

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA DE SÃO PAULO  
CÂMPUS BIRIGUI

NOME COMPLETO

TÍTULO:

subtítulo (se houver)

BIRIGUI

ANO

NOME COMPLETO

Título:

subtítulo (se houver)

Trabalho de Conclusão de Curso  
apresentado ao Instituto Federal de  
Educação, Ciência e Tecnologia de  
São Paulo, como requisito parcial para  
conclusão do curso de Engenharia da  
Computação.

Área de Concentração: Área de  
Concentração do Trabalho

Orientador: Prof./Profa. Me./Dr./Dra.  
Nome Completo

BIRIGUI

ANO

**FICHA CATALOGRÁFICA (obrigatória; impressa no verso da folha de rosto, não conta na paginação)**

Rissi, Viviane Renata Ventura.  
A importância da biblioteca : análise dos  
últimos 7 anos / Viviane R. V. Rissi, 2019  
90 f. : il.

Orientador: Nome direto

Monografia (Graduação)- Instituto Federal de  
Educação, Ciência e Tecnologia de São Paulo,  
Birigui, 2019.

1. Leitura (Ensino superior). 2. Livros  
(Investimento). 3. Leitura - Meios auxiliares.  
I. Instituto Federal de Educação, Ciência e  
Tecnologia. II. Título.

# ERRATA

ERRATA (OPCIONAL). Aponta erros ortográficos contidos no trabalho, que não foram corrigidos antes da impressão, da seguinte forma: constrói-se uma pequena tabela composta pelas colunas “Folha”, “Linha”, “Onde se lê”, e “Leia-se” e aponta-se nessas colunas a localização do erro, o erro propriamente dito e sua respectiva correção. Encabeçando a errata, deve-se especificar a referência do trabalho.

## **FOLHA DE APROVAÇÃO (obrigatório)**

NOME DO AUTOR DO TRABALHO

(Caso seja mais de um aluno, colocar em ordem alfabética)

**TÍTULO DO TRABALHO: (TAMANHO 14)**

Subtítulo do Trabalho, **se Houver (tamanho 12)**

Trabalho de Conclusão de Curso  
apresentado ao Instituto Federal de  
Educação, Ciência e Tecnologia de São  
Paulo, Câmpus Birigui, como requisito  
parcial para conclusão do curso de XXXX.

Orientador:

Co-orientador: (se houver)

### **Banca examinadora**

---

Membro 1, titulação e instituição

---

Membro 2, titulação e instituição

---

Membro 3, titulação e instituição

Birigui, \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_.

*Dedicatória (Opcional). Não digite a palavra Dedicatória. Texto no qual o autor do trabalho oferece homenagem ou dedica o seu trabalho a alguém (não usar ponto final)*

# AGRADECIMENTOS

Agradecimento (opcional). Folha que contém manifestação de reconhecimento a pessoas e/ou instituições que realmente contribuíram com o autor, devendo ser expressos de maneira simples.

Epígrafe (Opcional). Pensamentos retirados de um livro, uma música, um poema, normalmente relacionado ao tema do trabalho, seguida de indicação de autoria. As epígrafes podem ser colocadas também nas folhas de abertura de cada capítulo.

*“Any fool can write code that a computer can understand. Good programmers write code that humans can understand”.*

---

Martin Fowler



# RESUMO

Elemento obrigatório, constituído de uma sequência de frases concisas e objetivas, fornecendo uma visão rápida e clara do conteúdo do estudo. O texto deverá conter entre 150 a 250 palavras e ser antecedido pela referência do estudo. Também, não deve conter citações e deverá ressaltar o objetivo, o método, os resultados e as conclusões. O resumo deve ser redigido em parágrafo único, seguido das palavras representativas do conteúdo do estudo, isto é, palavras-chave, em número de três a cinco, separadas entre si por ponto e finalizadas também por ponto. Usar o verbo na terceira pessoa do singular, com linguagem impessoal (pronome SE), bem como fazer uso, preferencialmente, da voz ativa.

**Palavras-chave:** Palavra-chave 1. Palavra-chave 2. Palavra-chave 3. Palavra-chave n.

# ABSTRACT OU RÉSUMÉ OU RESUMEN

Elemento obrigatório. É a versão do resumo em português para o idioma de divulgação internacional. Deve ser antecedido pela referência do estudo.

**Keywords:** Keyword 1. Keyword 2. Keyword 3. Keyword n.

Elemento obrigatório. É a versão do resumo em português para o idioma de divulgação internacional. Deve ser antecedido pela referência do estudo.

**Mots-clés :** Mot 1. Mot 2. Mot 3. Mot n.

Elemento obrigatório. É a versão do resumo em português para o idioma de divulgação internacional. Deve ser antecedido pela referência do estudo.

**Palabras clave:** Palabra 1. Palabra 2. Palabra 3. Palabra n.

# LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de figura . . . . .	19
Figura 2 – Diagrama de Casos de Uso . . . . .	35
Figura 3 – Modelo de Domínio . . . . .	38
Figura 4 – Diagrama de Objetos . . . . .	39
Figura 5 – Diagrama de Classes . . . . .	40
Figura 6 – Diagrama de Atividades . . . . .	41
Figura 7 – Diagrama de Estados . . . . .	41
Figura 8 – Diagrama de Sequência . . . . .	45
Figura 9 – Diferentes Detalhamentos dos Serviços Relacionados ao Cliente . . . .	46
Figura 10 – Exemplo de Caso de Teste Elaborado na Ferramenta Testlink . . . . .	50

# LISTA DE QUADROS

Quadro 1 – Exemplo de quadro . . . . .	19
--	----

# LISTA DE TABELAS

Tabela 1	– Exemplo de tabela de 2 colunas . . . . .	19
Tabela 2	– Requisitos Funcionais. . . . .	26
Tabela 3	– Requisitos Funcionais. . . . .	33
Tabela 4	– Modelo para Especificação de Casos de Uso . . . . .	36
Tabela 5	– Convenção para Nome dos Objetos no Banco de Dados . . . . .	43
Tabela 6	– Tabelas Identificadas neste Trabalho . . . . .	43
Tabela 7	– Estado . . . . .	43
Tabela 8	– Cidade . . . . .	44
Tabela 9	– NomeDaTabelaNN . . . . .	44

# LISTA DE ABREVIATURAS E SIGLAS

1D	Uma dimensão
2D	Duas dimensões
3D	Três dimensões

# LISTA DE SÍMBOLOS

$\alpha$	Letra grega minúscula Alfa
$\beta$	Letra grega minúscula Beta

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>18</b>
<b>1.1</b>	<b>Justificativa</b>	<b>18</b>
<b>1.2</b>	<b>Objetivos</b>	<b>18</b>
1.2.1	Objetivo Geral	18
1.2.2	Objetivos Específicos	18
<b>1.3</b>	<b>Problema (Hipótese ou Questão de Pesquisa)</b>	<b>18</b>
<b>1.4</b>	<b>Organização Deste Trabalho</b>	<b>18</b>
<b>2</b>	<b>REVISÃO DA LITERATURA</b>	<b>19</b>
<b>3</b>	<b>MATERIAIS E MÉTODOS</b>	<b>23</b>
<b>4</b>	<b>DESCRIÇÃO GERAL DO PRODUTO</b>	<b>24</b>
<b>5</b>	<b>ELICITAÇÃO DE REQUISITOS E ANÁLISE</b>	<b>25</b>
<b>5.1</b>	<b>Requisitos do Usuário</b>	<b>25</b>
<b>5.2</b>	<b>Requisitos do Sistema</b>	<b>25</b>
5.2.1	Requisitos Funcionais	25
5.2.2	Requisitos Não-Funcionais	32
5.2.3	Restrições, Suposições e Dependências	33
5.2.4	Requisitos Adiados	34
<b>5.3</b>	<b>Casos de Uso</b>	<b>34</b>
5.3.1	Diagrama de Casos de Uso	34
5.3.2	Especificação dos Casos de Uso	35
<b>5.4</b>	<b>Modelo de Domínio</b>	<b>36</b>
<b>5.5</b>	<b>Diagrama de Objetos</b>	<b>38</b>
<b>5.6</b>	<b>Diagrama de Classes de Análise</b>	<b>39</b>
<b>5.7</b>	<b>Diagrama de Atividades</b>	<b>40</b>
<b>5.8</b>	<b>Diagrama de Estados</b>	<b>41</b>
<b>6</b>	<b>PROJETO DE SOFTWARE</b>	<b>42</b>
<b>6.1</b>	<b>Projeto de Interface</b>	<b>42</b>
<b>6.2</b>	<b>Projeto de Dados</b>	<b>42</b>
6.2.1	Mapeamento Objeto-Relacional	42
6.2.2	Estrutura das Tabelas no Banco de Dados	42
6.2.3	Diagrama de Pacotes	44
6.2.4	Diagrama de Classes de Projeto	44



<b>6.3</b>	<b>Projeto Procedimental</b>	<b>44</b>
6.3.1	Diagrama de Sequência	44
6.3.2	Diagrama de Atividades	45
6.3.3	Diagrama de Estados	45
<b>6.4</b>	<b>Projeto Arquitetural</b>	<b>45</b>
<b>7</b>	<b>IMPLEMENTAÇÃO</b>	<b>47</b>
<b>8</b>	<b>TESTE</b>	<b>48</b>
<b>9</b>	<b>IMPLANTAÇÃO</b>	<b>51</b>
<b>10</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>52</b>
<b>11</b>	<b>TRABALHOS RELACIONADOS</b>	<b>53</b>
<b>12</b>	<b>CONCLUSÕES/CONCLUSÕES PARCIAIS</b>	<b>54</b>
<b>13</b>	<b>CRONOGRAMA</b>	<b>55</b>
	<b>REFERÊNCIAS</b>	<b>56</b>
	<b>APÊNDICE A – TÍTULO DO APÊNDICE A</b>	<b>57</b>
	<b>APÊNDICE B – TÍTULO DO APÊNDICE B</b>	<b>58</b>
	<b>APÊNDICE C – TÍTULO DO APÊNDICE C</b>	<b>59</b>
	<b>APÊNDICE D – TÍTULO DO APÊNDICE D</b>	<b>60</b>
	<b>APÊNDICE E – TÍTULO DO APÊNDICE E</b>	<b>61</b>
	<b>ANEXO A – TÍTULO DO ANEXO A</b>	<b>62</b>
	<b>ANEXO B – TÍTULO DO ANEXO B</b>	<b>63</b>
	<b>ANEXO C – TÍTULO DO ANEXO C</b>	<b>64</b>
	<b>ANEXO D – TÍTULO DO ANEXO D</b>	<b>65</b>
	<b>ANEXO E – TÍTULO DO ANEXO E</b>	<b>66</b>

# 1 INTRODUÇÃO

Aqui deve-se introduzir o trabalho. Trata-se da parte inicial do texto, onde devem constar: a delimitação do assunto tratado, os objetivos da pesquisa e outros elementos necessários para situar o tema do trabalho.

## 1.1 Justificativa

Texto da justificativa. Conte como se chegou ao "questionamento", como se identificou o problema, por que e de onde ele surgiu. É a explicação do porquê a inquietação em questão existe.

## 1.2 Objetivos

Descrever o que se pretende alcançar ao final do projeto.

### 1.2.1 Objetivo Geral

Qual seu objetivo geral.

### 1.2.2 Objetivos Específicos

- Objetivo específico 1;
- Objetivo específico 2;
- Objetivo específico n.

## 1.3 Problema (Hipótese ou Questão de Pesquisa)

Escolha uma dentre as 3 formas de escrever: 1) o problema que o trabalho propõe resolver; ou 2) as hipóteses que se deseja comprovar (ou não) com o estudo; ou 3) as questões de pesquisa levantadas para serem investigadas.

## 1.4 Organização Deste Trabalho

Como seu trabalho está organizado (capítulos) após a introdução.

## 2 REVISÃO DA LITERATURA

Texto da revisão da literatura, dividido em seções e subseções.

Este é um exemplo de como usar figuras. Referência cruzada: Figura 1

Figura 1 – Exemplo de figura



**Fonte:** Elaborada pelo autor

Este é um exemplo de como usar tabelas. Referência cruzada: Tabela 1

Tabela 1 – Exemplo de tabela de 2 colunas

Coluna 1	Coluna 2
Dado 1a	Dado 2a
Dado 1b	Dado 2b
Dado 1c	Dado 2c
Dado 1d	Dado 2d

**Fonte:** Elaborada pelo autor

Este é um exemplo de como usar quadros. Referência cruzada: Quadro 1

Quadro 1 – Exemplo de quadro

Cores		
Nome	Hexa	Amostra
Preto	#000000	
Marrom	#993300	
Vermelho	#FF0000	
Laranja	#FF3300	
Amarelo	#FFFF00	
Branco	#FFFFFF	

**Fonte:** Elaborada pelo autor

Este é um exemplo de como usar equações. Referência cruzada: Equação 2.1

$$\sum_{i=1}^n i = \frac{n(n+1)}{2} \quad (2.1)$$

---

Exemplo de inserção de lista de código fonte (**não use acentos no código!**):

---

```
1  /**
2   * Classe de exemplo.
3   *
4   * @author David Buzatto
5   */
6  public class Grafo {
7
8      public static void main( String[] args ) {
9
10         System.out.println( "Exemplo de código fonte!" );
11         System.out.println( "Nao use acentos!" );
12
13     }
14
15 }
```

---

Este é um exemplo de como inserir texto sem formatação (ambiente verbatim):

Texto sem formatação, como espaçamento igual.

Exemplo de lista de itens:

- **Item 1:** texto...;
- **Item 2:** texto...;
  - **Subitem:** texto...;
  - **Subitem:** texto...;
  - **Subitem:** texto...;
- **Item 3:** texto...;
- **Item n:** texto....

Exemplo de lista numerada:

1. **Item:** texto...;
2. **Item:** texto...;
  - a) **Subitem:** texto...;
  - b) **Subitem:** texto...;
  - c) **Subitem:** texto...;

3. **Item:** texto...;
4. **Item:** texto....

Exemplos de comandos para texto e referências:

- Para iniciar um novo parágrafo, basta deixar uma linha em branco no código fonte;
- Não force o compilador a pular mais de uma linha, pois terá influência negativa na composição do documento;
- Sempre deixe o L<sup>A</sup>T<sub>E</sub>X realizar a formatação de parágrafos e posicionamento de elementos;
- Utilização de aspas simples (abertura ‘, fechamento ’): ‘Texto entre aspas simples’;
- Utilização de aspas duplas (abertura ‘‘, fechamento ’’): “Texto entre aspas duplas”;
- Negrito (comando `\textbf`): **texto em negrito**;
- Itálico (comando `\textit`): *texto em itálico*;
- Sublinhado (comando `\underline`): texto sublinhado;
- Negrito e itálico (usar comandos juntos): ***texto em negrito e itálico***;
- Alterar cor do texto (comando `\textcolor{cor}{texto}`):
  - Exemplo `\textcolor{red}{texto}`: texto vermelho;
  - Exemplo `\textcolor[RGB]{255, 102, 0}{texto}`: texto laranja;
  - Exemplo `\textcolor[HTML]{006AD7}{texto}`: texto azul;
- Ambiente matemático inline (comando `$ expressão $`):  $s = x^2 - 2x + 1$ ;
- Referência normal (comando `\cite`):
  - (AGAISSE; LERECLUS, 1995);
  - (ABEDI et al., 2014);
  - (CRICKMORE et al., 2016);
- Referência normal com mais de uma obra (comando `\cite`):
  - (ABEDI et al., 2014; AGAISSE; LERECLUS, 1995);
  - (AGAPITO-TENFEN et al., 2014; CRICKMORE et al., 2016; NELSON; COX, 2014);

- Referência nome e ano (comando `\citeauthorandyear`):

- Agaisse e Lereclus (1995);
- Abedi et al. (2014);
- Crickmore et al. (2016);

Exemplo 1 de citação direta:

Os 20 aminoácidos usualmente encontrados como resíduos em proteínas contém um grupo  $\alpha$ -carboxil, um grupo  $\alpha$ -amino e um grupo R distinto substituído no átomo de carbono  $\alpha$ . O átomo de carbono  $\alpha$  de todos os aminoácidos, com exceção da glicina, é assimétrico e, portanto, os aminoácidos podem existir em pelo menos duas formas estereoisoméricas. Somente os estereoisômeros L, com uma configuração relacionada à configuração absoluta da molécula de referência L-gliceraldeído, são encontrados em proteínas (NELSON; COX, 2014, p. 81).

Exemplo 2 de citação direta:

*These various insecticidal proteins are synthesized during the stationary phase and accumulate in the mother cell as a crystal inclusion which can account for up to 25% of the dry weight of the sporulated cells. The amount of crystal protein produced by a *B. thuringiensis* culture in laboratory conditions (about 0.5 mg of protein per ml) and the size of the crystals (24) indicate that each cell has to synthesize  $10^6$  to  $2 \times 10^6$   $\delta$ -endotoxin molecules during the stationary phase to form a crystal (AGAISSE; LERECLUS, 1995, p. 1).*

Exemplo de nota de rodapé<sup>1</sup>.

---

<sup>1</sup> Essa é uma nota de rodapé!

### 3 MATERIAIS E MÉTODOS

A metodologia especifica como os objetivos estabelecidos serão alcançados. As partes constitutivas da metodologia: a amostragem e as formas de coleta, de organização e de análise dos dados.

Deem uma lida no link abaixo para ter uma ideia do que colocar. Conversar com o orientador para definição da metodologia. <[https://blog.mettzer.com/metodologia-cientifica/amp/#O\\_QUE\\_E\\_METODOLOGIA\\_CIENTIFICA](https://blog.mettzer.com/metodologia-cientifica/amp/#O_QUE_E_METODOLOGIA_CIENTIFICA)>

## 4 DESCRIÇÃO GERAL DO PRODUTO

Faça uma breve apresentação, visão geral do produto a ser desenvolvido.

Braga (2008);



## 5 ELICITAÇÃO DE REQUISITOS E ANÁLISE

Esta seção do documento esclarece os requisitos para operação do sistema “Meal Control”, este sistema visa corrigir o processo atual de refeições diárias e mensais em hospitais. Pois o sistema atual, não permite o controle da distribuição de refeições, registro de dietas, acompanhamento da recuperação do paciente e o registro de refeições por setor, período e tipo de dieta. Com isso, gera uma defasagem na comunicação das equipes responsáveis pelo atendimento, ocasionando uma lentidão no processo de definição da alimentação necessária para o paciente. Logo, este sistema irá proporcionar melhor otimização na comunicação, melhora no monitoramento do paciente, estatísticas sobre as dietas em vigor, e consequentemente, um melhor desempenho nos atendimentos em geral.

### 5.1 Requisitos do Usuário

Texto da justificativa

### 5.2 Requisitos do Sistema

Fazer uma pequena definição de requisitos do sistema para introduzir a seção (informar a fonte). Assim, esta seção apresenta os requisitos identificados nesse trabalho.

#### 5.2.1 Requisitos Funcionais

Fazer uma pequena definição de requisitos funcionais para introduzir essa seção. Informar quais os atores identificados nessa etapa do desenvolvimento, como o parágrafo a seguir. Terminar o parágrafo informando em qual tabela os requisitos funcionais identificados estão apresentados, mas ou menos como no parágrafo a seguir.

De acordo com Braga (2008), Requisitos Funcionais são ..... Os atores que interagem com o sistema são: administrador, cliente e fornecedor. Os requisitos funcionais identificados para este sistema estão apresentados na Tabela 2

Tabela 2 – Requisitos Funcionais.

Começo da tabela			
Numeração	Ator	Descrição	Prioridade
RF01	Administrador	<p><b>Ação:</b> Cadastrar Hospital.</p> <p><b>Objeto:</b> Hospital.</p> <p><b>Atributos:</b> CNPJ (string), nome (string), CEP (string), telefone (string), CNAE (string), logradouro (string), número de logradouro (int).</p> <p><b>Exemplo:</b> Unimed.</p> <p><b>Restrições:</b> Os atributos CNAE, CNPJ, CEP, telefone são obrigatórios. Não podem ser inseridos dois ou mais hospitais com o mesmo CNPJ e telefone.</p> <p><b>Finalidade:</b> Esta função é usada para cadastro de funcionários em uma única instituição, permite excluir, alterar e adicionar hospitais.</p>	Essencial
RF02	Administrador	<p><b>Ação:</b> Cadastrar Funcionário.</p> <p><b>Objeto:</b> Funcionário.</p> <p><b>Atributos:</b> CPF (string), nome (string), CEP (string), telefone (string), data de nascimento (string), logradouro (string), número de logradouro (int), email (string), hospital (string).</p> <p>obs: Para cada cargo poderá existir atributos adicionais. O médico que terá o CRM (string), a nutricionista terá o CRN (string), para os demais haverá o cargo (string). Exemplo: Uma nutricionista.</p> <p><b>Restrições:</b> Os atributos CPF, email, data de nascimento, telefone são obrigatórios. Não podem ser inseridos dois ou mais funcionários com o mesmo CPF, email e telefone.</p> <p><b>Finalidade:</b> Esta função é usada para cadastro de funcionários no sistema podendo atribuir um funcionário para uma ou mais instituições, permite excluir, alterar e adicionar funcionários.</p>	Essencial

Continuação da tabela			
Numeração	Ator	Descrição	Prioridade
RF03	Administrador	<p><b>Ação:</b> Cadastrar Tipo de Refeição.</p> <p><b>Objeto:</b> Tipo de refeição. Atributos: hospital (string), nome (string), tipo (int). obs: Todos os devem ser previamente cadastrados, como o tipo de refeição, tipo de dieta e categoria de estado de saúde possuem a mesma estrutura optou-se por realizá-los na mesma função diferenciando eles pelo atributo tipo, considerando, o tipo de refeição sendo o tipo 2.</p> <p><b>Exemplo:</b> Café da manhã, almoço e jantar.</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios.</p> <p><b>Finalidade:</b> Não podem ser inseridos dois ou mais tipos de refeição iguais para o mesmo hospital. Esta função é usada para cadastro de tipos de refeições no hospital, permitindo que o sistema se adapte à rotina do hospital, permitindo que o sistema possa atuar em muitos hospitais sem necessidade de mudar o sistema.</p>	Essencial

Continuação da tabela			
Numeração	Ator	Descrição	Prioridade
RF05	Administrador	<p><b>Ação:</b> Cadastrar Tipo de Dieta.</p> <p><b>Objeto:</b> Tipo de dieta.</p> <p><b>Atributos:</b> hospital (string), nome (string), tipo (int). obs: Todos os devem ser previamente cadastrados, como o tipo de refeição, tipo de dieta e categoria de estado de saúde possuem a mesma estrutura optou-se por realizá-los na mesma função diferenciando eles pelo atributo tipo, considerando, o tipo de refeição sendo o tipo 3.</p> <p><b>Exemplo:</b> Líquida, Pastosa.</p> <p><b>Restrições:</b></p> <p>Todos os atributos são obrigatórios. Não podem ser inseridos dois ou mais tipos de refeição iguais para o mesmo hospital.</p> <p><b>Finalidade:</b></p> <p>Esta função é usada para cadastro de tipos de refeições no hospital, permitindo que o sistema se adapte à rotina do hospital, permitindo que o sistema possa atuar em muitos hospitais sem necessidade de mudar o sistema.</p>	Essencial

Continuação da tabela			
Numeração	Ator	Descrição	Prioridade
RF06	Administrador	<p><b>Ação:</b> Cadastrar Categoria Estado de Saúde.</p> <p><b>Objeto:</b> Categoria de estado de saúde.</p> <p><b>Atributos:</b> hospital (string), nome (string), tipo (int). obs: Todos os devem ser previamente cadastrados, como o tipo de refeição, tipo de dieta e categoria de estado de saúde possuem a mesma estrutura optou-se por realizá-los na mesma função diferenciando eles pelo atributo tipo, considerando, o tipo de refeição sendo o tipo 1.</p> <p><b>Exemplo:</b> Crítico, estável .</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios. Não podem ser inseridos dois ou mais tipos de refeição iguais para o mesmo hospital.</p> <p><b>Finalidade:</b> Esta função é usada para cadastro de tipos de refeições no hospital, permitindo que o sistema se adapte à rotina do hospital, permitindo que o sistema possa atuar em muitos hospitais sem necessidade de mudar o sistema.</p>	Essencial
RF07	Administrador	<p><b>Ação:</b> Cadastrar Unidade de Medida.</p> <p><b>Objeto:</b> Unidade de medida.</p> <p><b>Atributos:</b> nome (String).</p> <p><b>Exemplo:</b> ML, L, Kg.</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios. Não podem haver duas unidades com o mesmo nome.</p> <p><b>Finalidade:</b> Para controle da quantidade e produção do relatório mensal esta função permite que o usuário controle a distribuição de suas refeições.</p>	Essencial

Continuação da tabela			
Numeração	Ator	Descrição	Prioridade
RF08	Nutricionista	<p><b>Ação:</b> Cadastrar Item do cardápio.</p> <p><b>Objeto:</b> Item do cardápio.</p> <p><b>Atributos:</b> nome (String).</p> <p><b>Exemplo:</b> café, bolo, maçã.</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios. Não podem haver duas unidades com o mesmo nome.</p> <p><b>Finalidade:</b> Para montagem do futuro cardápio, esta função permite o registro de itens que virão a compor o cardápio.</p>	Essencial
RF09	Nutricionista	<p><b>Ação:</b> Cadastrar o cardápio.</p> <p><b>Objeto:</b> Cardápio.</p> <p><b>Atributos:</b> nome (String).</p> <p><b>Exemplo:</b> matinal, jantar.</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios. Não podem haver duas unidades com o mesmo nome.</p> <p><b>Finalidade:</b> Está gera um objeto contendo os itens que compõem uma refeição completa, com a quantidade de cada item, este objeto também possui uma atribuição para, um ou mais tipos de categoria de estado, dietas e tipo de refeição.</p>	Essencial
RF10	Nutricionista	<p><b>Ação:</b> Cadastrar Agendamento de refeições.</p> <p><b>Objeto:</b> Agendamento.</p> <p><b>Atributos:</b> Dia de início (Date).</p> <p><b>Exemplo:</b> refeições para a páscoa.</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios. Não podem haver dois cardápios com o mesmo os mesmos tipos se sobrepondo.</p> <p><b>Finalidade:</b> Está gera um objeto contendo os itens que compõem uma agenda de cardápios, organizados pelo seus tipos, pelo intervalo determinado pela nutricionista.</p>	Essencial

Continuação da tabela			
Numeração	Ator	Descrição	Prioridade
RF11	Recepcionista Administrador	<p><b>Ação:</b> Cadastrar Pacientes internados.</p> <p><b>Objeto:</b> Paciente.</p> <p><b>Atributos:</b> CPF (string), nome (string), CEP (string), telefone (string), data de nascimento (string), logradouro (string), número de logradouro (int), email (string).</p> <p><b>Exemplo:</b> Luiz Eduardo.</p> <p><b>Restrições:</b> Os atributos CPF, nome, telefone e data de nascimento são obrigatórios . Não podem haver dois pacientes com o mesmo CPF.</p> <p><b>Finalidade:</b> Está gera um objeto contendo os dados do paciente para futuras manipulações e métodos.</p>	Essencial
RF12	Administrador Enfermeira Médico	<p><b>Ação:</b> Alterar Pacientes internados.</p> <p><b>Objeto:</b> Paciente.</p> <p><b>Atributos:</b> CPF (string), nome (string), CEP (string), telefone (string), data de nascimento (string), logradouro (string), número de logradouro (int), email (string). Exemplo: Luiz Eduardo.</p> <p><b>Restrições:</b> Os atributos CPF, nome, telefone e data de nascimento são obrigatórios . Não podem haver dois pacientes com o mesmo CPF.</p> <p><b>Finalidade:</b> Está permite que o administrador mude o estado do paciente dentro do sistema, para um paciente internado e vice-versa, com o objetivo de não perder dados, e manter esse dados para futuras aplicações.</p>	Essencial

Continuação da tabela			
Numeração	Ator	Descrição	Prioridade
RF13	Administrador	<p><b>Ação:</b> Atribuir estado de saúde.</p> <p><b>Objeto:</b> Paciente.</p> <p><b>Atributos:</b></p> <p><b>Exemplo:</b> Luiz Eduardo em estado crítico.</p> <p><b>Restrições:</b> Todos os atributos são obrigatórios. Não podem haver dois estados de saúde para o mesmo paciente.</p> <p><b>Finalidade:</b> Esta função permite atribuir um tipo de estado de saúde previamente cadastrado para um paciente.</p>	Essencial
RF14	Cozinha Nutricionista Enfermeira	<p><b>Ação:</b> Visualização do cardápio. <b>Objeto:</b> cardápio. <b>Atributos:</b> <b>Exemplo:</b> jantar, matinal. <b>Restrições:</b> não possui <b>Finalidade:</b> Esta função permite a visualização do agendamento dos cardápios de acordo com a data. Dando preferência para os cardápios diários, está calcula e demonstra os itens e a quantidade por paciente.</p>	Essencial
RF15	Nutricionista Administrador Enfermeira Cozinha	<p><b>Ação:</b> Visualização do relatório mensal de refeições.</p> <p><b>Objeto:</b> agendamentos, cardápios.</p> <p><b>Atributos:</b></p> <p><b>Exemplo:</b></p> <p><b>Restrições:</b> não possui</p> <p><b>Finalidade:</b> Esta função permite que o sistema calcule com base nos agendamentos e cardápios, a quantidade de comida distribuída no período solicitado, devolvendo de maneira organizada um relatório de refeições que podem ser filtradas por tipo de dieta, período e setor.</p>	Essencial
<b>Fonte:</b> Elaborada pelo Autor			

### 5.2.2 Requisitos Não-Funcionais

Como todo sistema existem algumas exigências de operação que transcendem os requisitos funcionais, o que se torna essencial para sua operação fatores como o ambiente



e características que o sistema necessita para uma boa operação, consolidados na Tabela 3

Tabela 3 – Requisitos Funcionais.

Começo da tabela			
Numeração	Ator	Descrição	Prioridade
NF01	Desempenho	O sistema deve se comportar de maneira eficiente sem apresentar travamentos, lentidão, compatibilidade com o hardware.	Essencial
NF02	Otimização	O sistema deve apresentar um comportamento complementar ao processo de refeições diárias, login e cadastro.	Essencial
NF03	Backup de dados	O sistema deve apresentar processo de proteção e recuperação de todos os dados presentes, sendo eles dados pessoais, registro de refeições e cardápio e os demais dados registrados.	Essencial
NF03	Interface	O sistema deve ter uma interface simples e usual, para facilitar o acesso e operação dos servidores do hospital.	Essencial
NF04	Interface	O sistema deve ter uma interface simples e usual, para facilitar o acesso e operação dos servidores do hospital.	Essencial
NF05	Segurança	O sistema deve restringir o acesso para apenas os servidores do hospital, além de proteger os dados dos pacientes e servidores.	Essencial
NF06	Multi-plataforma	O sistema deve permitir acesso de diferentes plataformas ao sistema.	Essencial
NF07	Portabilidade	O sistema deve se conectar ao servidor e ao banco de dados do hospital.	Essencial
NF08	Comunicação	O sistema deve permitir uma comunicação clara entre os membros da equipe.	Essencial
NF09	Linguagem do sistema	O sistema deve ser desenvolvido na linguagem Flutter.	Essencial
<b>Fonte:</b> Elaborada pelo Autor			

tabular

### 5.2.3 Restrições, Suposições e Dependências

Atualmente este projeto está focando em realizar um pequeno bloco de uma sistema muito, que será necessário para que o projeto entre em execução e possa realizar .

- Cadastro de Pacientes;
- Acesso aos perfis de usuário;
- Interface, Banco de dados e o cadastro de funcionários.

#### 5.2.4 Requisitos Adiados

Este projeto atualmente está modelando um sistema que atendá os requisitos da cliente, embora ele a principio não irá ser executado.

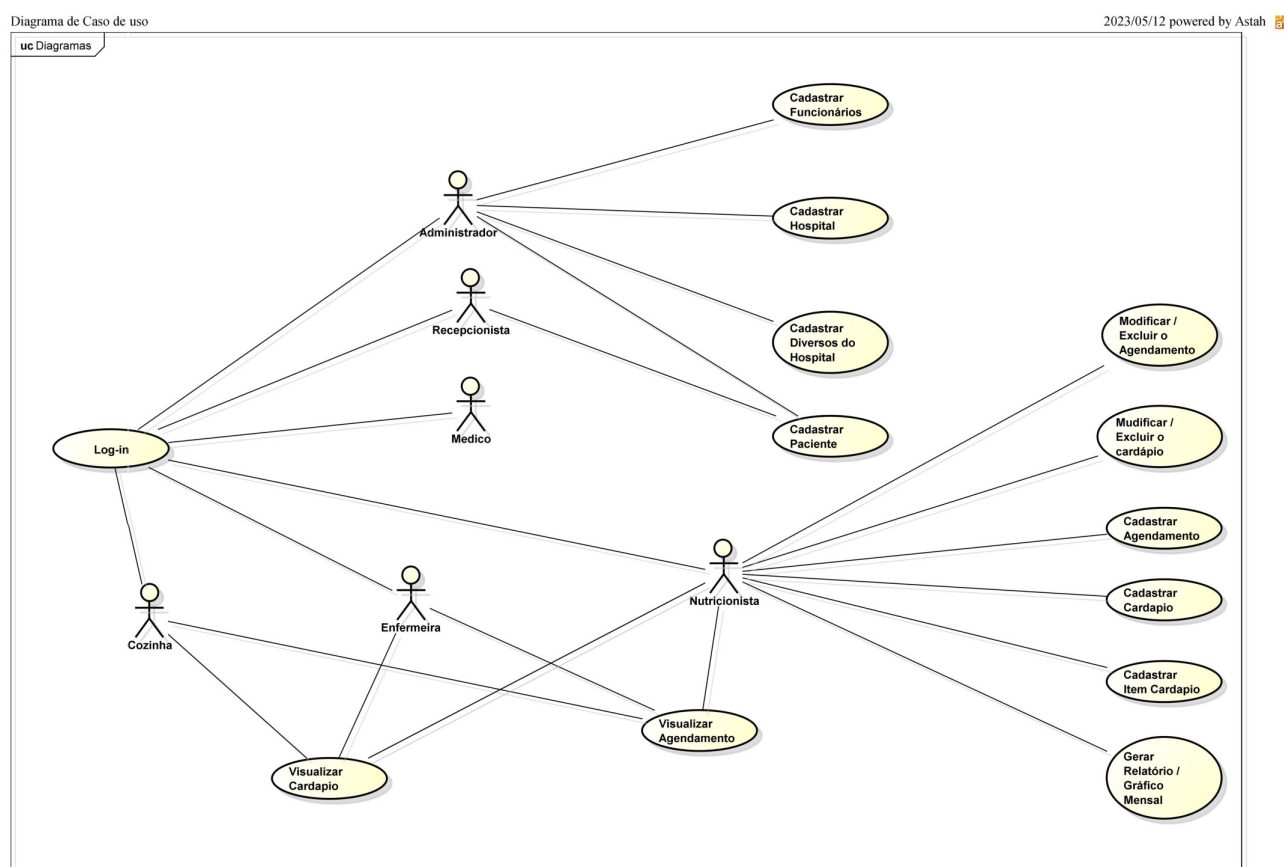
### 5.3 Casos de Uso

Forneça uma breve definição de casos de uso (com referência da fonte) e informe que os casos de uso identificados neste trabalho estão aqui apresentados.

#### 5.3.1 Diagrama de Casos de Uso

Um diagrama de caso de uso é uma representação gráfica que descreve as interações entre os atores e o sistema em um determinado contexto. Ele mostra os diferentes casos de uso do sistema, que são ações ou funcionalidades que o sistema oferece aos atores. 2.

Figura 2 – Diagrama de Casos de Uso



**Fonte:** Elaborada pelo autor

### 5.3.2 Especificação dos Casos de Uso

Uma especificação de caso de uso fornece detalhe textual para um caso de uso. Uma estrutura de tópicos de amostra de uma especificação de caso de uso é fornecida. É possível reutilizar e modificar a estrutura de tópicos conforme necessário em um documento de especificação de caso de uso. A Tabela 4 apresenta um modelo para especificação de casos de uso (extraído de IBM-Especificação de Casos de Uso).

Tabela 4 – Modelo para Especificação de Casos de Uso

Item	Descrição
Nome do Caso de Uso	Declara o nome do caso de uso. Geralmente, o nome expressa o objetivo ou resultado observável do caso de uso, como "Sacar Dinheiro" no caso de um caixa eletrônico.
Descrição Resumida	Descreve a função e o objetivo do caso de uso.
Fluxo de Eventos	Apresenta o fluxo básico e os fluxos alternativos. O fluxo de eventos descreve o comportamento do sistema; ele não descreve como o sistema funciona, os detalhes da apresentação ou os detalhes da interface com o usuário. Se forem trocadas informações, o caso de uso deverá ser específico sobre o que será transmitido de um lado para outro. Por exemplo, em vez de descrever uma ação como "o agente insere informações do cliente", indique que "o agente insere o nome e o endereço do cliente".
Fluxo Básico	Descreve o comportamento principal ideal do sistema.
Fluxos Alternativos	Descreve exceções ou desvios do fluxo básico, como o comportamento do sistema quando o agente insere um ID de usuário incorreto e a autenticação do usuário falha.
Requisitos Especiais	Requisitos não-funcionais que são específicos para um caso de uso mas que não são especificados no texto do fluxo do caso de uso de eventos. Exemplos de requisitos especiais incluem estes fatores: requisitos legais e regulamentares; padrões de aplicativo; atributos de qualidade do sistema, incluindo usabilidade, confiabilidade, desempenho e capacidade de suporte; sistemas operacionais e ambientes; requisitos de compatibilidade e restrições de design.
Condições Prévias	Um estado do sistema que deve estar presente antes de um caso de uso ser iniciado.
Pós-Condições	Uma lista de estados possíveis para o sistema imediatamente após a conclusão de um caso de uso.
Pontos de Extensão	Um ponto no fluxo de eventos do caso de uso em que outro caso de uso é referenciado.

**Fonte:** Extraído de IBM-Especificação de Casos de Uso

Elaborar as especificações para cada caso de uso, quando for necessário, como por exemplo: A especificação do caso de uso Finalizar Compra é

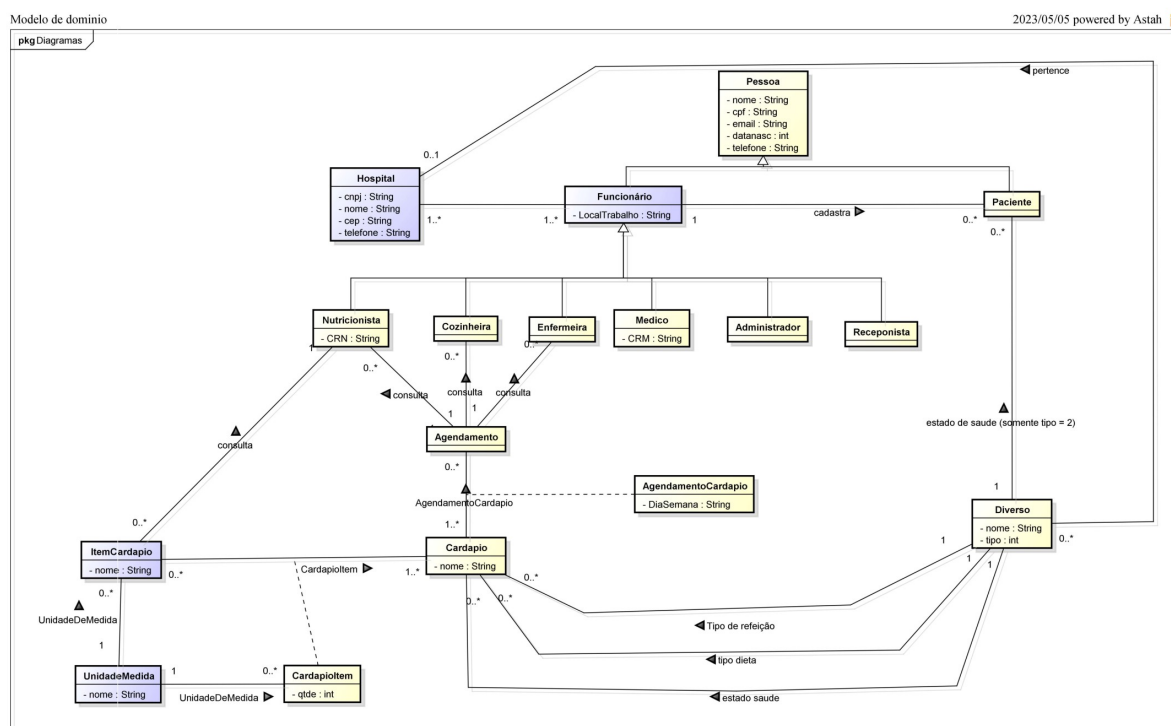
## 5.4 Modelo de Domínio

Um modelo de domínio é uma representação conceitual da estrutura e dos conceitos fundamentais de um determinado domínio de negócio. Ele descreve as entidades, seus atributos e os relacionamentos entre elas, permitindo uma visão abstrata do domínio em questão. 5.

De acordo com Braga (2008), o modelo de domínio é ..... A Figura 5 apresenta o modelo de domínio desenvolvido neste trabalho.

<b>Caso de Uso</b> Finalizar Comprar
<b>Referências</b> RF02
<b>Descrição Geral</b> O caso de uso inicia-se quando o cliente deseja efetuar compra dos produtos que estão inseridos no carrinho de compras.
<b>Atores</b> Cliente
<b>Pré-Condições</b> Cliente logado no sistema, produtos já inseridos no carrinho de compras.
<b>Garantia de Sucesso (Pós-Condições)</b> Pedido fechado, compra efetuada, sistema aguardando confirmação de pagamento.
<b>Requisitos Especiais</b>
<b>Fluxo Básico</b>  <ol style="list-style-type: none"><li>1. Cliente deseja finalizar compra, sistema solicita que informe a forma de pagamento e de entrega;</li><li>2. Cliente deseja efetuar pagamento em forma de cartão de crédito/débito;</li><li>3. Sistema solicita informações do cartão do cliente;</li><li>4. Sistema faz validação das informações;</li><li>5. Sistema gera o número do pedido;</li><li>6. Compra finalizada com sucesso.</li></ol>
<b>Fluxo Alternativo</b>  <ol style="list-style-type: none"><li>1. Cliente deseja efetuar pagamento através de boleto bancário<ol style="list-style-type: none"><li>1.1 Sistema gera boleto para o cliente;</li><li>1.2 Retorne ao passo 5.</li></ol></li></ol>

Figura 3 – Modelo de Domínio

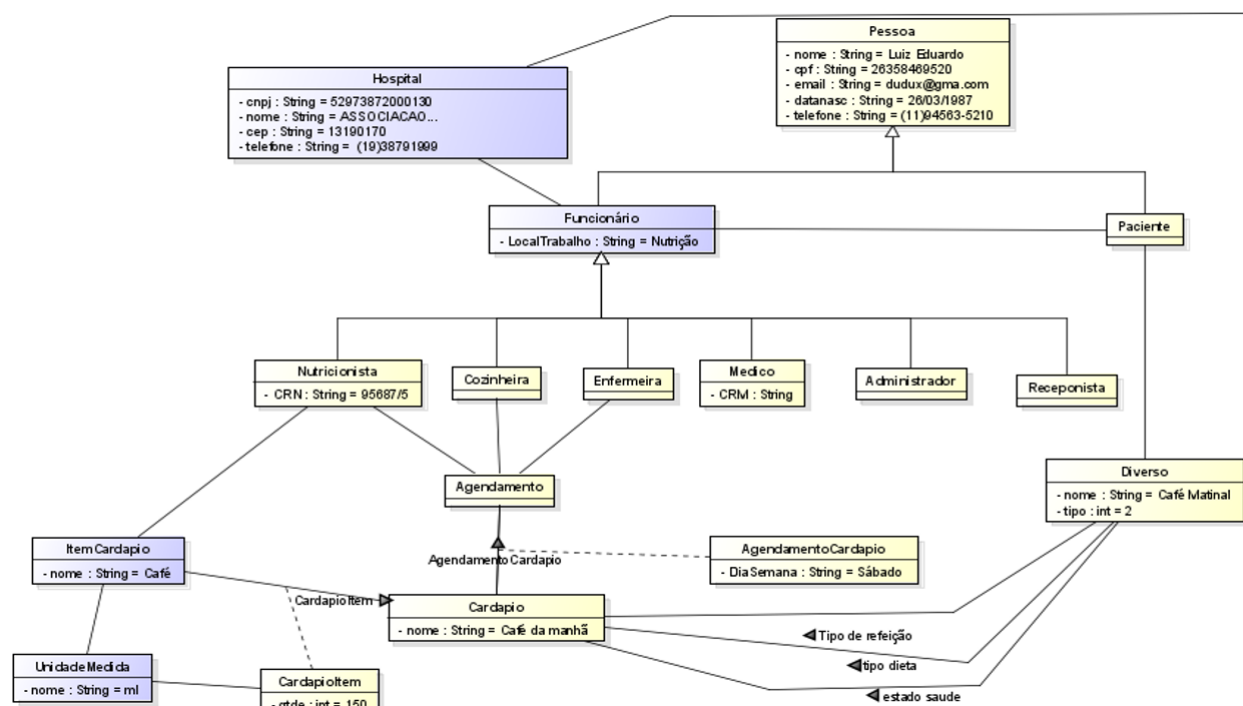


Fonte: Elaborada pelo autor

## 5.5 Diagrama de Objetos

Um diagrama de objetos é uma representação gráfica que mostra um conjunto de objetos e suas relações em um determinado momento. Ele descreve a estrutura estática de um sistema, destacando os objetos envolvidos e as relações entre eles, incluindo os atributos e métodos dos objetos.

Figura 4 – Diagrama de Objetos

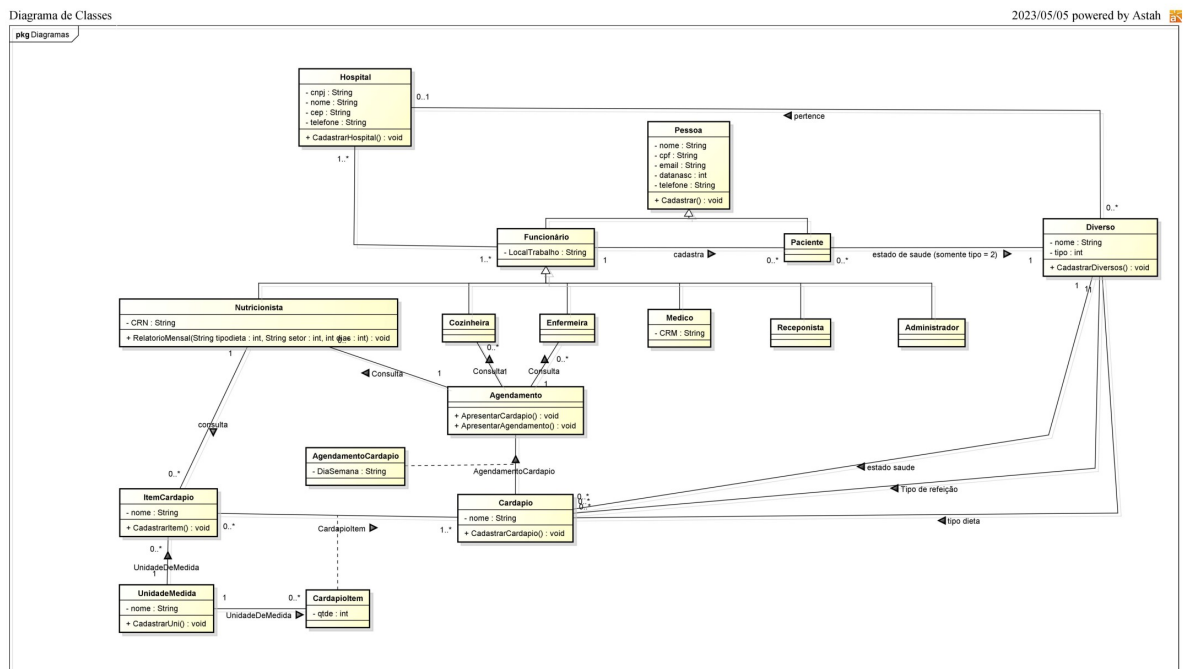


Fonte: Elaborada pelo autor

## 5.6 Diagrama de Classes de Análise

Um diagrama de classe de análise é uma representação gráfica da estrutura estática de um sistema de software em um nível de abstração mais alto. Ele identifica as principais classes do sistema, seus atributos e os relacionamentos entre as classes.

Figura 5 – Diagrama de Classes



**Fonte:** Elaborada pelo autor

## 5.7 Diagrama de Atividades

Um diagrama de atividade ilustra a natureza dinâmica de um sistema pela modelagem do fluxo de controle de atividade à atividade. Uma atividade representa uma operação em alguma classe no sistema que resulta em uma mudança no estado do sistema.

Tipicamente, diagramas de atividades são usados para modelar fluxos de processos, processos de negócios ou operações internas. o diagrama de atividades é similar a uma máquina de estados, mas tem um propósito diferente, o qual envolve capturar ações e seus resultados em termos de mudanças do estado do objeto.

O diagrama de atividades é representado por um gráfico de atividades que mostram o fluxo de uma atividade para outra. Esse fluxo é mostrado através de transições, que são setas direcionadas, mostrando o caminho entre os estados de atividade (ação).

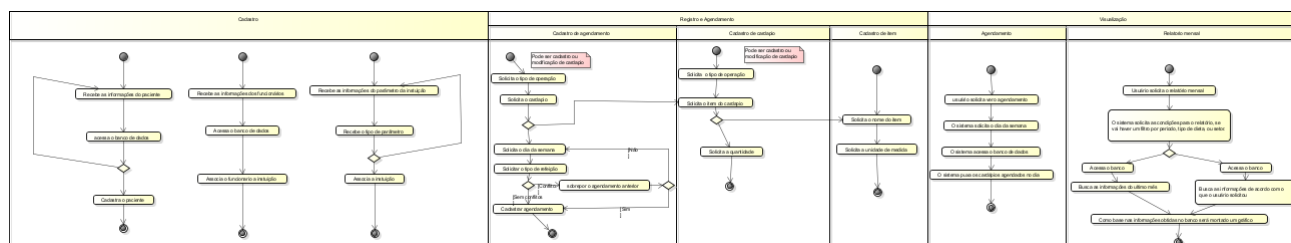
A Figura 6 mostra o diagrama de atividades para a operação `averiguarCredito` na classe `Pedido` da `Virtual LTDA`. Note que a atividade “Preparar Mensagem de Credito” define o que fazer, mas não como fazer (RIBEIRO, 2021).

Um diagrama de atividades é normalmente composto pelos seguintes elementos:



atividades (ações), estados de atividade (ação), transição, fluxo de objeto, estado inicial, estado final, branching, sincronização, raiais.

Figura 6 – Diagrama de Atividades

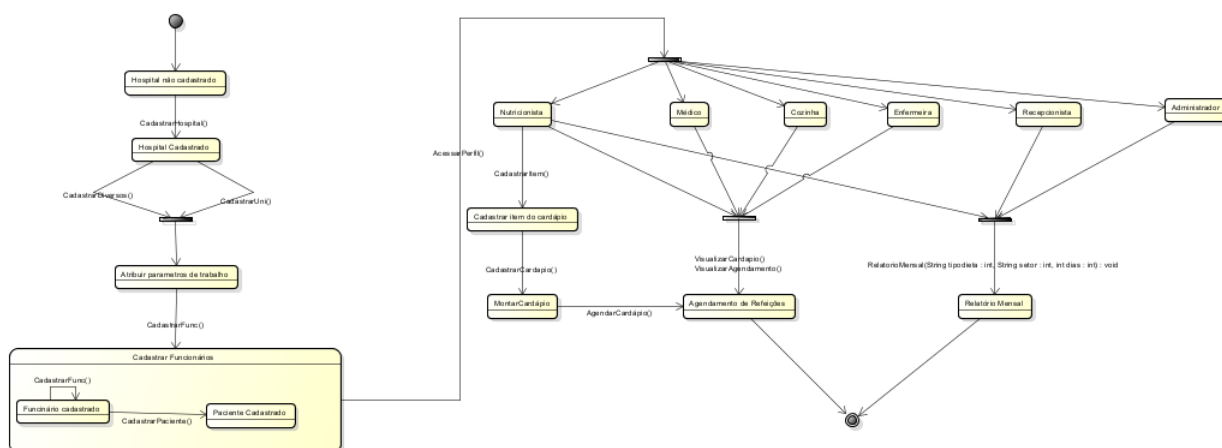


Fonte: Elaborada pelo autor

## 5.8 Diagrama de Estados

Um diagrama de estados é uma representação gráfica que descreve os diferentes estados de um objeto ou de um sistema ao longo do tempo. Ele mostra as transições entre os estados, os eventos que desencadeiam essas transições e as ações executadas em resposta a esses eventos. 7.

Figura 7 – Diagrama de Estados



Fonte: Elaborada pelo autor

## 6 PROJETO DE SOFTWARE

Parte principal do texto, que contém a exposição ordenada e pormenorizada do assunto. Divide-se em seções e subseções, que variam em função da abordagem do tema e do método.

### 6.1 Projeto de Interface

Apresentar o protótipo das interfaces do sistema. Podem ser construídas utilizando qualquer ferramenta. Apresente os padrões utilizados.

### 6.2 Projeto de Dados

Informar nesta seção qual o Banco de Dados a ser utilizado, qual a IDE utilizada para manipulação do Banco de Dados, assim como descrever quais os tipos de dados aceitos no Banco de Dados.

#### 6.2.1 Mapeamento Objeto-Relacional

Uma vez que foi elaborada uma modelagem orientada a objetos, utilizando o diagrama de classes UML, e o banco de dados a ser utilizado é relacional, deve-se identificar as relações. Assim, a partir dos requisitos e do modelo de domínio enumerar as relações que devem formar o banco de dados.

Regras para realizar o mapeamento: (Helen, colocar as regras aqui para que os alunos escrevam)

- Item 1;
- Item 2;
- Item n.

#### 6.2.2 Estrutura das Tabelas no Banco de Dados

Explicar qual o padrão adotado para o nome dos objetos de banco de dados, como por nome da chave primária, chave estrangeira, das constraints e das chaves únicas. Por exemplo: foi convencionado que o nome dos objetos devem obedecer o que está definido na Tabela 5.

Tabela 5 – Convenção para Nome dos Objetos no Banco de Dados

Objeto	Padrão Adotado
Chave Primária	NomeDaTabela_PK
Chave Estrangeira	NomeDaTabela_NomeDaTabelaEstrangeira_FK_nn, onde nn é a sequência de ocorrência do par NomeDaTabela e NomeDaTabelaEstrangeira
Check	NomeDaTabela_CK_nn, onde nn é a sequência de checks da tabela
Chave Única	NomeDaTabela_UK_nn, onde nn é a sequência de chave única da tabela

**Fonte:** Elaborada pelo autor

Para melhor compreensão, as tabelas do banco de dados propostas neste trabalho estão consolidadas na Tabela 6, as quais estão individualmente detalhadas.

Tabela 6 – Tabelas Identificadas neste Trabalho

Tabela do Banco de Dados	Tabela no Documento
Estado	Tabela 7
Cidade	Tabela 8
NomeDaTabelann	Tabela 9

**Fonte:** Elaborada pelo autor

Preencher o quadro a seguir para cada uma das relações identificadas no item Mapeamento OO-Relacional, que deve refletir as características das tabelas a serem criadas no banco de dados. Para melhor compreensão sobre qual a estrutura física das tabelas, seguem dois exemplos: a Tabela 6 é um exemplo da estrutura de dados Estado e a Tabela 6 é um exemplo da estrutura de dados Cidade.

Tabela 7 – Estado

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo	Grupo
id	Number	X	X			
nome	Varchar2(100)	X				1
sigla	Varchar2(2)	X				2

**Fonte:** Elaborada pelo autor

**INSTRUÇÃO:** Grupo é a chave única que você deseja criar (UK\_01, UK\_02)  
Ordem é a sequência dos campos que devem compor a chave única, quando existir mais de um campo na sua chave única

Tabela 8 – Cidade

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo	Grupo
id	Number	X	X			
nome	Varchar2(200)	X				1
idEstado	Number	X		Estado	id	1
latitude	Number	X				2
longitude	Number	X				2

**Fonte:** Elaborada pelo autor

Tabela 9 – NomeDaTabelaNN

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo	Grupo

**Fonte:** Elaborada pelo autor

### 6.2.3 Diagrama de Pacotes

Escreva aqui.

### 6.2.4 Diagrama de Classes de Projeto

Apresentar as classes de projeto identificadas. Alguns materiais para leitura extra sobre classes de projeto podem ser lidos em: <<http://www.ic.uff.br/~anselmo/cursos/ProjSoft/apresentacoes/Projeto%20detalhado%20-%20Classes.pdf>> <[https://edisciplinas.usp.br/pluginfile.php/383727/mod\\_resource/content/2/Aula07\\_DiagramaDeClasse.pdf](https://edisciplinas.usp.br/pluginfile.php/383727/mod_resource/content/2/Aula07_DiagramaDeClasse.pdf)> <<http://www.dsc.ufcg.edu.br/~jacques/cursos/apoo/html/proj1/proj8.htm>>

## 6.3 Projeto Procedimental

Projetar Software é o processo de aplicar várias técnicas e princípios com o propósito de se definir um dispositivo, processo ou sistema, com detalhes suficientes para permitir sua realização física (Taylor-59).

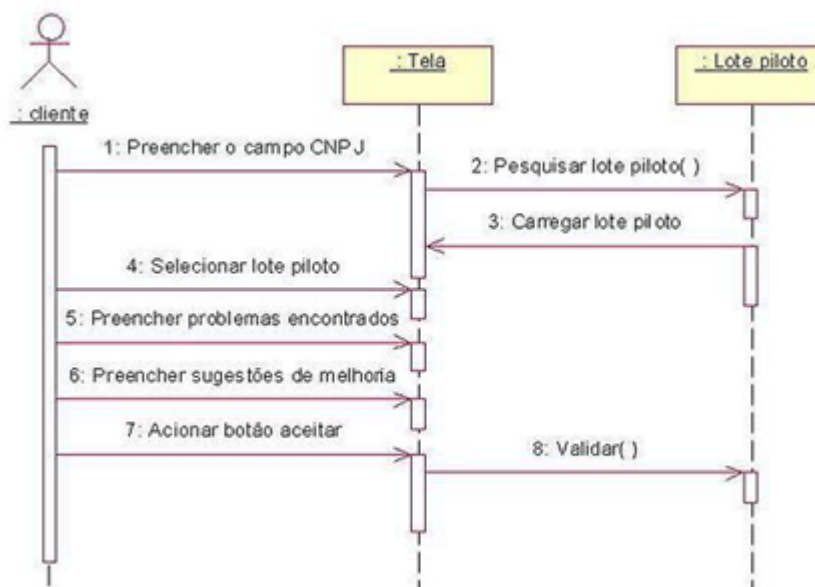
O Projeto de software é o núcleo técnico da Engenharia de Software. É a única maneira de se traduzir "com precisão", os requisitos do usuário para um produto ou sistema acabado. Meta: Traduzir requisitos numa representação de software (Portella).

### 6.3.1 Diagrama de Sequência

Os diagramas de sequência modelam as interações entre objetos em um único caso de uso e ilustram como as diferentes partes de um sistema interagem entre si para realizar

uma função e a ordem em que as interações ocorrem quando um caso de uso é executado (Creately, s.d.), conforme apresentado na Figura 8.

Figura 8 – Diagrama de Sequência



**Fonte:** Elaborada pelo autor

### 6.3.2 Diagrama de Atividades

Havendo necessidade de detalhamento de algum procedimento, este pode ser apresentado na forma de um diagrama de atividades o qual deve ser inserido nessa seção. Podem ser elaborados quantos diagramas de atividades se fizerem necessários.

### 6.3.3 Diagrama de Estados

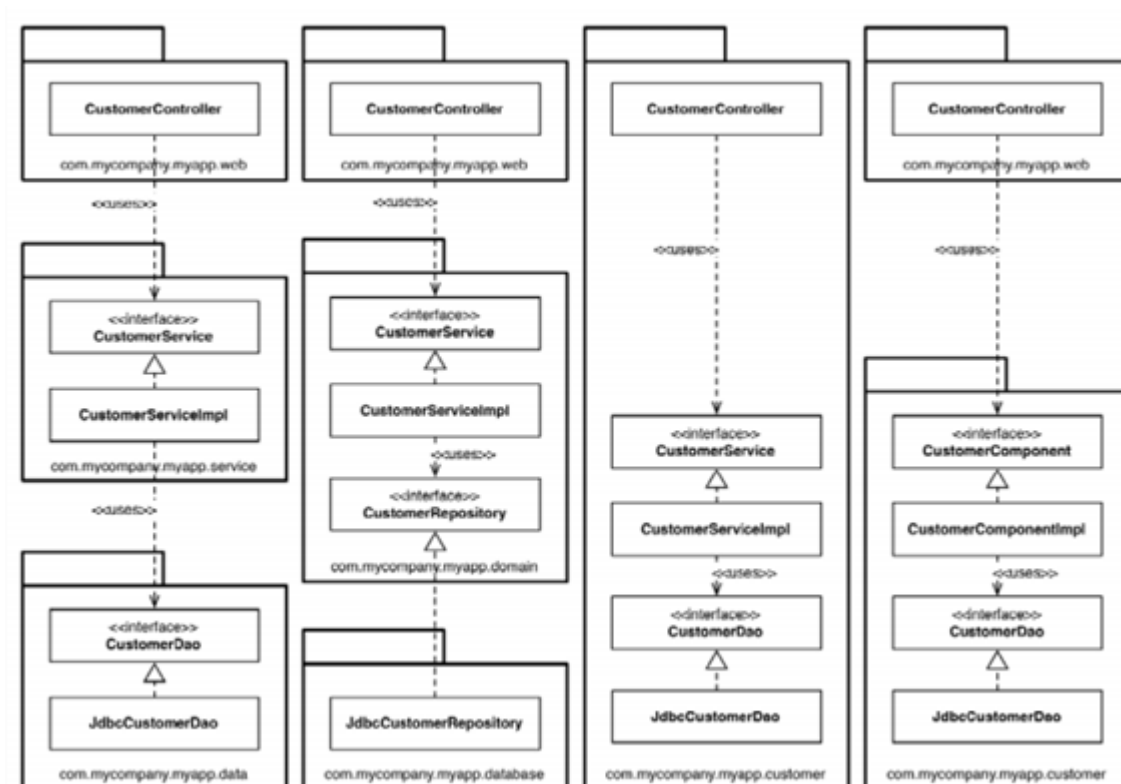
Sendo necessária a elaboração do diagrama de estados na fase de projeto, os novos diagramas devem ser apresentados nessa seção.

## 6.4 Projeto Arquitetural

O projeto arquitetural precede a etapa de construção da obra. O projeto arquitetural determina as partes de uma construção e como estas devem interagir. A arquitetura garante a unidade da obra, ou seja, a consistência entre as suas partes (Vergilio).

Ver algumas definições em (Silva), sendo que um exemplo está apresentado na Figura 9.

Figura 9 – Diferentes Detalhamentos dos Serviços Relacionados ao Cliente



Fonte: Elaborada pelo autor

## 7 IMPLEMENTAÇÃO

Descrever nesse capítulo as características do ambiente de desenvolvimento, servidor de aplicação, servidor de dados, pacotes utilizados etc etc etc.

## 8 TESTE

Descrever nesse capítulo quais e como foram os testes realizados. Os testes podem ser apresentados na forma de casos de teste. Um caso de teste consiste em conjunto de detalhes necessários para se realizar um teste de software.

A seguir encontra-se um modelo para especificação de um caso de teste (CEDRO, 2021):

### **Título**

O título do caso de teste deverá ser sucinto, simples e autoexplicativo com informações para que o Analista de Teste saiba a validação a qual o teste se propõe. Exemplos:

- Validar upload de arquivo;
- Validar cadastro de usuário com perfil administrador;
- Validar envio de ordem de compra.

### **Objetivo**

O objetivo do caso de teste é descrever o que será executado, fornecendo uma visão geral do teste que será realizado. Exemplos:

- Verificar se realiza o upload do arquivo com as extensões permitidas;
- Verificar se o cadastro é efetivado após preencher as informações corretamente;
- Verificar se a ordem de compra é enviada informando o ativo, quantidade e preço.

### **Pré-Condição**

São condições necessárias para que o caso de teste consiga ser executado. Evitar que não tenha alguma informação necessária (Exemplo: solicitar a edição de um usuário em específico e na pré-condição não informar que o usuário deve estar cadastrado). Exemplos:

- Usuário cadastrado e autenticado no sistema;
- Ordem de compra enviada e executada;
- Usuário com perfil Administrador.



**Passos**

Os passos são necessários para descrever todas as ações que o analista deve seguir durante a execução para chegar ao resultado esperado. Devendo iniciar com um verbo infinitivo (acessar, preencher, clicar, verificar) ou imperativo (acesse, preencha, clique, verifique). Exemplos:

- Acessar a tela Negociação > Boleta;
- Clicar no botão “Entrar”;
- Verificar se a edição foi salva no banco de dados;
- Preencher os campos do cadastro.










**Resultados Esperados**

Descrever o comportamento esperado do sistema após executar os passos detalhados. Informar os verbos no presente (valida, apresenta, recupera, retorna). Evitar frases como “O sistema **deve** retornar a mensagem...”, prefira usar “O sistema retorna a mensagem...” para não deixar nenhuma dúvida do resultado esperado. Exemplos:

- Sistema apresenta a tela de edição com os campos preenchidos.;
- A ordem é enviada e executada com o preço informado;
- O cadastro é salvo no banco de dados.

Os casos de teste podem ser especificados usando uma ferramenta de software. Caso este seja o caso, esta seção deve apresentar qual a ferramenta e foi utilizada no desenvolvimento deste trabalho. A Figura 10 apresenta um exemplo de caso de teste especificado usando a ferramenta Testlink.

Figura 10 – Exemplo de Caso de Teste Elaborado na Ferramenta Testlink

Caso de Teste			
 <b>CT-AUT-140: Validar cadastro de cliente</b>			
Versão 1  			
<b>Objetivo do Teste:</b>			
Verificar se realiza o cadastro do cliente informando nome, CPF e telefone.			
<b>Pré-condições</b>			
<ol style="list-style-type: none"><li>1. Usuário cadastrado e autenticado no Portal ABC;</li><li>2. Usuário com perfil Administrador;</li><li>3. Possuir CPF válido.</li></ol>			
  #	<b>Ações do Passo</b>	<b>Resultados Esperados:</b>	 
1	<ol style="list-style-type: none"><li>1 - Acessar a tela de cadastro de cliente no Portal ABC: Menu principal &gt; Cadastros &gt; Cliente;</li><li>2 - Preencher os campos com dados válidos:<ul style="list-style-type: none"><li>• Nome</li><li>• CPF</li><li>• Telefone</li></ul></li><li>3 - Clicar em Salvar;</li><li>4 - Verifique se o cadastro do cliente foi salvo no Banco de dados.</li></ol>	<ol style="list-style-type: none"><li>1 - Sistema exibe a tela de cadastro de cliente com os campos vazios;</li><li>2 - Após salvar o cadastro exibe a mensagem de sucesso: "Cliente cadastrado.";</li><li>3 - O registro do cliente é salvo no Banco de dados.</li></ol>	 

Fonte: Elaborada pelo autor

## 9 IMPLANTAÇÃO

Em qual servidor o sistema está implantado, em qual servidor está a aplicação e em qual servidor está o banco de dados, se foi feito registro de domínio, como deve ser feita a implantação e configuração do sistema, etc etc etc.

Caso o sistema esteja em execução em um ou mais clientes, apresentar nesse capítulo como foi a implantação. Sendo necessário, podem ser usados diagramas UML, como o diagrama de atividades, para apresentar as tarefas de implantação.

## 10 RESULTADOS E DISCUSSÃO

Texto dos resultados.

# 11 TRABALHOS RELACIONADOS

Texto dos trabalhos relacionados.

## 12 CONCLUSÕES/CONCLUSÕES PARCIAIS

Texto das conclusões.

**Obs:** Este capítulo deve ser intitulado “Conclusões Parciais” em trabalhos de graduação para a Validação de Projeto de TCC. Na Avaliação Final de TCC o nome do capítulo deve ser “Conclusões”.

## 13 CRONOGRAMA

Segue abaixo o cronograma das atividades que serão executadas até a Avaliação Final de TCC.

**Obs:** Para facilitar, crie o cronograma usando o modelo do Word contido no projeto (imagens/templateCronograma.docx), ou qualquer outro *software*, salve a imagem e atualize o arquivo imagens/cronograma.png.

		Meses											
		JAN	FEV	MAR	ABR	MAI	JUN	JUL	AGO	SET	OUT	NOV	DEZ
Atividades	1	X	X										
	2		X	X	X								
	3			X	X	X							
	4			X	X								
	5	X	X	X	X	X	X	X	X	X	X	X	X

1. Descrição da atividade 1;
2. Descrição da atividade 2;
3. Descrição da atividade 3;
4. Descrição da atividade 4;
5. Descrição da atividade 5.

**Obs:** Este capítulo deve ser elaborado e estar contido em trabalhos de graduação para a Validação de Projeto de TCC. Na Avaliação Final de TCC este capítulo não deve existir, visto que não haverá atividades após a Avaliação Final.

## REFERÊNCIAS

- ABEDI, Z. et al. Acute, sublethal, and combination effects of azadirachtin and *Bacillus thuringiensis* on the cotton bollworm, *Helicoverpa armigera*. **J. Insect Sci.**, v. 14, p. 30–37, 2014. Citado 2 vezes nas páginas 21 e 22.
- AGAISSE, H.; LERECLUS, D. How does *Bacillus thuringiensis* produce so much insecticidal crystal protein? **J. Bacteriol.**, v. 177, n. 21, p. 6027–6032, 1995. Citado 2 vezes nas páginas 21 e 22.
- AGAPITO-TENFEN, S. Z. et al. Effect of stacking insecticidal cry and herbicide tolerance epsps transgenes on transgenic maize proteome. **BMC Plant Biol**, v. 14, p. 346, 2014. Citado na página 21.
- BRAGA, D. F. M. M. d. S. Algoritmos de processamento da linguagem natural para sistemas de conversao texto-fala em português. 2008. Citado 3 vezes nas páginas 24, 25 e 36.
- CRICKMORE, N. et al. **Bacillus thuringiensis toxin nomenclature**. 2016. Disponível em: <<http://www.btnomenclature.info>>. Acesso em: 04 de out. de 2016. Citado 2 vezes nas páginas 21 e 22.
- NELSON, D. L.; COX, M. M. **Princípios de Bioquímica de Lehninger**. 6. ed. Porto Alegre: Artmed, 2014. 1328 p. Citado 2 vezes nas páginas 21 e 22.



# APÊNDICE A – TÍTULO DO APÊNDICE A

Texto do Apêndice A.

## APÊNDICE B – TÍTULO DO APÊNDICE B

Texto do Apêndice B.

# APÊNDICE C – TÍTULO DO APÊNDICE C

Texto do Apêndice C.

# APÊNDICE D – TÍTULO DO APÊNDICE D

Texto do Apêndice D.

# APÊNDICE E – TÍTULO DO APÊNDICE E

Texto do Apêndice E.

# ANEXO A – TÍTULO DO ANEXO A

Texto do Anexo A.

## ANEXO B – TÍTULO DO ANEXO B

Texto do Anexo B.

## ANEXO C – TÍTULO DO ANEXO C

Texto do Anexo C.



## ANEXO D – TÍTULO DO ANEXO D

Texto do Anexo D.

## ANEXO E – TÍTULO DO ANEXO E

Texto do Anexo E.