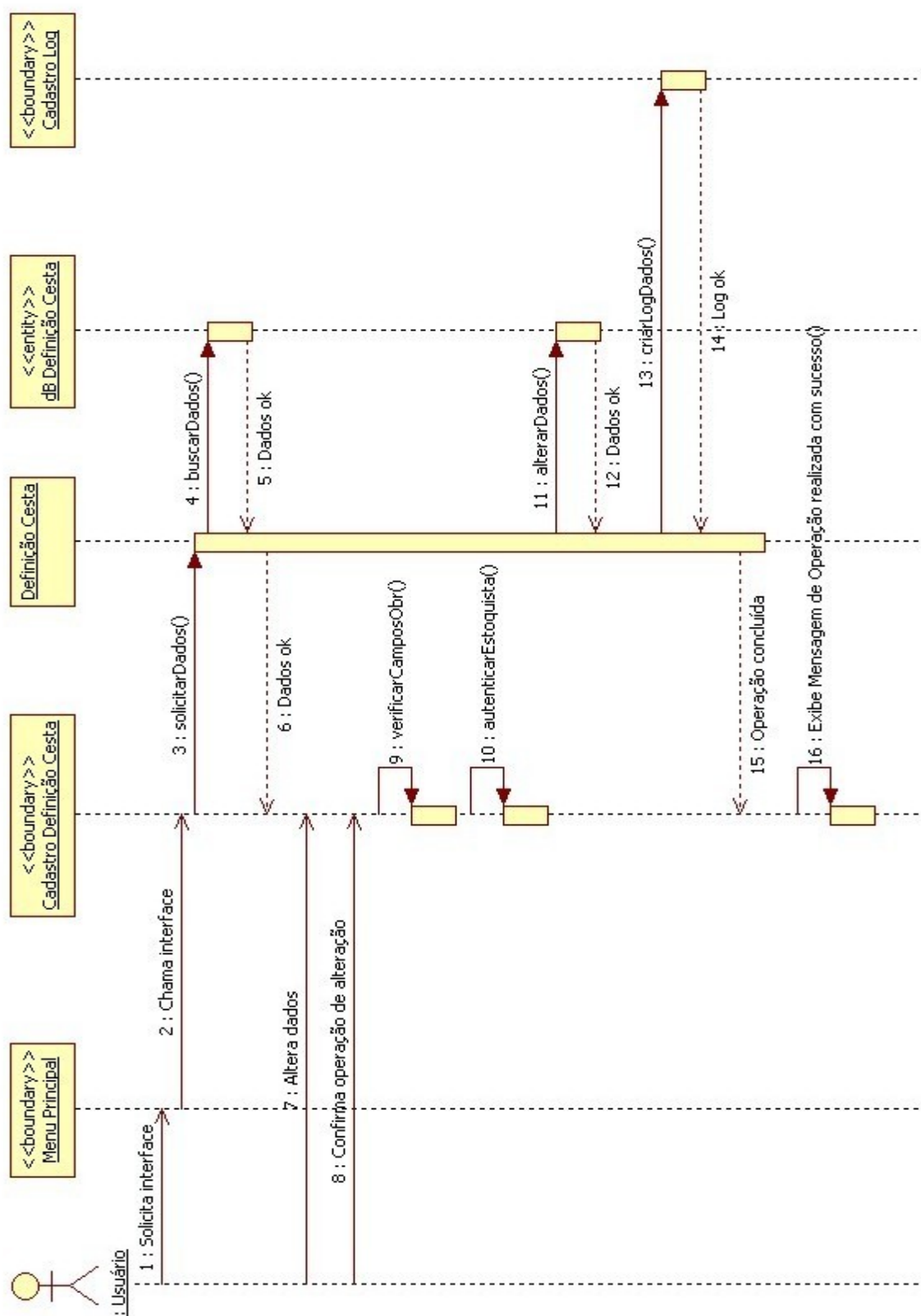


Figura 31 – Diagrama de Sequencia: Alterar Definição Cesta

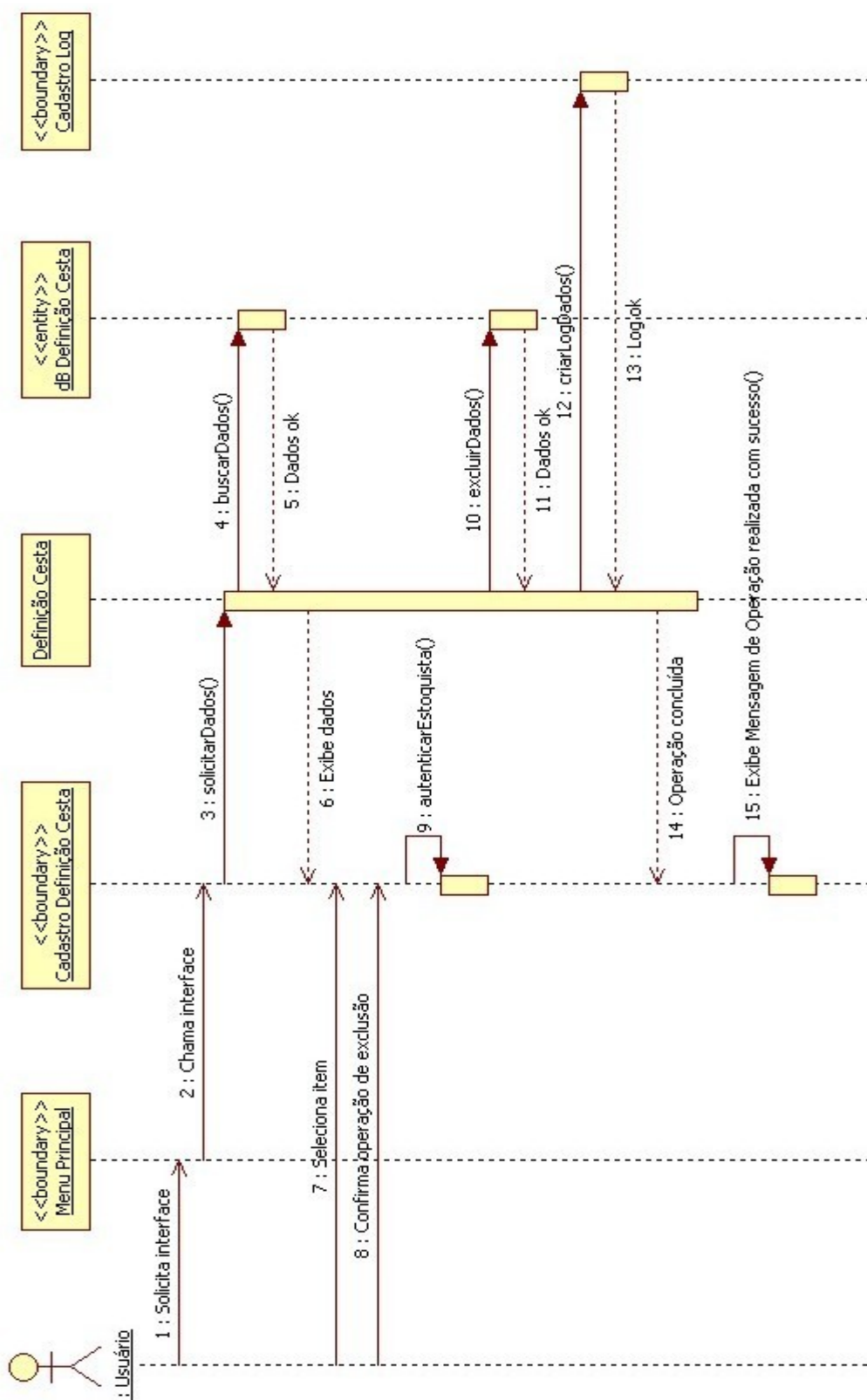


Figura 32 – Diagrama de Sequencia: Excluir Definição Cesta

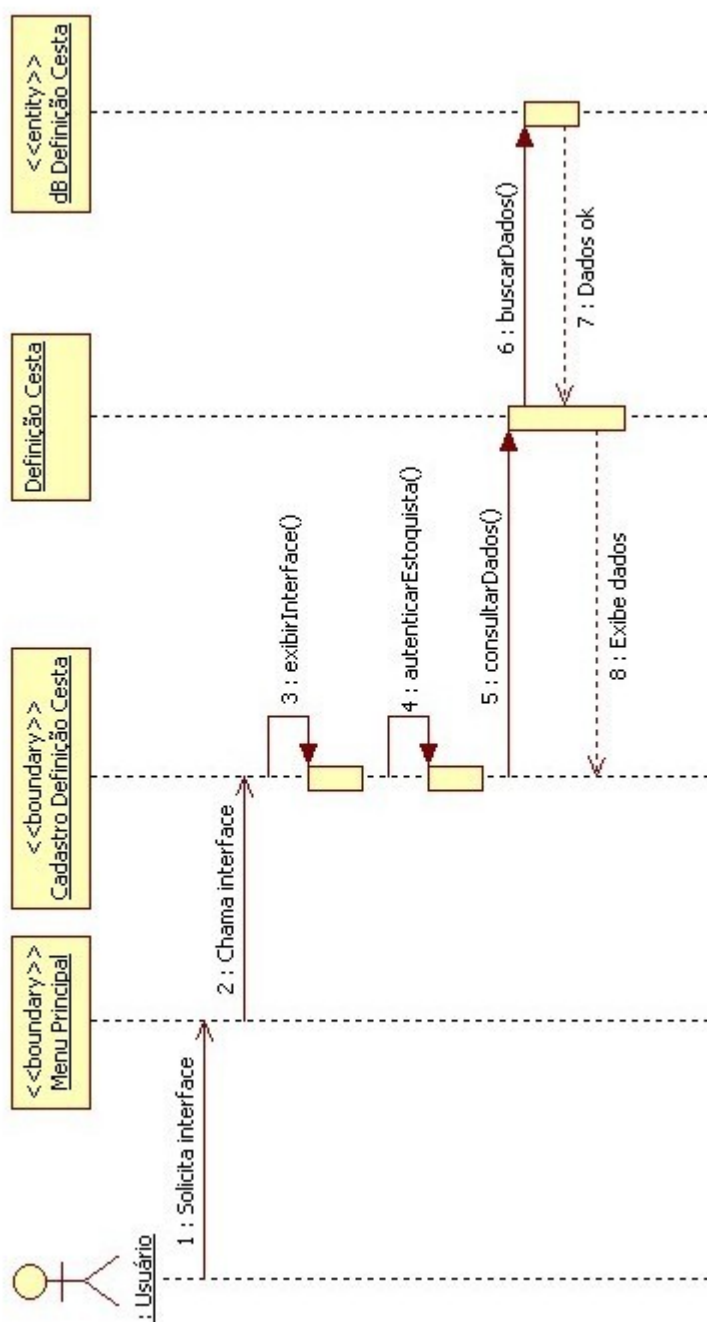


Figura 33 – Diagrama de Sequencia: Consultar Definição Cesta

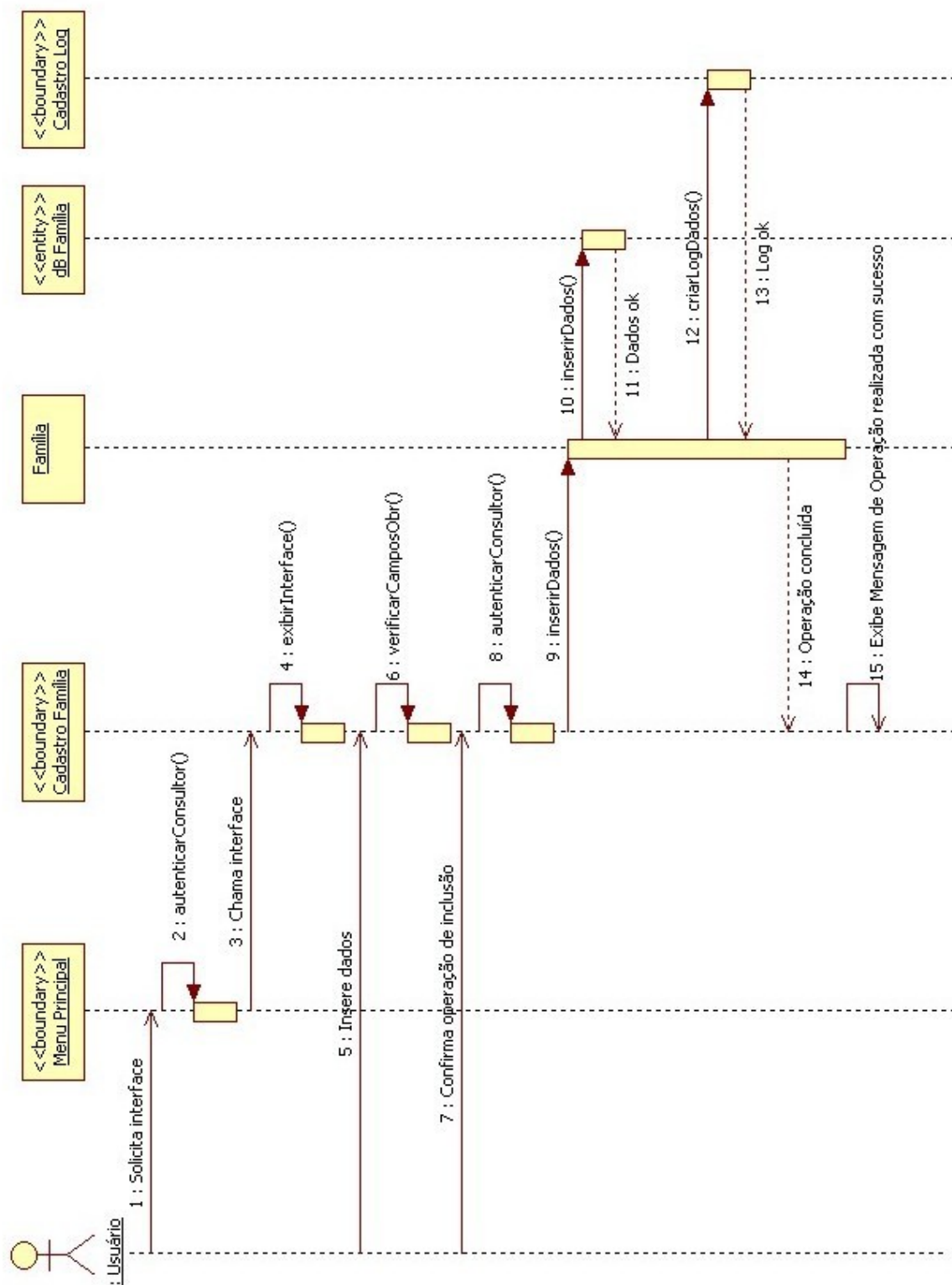


Figura 34 – Diagrama de Sequencia: Cadastrar Família



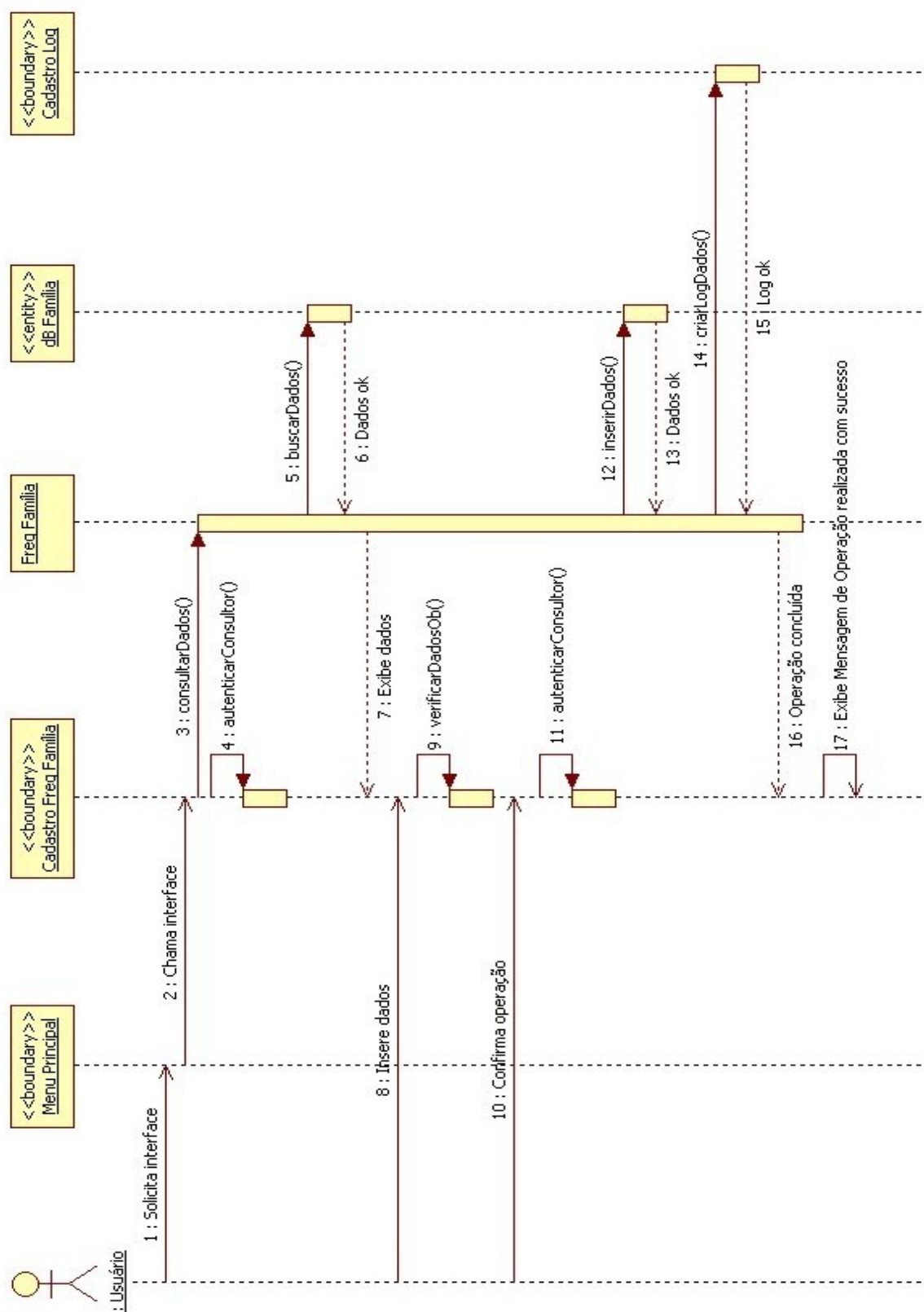


Figura 35 – Diagrama de Sequencia: Cadastrar Frequência de Família

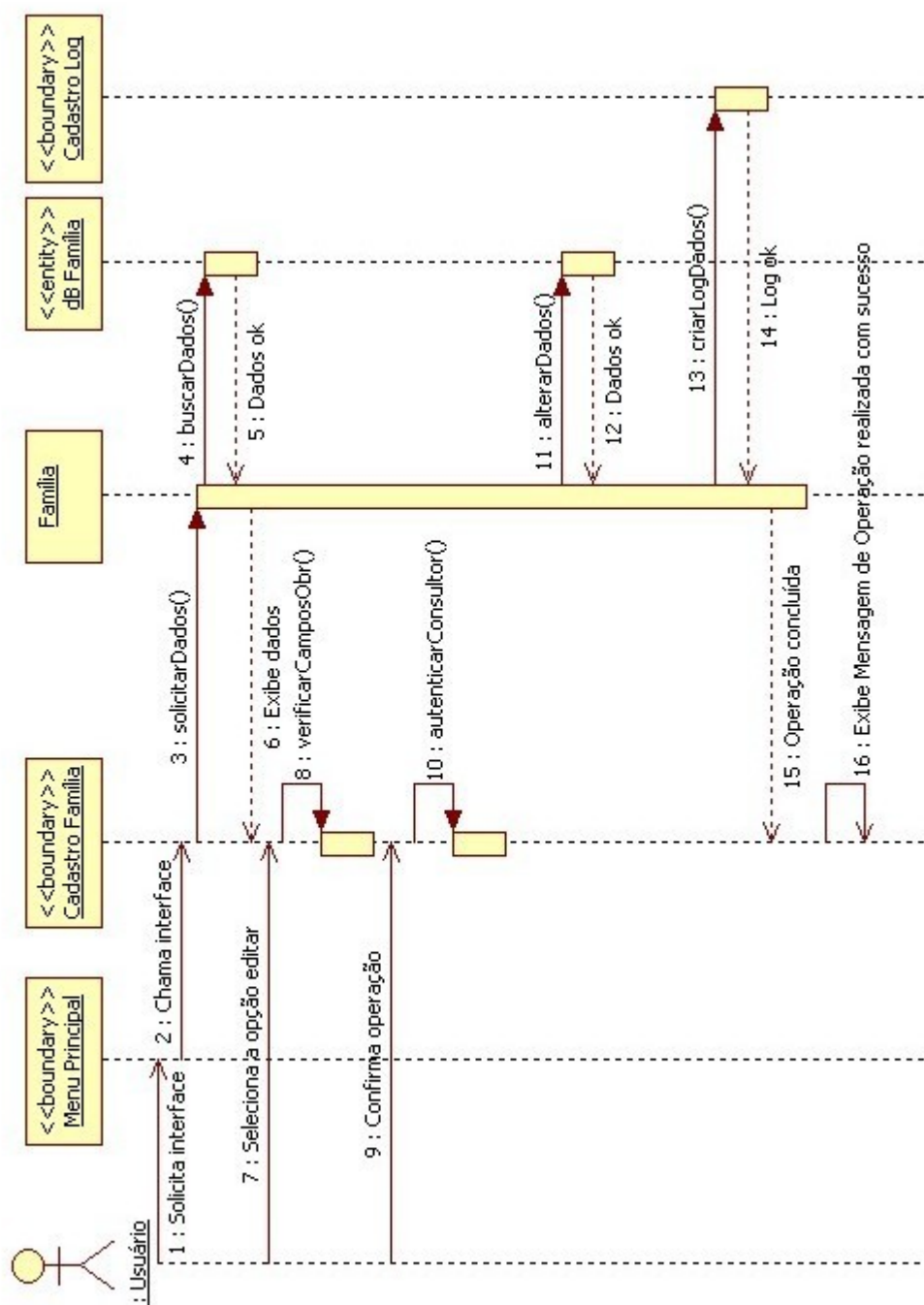


Figura 36 – Diagrama de Sequencia: Alterar Família

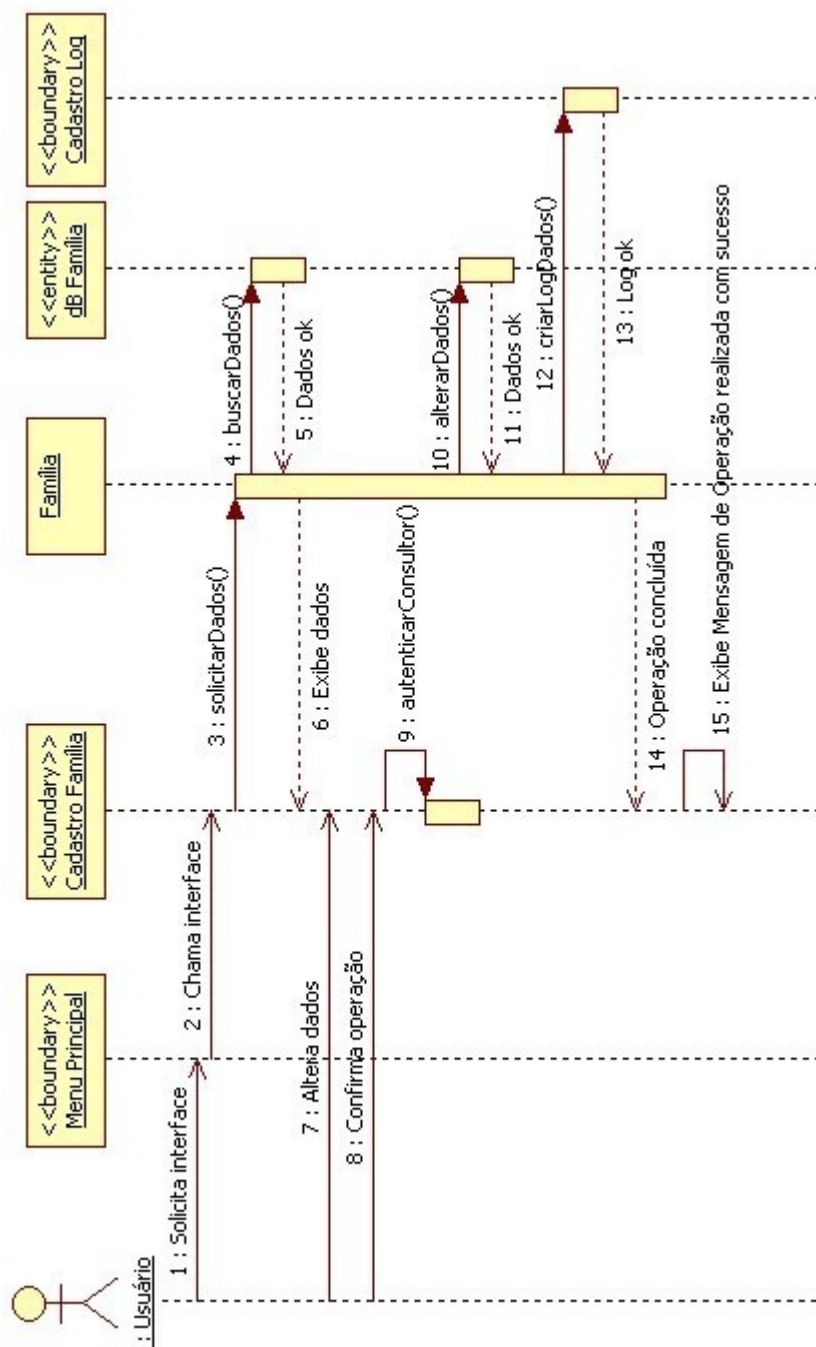


Figura 37 – Diagrama de Sequencia: Alterar Frequência Família

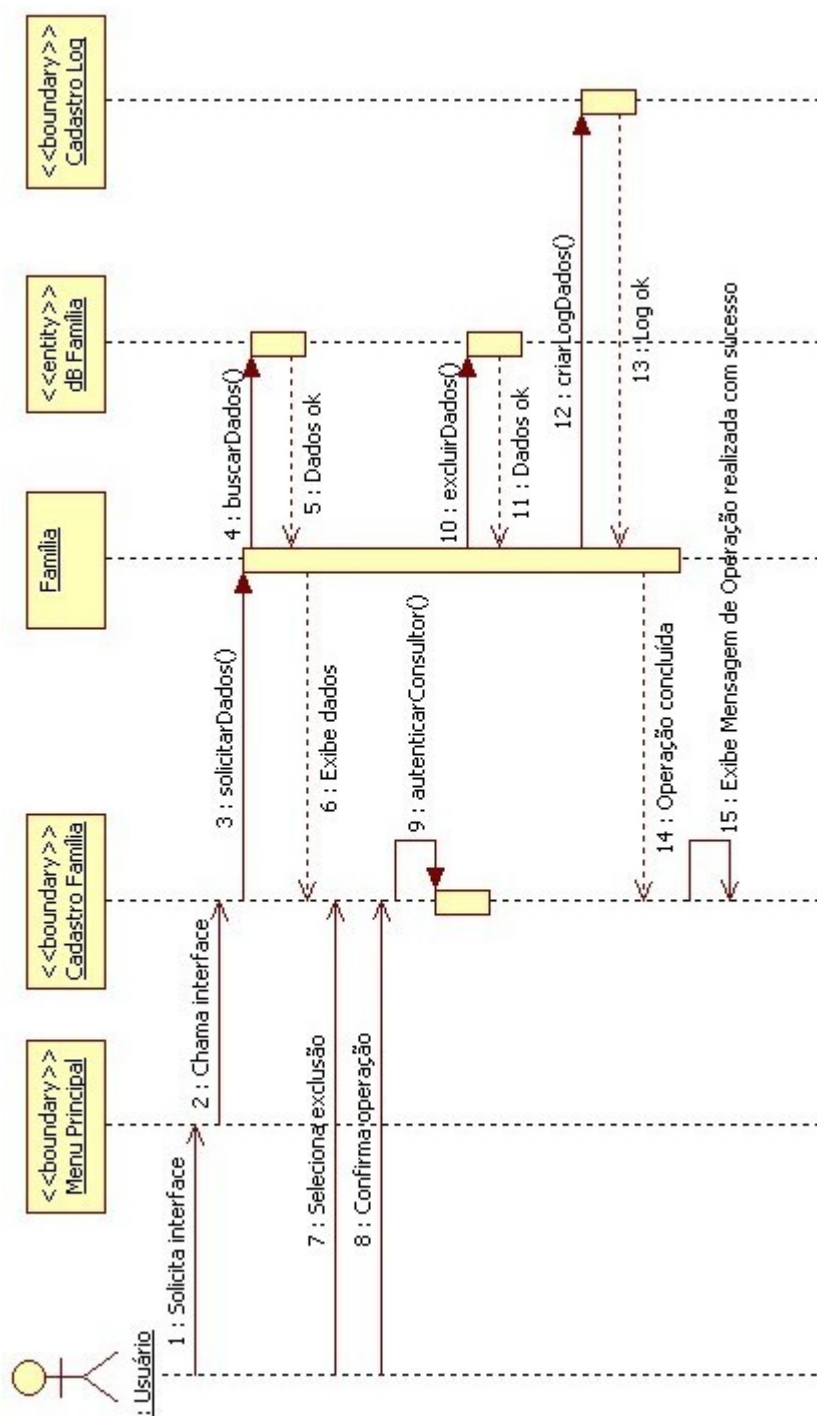


Figura 38 – Diagrama de Sequencia: Excluir Família

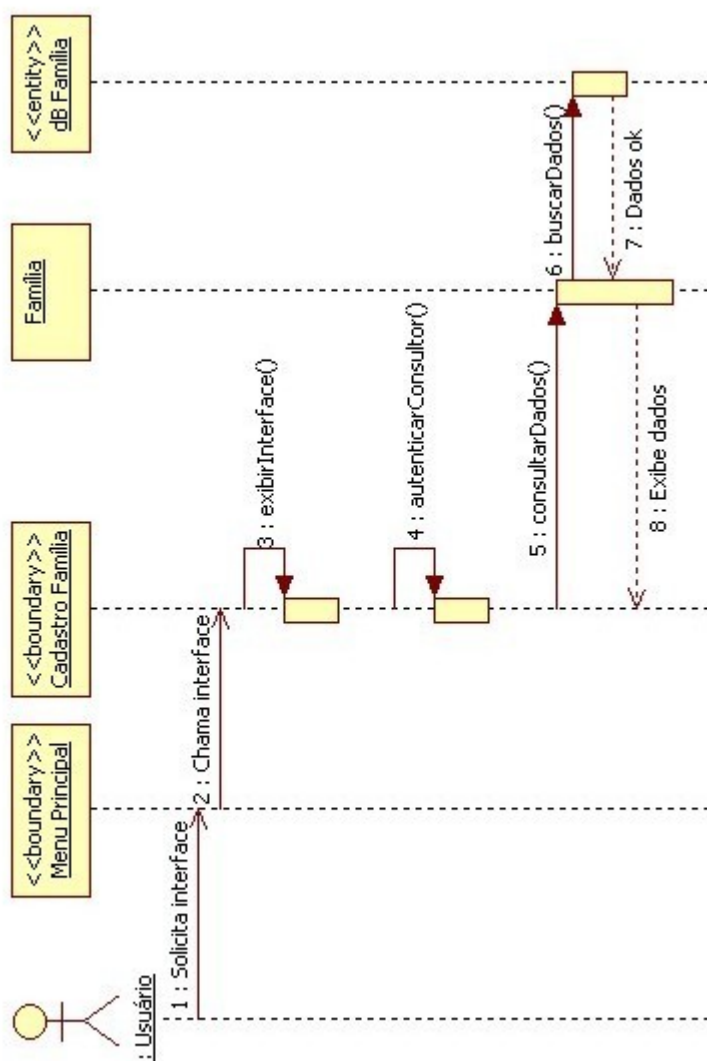


Figura 39 – Diagrama de Sequencia: Consultar Família

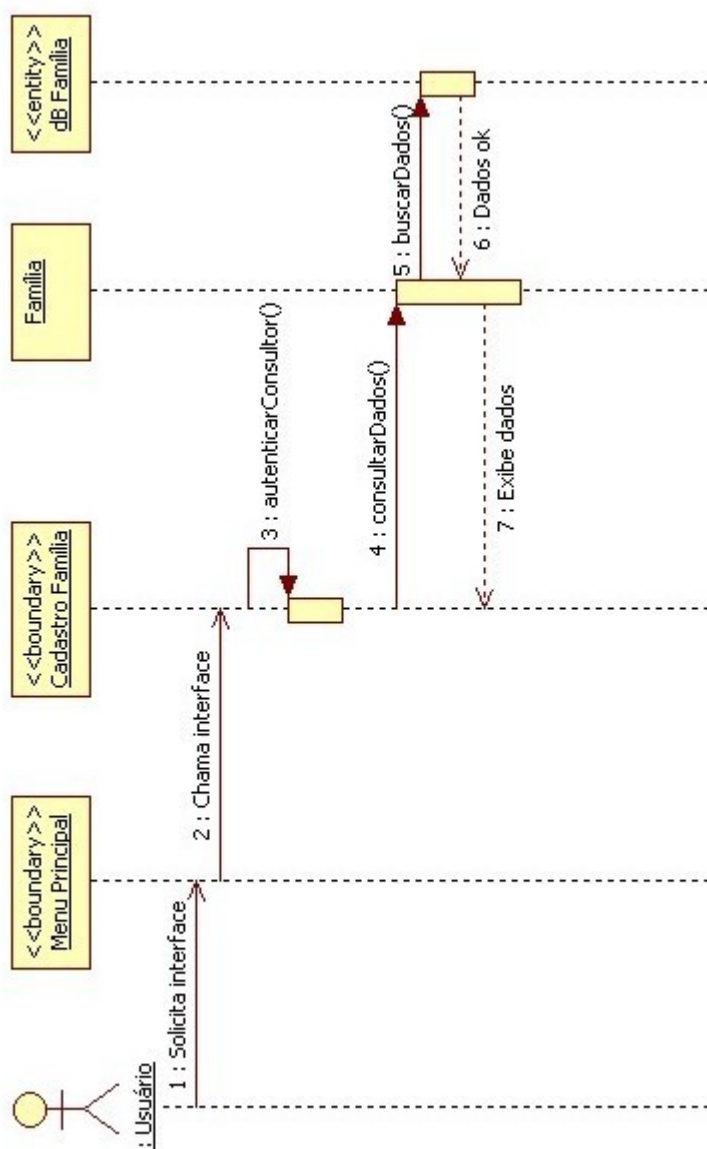


Figura 40 – Diagrama de Sequencia: Consultar Frequência Família

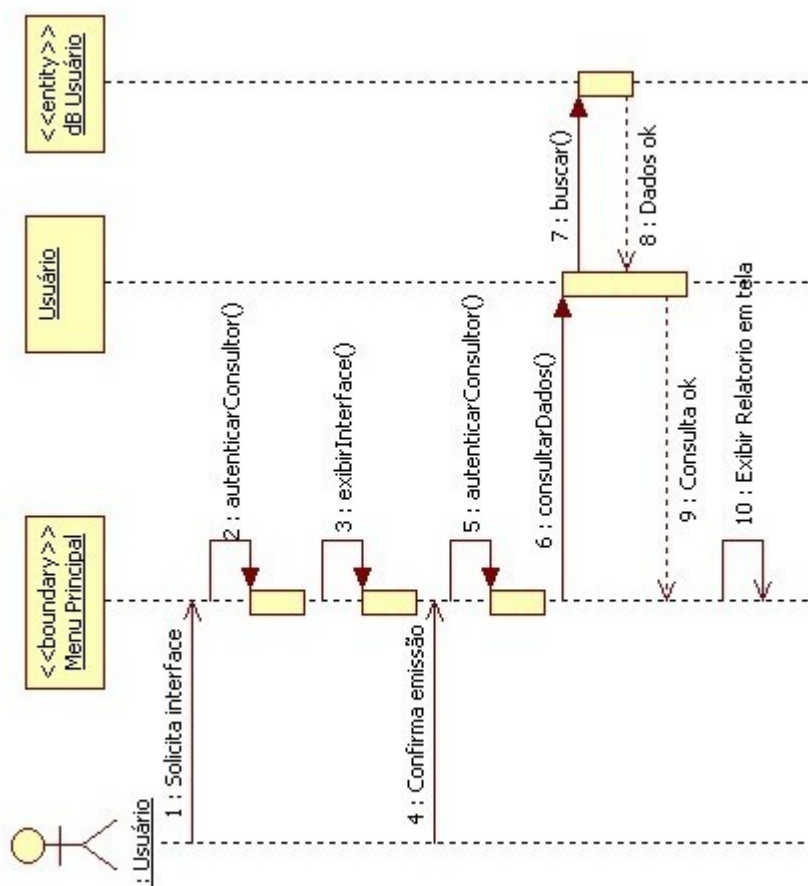


Figura 41 – Diagrama de Sequencia: Relatório Vencimento Matrícula

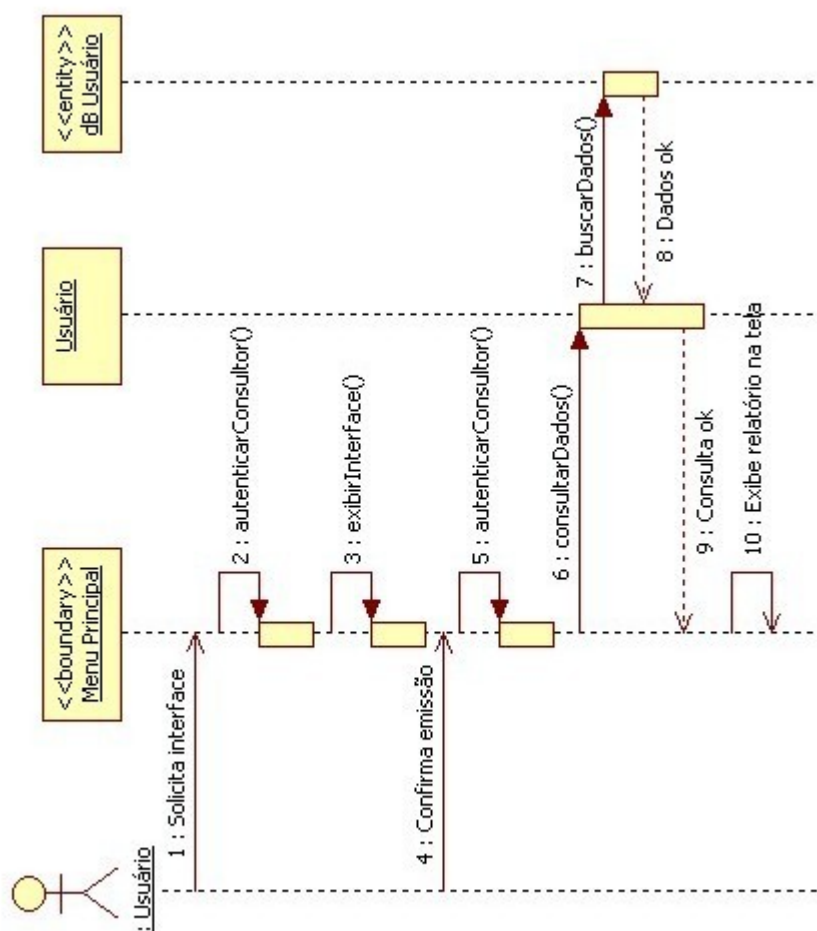


Figura 42 – Diagrama de Sequencia: Relatório Família Apta



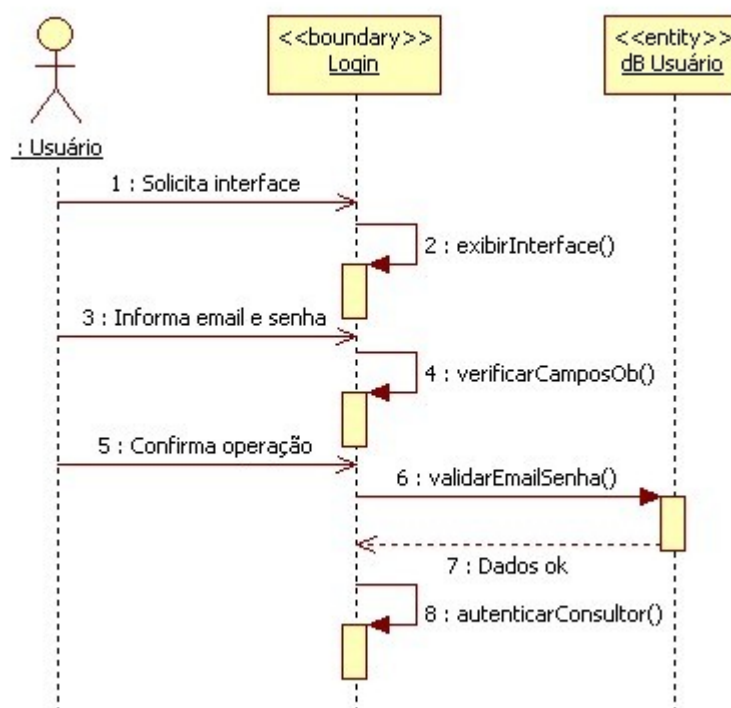


Figura 43 – Diagrama de Sequencia: Efetuar Login

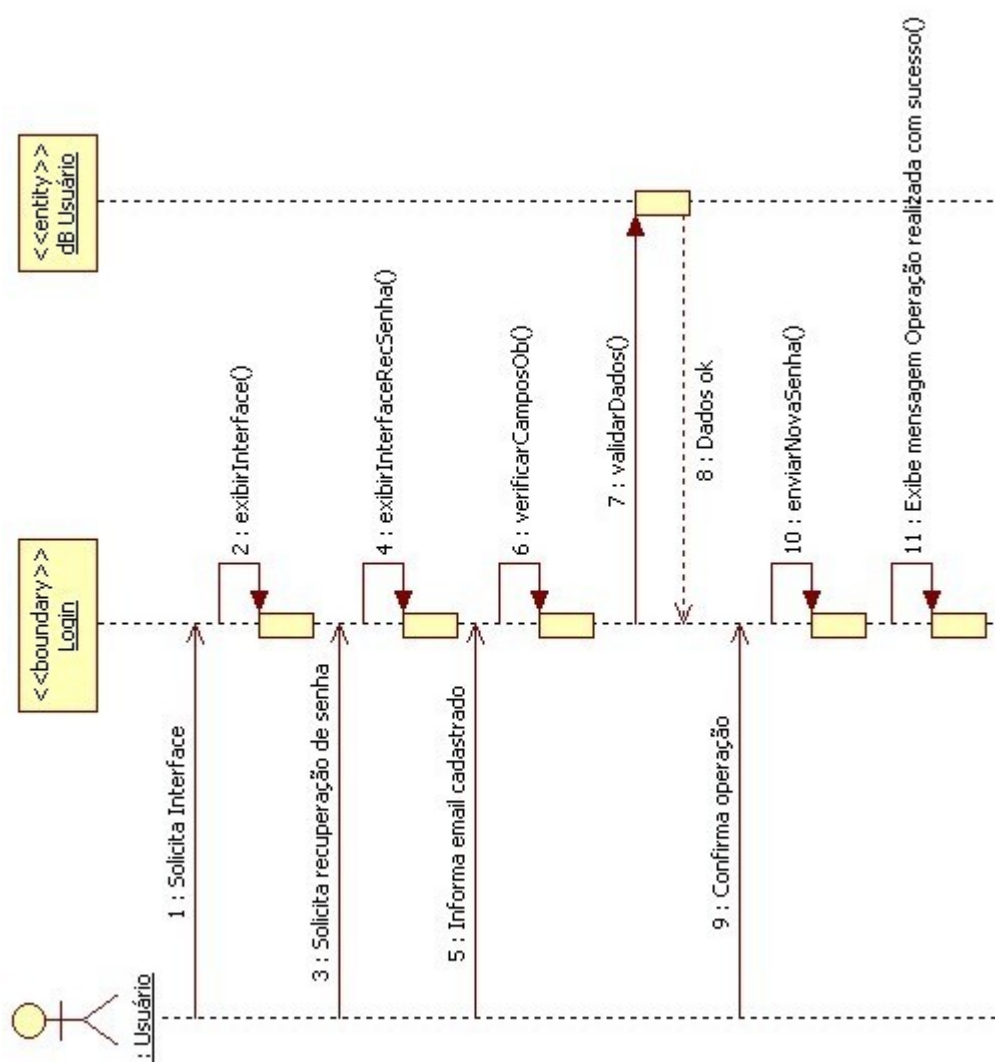


Figura 44 – Diagrama de Sequencia: Recuperar Senha

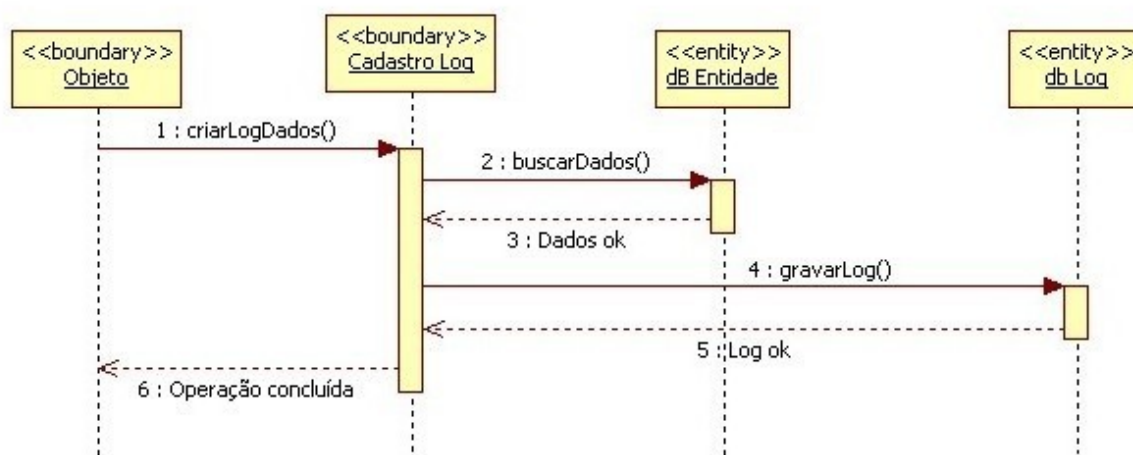


Figura 45 – Diagrama de Sequencia: Armazenar *Log Sistema*

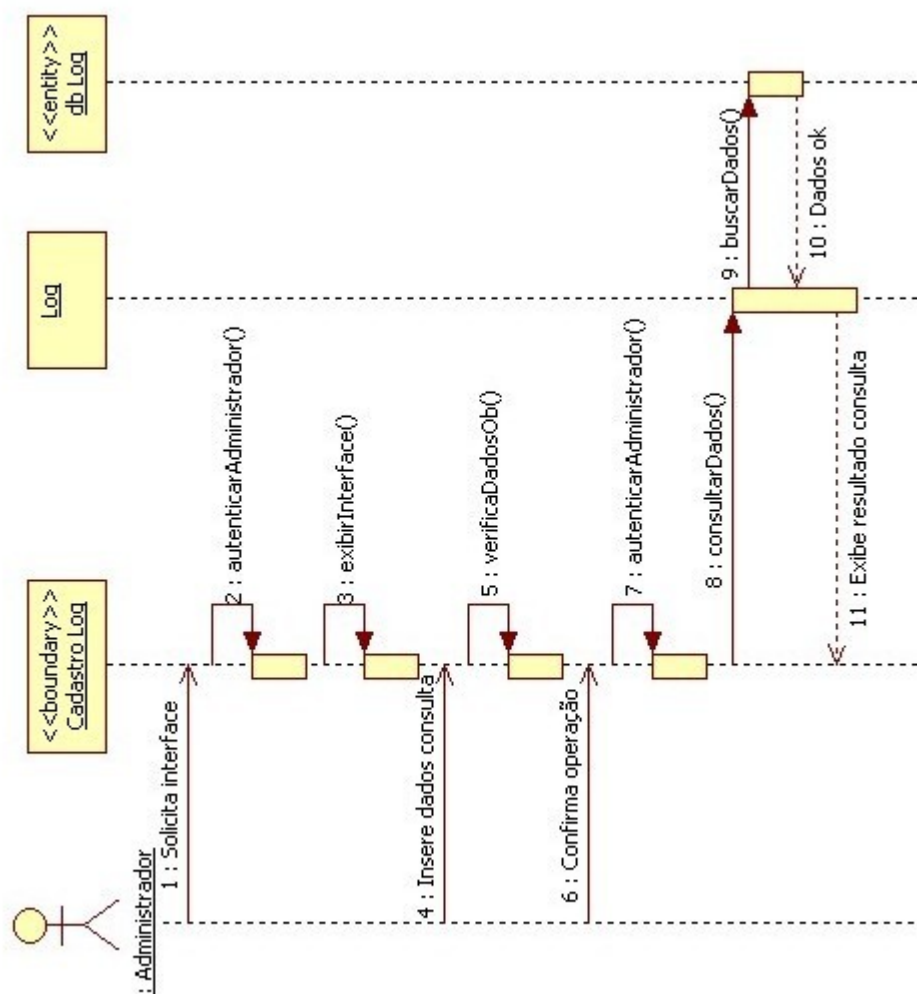


Figura 46 – Diagrama de Sequencia: Consultar Voluntário

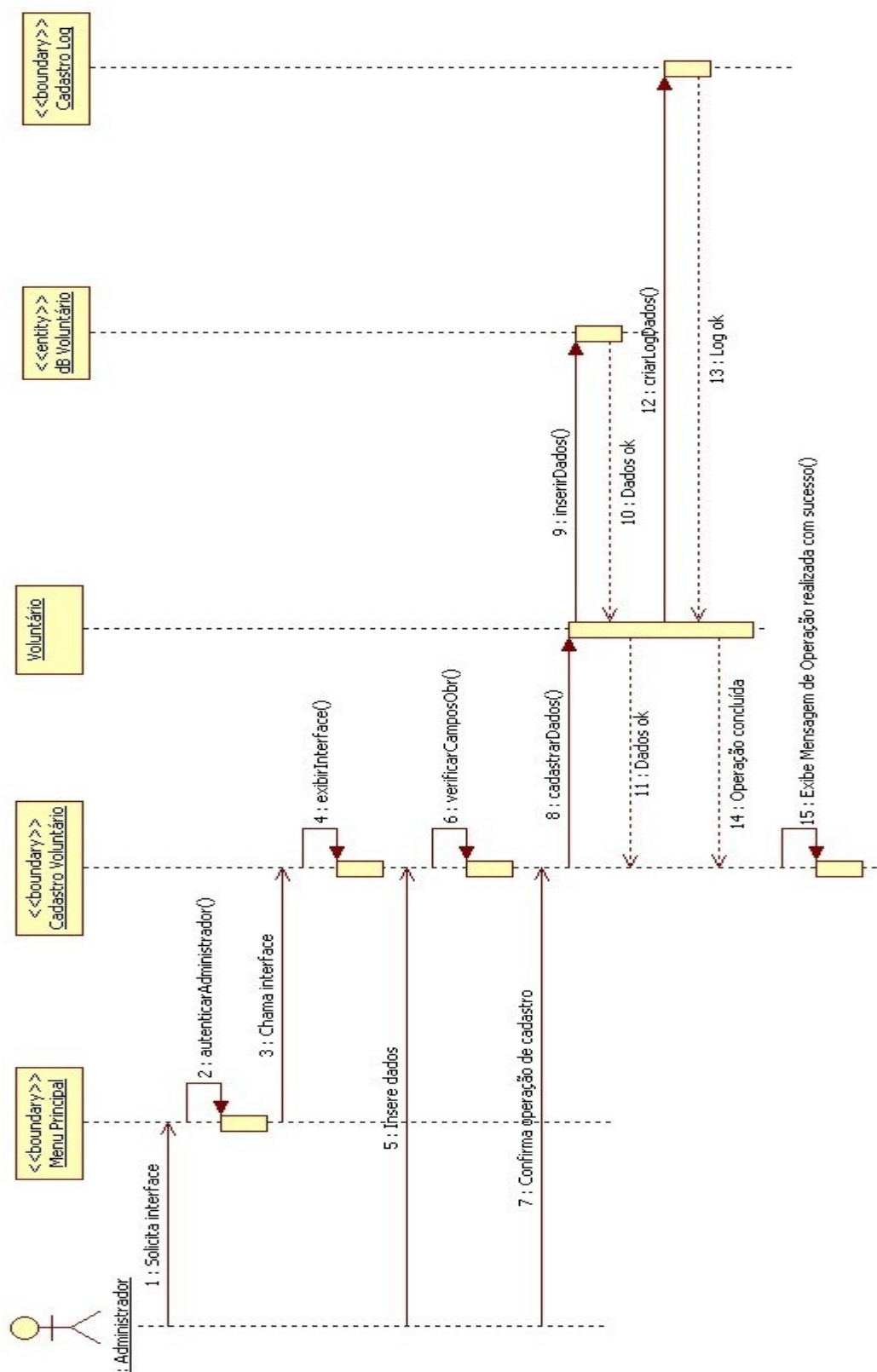


Figura 47 – Diagrama de Sequencia: Cadastrar Voluntário

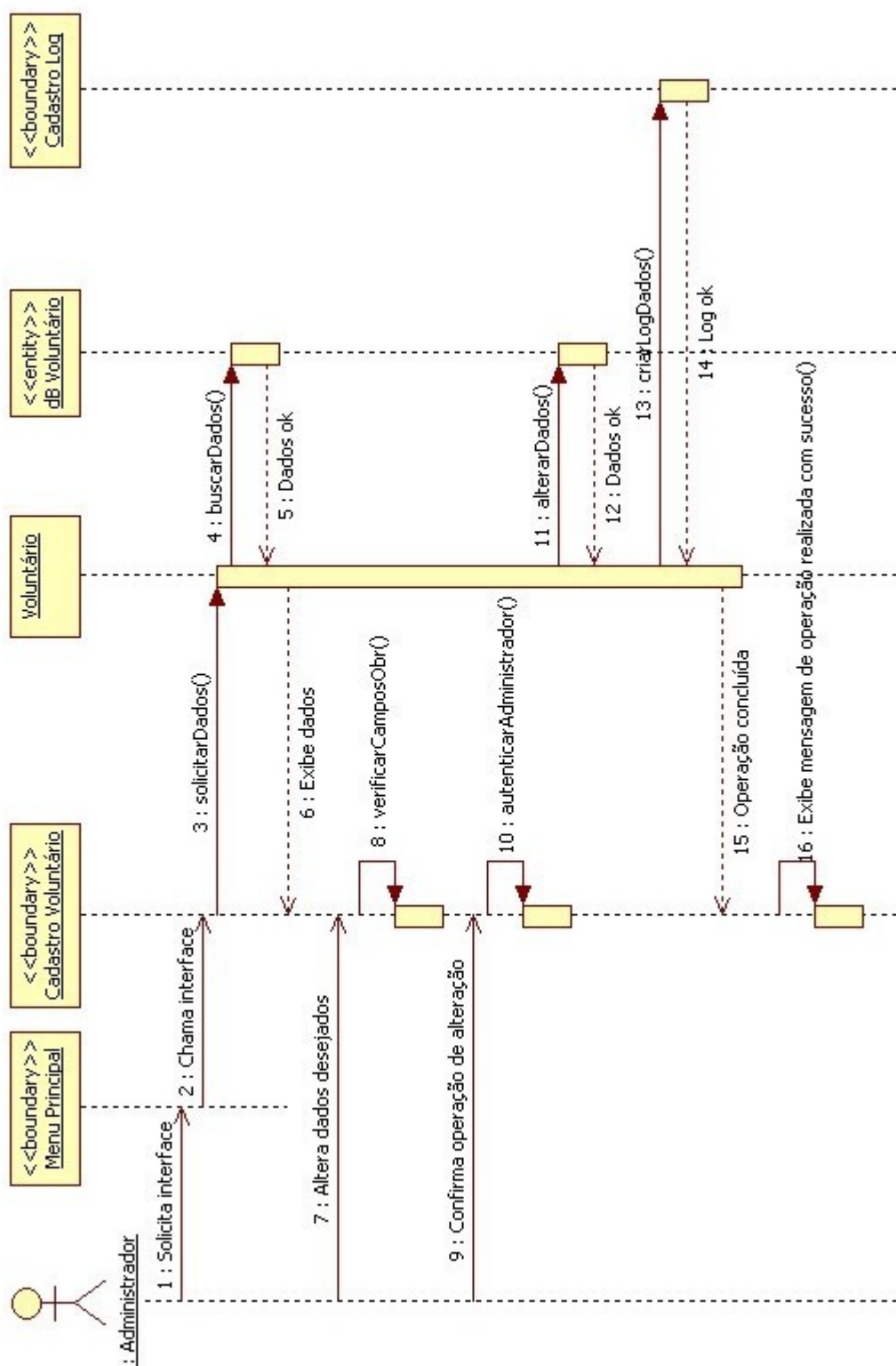


Figura 48 – Diagrama de Sequencia: Alterar Voluntário

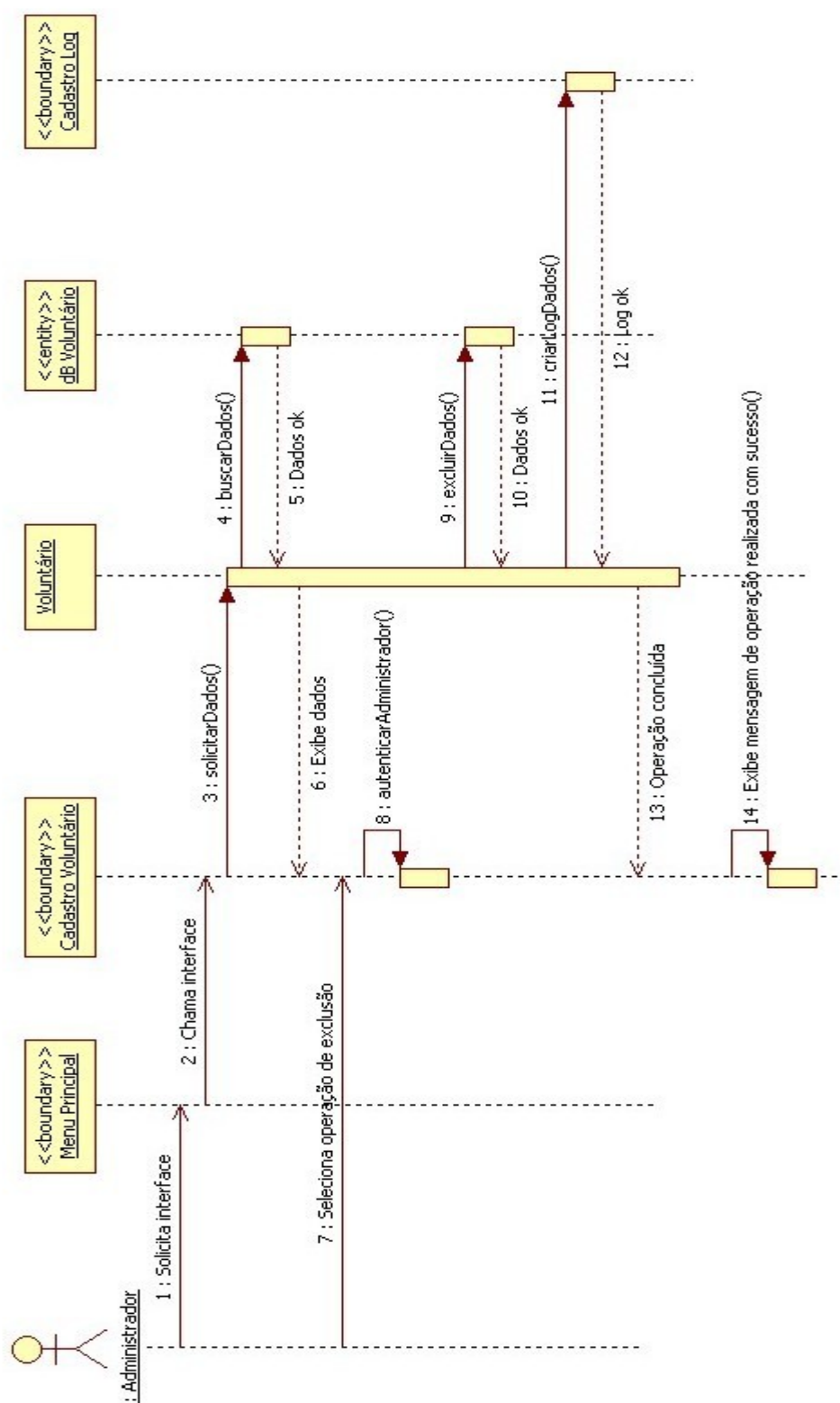


Figura 49 – Diagrama de Sequencia: Excluir Voluntário

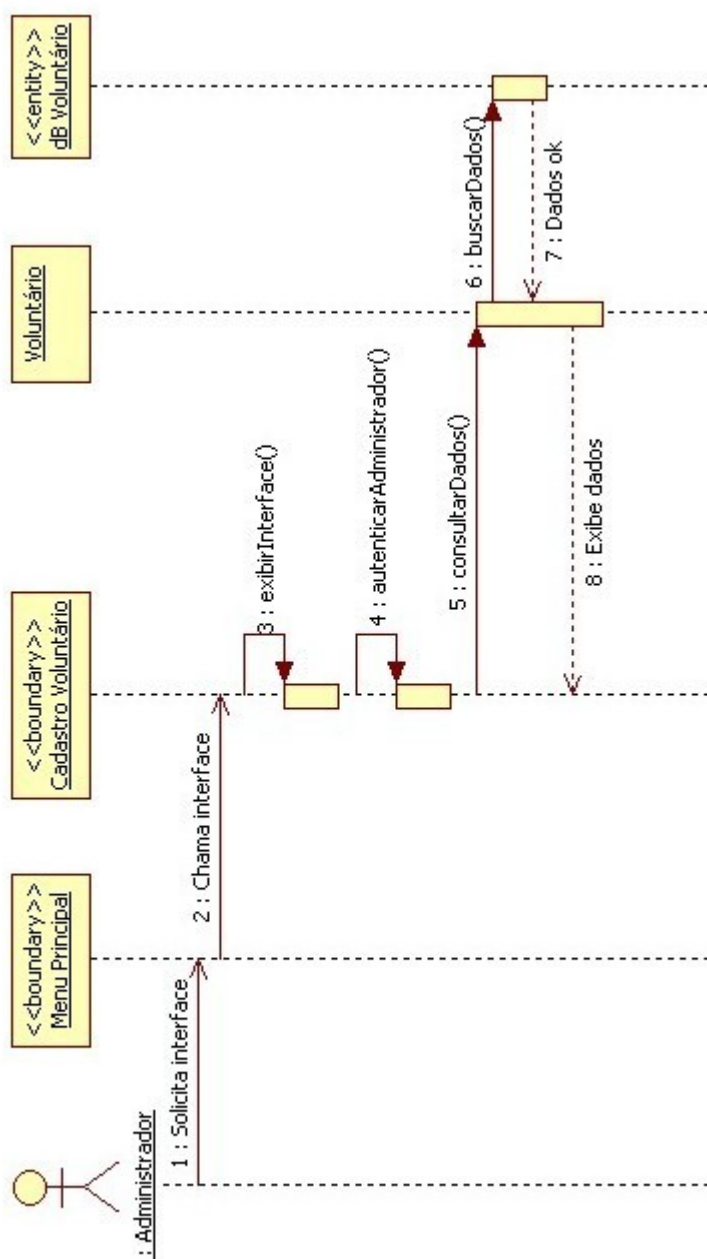


Figura 50 – Diagrama de Sequencia: Consultar Voluntário



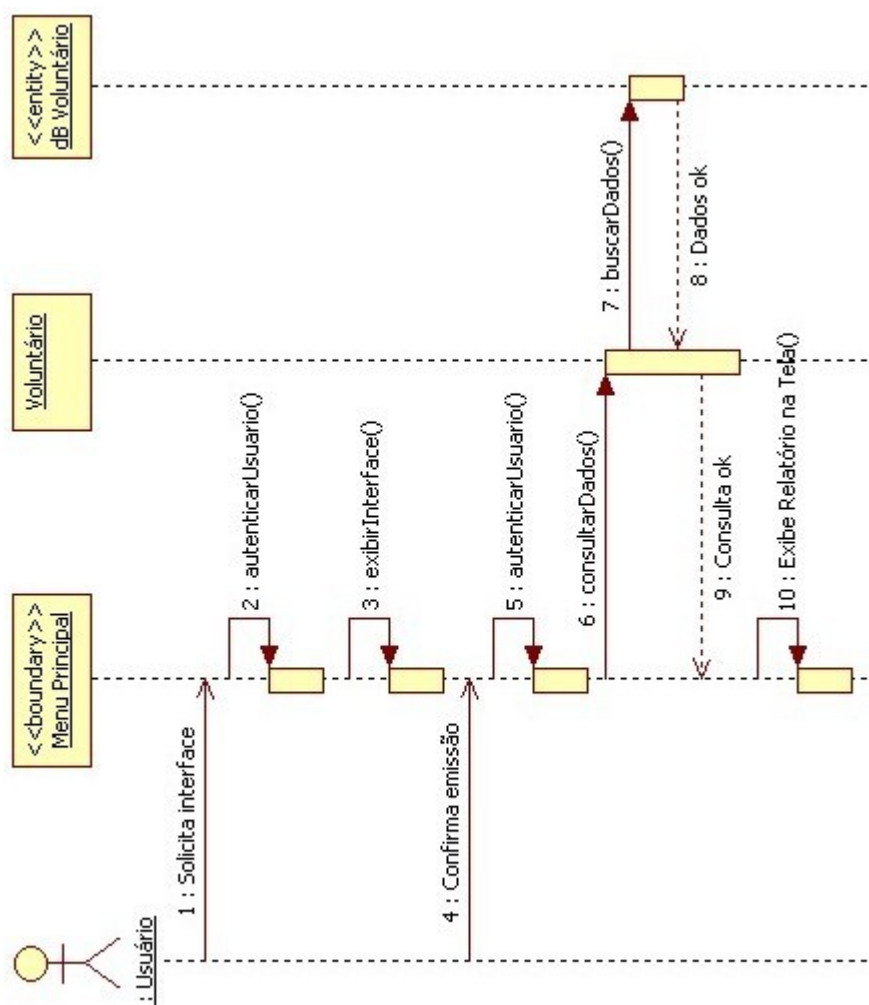


Figura 51 – Diagrama de Sequencia: Relatório Voluntário

#### **4.5 Diagrama de tabelas relacionais (DTR)**

Também conhecido como modelo lógico de dados, este por sua vez está um nível acima do modelo conceitual da dados, ou seja, o nível de abstração é menor para um simples usuário. Este modelo apresenta implementações, recursos de adequações de padrões, nomenclatura e normalização.

A normalização é um processo formal matemático que utiliza fundamentos da teoria dos conjuntos, visando substituir, de forma gradativa, um conjunto de entidade e relacionamentos por um outro mais adequado em relação às anomalias de atualização. Nessa etapa, define-se também as chaves primária e estrangeira de cada entidade, isso com relação ao DER(Diagrama de Entidade Relacional).

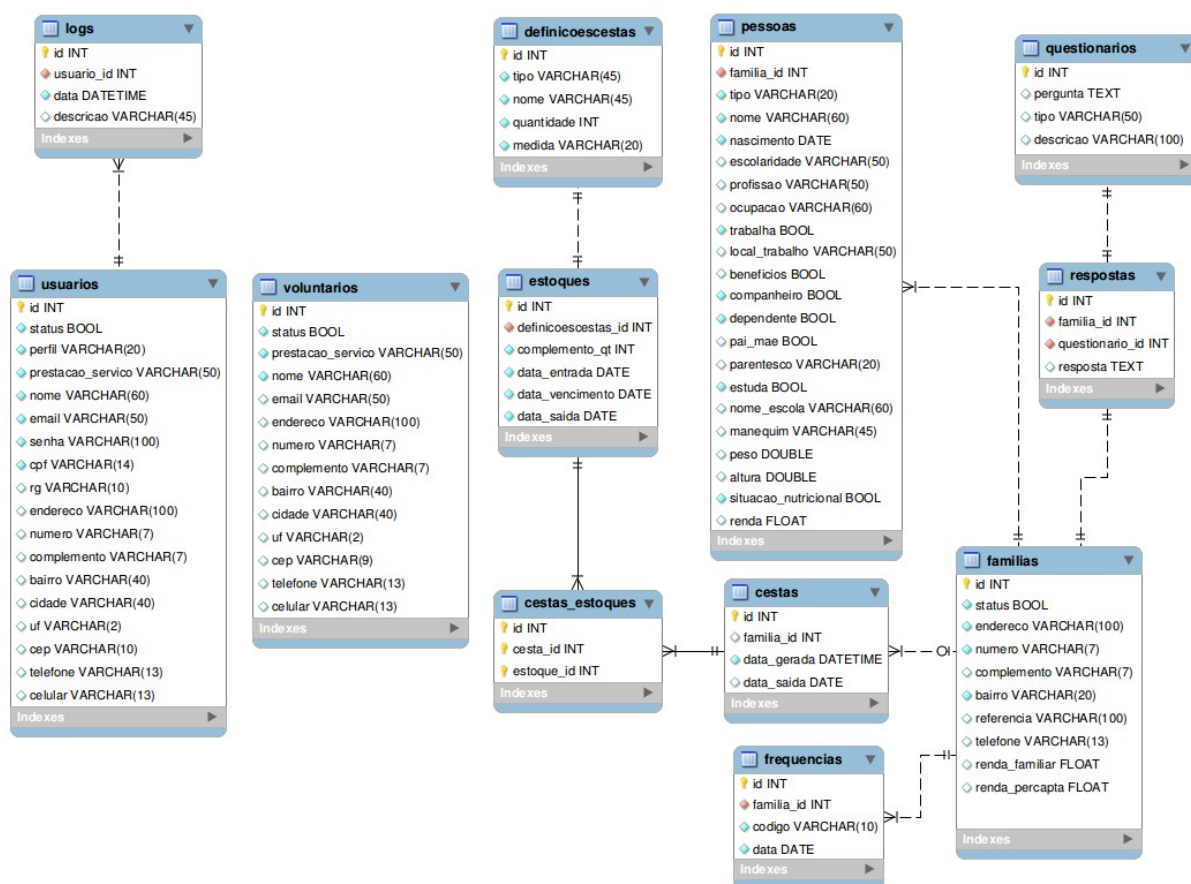


Figura 52 – DTR

## 5 MODELAGEM DE PROJETO

### 5.1 Considerações Iniciais

Este capítulo será composto por todos os elementos necessários ao projeto, que irão constituir o início de desenvolvimento do sistema proposto.

De acordo com Pressman (2006), concluiu-se que a modelagem do projeto está imersa numa abstração mais baixa do que a fase de modelagem e de análise, porém, em certos momentos, há uma interação – na qual seus modelos se misturam –; semelhante a ocorrida nos modelos de análise. Esses serão refinados e elaborados como parte do projeto.

Com muito pensar e analisar, definiu-se utilizar – de acordo com o porte do projeto e os prazos disponíveis para a sua construção – o modelo MVC (*Model View Controller*) para o desenvolvimento do sistema. Esse modelo permite que haja um desenvolvimento acelerado, diminuindo o tempo de codificação e focando nas funcionalidades essenciais às quais o projeto necessita para estar funcionando.

Optou-se em utilizar um banco de dados relacional por ser de fácil implementação, possuir uma vasta documentação e por ser o mais utilizado atualmente.

Um sistema com foco na *web*, porém com capacidade, também, de ser executado localmente, tornando-o um sistema flexível.

### 5.2 Arquitetura do Software

O projeto arquitetural do *software* equivale-se, analogamente, “à planta baixa” de uma casa.(PRESSMAN, 2006).

Essa etapa é aquela que a arquitetura torna-se a base para o projeto, consistindo numa representação na qual o engenheiro do *software* examinará os diversos aspectos que constituem o sistema, analisará a sua efetividade, levando em consideração as alterações arquiteturais para, assim, poder amenizar os riscos

associados.

De acordo com as necessidades do cliente de estar, por exemplo, trabalhando remotamente na aplicação com opção simultânea de vários acessos ao sistema, ficou definido que o desenvolvimento do sistema será com o foco voltado para a internet. A concluir, a arquitetura de funcionamento, estudou-se em uma tecnologia a ser utilizada para o desenvolvimento do sistema, e como este projeto possui uma visão focada no *software* livre, optou-se em utilizar a linguagem PHP – orientada a objetos –. Para a realização do desenvolvimento do *software*, optou-se, também, em utilizar o *framework CakePHP*, o qual possui uma infinidade de componentes e *plugins* que ajudarão no desenvolvimento do sistema, visto que a codificação desenvolvida será bem reduzida, pois a parte básica (*CRUD – Cread, Read, Update and Delete –*) será toda aproveitada, possibilitando uma maior agilidade na fluidez no desenvolvimento do *software*.

A seguir serão detalhadas as tecnologias utilizadas na concepção do projeto.

### 5.2.1 Framework CakePHP

O *CakePHP* é um *framework* de código aberto, voltado para o desenvolvimento ágil. Possui uma estrutura composta de vários componentes que se integram e facilitam na criação de uma nova aplicação web.

Agregando à sua estrutura a flexibilidade e rapidez na programação, proporcionando, ainda, ao programador um ambiente conciso para que se possa trabalhar. É um ambiente que fornece todas as ferramentas para dar início à programação, oferecendo a lógica específica da aplicação. Ao invés de escrever códigos do zero ou reinventá-los, o *CakePHP* oferece uma aplicação base simples para que possa dar início a um projeto.

Apesar de possuir compatibilidade com o PHP4 – não orientado a objetos –, possui em sua concepção um ambiente todo voltado para orientação a objetos que, por sua vez, trabalha em cima de uma arquitetura MVC (*Model View and Controller*).

### 5.2.2 MVC

O modelo de três camadas divide um aplicativo de maneira que a lógica do sistema resida no meio delas.

A arquitetura MVC (Modelo Visualização Controle) consegue gerar uma divisão eficaz das funcionalidades envolvidas na manutenção e na apresentação dos dados de um sistema.

Programando em MVC, há uma separação em três partes: O *model* representa os dados; a *view* representa a visualização dos dados e o *controller* manipula e roteia as requisições dos usuários.

A seguir segue uma ilustração do funcionamento do modelo MVC:

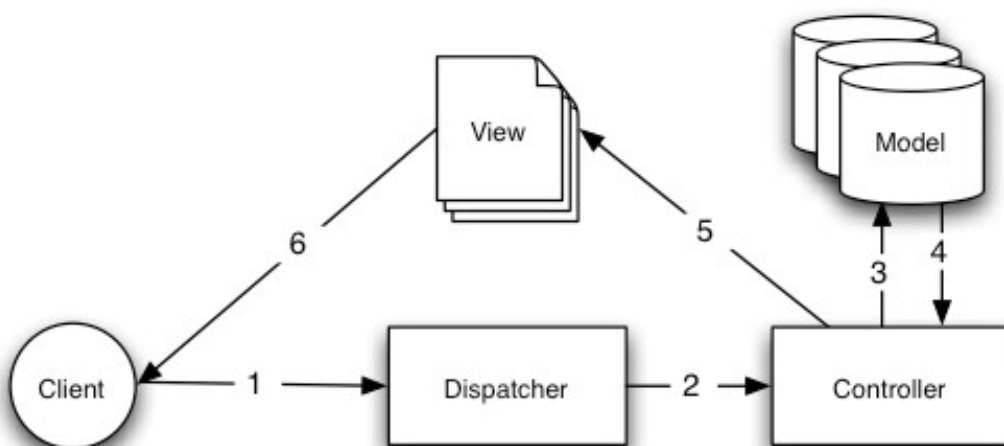


Figura 53 – Requisição básica de MVC

### 5.2.3 Ajax

O Ajax engloba um conjunto de tecnologias – *JavaScript*, *CSS (Cascading Style Sheets)* e *XML* – que se integram, possuindo o objetivo de renderizar páginas *web*, a utilizar recursos de *scripts* que serão executados pelo lado do cliente, buscando e carregando dados em *background* sem a necessidade de recarregar a página.

Já em uma aplicação *web* clássica, o navegador necessita buscar as

informações no servidor, para assim, estar retornando ao cliente. No Ajax, toda a lógica de processamento dos dados é passada diretamente para o cliente. Quando o usuário faz uma requisição, o *JavaScript* é o responsável por trazer as informações de forma assíncrona, sem causar o "*reload*" da página.

Os dados tratados ficam por conta do *script* que foi carregado quando obteve-se acesso à página, depois somente os dados que serão utilizados são carregados, tornando o *site* mais rápido e dinâmico.

### 5.3 Especificação de Interface

“Por decisão do colegiado de curso, é opcional a apresentação do item Especificação de Interfaces neste documento de projeto”

### 5.4 Estrutura de Dados

#### 5.4.1 Modelagem Física de Banco de Dados

No modelo físico é possível visualizar como os dados são armazenados no banco de dados, assim como seus detalhes e a sua forma de acesso. Nesse modelo serão apresentados os nomes definitivos dos atributos, o tipo de dado de cada atributo (de acordo com o SGBD MySQL) e restrições de integridade como a impossibilidade da entrada de informações duplicadas ou nulas, quando aplicável.

A nomenclatura das tabelas foram criadas de acordo com o "aconselhamento", padronização sugerida pelo *framework CakePHP*, no qual todos os nomes das tabelas devem ser escritos no plural; tabelas de relacionamento escritas no plural separadas por *underline*; e seus atributos no singular.

#### logs

Campo	Tipo	AI	Não nulo	PK/FK	Observação
-------	------	----	----------	-------	------------

id	int	Sim	Sim	PK	Identificador do log
usuario_id	int	Não	Sim	FK	Identificador do usuário
data	datetime	Não	Sim		Data da operação
descricao	varchar(45)	Não	Não		Descrição da operação

Quadro 32 – estrutura de dados da tabela *logs***usuarios**

Campo	Tipo	AI	Não nulo	PK/FK	Observação
id	int	Sim	Sim	PK	Identificador do usuário
grupo_id	int	Não	Sim	FK	Identificador do grupo
status	bool	Não	Sim		Status do usuário
nome	varchar(60)	Não	Sim		Nome do usuário
email	varchar(50)	Não	Sim		E-mail do usuário
senha	varchar(140)	Não	Sim		Senha do usuário
cpf	varchar(14)	Não	Sim		Documento de CPF
rg	varchar(10)	Não	Não		Documento de identidade
endereco	varchar(100)	Não	Não		Endereço residencial
numero	varchar(7)	Não	Não		Endereço residencial
complemento	varchar(7)	Não	Não		Endereço residencial
bairro	varchar(40)	Não	Não		Endereço residencial
cidade	varchar(40)	Não	Não		Endereço residencial
uf	varchar(2)	Não	Não		Endereço residencial
cep	varchar(9)	Não	Não		Endereço residencial
telefone	varchar(14)	Não	Não		Telefone para contato
celular	varchar(14)	Não	Não		Telefone para contato



prestacao_servico	varchar(20)	Não	Sim		Tipo de prestação de serviço
-------------------	-------------	-----	-----	--	------------------------------

Quadro 33 – estrutura de dados da tabela usuarios

**Voluntarios**

<b>Campo</b>	<b>Tipo</b>	<b>AI</b>	<b>Não Nulo</b>	<b>PK/FK</b>	<b>Observação</b>
id	int	Sim	Sim	PK	Identificador do voluntário
status	bool	Não	Sim		Status do voluntário
nome	varchar(60)	Não	Sim		Nome do voluntário
email	varchar(50)	Não	Não		E-mail do voluntário
endereco	varchar(100)	Não	Não		Endereço residencial
numero	varchar(7)	Não	Não		Endereço residencial
complemento	varchar(7)	Não	Não		Endereço residencial
bairro	varchar(40)	Não	Não		Endereço residencial
cidade	varchar(40)	Não	Não		Endereço residencial
uf	varchar(2)	Não	Não		Endereço residencial
cep	varchar(9)	Não	Não		Endereço residencial
telefone	varchar(14)	Não	Não		Telefone para contato
celular	varchar(14)	Não	Não		Telefone para contato
prestacao_servico	varchar(20)	Não	Não		Tipo de prestação de serviço

Quadro 34 – estrutura de dados da tabela voluntarios

**definicoescestas**

<b>Campo</b>	<b>Tipo</b>	<b>AI</b>	<b>Não Nulo</b>	<b>PK/FK</b>	<b>Observação</b>
--------------	-------------	-----------	-----------------	--------------	-------------------

id	int	Sim	Sim	PK	Identificador das definições de cestas
tipo	varchar(45)	Não	Sim		Tipo de mantimento cesta
nome	varchar(45)	Não	Sim		Nome do mantimento
quantidade	int	Não	Sim		Quantidade a cadastrar
medida	varchar(20)	Não	Sim		Unidade de medida

Quadro 35 – estrutura de dados da tabela definicoescestas

**estoques**

Campo	Tipo	AI	Não Nulo	PK/FK	Observação
id	int	Sim	Sim	PK	Identificador do estoque
definicoescestas_id	int	Não	Sim	FK	Identificado da definições de cestas
Quantidade	int	Não	Sim		Quantidade de itens
complemento_qt	int	Não	Sim		Quantidade, peso, do mantimento.
data_entrada	date	Não	Sim		Data de entrada mantimento
data_vencimento	date	Não	Sim		Data de validade do mantimento.

Quadro 36 – estrutura de dados da tabela estoques

**cestas\_estoques**

Campo	Tipo	AI	Não Nulo	PK/FK	Observação
id	int	Sim	Sim	PK	Identificador do relacionamento cesta / estoque
cesta_id	int	Não	Sim	PK	Identificador de cesta

estoque_id	int	Não	Sim	PK	Identificador do estoque
------------	-----	-----	-----	----	--------------------------

Quadro 37 – estrutura de dados da tabela cestas\_estoques

**cestas**

Campo	Tipo	AI	Não Nulo	PK/FK	Observação
id	int	Sim	Sim	PK	Identificador de cestas
familia_id	int	Não	Não	FK	Identificador de família
data_gerada	datetime	Não	Sim		Data da cesta gerada
data_saida	date	Não	Não		Data de saída da cesta

Quadro 38 – estrutura de dados da tabela cestas

**frequencias**

Campo	Tipo	AI	Não Nulo	PK/FK	Observação
id	int	Sim	Sim	PK	Identificador de frequências
familia_id	int	Não	Sim	FK	Identificador de família
codigo	varchar(10)	Não	Sim		Código da frequência
data	date	Não	Sim		Data da frequência

Quadro 39 – estrutura de dados da tabela frequencias

**familias**

Campo	Tipo	AI	Não Nulo	PK/FK	Observação
id	int	Sim	Sim	PK	Identificador de família
nome	varchar(60)	Não	Sim		Nome do responsável
parente_id	int	Não	Não		Para identificar as pessoas que compõe a

					família.
escolaridade	varchar(50)	Não	Não		Nível de escolaridade
profissao	varchar(100)	Não	Não		Profissão atuante
local_trabalho	varchar(50)	Não	Não		Local onde trabalha
beneficios	bool	Não	Não		Recebe algum benefício
prestacao_sevico	varchar(50)	Não	Sim		Prestação de serviço
parentesco	int	Não	Não		Parentesco com responsável
situacao	int	Não	Sim		Situação
nome_escola	varchar(60)	Não	Não		Nome da escola
manequim	varchar(45)	Não	Não		Manequim da pessoa
peso	double	Não	Não		Peso da pessoa
altura	double	Não	Não		Altura da pessoa
situacao_nutricional	bool	Não	Sim		Situação nutricional
cpf	varchar(14)	Não	Não		Numero do CPF
nascimento	date	Não	Sim		Data de nascimento
endereco	varchar(100)	Não	Sim		Endereço residencial
numero	varchar(7)	Não	Não		Endereço residencial
complemento	varchar(7)	Não	Não		Endereço residencial
bairro	varchar(40)	Não	Sim		Endereço residencial
referencia	varchar(100)	Não	Não		Endereço residencial
cidade	varchar(40)	Não	Não		Cidade da família
uf	varchar(2)	Não	Não		Estado da cidade
cep	varchar(9)	Não	Não		Endereço postal
telefone	varchar(14)	Não	Não		Telefone para contato

renda	float	Não	Não		Reanda para dependentes acima de 16 anos
renda_familiar	float	Não	Não		Renda familiar
renda_percapta	float	Não	Não		Renda per capita
companheiro	bool	Não	Sim		Companheiro(a) do(a) responsável
dependente	bool	Não	Sim		Dependente da família

Quadro 40 – estrutura de dados da tabela familias

**respostas**

<b>Campo</b>	<b>Tipo</b>	<b>AI</b>	<b>Não Nulo</b>	<b>PK/FK</b>	<b>Observação</b>
id	int	Sim	Sim	PK	Identificador de resposta
familia_id	int	Não	Sim	FK	Identificador de família
questionatio_id	int	Não	Sim	FK	Identificador de questionário
resposta	text	Não	Não		Respostas do questionário

Quadro 41 – estrutura de dados da tabela respostas

**questionarios**

<b>Campo</b>	<b>Tipo</b>	<b>AI</b>	<b>Não Nulo</b>	<b>PK/FK</b>	<b>Observação</b>
id	int	Sim	Sim	PK	Identificador de questionário
pergunta	varchar(200)	Não	Não		Pergunta realizada
tipo	varchar(50)	Não	Não		Identifica se é um booleano, texto ou select

extra	text	Não	Não		Define o que irá compor o campo 'tipo'.
ordem	int	Não	SIM		Define a ordem das perguntas do questionário.

Quadro 42 – estrutura de dados da tabela questionarios

#### 5.4.2 Estrutura do Banco de Dados

Abaixo segue o modelo físico de criação do banco de dados, elaborado e expresso em SQL utilizando a ferramenta MySQL WorkBench, usado para este projeto.

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL';
```

```
-- -----
-- Table `grupos`
-- -----
```

```
DROP TABLE IF EXISTS `grupos` ;
```

```
CREATE TABLE IF NOT EXISTS `grupos` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `nome` VARCHAR(200) NULL ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;
```

```
-- -----
-- Table `usuarios`
```

```

-----
DROP TABLE IF EXISTS `usuarios` ;

CREATE TABLE IF NOT EXISTS `usuarios` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `grupo_id` INT NOT NULL ,
  `status` TINYINT(1) NOT NULL ,
  `nome` VARCHAR(60) NOT NULL ,
  `email` VARCHAR(50) NOT NULL ,
  `senha` VARCHAR(140) NOT NULL ,
  `cpf` VARCHAR(14) NULL ,
  `endereco` VARCHAR(100) NULL ,
  `complemento` VARCHAR(7) NULL ,
  `numero` VARCHAR(7) NULL ,
  `bairro` VARCHAR(40) NULL ,
  `cidade` VARCHAR(40) NULL ,
  `uf` VARCHAR(2) NULL ,
  `cep` VARCHAR(9) NULL ,
  `telefone` VARCHAR(14) NULL ,
  `celular` VARCHAR(14) NULL ,
  `prestacao_servico` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_usuarios_grupos1` (`grupo_id` ASC) ,
  CONSTRAINT `fk_usuarios_grupos1`
    FOREIGN KEY (`grupo_id` )
    REFERENCES `grupos` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;

```

```

-----
-- Table `logs`

```

```
-- -----
DROP TABLE IF EXISTS `logs` ;
```

```
CREATE TABLE IF NOT EXISTS `logs` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `usuario_id` INT NOT NULL ,
  `data` DATETIME NOT NULL ,
  `descricao` VARCHAR(45) NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_logs_usuarios1` (`usuario_id` ASC) ,
  CONSTRAINT `fk_logs_usuarios1`
    FOREIGN KEY (`usuario_id`)
    REFERENCES `usuarios` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;
```

```
-- -----
-- Table `definicoescestas`
-- -----
```

```
DROP TABLE IF EXISTS `definicoescestas` ;
```

```
CREATE TABLE IF NOT EXISTS `definicoescestas` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `tipo` VARCHAR(45) NOT NULL ,
  `nome` VARCHAR(45) NOT NULL ,
  `quantidade` INT NOT NULL ,
  `medida` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;
```



```

-- -----
-- Table `estoques`
-- -----

DROP TABLE IF EXISTS `estoques` ;

CREATE TABLE IF NOT EXISTS `estoques` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `definicoescesta_id` INT NOT NULL ,
  `quantidade` INT NOT NULL ,
  `complemento_qt` INT NOT NULL ,
  `data_entrada` DATE NOT NULL ,
  `data_vencimento` DATE NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_estoques_definicoescestas1` (`definicoescesta_id` ASC) ,
  CONSTRAINT `fk_estoques_definicoescestas1`
    FOREIGN KEY (`definicoescesta_id`)
    REFERENCES `definicoescestas` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;

-- -----
-- Table `familias`
-- -----

DROP TABLE IF EXISTS `familias` ;

CREATE TABLE IF NOT EXISTS `familias` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `nome` VARCHAR(60) NOT NULL ,
  `parente_id` INT NULL ,
  `escolaridade` VARCHAR(50) NULL ,
  `profissao` VARCHAR(100) NULL ,

```

```

`ocupacao` VARCHAR(100) NULL ,
`local_trabalho` VARCHAR(50) NULL ,
`beneficios` TINYINT(1) NULL ,
`parentesco` INT NULL ,
`situacao` INT NOT NULL ,
`nome_escola` VARCHAR(60) NULL ,
`manequim` VARCHAR(45) NULL ,
`peso` DOUBLE NULL ,
`altura` DOUBLE NULL ,
`situacao_nutricional` TINYINT(1) NOT NULL ,
`cpf` VARCHAR(14) NULL ,
`nascimento` DATE NOT NULL ,
`endereco` VARCHAR(100) NOT NULL ,
`numero` VARCHAR(7) NULL ,
`complemento` VARCHAR(7) NULL ,
`bairro` VARCHAR(40) NOT NULL ,
`referencia` VARCHAR(100) NULL ,
`cidade` VARCHAR(40) NULL ,
`uf` VARCHAR(2) NULL ,
`cep` VARCHAR(9) NULL ,
`telefone` VARCHAR(14) NULL ,
`renda` FLOAT NULL ,
`renda_familiar` FLOAT NULL ,
`renda_percapta` FLOAT NULL ,
`companheiro` TINYINT(1) NOT NULL ,
`dependente` TINYINT(1) NOT NULL ,
PRIMARY KEY (`id`) )
ENGINE = InnoDB;

```

```

-- -----
-- Table `cestas`
-- -----

```

```

DROP TABLE IF EXISTS `cestas` ;

```

```

CREATE TABLE IF NOT EXISTS `cestas` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `familia_id` INT NULL ,
  `data_gerada` DATETIME NOT NULL ,
  `data_saida` DATE NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_cestas_pessoas1` (`familia_id` ASC) ,
  CONSTRAINT `fk_cestas_pessoas1`
    FOREIGN KEY (`familia_id`)
    REFERENCES `familias` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;

```

```

-- -----
-- Table `questionarios`
-- -----

DROP TABLE IF EXISTS `questionarios` ;

```

```

CREATE TABLE IF NOT EXISTS `questionarios` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `pergunta` VARCHAR(200) NULL ,
  `tipo` VARCHAR(50) NULL ,
  `extra` TEXT NULL ,
  `ordem` INT NOT NULL ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;

```

```

-- -----
-- Table `frequencias`
-- -----

```

```
DROP TABLE IF EXISTS `frequencias` ;
```

```
CREATE TABLE IF NOT EXISTS `frequencias` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `familia_id` INT NOT NULL ,
  `codigo` VARCHAR(10) NOT NULL ,
  `data` DATE NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_frequencias_pessoas1` (`familia_id` ASC) ,
  CONSTRAINT `fk_frequencias_pessoas1`
    FOREIGN KEY (`familia_id` )
    REFERENCES `familias` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-- -----
-- Table `cestas_estoques`
-- -----
```

```
DROP TABLE IF EXISTS `cestas_estoques` ;
```

```
CREATE TABLE IF NOT EXISTS `cestas_estoques` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `cesta_id` INT NOT NULL ,
  `estoque_id` INT NOT NULL ,
  PRIMARY KEY (`id`, `cesta_id`, `estoque_id`) ,
  INDEX `fk_cestas_has_estoques_cestas1` (`cesta_id` ASC) ,
  INDEX `fk_cestas_has_estoques_estoques1` (`estoque_id` ASC) ,
  CONSTRAINT `fk_cestas_has_estoques_cestas1`
    FOREIGN KEY (`cesta_id` )
    REFERENCES `cestas` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_cestas_has_estoques_estoques1`
```

```

        FOREIGN KEY (`estoque_id` )
        REFERENCES `estoques` (`id` )
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;

-- -----
-- Table `respostas`
-- -----

DROP TABLE IF EXISTS `respostas` ;

CREATE TABLE IF NOT EXISTS `respostas` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `familia_id` INT NOT NULL ,
  `questionario_id` INT NOT NULL ,
  `resposta` TEXT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_respostas_questionarios1` (`questionario_id` ASC) ,
  INDEX `fk_respostas_pessoas1` (`familia_id` ASC) ,
  CONSTRAINT `fk_respostas_questionarios1`
    FOREIGN KEY (`questionario_id` )
    REFERENCES `questionarios` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_respostas_pessoas1`
    FOREIGN KEY (`familia_id` )
    REFERENCES `familias` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `acos`
-----

DROP TABLE IF EXISTS `acos` ;

CREATE TABLE IF NOT EXISTS `acos` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `parent_id` INT NULL ,
  `model` VARCHAR(255) NULL ,
  `foreign_key` INT NULL ,
  `alias` VARCHAR(255) NULL ,
  `lft` INT NULL ,
  `rght` INT NULL ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;

-----
-- Table `aros`
-----

DROP TABLE IF EXISTS `aros` ;

CREATE TABLE IF NOT EXISTS `aros` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `parent_id` INT NULL ,
  `model` VARCHAR(255) NULL ,
  `foreign_key` INT NULL ,
  `alias` VARCHAR(255) NULL ,
  `lft` INT NULL ,
  `rght` INT NULL ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;

-----

```

```

-- Table `aros_acos`
-- -----

DROP TABLE IF EXISTS `aros_acos` ;

CREATE TABLE IF NOT EXISTS `aros_acos` (
  `id` INT NOT NULL ,
  `aro_id` INT NOT NULL ,
  `aco_id` INT NOT NULL ,
  `_create` CHAR(2) NULL ,
  `_read` CHAR(2) NULL ,
  `_update` CHAR(2) NULL ,
  `_delete` CHAR(2) NULL ,
  PRIMARY KEY (`id`, `aro_id`, `aco_id`) ,
  INDEX `fk_aros_has_acos_acos1` (`aco_id` ASC) ,
  INDEX `fk_aros_has_acos_aros1` (`aro_id` ASC) ,
  CONSTRAINT `fk_aros_has_acos_aros1`
    FOREIGN KEY (`aro_id` )
    REFERENCES `aros` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_aros_has_acos_acos1`
    FOREIGN KEY (`aco_id` )
    REFERENCES `acos` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-- -----

-- Table `voluntarios`
-- -----

DROP TABLE IF EXISTS `voluntarios` ;

CREATE TABLE IF NOT EXISTS `voluntarios` (
  `id` INT NOT NULL AUTO_INCREMENT ,

```

```

`nome` VARCHAR(60) NOT NULL ,
`email` VARCHAR(50) NULL ,
`endereco` VARCHAR(100) NULL ,
`numero` VARCHAR(7) NULL ,
`complemento` VARCHAR(7) NULL ,
`bairro` VARCHAR(40) NULL ,
`cidade` VARCHAR(40) NULL ,
`uf` VARCHAR(2) NULL ,
`cep` VARCHAR(9) NULL ,
`telefone` VARCHAR(14) NULL ,
`celular` VARCHAR(14) NULL ,
`prestacao_servico` VARCHAR(20) NOT NULL ,
PRIMARY KEY (`id`) )
ENGINE = InnoDB;

```

```

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

## 5.5 Detalhes Procedimentais

Serão descritos a seguir alguns detalhes procedimentais importantes, que possuem um enfoque de destaque no sistema. Os demais procedimentos estão descritos nas especificações de caso de uso.

### 5.5.1 Login

A primeira tela do sistema deverá ser a tela de *login*, que conterá dois campos (Usuário e Senha), na qual apenas usuários cadastrados e ativos no sistema terão acesso. O sistema também verificará qual o tipo de perfil do usuário para que, após efetuar o *login*, o acesso seja liberado apenas para as áreas que condizerem com o



seu perfil. Confirmada a permissão, o sistema irá exibir a tela principal (Consulta, Famílias), salvando o perfil do usuário logado na sessão.

### 5.5.2 Máscaras e validações

Para facilitar a utilização do sistema pelos usuários, com intuito de evitar erros – principalmente de digitação – na entrada dos dados, serão utilizadas alguns componentes nativos das tecnologias utilizadas para validar a entrada dos dados.

As máscaras funcionam como um marcador automático e inserem dados, muitas vezes padronizados, de forma dinâmica a fim de evitar erros por parte dos usuários. As máscaras serão utilizadas nos seguintes campos: CPF, CEP, Telefone, Celular, Renda, Renda Familiar e Renda Percapta. A máscara para CPF deverá ser no formato: “\_\_\_\_.\_\_\_\_.\_\_\_\_-\_\_”; a máscara para CEP deverá ser no seguinte formato: “\_\_\_\_-\_\_\_\_”; a máscara para Telefone e Celular deverá ser no seguinte formato: “(\_\_\_\_) \_\_\_\_-\_\_\_\_”; a máscara para os campos Renda, Renda Familiar e Renda Percapta deverão estar no seguinte formato: “\_\_\_\_.\_\_\_\_.\_\_\_\_,\_\_\_\_”. Os campos que contiverem DATAS, deverão ser utilizados *select's* para selecioná-las.

Componentes de validação utilizam expressões regulares ou comparação para validar o formato dos dados informados. Os seguintes componentes serão utilizados para validar o *e-mail* e a senha. Os campos Quantidade, Renda, Renda Familiar e Renda Percapta deverão permitir apenas entrada de números.

## 5.6 Observações Complementares

Para se ganhar tempo, flexibilidade, optou-se em criar duas tabelas (questionários e respostas), com intuito de minimizar o trabalho de programação e aumentar o rendimento.

A tabela "questionários" irá cadastrar dinamicamente, pela web, todas as perguntas complementares; as famílias, porém, referente à situação de moradia; a tabela "repostas" conterá as respostas dos usuários que irão utilizar o sistema. A tabela "questionários" foi projetada dessa maneira com intuito de minimizar a

manutenção do sistema e maximizar sua flexibilidade. Necessitando modificar/acrescentar perguntas ao questionário, as perguntas terão um “tipo” para informar se a pergunta a ser cadastrada é do tipo texto, booleano ou *select*.

## 6 PROCESSO DE TESTES

“Por decisão do colegiado de curso, é opcional a apresentação do Processo de Testes neste documento, até a especificação dos procedimentos de teste. No caso deste capítulo não ser apresentado, a execução dos testes e o registro de incidentes que são parte do capítulo 7 deverão se basear no Plano de Testes descrito no capítulo 2 deste documento.”

## 7 IMPLEMENTAÇÃO E TESTES

Neste capítulo serão detalhados os resultados das implementações e dos testes realizados. Será apresentado, também, as camadas implementadas e as especificações referentes à implantação.

### 7.1 Implementação

De acordo com a arquitetura MVC apresentada no capítulo 5.2.2, onde definiu-se que cada item implementado no projeto seria formado por três partes – *model* (ARQUIVO.php), *controller* (ARQUIVO\_controller.php) e *view* (ARQUIVOS.ctp) – responsáveis pelo funcionamento do sistema.

Abaixo segue uma listagem com a descrição / função de cada item implementado:

Item identificador	Descrição / Função
usuario.php	Classe que representa as restrições, regras de negócio que os usuários terão no sistema.

usuarios_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) do usuário, bem como especificações de controle dos usuários (login).
voluntario.php	Classe que representa as restrições, regras de negócio que os voluntários terão no sistema.
voluntarios_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) do voluntário, bem como especificações de controle dos voluntários.
familia.php	Classe que representa as restrições, regras de negócio que as famílias terão no sistema.
familias_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) da família, bem como especificações de controle das famílias.
questionarios.php	Classe que representa as restrições, regras de negócio que os questionários terão no sistema.
questionarios_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) do questionário, bem como especificações de controle dos questionários.
definicoescesta.php	Classe que representa as restrições, regras de negócio, que os mantimentos terão no sistema, bem como as interações com outras classes.
definicoescestas_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) do mantimento, bem como definir os mantimentos que estarão disponíveis para serem cadastrados no estoque.
frequencia.php	Classe que representa as restrições, regras de negócio, que as frequências das famílias terão no sistema, bem como as interações com outras classes.
frequencias_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) da frequência.
cesta.php	Classe que representa as regras de negócio que as cestas terão no sistema.
cestas_controller.php	Classe responsável por toda a lógica de compor a cesta básica. Aqui estão os algoritmos que pegam os alimentos disponíveis no estoque monta a cesta.

mantimento.php	Classe que representa as regras de negócio que os mantimentos terão no sistema.
mantimentos_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) do mantimento.
estoque.php	Classe que representa as regras de negócio que o estoque terá no sistema.
estoques_controller.php	Classe responsável pelo CRUD ( <i>create, read, update and delete</i> ) do estoque.
log.php	Classe responsável por fazer o relacionamento com a classe usuario.php, que irão se interagir para geração dos logs.
logs_controller.php	Classe responsável pela geração de logs no sistema.
admin_cadastrar.ctp admin_editar.ctp admin_index.ctp admin_login.ctp	Arquivos que representam a <i>view</i> do sistema que compõem cada módulo referente a cada controller do sistema, com exceção do <i>admin_login.ctp</i> que é referente apenas para o controlador <i>usuarios_controller.php</i> . São esses arquivos que irão realizar a exibição e interação com o controlador que por sua vez serão exibidos para o usuário.

Quadro 43 – Funcionalidades implementadas

### 7.1.1 Alocação dos Módulos

Segue abaixo uma lista de alocação lacônica de alocação dos módulos – referentes as funções no sistema – aos elementos da arquitetura projetada para o software no capítulo 5 deste documento.

Elemento Arquitetural	Módulo
Controle de Usuário	usuarios_controller.php
Controle de Voluntário	voluntarios_controller.php
Controle de Família	familias_controller.php

Controle de Questionário Família	questionarios_controller.php
Controle de Definição de Cesta	definicoescestas_controller.php
Controle de Frequência Família	frequencias_controller.php
Controle de Cesta	cestas_controller.php
Controle de Mantimento	mantimentos_controller.php
Controle de Estoque	estoques_controller.php
Controle de Log	logs_controller.php

Quadro 44 – Programas desenvolvidos

### 7.1.2 Considerações Sobre a Implantação

Na construção do sistema pode-se observar que um dos módulos, *plugin*, mais complexo de ser implementado foi o ACL (Lista de Controle de Acesso) o qual utiliza o algoritmo *Tree* que trabalha com a lógica *Tree* em níveis de profundidade infinitos. Para a sua implantação, foi utilizado o *book* do site *cakephp.org*, que ensina passo a passo como implementar o *plugin*, porém há certas complexidades para adequá-lo no sistema, como algumas regras que devem ser seguidas. Por exemplo a criação de algumas tabelas no banco para representar os componentes específicos do *plugin*.

Outro módulo que teve, de certa forma uma complexidade tão grande ou até maior que o ACL foi o módulo referente ao questionário das famílias. O que inicialmente seria um grande trabalho braçal de estar criando várias perguntas no sistema, referente a ficha de cadastro das famílias, acabou por se tornar um grande esforço intelectual e muito pouco braçal. Devido à existência de muitas perguntas, foi adotado uma lógica para sanar tal problema. A estratégia utilizada foi criar uma regra onde o usuário ,administrador, possa cadastrar as perguntas diretamente dentro do sistema – sem que precise mexer em uma linha de comando – e escolhe o tipo de resposta que deseja, a conter, também, um campo opcional para inserir as possíveis perguntas e escolher o tipo de resposta que desejar como: uma pergunta

contendo conteúdo apenas textual, de seleção, de múltiplas respostas ou booleana.

A questão de ser complexa ou crítica pode estar a caminhar em conjunto e foi o que ocorreu com o módulo de criação da cesta básica, sendo o mais crítico, pois precisa separar corretamente os alimentos que são cadastrados e enviados para o estoque, de modo juntar os mantimentos que compõem a cesta básica e assim, disponibilizá-las para as famílias aptas. Este módulo específico é o que faz a comunicação com das famílias com os alimentos que as mesmas irão receber, torando o seu funcionamento correto, algo crucial para o sistema. Diversos problemas foram encontrados e foi-se resolvendo um por um e efetuando testes para verificar se o mesmo estava a funcionar corretamente. A complicação maior que se teve foi de entender como implementar a lógica de mantimentos “quebrados”, ou seja, como o sistema deve entender que precisa, por exemplo, pegar 1kg de arroz de um cadastro e mais dois pacotes de 2kg de arroz de outros cadastros para compor corretamente a cesta. O problema foi resolvido com muito pensar e muita pesquisa. O resultado pode ser conferido no *controller* `cestas_controller.php`.

### 7.1.3 Outros Aspectos da Implementação

O sistema possui um registro de frequência das famílias, onde é registrada a presença de um representante de cada família que deverá comparecer as reuniões que acontecem todos os sábados, e caso hajam 3 faltas, sem justificativa, a família é considerada não apta para recebimento da cesta básica. Essa implementação foi uma parte interessante, pois com uma simples “chamada” de presença leva uma família a estar apta ou não para o recebimento da cesta básica.

## 7.2 Realização dos Testes

Baseando-se no Plano de Testes do capítulo 2, item 2.19, serão apresentados os incidentes dos testes realizados para as implementações feitas, separados por caso de uso.

Serão relatados os incidentes de teste e apresentado o seu resumo.

### *7.2.1 Registro de incidente de testes*

#### *7.2.1.1 Caso de uso UC01 – Cadastrar Usuário*

No primeiro teste realizado, o campo "Situação" encontrava-se com a opção "Ativo" marcado, estava salvando o usuário como "Inativo". Houve uma inversão no *array*. No segundo teste foi bem sucedido.

#### *7.2.1.2 Caso de uso UC02 – Alterar Usuário*

Devido a opção de "Inativar" usuário está diretamente ligada a alteração de usuário, o caso de uso "Inativar Usuário" foi concatenado com o caso de uso "Inativar Usuário".

Para o teste de alteração do usuário foi efetuado uma simulação de alteração com e sem inserção de dados e durante os testes não foi observado nenhum incidente.

#### *7.2.1.3 Caso de uso UC03 – Excluir Usuário*

Para o teste de exclusão do usuário foi efetuado uma simulação de exclusão de alguns usuários cadastrados e durante os testes não foi observado nenhum incidente.

#### *7.2.1.4 Caso de uso UC04 – Consulta Usuário*

Para o teste de consulta do usuário foi efetuado uma simulação de consulta dos usuários cadastrados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.5 Caso de uso UC05 – Incluir Estoque

Para o teste de inclusão no estoque foi efetuado uma simulação de inclusão de alguns mantimentos e durante os testes não foi observado nenhum incidente.

#### 7.2.1.6 Caso de uso UC06 – Alterar Estoque

Para o teste de alteração no estoque foi efetuado uma simulação de edição dos mantimentos cadastrados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.7 Caso de uso UC07 – Excluir Estoque

Para o teste de exclusão no estoque foi efetuado uma simulação de exclusão dos mantimentos cadastrados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.8 Caso de uso UC08 – Consultar estoque

Para o teste de consulta ao estoque foi efetuado uma simulação de consulta dos mantimentos cadastrados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.9 Caso de uso UC09 – Relatório Vencimentos Estoque

Para o teste de emissão de relatório de estoque foi efetuado uma simulação de emissão de relatório do estoque de alimentos e durante os testes não foi observado nenhum incidente.



#### 7.2.1.10 Caso de uso UC10 – Relatório cestas disponíveis

Para o teste de emissão de relatório de cestas disponíveis foi efetuado uma simulação de emissão de relatório das cestas disponíveis para as famílias aptas e durante os testes não foi observado nenhum incidente.

#### 7.2.1.11 Caso de uso UC11 – Relatório itens pendentes das cestas

Para o teste de emissão de relatório de itens pendentes nas cestas foi efetuado uma simulação de emissão de relatório das cestas que não estavam completas com todos os mantimentos e durante os testes verificou-se que o resultado não estava coerente, mas descobriu-se que era um erro de cálculo e o após corrigir o problema, o erro desapareceu.

#### 7.2.1.12 Caso de uso UC12 – Cadastrar Definição de Cesta

Para o teste de cadastro de definição de cesta foi efetuado uma simulação de cadastro de diversos mantimentos que compõe uma cesta e durante os testes não foi observado nenhum incidente.

#### 7.2.1.13 Caso de uso UC13 – Alterar Definição de Cesta

Para o teste de alteração de definição de cesta foi efetuado uma simulação de alteração dos mantimentos cadastrados que compõe uma cesta e durante os testes não foi observado nenhum incidente.

#### 7.2.1.14 Caso de uso UC14 – Excluir Definição de Cesta

Para o teste de exclusão de definição de cesta foi efetuado uma simulação de

exclusão dos mantimentos cadastrados que compõe uma cesta e durante os testes não foi observado nenhum incidente.

#### 7.2.1.15 Caso de uso UC15 – Consultar Definição de Cesta

Para o teste de consulta a definição da cesta foi efetuado uma simulação de consulta dos mantimentos cadastrados que definem a cesta básica e durante os testes não foi observado nenhum incidente.

#### 7.2.1.16 Caso de uso UC16 – Cadastrar Família

Talvez esse seja o caso de uso com mais erros encontrados. Os seguintes erros foram encontrados e foram corrigidos assim que descobertos: O campo escolaridade e parentesco não estavam visíveis; o campo data de nascimento não estava gravando a data no banco – teve-se que realizar uma inversão na data, para o padrão americano, para que conseguisse gravar no banco corretamente –; os campos de renda, renda per capita e renda familiar, devido a colocação da mascara, não estavam exibindo os valores corretamente.

#### 7.2.1.17 Caso de uso UC17 – Cadastrar Frequência Família

Para o teste de cadastro de frequência da família foi efetuado uma simulação de cadastro de algumas famílias aptas e durante os testes não foi observado nenhum incidente.

#### 7.2.1.18 Caso de uso UC18 – Alterar Família

Foram encontrados alguns problemas referentes a parte de alteração de

famílias. O campo telefone desaparecendo com os números após realizar uma edição e descobriu-se que era falta de caractere no banco de dados; os campos referentes a renda estavam gravando números de forma incompleto e verificou-se que era erro de configuração da mascara. Todos esses problemas foram corrigidos assim que descobertos e novos testes foram realizados, não havendo problemas.

#### 7.2.1.19 Caso de uso UC19 – Alterar Frequência Família

Para o teste de alteração de frequência das famílias foi efetuado uma simulação de alteração nas frequências das famílias cadastradas e ativas, e durante os testes não foi observado nenhum incidente.

#### 7.2.1.20 Caso de uso UC20 – Excluir Família

Para o teste de exclusão de família foi efetuado uma simulação de exclusão de algumas famílias cadastradas e durante os testes não foi observado nenhum incidente.

#### 7.2.1.21 Caso de uso UC21 – Consulta Família

Para o teste de consulta a família foi efetuado uma simulação de consulta de algumas famílias cadastradas e durante os testes não foi observado nenhum incidente.

#### 7.2.1.22 Caso de uso UC22 – Consulta Frequência Família

Para o teste de consulta a frequência da família foi efetuado uma simulação de consulta de algumas famílias ativas e com frequências cadastradas e durante os

testes não foi observado nenhum incidente.

#### 7.2.1.23 Caso de uso UC23 – Relatório Vencimento Matrícula

Para o teste de emissão de relatório de matrícula foi efetuado uma simulação de emissão de relatório das famílias cadastradas e ativas no sistema por mais de um ano, para isso trocou-se a data de algumas famílias para que a simulação dos relatórios pudessem ser realizados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.24 Caso de uso UC24 – Relatório Famílias aptas

Para o teste de emissão de relatório de família apta foi efetuado uma simulação de emissão de relatório das famílias cadastradas no sistema e durante os testes não foi observado nenhum incidente.

#### 7.2.1.25 Caso de uso UC25 – Efetuar Login

Para o teste de efetuar login, foi efetuado uma simulação de logar com usuários cadastrados e ativos, cadastrados inativos, não cadastrados e ou com senhas incorretas e durante os testes não foi observado nenhum incidente.

#### 7.2.1.26 Caso de uso UC26 – Recuperar Senha

Para o teste de efetuar recuperação de senha, foi efetuado uma simulação de tentar logar com usuários cadastrados, porém com a senha errada e foi utilizado o link para recuperação, onde a senha atual foi resetada e enviada para o e-mail do usuário uma nova senha, e durante os testes não foi observado nenhum incidente.

#### 7.2.1.27 Caso de uso UC27 – Armazenar log

Para o teste de armazenamento de *log* foi efetuado uma simulação de monitoração do sistema e durante os testes não foi observado nenhum incidente.

#### 7.2.1.28 Caso de uso UC28 – Consultar log do Sistema

Para o teste de consulta de *log* foi efetuado uma simulação de consultas referentes as alterações ocorridas no sistema e durante os testes não foi observado nenhum incidente.

#### 7.2.1.29 Caso de uso UC29 – Cadastrar Voluntário

Para o teste de cadastro de voluntário foi efetuado uma simulação de cadastro com e sem inserção de dados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.30 Caso de uso UC30 – Alterar Voluntário

Para o teste de alterar voluntário foi efetuado uma simulação de alteração com e sem inserção de dados e durante os testes não foi observado nenhum incidente.

#### 7.2.1.31 Caso de uso UC31 – Excluir Voluntário

Para o teste de exclusão de voluntário foi efetuado uma simulação de exclusão de alguns cadastros e durante os testes não foi observado nenhum

incidente.

#### 7.2.1.32 Caso de uso UC32 – Consultar Voluntário

Para o teste de consulta de voluntário foi efetuado uma simulação de consulta de alguns cadastros e durante os testes não foi observado nenhum incidente.

#### 7.2.1.33 Caso de uso UC33 – Relatório Voluntário

Para o teste de relatório de voluntário foi efetuado uma simulação de extração de relatório dos voluntários cadastros e durante os testes não foi observado nenhum incidente.

### *7.2.2 Resumo dos Testes*

Nas presentes datas, todos os casos de uso estavam implementados de tal forma a permitir que os testes fossem realizados, uma vez que as funcionalidades básicas de cada um já haviam sido programadas

<b>Caso de uso</b>	<b>Data execução</b>	<b>Resultado</b>	<b>Reexecução</b>	<b>Resultado</b>
UC01	27/08/2011	Falhou	29/08/2011	Passou
UC02	27/08/2011	Falhou	29/08/2011	Passou
UC03	27/08/2011	Passou	30/09/2011	Passou
UC04	27/08/2011	Passou	30/09/2011	Passou
UC05	25/09/2011	Passou	30/09/2011	Passou
UC06	25/09/2011	Passou	30/09/2011	Passou
UC07	25/09/2011	Passou	30/09/2011	Passou
UC08	27/08/2011	Passou	30/09/2011	Passou
UC09	25/09/2011	Passou	30/09/2011	Passou

UC10	25/09/2011	Passou	30/09/2011	Passou
UC11	25/09/2011	Falhou	30/09/2011	Passou
UC12	27/08/2011	Passou	30/09/2011	Passou
UC13	27/08/2011	Passou	30/09/2011	Passou
UC14	27/08/2011	Passou	30/09/2011	Passou
UC15	25/09/2011	Passou	30/09/2011	Passou
UC16	25/09/2011	Falhou	30/09/2011	Passou
UC17	25/09/2011	Passou	30/09/2011	Passou
UC18	25/09/2011	Falhou	30/09/2011	Passou
UC19	25/09/2011	Passou	30/09/2011	Passou
UC20	25/09/2011	Passou	30/09/2011	Passou
UC21	25/09/2011	Passou	30/09/2011	Passou
UC22	25/09/2011	Passou	30/09/2011	Passou
UC23	25/09/2011	Passou	30/09/2011	Passou
UC24	25/09/2011	Passou	30/09/2011	Passou
UC25	27/08/2011	Passou	30/09/2011	Passou
UC26	25/09/2011	Passou	30/09/2011	Passou
UC27	27/08/2011	Passou	30/09/2011	Passou
UC28	27/08/2011	Passou	30/09/2011	Passou
UC29	25/09/2011	Passou	30/09/2011	Passou
UC30	25/09/2011	Passou	30/09/2011	Passou
UC31	25/09/2011	Passou	30/09/2011	Passou
UC32	25/09/2011	Passou	30/09/2011	Passou
UC33	25/09/2011	Passou	30/09/2011	Passou

Quadro XX – Resumo dos testes

### 7.2.3 Considerações Sobre a Execução dos Testes

Como definido no capítulo 2, optou-se em utilizar o procedimento de teste “caixa preta”, o qual pode-se encontrar vários problemas durante a execução, sendo que grande partes foram problemas bem simples como até erro de digitação de

variáveis.

Cada parte do sistema foi testado e retestado diversas vezes com o intuito de minimizar a aparição de erros ou falhas para o usuário final. A parte de cadastro, exclusão e alteração de famílias contem um grande número de perguntas que estão interligadas, porem alguns processos de implementação foram construídos separadamente, como é o caso do cadastro do questionário que é um módulo à parte que foi acoplado na *view* da família. Muitos testes foram executados em cima desse processo para que estivesse refinado para utilização.

Outro ponto muito importante e complexo do sistema foi a parte de permissões o qual foi utilizado um *plugin* de terceiros para estar executando tal tarefa. Vários testes de retirar e incluir permissões foram executadas.

O módulo para compor a cesta básica foi o que deu mais trabalho para implementar como para testar, pois teve-se que criar vários registros distintos com quantidade de alimentos diferenciadas para comprovar que a cesta seria composta mesmo com registros “picados”, ou seja, por exemplo: montar uma cesta com 5kg de arroz, pegando 1kg de cada registro que foi cadastrado anteriormente. Todo esse processo é executado de forma automatizada.

#### 7.2.4 Outros Aspectos do Processo de Testes

### ~~7.4 Observações Complementares~~

~~— Foi utilizado um componente do *CakePHP* denominado *Behaviors* o qual possui a finalidade de adicionar funcionalidades extras aos *models*. Dessa forma traz agilidade e flexibilidade no desenvolvimento da aplicação.~~

## 8. DOCUMENTAÇÃO



### **8.1 Considerações Preliminares**

### **8.2 Manual de Instalação**

### **8.3 Manual de Usuário**

### **8.4 Outros Documentos**

## **9 IMPLEMENTAÇÃO E PRÉ-OPERAÇÃO**

### **9.1 Considerações Preliminares**

### **9.2 A Implantação**

### **9.3 Pré-operação**

## 10 MONITORAMENTO E CONTROLE

### 10.1 Considerações Preliminares

O monitoramento e controle defini-se em trilhar um caminho para que o projeto tenha condições de ser acompanhado. Caso o projeto perca as definições propostas, definidas, no plano de negócio (capítulo 2), há um controle que sugere uma correção para que o projeto não se distancie do que foi proposto e consiga alcançar o seu fechamento das datas, custos e riscos que foram previstos.

### 10.2 Primeiro monitoramento e controle

Marco – Final da fase da Especificação de Requisitos

#### 10.2.1 Prazo

De acordo com a elaboração do cronograma realizado no capítulo 2, item 2.8, o planejado foi executado até a data final proposta, não havendo necessidade de ajustes de projeto ou cronograma. Seguem as datas no quadro:

Prazos					
	Previsto		Realizado		
Atividade	Início	Término	Início	Término	Diferença
Levantamento preliminar de requisitos	14/02/2011	14/02/2011	14/02/2011	14/02/2011	Em dia
Especificação de requisitos	15/02/2011	16/02/2011	15/02/2011	16/02/2011	Em dia
Escopo	17/02/2011	17/02/2011	17/02/2011	17/02/2011	Em dia
Plano processo de desenvolvimento	18/02/2011	18/02/2011	18/02/2011	18/02/2011	Em dia
Estrutura analítica	21/02/2011	21/02/2011	21/02/2011	21/02/2011	Em dia

Estimativas	22/02/2011	23/02/2011	22/02/2011	23/02/2011	Em dia
Plano organização	24/02/2011	24/02/2011	24/02/2011	24/02/2011	Em dia
Monitoramento e controle	25/02/2011	25/02/2011	25/02/2011	25/02/2011	Em dia
Cronograma	28/02/2011	02/03/2011	28/02/2011	02/03/2011	Em dia
Recursos humanos	03/03/2011	03/03/2011	03/03/2011	03/03/2011	Em dia
Recursos gerais	04/03/2011	04/03/2011	04/03/2011	04/03/2011	Em dia
Plano de custo	07/03/2011	08/03/2011	07/03/2011	08/03/2011	Em dia
Plano de teste	09/03/2011	09/03/2011	09/03/2011	09/03/2011	Em dia
Plano de treinamento	10/03/2011	11/03/2011	10/03/2011	11/03/2011	Em dia
Plano de implantação	14/03/2011	15/03/2011	14/03/2011	15/03/2011	Em dia
Especificação de requisitos	16/03/2011	18/03/2011	16/03/2011	23/04/2011	Atraso de 35 dias

Quadro 45 – Prazo do Primeiro Monitoramento e Controle

### 10.2.2 Produção

Por decisão do Colegiado do curso, a apresentação do monitoramento e controle de produção é opcional e por este motivo não será elaborado neste projeto.

### 10.2.3 Custos

Os valores contidos abaixo são resultado da efetivação de um cálculo dos gastos ocorridos até o momento, ou seja, no primeiro marco do primeiro monitoramento e controle.

Recurso	Previsto até 18/03/11 em R\$	Realizado até 23/ 03/11 em R\$	Observações
Software	16,50	16,50	Não houve alterações pois o valor do item se mantém a partir da data início
Hardware	132,50	132,50	Não houve alterações pois o

			valor do item se mantem a partir da data início
Mão de Obra	560,00	780,00	Aumento do resultado obtido do cálculo de 2 Gerentes e 2 Analistas
Despesas Gerais	401,94	474,12	Aumento do resultado obtido pelo aumento dos meses trabalhados
<b>TOTAL</b>	<b>R\$ 1.110,94</b>	<b>R\$ 1.403,12</b>	

Quadro 46 – Custo do Primeiro Monitoramento e Controle

#### 10.2.4 Riscos

Por decisão do Colegiado do curso, a apresentação do monitoramento e controle de riscos é opcional e por este motivo não será elaborado neste projeto.

#### 10.2.5 Fechamento do Primeiro Monitoramento e Controle

Apesar de não conseguir um fechamento correspondente ao primeiro cronograma constituído, foi possível perceber que houve um erro, humano, de digitação quando foi composto o primeiro cronograma. Apesar do ocorrido, o erro foi verificado e corrigido e enquadrado no novo cronograma, agora, corrigido e adequado para seguir com o projeto.

### 10.3 Segundo monitoramento e controle

Marco – Final da fase de Análise e Modelagem

#### 10.3.1 Prazo

De acordo com a elaboração do cronograma realizado no capítulo 2, item 2.8, o planejado foi executado até a data final proposta, não havendo necessidade de

ajustes de projeto ou cronograma. Seguem as datas no quadro:

Prazos					
	Previsto		Realizado		
Atividade	Início	Término	Início	Término	Diferença
Levantamento preliminar de requisitos	14/02/2011	14/02/2011	14/02/2011	14/02/2011	Em dia
Especificação de requisitos	15/02/2011	16/02/2011	15/02/2011	16/02/2011	Em dia
Escopo	17/02/2011	17/02/2011	17/02/2011	17/02/2011	Em dia
Plano processo de desenvolvimento	18/02/2011	18/02/2011	18/02/2011	18/02/2011	Em dia
Estrutura analítica	21/02/2011	21/02/2011	21/02/2011	21/02/2011	Em dia
Estimativas	22/02/2011	23/02/2011	22/02/2011	23/02/2011	Em dia
Plano organização	24/02/2011	24/02/2011	24/02/2011	24/02/2011	Em dia
Monitoramento e controle	25/02/2011	25/02/2011	25/02/2011	25/02/2011	Em dia
Cronograma	28/02/2011	02/03/2011	28/02/2011	02/03/2011	Em dia
Recursos humanos	03/03/2011	03/03/2011	03/03/2011	03/03/2011	Em dia
Recursos gerais	04/03/2011	04/03/2011	04/03/2011	04/03/2011	Em dia
Plano de custo	07/03/2011	08/03/2011	07/03/2011	08/03/2011	Em dia
Plano de teste	09/03/2011	09/03/2011	09/03/2011	09/03/2011	Em dia
Plano de treinamento	10/03/2011	11/03/2011	10/03/2011	11/03/2011	Em dia
Plano de implantação	14/03/2011	15/03/2011	14/03/2011	15/03/2011	Em dia
Especificação de requisitos	16/03/2011	18/03/2011	16/03/2011	23/04/2011	Atraso de 35 dias
Diagrama de classe	23/04/2011	03/06/2011	16/05/2011	27/05/2011	Em dia
Diagrama de sequência	23/04/2011	03/06/2011	03/05/2011	20/05/2011	Em dia
DTR	23/04/2011	03/06/2011	09/05/2011	15/05/2011	Em dia

Quadro 47 – Prazo do Segundo Monitoramento e Controle

### 10.3.2 Produção

Por decisão do Colegiado do curso, a apresentação do monitoramento e controle de produção é opcional e por este motivo não será elaborado neste projeto.

### 10.3.3 Custos

Os valores contidos abaixo são resultado da efetivação de um cálculo dos gastos ocorridos até o momento, ou seja, no segundo marco do segundo monitoramento e controle.

Recurso	Previsto até 03/06/11 em R\$	Realizado até 15/ 05/11 em R\$	Observações
Software	16,50	16,50	Não houve alterações pois o valor do item se mantém a partir da data início
Hardware	132,50	132,50	Não houve alterações pois o valor do item se mantém a partir da data início
Mão de Obra	3.060,00	1.800,00	Redução do resultado obtido do cálculo de 2 Gerentes e 2 Analistas
Despesas Gerais	1.867,59	1.109,70	Redução do resultado obtido pela diminuição nos meses trabalhados
<b>TOTAL</b>	<b>R\$ 5.076,59</b>	<b>R\$ 3.058,70</b>	

Quadro 48 – Custo do Segundo Monitoramento e Controle

### 10.3.4 Riscos

Por decisão do Colegiado do curso, a apresentação do monitoramento e controle de riscos é opcional e por este motivo não será elaborado neste projeto.

### 10.3.5 Fechamento do Segundo Monitoramento e Controle

Diferente do primeiro monitoramento e controle, conseguimos adiantar consideravelmente os trabalhos propostos para o segundo monitoramento e controle, a terminar todas as atividades dentro do prazo estabelecido. Conseguindo terminar uma semana antes do previsto.

## 10.4 Terceiro monitoramento e controle

Marco – Final da fase de teste

### 10.4.1 Prazo

De acordo com a elaboração do cronograma realizado no capítulo 2, item 2.8, o planejado foi executado até a data final proposta, não havendo necessidade de ajustes de projeto ou cronograma. Seguem as datas no quadro:

Prazos					
	Previsto		Realizado		
Atividade	Início	Término	Início	Término	Diferença
Levantamento preliminar de requisitos	14/02/2011	14/02/2011	14/02/2011	14/02/2011	Em dia
Especificação de requisitos	15/02/2011	16/02/2011	15/02/2011	16/02/2011	Em dia
Escopo	17/02/2011	17/02/2011	17/02/2011	17/02/2011	Em dia
Plano processo de desenvolvimento	18/02/2011	18/02/2011	18/02/2011	18/02/2011	Em dia
Estrutura analítica	21/02/2011	21/02/2011	21/02/2011	21/02/2011	Em dia
Estimativas	22/02/2011	23/02/2011	22/02/2011	23/02/2011	Em dia
Plano organização	24/02/2011	24/02/2011	24/02/2011	24/02/2011	Em dia
Monitoramento e controle	25/02/2011	25/02/2011	25/02/2011	25/02/2011	Em dia

Cronograma	28/02/2011	02/03/2011	28/02/2011	02/03/2011	Em dia
Recursos humanos	03/03/2011	03/03/2011	03/03/2011	03/03/2011	Em dia
Recursos gerais	04/03/2011	04/03/2011	04/03/2011	04/03/2011	Em dia
Plano de custo	07/03/2011	08/03/2011	07/03/2011	08/03/2011	Em dia
Plano de teste	09/03/2011	09/03/2011	09/03/2011	09/03/2011	Em dia
Plano de treinamento	10/03/2011	11/03/2011	10/03/2011	11/03/2011	Em dia
Plano de implantação	14/03/2011	15/03/2011	14/03/2011	15/03/2011	Em dia
Especificação de requisitos	16/03/2011	18/03/2011	16/03/2011	23/04/2011	Atraso de 35 dias
Diagrama de classe	23/04/2011	03/06/2011	16/05/2011	27/05/2011	Em dia
Diagrama de sequência	23/04/2011	03/06/2011	03/05/2011	20/05/2011	Em dia
DTR	23/04/2011	03/06/2011	09/05/2011	15/05/2011	Em dia
Desenvolvimento do Sistema	06/06/2011	30/11/2011	28/05/2011	30/08/2011	Em dia
Testes preliminares	16/06/2011	13/09/2011	10/06/2011	30/08/2011	Adiantado

Quadro 49 – Prazo do Terceiro Monitoramento e Controle

#### 10.4.2 Produção

Por decisão do Colegiado do curso, a apresentação do monitoramento e controle de produção é opcional e por este motivo não será elaborado neste projeto.

#### 10.4.3 Custos

Os valores contidos abaixo são resultado da efetivação de um cálculo dos gastos ocorridos até o momento, ou seja, no terceiro marco do monitoramento e controle.

Recurso	Previsto até 13/09/11 em R\$	Realizado até 30/ 08/11 em R\$	Observações
Software	16,50	16,50	Não houve alterações pois o valor



			do item se mantem a partir da data início
Hardware	132,50	132,50	Não houve alterações pois o valor do item se mantem a partir da data início
Mão de Obra	10223,52	8227,27	Redução do resultado obtido do cálculo de 2 Gerentes e 2 Analistas
Despesas Gerais	3504,51	3271,99	Redução do resultado obtido pela diminuição nos meses trabalhados
<b>TOTAL</b>	<b>13877,03</b>	<b>11648,26</b>	

Quadro 50 – Custo do Terceiro Monitoramento e Controle

#### 10.4.4 Riscos

Por decisão do Colegiado do curso, a apresentação do monitoramento e controle de riscos é opcional e por este motivo não será elaborado neste projeto.

#### 10.4.5 Fechamento do Terceiro Monitoramento e Controle

Conseguimos finalizar os trabalhos propostos para o terceiro monitoramento e controle terminando todas as atividades dentro do prazo estabelecido.

## 11 CONSIDERAÇÕES FINAIS

Realizou-se, como registrado está no Capítulo 1, uma contextualização, na qual há a descrição do cliente em si, como também o recolhimento de alguns dados iniciais que proporcionaram, assim, o início do projeto. Essas informações foram essenciais para a identificação das necessidades do cliente e a consequente construção deste projeto.

Estabeleceu-se, no Capítulo 2, procurando detalhar todo o plano de negócios que deverá ser seguido, a concretização e a realização do sistema. Esse capítulo, vale dizer, é um dos mais importantes para a construção do projeto, pois nele estão todos os recursos descritos e, ainda, tudo que se baseia para que se possa dar continuidade aos procedimentos ali canalizados: detalhando, por meio de gráficos, procedimentos que serão seguidos no decorrer do projeto e que receberão continuamente revisão e análise pela equipe de desenvolvimento.

Definiu-se, no capítulo 3, uma base mais esclarecedora de como o projeto deverá se comportar para atingir sucesso em sua finalização. Dessa forma, obteve-se uma maturidade maior com relação aos requisitos recolhidos que compõem o projeto. Se comparado ao capítulo 2, que definiu todo o plano de negócios a ser seguido, o capítulo 3 definiu todos os requisitos que irão compor o sistema.

Pode-se notar que, com a conclusão do capítulo 4, de monitoramento e controle, foi possível identificar possíveis erros cometidos durante o projeto, além de ser corrigida a diferença de dias, que inicialmente foi expressa de maneira equivocada; com as novas adaptações. No entanto, a continuidade do projeto seguiu em frente.

Concluiu-se a primeira parte do projeto (Projetos 1) que é referente ao primeiro semestre. Nessa última, houve a criação do diagrama de tabelas relacionais, diagrama de classe e o diagrama de sequência o qual foi definido como o mais trabalhoso e com um nível de dificuldade semelhante ao diagrama de classes; nesse último, separou-se um tempo específico para analisar e definir como deveriam ser criadas e relacionadas as classes, principalmente a classe família – a maior classe do sistema – que precisou ser dividida para uma melhor compreensão e entendimento. Conseguiu-se concluir com sucesso todos os quesitos que nos foram destinados; terminou-se, assim, tudo uma semana antes do previsto no segundo monitoramento e controle.

Ao término dos capítulos Modelagem de Projeto e Preliminares da Implantação, pode-se notar que o projeto, agora, iniciou-se em uma nova fase de grau mais baixo, dando início à construção do *software*. Nessa etapa conclui-se toda a estrutura do banco de dados que será implementada ao *software*, bem como o início da programação, na qual foram implementados e testados alguns casos de

uso que fora definido anteriormente no capítulo 3. Os capítulos que agora começaram a ser desenvolvidos, são um reflexo de toda a modelagem de análise que foi implementada na primeira parte do projeto e que agora encontra-se no seu desenvolvimento prático.

## BIBLIOGRAFIA

FACHIN, Odília. **Fundamentos de metodologia**. 4.ed. São Paulo: Saraiva, 2003.

PRESSMAN, S. R.; **Engenharia de software**. 6. ed. São Paulo. McGraw-Hill.2006.

LONGSTREET, D.; **Function Point Training and Analysis Course**. Lumberton, 2008. Disponível em: <<http://www.softwaremetrics.com/Function%20Point%20Training%20Booklet%20New.pdf>>, Acesso em: 7 de mar. de 2011.

PROJEC MANAGEMENT INSTITUTE, INC. **Guia PMBOK**. 3 ed. Four Campus Boulevard. Newtown Square, Pennsylvania. 2004.

TONSIG, S. L. **Engenharia de Software**. São Paulo. Futura, 2003.

LARMAN, CRAIG. **Utilizando UML e Padrões - Uma introdução à análise e aos projeto orientados a objetos e ao desenvolvimento iterativo**. São Paulo. Bookman, 2007.

<http://book.cakephp.org/pt>, Acesso em 21 de agosto de 2011.

## Anexo I – Tabelas Relativas ao FPA

Tabela 1 – Complexidade de Entrada			
Campos(TD) Arquivos(AR)	1 a 4 itens de arquivos referenciados	5 a 15 itens de arquivos referenciados	16 ou mais itens de arquivos referenciados
0 ou 1 tipo de arquivos referenciado	Simples	Simples	Médio
2 tipos de arquivos referenciados	Simples	Médio	Complexo
3 ou mais tipos de arquivos referenciados	Médio	Complexo	Complexo

Quadro 51 – Tabela de Complexidade de Entrada

Tabela 2 – Complexidade de Saída			
Campos(TD) Arquivos(AR)	1 a 5 itens de arquivos referenciados	6 a 19 itens de arquivos referenciados	10 ou mais itens de arquivos referenciados
0 ou 1 tipo de arquivos referenciado	Simples	Simples	Médio
2 ou 3 tipos de arquivos referenciados	Simples	Médio	Complexo
4 ou mais tipos de arquivos referenciados	Médio	Complexo	Complexo

Quadro 52 – Tabela de Complexidade de Saída

Tabela 3 – Complexidade ALI			
Campos(TD) Registros(TR)	1 a 19 itens de arquivos referenciados	20 a 50 itens de arquivos referenciados	51 ou mais itens de arquivos referenciados
1 tipo de registro lógico	Simples	Simples	Médio
2 a 5 tipos de	Simples	Médio	Complexo

registros lógicos			
6 ou mais tipos de registros lógicos	Médio	Complexo	Complexo

Quadro 53 – Tabela de Complexidade ALI

Tabela 4 – Complexidade AIE			
Campos(TD) Registros(TR)	1 a 19 itens de arquivos referenciados	20 a 50 itens de arquivos referenciados	51 ou mais itens de arquivos referenciados
1 tipo de registro lógico	Simple	Simple	Médio
2 a 5 tipos de registros lógicos	Simple	Médio	Complexo
6 ou mais tipos de registros lógicos	Médio	Complexo	Complexo

Quadro 54 – Tabela de Complexidade AIE

Tabela 5 – Complexidade Consulta			
Campos(TD) Arquivos(AR)	1 a 5 itens de arquivos referenciados	6 a 19 itens de arquivos referenciados	20 ou mais itens de arquivos referenciados
0 ou 1 tipo de arquivos referenciado	Simple	Simple	Médio
2 ou 3 tipos de arquivos referenciados	Simple	Médio	Complexo
4 ou mais tipos de arquivos referenciados	Médio	Complexo	Complexo

Quadro 55 – Tabela de Complexidade de Consulta

Tabela 6 – Tabela de Pesos FPA				
Função	NºOcorrências	Complexidade	Peso	Resultado
Entrada Externa		Simple	x 3	=
		Médio	x 4	=
		Complexo	x 6	=

TOTAL 1 =			
Saída Externa	Simples	x 4	=
	Médio	x 5	=
	Complexo	x 7	=
TOTAL 2 =			
ALI	Simples	x 7	=
	Médio	x 10	=
	Complexo	x 15	=
TOTAL 3 =			
AIE	Simples	x 5	=
	Médio	x 7	=
	Complexo	x 10	=
TOTAL 4 =			
Consultas	Simples	x 3	=
	Médio	x 4	=
	Complexo	x 6	=
TOTAL 5 =			

Quadro 56 – Tabela de pesos para FPA