**FAI – Centro de Ensino Superior em gestão, Tecnologia e Educação**

**CURSO DE SISTEMAS DE INFORMAÇÃO**

**camila maria da costa**

**Diego vinicius cesar do amaral**

**jéssica souza pivoto**

**Marcos Henrique Azevedo PINTO JÚNIOR**

**Marcos paulo moreno pereira**

**EZMART - SOLUÇÃO NA TOMADA DE DECISÕES**

**SANTA RITA DO SAPUCAÍ- MG**

**2018**

**FAI – Centro de Ensino Superior em gestão, Tecnologia e Educação**

**CURSO DE SISTEMAS DE INFORMAÇÃO**

**camila maria da costa**

**Diego vinicius cesar do amaral**

**jéssica souza pivoto**

**Marcos Henrique Azevedo PINTO JÚNIOR**

**Marcos paulo moreno pereira**

**EZMART – sOLUÇÃO NA TOMADA DE DECISÕES**

Projeto final de curso apresentado a FAI – Centro de Ensino Superior em Gestão, Tecnologia e Educação, como requisito parcial para obtenção do Título de Bacharel em Sistemas de Informação sob a orientação da profa. Eunice Gomes Siqueira.

**SANTA RITA DO SAPUCAÍ- MG**

**2018**

**FOLHA DE APROVAÇÃO**

**HISTÓRICO DE REVISÕES**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autores | Descrição |
| 04/09/2018 | V 2.1.8 | Diego Vinicius César do Amaral  Jéssica Souza Pivoto | Esta revisão é referente à atualização das tabelas de estimativas. Essas tabelas contemplam as estimativas de esforços, de Caso de Uso e de Pontos de Função e seus respectivos valores da Fase 3. |
| 02/09/2018 | V 2.1.7 | Jéssica Souza Pivoto | Esta revisão é referente à atualização dos apêndices que são descritos neste documento. Além disso, refere-se ao incremento de informações baseadas no livro *Design Patterns*, da seção 6.5.1. |
| 27/08/2018 | V 2.1.6 | Jéssica Souza Pivoto | Esta revisão é referente à elaboração antecipada de alguns artefatos que devem ser entregues na Fase 4. Isso inclui o Capítulo 7 até a seção 7.3.2. |
| 25/08/2018 | V 2.1.5 | Camila Maria da Costa  Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto | Esta revisão é referente à elaboração do Capítulo 6, que compõe a 3° Fase do Projeto de Final de Curso. Especificamente, as seções 6.5.2, 6.8.2 e 6.8.3. |
| 18/08/2018 | V 2.1.4 | Camila Maria da Costa  Jéssica Souza Pivoto | Esta revisão é referente à elaboração do Capítulo 6, especificamente da seção 6.5.1. |
| 13/08/2018 | V 2.1.3 | Camila Maria da Costa | Esta revisão é referente à adequação geral do documento, que inclui melhoria do vocabulário, nível de detalhamento das Seções, e mudança no gerenciamento de versões do projeto, que inicialmente era o SVN e atualmente é o GIT. Todas as justificativas encontram-se disponíveis no Capítulo 4, que se trata da gerência do projeto. |
| 11/08/2018 | V 2.1.2 | Jéssica Souza Pivoto | Esta revisão é referente à adequação da seção 6.7 com os as respectivas subseções 6.7.1 e 6.7.2, que se trata da arquitetura de Sistemas Distribuídos. Os pontos a serem adequados nesta revisão foram apontados pelo professor da disciplina de Sistemas Distribuídos. |
| Data | Versão | Autores | Descrição |
| 03/08/2018 | V 2.1.1 | Camila Maria da Costa  Jéssica Souza Pivoto | Esta revisão é referente à adequação do Capítulo 4 de acordo com as correções do Professor Afonso e incremento da seção Estratégias de Varejo *Omnichannel* na Revisão Bibliográfica. |
| 26/07/2018 | V 2.1.0 | Eunice Gomes de Siqueira | Esta revisão é referente revisão parcial da Fase 2. |
| 27/06/2018 | V 2.0.0 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo | Esta revisão é referente à revisão geral deste documento para a segunda entrega, definida pelo cronograma geral proposto na disciplina de Projeto Final de Curso, o que inclui mudanças pontuais nos textos. |
| 16/06/2018 | V 1.0.4 | Diego Vinicius Cesar do Amaral  Marcos Henrique Azevedo | Esta revisão é referente à elaboração da Estimativa por Pontos de Função e a entrevista proposta para 3 possíveis usuários*.* |
| 02/06/2018 | V 1.0.3 | Diego Vinicius Cesar do Amaral | Esta revisão é referente à elaboração dos diagramas, no que diz respeito à arquitetura e projeto do sistema distribuído descritos no Capítulo 6. |
| 26/05/2018 | V 1.0.2 | Jéssica Souza Pivoto | Esta revisão é referente à elaboração dos desafios de sistemas distribuídos com base no livro de Coulouris, Dollimore e Kindberg (2017) e as soluções para o projeto EzMart. |
| 25/05/2018 | V 1.0.1 | Jéssica Souza Pivoto | Esta revisão é referente à elaboração da seção 6.3 até a 6.4 no que diz respeito aos itens descritos. |
| 24/05/2018 | V 1.0.0 | Jéssica Souza Pivoto | Esta revisão é referente à elaboração do Capítulo 6 até a seção 6.2.2 no que diz respeito aos itens descritivos. |
| 23/05/2018 | V 0.9 | Jéssica Souza Pivoto | Esta revisão é referente à elaboração de um tópico da seção de Requisitos (Requisitos não funcionais), que aborda os padrões adotados pelo Governo Eletrônico Brasileiro (e-GOV) para a acessibilidade, responsividade e usabilidade dos do sistema. |

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autores | Descrição |
| 20/05/2018 | V 0.8 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto | Esta revisão é referente à elaboração do modelo lógico, derivado do modelo Entidade-Relacionamento e a elaboração dos itens referentes à gestão da integração. |
| 18/05/2018 | V 0.7 | Jéssica Souza Pivoto | Esta revisão é referente à alteração dos requisitos de gerenciamento de linhas e produtos. |
| 11/05/2018 | V 0.6 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à adequação das atividades e seu sequenciamento na Estrutura Analítica de Projeto (EAP) e à estimativa de esforço por Pontos de Casos de Uso. |
| 06/05/2018 | V 0.5 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à adequação dos Casos de Uso, juntamente com a descrição dos mesmos, que se encontra em Apêndice. Além da adequação do Modelo Entidade- Relacionamento após a especificação dos Requisitos Funcionais. |
| 05/05/2018 | V 0.4 | Jéssica Souza Pivoto | Esta revisão é referente ao complemento do Capítulo 4 que consiste na “Gestão da qualidade” e “Gestão dos riscos”. A tabela de “Gestão dos riscos” contempla esta revisão e se encontra em Apêndice. |
| 01/05/2018 | V 0.3 | Jéssica Souza Pivoto | Esta revisão é referente à adequação do Capítulo 5, de acordo com todos os pontos abordados e/ou sugeridos pela professora orientadora. |

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autores | Descrição |
| 21/04/2018 | V 0.2 | Jéssica Souza Pivoto | Esta revisão é referente à continuação da adequação de todos os demais pontos abordados pela orientadora, conforme correção da entrega V0.0.9. |
| 13/04/2018 | V 0.1 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à adequação dos pontos iniciais abordados pela professora orientadora, conforme correção da entrega V0.0.9. |
| 07/04/2018 | V 0.0.9 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente às mudanças pontuais nos requisitos; revisão dos artefatos e revisão geral da documentação e dos apêndices. |
| 06/04/2018 | V 0.0.8 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à reunião de apoio para entrega da primeira fase/*sprint,* em consequência das mudanças e novos artefatos. |
| 20/03/2018 | V 0.0.7 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo | Esta revisão é referente à elaboração e acompanhamento de mudanças dos casos de uso, assim como do Modelo Entidade-Relacionamento de acordo com as mudanças pontuais nos requisitos funcionais. |
| 16/03/2018 | V 0.0.6 | Jéssica Souza Pivoto | Esta revisão é referente ao documento da primeira fase para o Projeto de Final de Curso, que contempla os Capítulos 3, 4 e 5, assim como seus subitens. |
| 09/03/2018 | V 0.0.4 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à pesquisa de artigos científicos e de base tecnológica para auxílio no desenvolvimento e embasamento do projeto. Com ela, o Capítulo 2 foi descrito. |

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autores | Descrição |
| 06/03/2018 | V 0.0.2 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à documentação das tarefas realizadas na reunião da concepção inicial do projeto, e na planilha de gerência, de acordo com a EAP elaborada no MS-Project com o intuito de documentar as atividades de cada integrante para as fases posteriores. |
| 05/03/2018 | V 0.0.2 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente às atividades de definição dos requisitos funcionais do projeto e dos papéis de cada integrante do projeto com suas respectivas atividades. |
| 09/02/2018 | V 0.0.1 | Diego Vinicius Cesar do Amaral  Jéssica Souza Pivoto  Marcos Henrique Azevedo  Marcos Paulo Moreno Pereira | Esta revisão é referente à fase inicial de concepção do projeto, quando são coletadas informações como a justificativa da proposta; objetivo geral; características principais e desejáveis. |

**AGRADECIMENTOS**

Agradecimentos oportunos à Instituição de Ensino “FAI- Centro de Ensino Superior em Gestão, Tecnologia e Educação”, pelo ambiente criativo, amigável e propício à inovação, assim como pelo suporte tecnológico que fornece um alicerce ao curso de Sistemas de Informação. Agradecimentos ao corpo docente, direção e administração que oportunizam a janela que hoje se vislumbra um horizonte superior, eivado pela confiança no mérito e ética presentes.

Agradecimentos aos professores das disciplinas enredadas neste projeto para conclusão do curso, Júlio Resende Ribeiro, Afonso Celso Soares, Fábio Gavião Avelino de Méllo e aos demais pela orientação na elaboração de todos os artefatos produzidos no presente projeto. Significativos agradecimentos a todos os envolvidos, colaboradores ou patrocinadores, familiares e amigos, por originarem efeito motivacional e impulsionarem a continuidade do desenvolvimento da ideia deste projeto.

**RESUMO**

Em função da necessidade da comercialização, *marketing*, e de revelar os melhores preços, o sistema *Web* EzMartpropõe de maneira ágil, simples e segura a veiculação de produtos, preços, localizações, notícias e promoções dos supermercados. Para que haja o completo desenvolvimento desta entidade, EzMart é desenvolvido através da análise, modelagem e gerenciamento de uma aplicação *Web,* que oferece suporte aos clientes do supermercado e aos estabelecimentos. Por este motivo, a aplicação *Web* é desenvolvida com foco em dois perfis de usuários conhecidos como pessoas físicas e pessoas jurídicas. Ante a necessidade de pessoas físicas, o sistema possui telas de cadastros, compara os melhores preços ao montar listas de compras virtuais e notifica outros usuários a cerca de uma atividade específica, e ante a necessidade de pessoas jurídicas, o sistema anuncia o nome de sua marca e disponibiliza importantes notificações acerca do seu estabelecimento. Desta maneira para o desenvolvimento do arquétipo é empregada a linguagem de programação JAVA *Web*, e incorporação do HTML5, CSS3 e Java Script, em congruência com os padrões de projeto dispostos na diretriz do curso de Sistemas de Informação, assim como para acondicionar informações de forma segura, o Banco de Dados *Postgres*. Portanto, a expansão deste trabalho colabora na realização de técnicas e experiências adquiridas no curso de graduação de Sistemas de Informação, que engloba as disciplinas compreendidas ao decorrer do curso, cujos conceitos são abordados de maneira conceitual e prática com o objetivo de introduzir e finalizar cada artefato desenvolvido para o cumprimento deste trabalho.

**Palavras-chave:** Aplicação *Web*. Supermercados. Mercado varejista. Tecnologia. Sistemas de Informação.

**LISTA DE FIGURAS**

[Figura 1 - Evolução da tecnologia da informação e comunicação nas organizações durante o séc. XX 28](#_Toc523855349)

[Figura 2 - Categorias mais vendidas nos supermercados 30](#_Toc523855350)

[Figura 3 - Faturamento por região 30](#_Toc523855351)

[Figura 4 - Faturamento por Unidade da Federação 31](#_Toc523855352)

[Figura 5 - Participação dos tamanhos no total de lojas 31](#_Toc523855353)

[Figura 6 - Principais formas de investimento 32](#_Toc523855354)

[Figura 7 - Formas de pagamento mais utilizadas 33](#_Toc523855355)

[Figura 8 - Atributos para escolha do local das compras 35](#_Toc523855356)

[Figura 9 - Participação dos supermercados, por seção, nas compras dos consumidores 37](#_Toc523855357)

[Figura 10 - Mapeamento dos consumidores em relação a loja 39](#_Toc523855358)

[Figura 11 - Porcentagem acumuladas de clientes em raio concêntricos 40](#_Toc523855359)

[Figura 12 - Curva de área de influência 41](#_Toc523855360)

[Figura 13 - Relação entre as partes interessadas e o projeto 53](#_Toc523855361)

[Figura 14 - Poder X Interesse do projeto EzMart 54](#_Toc523855362)

[Figura 15 - Modelo Incremental 57](#_Toc523855363)

[Figura 16 - Quadro Kanban 63](#_Toc523855364)

[Figura 17 - Visão geral do gerenciamento do escopo de um projeto 69](#_Toc523855365)

[Figura 18 - Definir o escopo: entradas, ferramentas e técnicas, e saídas 71](#_Toc523855366)

[Figura 19 - Diagrama do fluxo de dados do processo “Definir escopo” 71](#_Toc523855367)

[Figura 20 - Exemplo de uma matriz de rastreabilidade de requisitos 73](#_Toc523855368)

[Figura 21 - Diagrama do fluxo de dados do processo de planejar o gerenciamento do cronograma 77](#_Toc523855369)

[Figura 22 - Definir atividades: entradas, ferramentas e técnicas, e saída 78](#_Toc523855370)

[Figura 23 - Método do diagrama de precedência – tipos de relações 81](#_Toc523855371)

[Figura 24 - Exemplo de diagrama de rede de um projeto 81](#_Toc523855372)

[Figura 25 - Visão geral do gerenciamento da integração de um projeto 85](#_Toc523855373)

[Figura 26 - Gerenciador de versões do EzMart 91](#_Toc523855374)

[Figura 27 - Opções que o TortoiseSVN oferece como recursos para o gerenciamento de versões 92](#_Toc523855375)

[Figura 28- Repositório GIT do projeto EzMart 93](#_Toc523855376)

[Figura 29- Clone ou download do repositório do projeto EzMart 94](#_Toc523855377)

[Figura 30 - Realizar o controle integrado de mudanças: entradas ferramentas e saída 96](#_Toc523855378)

[Figura 31 - Visão geral do gerenciamento da qualidade do projeto 99](#_Toc523855379)

[Figura 32- Visão geral do gerenciamento dos riscos do projeto 104](#_Toc523855380)

[Figura 33 - Planejar o gerenciamento dos riscos: entradas, ferramentas e técnicas, e saídas 107](#_Toc523855381)

[Figura 34 - Diagrama do fluxo dos dados do processo “Planejar o gerenciamento dos riscos” 108](#_Toc523855382)

[Figura 35 - Tipos de requisitos funcionais 142](#_Toc523855383)

[Figura 36 - Comportamento de um site responsivo 145](#_Toc523855384)

[Figura 37 - Tipos de resoluções de telas 145](#_Toc523855385)

[Figura 38 - Exemplo de padrão na convenção de codificação Java 147](#_Toc523855386)

[Figura 39- Padão MVC 162](#_Toc523855387)

[Figura 40- Computadores registrados na Internet 171](#_Toc523855388)

[Figura 41-Computadores e servidores na Internet 172](#_Toc523855389)

[Figura 42- Exemplo da arquitetura cliente-servidor 179](#_Toc523855390)

[Figura 43- Diagrama de sistemas distribuídos 181](#_Toc523855391)

[Figura 44- A importância das cores 186](#_Toc523855392)

[Figura 45- Logotipo do sistema EzMart 187](#_Toc523855393)

[Figura 46- Página de listagem de linhas do sistema EzMart na versão V 1.0 188](#_Toc523855394)

[Figura 47- Página de cadastro para um usuário consumidor V 3.0 190](#_Toc523855395)

[Figura 48- Página de listagem de linhas com opções de alterar e excluir V 3.0 191](#_Toc523855396)

[Figura 49- Recurso de Breadcrumb do sistema EzMart V 3.0 192](#_Toc523855397)

[Figura 50- Mensagens informativas sobre campos obrigatórios 193](#_Toc523855398)

**LISTA DE QUADROS**

[Quadro 1- Quadro dos integrantes 66](#_Toc523855424)

[Quadro 2- Recursos de hardware 66](#_Toc523855425)

[Quadro 3- Recursos de Software 67](#_Toc523855426)

[Quadro 4- Resumo do esforço 82](#_Toc523855427)

[Quadro 5- Estimativas do esforço 153](#_Toc523855428)

[Quadro 6- Padrões de projeto 160](#_Toc523855429)

[Quadro 7-Documentos relevantes para os testes do sistema EzMart 195](#_Toc523855430)

[Quadro 8- Equipamentos utilizados nos testes do sistema EzMart 195](#_Toc523855431)

[Quadro 9- Recursos de software para os testes do sistema EzMart 196](#_Toc523855432)

[Quadro 10- Requisitos do sistema EzMart submetidos à testes 197](#_Toc523855433)

[Quadro 11- Rastreabilidade entre requisitos e casos de testes 199](#_Toc523855434)

**LISTA DE SIGLAS E ABREVIATURAS**

|  |  |
| --- | --- |
| ABNT - | Associação Brasileira de Normas Técnicas |
| AMA - | *American Marketing Association* |
| CASE - | *Computer-Aided Software Engineering* |
| CPF - | Cadastro de Pessoa Física |
| CNPJ - | Cadastro de Pessoa Jurídica |
| CSS - | *Cascading Style Sheets* |
| DAO - | *Data Access Object* |
| DDL - | *Data Definition Language* |
| DML - | *Data Manipulation Language* |
| EAP - | Estrutura Analítica do Projeto |
| E-GOV - | Governo Eletrônico Brasileiro |
| FAI - | FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação |
| GIS - | *Geographic Information System* |
| GUI - | *Graphic User Interface* |
| JPG - | *Joint Photographic Group* |
| HTML - | *Hypertext Markup Language* |
| HTTP - | *Hipertext Transfer Protocol* |
| IBGE - | Instituto Brasileiro de Geografia e Estatística |
| IDE - | *Integrated Development Environment* |
| IEEE - | *Institute of Electrical and Electronics Engineers* |
| IP - | *Internet Protocol* |
| JDBC - | *Java Database Connectivity* |
| MER - | Modelo Entidade-Relacionamento |
| MVC - | *Model-View-Control* |
| PDF - | *Portable Document File* |
| PIB - | Produto Interno Bruto |
| PMBOK - | *Project Management Body of Knowledge* |
| PMI - | *Project Management Institute* |
| PROCON- | Programa de Proteção e Defesa do Consumidor |
| PNG - | *Portable Network Graphics* |
| SAD - | Sistema de Apoio à Decisão |
| SGBD - | Sistema Gerenciador de Banco de Dados |
| SPC - | Sistema de proteção ao crédito |
| SQL - | *Structured Query Language* |
| TCP - | *Transmission Control Protocol* |
| TI - | Tecnologia da Informação |
| UML - | *Unified Modeling Language* |

**SUMÁRIO**

[1 INTRODUÇÃO 20](#_Toc523855435)

[2 REVISÃO BIBLIOGRÁFICA 24](#_Toc523855436)

[2.1 PROCESSO DECISÓRIO 24](#_Toc523855437)

[2.1.1 Níveis de Decisão e os Sistemas de Informação Automatizados 25](#_Toc523855438)

[2.1.2 Tecnologias da Informação que Apoiam a Tomada de Decisão 27](#_Toc523855439)

[2.2 SETOR SUPERMERCADISTA 28](#_Toc523855440)

[2.2.1 Comportamento do Consumidor no Setor Supermercadista 33](#_Toc523855441)

[2.2.2 Área de Influência do Estabelecimento 37](#_Toc523855442)

[2.2.3 Estratégias de Varejo *Omnichannel* 42](#_Toc523855443)

[2.3 TRABALHOS RELACIONADOS 43](#_Toc523855444)

[2.3.1 Sistemas de Apoio à Decisão no Varejo 43](#_Toc523855445)

[2.3.2 Negócios baseados em Tecnologias *Web* 44](#_Toc523855446)

[2.3.2.1 Trabalho: Desenvolvimento de um sistema de colaboração em massa para reduzir gastos dos consumidores em compras de supermercado 45](#_Toc523855447)

[2.3.2.2 Trabalho: Gastometro 45](#_Toc523855448)

[3 O OBJETIVO DO PROJETO 47](#_Toc523855449)

[3.1 FORMULAÇÃO DO PROBLEMA 47](#_Toc523855450)

[3.2 OBJETIVOS 48](#_Toc523855451)

[3.3 JUSTIFICATIVA 49](#_Toc523855452)

[3.4 PÚBLICO ALVO 50](#_Toc523855453)

[3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS 50](#_Toc523855454)

[4 GERÊNCIA DE PROJETO 51](#_Toc523855455)

[4.1 PLANO DE PROJETO 51](#_Toc523855456)

[4.1.1 Partes Interessadas 52](#_Toc523855457)

[4.1.2 Modelo de Ciclo de Vida 55](#_Toc523855458)

[4.1.3 Recursos Necessários 64](#_Toc523855459)

[4.2 ÁREAS DE CONHECIMENTO 67](#_Toc523855460)

[4.2.1 Gestão do Escopo 68](#_Toc523855461)

[4.2.2 Gestão do Tempo 75](#_Toc523855462)

[4.2.3 Gestão da Integração 83](#_Toc523855463)

[4.2.4 Gestão da Qualidade 98](#_Toc523855464)

[4.2.5 Gestão dos Riscos 102](#_Toc523855465)

[5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS 109](#_Toc523855466)

[5.1 REQUISITOS DO SISTEMA DE SOFTWARE 109](#_Toc523855467)

[5.1.1 Requisitos Funcionais 110](#_Toc523855468)

[5.1.2 Requisitos Não Funcionais 141](#_Toc523855469)

[5.2 ANÁLISE DOS REQUISITOS 150](#_Toc523855470)

[5.2.1 Modelos de Caso de Uso 151](#_Toc523855471)

[5.2.2 Modelo Conceitual dos Dados 152](#_Toc523855472)

[5.3 MÉTRICAS PARA ESTIMATIVA DO ESFORÇO 152](#_Toc523855473)

[6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE 154](#_Toc523855474)

[6.1 VISÃO ESTRUTURAL 155](#_Toc523855475)

[6.1.1 Diagrama de Pacotes 155](#_Toc523855476)

[6.1.2 Diagrama de Classes 155](#_Toc523855477)

[6.1.3 Diagrama de Objetos 156](#_Toc523855478)

[6.2 VISÃO COMPORTAMENTAL 156](#_Toc523855479)

[6.2.1 Projeto das Interações 156](#_Toc523855480)

[6.2.2 Diagrama de Atividades 157](#_Toc523855481)

[6.3 VISÃO DE DADOS 157](#_Toc523855482)

[6.3.1 Modelo Operacional 158](#_Toc523855483)

[6.3.2 Dicionário de Dados do Modelo Operacional 158](#_Toc523855484)

[6.4 VISÃO FÍSICA 158](#_Toc523855485)

[6.4.1 Diagrama de Componentes 158](#_Toc523855486)

[6.4.2 Estrutura Física do Banco de Dados 159](#_Toc523855487)

[6.5 PADRÕES, CONVENÇÕES E GUIAS 159](#_Toc523855488)

[6.5.1 *Design Patterns* 159](#_Toc523855489)

[6.5.2 Convenções e Guias para Codificação 163](#_Toc523855490)

[6.6 ANÁLISE DE COMPLEXIDADE 164](#_Toc523855491)

[6.7 PROJETO DE SISTEMAS DISTRIBUÍDOS 165](#_Toc523855492)

[6.7.1 Procedimentos para Tratamento dos Desafios 166](#_Toc523855493)

[6.7.2 Tecnologias e Arquiteturas de Distribuição 177](#_Toc523855494)

[6.8 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR 182](#_Toc523855495)

[6.8.1 Perfil de Usuário 183](#_Toc523855496)

[6.8.2 Aspectos Visuais da Interface de Usuário 184](#_Toc523855497)

[6.8.3 Heurísticas de Usabilidade 189](#_Toc523855498)

[7 PLANO DE TESTES 194](#_Toc523855499)

[7.1 Finalidade 194](#_Toc523855500)

[7.2 Escopo 194](#_Toc523855501)

[7.2.1 Referências a Documentos Relevantes 194](#_Toc523855502)

[7.2.2 Ambiente para a Realização dos Testes 195](#_Toc523855503)

[7.3 Especificação dos casos de teste 196](#_Toc523855504)

[7.3.1 Item a Testar 196](#_Toc523855505)

[7.3.2 Rastreabilidade entre Requisitos e Casos de Teste 197](#_Toc523855506)

[7.3.3 Descrição dos Casos de Teste 199](#_Toc523855507)

[7.4 Resultado dos casos de teste 199](#_Toc523855508)

[7.4.2 Resultados 199](#_Toc523855509)

[8 PLANO DE IMPLANTAÇÃO 200](#_Toc523855510)

[8.1 METODOLOGIA 200](#_Toc523855511)

[8.1.1 Descrição da Metodologia 200](#_Toc523855512)

[8.1.2 Matriz de Responsabilidades 200](#_Toc523855513)

[8.2 TREINAMENTOS PREVISTOS 200](#_Toc523855514)

[8.3 CRONOGRAMA DE IMPLANTAÇÃO 200](#_Toc523855515)

[8.4 DOCUMENTO DE APOIO À IMPLANTAÇÃO 200](#_Toc523855516)

[8.5 VISÃO DE IMPLANTAÇÃO 200](#_Toc523855517)

[9 CONCLUSÃO 201](#_Toc523855518)

[REFERÊNCIAS 202](#_Toc523855519)

[OBRAS CONSULTADAS 208](#_Toc523855520)

[APÊNDICE A – ESTRUTURA ANALÍTICA DE PROJETO (EAP) 211](#_Toc523855521)

[APÊNDICE B – CRONOGRAMA DO PROJETO 212](#_Toc523855522)

[APÊNDICE C – RELATÓRIO DE DESEMPENHO 213](#_Toc523855523)

[APÊNDICE D – PLANILHA PARA ACOMPANHAMENTO DOS RISCOS DE AMEAÇA DO PROJETO 214](#_Toc523855524)

[APÊNDICE E – DIAGRAMAS DE CASOS DE USO 215](#_Toc523855525)

[APÊNDICE F – DESCRIÇÃO DOS CASOS DE USO 216](#_Toc523855526)

[APÊNDICE G – DIAGRAMA ENTIDADE RELACIONAMENTO 217](#_Toc523855527)

[APÊNDICE H – ESTIMATIVA DE PONTOS DE FUNÇÃO E PONTOS DE CASO DE USO 218](#_Toc523855528)

[APÊNDICE I – LISTA DE VERIFICAÇÃO DA QUALIDADE 219](#_Toc523855529)

[APÊNDICE J – DIAGRAMA DE PACOTES 220](#_Toc523855530)

[APÊNDICE K – DIAGRAMA DE CLASSES 221](#_Toc523855531)

[APÊNDICE L – DIAGRAMA DE OBJETOS 222](#_Toc523855532)

[APÊNDICE M – DIAGRAMA DE SEQUÊNCIA 223](#_Toc523855533)

[APÊNDICE N – DIAGRAMA DE VISÃO GERAL DA INTERAÇÃO 224](#_Toc523855534)

[APÊNDICE O – DIAGRAMA DE ATIVIDADES 225](#_Toc523855535)

[APÊNDICE P – MODELO OPERACIONAL DOS DADOS 226](#_Toc523855536)

[APÊNDICE Q – DICIONÁRIO DE DADOS 227](#_Toc523855537)

[APÊNDICE R – DIAGRAMA DE COMPONENTES 228](#_Toc523855538)

[APÊNDICE S – ESTRUTURA FÍSICA DO BANCO DE DADOS 229](#_Toc523855539)

[APÊNDICE T – MANUAIS DE INSTALAÇÃO E CONFIGURAÇÃO 230](#_Toc523855540)

[APÊNDICE U – MANUAL DO USUÁRIO 231](#_Toc523855541)

[APÊNDICE V – DIAGRAMA DE IMPLANTAÇÃO 232](#_Toc523855542)

[APÊNDICE X – DIAGRAMAS UML UTILIZADOS NESTE PROJETO 233](#_Toc523855543)

[APÊNDICE W– HISTÓRICO DE REALIZAÇÃO 234](#_Toc523855544)

[APÊNDICE Y– ENTREVISTA COM O USUÁRIO 235](#_Toc523855545)

[APÊNDICE Z– SOLICITAÇÕES DE MUDANÇAS NO PROJETO 236](#_Toc523855546)

# 1 INTRODUÇÃO

As tecnologias de informação e comunicação influenciam na forma com que os consumidores pensam e agem no momento de uma compra. A Internet tornou a tela do computador uma vitrine, cheia de opções e promoções, além de abrir espaço para as redes sociais disseminarem opiniões e relatarem fatos sobre produtos e processos de aquisição. Diante desta realidade, o varejo passa por uma ruptura de conceitos tradicionais para se adaptar às novas condições de seu interlocutor mais importante: o cliente.

Segundo Silva e Teixeira (2002, p. 70), a informação tem “papel crescente no presente e no futuro das empresas”. Ela é a principal entrada do processo decisório do qual emanam decisões para os níveis estratégicos, táticos e operacionais que permitem atingir os objetivos de uma organização e a dinamização de suas atividades. As organizações devem ficar atentas à integração de suas áreas funcionais, para que a informação flua em todos os níveis de decisão, utilizando-se de tecnologias para melhor dispor de informações qualificadas e atualizadas.

No comércio varejista, as tecnologias de informação apoiam os pequenos, médios ou grandes negócios. Elas podem ser usadas para melhorar o fluxo das informações entre os processos, para a publicidade e propaganda de produtos e serviços oferecidos e, principalmente, para subsidiarem a tomada de decisão de compra de consumidores e clientes.

É neste contexto, que se idealizou e se desenvolveu o projeto EzMart, que este documento descreve. EzMart visa atender dois importantes aspectos: o auxílio na tomada de decisões de um consumidor do mercado varejista e o auxílio do *marketing* para os estabelecimentos deste mercado. Em função da necessidade da comercialização, *marketing*, e de revelar os melhores preços, a aplicação *Web* em foco propõe de maneira ágil, simples e segura a veiculação de produtos, preços, localizações, notícias e promoções dos supermercados e para que haja o completo desenvolvimento desta entidade, o foco é voltado em analisar, modelar, gerenciar e desenvolver uma aplicação *Web,* para suporte aos clientes do supermercado e a esses estabelecimentos. Por este motivo, a aplicação *Web* possui telas de cadastros, ante a necessidade de pessoas físicas ao comparar os melhores preços, ao montar listas de compras virtuais e ao notificar outros usuários a cerca de uma atividade específica. E ante a necessidade de pessoas jurídicas ao anunciar o nome de sua marca e disponibilizar importantes notificações acerca do seu estabelecimento.

Neste documento, o leitor encontra a concepção do projeto, justificada por bases sólidas de pesquisas, os objetivos, os requisitos e os recursos técnicos empregados na etapa de desenvolvimento do trabalho.

O Capítulo 1 traz a introdução do projeto, que tem como foco principal orientar o leitor sobre o tema do projeto, assim como importantes vertentes. Essas vertentes dirigem o leitor à compreensão dos benefícios para a sociedade, assim como propõe uma maneira de instigá-lo a conhecer o importante papel que a tecnologia da informação proporciona para o varejo. Outra importante vertente é a estruturação deste documento em Capítulos.

O Capítulo 2 aborda os conceitos e fundamentos técnico-científicos obtidos por meio da revisão bibliográfica sobre o tema.

O Capítulo 3 refere-se formulação do problema e o objetivo geral deste projeto. A primeira seção desse Capítulo refere-se, primeiramente, à formulação do problema mediante as pesquisas realizadas. A segunda seção refere-se aos objetivos. Já a terceira seção comprova, por meio de justificativas plausíveis, a importância do projeto para a resolução do problema encontrado e descrito no início desse Capítulo. A quarta seção apresenta público alvo do projeto, explicando qual o perfil do público que têm interesse no sistema de software desenvolvido. A quinta e última seção descreve todos os níveis de decisão e os grupos funcionais que são atendidos pelo projeto.

O Capítulo 4 trata da gerência aplicada ao projeto, mostrando como devem ser gerenciados todos os recursos, sejam eles físicos, humanos ou de tempo. Desta maneira, o plano de projeto compreendendo as partes interessadas (ou *stakeholders)* está na primeira seção desse Capítulo. A segunda seção proporciona ao leitor o conhecimento sobre o modelo de ciclo de vida adotado para a execução do projeto. Finalizando o Capítulo 4, a última seção expõe de maneira organizada as áreas de conhecimento: escopo, tempo, integração, qualidade e riscos, de acordo com o guia de conhecimento do *Project Management Institute* (PMBoK).

O Capítulo 5 é de importância relevante, por trazer os requisitos especificados para o sistema de software. O Capítulo traz na primeira seção os requisitos funcionais (RF) e na segunda seção os requisitos não funcionais (RNF). Entretanto, para que esses requisitos sejam compreendidos há a representação de todos eles em modelos. Os modelos representativos para o sistema a ser desenvolvido são: o modelo de caso de uso e o modelo conceitual dos dados, que estão na terceira e última seção.

O Capítulo 6 mostra as visões obtidas por meio de diagramas da *Unified Modeling Language* (UML) e os padrões, as convenções e guias utilizados no desenvolvimento do sistema de software. A primeira visão é a visão estrutural e contempla os diagramas de pacotes, classes e objetos. A segunda visão é a visão comportamental, que contempla o projeto das interações e o diagrama de atividades. A terceira visão é a visão lógica dos dados persistidos pelo sistema, que mostra o modelo operacional e o dicionário de dados do modelo operacional. A fim de que os códigos-fonte do sistema de software desenvolvido sejam de fácil manutenção, alta legibilidade e compreensão, são necessários padrões, convenções e guias. *Design* *Patterns* e convenções para codificação para o código-fonte estão na seção seguinte. Em continuidade do Capítulo, há também a seção com uma análise da complexidade de partes do código-fonte desenvolvido. A penúltima seção do Capítulo 6 descreve os procedimentos para tratamento de desafios no projeto de um sistema distribuído e as tecnologias e arquiteturas de distribuição que são incorporadas ao projeto. A última seção trata do projeto da interação humano-computador, fornecendo elementos específicos a respeito do perfil do usuário final, assim como os aspectos visuais da interface de usuário e as heurísticas de usabilidade, que são evidentemente importantes para viabilizar o uso do sistema por todos os perfis de usuário.

O Capítulo 7 trata da finalidade do plano de testes e sua importância para que a qualidade do projeto seja preservada em todas suas fases. Por sua vez, os testes devem possuir referências e documentos relevantes que especifiquem os cenários e o ambiente em que são realizados. Portanto, as seções desse Capítulo descrevem quais itens e/ou cenários devem ser testados, qual a rastreabilidade entre os requisitos e os casos de testes e qual o resultado dos testes, que impactam diretamente na qualidade do sistema final.

O Capítulo 8 contempla o plano de implantação e como seções destacam-se a descrição da metodologia, a matriz de responsabilidade, os treinamentos previstos, o cronograma de implantação e seu documento e a visão geral da implantação. O planejamento da implantação do sistema final leva em consideração as necessidades de recursos de hardware e de software para o pleno funcionamento e do sistema em seu ambiente de operação.

Não menos importante que a introdução ao assunto, é a conclusão retirada ao longo de todo tempo utilizado para a execução deste projeto. No Capítulo 9 estão os relatos das dificuldades encontradas, assim como os caminhos encontrados para resolução dos problemas, tornando-se consideráveis informações para outros projetos que venham a ser realizados.

Por fim, as referências e as obras consultadas complementam o embasamento científico que sustenta o projeto. Nos apêndices são encontrados outros artefatos produzidos que auxiliam no entendimento de todos os Capítulos deste documento.

# 2 REVISÃO BIBLIOGRÁFICA

O projeto EzMart visa oferecer ao mercado varejista, em particular, os supermercados e seus consumidores e clientes, um meio eficaz para tomada de decisão baseada em tecnologias de informação e comunicação relevantes.

Neste Capítulo está a fundamentação teórica que se constitui na base para a realização do projeto. A seção 2.1 aborda as etapas do processo decisório. A seção 2.2 destaca o setor supermercadista do varejo brasileiro. A seção 2.3 exibe os trabalhos relacionados ao projeto e que apresentam soluções ou estudos semelhantes.

## 2.1 PROCESSO DECISÓRIO

Tomar decisões é algo crucial independente de qual seja a situação e requer a todo tempo um esforço tanto dos administradores das organizações quanto dos clientes do negócio. Os administradores estão focados em tomar decisões de forma rápida, analisando dados e obtendo informações úteis, descentralizando responsabilidades e aumentando o nível de conhecimento em todos os setores da organização. Já os clientes, estão focados em decidir entre todas as opções disponíveis, qual delas é a melhor, a que tem um melhor preço, uma melhor qualidade, dentre outros.

O grande desafio para o sucesso de um empreendimento é a tomada de decisão baseada em informações confiáveis, precisas, atuais e completas e, nesse sentido, a empresa é uma das partes do ramo empresarial onde a demanda pela análise rápida de informações acontece diariamente.

Segundo Pereira e Fonseca (2009), o conhecimento e a situação existencial do decisor no momento em que são tomadas as decisões são faces de uma mesma moeda e influenciam diretamente a avaliação e a escolha das alternativas para solução de um problema.

De acordo com Choo (1998), o processo decisório é dirigido pela busca de alternativas que sejam boas o suficiente, em detrimento da busca pela melhor alternativa existente.

Para Simon (1965), existem 6 fases não lineares no processo decisório: inteligência, concepção, escolha, implementação da decisão, monitoramento e revisão. As teorias contemporâneas do processo decisório vêm mudando e se adaptando ao contexto, mas tomam como referência os estudos da racionalidade limitada de Simon. Por isso, Godoy e Ruggiero (2006, p.167) afirmam:

Por se tratar de uma atividade não descartável e de fundamental importância, os estudos do processo decisórios refletem a necessidade da busca permanente do aprimoramento da capacidade de decidir. Neste sentido, decidir pode ser visto como uma escolha entre alternativas que, na maioria das vezes, são afetadas por situações futuras sobre as quais não se têm controle.

Neste sentido, é de total importância que o gestor procure se atualizar, assim como os clientes do negócio, para que conheçam meios que os ajudem a tomar decisões mais acertadas.

Para Oliveira (1992), a informação auxilia no processo decisório, pois quando a informação é devidamente estruturada torna-se essencial para a empresa, que deseja impetrar os seus objetivos.

Devido à complexidade que envolve o processo decisório nas empresas, as tecnologias da informação são mecanismos muito apropriados para conduzir às melhores decisões que podem influenciar um negócio diretamente.

Segundo Batista (2004), existem dois elementos fundamentais para a tomada de decisão: os canais de informação e as redes de comunicação. Os canais de informação referem-se aos locais onde se podem retirar os dados que serão utilizados pelas redes, que são responsáveis pela disseminação das informações. Por sua vez, os canais de informação podem ser referenciados como sendo também sistemas de apoio chamados de Sistema de Informação Gerencial (SIG) e Sistema de Apoio à Decisão (SAD).

### 2.1.1 Níveis de Decisão e os Sistemas de Informação Automatizados

Os níveis de decisão são amplamente dependentes apesar de distintos, trabalhando de forma integrada para alcançar resultados dentro da organização.

Segundo o Portal da Educação (2013), cada nível de decisão é definido da seguinte forma:

1. estratégico: é o nível mais elevado da hierarquia, compreende presidentes, diretores e demais gestores da alta cúpula e decidem os objetivos globais da empresa. Eles analisam as tendências do mercado, a situação financeira e a sua influência na organização, as mudanças de comportamento do consumidor e definem as estratégias políticas e financeiras para adaptar a organização às adversidades encontradas;
2. tático: neste nível estão inclusos os gerentes ou chefes de cada setor. Nessa posição, as tarefas a serem desempenhadas são todas da área organizacional (financeiro, recursos humanos, linhas diferenciadas de produtos, etc.). Esse nível administrativo é também responsável pela implementação das decisões estratégicas tomadas pelo nível acima;
3. operacional: nesse nível os administradores devem extrair o máximo das potencialidades do colaborador por meio das ferramentas administrativas que lhe são concedidas, seu papel é aperfeiçoar a produção de bens e serviços de maneiras satisfatórias em curto prazo, sempre seguindo as diretrizes estabelecidas no nível tático. Dentre os cargos inerentes a essa posição estão os chefes de equipe e os supervisores.

De acordo com Florenzano (2015), para cada nível organizacional existe um tipo de sistema de informação.

Silva e Teixeira (2002, p.72) denotam que SADs como “sistemas de informações ou modelos analíticos projetados para ajudar a gerenciar profissionais a tomar decisões mais eficazes”. Segundo Chaves e Falsarella (1995), para desenvolver um SAD é necessário construir um ambiente de apoio à decisão. Em alguns casos, isso não significa construir um novo sistema de informação, mas, sim, incorporar aos sistemas existentes ambientes aplicativos (que façam análise de alternativas e forneçam soluções aos problemas) e/ou ferramentas de apoio à decisão (que auxiliam na simulação de situações, na representação gráfica das informações etc.) que forneçam informações e subsídios para o processo de tomada de decisão.

Silva e Teixeira (2002, p.72) afirmam ainda, que um Sistema de Informação Executiva (SIE), “é um sistema de informação desenhado para facilitar a tomada de decisões dos executivos ou da alta direção”. Segundo Chaves e Falsarella (1995, p.27), as principais características e funções desses sistemas são: gerar mapas, gráficos e dados que possam ser submetidos à análise estatística para suprir os executivos com informações comparativas, fáceis de entender; permitir que o executivo se comunique com o mundo interno e externo, por meio de interfaces amigáveis (correio eletrônico, teleconferência etc); oferecer ao executivo, ferramentas de organização pessoal e de gerenciamento de projetos, tarefas e pessoas.

De acordo com Silva e Teixeira (2002, p.71):

Um sistema de Informação Gerencial (SIG) são sistemas que fornecem informações integradas e sumarizadas, provenientes de diversos sistemas transacionais que possibilitam aos gerentes do nível médio visualizar o desempenho de seu departamento e, mesmo, da organização como um todo.

E, de acordo com Chaves e Falsarella (1995, p. 26), as principais funções e características dos SIGs são: integrar dados das diversas aplicações e transformá-los em informação; suprir gerentes com informações para que estes possam comparar o desempenho atual da organização com o que foi planejado; produzir relatórios que auxiliem os gerentes na tomada de decisões.

Silva e Teixeira (2002, p. 71) afirmam que:

Um Sistema de Processamento de Transações (STP) são tradicionais sistemas de processamento de dados, ou como também são chamados sistemas transacionais. O próprio nome define cuida das transações básicas da organização. O início do processo de informatização de qualquer empresa é baseado no desenvolvimento e na implantação desses sistemas, que realizam e registram as operações diárias de rotina, necessárias à operação da empresa. Os exemplos típicos são os sistemas de folha de pagamento, compras, contas a receber, controle de estoque etc.

Rossetti e Morales (2007) afirmam que pela ótica do conhecimento e na busca de orientar indivíduos e empresas a conquistar espaços no mercado, muitas pesquisas estão sendo conduzidas, e muitas delas enfatizam a informação e o conhecimento como sendo hoje, os bens de maior valor.

### 2.1.2 Tecnologias da Informação que Apoiam a Tomada de Decisão

A tecnologia da informação tem sido intensamente empregada como instrumento para os mais diversos fins. Rossetti e Morales (2007) afirmam que a tecnologia da informação é utilizada por indivíduos e organizações, para acompanhar a velocidade com que as transformações vêm ocorrendo no mundo; para aumentar a produção, melhorar a qualidade dos produtos; como suporte à análise de mercados; para tornar ágil e eficaz a interação com mercados, com clientes e até com competidores.

De acordo com Rossetti e Morales (2007, p. 1), a tecnologia da informação e comunicação:

É usada como ferramenta de comunicação e gestão empresarial, de modo que organizações e pessoas se mantenham operantes e competitivas nos mercados em que atuam. Em face disso, além de sua rápida evolução, é cada vez mais intensa a percepção de que a tecnologia de informação e comunicação não pode ser dissociada de qualquer atividade, como importante instrumento de apoio à incorporação do conhecimento como o principal agregador de valor aos produtos, processos e serviços entregues pelas organizações aos seus clientes.

De acordo com Rossetti e Morales (2007), a Figura 1 remete à visão piramidal da evolução dos sistemas de informação referida por Martin (1982), e fornece uma ideia de como a tecnologia da informação e comunicação evoluiu nas organizações ao longo do século XX.

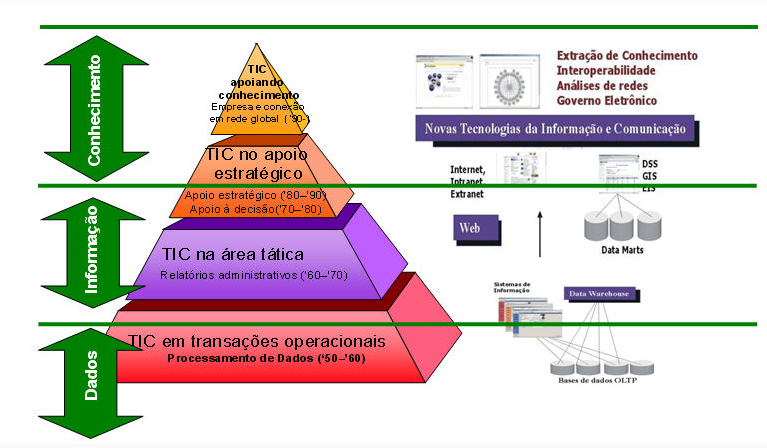


Figura 1 - Evolução da tecnologia da informação e comunicação nas organizações durante o séc. XX

Fonte: Rosseti e Morales (2007)

## 2.2 SETOR SUPERMERCADISTA

A estrutura do comércio brasileiro é formada basicamente por 3 segmentos de atividades: comércio por atacado, comércio varejista e comércio de veículos automotores, peças e motocicletas. No comércio varejista, uma mercadoria vendida é destinada ao consumidor final, para uso pessoal ou doméstico; e no comércio atacadista, uma mercadoria é vendida destinada ao consumidor intermediário, principalmente, para outros estabelecimentos (IBGE, 2016).

De acordo com a Pesquisa Anual de Comércio realizada pelo IBGE em 2016, a participação desses segmentos na receita operacional líquida do Comércio foi assim constituída: 45,3% do atacado, 45,1% do varejo e 9,6% do comércio de veículos. Dentro do segmento varejista, o setor de hiper e supermercados é o que teve a maior receita operacional líquida, com a maior média de pessoas ocupadas por empresa.

Segundo Ferreira, Venancio e Abrantes (2009, p. 1):

No início dos anos de 1980 esse segmento comercializava 75% dos produtos do setor varejista, passando para 82,6% no final daquela década. No final dos anos de 1990, sua participação ampliou-se para 86,1% do volume total de vendas de bens de consumo diário, revelando a importância desse segmento.

Supermercados são estabelecimentos comerciais classificados como de autosserviço, pois o consumidor final compra o produto sem necessariamente ter um funcionário da loja intermediando o processo.

O setor de supermercados se destaca pelo elevado índice de crescimento nos últimos anos. Em 2016, o faturamento nominal do setor foi de R$ 338,7 bilhões contra R$ 124,1 bilhões em 2006, representando um aumento de cerca de 63,35% em 10 anos (BRADESCO, 2017).

De acordo com a ABRAS (2017), o setor supermercadista brasileiro fechou o ano de 2016 com 89 mil lojas e 1,8 milhão de funcionários diretos, o que demonstra nas pesquisas o alto índice de empregabilidade, faturamento e expansão deste setor.

A Figura 2 apresenta as categorias mais vendidas nos supermercados e, que desta maneira, tornam os supermercados um potencial setor em expansão.

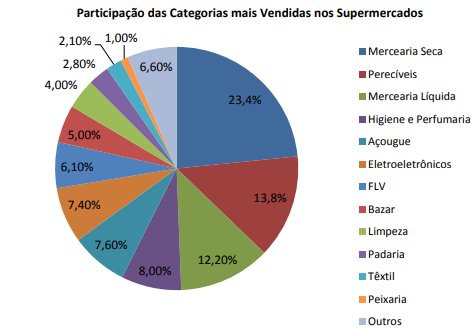


Figura 2 - Categorias mais vendidas nos supermercados

Fonte: ABRAS (2011)

Já o faturamento do setor está distribuído nas regiões brasileiras conforme a Figura 3.

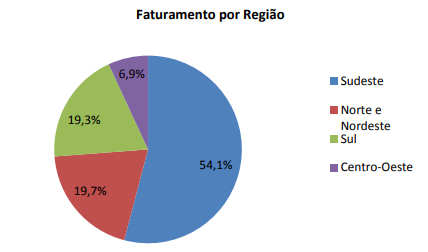


Figura 3 - Faturamento por região

Fonte: ABRAS (2011)

Quando se analisam as unidades federativas, pode-se observar a grande participação de São Paulo (36,2%), que concentra mais de um terço do faturamento do setor. Em seguida vêm Rio de Janeiro (9,8%), Rio Grande do Sul (8,5%), Minas Gerais (7,1%) e Paraná (6,6%), como mostra a Figura 4.

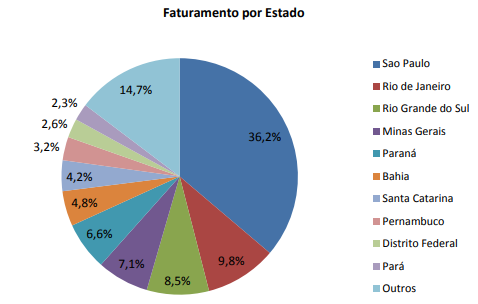


Figura 4 - Faturamento por Unidade da Federação

Fonte: ABRAS (2011)

Com relação ao número de lojas e à metragem, entre as 300 maiores do ranking da ABRAS, o cenário é apresentado na Figura 5.

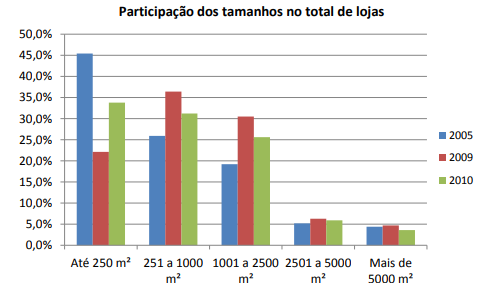


Figura 5 - Participação dos tamanhos no total de lojas

Fonte: ABRAS (2011)

O crescimento econômico brasileiro nos anos 2000, acompanhado de um cenário estável à época, com aumento da renda da população e a facilidade do acesso ao crédito contribuiu para que houvesse um crescimento significativo nessa década. Em 2010, houve investimentos de R$ 4 bilhões no setor, sendo a maior parte destinada à expansão pelo aumento de lojas, ampliação da área de vendas e aquisições de concorrentes. Isso é apresentado na Figura 6.

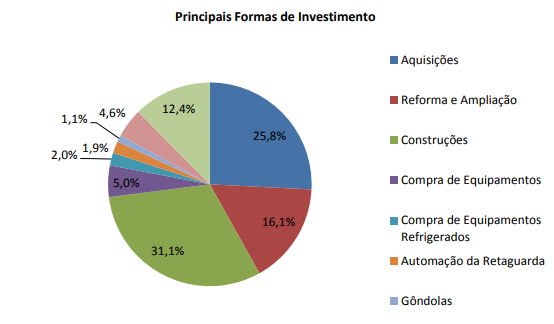


Figura 6 - Principais formas de investimento

Fonte: ABRAS (2011)

Segundo levantamento da Agência Nacional de Telecomunicações (Anatel), em outubro de 2010, existia cerca de 121 milhões de usuários de *smartphones* no Brasil. Além de representar uma forma de pagamento, essa tecnologia vem impactando de diversas formas o segmento. Primeiro, o consumidor pode cotar o preço de um concorrente estando dentro de um supermercado. Segundo, esses aparelhos podem ser uma importante fonte de informação sobre os produtos. A partir da leitura do código de barras, eles poderão oferecer diversos dados aos consumidores. E, ainda, qualquer consumidor pode atestar ou reclamar da qualidade de um estabelecimento enquanto está sendo atendido. E, se antes as notícias do tipo “boca-a-boca” demoravam a chegar até mais pessoas, sabe-se que com a iminência das redes sociais, elas são canais que propulsionam a circulação desse tipo de notícia.

Segundo a ABRAS (2011), o pagamento mais utilizado ainda é o dinheiro. Contudo, houve um crescimento elevado em outras formas de pagamento, como algumas citadas neste texto e mostradas na Figura 7.

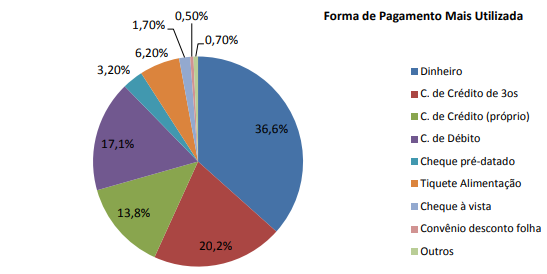


Figura 7 - Formas de pagamento mais utilizadas

Fonte: ABRAS (2011)

Ferreira, Venancio e Abrantes (2007) afirmam que o processo de ganhos de eficiência precisa ser mensurado e acompanhado, como forma de propiciar ações por parte dos tomadores de decisão, tanto público ou privado, para viabilizar o crescimento orientado desse setor. Nessa direção, sob a ótica do bem-estar econômico, salienta-se que a presença de pequenos supermercados também é positiva, em razão da equalização de preços, via concorrência, fato que reforça a necessidade de acompanhamento contínuo da atividade.

### 2.2.1 Comportamento do Consumidor no Setor Supermercadista

Diante do atual cenário mundial, onde as empresas ou estabelecimentos se preocupam em manterem-se competitivas, os administradores desses negócios têm reconhecido amplamente a importância de orientar esforços no sentido de conhecer e atender melhor os seus consumidores.

Reforçando a percepção no que diz respeito à orientação para a qualidade do atendimento prestado especialmente aos consumidores, Whiteley (1992 apud ROJO, 1998, p. 17) mostra que uma empresa efetivamente voltada para o cliente tem ingredientes que ele denomina os "sete imperativos fundamentais". São eles:

1. crie uma visão que preserve o cliente: todos os funcionários da empresa, desde o presidente, devem estar empenhados na missão de satisfazer os clientes;
2. sature sua empresa com a voz do cliente: deve ser criada uma real intimidade entre a empresa e o cliente, revolucionando sua conduta e mudando a posição competitiva;
3. aprenda com os vitoriosos: as grandes empresas não conseguem e nem tentam esconder seu estilo de fazer negócios. Procure estudar seus métodos e filosofia;
4. aqueles que estudam os vitoriosos fortalecem seu compromisso em servir os próprios clientes;
5. libere os defensores de seus clientes: a maioria dos empregados quer servir bem os clientes e um dos fatores mais fortemente relacionados à permanência dos funcionários na organização é a percepção de que a empresa está cumprindo essa função. Os dirigentes devem, portanto, mostrar aos funcionários que a tarefa primordial na empresa é servir aos clientes;
6. derrube as barreiras ao desempenho conquistador de clientes: quanto mais se aprende sobre qualidade, mais se percebe que devem ser eliminadas quaisquer barreiras que prejudiquem o atendimento aos clientes;
7. esteja sempre medindo: nas organizações em processo mais acelerado de melhoria, mede-se tudo o que possa informar sobre a satisfação do cliente. Devem ser analisados não apenas seu próprio passado e os desejos dos clientes, mas também o desempenho de quem, em qualquer parte do mundo, esteja realizando melhor do que você uma tarefa semelhante.
8. faça o que você diz: dirigentes bem-sucedidos que cumprem o que prometem, com enfoque no cliente, estão criando uma nova visão da liderança em qualidade. Líderes de corporações que efetivamente buscam a melhoria constante de qualidade estão sempre aprendendo, acreditando e investindo nas pessoas.

A Figura 8 mostra os atributos mais citados pelos consumidores para a escolha do local das compras. Rojo (1998) afirma que para o levantamento dessas informações, foi solicitado a cada entrevistado que citasse os três atributos mais importantes para escolha do supermercado em que faz suas compras. Uma vez identificados os atributos mais importantes, foi solicitado aos entrevistados que fizessem uma avaliação desses atributos para cada supermercado que costuma frequentar.

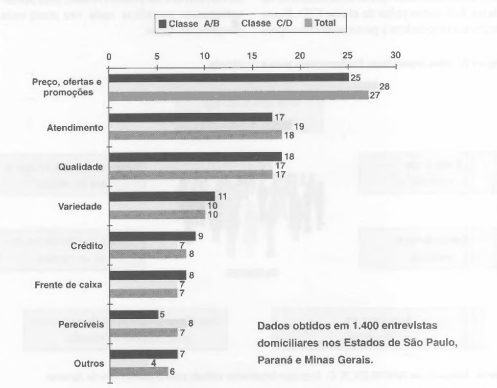


Figura 8 - Atributos para escolha do local das compras

Fonte: Rojo (1998)

Segundo Rojo (1998, p. 20):

O atendimento aparece como o segundo atributo mais importante, praticamente com a mesma frequência que a qualidade. Essa percepção também corresponde ao manifestado pelos supermercadistas, que consideram o atendimento a variável mais importante, após o preço. O atendimento é percebido pelos próprios supermercadistas como fator de atração e principalmente de fidelização do consumidor. Para os supermercadistas é fundamental que o cliente sinta que foi atendido com cordialidade e eficiência durante suas compras na loja.

Rojo (1998) afirma que a qualidade é uma variável importante, assim como o atendimento. A imagem de qualidade do estabelecimento apoia-se em dois fatores: padrão arquitetônico e tecnológico do supermercado e produtos perecíveis. Rojo (1998, p. 21) ainda afirma que:

A loja moderna reforça a percepção que o consumidor tem em relação à qualidade do supermercado. No entanto, é pelos produtos perecíveis que os consumidores baseiam grande parte de sua percepção em relação à qualidade da loja. Os produtos industrializados, como os enlatados, são iguais em todos os supermercados, mas, quando o cliente compara a qualidade de uma fruta ou verdura, identifica mais facilmente as diferenças entre lojas concorrentes. Dessa forma, os consumidores ressaltam que o supermercado precisa cuidar muito bem da exposição e da qualidade dos produtos perecíveis.

O outro fator importante para o consumidor é a variedade, conforme Rojo (1998, p.21):

[...] também um fator importante de diferenciação para a escolha do supermercado. Os supermercadistas reconhecem a importância dessa variável e demonstram isso na medida em que estão procurando aplicar cada vez mais o conceito de gerenciamento de categorias em suas lojas. Para o varejista, a gestão da variedade pode ter forte impacto em seu investimento e giro de estoques, afetando diretamente a rentabilidade da loja.

O comportamento do consumidor baseia-se também na oferta de crédito. Rojo (1998, p. 22) afirma que o crédito por meio de cheques pré-datados e cartões atrai o consumidor:

Entre os consumidores, 52% consideram que as alternativas de crédito oferecidas pelos supermercados atendem plenamente a suas expectativas, mas mesmo os mais satisfeitos demonstram que "quanto mais prazo melhor". Para o supermercadista o crédito é considerado um mal necessário: "se a concorrência oferece prazo, não podemos ficar muito atrás, mas o risco aumenta devido à inadimplência".

Também associada à imagem de atendimento da loja, a rapidez para pagar as compras parece ser um ponto de estrangulamento nos supermercados. Apesar da informatização, para muitos consumidores esse ainda é o pior momento na loja. Rojo (1998) afirma que apenas 27% dos consumidores consideram excelente a frente de caixa nas lojas que frequentam. Não menos importante do que os outros fatores, os produtos perecíveis muito contribuem para formar a imagem de qualidade do supermercado. Rojo (1998, p. 22) explica que as avaliações feitas pelos consumidores demonstram que ainda há muito a percorrer, pois apenas 35% dos entrevistados consideram excelentes as seções de perecíveis das lojas que frequentam. As causas da insatisfação são:

1. açougue: - filas (quando a loja vende com serviço de atendimento); - embalagens com sangue (autoserviço); - cortes inadequados; - odor no açougue; - falta de higiene; - variedade inadequada das embalagens de autoserviço;
2. frutas e verduras: - verduras e frutas com má aparência; - poucas balanças para pesagem; - falta de cortesia no atendimento (pesagem) - falta de padrão nos produtos (variação de qualidade na mesma banca);
3. frios e laticínios: - produtos com pouco tempo para vencer a validade; - má conservação de produtos refrigerados;
4. padaria: - pouca variedade, principalmente na confeitaria; - falta de qualidade na apresentação dos produtos.

Entretanto, caso seja analisado o setor supermercadista de forma geral, verifica-se que grande parte das compras de perecíveis é realizada em açougues, feira ou “sacolões”, conforme mostra a Figura 9.

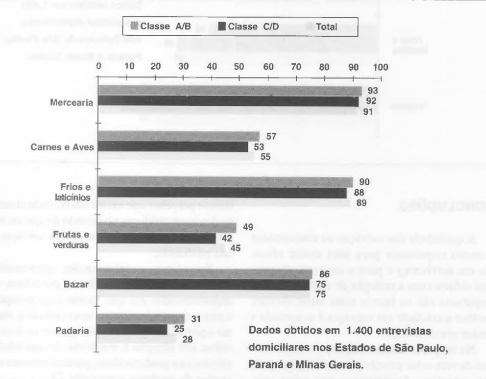


Figura 9 - Participação dos supermercados, por seção, nas compras dos consumidores

Fonte: Rojo (1998)

Rojo (1998) em conclusão, afirma que as informações apresentadas sobre como os consumidores percebem os supermercados em que fazem suas compras tornam claro que existe a necessidade e muitas oportunidades para que o setor se desenvolva em direção à melhoria de qualidade, eficiência e produtividade, particularmente em seções de produtos perecíveis.

### 2.2.2 Área de Influência do Estabelecimento

No setor varejista, o desempenho de supermercado depende, grandemente, de sua localização, pois a maior parte das vendas de uma loja vem de consumidores e clientes que moram dentro de uma área geográfica relativamente pequena em torno da loja.

Parente e Kato (2001) afirmam que ao investigar onde os clientes estão localizados é possível identificar a dimensão geográfica da demanda de mercado disponível para certa loja. No varejo, em geral, essa dimensão geográfica é uma variável pouco controlável, pois as lojas podem não conseguir determinar os limites geográficos de onde se originam seus consumidores e clientes.

Segundo a *American Marketing Association* (AMA) (apud PARENTE; KATO, 2001, p. 47), a área de influência é “uma área geográfica contendo os consumidores de uma empresa particular ou grupo de empresas para bens ou serviços específicos”. Segundo os autores, essa área de influência tem sido estudada há muito tempo por causa de sua importância no desempenho de uma loja, sendo que foi Applebaum, em 1966, que identificou a área de influência de supermercados por meio da técnica de *customer spotting* ou mapeamento de clientes. Essa técnica consiste em identificar, em um mapa, a localização da procedência (em geral, o local da residência) de uma amostra representativa de clientes de uma loja.

Mediante a análise da dispersão geográfica dos clientes em torno de uma loja, três segmentos foram identificados de uma área de influência nos estudos de Applebaum e abordados por Parente e Kato (2001, p. 47) conforme seguem:

1. área de influência primária – região mais próxima da loja, apresentando maior densidade de clientes, onde estão concentrados cerca de 60% a 75% dos clientes;
2. área de influência secundária – região em torno da área de influência primária, onde estão concentrados cerca de 15% a 25% dos clientes;
3. área de influência terciária – região que contem a parcela restante dos clientes que moram mais afastados da loja (cerca de 10%).

Parente e Kato (2001, p. 47) afirmam que os estudos sobre áreas de influência são aprimorados pelo uso de *Geographic Information System* (GIS), que permite “maior capacidade de fazer ‘ligações’, isto é, relacionar uma informação com outras, a fim de dar sentido ao que seria um conjunto desconexo de informações”.

No estudo de Parente e Kato (2001), alguns conceitos referentes aos padrões de concentração e dispersão de clientes em torno dos supermercados foram confirmados por meio de um processo de investigação baseado na lógica indutiva, e uma delas foi realizada por mapas geocodificados.

A Figura 10 mostra o mapeamento dos consumidores de acordo com a loja feito por Parente e Kato (2001).

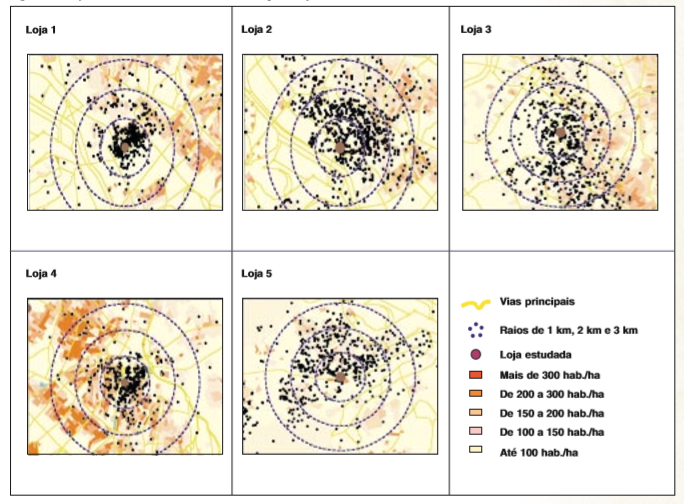


Figura 10 - Mapeamento dos consumidores em relação a loja

FONTE: Parente e Kato (2001)

Os mapas da Figura 10 indicam a localização dos clientes (cada ponto representa um cliente pesquisado) em torno das lojas pesquisadas. Uma inspeção visual indica grandes variações na habilidade das unidades de atrair clientes. As lojas 1 e 4 mostram uma pequena área de influência, com clientes densamente concentrados em áreas próximas da loja. Por outro lado, as lojas 2, 3 e 5 apresentam um padrão menos concentrado e, consequentemente, uma área de influência maior. Parente e Kato (2001) afirmam que as áreas mais densas de clientes se encontram em regiões mais próximas da loja, em regiões mais densamente povoadas e ao longo das principais vias de acesso.

A Figura 11 representa o fenômeno da dispersão dos clientes, indicando a percentagem acumulada de clientes dentro de diferentes raios traçados em torno da loja.

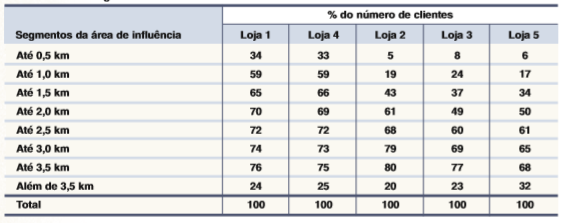


Figura 11 - Porcentagem acumuladas de clientes em raio concêntricos

Fonte: Parente e Kato (2001)

Outra técnica para viabilizar e justificar as áreas de influência de um supermercado para seus clientes é a de curvas relacionadoras de distância e porcentagens acumuladas de clientes. Parente e Kato (2001) afirmam que para proporcionar uma representação mais sumarizada dos fenômenos da distribuição dos clientes em torno de lojas, foram desenvolvidas curvas da área de influência com base nos dados da Figura 11, relacionando a distância e o percentual acumulado de clientes. Ao capturar a maior parte das informações contidas nos mapas de *customer spotting*, as curvas oferecem uma forma útil de retratar o padrão de concentração e dispersão dos clientes em torno de um supermercado.

Segundo Parente e Kato (2001), “ao capturar a maior parte das informações contidas nos mapas de *customer spotting*, as curvas oferecem uma forma útil de retratar o padrão de concentração e dispersão dos clientes em torno de uma loja”. Essa informação pode ser analisada na Figura 12.

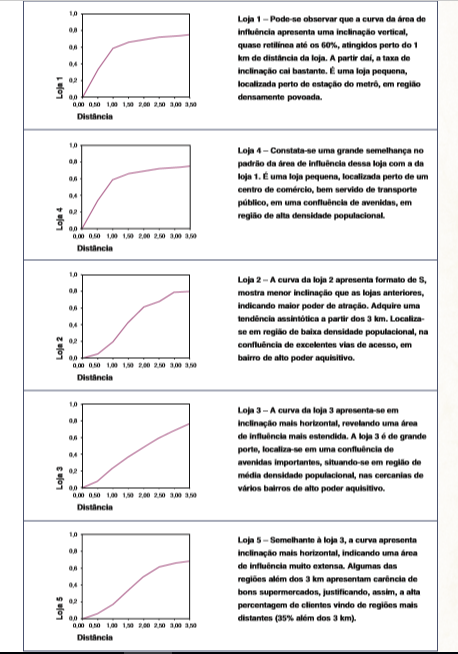


Figura 12 - Curva de área de influência

Fonte: Parente e Kato (2001)

No Brasil percebe-se que consumidores típicos não limitam suas compras de alimentos a uma única loja, ao contrário, têm um repertório de lojas, escolhidas para diferentes tipos de compras, afirmam Sheth, Mittal e Newman (1999).

### 2.2.3 Estratégias de Varejo *Omnichannel*

O conceito *Omnichannel* já é uma realidade muito conhecida nos EUA e Europa, começou recentemente a ganhar força e ser reconhecido também no Brasil. De uma maneira simples, o termo pode ser compreendido como a utilização de vários canais para atingir e estabelecer um relacionamento com o consumidor final, ou melhor, define um conjunto de mercadorias ou serviços por meio de muitos canais difundidos.

Num contexto mais amplo, o termo surgiu como tendência e referência mundial para mudar completamente a forma de atuação das empresas no segmento de varejo que desejam acompanhar tal mudança que acontece principalmente devido ao comportamento e ao novo perfil do consumidor.

Desta maneira, sabe-se que o comportamento do consumidor é reflexo de uma mudança inédita, acompanhada principalmente pela evolução tecnológica e digital, o fato de tornar-se *Omnichannel.* No entanto, não impõe limites ou definições para o que realmente pode ser feito num determinado negócio, utilizando seus conceitos e metodologias. Para Belmiro e Pastore (2018), o tema *Omnichannel* é emergente e demonstra relevância, ineditismo, mas ainda exige robustez teórica, pois se trata de um tema de natureza empírica em busca de novas teorias para dar-lhe mais suporte conceitual.

De acordo com Belmiro e Pastore (2018, p. 1 apud Cummins; Peltier & Dixon, 2014), “na tentativa de definir um modelo operacional, o varejo *Omnichannel* tem as seguintes características”:

1. Jornadas de compras selecionadas e percorridas pelo próprio consumidor;
2. Desenvolvimento de canais para atuar em sinergia, ao contrário de canais sendo desenvolvidos isoladamente, do começo ao fim do processo;
3. Inclusão de canais digitais na oferta da loja como uma extensão física do serviço on-line;
4. Segmentação complexa de consumidores, não segmentando mais em grupos ou clusters, partindo do pressuposto de que cada indivíduo se envolve com um único canal preferido;
5. Desenvolvimento de uma imagem única e universal e de um portfólio de produtos e serviços que abranja todos os canais, em oposição a canais de marcas diferentes com ofertas distintas;
6. Aumento da modularização dos serviços para permitir transições de canal em oposição a obrigação a lidar apenas com uma empresa específica (*lock-in*) para canais individuais;
7. Desenvolvimento de complementos on-line para serviços baseados no fornecimento de bens físicos.

Segundo Belmiro e Pastore (2018), a inovação no varejo que passa pela estratégia *Omnichannel* tem em seu foco central a proposição de novas e perfeitas experiências aos clientes dos varejistas. Desta maneira, toda evolução presente na interatividade humana procura encurtar o caminho para se atingir a satisfação do consumidor com o consumo e a partir de então, garantir a fidelidade deste em relação às marcas com as quais se identifica, sem que outros fatores possam influenciar negativamente nessa relação. Futuros estudos devem avançar na questão e proporcionar novos achados até a consolidação de uma teoria capaz de delinear a completa estratégia de marketing *Omnichannel*, para um consumidor *Omnichannel*.

## 2.3 TRABALHOS RELACIONADOS

### 2.3.1 Sistemas de Apoio à Decisão no Varejo

Os Sistemas de Apoio à Decisão (SAD) são implementados em diversas organizações com os mais variados segmentos de atuação. Rodrigues (1998) afirma que no setor de comércio varejista o exemplo mais citado é o da rede *Walmart*, que na busca por melhores contatos com os consumidores e parceiros importantes para seus negócios, utiliza *Data Mining* aliada ao *Data Warehouse.*

Rodrigues (1998) afirma ainda que, inicialmente, o *Walmart* buscava relações entre consumo de alguns produtos e os dias da semana e o consumo de produtos por vendas casadas. A relação mais curiosa encontrada foi que o de consumo de fraldas descartáveis sempre aparecia aliado ao consumo de cerveja durante às sextas-feiras. Essa relação foi confirmada com a aproximação destes 2 produtos nas prateleiras da rede e um sucessivo aumento nas vendas aconteceu, isto porque sempre que o consumidor ia ao supermercado comprar fraldas para o fim de semana aproveitava para também abastecer a geladeira com cerveja.

Rodrigues (1998) cita outro exemplo de SADs, especialmente, no ramo dos supermercados. Esse exemplo é o da rede varejista *ShopKo*, que para aumentar suas vendas e sobreviver à concorrência passou a adotar ferramentas de *Data Mining*, conseguindo aumentar suas vendas em 90%, principalmente, por meio de vendas indiretas de outros produtos, com isso conseguiu manter-se no mercado resistindo à concorrência da *Walmart* (RODRIGUES, 1998).

Vale a pena ressaltar que não só os supermercados se destacam em sucesso na utilização do Sistema de Apoio à Decisão, mas também outros estabelecimentos do mercado varejista. Bispo (1998) afirma que as lojas brasileiras que utilizaram ferramentas de *Data Mining* conseguiram reduzir seus produtos expostos de 51000 para 14000, estabelecendo uma melhor regionalização de produtos e, com isso, foi possível obter economia no posicionamento de produtos, estoques, pessoal e em outros custos operacionais.

### 2.3.2 Negócios baseados em Tecnologias *Web*

Por permitir o acesso à informação pela rede Internet, as tecnologias destinadas à *World Wide Web* contribuem para o surgimento de novas oportunidades de negócios. De acordo com Zaneti Junior e Vidal (2006), as atividades das organizações que envolvem interação com os clientes e fornecedores (por exemplo, compra, venda, atendimento pós-venda, suporte, recrutamento e divulgação de produtos, serviços e pedidos) podem ser transformadas para aproveitar os benefícios das tecnologias *Web*.

É possível utilizar os espaços virtuais da *Web* de forma similar ou melhor ao que é feito por outras mídias, como jornais e revistas. Em muitos casos, a localização física da empresa torna-se menos importante, pois, de certa forma, na *Web* todas elas estão à mesma distância virtual dos parceiros de negócios.

O espaço virtual e a forma como a interação com os parceiros é projetada têm impactos significativos para um negócio apoiado por tecnologias *Web*. Algumas atividades podem ser feitas sem intermediação, como é o caso de alguns serviços oferecidos por órgãos governamentais e, em outros, novos intermediários podem ser criados como serviços de busca na rede ou de agregação de informações (YANES, 1998, p.42).

Ao longo dos últimos anos, tem havido grande impulso nos negócios realizados via plataformas virtuais, principalmente, no comércio tipo empresa para consumidor (ou *Business to Consumer -* B2C), empresa para empresa (*Business to Business* - B2B) e Consumidor para Consumidor (*Consumer to Consumer* – C2C) e, isso tem acontecido, devido às oportunidades geradas pela evolução das tecnologias *Web*.

Segundo Zaneti Junior e Vidal (2006, p. 233) previram isso:

A Tecnologia Web deve causar grande impacto nos sistemas de informação das empresas, tanto na forma como estão sendo ou serão construídos quanto na maneira como estão sendo ou serão utilizados.

Neste sentido, nas próximas seções são apresentados 2 trabalhos que utilizam tecnologias *Web* como oportunidades de novos negócios.

### 2.3.2.1 Trabalho: Desenvolvimento de um sistema de colaboração em massa para reduzir gastos dos consumidores em compras de supermercado

Cunico (2012) elaborou uma aplicação *web* visando a colaboração dos consumidores para reduzirem gastos com compras em supermercados. Pela aplicação, o usuário efetua um cadastro em um *website* para obter a autenticação. Depois de autenticado, o usuário pode incluir sua lista de compras que é analisada pelo sistema, retornando a indicação do mercado que oferece o menor preço para cada um dos itens da lista. O sistema também informa ao usuário o mercado onde a compra inteira totalizaria o menor custo. Com a colaboração dos usuários, ainda é necessário que o sistema seja alimentado por dados relacionados aos preços dos itens. Para isso, um usuário pode incluir os produtos e preços dos mercados onde ele já fez compras.

Panchal e Fathianathan (2008) afirmam que o sucesso de trabalhos como este, depende da quantidade e diversidade dos indivíduos envolvidos, cada um com seus respectivos conhecimentos e habilidades.

### 2.3.2.2 Trabalho: Gastometro

Uma pesquisa do Serviço de Proteção ao Crédito (SPC) revelou que 34% dos brasileiros gastam mais do que planejam em compras de supermercados todos os meses (FLIPIT, 2014 apud ALBRECHT, 2014, p. 11) e sugere que elaboração de listas de compras:

Estas, não só evitam que você esqueça os itens de maior necessidade, como também ajudam a manter a disciplina. Fazer compras para o mês todo pode induzi-lo a exagerar, já que você tende a pensar que, como está comprando para um período prolongado, tem de levar mais do que realmente precisa. Outra dica importante é comprar toda semana, desta maneira você fica por dentro das promoções, e para aproveitar ao máximo estas, consulte os preços em mais de um supermercado.

Albrecht (2014) ainda explicita que cerca de 82% dos consumidores já pesquisaram um produto ou serviço utilizando *smartphones* antes da compra e que essas pesquisas influenciaram suas decisões de compra. Com isso Albrecht (2014) afirma que ter um *website* preparado para acesso via dispositivo móvel é essencial. Além disso, é necessário ter uma estratégia para vários canais (*mobile* e *web*) a fim de envolver os consumidores nos diversos caminhos até a compra.

Albrecht (2014) desenvolveu o Gastometro, que é um aplicativo para dispositivo móvel com sistema operacional *Android.* Esse aplicativo possibilita o lançamento de itens e preços para a loja onde se realiza a compra, dessa forma, o usuário pode ter controle sobre o que está comprando e quanto irá pagar. Outras funcionalidades presentes no aplicativo são as listas de compra, lista para pesquisa de preços, além de conversão de moedas, para as compras internacionais.

# 3 O OBJETIVO DO PROJETO

Este Capítulo aborda o tema principal do projeto e, a partir disto, manifesta os problemas emergentes que demandam uma proposta de solução. Também apresenta os objetivos, a justificativa, o público a quem se destina o sistema de *software* e os grupos funcionais com os respectivos níveis de decisão afetados.

Os objetivos delineiam diferentes pontos de vista para avaliação do resultado e também norteiam diretrizes gerais para realização de subprojetos (outros produtos, processos ou funcionalidades) que podem derivar do projeto principal.

## 3.1 FORMULAÇÃO DO PROBLEMA

Em um supermercado é fundamental a qualidade dos produtos comercializados pela loja, assim como é importante prestar o serviço adequadamente para conseguir a satisfação dos consumidores e conquistar clientes.

Berry e Parasuraman (1992) destacam que o desempenho confiável e preciso do serviço é o coração da excelência do *marketing*. Existem circunstâncias em que a empresa oferece um serviço de forma descuidada e comete erros que poderiam ser evitados ou faz promessas sedutoras que não são cumpridas, o que estremece a confiança do consumidor e abala sua imagem de excelência em serviço.

Conforme a pesquisa de Rojo (1998), o atendimento é segundo atributo mais valorizado pelos consumidores que frequentam supermercados, item que está atrás apenas da qualidade dos produtos vendidos, principalmente, os perecíveis. A pesquisa evidenciou também a necessidade de melhoria nos processos e no atendimento pessoal, especialmente, na frente de caixa. Queixas como falta de troco, demora em passar o produto, falta de cuidado no manuseio das compras e mau humor dos operadores de caixa são comuns entre os consumidores. Isso afeta diretamente a satisfação do consumidor em relação à loja que frequenta.

Observa-se, assim, a necessidade de os supermercados implementarem meios para avaliarem continuamente como está a qualidade de seus produtos e também a prestação de seus serviços, pois, conforme Rojo (1998, p.20):

O atendimento é percebido pelos próprios supermercadistas como fator de atração e principalmente de fidelização do consumidor. Para os supermercadistas é fundamental que o cliente sinta que foi atendido com cordialidade e eficiência durante suas compras na loja.

Ainda na pesquisa de Rojo (1998) pôde-se observar a importância de as lojas manterem preços competitivos e realizarem ofertas e promoções, pois são ações que atraem consumidores de todas as classes econômicas. Rojo (1998, p. 19) afirma que: “com o mercado competitivo, além de preços baixos, os supermercados estão oferecendo mais qualidade e serviços; dessa forma, o cliente irá escolher aquele que lhe proporcionar maior valor”.

É evidente a necessidade de que um supermercado procure ser competitivo em preços, para que ele se sobressaia em relação aos seus concorrentes. Entretanto, Rojo (1998) afirma que essa competitividade deve ser consequência da eficiência e da produtividade, sem necessariamente sacrificar suas margens de lucro.

Para muitos consumidores, a variedade é também um fator importante de diferenciação para a escolha do supermercado. Rojo (1998) afirma que os supermercadistas reconhecem a importância desse atributo. Contudo, para um varejista, é um desafio a gestão da variedade, pois ela requer um balanceamento entre investimento e giro de estoques, influenciando, diretamente, na rentabilidade do supermercado.

Assim, pode-se perceber que há muitos desafios a serem superados no segmento supermercadista, a fim de que se consiga o aprimoramento desse negócio para conseguir maior satisfação dos consumidores e fidelização de clientes. Neste cenário, vislumbram-se que as tecnologias de informação e comunicação não podem ser dissociadas da capacidade de ajudarem na superação desses desafios e que os sistemas de *software* podem oferecer meios eficazes para a obtenção de informações para tomada de decisão, a fim de que essas organizações se mantenham operantes e competitivas nos segmentos em que atuam.

## 3.2 OBJETIVOS

O objetivo do projeto EzMart é desenvolver um sistema de *software,* baseado em tecnologias *web*, a ser utilizado como um canal de comunicação digital entre supermercados e seus consumidores.

Para este sistema de *software*, os seus objetivos são assim elencados:

1. coletar *feedback* dos consumidores em relação ao supermercado e assim proporcionar informações mais assertivas para cada um deles;
2. divulgar aos consumidores preços, ofertas e promoções dos supermercados, por meio de propagandas a serem vinculadas nas páginas do *website* do sistema;
3. permitir que o consumidor compare preços de sua lista de produtos em todos os supermercados desejados ou mais próximos;
4. permitir que o consumidor localize os supermercados de acordo com um raio de distância selecionado;
5. permitir que o consumidor crie e gerencie suas listas de produtos.

## 3.3 JUSTIFICATIVA

Conforme Chopra e Meindln (2001, p.354):

Informação é essencial para tomar boas decisões de gerenciamento de cadeia de suprimentos porque ela proporciona o conhecimento do escopo global necessário para tomar boas decisões. A tecnologia da informação proporciona as ferramentas para reunir estas informações e analisá-las objetivando tomar as melhores decisões sobre a cadeia de suprimentos.

Projetos que visam desenvolver soluções para o Comércio são relevantes para o país, uma vez que as atividades comerciais empregam significativa parcela da população e contribuem, em grande medida, para a composição do Produto Interno Bruto (PIB). Segundo o IBGE (2016), o setor de hiper e supermercados:

1. é o maior do comércio brasileiro em receita líquida, correspondendo à 12,6% de participação no setor de comércio;
2. tem a maior média de empregados por empresa;
3. reúne mais de 89 mil lojas estabelecidas no Brasil.

O crescimento do mercado supermercadista é real e suas demandas devem ser atendidas.

Entretanto, o desempenho de supermercados depende, grandemente, de sua localização, pois a maior parte das vendas de uma loja vem de clientes que moram dentro de uma área geográfica relativamente pequena em torno da loja. Com isso, Parente e Kato (2001) afirmam que os estudos varejistas dão destaque ao conceito de área de influência que é a dimensão geográfica da demanda de mercado disponível para certa loja.

Portanto, o desenvolvimento do EzMart apoia-se nos diferentes estudos de diversos autores aqui tratados, para que sejam evidenciados os pontos positivos de inserção desta tecnologia.

## 3.4 PÚBLICO ALVO

Segundo Zenone (2007, p.54), com a definição de um público alvo é possível identificar um segmento particular ou segmentos da população que se desejam servir. O mercado consiste em muitos tipos de clientes, produtos e necessidades. É preciso determinar que segmentos oferecem as melhores oportunidades para um negócio.

Desta maneira, o sistema de *software* é destinado ao segmento varejista, no ramo de supermercados e aos consumidores deste mercado.

## 3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS

O sistema de *software* possui diferentes perfis de usuário. O primeiro perfil consiste no grupo de consumidores, que por sua vez, estão inseridos no nível operacional. O segundo grupo consiste no grupo de supermercados que são gerenciados pelos administradores de cada local e que estão inseridos no nível tático.

# 4 GERÊNCIA DE PROJETO

Segundo Pressman (2016, p. 685), um gerenciamento eficiente do desenvolvimento de *software* se concentra nos quatro “ps”: pessoas, produto, processo e projeto. Portanto, essa ordem não é arbitrária. Afirma ainda:

O gerente que se esquecer de que o trabalho do engenheiro de *software* consiste em esforço humano nunca terá sucesso no gerenciamento de projeto. Da mesma forma, o que não estimula a ampla comunicação entre os envolvidos no início da evolução de um produto corre o risco de desenvolver uma solução elegante para o problema errado. [...] Aquele que embarca sem um plano de projeto sólido compromete o sucesso de um projeto.

Portanto, o gerenciamento de projetos é a aplicação de conhecimentos, habilidades e técnicas para a execução de projetos de forma efetiva e eficaz. Trata-se de uma competência estratégica para organizações, permitindo com que elas unam os resultados dos projetos com os objetivos do negócio – e, assim, melhor competir em seus mercados. Neste Capítulo há uma abordagem sistemática para o gerenciamento do projeto produzido, contendo as partes interessadas, o modelo de ciclo de vida adotado para sobrevivência do projeto, e os recursos e esforços necessários para o desenvolvimento deste artefato. As áreas de conhecimento priorizadas também são tratadas neste Capítulo, como escopo, tempo, integração, qualidade e riscos.

Vale ressaltar que todo o conteúdo abordado neste Capítulo apresenta uma sólida composição teórica embasada em importantes fatores do gerenciamento de projetos, como em destaque, o PMBoK publicado pelo PMI (2013), referência nos padrões de gerenciamento de projetos.

## 4.1 PLANO DE PROJETO

Esta seção é responsável pela elaboração do plano de projeto. Dentre as vertentes desta seção, o primeiro subitem é responsável por evidenciar as partes interessadas no projeto EzMart, utilizando para isto uma matriz de Poder x Interesse.

O segundo subitem trata de forma clara e objetiva o modelo de ciclo de vida adotado para sobrevivência do projeto, ideal para fornecer um modelo ou uma representação que conduza todo o desenvolvimento.

Portanto, é necessário que haja um devido estudo a respeito de todos os esforços necessários para o cumprimento do plano de projeto, assim como todos os recursos necessários, sejam eles de *hardware* ou de *software*.

### 4.1.1 Partes Interessadas

Para Freeman (1984, p.25), *stakeholder* é qualquer grupo ou indivíduo que pode afetar ou é afetado pela realização do objetivo almejado. Assim, cada interessado tem uma visão diferente do projeto e obtém diferentes benefícios quando o projeto é concluído com êxito e está sujeito a diferentes riscos negativos, caso o trabalho venha a fracassar.

Outro embasamento é referido pelo PMI (2013), que define que um *stakeholder* também pode ser chamado de parte interessada e que por sua vez, inclui todos os membros da equipe do projeto, assim como todas as entidades interessadas dentro ou fora da organização. Da mesma forma citada por Freeman, o PMI (2013, p. 33), “A equipe do projeto identifica as partes interessadas internas e externas, positivas e negativas, e as partes executoras e orientadoras a fim de determinar os requisitos do projeto e as expectativas de todas as partes envolvidas”. Desta maneira, o gerente do projeto, que é uma parte interessada precisa gerenciar a influência destas partes em relação a todos os requisitos do projeto a fim de garantir um resultado bem-sucedido. A Figura 13 ilustra a relação entre o projeto, a equipe de projeto e as diversas partes interessadas.

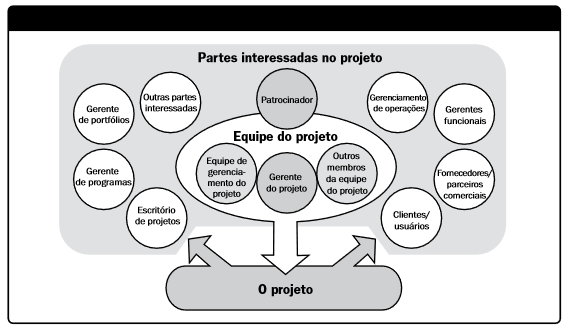


Figura 13 - Relação entre as partes interessadas e o projeto

Fonte: PMI (2013, p.33)

Com isso, as partes interessadas neste projeto são podem ser compreendidas a seguir.

1. instituição de Ensino: FAI – Centro de Ensino Superior em Gestão, Tecnologia e Educação. Essa parte representa a o trilho inicial para concepção do projeto. Fornece de forma específica a motivação necessária, os conhecimentos necessários, o ambiente tecnológico e o amparo pessoal que norteia o desenvolvimento do projeto;
2. gerente do projeto: Jéssica Souza Pivoto é a integrante responsável por liderar a equipe e conduzir a gerência do projeto. Possui papel fundamental para coordenar a equipe a desenvolver artefatos com qualidade e dentro dos prazos previstos;
3. equipe do projeto: formada por Camila Maria da Costa, Diego Cesar do Amaral, Jéssica Souza Pivoto, Marcos Henrique Azevedo e Marcos Paulo Moreno Pereira. Os integrantes auto-organizáveis deste projeto possuem papel fundamental para elaboração e artefatos como codificação, testes, documentação e execução do planejamento realizado pelo gerente do projeto;
4. organização de supermercados: Pessoas Jurídicas (PJ) que necessitam de uma fonte inédita de marketing para divulgar seus produtos e/ou serviços, e de um meio ágil de comunicação com o consumidor, de modo a melhorar diversos pontos registrados por eles;
5. usuários consumidores: Pessoas Físicas (PF) que necessitam de uma ferramenta baseada em tecnologia que os auxilie na busca por melhores estabelecimentos de supermercado e em principal, que ofereçam produtos mais baratos;
6. orientadores: são responsáveis por avaliar os artefatos produzidos em todas as entregas, em questão de qualidade, prazo e conteúdo. Possuem papel fundamental na motivação pessoal de cada integrante, e conduz de maneira inteligente e específica todas as qualidades encontradas no grupo, para que o trabalho contribua significativamente com a sociedade em que é inserida. Os coordenadores são: Eunice Gomes Siqueira e Afonso Celso Soares.

A Figura 14 demonstra as partes interessadas na forma de uma matriz que relaciona o grau de interesse e poder sobre o desenvolvimento deste projeto.

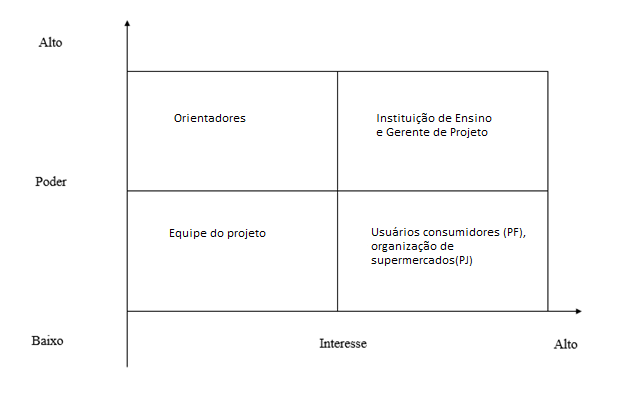


Figura 14 - Poder X Interesse do projeto EzMart

Fonte: Equipe (2018)

### 4.1.2 Modelo de Ciclo de Vida

Para manter a qualidade do *software* e permitir sua conclusão de forma satisfatória, são utilizados os modelos de processo em todo ciclo de vida do projeto, que dispõem de diversas atividades, tarefas e artefatos. Para o presente projeto, há um item inovador: a adaptação do *Scrum* ao modelo incremental.

O PMI (2013, p.37) descreve que:

Ciclo de vida do projeto é a série de fases pelas quais um projeto passa, do início ao término. As fases são geralmente sequenciais e os seus nomes e números são determinados pelas necessidades de gerenciamento e controle da(s) organização(ões) envolvida(s) no projeto, a natureza do projeto em si e sua área de aplicação. As fases podem ser desmembradas por objetivos funcionais ou parciais, resultados ou entregas intermediárias, marcos específicos no escopo geral do trabalho, ou disponibilidade financeira. As fases são geralmente limitadas pelo tempo, com um início e término ou ponto de controle. Um ciclo de vida pode ser documentado em uma metodologia. O ciclo de vida do projeto pode ser definido ou moldado de acordo com aspectos exclusivos da organização, setor ou tecnologia empregada.

Com isso, o modelo de ciclo de vida é a primeira escolha a ser feita no processo de *software*. A partir desta escolha definir-se-á desde a maneira mais adequada de obter as necessidades do cliente, até quando e como o cliente deve receber uma versão do sistema.

Processo de *software* é o conjunto de atividades que constituem o desenvolvimento de um sistema computacional. Estas atividades são agrupadas em fases, como: definição de requisitos, análise, projeto, desenvolvimento, teste e implantação. Em cada fase são definidas, além das suas atividades, as funções e responsabilidades de cada membro da equipe, e como produto resultante, os artefatos.

Entretanto, o que diferencia um processo de *software* do outro é a ordem em que as fases ocorrem, o tempo e a ênfase dados a cada fase, as atividades presentes, e os produtos entregues.

Com o crescimento do mercado de *software*, houve uma tendência a repetirem-se os passos e as práticas que deram certo. A etapa seguinte é a formalização em modelos de ciclo de vida.

Segundo Trammel (1996), o modelo incremental pode ser definido como um modelo de ciclo de vida iterativo e incremental com características do ciclo de vida sequencial.

Pressman (2006), afirma que modelo incremental combina elementos do modelo em cascata aplicado de maneira iterativa, onde a cada incremento é fornecida uma parte do sistema aos clientes. O cliente define quais são os serviços principais e a partir da identificação dos serviços é feito o levantamento de requisitos e mapeado o primeiro incremento. Os processos de especificação, projeto, implementação e testes são feitos a cada incremento até o produto final ser entregue ao cliente gerando um sistema completo. Segundo Pressman (2006) a cada incremento entregue ao cliente final, é desenvolvido um planejamento para o próximo incremento. Este processo é feito a cada termino e início dos incrementos e tem por objetivo alterar o núcleo do produto para atender o cliente.

A iteração do usuário com o que lhe é entregue de artefato, é primordial no desenvolvimento incremental, principalmente pelo retorno (*feedback*) que é feito ao mediador da comunicação, que norteia todas as demais atividades, sejam elas as que estão sendo realizadas ou as que estão planejadas futuramente, impactando-as diretamente.

Com isso, o modelo de processo Incremental, segundo Medeiros (2016, p. 2):

“[...] combina elementos dos fluxos de processos tanto lineares quanto paralelos [...]” e identifica-se que, com o passar do tempo, o Modelo Incremental aplica sequências lineares de forma escalonada e “[...] cada uma das sequências lineares gera um incremento do *software* [...]”. Esses incrementos são entregáveis e prontos para o cliente.

A Figura 15 ilustra o modelo incremental cujo projeto EZMart se baseia.

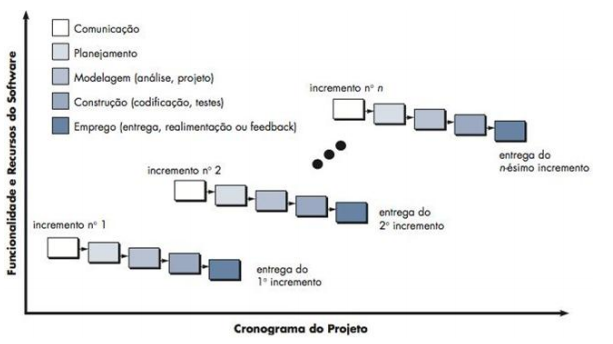


Figura 15 - Modelo Incremental

Fonte: Pressman (2010)

Com isso, o projeto é construído em quatro fases, utilizando recursos apresentados na seção 4.1.3. Cada fase proposta consiste na entrega de um incremento, conforme descrição.

1. **Incremento 1**

Entrega: 07 de abril de 2018 (data marco da Fase 1).

Itens da entrega:

1. Capítulo 2: Revisão bibliográfica;
2. Capítulo 3: Objetivo do projeto e suas seções;
3. Capítulo 4: Gerência de projeto (exceto “Gestão da Qualidade” e “Gestão de Riscos”);
4. Capítulo 5: Especificação e análise dos requisitos e suas seções;
5. Referências.
6. **Incremento 2**

Entrega: 30 de julho de 2018 (data marco da Fase 2).

Itens da entrega:

1. correções da Fase 1;
2. Capítulo 4: “Gestão de Riscos”, “Gestão da Qualidade”, “Gestão da Integração”;
3. Capítulo 5: Métricas para estimativa do esforço;
4. Capítulo 6: Arquitetura e Projeto e suas sessões (exceto “Visão Física”, “Padrões, Convenções e Guias”, “Análise da Complexidade”, “Projeto da Interação Humano-Computador”);
5. referências atualizadas;
6. codificação de no mínimo 30% dos casos de uso essenciais.
7. **Incremento 3**

Entrega: 08 de setembro de 2018 (data marco da Fase 3).

Itens da entrega:

1. correções da Fases 1 e 2;
2. Capítulo 1: Introdução;
3. Capítulo 6: Arquitetura e Projeto e suas seções (exceto “Análise da Complexidade”);
4. referências atualizadas;
5. codificação de no mínimo 60% dos casos de uso essenciais.
6. **Incremento 4**

Entrega: 10 de novembro de 2018 (data marco da Fase 4).

Itens da entrega:

1. correções da Fases 1 a 3;
2. resumo;
3. Capítulo 6: “Análise de Complexidade”;
4. Capítulo 7: Plano de Testes;
5. Capítulo 8: Plano de Implantação;
6. Capítulo 9: Conclusão;
7. obras Consultadas e Glossário;
8. codificação de casos de usos essenciais testados, dois casos de usos importantes ou desejáveis codificados e testados. Sistema pronto para demonstração.

As principais características do modelo incremental segundo Sommerville (2007) são:

1. os processos de especificação, projeto e implementação são concorrentes;
2. o sistema é desenvolvido em uma série de incrementos;
3. as ferramentas de apoio ao desenvolvimento interativo.

De acordo com Sommerville (2007), as vantagens do desenvolvimento incremental são:

1. entrega dos incrementos ao cliente (partes do sistema em funcionamento);
2. participação dos clientes ou usuários no processo de desenvolvimento incremental;

Alguns dos problemas do desenvolvimento incremental segundo Sommerville (2007) são:

1. no gerenciamento de grandes sistemas;
2. em montar um contrato para atender ambas as partes interessadas.

Portanto, enfatizando os problemas da metodologia incremental com o intuito de propiciar uma nova solução, as metodologias ágeis aparecem com um novo propósito, descrito como princípios do Manifesto para Desenvolvimento Ágil de *software*. Desta maneira, criou-se “O Manifesto para Desenvolvimento Ágil de *software*” (Manifesto Ágil, 2001), modificando a importância dos valores aplicados no desenvolvimento de *software*: valorizando-se mais indivíduos e interações, *software* em funcionamento, colaboração com o cliente e resposta às mudanças do que processos e ferramentas, documentação abrangente, negociação de contratos e seguir planos.

Este manifesto apresentou doze princípios:

1. a prioridade é satisfazer o cliente através da entrega contínua e adiantada de *software* com valor agregado;
2. mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente;
3. entregar frequentemente o *software* funcionando;
4. pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto;
5. construção de projetos em torno de indivíduos motivados;
6. proporcionar contado direto com os integrantes de modo a facilitar a comunicação entre eles;
7. *software* funcionando é a medida primária de progresso;
8. os processos ágeis promovem desenvolvimento sustentável;
9. contínua atenção à excelência técnica e bom design aumenta a agilidade;
10. simplicidade é essencial e deve ser assumida em todos os aspectos do projeto;
11. as melhores arquiteturas, requisitos e *designs* emergem de equipes auto organizadas;
12. em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo com a necessidade.

Segundo Pressman (2006), definir um processo como “ágil” significa permitir que a equipe possa adaptar e aperfeiçoar tarefas, manter apenas os trabalhos essenciais implementados, simplificar os trabalhos essenciais, fazer entregas incrementais do *software* funcionando ao cliente o mais rápido possível. Pressman (2006) afirma que o *Scrum* é uma metodologia ágil de desenvolvimento de *software*, seu nome está relacionado a uma atividade que ocorre durante o jogo de *rugby* onde o grupo de jogadores se junta ao redor da bola e trabalham juntos para mover a bola pelo campo.

Em análise, os princípios do *Scrum* compatíveis com o Manifesto Ágil são:

1. pequenas equipes;
2. time auto-organizável;
3. processo adaptado para garantir que o melhor produto seja produzido;
4. trabalho de desenvolvimento dividido, onde todos os integrantes participam de todas as atividades que compõe as entregas;
5. testes e documentação são realizados durante todo o desenvolvimento.

Os papéis no *Scrum* são:

1. *scrum master*: responsável por garantir o entendimento do processo do *Scrum*, responsável por motivar e treinar a equipe;
2. *product owner*: responsável pela priorização de requisitos que devem ser implementados, e responsável pelo gerenciamento do *product backlog* (lista de funcionalidades que serão implementadas no projeto);
3. time (Equipe): responsável pela execução e implementação das funcionalidades.

Os principais artefatos do *Scrum* são:

1. *Sprint*: é uma interação de tempo relativamente curto, porém pode variar, onde todas as *sprints* têm como resultado um incremento do produto que pode ser entregue ao cliente. Segundo Pressman (2016, p. 79):

*Sprints* consistem em unidades de trabalho solicitadas para atingir um requisito estabelecido no registro de trabalho (*backlog*) e que precisa ser ajustado dentro de um prazo já fechado. Alterações (por exemplo, itens do registro de trabalho- *backlog* *work items*) não são introduzidas durante execução de urgências (*sprints*). Portanto, o *sprint* permite que os membros de una equipe trabalhem em um ambiente de curto prazo, porém estável.

1. *Product backlog*: lista priorizada do que o produto precisa ou pode ter;
2. *backlog* da *Sprint*: lista de tarefas para produzir o *backlog* do *product* em um incremento entregável. Com isso Pressman (2016, p. 79) afirma:

*Backlog* é uma lista com prioridades dos requisitos ou funcionalidades do projeto que fornecem valor comercial ao cliente. Os itens podem ser adicionados a esse registro a qualquer momento (é assim que as alterações são introduzidas). O gerente de produto avalia o registro e atualiza as prioridades conforme solicitado.

1. *burndown*: é uma medida, pode variar dentro do contexto. Pode-se ter um *burndown* de *sprint* para medir o quanto falta de tarefas dentro do tempo da *sprint*.

Contudo, é necessário avaliar o processo o qual deve ser dotado cada metodologia. Pode-se notar que alguns conceitos do *Scrum* podem e devem ser implementados, outros não possuem tal necessidade. Mesmo com a implementação parcial desta metodologia é possível descrever fatores positivos decorrentes da quebra do paradigma do modelo incremental “puro”. Como impactos positivos dessa associação, notam-se:

1. a melhora na iteração entre os membros de diferentes papéis dentro da equipe;
2. redução dos esforços;
3. multidisciplinaridade da equipe;
4. agilidade no desenvolvimento;
5. melhor organização das tarefas e seus responsáveis.

Portanto, as fases descritas e incorporadas ao cronograma macro para o Projeto de Final de Curso (PFC) contam com o emprego da metodologia Incremental, sendo que todas as fases possuem seus pacotes de trabalho e que no fim produzem um incremento para o cliente. Na fase da montagem dos pacotes de trabalho é onde a metodologia *Scrum* é inserida. Esse feito traz uma melhor estruturação e organização interna da equipe, assim como norteia o desenvolvimento baseado na agilidade.

Desta maneira, o *Scrum* baseia-se em ferramentas com forte viés visual tendo sido aplicadas ao gerenciamento ágil de projetos, como ***Kanban*** — técnica desenvolvida pelo engenheiro industrial *Taiichi Ohno*, da *Toyota*, que significa em japonês mural de registro ou placa visível — e ***Project Model Canvas*, como exemplificado na Figura 16. O gerenciamento ágil ou o modelo de ciclo de vida adaptativo é exposto no item 4.1.2.1 conforme o PMI (2013).**

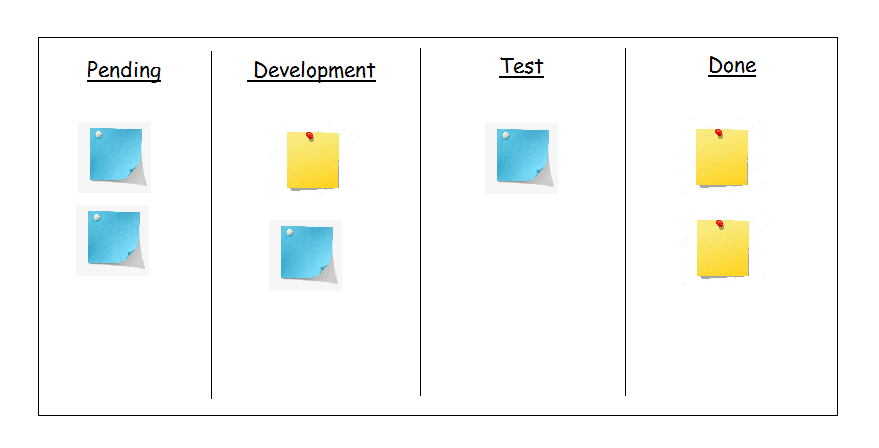


Figura 16 - Quadro Kanban

Fonte: Equipe (2018)

#### 4.1.2.1 Ciclo de Vida Adaptativo

Os ciclos de vida adaptativos são conhecidos como direcionados à mudança ou utilizadores de métodos ágeis. Segundo PMI (2013, p. 46), os modelos adaptativos são projetados para reagir a altos níveis de mudança e envolvimento contínuo das partes interessadas.

Os métodos adaptativos são também iterativos e incrementais, a diferença é que as iterações são muito rápidas (geralmente com uma duração de 2 a 4 semanas), com tempo e recursos fixos. Os projetos adaptativos geralmente executam vários processos em cada iteração, embora as primeiras iterações possam se concentrar mais nas atividades de planejamento.

O escopo do projeto pode ser decomposto em um conjunto de requisitos e trabalhos a serem executados, chamado de *backlog* do projeto. É afirmado no PMI (2013, p. 46) “que no início de uma iteração, a equipe trabalhará para determinar a quantidade de itens altamente prioritários da lista de *backlog* que podem ser entregues na próxima iteração”.

Ainda PMI (2013, p. 46), descreve que:

No final de cada iteração, o produto deve estar pronto para a análise pelo cliente. Isso não significa que o cliente deve aceitar a entrega, mas simplesmente que o produto não deve incluir características inacabadas, incompletas, ou que não podem ser usadas. Os representantes do patrocinador e do cliente devem estar continuamente envolvidos no projeto para fornecer o *feedback* sobre as entregas à medida que elas são criadas, a fim de garantir que o *backlog* do produto reflita suas necessidades atuais.

Os métodos adaptativos, conhecidos como métodos ágeis, como o *Scrum*, utilizado no projeto EzMart com colaboração do modelo de ciclo de vida incremental, geralmente são preferidos quando se lida com um ambiente em rápida mutação, quando os requisitos e escopo são difíceis de definir antecipadamente, e quando é possível definir pequenas melhorias incrementais que agregarão valor às partes interessadas.

### 4.1.3 Recursos Necessários

Existem duas saídas do processo de estimativa de recursos necessários para uma determinada atividade e são elas:

1. a identificação dos recursos;
2. a descrição dos recursos, com as devidas quantidades e tipos de recursos.

A quantidade de detalhes e o nível de especificação podem variar por área de aplicação. A documentação dos recursos necessários pode incluir quais os tipos de recursos que devem ser aplicados além das premissas feitas para determinar a disponibilidade dos recursos e em que quantidades em que são utilizados. É no processo de desenvolvimento do cronograma ou em responsabilidade a coordenação do projeto, que os recursos necessários são especificados.

São utilizados ao longo do projeto recursos humanos, de *hardware* e de *software*, como mostrado a seguir.

#### 4.1.3.1 Recursos Humanos

Gerenciar os recursos humanos de um projeto inclui organizar, guiar e controlar toda a equipe do projeto. Segundo o PMI (2013, p.255):

A equipe do projeto consiste das pessoas com papéis e responsabilidades designadas para completar o projeto. Os membros da equipe do projeto podem ter vários conjuntos de habilidades, atuar em regime de tempo integral ou parcial, e podem ser acrescentados ou removidos da equipe à medida que o projeto progride. Os membros da equipe do projeto também podem ser referidos como pessoal do projeto. Embora os papéis e responsabilidades específicos para os membros da equipe do projeto sejam designados, o envolvimento de todos os membros da equipe no planejamento do projeto e na tomada de decisões pode ser benéfico. A participação dos membros da equipe durante o planejamento agrega seus conhecimentos ao processo e fortalece o compromisso com o projeto.

Desta maneira, a equipe é formada por cinco estudantes do 4º ano de Sistemas de Informação, nomeados no Quadro 1.

|  |  |
| --- | --- |
| Integrante | Responsabilidade |
| Camila Maria da Costa | Pesquisadora e testadora. |
| Diego Vinícius Cesar do Amaral | Pesquisador e responsável pela arquitetura do sistema. |
| Jéssica Souza Pivoto | *Scrum* *master* e *product owner.* |
| Marcos Henrique Azevedo | Desenvolvedor da arquitetura de Banco de Dados. |
| Marcos Paulo Moreno Pereira | Desenvolvedor *Front-end.* |

Quadro 1- Quadro dos integrantes

Fonte: Equipe (2018)

#### 4.1.3.2 Recursos de Hardware

Como recursos de *hardware*, são necessários todos os dispositivos físicos ou equipamentos utilizados para processamento de informações. Esses recursos são os equipamentos utilizados pela equipe no projeto EzMart com o mesmo objetivo. Eles estão especificados no Quadro 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Equipamento | Espaço de armazenamento permanente | Processador | Memória RAM | Sistema operacional |
| *Notebook Dell* | 1TB | I7 | 8 GB | *Windows* 10 |
| *Notebook* Positivo | 500GB | Intel Celeron | 2 GB | *Windows* 10 |
| *Notebook* Asus | 1TB | I5 | 8 GB | *Windows* 10 |
| *Desktop* Asus | 500 GB | AMD Phenon IIx4 | 12 GB | *Windows* 7 |

Quadro 2- Recursos de hardware

Fonte: Equipe (2018)

#### 4.1.3.3 Recursos de Software

Como recursos de *softwares* são necessários alguns *softwares* gerenciais ou como auxiliares na execução das tarefas ao longo do projeto e do cronograma EzMart. Os *softwares* que auxiliam na execução das tarefas são conhecidos como ferramentas CASE. Segundo Pressman (2002), são ferramentas automatizadas que auxiliam gerentes e profissionais de engenharia de *software* em toda atividade com o processo de *software*. Desta maneira, pode-se compreender que todas as ferramentas necessárias e descritas no Quadro 3, compõe um conjunto de ferramentas que auxiliam os integrantes da equipe durante uma ou mais fases do processo de desenvolvimento do EzMart.

|  |  |
| --- | --- |
| Ferramenta | Versão |
| Bootstrap | 3.3.7 |
| BrModelo | 2.0 |
| Google Chrome | 57.0.2987.93 |
| Ideas Modeler | 10 |
| Marvel | 0.9.10 |
| Microsoft Office Excel, Power Point e Word | 2013 / 2016 |
| Microsoft Project | 2013 |
| NetBeans | 8.1 |
| PostgreSQL | 9.4 |
| SpringMVC | 3 |
| SeleniumIDE | 3.0.1 |
| WBS Tool | 0.9 Beta |

Quadro 3- Recursos de Software

Fonte: Equipe (2018)

## 4.2 ÁREAS DE CONHECIMENTO

De acordo PMI (2013, p. 60), são dez áreas de conhecimento abordadas na gerência de um projeto, sendo elas o “Gerenciamento do escopo”, o “Gerenciamento do tempo”, o “Gerenciamento da integração”, o “Gerenciamento dos custos”, o “Gerenciamento da qualidade”, o “Gerenciamento dos recursos humanos”, o “Gerenciamento das comunicações”, o “Gerenciamento dos riscos”, o “Gerenciamento das aquisições” e o “Gerenciamento das partes interessadas”.

Nesta seção são abordados os processos das áreas de conhecimento: escopo, tempo, integração, qualidade e riscos e como o gerenciamento destes estão aplicados no projeto.

### 4.2.1 Gestão do Escopo

Segundo Tenstep (2007), o escopo é a maneira de descrever os direitos e obrigações entre a empresa-cliente (ou um contratante qualquer) e a consultoria (ou um contratado qualquer), definindo o que a contratada irá realizar e o que não irá realizar. A finalidade de definir o escopo é descrever e obter um consenso sobre os limites lógicos da atuação da contratada.

PMI (2013, p. 105), contribui com a explicação de Tennstep (207) afirmando que:

O gerenciamento do escopo do projeto inclui os processos necessários para assegurar que o projeto inclui todo o trabalho necessário, e apenas o necessário, para terminar o projeto com sucesso. O gerenciamento do escopo do projeto está relacionado principalmente com a definição e controle do que está e do que não está incluso no projeto.

Os processos desta área de conhecimento interagem entre si e também com outras áreas de conhecimento. A Figura 17 apresenta a visão geral do gerenciamento do escopo do projeto, com uma descrição de cada uma de suas atividades.

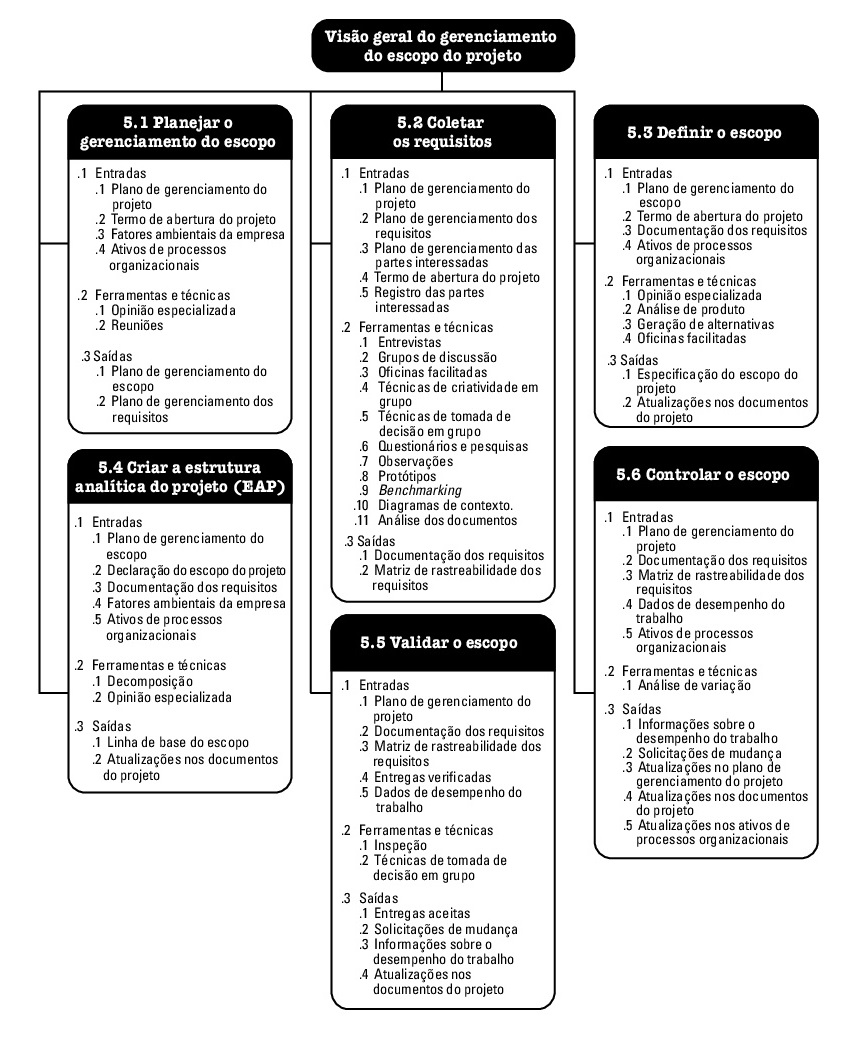


Figura 17 - Visão geral do gerenciamento do escopo de um projeto ­­

Fonte: PMI (2013, p. 106)

Segundo PMI (2013, p.105) e de acordo com a Figura 17, um projeto deve:

1. planejar o gerenciamento do escopo: o processo de criar um plano de gerenciamento do escopo do projeto que documenta como tal escopo será definido, validado e controlado;
2. coletar os requisitos: o processo de determinar, documentar e gerenciar as necessidades e requisitos das partes interessadas a fim de atender aos objetivos do projeto;
3. definir o escopo: o processo de desenvolvimento de uma descrição detalhada do projeto e do produto;
4. criar a EAP: o processo de subdivisão das entregas e do trabalho do projeto em componentes menores e mais facilmente gerenciáveis;
5. validar o escopo: o processo de formalização da aceitação das entregas concluídas do projeto;
6. controlar o escopo: o processo de monitoramento do andamento do escopo do projeto e do produto e gerenciamento das mudanças feitas na linha de base do escopo.

Ainda, o PMI (2013, p.105) afirma que:

Os processos usados para gerenciar o escopo do projeto, bem como as ferramentas e técnicas de suporte, podem variar por projeto. A linha de base do escopo para o projeto é a versão aprovada da especificação do escopo do projeto, da estrutura analítica do projeto (EAP), e o respectivo dicionário da EAP. Uma linha de base só pode ser alterada através de procedimentos formais de controle de mudança e é usada como uma base de comparação durante a execução dos processos validar o escopo e controlar o escopo, bem como outros processos de controle.

Desta maneira, a conclusão do escopo do projeto é medida em relação ao plano de gerenciamento do projeto. Os processos de gerenciamento do projeto mostrados na Figura 18 precisam e devem estar bem integrados aos das outras áreas de conhecimento para que o trabalho do projeto resulte na entrega do escopo do produto especificado como EzMart.

O PMI (2013, p. 120) auxilia na compreensão do escopo do projeto, acrescentando que a definição do escopo do projeto é o processo de desenvolvimento de uma descrição detalhada do projeto e do produto. O principal benefício desse processo é que ele descreve os limites do projeto, serviços ou resultados ao definir quais dos requisitos coletados serão incluídos e quais serão excluídos do escopo do projeto. As entradas, ferramentas e técnicas, e saídas desse processo são ilustradas na Figura 18.

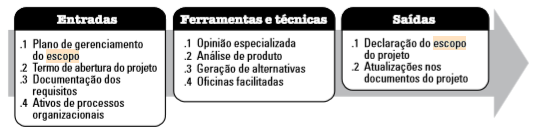


Figura 18 - Definir o escopo: entradas, ferramentas e técnicas, e saídas

Fonte: PMI (2013, p.120)

A Figura 19 ilustra o diagrama de fluxo de processos.

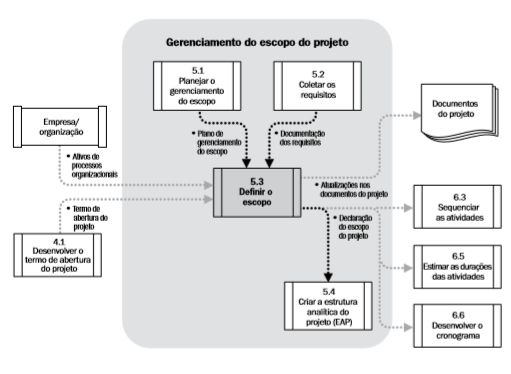


Figura 19 - Diagrama do fluxo de dados do processo “Definir escopo”

Fonte: PMI (2013, p. 120)

Desta maneira, já que todos os requisitos identificados no processo “Coletar requisitos” podem não estar incluídos no projeto, neste momento o processo “Definir o escopo” seleciona os requisitos finais do projeto a partir da documentação de requisitos entregue durante o processo “Coletar requisitos”. Em seguida define uma descrição detalhada do projeto e produto, do serviço ou resultado.

Segundo o PMI (2013, p. 121) a preparação detalhada da especificação do escopo é crítica para o sucesso do projeto e baseia-se nas entregas principais, premissas e restrições que são documentadas durante a iniciação do projeto.

Durante o planejamento do projeto, o seu escopo é definido e descrito com maior especificidade conforme as informações a respeito do projeto são conhecidas. Os riscos existentes, premissas e restrições são analisados para verificar sua integridade e acrescentados ou atualizados conforme necessário. O processo definir o escopo pode ser altamente iterativo.

Em projetos de ciclo de vida incremental, deve ser desenvolvida uma visão de alto nível para o projeto em geral, mas o escopo detalhado é determinado em uma iteração de cada vez e o planejamento detalhado para a iteração seguinte é executado à medida que o trabalho no escopo do projeto e entregas atuais avança. Neste caso, à medida que o trabalho no escopo do projeto avança, os requisitos podem sofrer mudanças que impactem diretamente no escopo, e por esses motivos deve-se aplicar a matriz de rastreabilidade de requisitos. PMI (2013) afirma que a matriz de rastreabilidade é necessária a medida que há mudanças no escopo do produto e a define como:

A matriz de rastreabilidade de requisitos é uma tabela que liga os requisitos de produto desde as suas origens até as entregas que os satisfazem. A utilização de uma matriz de rastreabilidade ajuda a garantir que cada requisito adiciona valor de negócio através da sua ligação aos objetivos de negócio e aos objetivos do projeto. Ela fornece um meio de rastreamento do início ao fim do ciclo de vida do projeto, ajudando a garantir que os requisitos aprovados na documentação sejam entregues no final do projeto. Finalmente, ela fornece uma estrutura de gerenciamento das mudanças do escopo do produto.

A Figura 20 exemplifica uma matriz de rastreabilidade segundo o PMBoK (2013), onde os atributos associados a cada requisito devem ser registrados na matriz, pois auxiliam na definição de informações chave a respeito do requisito. PMBoK (2013, p. 119) afirma que:

Os atributos típicos usados na matriz de rastreabilidade dos requisitos podem incluir: um identificador único, uma descrição textual do requisito, os argumentos para sua inclusão, proprietário, fonte, prioridade, versão, status atual (se está ativo, cancelado, adiado, adicionado, aprovado, designado, concluído) e a data do status.

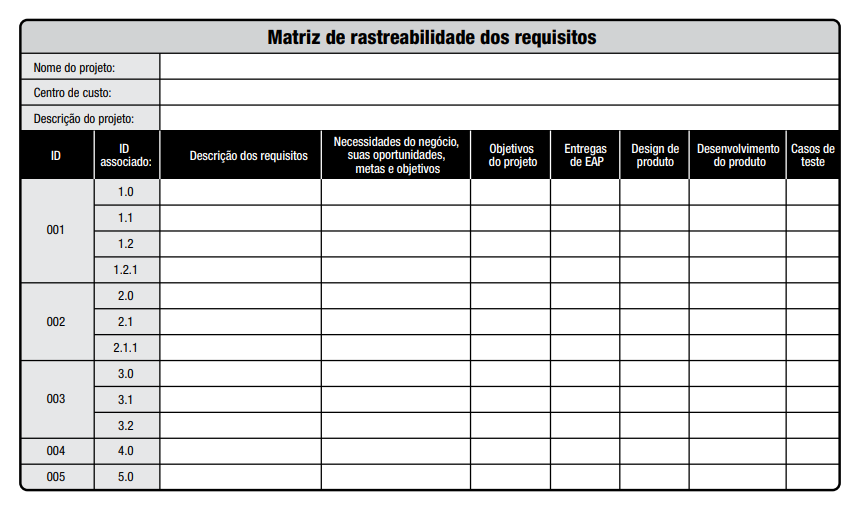


Figura 20 - Exemplo de uma matriz de rastreabilidade de requisitos

Fonte: PMI (2013, p. 119)

Desta maneira, é possível que caso haja qualquer mudança no escopo do projeto, haja também a adequação ou a rastreabilidade dos requisitos que contemplam esta mudança.

#### 4.2.1.1 Estrutura Analítica de Projeto

Como define o PMI (2013, p. 105), o processo de “subdivisão das entregas e do trabalho do projeto em componentes menores e mais facilmente gerenciáveis” permite a elaboração da EAP.

Dessa forma, o objetivo da EAP é reduzir a complexidade do projeto, decompondo-o em partes menores, denominadas entregas. Decompondo os pacotes até se tornarem gerenciáveis, é possível determinar para cada um deles o escopo, duração e esforço.

Para Carvalho e Rabechini Júnior (2011, p. 90), a “arte de fazer uma boa estrutura analítica é dominar a decomposição. Enquanto o gerente estiver inseguro quanto às estimativas, é necessário continuar decompondo em níveis menores de trabalho”.

Para PMI (2013, p. 109), no plano de gerenciamento do escopo está incluso “o processo que habilita a criação da EAP a partir da especificação do escopo do projeto detalhada e o processo que estabelece como a EAP será mantida e aprovada. ”.

A EAP do projeto EzMart encontra-se no Apêndice A deste documento.

#### 4.2.1.2 Dicionário EAP

De acordo com PMI (2013, p.132), o dicionário da EAP é um “documento que fornece informações detalhadas sobre entregas, atividades e agendamento de cada componente da estrutura analítica do projeto (EAP). O dicionário da EAP é um documento que dá suporte à EAP”.

Segundo PMI (2013), estão contidas no dicionário da EAP algumas informações, como: marcos do cronograma, recursos necessários, estimativa de custos, organização responsável, descrição do trabalho, entre outras.

Para complementar essas informações, o PMI (2013, p. 131) afirma que “a linha de base do escopo é a versão aprovada de uma especificação de escopo do projeto, de uma estrutura analítica do projeto (EAP) e seu dicionário da EAP associado, que só pode ser mudada através de procedimentos de controle formais, e é usada como uma base de comparação. ” As informações contidas no dicionário da EAP incluem (mas não estão limitadas) à:

1. código de identificador da conta;
2. descrição do trabalho;
3. premissas e restrições;
4. organização responsável;
5. marcos do cronograma;
6. atividades do cronograma associadas;
7. recursos necessários;
8. estimativa de custos;
9. requisitos de qualidade;
10. critérios de aceitação;
11. referências técnicas; e
12. informações sobre acordos.

O Dicionário da EAP encontra-se no Apêndice A deste documento, descrito no campo "Notes" do arquivo gerado pelo MS-Project.

### 4.2.2 Gestão do Tempo

Segundo a norma técnica NBR ISO 10006 (ABNT, 1997, p. 2), projeto é "processo único, consistido de um grupo de atividades coordenadas e controladas com datas para início e término, empreendido para alcance de um objetivo conforme requisitos específicos, incluindo limitações de tempo, custo e recursos".

De acordo com o PMI (2013, p. 141), os processos necessários para gerenciar o tempo de um projeto são:

1. planejar o gerenciamento do cronograma – o processo de estabelecer as políticas, os procedimentos e a documentação para o planejamento, desenvolvimento, gerenciamento, execução e controle do cronograma do projeto;
2. definir as atividades – o processo de identificação e documentação das ações específicas a serem realizadas para produzir as entregas do projeto;
3. sequenciar as atividades – o processo de identificação e documentação dos relacionamentos entre as atividades do projeto;
4. estimar os recursos das atividades – o processo de estimativa dos tipos e quantidades de material, recursos humanos, equipamentos ou suprimentos que serão necessários para realizar cada atividade;
5. estimar as durações das atividades – o processo de estimativa do número de períodos de trabalho que serão necessários para terminar atividades específicas com os recursos estimados;
6. desenvolver o cronograma – o processo de análise das sequências das atividades, suas durações, recursos necessários e restrições do cronograma visando criar o modelo do cronograma do projeto;
7. controlar o cronograma – o processo de monitoramento do andamento das atividades do projeto para atualização no seu progresso e gerenciamento das mudanças feitas na linha de base do cronograma para realizar o planejado.

Assim como os processos contidos na área de conhecimento “Escopo”, os processos desta área do conhecimento também interagem com outras áreas do conhecimento.

PMI (2013, p.141) afirma que em alguns projetos, especialmente naqueles de escopo menor, onde se pode compreender o projeto EzMart, os processos “definir as atividades”, “sequenciar as atividades”, “estimar os recursos das atividades”, “estimar as durações das atividades” e “desenvolver o modelo do cronograma” estão tão estreitamente conectados que são vistos como um único processo que pode ser realizado por uma pessoa em um período de tempo relativamente curto. Estes processos são aqui representados como elementos distintos, pois as ferramentas e técnicas para cada processo são diferentes e são tratados a seguir.

Com isso, é necessário que se tenha um planejamento do cronograma, onde é também necessário estabelecer as políticas, os procedimentos e a documentação para o planejamento, desenvolvimento, gerenciamento, execução e controle do cronograma do projeto e o PMI (2013, p. 145) afirma que “o principal benefício deste processo é o fornecimento de orientação e instruções sobre como o cronograma do projeto será gerenciado ao longo de todo o projeto. ”.

A Figura 21 demonstra o diagrama do fluxo de dados do processo de planejar o gerenciamento do cronograma

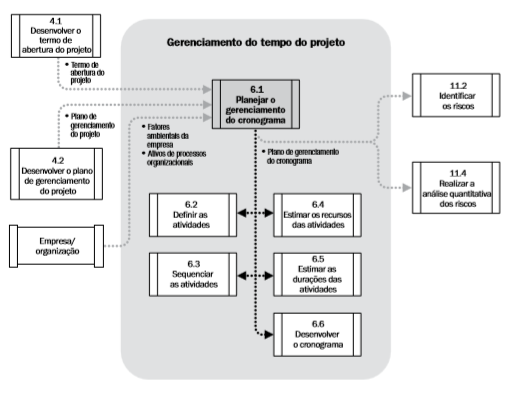


Figura 21 - Diagrama do fluxo de dados do processo de planejar o gerenciamento do cronograma

Fonte: PMI (2013, p.145)

O plano de gerenciamento do cronograma é um componente do plano de gerenciamento do projeto, segundo o PMI (2013, p. 146):

O plano de gerenciamento do cronograma pode ser formal ou informal, altamente detalhado ou generalizado, baseado nas necessidades do projeto, e inclui os limites de controle apropriados. O plano de gerenciamento do cronograma define como as contingências do cronograma serão reportadas e avaliadas. O plano de gerenciamento do cronograma pode ser atualizado para refletir uma mudança na maneira como o cronograma é gerenciado.

Portanto, é imprescindível gerenciar o cronograma, e isso é feito no projeto EzMart, onde o gerente de projeto tem o papel de definir as atividades, sequenciá-las, definir o tempo de duração e atentar-se aos momentos ao longo do projeto em que houve mudanças no escopo e que impactaram na reestimativa das atividades, sequenciamento ou definição da duração das mesmas.

O cronograma de atividades do projeto encontra-se no Apêndice B deste documento.

#### 4.2.2.1 Lista de atividades, diagrama de rede e cronograma

Segundo o PMI (2013, p. 150) a definição das atividades está no processo de “Definir atividades” e no fluxo de dados do processo, encontra-se como artefato de saída deste processo, assim como mostra a Figura 22.

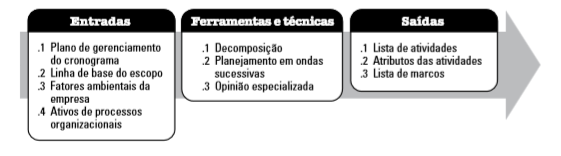


Figura 22 - Definir atividades: entradas, ferramentas e técnicas, e saída

Fonte: PMI (2013, p.149)

Desta maneira, o PMI (2013, p. 150) afirma que:

Implícitos neste processo estão a definição e o planejamento das atividades do cronograma de tal modo que os objetivos do projeto sejam alcançados. O processo Criar a EAP identifica as entregas no nível mais baixo da estrutura analítica do projeto (EAP), o pacote de trabalho. Os pacotes de trabalho são tipicamente decompostos em componentes menores chamados atividades que representam o esforço de trabalho necessário para completar o pacote de trabalho.

Portanto, a lista de atividades apresenta todas as atividades que deverão ser realizadas para o cumprimento dos pacotes de trabalho no tempo adequado e de acordo com o cronograma. Estas atividades estão contidas dentro de cada pacote de trabalho da Estrutura Analítica de Projeto (EAP). Desta maneira, para finalizar um pacote de trabalho, é necessário que as suas respectivas atividades sejam desenvolvidas. Entretanto, para que seja possível finalizar cada atividade, é necessário decompô-las para que seja possível gerenciá-las. Assim, o PMI (2013, p. 150), afirma que “a decomposição é uma técnica usada para dividir e subdividir o escopo do projeto e suas entregas em partes menores e mais fáceis de gerenciar. ”.

Segundo o PMI (2013, p. 152):

A lista de atividades é uma lista abrangente que inclui todas as atividades do cronograma necessárias no projeto. A lista de atividades também inclui o identificador de atividades e uma descrição do escopo de trabalho de cada atividade em detalhe suficiente para assegurar que os membros da equipe do projeto entendam qual trabalho precisa ser executado. Cada atividade deve ter um título exclusivo que descreve o seu lugar no cronograma, mesmo que tal atividade seja mostrada fora do contexto do cronograma do projeto.

No entanto, a lista de atividades, a EAP, e o dicionário da EAP podem ser desenvolvidos sequencialmente ou paralelamente, com ambos servindo de base para o desenvolvimento da lista final das atividades. Segundo PMI (2013, p. 150), cada pacote de trabalho é decomposto em atividades menores necessárias para a produção das entregas previstas por ele. O envolvimento de membros da equipe na decomposição pode gerar resultados melhores e mais precisos.

De acordo com isso, é necessário que depois de definir todas as atividades, haja o sequenciamento das atividades, para que seja possível a criação de uma lógica entre elas. Isso auxilia no gerenciamento, observando-se qual atividade já terminou e quais as faltantes para completar um pacote de trabalho, pois as atividades possuem durações, durante as quais o trabalho daquela atividade é executado.

PMI (2013, p. 150) define o sequenciamento de atividades como “o processo de identificação e documentação dos relacionamentos entre as atividades do projeto”. E, informa que o principal benefício deste processo é definir a sequência lógica do trabalho a fim de obter o mais alto nível de eficiência em face de todas as restrições do projeto.

Para que seja possível o sequenciamento das atividades, após sua definição, o PMI (2013, p. 150) afirma que todas as atividades e marcos, com exceção do primeiro e do último, devem ser conectados a pelo menos um predecessor com uma relação lógica término para início ou início para início e a pelo menos um sucessor com uma relação lógica término para início ou término para término.

As relações lógicas devem ser projetadas para criar um cronograma de projeto realista. O uso de tempo de antecipação ou de espera pode ser necessário entre as atividades para dar suporte a um cronograma de projeto realista e executável. O sequenciamento pode ser executado através do uso de software de gerenciamento de projetos ou do uso de técnicas manuais ou automatizadas (PMI, 2013, p. 150).

De acordo com o sequenciamento de atividades, é necessário que seja aplicado um método de precedência em todas as atividades que foram definidas, e o método aplicado neste projeto na EAP, é o método do diagrama de precedência, que segundo o PMI (2013, p. 156):

O método do diagrama de precedência (MDP) é uma técnica usada para construir um modelo de cronograma em que as atividades são representadas por nós e ligadas graficamente por um ou mais relacionamentos lógicos para mostrar a sequência em que as atividades devem ser executadas. Atividade no nó (ANN) é um método de representação de um diagrama de precedência. Este é o método usado pela maioria dos pacotes de software de gerenciamento de projetos.

A Figura 23 ilustra os tipos de relações entre as atividades, que podem ser de quatro maneiras diferentes:

1. término para início (TI). Um relacionamento lógico em que uma atividade sucessora não pode começar até que uma atividade predecessora tenha terminado. Exemplo: Uma cerimônia de entrega de prêmios (sucessora) não pode começar até que a corrida (predecessora) termine;
2. término para término (TT). Um relacionamento lógico em que uma atividade sucessora não pode terminar até que a atividade predecessora tenha terminado. Exemplo: A redação de um documento (predecessora) deve ser terminada antes que o documento seja editado (sucessora);
3. início para início (II). Um relacionamento lógico em que uma atividade sucessora não pode ser iniciada até que uma atividade predecessora tenha sido iniciada. Exemplo: A nivelação do concreto (sucessora) não pode ser iniciada até que a colocação da fundação (predecessora) seja iniciada.;
4. início para término (IT). Um relacionamento lógico em que uma atividade sucessora não pode ser terminada até que uma atividade predecessora tenha sido iniciada. Exemplo: O primeiro turno da guarda de segurança (sucessora) não pode terminar até que o segundo turno da guarda de segurança (predecessora) comece.

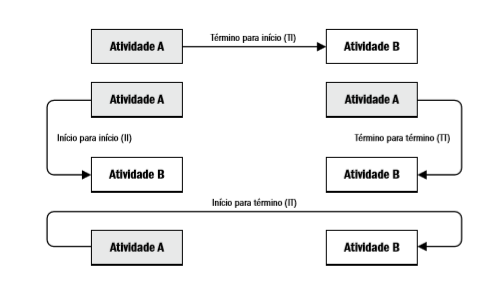


Figura 23 - Método do diagrama de precedência – tipos de relações

Fonte: PMI (2013, p. 157)

Com isso, nas saídas do processo de “Definir atividades” encontra-se também o diagrama de rede do cronograma do projeto, onde o PMI (2013, p.159) afirma que uma representação gráfica das relações lógicas, também chamadas de dependências, entre as atividades do cronograma do projeto.

O diagrama de rede pode ser produzido manualmente ou utilizando-se um *software* para gerenciamento de projetos. A Figura 24 representa um exemplo de diagrama de rede.

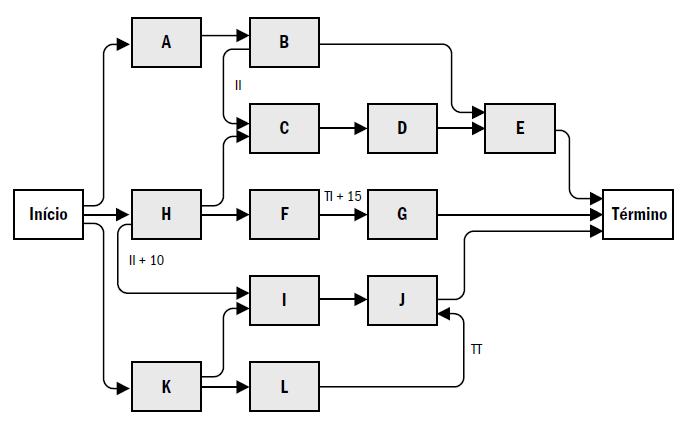


Figura 24 - Exemplo de diagrama de rede de um projeto

Fonte: Adaptada de Pressman (2011)

As atividades e o sequenciamento das mesmas encontram-se disponíveis em um cronograma de atividades. Portanto, é a partir do cronograma de atividades que as atividades são definidas, sequenciadas e estimadas em relação ao seu período de duração.

As atividades do projeto EzMart encontram-se no Apêndice B deste documento.

#### 4.2.2.2 Quadro com resumo do esforço

O quadro com resumo do esforço tem como objeto a apresentação do esforço estimado pela equipe EzMart em comparativo com o esforço efetivamente realizado. Depois de obter a duração de cada pacote de trabalho que compõe a entrega, o esforço estimado pode ser obtido do dicionário da EAP. Além disso, no início de cada fase, é feita uma estimativa *Ad-hoc* pela equipe.

Portanto, o Quadro 4 é dividido em quatro fases de acordo com o cronograma do Projeto Final de Curso (PFC), e tem três pontos de controle: Estimativa *Ah-hoc*, Dicionário da EAP e Efetivo Realizado.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1ª. Fase** | **2ª. Fase** | **3ª. Fase** | **4ª. Fase** |
| Estimativa *Ad-hoc* | 400/400h | 600/1000 | 1000/2000 |  |
| Dicionário EAP | 356/356h | 523/879 | 878/1757 |  |
| Efetivo Realizado | 250/250h | 508/758 | 664/1422 |  |

Quadro 4- Resumo do esforço

Fonte: Equipe (2018)

Durante o planejamento da primeira fase do projeto, a equipe estimou um alto esforço, pois não possuía experiência com relação ao planejamento das atividades e ao manejo correto da ferramenta de EAP, de modo a utilizá-la ao favor de uma melhor administração das atividades. Ao final da primeira fase, o efetivo realizado ficou abaixo do planejado, isso se deve ao fato de que as atividades foram planejadas com muitas horas e, na prática, o tempo gasto ficou abaixo do esperado.

Durante o planejamento da segunda fase do projeto, a equipe estimou um esforço de 600h devido à complexidade e quantidade de atividades. Com o amadurecimento da equipe em relação à primeira fase o efetivo realizado ficou mais próximo do estimado, com 508Hh.

Mediante o planejamento da terceira fase, a equipe estimou 1000 horas de esforço dividido entre todos os integrantes. Por se tratar da etapa onde deve ser entregue 60% dos Casos de Uso implementados, e seus respectivos testes, justifica-se o aumento das horas em relação à segunda fase. Entretanto, o efetivo realizado ficou abaixo do previsto, em função do adiantamento das atividades. Como havia folga dentre as atividades e a equipe comprometeu-se em ultrapassar a meta de 60% dos Casos de Uso implementados, quase todas as atividades foram antecipadas, o que gerou horas a menos no efetivo realizado.

### 4.2.3 Gestão da Integração

O gerenciamento da integração do projeto inclui os processos e atividades para identificar, definir, combinar, unificar e coordenar os vários processos e atividades dentro dos grupos de processos de gerenciamento do projeto. Segundo o PMI (2013, p. 63):

No contexto de gerenciamento de projetos, integração inclui características de unificação, consolidação, comunicação e ações integradoras que são essenciais para a execução controlada do projeto até a sua conclusão, a fim de gerenciar com sucesso as expectativas das partes interessadas e atender aos requisitos.

Elementos para o controle de alterações em projetos também são fornecidas na gestão de integração, visto que, elas precisam ser administradas.

Dessa forma, a gestão da integração tem extrema importância para o projeto, uma vez que por meio dela é feito o monitoramento do andamento do projeto, o controle das alterações e o controle do desempenho de suas atividades.

Segundo o PMI (2013, p. 63), os processos da gestão da integração são:

1. desenvolver o termo de abertura do projeto - o processo de desenvolver um documento que formalmente autoriza a existência de um projeto e dá ao gerente do projeto a autoridade necessária para aplicar recursos organizacionais às atividades do projeto;
2. desenvolver o plano de gerenciamento do projeto - o processo de definir, preparar e coordenar todos os planos subsidiários e integrá-los a um plano de gerenciamento de projeto abrangente. As linhas de base e os planos subsidiários integrados do projeto podem ser incluídos no plano de gerenciamento do projeto;
3. orientar e gerenciar o trabalho do projeto - o processo de liderar e realizar o trabalho definido no plano de gerenciamento do projeto e a implementação das mudanças aprovadas para atingir os objetivos do projeto;
4. monitorar e controlar o trabalho do projeto - o processo de acompanhar, revisar e registrar o progresso do projeto para atender aos objetivos de desempenho definidos no plano de gerenciamento do projeto;
5. realizar o controle integrado de mudanças - o processo de revisar todas as solicitações de mudança, aprovar as mudanças e gerenciar as mudanças nas entregas, ativos de processos organizacionais, documentos do projeto e no plano de gerenciamento do projeto, e comunicar a decisão sobre os mesmos;
6. encerrar o projeto ou fase - o processo de finalização de todas as atividades de todos os grupos de processos de gerenciamento do projeto para encerrar formalmente o projeto ou a fase.

Ainda, de acordo com Carvalho e Rabechini Júnior (2011, p. 69), "A integração exerce papel essencial no gerenciamento de projetos na medida em que, inicialmente, cria as condições propícias para o desenvolvimento adequado de um projeto". Portanto, essa seção contempla aspectos de gerenciamento como monitoramento do projeto, controle da configuração e controle de mudanças no escopo.

A necessidade do gerenciamento da integração do projeto fica evidente em situações onde os processos distintos interagem. No projeto EzMart, há a necessidade do gerenciamento da integração, pois há a situação onde os processos distintos interagem, por exemplo, processos do modelo incremental e processos do modelo ágil *Scrum.* Com isso, o PMI (2013, p. 63), afirma que “as entregas do projeto também podem precisar ser integradas às operações em progresso da organização executora ou da organização do cliente e ao planejamento estratégico de longo prazo que considera problemas ou oportunidades futuras. ”.

A Figura 25 demonstra a visão geral do gerenciamento da integração de um projeto.

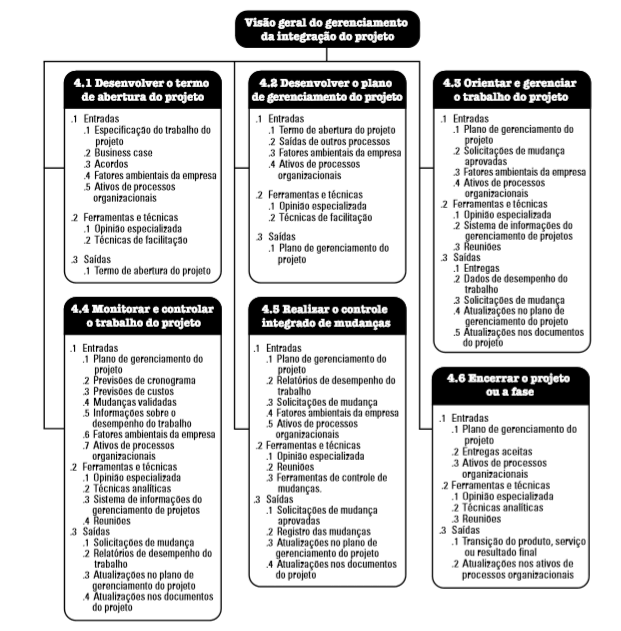


Figura 25 - Visão geral do gerenciamento da integração de um projeto

Fonte: PMI (2013, p. 63)

De acordo com a estrutura geral da integração, os subitens dessa seção podem ser compreendidos e detalhados a seguir como Monitoramento, Controle de configuração e Controle de mudanças.

#### 4.2.3.1 Monitoramento

De acordo com o PMI (2013, p. 88), “O monitoramento é um aspecto do gerenciamento executado do início ao término do projeto”. Com isso, a equipe de gerenciamento do projeto consegue obter informações claras sobre o andamento do mesmo, de modo que, se consiga identificar possíveis áreas que venham a necessitar de maior atenção para sua conclusão.

O monitoramento pertence à um grupo de processos, e consiste dos processos necessários para acompanhar, analisar e organizar o progresso e o desempenho do projeto em desenvolvimento, neste caso, o EzMart. Segundo PMI (2013, p.57):

O grupo de processos de monitoramento e controle consiste dos processos necessários para acompanhar, analisar e organizar o progresso e o desempenho do projeto; identificar quaisquer áreas nas quais serão necessárias mudanças no plano; e iniciar as respectivas mudanças. O principal benefício deste grupo de processos é a medição e análise do desempenho do projeto a intervalos regulares, em ocorrências apropriadas ou em condições excepcionais, a fim de identificar as variações no plano de gerenciamento do projeto.

Este grupo de processos, de acordo com o PMI (2013, p. 57) envolve:

1. controlar as mudanças e recomendar ações corretivas ou preventivas em antecipação a possíveis problemas;
2. monitorar as atividades contínuas do projeto em relação ao plano de gerenciamento do projeto e a linha de base de desempenho do mesmo, e,
3. influenciar os fatores que poderiam impedir o controle integrado de mudanças ou de gerenciamento de configurações para que somente as mudanças aprovadas sejam implementadas.

Para controle e monitoramento do projeto EzMart, está disposto no Apêndice C o relatório de desempenho do projeto, que consiste na análise, organização e o acompanhamento de todas as fases do projeto. Este monitoramento é importante, pois fornece à equipe do projeto uma visão mais ampla, estruturada e compactada através do diagrama. Isso é relatado pelo PMI (2013, p. 57):

O grupo de processos de monitoramento e controle não apenas monitora e controla o trabalho sendo feito dentro do grupo de processos, mas também monitora e controla todo o esforço do projeto. Nos projetos de várias fases, o grupo de processos de monitoramento e controle coordena as fases do projeto para implementar ações corretivas ou preventivas para que o projeto cumpra o seu plano de gerenciamento. Esta revisão pode resultar em atualizações recomendadas e aprovadas para o plano de gerenciamento do projeto. Por exemplo, uma data de término de atividade não cumprida pode exigir ajustes e compensações entre os objetivos do orçamento e do organograma. A fim de reduzir ou controlar as despesas indiretas, procedimentos de gerenciamento por exceção e outras técnicas podem ser consideradas de forma apropriada.

Além do mais, o PMI (2013, p. 93) determina que os relatórios de desempenho do trabalho são a representação física ou eletrônica das informações de desempenho do trabalho compiladas em documentos do projeto para suportar decisões, ações, ou criar conscientização.

As informações do projeto podem ser comunicadas verbalmente, de pessoa para pessoa. No entanto, a fim de registrar, armazenar e, às vezes, distribuir as informações sobre o desempenho do trabalho é necessária uma representação física ou eletrônica na forma de documentos de projeto. Os relatórios de desempenho do trabalho são um subconjunto de documentos do projeto que visam conscientizar e gerar decisões ou ações. Métricas específicas de desempenho do trabalho podem ser definidas no início do projeto e incluídas nos relatórios normais de desempenho do trabalho fornecidos às principais partes interessadas.

Exemplos de relatórios de desempenho do trabalho incluem relatórios de status, memorandos, justificativas, notas informativas, recomendações e atualizações. E desta maneira, o Apêndice C, traz assim como o descritivo do PMI (2013) as notas de desempenho referente ao projeto EzMart, sendo o orientador deste relatório o professor Afonso Celso Soares.

#### 4.2.3.2 Controle de Configuração

A gestão da configuração consiste em um processo importante em todas as fases do projeto, pois com o controle de todas as configurações, documentos, ou código do projeto, é possível que seja mais bem organizado e estruturado todos os artefatos que são produzidos. O PMI (2013, p. 96), completa essa informação, afirmando que o “controle da configuração foca as especificações das entregas e dos processos, enquanto o controle de mudanças foca a identificação, documentação e aprovação ou rejeição das mudanças nos documentos, nas entregas ou linhas de base do projeto”.

O controle da configuração contempla algumas atividades incluídas no processo de “Realizar o controle integrado de mudanças”. De acordo com o PMI (2013, p. 96), são descritas:

1. identificação da configuração: a identificação e seleção de um item de configuração para fornecer a base pela qual a configuração do produto é definida e verificada, produtos e documentos são rotulados, mudanças são gerenciadas e a responsabilidade é mantida;
2. registro da situação da configuração: informações são registradas e reportadas indicando quando os dados apropriados a respeito do item de configuração devem ser fornecidos. Essas informações incluem uma lista de identificação de configurações aprovadas, andamento das propostas de mudanças na configuração e andamento da execução das mudanças aprovadas;
3. verificação e auditoria da configuração: a verificação e auditorias da configuração garantem que a composição dos itens de configuração de um projeto está correta e que as mudanças correspondentes foram registradas, avaliadas, aprovadas, acompanhadas e corretamente efetuadas. Isso assegura que os requisitos funcionais definidos na documentação da configuração foram atendidos.

De acordo com o cronograma e a divisão das atividades, dispostas em cada incremento, cada integrante da equipe EzMart possui um papel dentro do desenvolvimento do projeto, e exatamente por isso é necessário que todos os artefatos produzidos por cada um sejam mantidos em um repositório, a fim de facilitar a consulta, a alteração, o estudo ou auxiliar no desenvolvimento de outros artefatos.

Desta maneira, o repositório do *Google* auxilia no controle da configuração em relação aos documentos, ou outros artefatos como diagramas, documentos de apoio, etc. Este repositório controla e cria versões de *backup* desses artefatos. Todos os integrantes possuem permissão para modificações, e essas modificações devem ser mantidas e descritas no repositório. Por exemplo, caso um ou mais itens deste documento do PFC seja modificado, é necessário que o integrante responsável pela mudança, altere e salve as modificações seguindo o formato: dia/mês/ano/Nome do projeto/Descrição da mudança, lembrando que a descrição da mudança deve ser mais breve o possível. Vale ressaltar que há quatro pastas que controlam as fases do projeto, onde essas versões devem ser dispostas.

Já para o controle de todo o código do projeto EzMart, é utilizado o serviço do repositório *SubVersion* (SVN). O controle de versões é a arte de gerenciar todas e quaisquer mudanças no código, e para os programadores é um paradigma obrigatório e essencial, conhecido também por ser uma boa prática no momento da codificação de todo e qualquer projeto. O SVN é essa ferramenta de controle de versões referente ao código do EzMart, que permite, além do desenvolvimento colaborativo a partir de um repositório único, a unificação do conteúdo, armazenamento de *logs,* e a geração de estatísticas diversas. Desta maneira, é necessário que todos os integrantes compreendam do que se trata esta ferramenta, e a utilize em seu favor. Para um melhor conhecimento, que pode ser documentado, segue algumas considerações. São elas:

1. repositório: é o local aonde estão contidos todos os arquivos do projeto EzMart, e é armazenado no Banco de Dados do SVN;
2. *working copy*: consiste na cópia de trabalho local na qual o desenvolvedor atua e é criada sempre que é feito o *checkout* do projeto EzMart;
3. *checkout*: consiste no ato de fazer o *download* do projeto para a máquina local, de modo que seus arquivos estejam vinculados ao SVN e passíveis de manipulação. O projeto para qual será feito o *checkout* deve existir no repositório;
4. *import*: consiste no ato de enviar os arquivos de um novo projeto para o repositório, e após o *import* obrigatoriamente deve ser realizado um *checkout* para que a *working* *copy* seja vinculada ao projeto;
5. *export*: consiste no ato de obtenção de um projeto do projeto sem vinculação ao SVN;
6. *commit*: consiste no ato de envio das modificações realizadas localmente para o servidor SVN;
7. *update*: consiste no ato de obtenção das atualizações presentes do servidor SVN, atualizando a cópia local;
8. *revision*: é o número que identifica cada uma das alterações ou o conjunto delas que são realizadas em um repositório;
9. *head*: consiste na revisão mais recente do repositório;
10. diretórios especiais:
11. *trunk*: armazena a versão funcional mais recente do desenvolvimento;
12. *branches*: armazena versões de desenvolvimento paralelas;
13. *tags*: armazena etiquetas para facilitar a localização das revisões.
14. *merge*: consiste na mesclagem de revisões entre os diretórios especiais;
15. *switch*: consiste na alteração do repositório utilizado por uma *working* *copy*. É realizada uma atualização ou mesclagem dos arquivos para assegurar que a *working* *copy* contenha exatamente o mesmo conteúdo do repositório somado com as alterações locais;
16. *relocate*: consiste na realocação do endereço de um repositório.

Todas as informações descritas foram compreendidas e embasadas na documentação SVN. Com isso, o integrante responsável pelo controle da configuração, Marcos Paulo Moreno Pereira, é também o responsável pelo controle do repositório SVN. Em consenso com a equipe, foram definidas boas práticas de versionamento, assim como para a documentação. São elas:

1. todas as revisões devem ser comentadas para facilitar o entendimento de cada alteração que foi realizada;
2. o código sempre deve estar pronto para ser compilado e colocado em produção se necessário;
3. os *commits* não devem causar a “quebra” das *builds*;
4. as alterações no código-fonte devem ser submetidas ao repositório sempre que forem concluídas;
5. devem ser divididas as implementações em pequenos pacotes compiláveis e funcionais;
6. o projeto no repositório deve conter quaisquer componentes e ferramentas necessárias para o funcionamento da aplicação na máquina do desenvolvedor;
7. todo e qualquer *backup* de versões deve ser mantido no repositório, obrigatoriamente com uma *tag*.

Com isso, a ferramenta de apoio utilizada no SVN é a TortoiseSVN que consiste na alternativa técnica escolhida dentre as demais, que supre a equipe em relação à interface gráfica para o ambiente W*indows.*

As Figuras 26 e 27 demonstram o exemplo da ferramenta TortoiseSVN aplicada no projeto EzMart, a fim de garantir a consistência do código.

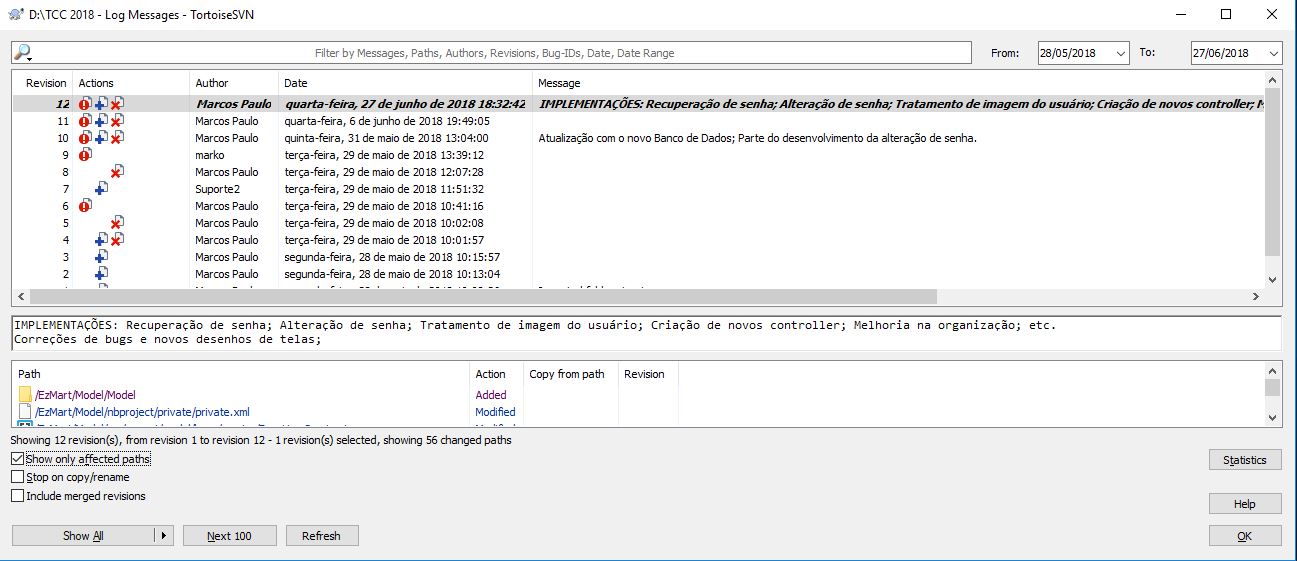


Figura 26 - Gerenciador de versões do EzMart

Fonte: adaptada de TortoiseSNV (2018)

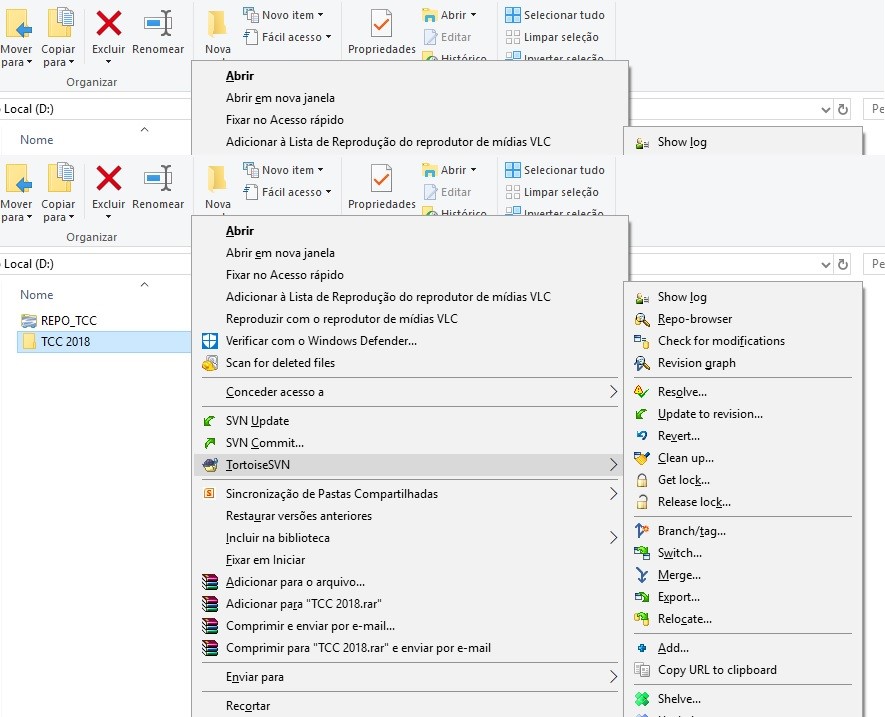


Figura 27 - Opções que o TortoiseSVN oferece como recursos para o gerenciamento de versões

Fonte: adaptada de TortoiseSNV (2018)

Neste projeto também é utilizado o *Google Drive* como segundo repositório, de forma a aumentar a disponibilidade de documentos e da aplicação entre os integrantes da equipe. Não há um único responsável pela administração deste repositório, entretanto, a integrante nomeada com o papel de Gerente de Projeto, Jéssica Souza Pivoto, é a responsável por treinar os outros integrantes às boas práticas para organização de todo conteúdo que é produzido. Exemplos de boa prática consistem na nomeação correta do artefato, como quem produziu, qual a data, à qual entrega se refere e a pasta criada para tal organização e efetuar a desconexão da conta sempre que não estiver sendo utilizada.

Desta maneira, o *Google Drive*, na forma usual, mantém as versões anteriores dos documentos durante 30 dias, porém há uma opção que reconfigura esta opção, para que o *Google Drive* continue mantendo versões anteriores, independentemente do tempo decorrido.

Entretanto, ao final da segunda fase, um dos integrantes da equipe EzMart solicitou uma mudança que foi aprovada pela gerente de projeto Jéssica Souza Pivoto, frente às maiores necessidades de um repositório que não dependesse exclusivamente de um computador como servidor. Esta justificativa foi aceita e compreendida não só pela gerente de projetos, como em unanimidade da equipe, pois apenas um computador era utilizado como servidor do sistema EzMart, o que, por sua vez, é desvantajoso. Para solucionar este problema, a equipe tornou-se adepta do GITHub.

O GITHub e o SNV que foi tratado anteriormente ainda nessa seção, possuem os mesmos objetivos, que é fornecer a equipe de um projeto, um sistema de controle de versões de modo com que todos os envolvidos na construção dos artefatos administrem, controlem e gerenciem da melhor forma o que é produzido. No entanto, o GITHub fornece a vantagem de redução de custos com o servidor, pois o repositório central funciona como repositório oficial, ao invés de ter uma função de processador central dos pedidos. Assim, a carga de processamento é distribuída pelas máquinas dos programadores e não se centraliza em apenas um computador.

A Figura 28 ilustra o repositório do GITHub utilizado por todos os integrantes do projeto EzMArt.

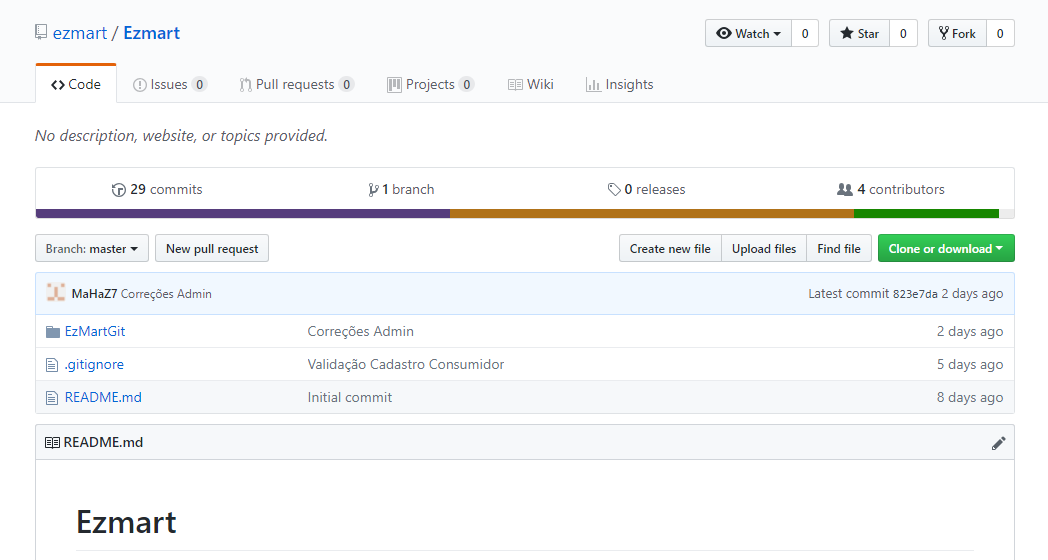


Figura 28- Repositório GIT do projeto EzMart

Fonte: Adaptada de GITHub (2018)

Em detalhamento, assim como o SNV, o GITHub possui os mesmos conceitos de *branches commits, chechout,* etc,no entanto, há alguns conceitos que diferenciais entre eles.

Para começar a utilizá-lo, primeiramente é necessário realizar a configuração inicial, como nome do usuário e seu *e-mail.*

1. $ git config --global user.name "nome"
2. $ git config --global user.email [nome@dominio.pt](mailto:nome@dominio.pt)

Estas configurações são de extrema importância, uma vez que para cada usuário que realiza quaisquer modificações, o GITHub fornece uma série de informações a respeito deste usuário, como hora de realização da mudança, mensagem de *commit,* dentre outras.

Em seguida, o usuário que tem acesso ao repositório pode obter a versão atualizada do *software* ou documento, ao fazer o clone do projeto ou documento em um repositório local através:

1. $ git clone <https://nomeutilizador@bitbucket.org/grownunder/teste-blog.git>

Entretanto, o endereço https fornecido como exemplo, deve ser substituído pelo endereço correto do repositório referência de onde o projeto está sendo baixado, como mostra a Figura 29.

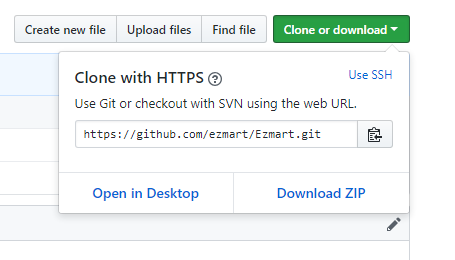


Figura 29- Clone ou download do repositório do projeto EzMart

Fonte: Adaptada de GITHub (2018)

Caso o usuário não tenha um repositório local e deseje iniciá-lo criando um projeto, é necessário:

1. $cd <pasta-do-projeto>
2. $ git init

Após essas configurações, o usuário pode adicionar os arquivos que foram criados ou modificados através do comando:

1. $ git add .

Quando um ou mais arquivos são adicionados, pela sequência lógica do GIT, ele está pronto para receber o comando de *commit.* Isso pode ser feito através do comando:

1. $ git commit -m “mensagem de commit”

Outra funcionalidade interessante e importante para os usuários do GITHub é a criação, listagem e *checkout* nos *branches* que foram criados pelo administrador ou quaisquer usuários do projeto, para melhor organizar as versões do que é produzido como artefato. Isso é mostrado respectivamente nos itens a, b e c.

1. $ git checkout -b <nome-novo-branch>
2. $ git branch -a (lista todos)
3. $ git checkout <nome do branch destino>

Por fim, após adicionadas todas as mudanças ou criações no repositório local da máquina do usuário, é necessário que o mesmo as organize no repositório do GITHub, fornecendo assim, um local seguro, confidencial e integro para tudo que é produzido como artefato no projeto. Para isso, o usuário deve fazer assim como mostra o comando abaixo:

1. git push origin <nome-branch>

Existem inúmeras outras funcionalidades, no entanto, as essenciais que são necessárias para os usuários e que foram utilizadas neste projeto foram demonstradas acima.

Em resumo, todos os repositórios com controle de versões que foram utilizados neste projeto, contribuíram significativamente para aumentar a experiência dos integrantes da equipe, pois sabe-se que o mercado de trabalho preza pelas experiências e valoriza os profissionais que sabem das diferenças, vantagens e desvantagens de quaisquer alternativas técnicas que são mencionadas.

#### 4.2.3.3 Controle de Mudanças

O controle de mudanças consiste no processo de revisar todas as solicitações de mudança referente ao escopo principal do projeto, assim como aprovar as mudanças e gerenciá-las. Segundo o PMI (2013, p. 93):

Realizar o controle integrado de mudanças é o processo de revisar todas as solicitações de mudança, aprovar as mudanças e gerenciar as mudanças sendo feitas nas entregas, ativos de processos organizacionais, documentos do projeto e no plano de gerenciamento do projeto, e comunicar a disposição dos mesmos. Ele revisa todas as solicitações de mudança ou modificações nos documentos do projeto, entregas, linhas de base ou no plano de gerenciamento do projeto, e aprova ou rejeita as mudanças.

O controle de mudança fornece inúmeros benefícios, e dentre eles, o PMI (2013, p.93) destaca que o principal benefício deste processo é “ permitir que as mudanças documentadas no âmbito do projeto sejam consideradas de forma integrada, reduzindo os riscos do projeto que frequentemente resultam das mudanças feitas sem levar em consideração os objetivos ou planos gerais do projeto. ” As entradas, ferramentas e técnicas, e saídas deste processo estão ilustradas na Figura 30.

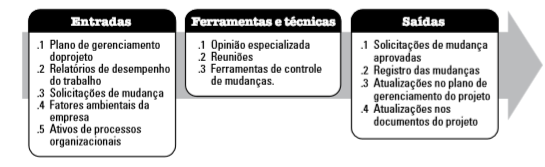


Figura 30 - Realizar o controle integrado de mudanças: entradas ferramentas e saída

Fonte: PMI (2013, p.93)

O processo de controlar as mudanças deve ser conduzido do início do fim do projeto, e não em uma etapa específica do processo, e é de responsabilidade final do gerente de projeto, que no caso o projeto EzMart é Jéssica Souza Pivoto. É através do gerenciamento das mudanças que se garante integridade das informações e artefatos do projeto. De acordo com PMI (2013, p. 95):

O plano de gerenciamento do projeto, a especificação do escopo do projeto e outras entregas são mantidas através do gerenciamento cuidadoso e contínuo das mudanças, através da rejeição ou aprovação das mesmas, assegurando assim que somente as mudanças aprovadas sejam incorporadas à linha de base revisada.

Por sua vez, todas as mudanças podem ser solicitadas por qualquer uma das partes interessadas no projeto, que são tratadas na seção 4.1.1. PMI (2013, p. 95) afirma que as mudanças podem ser iniciadas verbalmente entre as partes do projeto, porém ao final, devem ser sempre escritas e lançadas no sistema de gerenciamento de mudanças e/ou no gerenciamento de configurações. “As solicitações de mudança estão condicionadas ao processo especificado nos sistemas de controle de mudanças e de configuração. Estes processos de solicitação de mudança podem requerer informações sobre impactos estimados no tempo e custos”.

Entretanto, após a iniciação verbal das mudanças e suas devidas documentações, é necessário que as mesmas mudanças sejam aprovadas ou rejeitadas por uma pessoa responsável e segundo o PMI (2013, p. 95), essa pessoa pode ser um patrocinador ou gerente do projeto. No caso do projeto EzMart, a responsável por aprovar ou rejeitar as mudanças solicitadas é Jéssica Souza Pivoto. No entanto, inclui na responsabilidade do gerente de projetos, revisar as estimativas feitas no cronograma, como o sequenciamento das atividades ou duração das mesmas. Isso também é afirmado no PMI (2013, p.95):

Solicitações de mudança aprovadas podem requerer novas ou revisadas estimativas de custos, sequenciamento de atividades, datas de cronograma, requisitos de recursos e análise de alternativas de resposta aos riscos. Essas mudanças podem exigir ajustes no plano de gerenciamento, ou em outros documentos do projeto. O nível de controle de mudança aplicado depende da área de aplicação, complexidade do projeto em questão, requisitos contratuais e o contexto e ambiente no qual o projeto é executado. Pode ser necessária a aprovação do cliente ou do patrocinador para certas requisições de mudança após a aprovação pelo CCM (comitê de controle de mudanças), a menos que eles participem do CCM.

No caso do projeto EzMart, o comitê de controle de mudanças é composto pelos cinco membros do projeto, sob responsabilidade da gerente Jéssica Souza Pivoto. O comitê fornece o espaço necessário para discussão da mudança que foi feita por uma das partes interessadas do projeto.

Vale ressaltar que toda e qualquer mudança requerida por um dos membros da equipe é realizada através de um *e-mail*. Desta maneira, há um controle interno da equipe EzMart. No entanto, há mudanças que são solicitadas através do documento do PFC em cada fase de desenvolvimento. Neste caso, as partes interessadas especificadas como orientadores (seção 4.1.1) anotam informalmente em cada seção do documento que deve ser entregue na fase correspondente, as solicitações de mudança. Vale ressaltar, portanto, que todos os documentos são arquivados numa pasta com a *tag* “Número da fase” para possíveis consultas nas solicitações de mudança.

Neste caso, para cada solicitação de mudança é necessária a aprovação do comitê, já que as solicitações não são oriundas da equipe interna, outra parte interessada do projeto EzMart. O comitê é responsável por avaliar o pedido de mudança, analisar o impacto que a mudança trará para o projeto, analisar em qual etapa do projeto haverá um impacto em maior ou menor escala e por fim, qual a demanda total de tempo a mudança deve requerer. Esses são fatores primordiais para aprovação ou rejeição da mudança. Por outro lado, em comparação, caso a mudança seja sugerida por membros internos da equipe, o responsável por avaliar é somente o gerente do projeto.

As solicitações de mudanças são documentadas sempre que há necessidade e encontram-se relatadas no Apêndice Z.

### 4.2.4 Gestão da Qualidade

Todos os projetos de *software* devem atender de forma incontestável todos os requisitos do sistema, juntamente com as necessidades dos clientes. Cabe à gestão da qualidade garantir que um projeto seja executado do início ao fim obedecendo todas as normas para que a qualidade seja assegurada.

Heldman (2009) afirma que a qualidade consiste na principal preocupação em relação a todos os projetos de *software,* é através dela é possível determinar se as expectativas das partes interessadas foram atendidas.

De acordo com o PMI (2013, p. 227), “O gerenciamento da qualidade do projeto trabalha para garantir que os requisitos do projeto, incluindo os requisitos do produto sejam cumpridos e validados”. Em conjunto, segundo *Project Builder* (2016), isto acontece porque a gestão de qualidade:

[...] se encarrega de garantir que todos os requisitos serão devidamente cumpridos e, consequentemente, que o cliente final ficará satisfeito. Por meio da gestão da qualidade em projetos é possível monitorar não só as ações desenvolvidas pela equipe como também o nível de excelência com que são realizadas.

Desta maneira, uma visão geral dos processos de gerenciamento da qualidade é fornecida pela Figura 31.

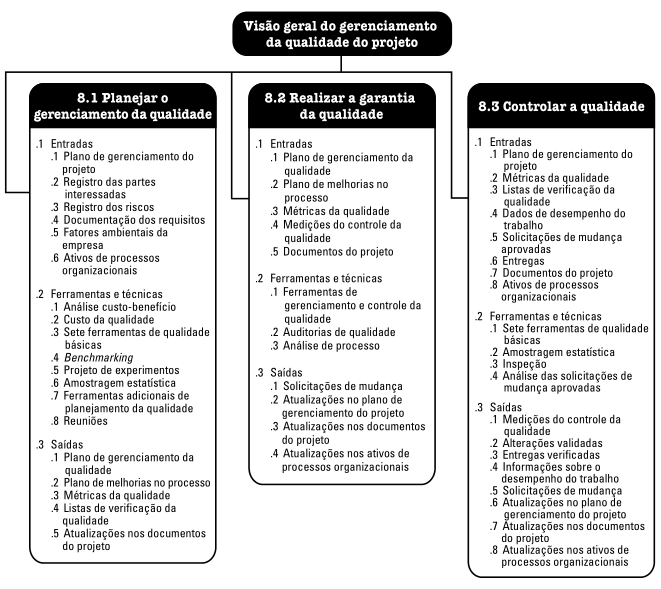


Figura 31 - Visão geral do gerenciamento da qualidade do projeto

**Fonte: adaptada do PMI (2013)**

De posse de todos os conhecimentos adquiridos sobre escopo, normas, e diretrizes referente ao projeto EzMart, se torna possível planejar, monitorar e garantir que os requisitos do *software* sejam atendidos e que os processos aplicados sejam os mais adequados possíveis.

Lima (2009) afirma que a gestão da qualidade em projetos pode possuir dois focos, sendo o primeiro a qualidade do produto do projeto, que, por sua vez, refere-se aos requisitos de sistema, técnico ou de desempenho, e o segundo é a qualidade do projeto, que é relacionada à qualidade no trabalho necessário para entrega do projeto.

Desta maneira, para que a qualidade do projeto EzMart seja assegurada, é necessário que a equipe de construção do *software* seja dividida em papéis distintos e que são responsabilizados por determinada área do conhecimento, devidamente com cada processo ao qual pertence. E, portanto, para que haja responsabilidades referentes a cada papel, a equipe EzMart foi dividida em três papéis principais. São eles:

1. gerente do projeto;
2. responsável técnico;
3. responsável pela pesquisa científica e documentação.

Assim divididos, cada papel se responsabiliza por determinadas atividades, ações e tarefas. O papel de gerente de projeto é desempenhado por Jéssica Souza Pivoto. Este profissional, durante a execução de um projeto, define papéis, atribui tarefas, acompanha e documenta o andamento da sua equipe através de ferramentas e técnicas apuradas, administra investimentos e integra as pessoas para trabalharem juntas por um só objetivo. Também é função deste papel, monitorar possíveis riscos e estar sempre preparado para mudar de estratégia rapidamente, se necessário. Um gerente de projetos deve ter organização, agilidade na tomada de decisões e visão para calcular riscos. Deve ser inspirador para sua equipe, saber planejar minuciosamente cada passo, ser flexível para mudar de estratégia e rápido para reorganizá-la. Bons gerentes são capazes de inspirar e motivar sua equipe além do esperado. Eles sabem como gerenciar pessoas e incentivá-las a usar o que têm de melhor em nome do projeto. Essas habilidades são conhecidas como competências interpessoais. Segundo Robbins (2010), as habilidades humanas são as que proporcionam a capacidade de trabalhar com outras pessoas, compreendendo-as, motivando-as, tanto individualmente quanto em grupo. Muitas dessas habilidades são provenientes dos relacionamentos interpessoais.

Outro papel importante é o papel de responsável técnico do projeto, e desta forma, Marcos Paulo Moreno Pereira, é incumbido da tarefa de pesquisar e selecionar as melhores tecnologias para atender os requisitos do *software,* atuando na manipulação e controle dessas tecnologias. Marcos Henrique Azevedo é responsável pela manipulação dos dados no Banco de Dados, atuando como responsável técnico juntamente com Marcos Paulo Moreno.

Já os responsáveis técnicos pela pesquisa científica são Jéssica Souza Pivoto, além da tarefa descrita como gerente de projeto, e Camila Maria da Costa. A responsabilidade de pesquisa científica tem como objetivo atribuir conhecimento teórico a fim de que o sistema EzMart situe-se no mercado, além de conhecer seus concorrentes e assim estabelecer estratégias para sobressair no ramo da inovação. Diego Vinicius Cesar do Amaral é o responsável pela parte da documentação que tem como objetivo estruturar e analisar de forma comportamental e de dados, todos os requisitos do sistema. O papel de pesquisa e documentação engloba a adequação de aos padrões científicos e as diretivas de documentação da FAI.

EzMart estabelece procedimentos que asseguram na qualidade do software. São eles:

1. utilizar as normas da Associação Brasileira de Normas Técnicas (ABNT) e diretrizes para elaboração de trabalhos científicos da FAI;
2. embasar o projeto utilizando trabalhos relacionados, artigos e fontes de referência confiáveis segundo pesquisas de embasamento científico;
3. verificar se as citações, figuras, tabelas e quadros são referenciados de forma correta em suas respectivas listas de referência no início do documento;
4. elaborar o conteúdo de cada Capítulo dentro do prazo estabelecido na seção “Modelo de ciclo de vida”;
5. revisar e corrigir o documento antes da entrega;
6. definir um código para controle de versões do documento;
7. desenvolver a documentação de análise do projeto antes do desenvolvimento do sistema;
8. estabelecer e monitorar a realização de testes em vários níveis do desenvolvimento;
9. utilizar o padrão UML para modelar os artefatos;
10. adotar e respeitar as convenções de codificação definidas pela *Java Code Conventions*;
11. implementar e seguir adequadamente os requisitos funcionais e não funcionais;
12. utilizar a ferramenta automática *Subversion* para o controle de versões do código fonte;
13. acompanhar o andamento do projeto e seus riscos com reuniões periódicas.

Os critérios para atingir a qualidade do projeto, na forma de lista de verificações, encontram-se no Apêndice I de acordo com as fases do projeto.

Portanto, a equipe de gerenciamento do projeto deve determinar níveis adequados de exatidão e precisão para o uso no plano de gerenciamento da qualidade. A abordagem básica do gerenciamento qualidade descrita nesta seção pretende ser compatível com os padrões de qualidade da Organização Internacional para Padronização (ISO). Todos os projetos devem ter um plano de gerenciamento da qualidade, inclusive o EzMArt. A equipe deste projeto deve seguir o plano de gerenciamento da qualidade e dispor de dados que comprovem a conformidade com o mesmo.

### 4.2.5 Gestão dos Riscos

O objetivo de um gerenciamento proativo de riscos é aumentar a probabilidade e o impacto dos eventos positivos (oportunidades) e reduzir a probabilidade e o impacto dos eventos negativos (ameaças) do projeto.

Os riscos do projeto podem influenciar o objetivo almejado, negativamente ou positivamente, sob a forma de ameaças ou oportunidades, por isso o gerenciamento dos riscos é essencial em qualquer projeto para aumentar as taxas de sucesso, porque não adianta somente identificá-los e sim gerenciá-los.

Segundo o PMI (2013, p. 309):

O Gerenciamento dos riscos do projeto inclui os processos de planejamento, identificação, análise, planejamento de respostas e controle de riscos de um projeto. Os objetivos do gerenciamento dos riscos do projeto são aumentar a probabilidade e o impacto dos eventos positivos e reduzir a probabilidade e o impacto dos eventos negativos no projeto.

O risco de um projeto é um evento ou uma condição incerta que, caso ocorra, afetará positivamente ou negativamente o objetivo do projeto, como escopo, cronograma, custo e qualidade. Segundo PMI (2013, p. 310), um risco pode ter “uma ou mais causas e, se ocorrer, pode ter um ou mais impactos. Uma causa pode ser um requisito, premissa, restrição ou condição potencial que crie a possibilidade de resultados negativos ou positivos”. A Figura 32 ilustra a visão geral do gerenciamento dos riscos de um projeto.

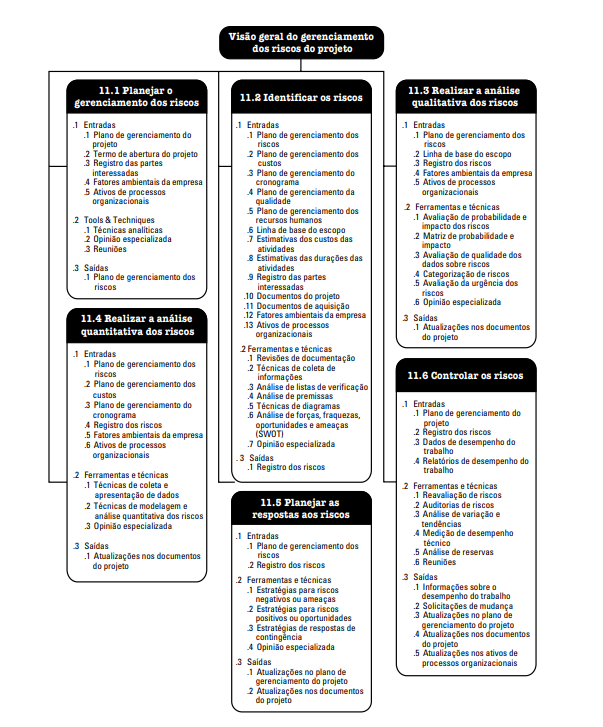


Figura 32- Visão geral do gerenciamento dos riscos do projeto

**FONTE: adaptada de PMI (2013)**

O PMI (2013, p. 309) cita seis processos para o gerenciamento de riscos, que são:

1. planejar o gerenciamento dos riscos - o processo de definição de como conduzir as atividades de gerenciamento dos riscos de um projeto;
2. identificar os riscos - o processo de determinação dos riscos que podem afetar o projeto e de documentação das suas características;
3. realizar a análise qualitativa dos riscos - o processo de priorização de riscos para análise ou ação posterior através da avaliação e combinação de sua probabilidade de ocorrência e impacto;
4. realizar a análise quantitativa dos riscos - o processo de analisar numericamente o efeito dos riscos identificados nos objetivos gerais do projeto;
5. planejar as respostas aos riscos - o processo de desenvolvimento de opções e ações para aumentar as oportunidades e reduzir as ameaças aos objetivos do projeto;
6. controlar os riscos - o processo de implementar planos de respostas aos riscos, acompanhar os riscos identificados, monitorar riscos residuais, identificar novos riscos e avaliar a eficácia do processo de gerenciamento dos riscos durante todo o projeto.

Ainda segundo o PMI (2013), existem técnicas para o tratamento de riscos, sejam eles de ameaça ou oportunidade. A seguir é apresentado as 4 técnicas para o tratamento de riscos ameaçadores, que são:

1. prevenir – esta técnica condiz em eliminar a ameaça ou proteger o projeto contra o seu impacto. Envolve a alteração do plano de gerenciamento do projeto para eliminar totalmente a ameaça. Um exemplo de aplicação desta técnica é a retirada de um requisito do escopo do projeto, que estava associado a um risco;
2. transferir - a transferência de riscos é uma estratégia de resposta ao risco em que a equipe do projeto transfere o impacto de uma ameaça para terceiros, juntamente com a responsabilidade pela sua resposta. Transferir o risco simplesmente passa a responsabilidade de gerenciamento para outra parte, mas não o elimina. Por exemplo, quando um comprador tem recursos que o vendedor não possui, pode ser prudente transferir uma parte do trabalho e o risco correspondente de volta ao comprador por meio de um contrato. Em muitos casos, o uso de um contrato de custo mais remuneração pode transferir o risco do custo para o comprador, enquanto um contrato de preço fixo pode transferir o risco para o vendedor;
3. mitigar - mitigação de riscos é uma estratégia de resposta ao risco em que a equipe do projeto age para reduzir a probabilidade de ocorrência ou o impacto do risco. Ela implica na redução da probabilidade e/ou do impacto de um evento de risco adverso para dentro de limites aceitáveis. Adotar uma ação antecipada para reduzir a probabilidade e/ou o impacto de um risco ocorrer no projeto em geral é mais eficaz do que tentar reparar o dano depois de o risco ter ocorrido. Um exemplo de aplicação desta técnica é a realização de *backups* diários. Isto implica em reduzir o impacto do risco, mas não muda a probabilidade de ocorrer a perda de dados;
4. aceitar - a aceitação de risco é uma estratégia de resposta pela qual a equipe do projeto decide reconhecer a existência do risco e não agir, a menos que o risco ocorra. Essa estratégia é adotada quando não é possível ou econômico abordar um risco específico de qualquer outra forma. Essa estratégia indica que a equipe do projeto decidiu não alterar o plano de gerenciamento do projeto para lidar com um risco, ou não conseguiu identificar outra estratégia de resposta adequada. Cabe destacar que esta técnica é utilizada tanto para os riscos de ameaça quanto para os riscos de oportunidade e deve-se obrigatoriamente haver um plano de contingencia.

Portanto, o PMI (2013, p.313) afirma que planejar o gerenciamento dos riscos é o processo de definição de como conduzir as atividades de gerenciamento dos riscos de um projeto e cita o benefício do gerenciamento de riscos que “garante que o grau, tipo, e visibilidade do gerenciamento dos riscos sejam proporcionais tanto aos riscos quanto à importância do projeto para a organização”.

De acordo com o PMI (2013) o plano de gerenciamento dos riscos é muito importante na comunicação, obtenção de acordo e apoio das partes interessadas para garantir que o processo de gerenciamento dos riscos seja apoiado e executado de maneira efetiva. A Figura 33 ilustra as entradas, ferramentas, técnicas e saída do processo de gerenciamento dos riscos de um projeto. A Figura 34 por sua vez, ilustra o diagrama de fluxo de dados do processo. Mediante a essas informações, nota-se a importância de conhecê-las e aprofundá-las para obter um conhecimento adequado sobre o gerenciamento de riscos do projeto EzMart, para que a partir disso a planilha de riscos seja elaborada, e para que todos os riscos nela identificados possam ser controlados e monitorados ao longo da vida do projeto. O planejamento cuidadoso dos riscos do projeto aumenta a probabilidade de êxito dos outros processos de gerenciamento dos riscos. PMI (2013, p.313) afirma que:

O planejamento também é importante para fornecer recursos e tempo suficientes para as atividades de gerenciamento dos riscos, e para estabelecer uma base acordada para a avaliação dos riscos. O processo de planejar o gerenciamento dos riscos deve começar quando o projeto é concebido, e ser concluído na fase inicial do planejamento do projeto.

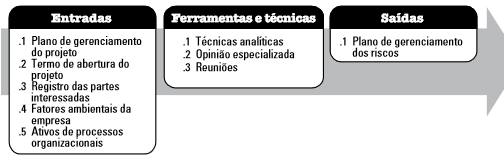


Figura 33 - Planejar o gerenciamento dos riscos: entradas, ferramentas e técnicas, e saídas

Fonte: PMI (2013, p.313)

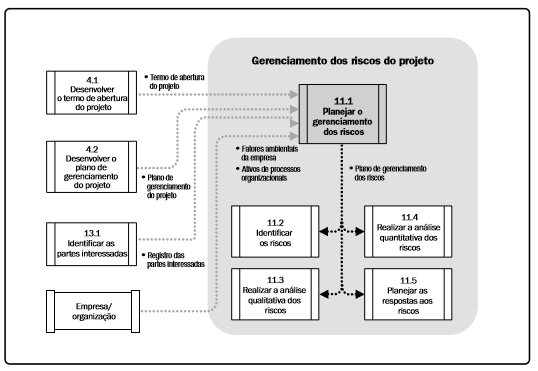


Figura 34 - Diagrama do fluxo dos dados do processo “Planejar o gerenciamento dos riscos”

Fonte: PMI (2013, p.313)

Portanto, os riscos de ameaça para o projeto EzMart encontram-se em uma planilha disposta no APÊNDICE D de acordo com as fases do projeto.

# 5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS

A especificação dos requisitos de um sistema é evidentemente importante, pois tem como objetivo obter conhecimento sobre o domínio do *software*, por meio do levantamento das necessidades das partes interessadas.

Para Pressman (2010, p.116):

A engenharia de requisitos ajuda os engenheiros de *software* a compreender melhor com o que eles vão trabalhar para resolver. Ela inclui o conjunto de tarefas que levam a um entendimento de qual será o impacto do *software* sobre o negócio, do que o cliente quer e de como os usuários finais vão interagir com o *software.*

Em complemento, O PMI (2013, p. 117) afirma que os requisitos devem estar devidamente documentados, para que descrever como os requisitos individuais atendem às necessidades do negócio para o projeto.

Os requisitos podem começar em um alto nível e tornarem-se progressivamente mais detalhados conforme mais informações sobre estes são conhecidas. Antes das linhas de base serem estabelecidas, os requisitos devem ser não ambíguos (mensuráveis e passíveis de testes), rastreáveis, completos, consistentes e aceitáveis para as principais partes interessadas. O formato de um documento de requisitos pode variar de uma simples lista categorizada por partes interessadas e prioridades a formas mais elaboradas contendo um resumo executivo, descrições detalhadas e anexos.

Portanto, este Capítulo apresenta a especificação e a análise dos requisitos do sistema EzMart. Essa atividade é fundamental no desenvolvimento do sistema, por permitir que sejam moldadas as especificações de acordo com a necessidade dos envolvidos e por permitir conduzir a um melhor conhecimento sobre a gestão de mudanças, evolução dos requisitos e seus impactos.

## 5.1 REQUISITOS DO SISTEMA DE SOFTWARE

De acordo com Pressman (2016, p.133):

A engenharia de requisitos constrói uma ponte entre o projeto e a construção, mas onde começa essa ponte? Alguém pode argumentar que ela tem sua base nos envolvidos do projeto (por exemplo, gerentes, clientes e usuários), em que é definida a necessidade do negócio, são descritos cenários de usuários, delineadas funções e recursos e identificadas restrições de projeto. Outros poderiam sugerir que ela se inicia com uma definição mais abrangente do sistema, em que o *software* é apenas um componente de domínio do sistema mais abrangente. Porém, independentemente do ponto de partida, a jornada pela ponte nos leva bem à frente no projeto, permitindo que examinemos o contexto do trabalho de *software* a ser realizado; as necessidades específicas a que o projeto e a construção devem atender; as prioridades que orientam a ordem na qual o trabalho deve ser concluído; e as informações, funções e comportamentos que terão um impacto profundo no projeto resultante.

Assim, nesta seção, encontram-se os requisitos de acordo com cada módulo do sistema EzMart. Os requisitos estão divididos em Requisitos Funcionais (RF) e Não Funcionais (RNF) e para cada um mostra-se a prioridade no projeto. Os níveis de prioridade podem ser:

1. essencial;
2. importante;
3. desejável.

Os requisitos classificados como essenciais são indispensáveis para o funcionamento da aplicação. Os requisitos importantes ajudam a complementar o funcionamento do sistema e os requisitos classificados como desejáveis são aqueles que podem agregar mais satisfação com o funcionamento do sistema, pois constituem funcionalidades adicionais para o sistema EzMart. A ordem de prioridade para desenvolvimento, portanto, deve ser essencial, importante e desejável.

### 5.1.1 Requisitos Funcionais

Mendes (2016, p. 3) define os requisitos funcionais como:

Um requisito de sistema de *software* que especifica uma função que o sistema ou componente deve ser capaz de realizar. Estes são requisitos que definem o comportamento do sistema, ou seja, o processo ou transformação que componentes de *software* ou *hardware* efetuam sobre as entradas para gerar as saídas. Esses requisitos capturam as funcionalidades sob o ponto de vista do usuário.

Os requisitos funcionais são subdivididos de acordo com as necessidades das partes interessadas no projeto. Antecedendo à descrição deles, faz-se necessária a abordagem da perspectiva de alguns interessados no presente projeto. São eles:

1. consumidor (Pessoa Física ou Jurídica);
2. estabelecimento de supermercado (Pessoa Jurídica).

Portanto, de acordo com as necessidades dessas partes em relação às funcionalidades, os requisitos podem ser agrupados em módulos. Os módulos são um agrupamento de determinadas funções de acordo com um propósito específico. Eles são mostrados a seguir.

1. Módulo de Acesso

Este módulo agrupa um conjunto de funcionalidades relacionadas aos consumidores e estabelecimentos.

**RF 01 – Realizar autenticação**

Este requisito refere-se à funcionalidade de acesso ao sistema a partir dos cadastros dos usuários devidamente realizados para consumidores, estabelecimentos e administradores do sistema.

Os usuários deverão informar:

1. *e-mail*;
2. senha.

Esses dados são necessários para permitir a autenticação e acesso ao sistema, que deve apresentar um perfil de acesso particular para cada usuário do sistema.

O sistema deve assegurar algumas condições. São elas:

1. rigidez no controle de autenticação do usuário: caso haja dados incorretos nos campos de autenticação, o sistema não deve permitir o acesso do usuário;
2. mensagens informativas que orientem o usuário em caso de falhas: caso haja falta de conexão com o servidor da aplicação ou os dados de entrada de autenticação estejam incorretos, é necessário que o sistema exiba uma alerta ao usuário, que indique o problema e a solução. Por exemplo: caso o dado “Senha” esteja incorreto, o sistema deve exibir a mensagem: “*E-mail* e/ou senha incorreto(s). Tente outra vez”. Caso haja falta de conexão com o servidor no momento da autenticação, o sistema deve exibir, por exemplo, a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o usuário se atente às mensagens para que solucione os erros e obtenha uma melhor experiência com o sistema.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 04 - Cadastrar consumidor;
2. RF 05 - Cadastrar estabelecimento.

**RF 02 – Recuperar senha**

Este requisito refere-se à funcionalidade de recuperação da senha dos respectivos perfis de acesso dos consumidores, estabelecimentos ou administradores do sistema, caso haja esquecimento ou perda da senha para autenticação do usuário.

Para que o requisito seja atendido, o sistema deve disponibilizar uma opção ao usuário, responsável por fornecer um fluxo para obtenção da nova senha. Por exemplo:

1. Esqueceu sua senha?

Ao selecionar a opção disponibilizada*,* o sistema deve fornecer um campo para o usuário confirmar seu *e-mail*.

Desta maneira, o usuário deve receber uma senha para o acesso ao sistema, enviada ao *e-mail* que foi informado, desde que este esteja correto. Caso o *e-mail* esteja incorreto, o sistema deve exibir uma mensagem informativa, por exemplo, “*E-mail* não encontrado” ou “*E-mail* incorreto”. Essas mensagens devem fornecer orientações para o usuário, a fim de que guiá-lo a melhores experiências.

O sistema deve assegurar que algumas condições sejam devidamente cumpridas. São elas:

1. a nova senha gerada deve conter no mínimo seis caracteres incluindo números e letras e excluindo caracteres especiais;
2. a confirmação de senha deve conter a mesma senha inserida no campo “Senha”;
3. mensagens informativas que orientem o usuário em caso de falhas: caso haja inconsistências no momento da entrada da nova senha. Por exemplo, caso a senha confirmada esteja diferente da primeira senha informada, o sistema deve exibir uma mensagem que indique o problema e a solução para o usuário. Neste caso, por exemplo: “As senhas não correspondem. Redigite-as”;
4. mensagens informativas que notifiquem o usuário em caso de operação bem-sucedida: caso haja sucesso ao receber o *e-mail* informado para enviar a senha temporária, o sistema deve exibir uma mensagem para o usuário. Por exemplo: “A senha temporária foi enviada para o *e-mail* informado”.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 04 - Cadastrar consumidor;
2. RF 05 - Cadastrar estabelecimento.

**RF 03 – Alterar senha**

Este requisito refere-se à funcionalidade de alteração da senha dos respectivos perfis de acesso dos consumidores, estabelecimentos ou administradores do sistema, caso o usuário deseje alterar apenas sua senha.

Para efetuar a alteração da senha, o usuário deve selecionar a opção referente à alteração da senha no respectivo perfil. Esta opção deve redirecionar o usuário para um formulário que contenha os seguintes campos para sua devida alteração:

1. senha atual;
2. nova senha;
3. confirmação da nova senha.

Com isso, o sistema deve garantir algumas condições. São elas:

1. a nova senha não deve ser igual à senha anterior: neste caso o sistema deve exibir uma mensagem informativa caso haja inconsistência no momento da criação da nova senha. Por exemplo: “A nova senha digitada está igual à senha atual. Modifique-a”.
2. a nova senha deve conter no mínimo seis caracteres incluindo números e letras e excluindo caracteres especiais;
3. a confirmação da senha deve conter a mesma senha inserida no campo “Senha”.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 06 - Editar perfil do consumidor;
3. RF 16 - Editar perfil do estabelecimento.

**RF 04 – Cadastrar consumidor**

Este requisito refere-se à funcionalidade de cadastro para consumidores, sendo pré-condição para atender aos outros requisitos funcionais. Para cadastro, o sistema deve exibir um formulário que contenha os seguintes campos:

1. nome;
2. sobrenome;
3. *e-mail*;
4. senha;
5. confirmar senha;
6. logradouro;
7. número;
8. bairro;
9. município;
10. Unidade da Federação (UF);
11. CEP;
12. CPF;
13. Telefone fixo e telefone celular.

O sistema deve assegurar algumas condições:

1. a rigidez no controle de cadastro de usuário: deve existir somente um cadastro para cada CPF e *e-mail*;
2. mensagens informativas que orientem o usuário em caso de falha: caso haja falta de conexão com o servidor da aplicação, ou os dados estejam incorretos ou não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique o problema e a solução Por exemplo: Caso o usuário digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Campo nome contém caracteres especiais. Remova para prosseguir”. Caso haja falta de conexão com o servidor, o sistema deve exibir a mensagem: “Falha de conexão com o servidor. Tente novamente”. Desta maneira, é fundamental que o consumidor siga regras para cadastro de uma nova conta de usuário, gerando assim, dados consistentes e íntegros;
3. mensagem informativa que notifique o sucesso ao cadastrar uma conta de um usuário consumidor. Por exemplo: “Usuário cadastrado com sucesso”;
4. o cadastro deve ser confirmado pelo usuário acessando seu *e-mail* e ativando um *link* de confirmação;
5. os campos “CPF”, “CEP” e “Telefone” devem conter formatação padrão. O CPF deve conter formatação padrão em: XXX.XXX.XXX-XX; o CEP deve ter a formatação padrão em: XX.XXX-XXX; e o telefone deve conter a formatação padrão em (XX) X XXXX-XXXX para celular ou (XX) XXXX-XXXX para telefone fixo.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s): nenhum.

**RF 05 – Cadastrar estabelecimento**

Este requisito refere-se à funcionalidade de cadastro de estabelecimentos comerciais. Para cadastro, o sistema deve exibir um formulário que contenha os seguintes campos:

1. nome fantasia;
2. razão social;
3. *e-mail*(s);
4. senha;
5. confirmar senha;
6. logradouro;
7. número;
8. bairro;
9. município;
10. UF;
11. CEP;
12. CNPJ;
13. Telefone fixo e telefone celular.

O sistema deve assegurar algumas condições:

1. a rigidez no controle de cadastro de usuário: deve haver somente um cadastro para cada CNPJ e *e-mail*;
2. mensagens informativas que orientem o usuário em caso de falha: caso haja falta de conexão com o servidor da aplicação, ou os dados estejam incorretos ou não sigam a formatação padrão, o sistema deve exibir uma alerta ao usuário, que indique o problema e a solução Por exemplo: caso o usuário digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “O nome contém caracteres especiais. Remova para prosseguir”. Caso haja falta de conexão com o servidor, o sistema deve exibir a mensagem: “Falha de conexão com o servidor da aplicação. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o estabelecimento siga regras para cadastro de uma nova conta de usuário, gerando assim, dados consistentes e íntegros;
3. mensagem informativa que notifique o sucesso ao cadastrar uma conta de um usuário estabelecimento. Por exemplo: “Usuário cadastrado com sucesso”;
4. o cadastro deve ser confirmado pelo usuário acessando seu *e-mail* e ativando o *link* de confirmação;
5. o campo “CNPJ”, “CEP” e “Telefone” devem conter formatação padrão. O CNPJ deve ter formatação padrão em: XX.XXX.XXX/YYYY-ZZ. O CEP deve conter a formatação padrão em: XX.XXX-XXX. O telefone deve conter formatação padrão em (XX) X XXXX-XXXX para celular ou (XX) XXXX-XXXX para telefone fixo.

Outro aspecto deste requisito é determinar a localização do estabelecimento por meio da indicação de suas coordenadas geográficas:

1. latitude;
2. longitude.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s): nenhum.

1. Módulo do consumidor

Este módulo agrupa um conjunto de funcionalidades relacionadas aos consumidores.

**RF 06 – Editar perfil do consumidor**

Esse requisito refere-se à funcionalidade de edição do perfil do consumidor. O formulário para atualização dos dados deve conter os seguintes dados:

1. nome;
2. sobrenome;
3. *e-mail*;
4. logradouro;
5. número;
6. bairro;
7. município;
8. UF;
9. CEP;
10. CPF;
11. Telefone fixo e telefone celular.

O sistema deve assegurar algumas condições, são elas:

1. a rigidez no controle de cadastro de usuário: deve existir somente um cadastro para cada CPF e *e-mail*;
2. mensagens informativas que orientem o usuário em caso de falha: caso haja falta de conexão com o servidor da aplicação, ou os dados estejam incorretos ou não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique o problema e a solução Por exemplo: caso o usuário digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Nome contém caracteres especiais. Remova para prosseguir”. Caso haja falta de conexão com o servidor, o sistema deve exibir a mensagem: “Falha de conexão com o servidor. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o consumidor siga regras para a alteração de sua conta de usuário, para que assim os dados se mantenham consistentes e íntegros;
3. mensagem informativa que notifique o sucesso ao alterar os dados cadastrais de uma conta de usuário consumidor. Por exemplo: “Usuário cadastrado com sucesso”;
4. os campos “CPF”, “CEP” e “Telefone” devem conter formatação padrão. O CPF deve ter formatação padrão em: XXX.XXX.XXX-XX; CEP deve ter a formatação padrão em: XX.XXX-XXX; e o telefone deve conter a formatação padrão em (XX) X XXXX-XXXX para celular ou (XX) XXXX-XXXX para telefone fixo.
5. não deve ser possível a modificação do CPF do usuário;

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 04 - Cadastrar consumidor.

**RF 07 – Avaliar estabelecimento**

Esse requisito refere-se à funcionalidade de avaliação do consumidor sobre o estabelecimento que frequenta.

O mecanismo de avaliação se torna necessário à medida que o estabelecimento utilizará do sistema como meio de obtenção de *feedback* dos seus consumidores. Com isso, o estabelecimento pode avaliar o grau de satisfação geral do público que o frequenta, assim como ter referência sobre os preços, atendimento, acessibilidade, etc..

Nesta avaliação é necessário que o consumidor avalie o estabelecimento de acordo com pontos-chave, são eles:

1. grau de satisfação;
2. preços dos produtos;
3. ambiente;
4. atendimento;
5. diversidade de produtos;
6. acessibilidade;
7. data de avaliação;
8. opinião.

Nessas condições, o usuário deve ter acesso a um campo selecionável com os itens descritos de “a” até “h”. Assim, o usuário pode selecionar o grau de satisfação em relação a todos os itens de avaliação. Somente no último item da avaliação, o sistema deve exibir um campo editável para que o consumidor descreva um problema específico que foi encontrado ou faça sugestões ou elogios ao estabelecimento. O campo para a opinião deve possuir um tamanho mínimo de 3 caracteres e máximo de 100 caracteres.

O sistema deve assegurar que para cada CPF cadastrado, haja apenas uma avaliação.

O sistema deve calcular a média de todas as avaliações e disponibilizá-la no perfil do estabelecimento.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 01- Realizar autenticação.

**RF 08 – Editar avaliação do estabelecimento**

Esse requisito refere-se à funcionalidade de alteração dos itens da avaliação realizada sobre o estabelecimento por parte do consumidor.

O formulário de edição da avaliação deve conter os seguintes campos para que o consumidor altere os itens da avaliação sobre o estabelecimento:

1. grau de satisfação;
2. preços dos produtos;
3. ambiente;
4. atendimento;
5. diversidade de produtos;
6. acessibilidade;
7. data de avaliação;
8. opinião.

O mecanismo de alteração dos itens da avaliação se torna necessário à medida que o consumidor pode alterar sua percepção em relação ao estabelecimento. Isso pode ser decorrente de outra visita ao estabelecimento que resultou em mudanças nos fatores que são abordados na pesquisa para avaliação do estabelecimento.

Da mesma maneira, é necessário que ao se alterar uma avaliação sobre o estabelecimento, o sistema exiba uma mensagem informativa que confirme a edição de uma avaliação por parte do consumidor. Por exemplo: “Avaliação alterada com sucesso”.

Após uma alteração na avaliação, o sistema deve calcular a média de todas as avaliações (inclusive das que foram alteradas) e disponibilizá-la no perfil do estabelecimento.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 07 - Avaliar estabelecimento.

**RF 09 – Selecionar raio de distância para obter estabelecimentos localizados na área**

Este requisito refere-se à funcionalidade de seleção de um raio de distância pelo consumidor para encontrar estabelecimentos dentro do raio que for selecionado. Neste caso, o consumidor deve selecionar um raio de distância em km, a fim de identificar o limite máximo de distância para seleção dos estabelecimentos disponíveis que estarão cadastrados no sistema.

Isso deve ser realizado de três maneiras distintas:

1. permissão de localização do usuário por meio do *Global Positioning System* (GPS);
2. opção selecionável para endereço de cadastro do consumidor;
3. campo para inserção de uma localização geográfica específica.

Desta maneira, o sistema deve exibir mensagens informativas que orientem o usuário em caso de falhas para proporcionar a ele uma melhor experiência de uso. Caso haja falha de conexão com o servidor da aplicação, o sistema deve orientar o usuário sobre a necessidade da conexão para que seja possível a seleção do raio de distância. O sistema deve exibir, por exemplo, a mensagem: “Falha de conexão com o servidor. Verifique sua conexão e tente novamente”.

Em caso de sucesso, o sistema deve exibir a listagem de todos os estabelecimentos que estão dentro do raio de distância indicado pelo consumidor.

Prioridade: 🗹 Importante

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 05 - Cadastrar estabelecimento;
3. RF 10 - Gerenciar lista de compras.

**RF 10 – Gerenciar lista de compras**

Este requisito refere-se à funcionalidade de gerenciamento de uma lista de compras criada pelo consumidor.

O gerenciamento da lista de compras permite que o sistema disponibilize ao consumidor a funcionalidade de:

1. criar uma lista de compras;
2. consultar uma lista de compras;
3. alterar uma lista de compras;
4. excluir uma lista de compras.

Uma lista de compras pode ser criada sob duas circunstâncias e, para isso, o usuário deve selecionar a forma que deseja criá-la:

1. para um estabelecimento específico - neste caso, o consumidor deve selecionar o estabelecimento para o qual deseja realizar suas compras ou realizar o orçamento de preços desse supermercado. Em seguida, deve haver uma opção para inclusão dos produtos e quantidades. À medida que os itens forem adicionados à lista, os seus respectivos preços devem ser apresentados ao usuário;
2. sem que haja um estabelecimento específico - neste caso, não há a necessidade de seleção de um estabelecimento, somente deve ser feita a inclusão dos produtos e quantidades na lista que, ao serem selecionados, também devem ser apresentados os respectivos preços ao usuário.

Ao término da criação da lista de compras, independente da circunstância escolhida, o sistema deve exibir a opção para gravar a lista criada. Caso o usuário não grave a lista, o sistema deve garantir que a lista seja descartada, no entanto, deve exibir uma mensagem de confirmação para o usuário, por exemplo: “Deseja descartar a lista?”, com as opções “Sim” e “Não”.

Portanto, é necessário que o sistema possa prover mensagens informativas que orientem cada ação do usuário. Ao criar, alterar ou excluir uma lista, é necessário que o sistema confirme a ação para o usuário consumidor. Por exemplo: “Lista cadastrada com sucesso”, e/ou “Lista alterada com sucesso” e/ou “Lista removida com sucesso”.

Além do mais, ao pesquisar qualquer produto pelos filtros a serem disponibilizados pelo sistema e adicioná-lo à lista, o sistema deve criá-la automaticamente com um nome padrão, por exemplo: “Lista 1” e deve permitir que o usuário modifique o nome da lista, pelo campo “Nome”, ao longo de seu uso.

Prioridade: 🗹 Importante

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação.

**RF 11 – Direcionar lista de compras**

Este requisito refere-se à funcionalidade de direcionamento da lista de compras elaborada. Para isso, o consumidor terá as opções de:

1. enviar para o *e-mail* do usuário;
2. enviar para qualquer outro *e-mail*;
3. fazer *download* da lista.

Diante das opções de direcionamento de listas de compras, o sistema deve:

1. prover o campo para envio da lista para o consumidor ou outra pessoa. Neste campo, o usuário deve preencher o *e-mail* que deseja direcionar a lista de compras;
2. prover a opção de gerar a lista que foi elaborada para que o consumidor possa imprimi-la ou gravá-la em formato PDF;
3. confirmar a ação do usuário com uma mensagem informativa. Por exemplo: “Lista direcionada com sucesso”;
4. direcionar o usuário caso não haja conexão com o servidor de *e-mail* no momento do envio ou o *e-mail* informado não esteja no padrão correto. Por exemplo, caso não haja conexão com o servidor de *e-mail* no momento do direcionamento da lista, o sistema deve exibir a mensagem: “Falha de conexão com o servidor de *e-mail*. Verifique sua conexão e tente novamente”. Caso o *e-mail* informado pelo usuário não seja válido, o sistema deve exibir a mensagem: “*E-mail* incorreto. Digite um *e-mail* correto para direcionar sua lista”.

Prioridade: 🗹 Desejável.

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 10 - Gerenciar lista de compras.

**RF 12 – Marcar uma lista de compras como favorita**

Este requisito é referente à funcionalidade de marcar como favorita uma lista de compras que foi criada. Dessa mesma maneira, é necessário que o sistema confirme a ação de marcar como favorita a lista de compras do consumidor por meio de uma mensagem. Por exemplo: “Lista X foi marcada como favorita”, sendo que “X” refere-se ao nome da lista que foi criada pelo usuário.

Esse requisito serve para agilizar o processo de gerenciamento de uma lista. A partir desse requisito, o consumidor pode obter fluxos alternativos que o levem para:

1. direcionamento da lista conforme Requisito RF 11;
2. alteração de algum item da lista;
3. gerenciamento da lista em geral.

Prioridade: 🗹 Desejável.

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 10 - Gerenciar lista de compras.

**RF 13 – Apresentar produtos ofertados em promoção**

Este requisito refere-se à funcionalidade apresentação de todos os produtos ofertados de um período específico para um consumidor e que estão em promoção. Geralmente cada estabelecimento determina um período específico para que seus produtos estejam disponíveis por um preço promocional. Desta maneira, o usuário deve ter acesso a dois fluxos:

1. acesso às promoções na página inicial do sistema;
2. acesso às promoções na página de pesquisa de produto.

O sistema deve, ao disponibilizar um produto ofertado em promoção, permitir que itens sejam expostos para o usuário. São eles:

1. foto do produto;
2. nome do produto;
3. marca fornecedora do produto;
4. descrição do produto;
5. preço antigo do produto;
6. preço atual do produto (preço em promoção);
7. período da promoção.
8. unidade do produto.

Desta maneira, o usuário obtém um fluxo mais simples para adicionar o produto em que está em promoção à lista.

Prioridade: 🗹 Importante

Requisito(s) relacionado(s):

1. RF 14 - Pesquisar produto;
2. RF 17 - Gerenciar preço de produto.

**RF 14 – Pesquisar produto**

Este requisito refere-se à funcionalidade de pesquisa referente a todos os produtos ofertados por um estabelecimento pelo consumidor.

Desta maneira, os consumidores terão acesso a determinados filtros, que facilitam sua busca. Por sua vez, os filtros são:

1. por linha;
2. por marca; Não está no modelo de dados
3. por fornecedor;
4. por produtos em promoção.

Caso o usuário selecione um produto, o sistema deve disponibilizar a opção de adicioná-lo à lista de compras.

Por outro lado, é necessário que o sistema apresente mensagens de falha caso o produto pesquisado pelo usuário não exista, ou caso o usuário tenha pesquisado por um produto de maneira incorreta. Assim, o sistema deve exibir uma mensagem genérica que informe ao usuário que o produto pesquisado não foi encontrado. Por exemplo: “O produto pesquisado não foi encontrado”.

Prioridade: 🗹 Essencial.

Requisito(s) relacionado(s):

1. RF 17 - Gerenciar preço de produtos.
2. Módulo do estabelecimento

Este módulo agrupa um conjunto de funcionalidades relacionadas ao estabelecimento comercial, no caso a Pessoa Jurídica.

**RF 15 – Gerenciar cadastro de promoção**

Este requisito refere-se à funcionalidade de gerenciar os cadastros de todas as promoções referentes aos produtos, por parte do estabelecimento. Desta maneira, o sistema deve permitir que o estabelecimento possa:

1. criar um cadastro de promoção;
2. consultar um cadastro de promoção;
3. alterar um cadastro de promoção;
4. excluir um cadastro de promoção.

Cada estabelecimento determina um período específico para que seus produtos estejam disponíveis por um preço ofertado menor. Desta maneira, o usuário tem acesso a dois fluxos:

1. acesso às promoções na página inicial;
2. acesso às promoções na página de pesquisa.

O sistema deve garantir que seja disponibilizado um formulário que contenha os seguintes campos a serem preenchidos pelo estabelecimento. São eles:

1. foto do produto;
2. nome do produto;
3. marca fornecedora do produto;
4. descrição do produto;
5. preço antigo do produto;
6. preço atual do produto (preço em promoção);
7. período da promoção;
8. unidade do produto.

Além dessa garantia, o sistema deve prover:

1. mensagens informativas que orientem o usuário em caso de falhas. Caso haja falta de conexão com o servidor da aplicação, os dados estejam incorretos e não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique a falha ou sucesso. Por exemplo: Caso o usuário digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Nome contém caracteres especiais. Remova para prosseguir”, ou caso haja falta de conexão com o servidor da aplicação, o sistema deve exibir a mensagem: “Falha de conexão com o servidor da aplicação. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o usuário siga regras para criação de uma promoção, gerando assim, dados consistentes e íntegros;
2. mensagem informativa que confirme a edição ou exclusão do um cadastro de uma promoção. Por exemplo: “Promoção alterada com sucesso” e/ou “Promoção excluída com sucesso”.

Prioridade: 🗹 Importante

Requisito(s) relacionado(s):

1. RF 14 - Pesquisar produto;
2. RF 17 - Gerenciar preço de produto.

**RF 16 – Editar perfil do estabelecimento**

Esse requisito refere-se à funcionalidade de edição do perfil do estabelecimento. O formulário para atualização dos dados deve conter os seguintes itens:

1. nome da empresa;
2. razão social;
3. *e-mail*;
4. logradouro;
5. número;
6. bairro;
7. município;
8. UF;
9. CEP;
10. Telefone fixo e telefone móvel.

O sistema deve assegurar algumas condições e são elas:

1. a rigidez no controle de cadastro de usuário: o sistema deve assegurar que haja somente um cadastro para cada CNPJ e *e-mail*;
2. mensagens informativas que orientem o usuário em caso de falha: caso haja falta de conexão com o servidor da aplicação, os dados estejam incorretos ou não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique o problema e a solução Por exemplo: caso o usuário digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Nome contém caracteres especiais. Remova para prosseguir”. Caso haja falta de conexão, o sistema deve exibir a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o usuário siga regras para alteração de uma conta do estabelecimento, gerando assim, dados consistentes e íntegros;
3. mensagem informativa que notifique o sucesso ao cadastrar uma conta de um usuário do tipo estabelecimento. Por exemplo: “Usuário cadastrado com sucesso”;
4. o cadastro deve ser confirmado pelo usuário acessando seu *e-mail* e ativando o *link* de confirmação;
5. o campo “CNPJ”, “CEP” e “Telefone” devem conter formatação padrão. O CNPJ deve ter formatação padrão em: XX.XXX.XXX/YYYY-ZZ. O CEP deve conter a formatação padrão em: XX.XXX-XXX. O telefone deve conter formatação padrão em (XX) X XXXX-XXXX para celular ou (XX) XXXX-XXXX para telefone fixo.

Prioridade: 🗹 Essencial

Requisito(s) relacionado(s):

1. RF 01 - Realizar autenticação;
2. RF 05 - Cadastrar estabelecimento.

**RF 17 – Gerenciar preço de produto**

Este requisito refere-se à funcionalidade de gerenciar os preços dos produtos que foram devidamente cadastrados por parte do estabelecimento e/ou administradores do sistema. Desta maneira, o sistema deve permitir que o estabelecimento possa:

1. consultar o preço de um produto;
2. alterar o preço de um produto;
3. excluir um preço de um produto que implica em excluir todo o cadastro de um produto.

Para que seja possível alterar um preço, o sistema deve exibir um formulário que contenha os seguintes campos:

1. foto do produto;
2. nome do produto;
3. marca fornecedora do produto;
4. descrição do produto;
5. preço do produto;
6. unidade do produto;
7. linha do produto.

Somente o campo “e)” não deve ter um cadastro prévio, sendo necessário o preenchimento por parte do estabelecimento. Os outros dados estarão cadastrados, sendo necessária somente a escolha entre eles e como foco principal deste requisito, a alteração do preço. Esse requisito é necessário a medida que cada estabelecimento determina um preço para seus produtos. Por exemplo, dois estabelecimentos podem oferecer o mesmo produto com preços diferentes.

Havendo uma alteração do preço em um cadastro de produto, o sistema deve assegurar que:

1. em caso de opção pela exclusão de um cadastro de produto, haja a confirmação por parte do estabelecimento que o anunciou. Esse item informa que não é possível que haja a exclusão apenas do item “Preço”, implicando que o sistema exclua todo o cadastro do produto;
2. se o preço do produto for apenas alterado é necessário que o usuário grave a(s) modificação(ões), para que os dados do produto sejam atualizados, o que deve garantir a mudança permanente desses dados;
3. um cadastro de produto não deve ser gravado caso o estabelecimento não preencha o “Preço do produto” ;
4. mensagens informativas que orientem o usuário em caso de falhas. Caso haja falta de conexão com o servidor da aplicação, os dados estejam incorretos ou não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique a falha ou o sucesso. Por exemplo: caso o usuário insira caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Nome contém caracteres especiais. Remova para prosseguir” ou caso haja falta de conexão, o sistema deve exibir a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o usuário siga regras para alteração do preço do produto, gerando assim, dados consistentes e íntegros;
5. mensagem informativa que confirme a edição de um preço de um produto ou a exclusão de todo seu cadastro. Por exemplo: “Produto alterado com sucesso” e/ou “Produto excluído com sucesso”.

Prioridade: 🗹 Essencial.

Requisito(s) necessário(s):

1. RF 01- Realizar autenticação;
2. RF 22 – Gerenciar cadastro de produto.

**RF 18 – Realizar plano de assinatura**

Este requisito refere-se à funcionalidade da realização de um plano de assinatura por parte do estabelecimento.

Os planos consistem em três níveis:

1. plano bronze;
2. plano prata;
3. plano ouro.

O plano bronze consiste em:

1. exibição da média das avaliações realizadas pelos consumidores. Isso deve influenciar na popularidade do estabelecimento em relação aos demais.

O plano prata consiste em:

1. conteúdo do plano bronze;
2. exibição de cada avaliação realizada.

O plano ouro consiste em:

1. conteúdo do plano prata;
2. visualização do *dashboard*.

Prioridade: 🗹 Essencial

Requisito(s) necessário(s):

1. RF 01 - Realizar autenticação;
2. RF 05 - Cadastrar estabelecimento.

**RF 19 – Visualizar *dashboard***

Este requisito refere-se à funcionalidade de visualização de *dashboard* por parte do estabelecimento.

Os estabelecimentos poderão observar estatísticas referentes a:

1. satisfação geral dos consumidores;
2. gráfico de onde está localizado o maior número de consumidores;
3. exibição dos produtos mais procurados;
4. exibição dos produtos que foram adicionado mais vezes em uma lista de compra;
5. quantidade de votos por cada item da avaliação.

Prioridade: 🗹 Essencial

Requisito(s) necessário(s):

1. RF 01 - Realizar autenticação;
2. RF 18 - Realizar plano de assinatura.
3. Módulo do administrador

Este módulo agrupa um conjunto de funcionalidades relacionadas ao usuário que é o responsável pela administração do sistema. Esse administrador ou administradores devem ter acesso ao gerenciamento dos módulos. Os seguintes módulos são:

1. módulo consumidor;
2. módulo estabelecimento.

**RF 20 – Gerenciar linhas**

Este requisito refere-se à funcionalidade de gerenciar linhas (ou categorias) as quais os produtos pertencem. Desta maneira, o sistema deve permitir:

1. consultar uma linha;
2. alterar uma linha;

Desta maneira, o sistema deve permitir:

1. mensagens informativas que orientem o usuário em caso de falhas. Caso haja falta de conexão com o servidor, o sistema deve exibir a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”, ou caso o usuário não selecione uma linha para um produto, é necessário que o sistema informe ao usuário a necessidade dessa ação, por exemplo: “Um produto sempre deve pertencer a uma linha. Selecione uma linha para prosseguir”. Desta maneira, é fundamental que o usuário siga regras para alteração de uma linha, gerando assim, dados consistentes e íntegros;
2. mensagens informativas que confirmem o sucesso da operação. Por exemplo, caso haja sucesso na alteração de uma linha é necessário que o sistema exiba a mensagem: “Linha alterada com sucesso”;
3. não é possível a exclusão de uma linha. Neste caso, o sistema não deve permitir que um produto não esteja contido em uma linha e, desta forma, só deve ser possível a edição de uma linha para que um produto sempre pertença a uma delas. Isso facilita, por exemplo, o processo de pesquisa a um item por uma linha.

A opção de consulta a uma linha fornece ao usuário a opção de listagem todos os produtos os quais pertencem a linha que foi pesquisada.

Vale ressaltar que as linhas são pré-cadastradas pelos administradores do sistema. No momento de alteração, é necessário que os clientes apenas selecionem a opção dentre as que estão disponíveis previamente. Desta maneira, o sistema garante que não haja em um cadastro de produto, um produto que não pertença a uma linha, pois um produto sempre deve pertencer a uma linha.

Prioridade: 🗹 Essencial

Requisito(s) necessário(s):

1. RF 01- Realizar autenticação.

**RF 21 – Gerenciar cadastro de consumidor e/ou estabelecimento**

Este requisito refere-se à funcionalidade de gerenciamento dos cadastros dos usuários do sistema que por sua vez são os consumidores e estabelecimentos, sendo permitido:

1. criar cadastros de consumidores e/ou estabelecimentos;
2. consultar cadastros de consumidores e/ou estabelecimentos;
3. alterar cadastros de consumidores e/ou estabelecimentos;
4. excluir cadastros de consumidores e/ou estabelecimentos.

Desta maneira, o sistema deve permitir:

1. mensagens informativas que orientem o administrador em caso de falhas. Caso haja falta de conexão com o servidor, os dados estejam incorretos ou não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique a falha ou sucesso. Por exemplo: caso o administrador digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Nome contém caracteres especiais. Remova para prosseguir”, ou caso haja falta de conexão, o sistema deve exibir a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o administrador também siga regras para gerenciamento das contas dos usuários, gerando assim, dados consistentes e íntegros e que repercutem as mudanças.
2. os administradores só podem alterar as senhas dos usuários caso essa ação seja requisitada;
3. mensagens informativas que confirmem o sucesso da operação. Por exemplo, caso haja sucesso na criação, edição ou remoção de um cadastro de consumidor e/ou estabelecimento, é necessário que o sistema exiba a mensagem: “Usuário cadastrado com sucesso”, ou “Usuário alterado com sucesso” e “Usuário excluído com sucesso”;
4. caso o administrador exclua um cadastro de um usuário consumidor e/ou estabelecimento, esses usuários não podem acessar o sistema;
5. o campo “CPF”, “CEP” e “Telefone” devem conter formatação padrão. CPF deve ter formatação padrão em: XXX.XXX.XXX-XX. O CEP deve conter a formatação padrão em: XX.XXX-XXX. O telefone deve conter formatação padrão em (XX) X XXXX-XXXX para celular ou (XX) XXXX-XXXX para telefone fixo.

Prioridade: 🗹 Essencial

Requisito(s) necessário(s):

1. RF 01- Realizar autenticação.

**RF 22 – Gerenciar cadastro de produto**

Este requisito refere-se à funcionalidade de gerenciar os cadastros dos produtos do sistema, sendo permitido:

1. criar um produto;
2. consultar um produto;
3. alterar um produto;
4. excluir um produto.

O sistema deve prover um formulário que contenha os dados descritos a seguir. Este formulário será necessário para o cadastro de um produto.

1. foto do produto;
2. nome do produto;
3. marca fornecedora do produto;
4. descrição do produto;
5. preço do produto;
6. preço atual do produto (Se estiver em promoção);
7. período da promoção (Se estiver em promoção);
8. unidade do produto.

Desta maneira, o sistema deve permitir:

1. mensagens informativas que orientem o administrador em caso de falhas. Caso haja falta de conexão, os dados estejam incorretos ou não sigam a formatação padrão, é necessário que o sistema exiba uma alerta ao usuário, que indique a falha ou sucesso. Por exemplo: caso o administrador digite caracteres especiais no campo “Nome”, o sistema deve exibir a mensagem de “Nome contém caracteres especiais. Remova para prosseguir”, ou caso haja falta de conexão, o sistema deve exibir a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o usuário siga regras para gerenciamento dos cadastros dos produtos, gerando assim, dados consistentes e íntegros;
2. mensagens informativas que confirmem o sucesso da operação. Por exemplo, caso haja sucesso na criação, edição ou remoção de um cadastro de um produto é necessário que o sistema exiba a mensagem: “Produto cadastrado com sucesso” ou “Produto alterado com sucesso” ou “Produto excluído com sucesso”;
3. no campo “Foto”, o sistema deve garantir que os formatos permitidos sejam JPEG e PNG. Os demais formatos não devem ser permitidos para *upload* .

Prioridade: 🗹 Essencial

Requisito(s) necessário(s):

1. RF 01- Realizar autenticação.

**RF 23 – Gerenciar plano de assinatura**

Este requisito refere-se à funcionalidade de gerenciar os cadastros dos planos de assinatura do sistema. Desta maneira, o sistema deve permitir:

1. criar um plano de assinatura;
2. consultar um plano de assinatura;
3. alterar um plano de assinatura;
4. excluir um plano de assinatura.

Desta maneira, o sistema deve permitir:

1. mensagens informativas que orientem o usuário em caso de falhas. Caso haja falta de conexão com o servidor da aplicação, o sistema deve exibir a mensagem: “Falha de conexão. Verifique sua conexão e tente novamente”. Desta maneira, é fundamental que o administrador siga regras para gerenciamento dos planos de assinatura, gerando assim, dados consistentes e íntegros;
2. mensagens informativas que confirmem o sucesso da operação. Por exemplo, caso haja sucesso na criação, edição ou remoção de um cadastro de plano de assinatura, é necessário que o sistema exiba a mensagem: “Plano de assinatura cadastrado com sucesso” ou “Plano de assinatura alterado com sucesso” ou “Plano de assinatura excluído com sucesso”;
3. somente o administrador pode criar um novo plano de assinatura. Desta maneira, todas as opções de planos de assinaturas são provenientes do administrador do sistema, que pode gerenciá-los e aumentá-los de acordo com a demanda e *feedback* dos estabelecimentos.

Prioridade: 🗹 Essencial

Requisito(s) necessário(s):

1. RF 01 - Realizar autenticação.

### 5.1.2 Requisitos Não Funcionais

Os requisitos não funcionais são aqueles diretamente relacionados às propriedades do sistema como confiabilidade, tempo de resposta e espaço de armazenamento disponível para instalação. Os requisitos não funcionais podem também definir restrições, como a capacidade dos dispositivos de entrada e saída, representações de dados nas interfaces do sistema, entre outros. Segundo Sommerville (2007), os requisitos não funcionais ou RNF são restrições impostas sobre os serviços ou funções oferecidas pelo sistema. Entre elas, estão restrições de tempo, restrições sobre o processo de desenvolvimento e padrões a serem seguidos. Os requisitos não funcionais aplicam-se, frequentemente, ao sistema como um todo. Em geral, não se aplicam às características ou serviços individuais do sistema.

Segundo a proposta de Sommerville (2007), as categorias de requisitos não funcionais são compreendidas de três formas: requisitos de produto, organizacionais e externos. De acordo com ele: “os requisitos não funcionais, como o nome sugere, são aqueles não diretamente relacionados às funções específicas fornecidas pelo sistema” (SOMMERVILLE, 2007, p.82).

A Figura 33 ilustra as 3 categorias.

O projeto EzMart se baseia nas categorias de RNFs de Sommerville e classifica os requisitos não funcionais nas próximas seções.

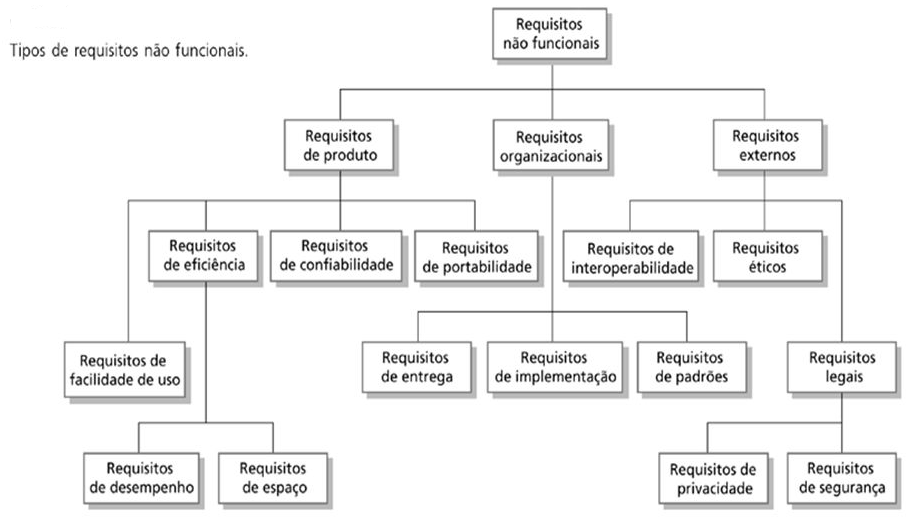


Figura 35 - Tipos de requisitos funcionais

Fonte: Sommerville (2007)

#### 5.1.2.1 Requisitos de Produto

Os requisitos de produto especificam o comportamento do produto, como desempenho, taxa de aceitação de falhas e usabilidade. De acordo com Sommerville (2008, p. 83), os requisitos não funcionais de produto “especificam o comportamento do produto”.

1. Requisitos de eficiência

**RNF 01 – Disponibilidade do sistema**

Para Kardec e Nascif (2002), a disponibilidade é o tempo em que o equipamento, sistema ou instalação está disponível para operar ou em condições de produzir. Ela pode ser calculada pela relação entre a diferença do número de horas do período considerado (horas calendário) com o número de horas de intervenção para manutenções (manutenção preventiva por tempo ou por estado, manutenção corretiva e outros serviços) para cada item observado e o número total de horas.

Desta maneira, o sistema EzMart deve oferecer a taxa de disponibilidade do servidor da aplicação de 98%, aproximadamente, 358 dias por ano.

**RNF 02 – Escalabilidade**

Escalabilidade é um termo usado descrever uma propriedade de sistemas. Hill em 1990 propôs um desafio para descrever formalmente o que é a escalabilidade e, depois dela, muitas definições foram realizadas, formais ou informais, sendo que não há apenas uma definição amplamente aceita, por se tratar de um tópico multidimensional. Clements et al. (2002) afirmam que a arquitetura de sistemas é um tópico multidimensional, assim como a escalabilidade. Portanto, ao se referir ao termo escalabilidade devem-se levar em consideração importantes aspectos como desempenho, manutenibilidade, usabilidade, confiabilidade, segurança e disponibilidade (DUBOC et al., 2007). Duboc et al. (2007, p. 375) definem escalabilidade como:

Uma qualidade de sistemas de *software* caracterizada pelo impacto causal que aspectos de escalabilidade do ambiente do sistema e seu projeto tem em certas qualidades mensuráveis a medida que estes aspectos variam dentro de uma faixa operacional esperada. Se um sistema pode acomodar esta variação de alguma maneira que é aceitável para os interessados, então o sistema é escalável.

Portanto, a escalabilidade pode ser compreendida como a capacidade do sistema de acomodar cargas de trabalho variantes, enquanto continua a satisfazer todos os outros requisitos funcionais e não funcionais.

O sistema EzMart deve prover a capacidade de processar crescentes e variantes cargas de trabalho, mantendo ou aumentando o seu desempenho, conforme a demanda.

1. Requisitos de confiabilidade

**RNF 03 – Consistência de dados**

A consistência de dados refere-se a todas as transações em um banco de dados que não violem nenhuma restrição de integridade durante a execução de alguma tarefa. Uma transação é a execução de um programa, vista pelo Sistema Gerenciador de Banco de Dados (SGBD) como uma série de leituras e escritas. Um SGBD deve garantir quatro propriedades de transação para manter os dados protegidos de acesso concorrente e de falhas de sistema:

1. atomicidade: a execução de toda transação deve ser considerada atômica; ou todas as ações são executadas ou nenhuma delas é.;
2. consistência: cada transação deve preservar a consistência do Banco de Dados;
3. isolamento: toda transação é isolada ou protegida das ações de outras transações concorrentes;
4. durabilidade: uma vez informada a conclusão de uma transação de maneira satisfatória, o SGBD deve persistir os resultados da transação mesmo que o sistema sofra uma queda antes que esses resultados sejam persistidos no disco.

O acrônimo ACID é utilizado para referências das quatro propriedades citadas acima: atomicidade, consistência, isolamento e durabilidade.

O sistema EzMart deve implementar regras e restrições para a gravação dos dados no Banco de Dados, assim como validações de dados conforme descrição em cada requisito funcional tratado neste documento.

1. Requisitos de usabilidade

**RNF 04 – *Layout* responsivo**

Usuários de dispositivos móveis que fazem uso da *web* se deparam em diversas situações com problemas de visualização de sites não adaptativos aos diversos dispositivos existentes, como fontes pequenas, imagens cortadas, *links* ilegíveis e conteúdos distorcidos. A tecnologia móvel tem avançado, fazendo com que o acesso à *Web* seja feito por meio de vários tipos de dispositivos, sendo necessário considerar que suporte além de monitores *desktop*, telas pequenas, *touch screen, smart* TV, redes móveis e muitas outras tecnologias (FISHER; SHARKIE, 2013, p. 1).

A Figura 34 ilustra o comportamento de um site responsivo.

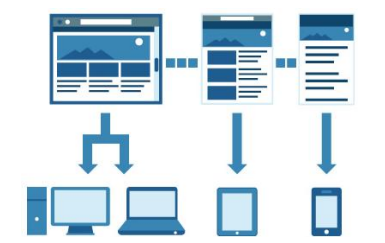


Figura 36 - Comportamento de um site responsivo

Fonte: Master Studio Web (2012)

O conteúdo de um site responsivo se adapta à área de projeção disponível, adequando a visualização e a navegabilidade sem ocultar as informações, independente do dispositivo, independente da resolução, tamanho de tela, interface com *touch* ou mouse, seja ele móvel ou não. Neste contexto é importante conhecer algumas resoluções de telas, levando em consideração sua orientação retrato e paisagem. A Figura 35 ilustra as resoluções.

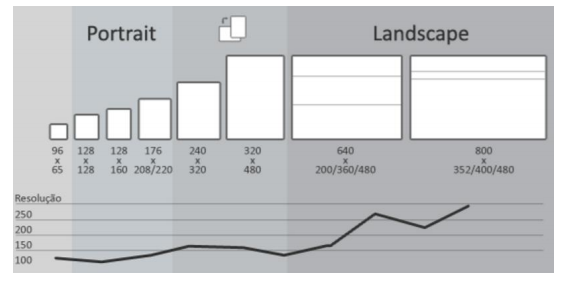


Figura 37 - Tipos de resoluções de telas

Fonte: Zemel (2013, p. 21)

Não é o tamanho físico da tela do dispositivo que importa no *design* responsivo, mas sua resolução. O *Web design* responsivo envolve uma série de técnicas e tecnologias combinada para fazer uma única aplicação funcionar em uma variedade de dispositivos de modo mais prático possível (FISHER; SHARKIE, 2013, p. 2).

O sistema EzMart deve ter um *layout* responsivo e acessível a todos os possíveis usuários e adaptado à maioria dos dispositivos de acesso.

**RNF 05 – Acessibilidade**

O Modelo de Acessibilidade de Governo Eletrônico (e-MAG) foi formulado para orientar profissionais que tenham contato com publicação de informações ou serviços na Internet a desenvolver, alterar e/ou adequar páginas, sites e portais, tornando-os acessíveis ao maior número de pessoas possível (E-GOV, 2000).

O sistema EzMart deve seguir o e-MAG para tornar a aplicação *web* acessível a todos os possíveis usuários.

**RNF 06 – Interface gráfica de usuário minimalista**

No sistema EzMart, as imagens devem ter tamanhos flexíveis a fim de que sejam mostradas adaptadas à tela do dispositivo. Devem-se privilegiar os dados e as informações necessárias para o usuário no momento da interação, nunca de forma a levar a um excesso de conteúdo ou exibir “informações vazias” dentro de um contexto. Para atingir uma interface minimalista deve-se atentar à clareza das informações, simplicidade dos componentes do *layout* e utilizar uma linguagem familiar ao usuário.

#### 5.1.2.2 Requisitos Organizacionais

Os requisitos organizacionais são decorrentes de políticas e procedimentos da organização do cliente e do desenvolvedor, que possuem padrões a serem utilizados, prazos das entregas e metodologias aplicadas (SOMMERVILLE, 2007).

1. Requisitos de entrega:

**RNF 07 – Datas para entrega dos incrementos**

As datas-marco para entrega dos incrementos estão descritas detalhadamente na seção 4.1.2.

1. Requisitos de entrega

**RNF 08 – Convenção de codificação**

As convenções de codificação definem um conjunto de regras que devem ser seguidas durante a escrita do código-fonte, visando o aumento da legibilidade do código-fonte produzido por um programador e à facilidade na manutenção do código por qualquer programador, podendo reduzir o tempo na migração, no rastreamento e correção de erros e diminuir a reescrita de código por falta de documentação.

Para codificação do EzMart deve ser utilizada a *Java Code Conventions* publicada pela *Sun Microsystems*, criadora da linguagem de programação Java. Como exemplo, podem-se notar as convenções para nomes de classes, atributos, objetos, métodos e variáveis. A Figura 36 mostra uma convenção de codificação para escrever classes, definindo com o uso do “*Upper Camel Case*”.



Figura 38 - Exemplo de padrão na convenção de codificação Java

Fonte: Equipe (2018)

**RNF 09 – Ambiente de desenvolvimento**

A *Integrated Development Environment* (IDE) utilizada para a programação do sistema EzMart deve ser o *NetBeans* 8.1.

**RNF 10 – Linguagem de programação**

Para a programação de todos os módulos deve ser utilizada a linguagem de programação Java, com o *Java Development Kit* (JDK) versão 8 e o *Java Runtime Environmnet* (JRE) versão 1.8.0.65.

**RNF 11 – Política de senhas**

As senhas de acesso dos usuários devem ser formadas por no mínimo 06 caracteres, incluindo números e letras e excluindo caracteres especiais;

Todas as senhas devem ser armazenadas de forma criptografada pelo sistema EzMart. Para criptografia, o método a ser utilizado é o *Message-Digest Algorithm* (MD-5) que utiliza um algoritmo de *hash* de 128 bits unidirecional desenvolvido pela *RSA Data Security, Inc.*

1. Requisitos de padrões

**RNF 12 – Padrão de documentação**

Toda a documentação do projeto deve seguir as diretrizes para elaboração de trabalhos científicos da FAI – Centro de Ensino Superior em Gestão, Tecnologia e Educação. Essas diretrizes técnicas são embasadas nas normas ABNT, que garantem formatação apropriada para todos os trabalhos técnico-científicos. As normas abordam, por exemplo, estrutura do trabalho, formato de letras e tamanhos, formatação dos títulos e subtítulos e assim por diante.

**RNF 13 – Processo de desenvolvimento**

Vide o detalhamento na seção 4.1.2.

#### 5.1.2.3 Requisitos Externos

Segundo Sommerville (2007), os requisitos externos são provenientes de fatores externos ao sistema e seu processo, sendo composto por sistemas de outras organizações que interagem com o sistema e requisitos legais que devem ser seguidos para certificar que o sistema funciona em conformidade com as leis vigentes e seguindo requisitos éticos.

1. Requisitos de interoperabilidade:

**RNF 14 – Compatibilidade de navegadores**

O sistema deve assegurar a compatibilidade com o navegador *Google Chrome.* 70.0.3510.2

**RNF 15 – Provedor de mapas**

Para localização dos estabelecimentos mais próximos, de acordo com o raio de distância selecionado pelos consumidores, o sistema deve utilizar a plataforma do *Google Maps*.

**RNF 16 – Sistema Gerenciador de Banco de dados (SGDB)**

O SGBD que deve ser utilizado para persistência dos dados do sistema EzMart é o *PostgresSQL* 9.4.11.

**RNF 17 – servidor Web**

O servidor de aplicação que deve ser utilizado é o *Apache Tomcat* 8.0.27

**RNF 18 – servidor de E-mail**

O servidor de *E-mail* que deve ser utilizado é o *Gmail*.

1. Requisitos éticos

**RNF 19 – Confidencialidade das informações**

O sistema EzMart deve garantir a confidencialidade dos dados e para isso deve implementar um mecanismo de controle de acesso, que permite que apenas usuários autorizados tenham acesso à determinadas informações e/ou funcionalidades. Isto pode ser compreendido como um aspecto estratégico, pois a confidencialidade da informação deve proteger o capital intelectual no portfólio de informações dos estabelecimentos e por consequência as vantagens competitivas do estabelecimento.

**RNF 20 – Privacidade nos dados cadastrais**

A proteção dos dados dos consumidores deve ocorrer observando o direito à privacidade garantidos pela Constituição Federal e pelo Código de Defesa do Consumidor que tratam da proteção a liberdade dos indivíduos em suas relações de consumo.

Portanto, o sistema EzMart deve proteger dados pessoais, cadastrais e hábitos do consumidor para usuários autorizados. Neste caso, somente com os dados de autenticação referentes a cada usuário é que será possível que as informações sejam visualizadas.

## 5.2 ANÁLISE DOS REQUISITOS

Segundo Pressman (2016, p. 171):

[...] a abordagem de modelagem de análise, denominada análise orientada a objetos, concentra-se na definição de classes e na maneira como colaboram entre si para atender aos requisitos dos clientes.

Nesta seção, utilizam-se os Diagramas de Casos de Uso para apresentar a visão funcional do sistema EzMart.

### 5.2.1 Modelos de Caso de Uso

Pressman (2016, p. 173) afirma que embora haja muitas maneiras de se medir o sucesso de um sistema ou produto, a satisfação do usuário está no topo da lista.

[...] Se você entender, como usuários (e outros atores) querem interagir com um sistema, sua equipe de *software* estará mais capacitada a caracterizar, de maneira apropriada, os requisitos e a construir modelos de análise e projeto proveitosos. Portanto, a modelagem de requisitos como UML começa com a criação de cenários na forma de casos de uso, diagramas de atividades e diagramas de raias.

Um Diagrama de Caso de Uso:

1. é uma forma de modelagem de requisitos;
2. provê uma representação gráfica das funcionalidades do sistema, mostrando-as sendo executada pelos atores;
3. ajuda a definir o escopo do sistema.

Guedes (2009, p. 52) afirma que este é o diagrama mais abstrato e, portanto, o mais flexível e informal.

Este diagrama tem por objetivo apresentar uma visão externa geral das funcionalidades que o sistema deverá oferecer aos usuários, sem se preocupar com a questão de como tais funcionalidades serão implementadas. O diagrama de caso de uso é de grande auxílio para identificação e compreensão dos requisitos do sistema, ajudando a especificar e visualizar e documentar as características, funções e serviços do sistema desejado pelo usuário.

Por sua vez, após a elaboração do diagrama de casos de uso do sistema, eles precisam ser documentados, fornecendo instruções em linhas gerais de como deve ser o seu funcionamento, quais atividades se deve executar, qual evento deve forçar sua execução, quais atores podem utilizá-los e quais as restrições, entre outras.

Mediante a essas informações, os Diagramas de Caso de Uso para o sistema EzMart encontram-se disponíveis no Apêndice E.

Assim como os diagramas, encontra-se disponível no Apêndice F, a descrição textual de todos os casos de uso, com seus respectivos cenários de sucesso e falha.

### 5.2.2 Modelo Conceitual dos Dados

Visão de dados é algo essencial para a construção de um sistema de *software*. Ela mostra, de maneira conceitual, os dados que serão manipulados e que darão origens às informações oferecidas para seus usuários.

O modelo utilizado para construir a visão conceitual dos dados é o Modelo Entidade-relacionamento (MER), proposto por Peter Chen, em 1976, e que teve como base a Teoria Relacional de Edgar F. Codd, de 1970.

Segundo Alexandruk (2011, p. 56, grafo nosso):

Um MER é um modelo formal, preciso, não ambíguo. Isto significa que diferentes leitores de um mesmo MER devem sempre entender exatamente o mesmo. Tanto é assim, que um MER pode ser usado como entrada de uma ferramenta CASE (*Computer Aided software Engineering*) na geração de um Banco de Dados relacional.

O diagrama Entidade-Relacionamento do sistema EzMart encontra-se disponível no Apêndice G.

## 5.3 MÉTRICAS PARA ESTIMATIVA DO ESFORÇO

Segundo Pressman (2011), as métricas de Pontos de Função (PF) e Pontos por Caso de Uso (PCU) podem ser compreendidas como uma medida da funcionalidade fornecida pelo *software*, que pode ser usada para calcular o esforço e custo necessário do projeto, antecipar o número de erros que serão encontrados e pressupor o número de recursos humanos.

Vazquez, Simões e Albert (2009) também enumeram os seguintes benefícios obtidos com as estimativas:

1. controlar o andamento da produtividade de um determinado *software*. Um sistema pode ter mais de uma equipe envolvida em seu desenvolvimento, é possível avaliar a produtividade de diferentes equipes pela quantidade de PF entregados;
2. realizar a medição do tamanho funcional do *software* e com isso estimar, custo, esforço e prazo. Uma vez realizada a medição ou estimativa dos PF totais do sistema é possível utilizar este número para realizar derivações;
3. saber o tamanho funcional de um *software* para realizar comparações. Pode ser realizada uma avaliação entre dois ou mais sistemas;
4. tomar decisões do tipo “*Make or Buy*”, seria a decisão de desenvolver um sistema ou comprar uma solução pronta no mercado;
5. utilizar a medida para fundamentar contratos de compra e venda de *softwares* ou contratar serviços.

Neste projeto, são aplicadas as métricas de PF (segundo o IPFUG) e de PCU (segundo os métodos de Schneider e Winters e de Karner). Portanto, empregando essas métricas, é possível estimar o esforço em Homens-hora (Hh) necessário para o desenvolvimento do sistema EzMart, conforme mostra o Quadro 5.

|  |  |  |  |
| --- | --- | --- | --- |
| **Métrica** | **Fase 2** | **Fase 3 (Reestimativa)** | **Fase 4 (Reestimativa)** |
| Pontos por Função | 2969 Hh | 3111Hh |  |
| PCU – Schneider e Winters | 3386 Hh | 3529Hh |  |
| PCU – Karner | 2419 Hh | 2521Hh |  |

Quadro 5- Estimativas do esforço

Fonte: Equipe (2018)

O detalhamento dos valores encontra-se no Apêndice H.

# 6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE

Arquitetura de sistema é uma estrutura ou organização dos componentes que pertencem à aplicação e a maneira pela qual esses componentes interagem. Pressman (2011, p. 230) afirma que a arquitetura de um programa ou sistema computacional é a estrutura que abrange os componentes do *software,* as propriedades externamente visíveis destes componentes e as relações entre eles.

Um sistema é pode oferecer em certo número de visões, cada uma representando uma projeção de descrição completa e mostrando aspectos particulares dele.

A *Unified Modeling Language* (UML) ou Linguagem de Modelagem Unificada é uma linguagem gráfica e visual utilizada para modelar visões de sistemas computacionais orientados a objetos. Esta linguagem é padronizada pela *Object Management Group* (OMG), sendo adotada internacionalmente pela indústria de *software*.

De acordo com Guedes (2009, p. 20), a UML:

[...] começou com a união do método de Booch ao OMT de Jacobson, o que resultou no lançamento do Método Unificado, no final de 1995. Logo em seguida, Rumbaugh juntou-se a Booch e Jacobson na Rational Software, e seu método OOSE começou também a ser incorporado à nova metodologia. O trabalho de Booch, Jacobson e Rumbaugh, conhecidos popularmente como “Os Três Amigos”, resultou no lançamento, em 1996, da primeira versão da UML propriamente dita.

Cada visão é descrita por um número de diagramas da UML que contém informações que dão ênfase a aspectos particulares do sistema. Existe em alguns casos certa sobreposição entre os diagramas o que significa que um deste pode fazer parte de mais de uma visão.

Portanto, neste Capítulo são demonstradas diversas visões do sistema EzMart: estrutural, comportamental, de dados e física. Também mostra a análise de complexidade de partes do código do sistema, o projeto do sistema distribuído e o projeto da interação humano-computador.

## 6.1 VISÃO ESTRUTURAL

A visão estrutural permite observar os aspectos estáticos do sistema, sendo modelados por meio de alguns dos diagramas da UML. As classes, pacotes e seus respectivos relacionamentos são apresentados pelos diagramas estruturais, que auxiliam na especificação, construção e documentação do sistema.

Os seguintes diagramas são apresentados nesta seção: diagrama de pacotes, diagramas de classes e diagramas de objetos.

### 6.1.1 Diagrama de Pacotes

O objetivo do diagrama de pacotes elaborado por meio da UML é representar os subsistemas englobados por um sistema de forma a determinar as partes que o compõem. Guedes (2009, p.33) afirma que ele pode ser utilizado de forma independente ou associado a outros diagramas.

No caso da modelagem do projeto EzMart, este diagrama deve ser tratado de maneira independente, de forma a oferecer somente e visão dos pacotes do sistema. O diagrama de pacotes do sistema EzMart encontra-se disponível no Apêndice J.

### 6.1.2 Diagrama de Classes

Segundo Guedes (2009), o diagrama de classes é o provavelmente o diagrama mais utilizado e é um dos mais importantes da UML e serve de apoio para a maioria dos demais diagramas. Guedes (2009, p. 31) define o objetivo deste diagrama como sendo “definir a estrutura das classes utilizadas pelo sistema, determinando os atributos e os métodos que cada classe tem, além de estabelecer como as classes se relacionam e trocam informações entre si. ”.

O diagrama de classes de negócio do sistema EzMart encontra-se disponível no Apêndice K.

### 6.1.3 Diagrama de Objetos

O diagrama de objetos está diretamente relacionado ao diagrama de classes, tratado na seção anterior, pois se trata de um complemento para o diagrama de classes.

Guedes (2009, p. 32) explica que o diagrama de objetos fornece uma visão dos valores armazenados pelos objetos de um diagrama de classes em um determinado momento da execução de um sistema*.*

Um diagrama de objetos do sistema EzMart encontram-se disponível no Apêndice L.

## 6.2 VISÃO COMPORTAMENTAL

A visão comportamental mostra o comportamento dinâmico do sistema, retratando as interações existentes entre os objetos das classes, as modificações ocorridas durante sua execução, bem como, o fluxo adotado para atender um determinado cenário e suas respostas a eventos ou estímulos externos.

Encontra-se nesta seção, o projeto das interações e o diagrama de atividades.

### 6.2.1 Projeto das Interações

O projeto das interações diz respeito ao comportamento do sistema, mostrando a maneira como um grupo de objetos se relaciona durante a execução de um caso de uso, isso, baseado na troca de mensagens entre os mesmos.

Nesta seção são apresentados os diagramas de sequência e o seu respectivo diagrama da visão geral de interação.

#### 6.2.1.1 Diagramas de Sequência

O diagrama de sequência é um diagrama comportamental, que se preocupa com a ordem temporal em que as mensagens são trocadas entre os objetos envolvidos em um determinado processo, afirma Guedes (2009, p.33). Ele ainda diz que em geral, este diagrama baseia-a em um caso de uso definido pelo diagrama de mesmo nome e apoia-se no diagrama de classes para determinar os objetos das classes envolvidas em um processo.

O diagrama de sequência do caso de uso “Criar lista de compra” está disponível no Apêndice M.

#### 6.2.1.2 Diagrama de Visão Geral de Interação

O diagrama de visão geral da interação é uma variação do diagrama de atividade que fornece uma visão geral dentro de um sistema ou processo de negócio. Este diagrama passou a existir apenas a partir da UML, versão 2. (GUEDES, 2009).

O diagrama de visão geral de interação encontra-se disponível no Apêndice N.

### 6.2.2 Diagrama de Atividades

Guedes (2009) descreve este diagrama como sendo um caso especial do antigo diagrama de gráfico de estados (ou diagrama de máquina de estados). Esse diagrama é utilizado para descrever os passos a serem percorridos para a conclusão de uma atividade específica, podendo esta ser representada por um método com certo grau de complexidade, um algoritmo, ou mesmo através de um processo completo. O diagrama de atividade concentra-se na representação do fluxo de controle de uma atividade.

Um diagrama de atividades do sistema EzMart encontra-se disponível no Apêndice O.

## 6.3 VISÃO DE DADOS

A visão de dados tem como propósito mostrar as necessidades de dados em um banco de dados automatizado, isso, por meio de atividades técnicas bem definidas e organizadas.

São apresentados nesta seção o modelo operacional (ou modelo lógico) e o dicionário de dados do modelo operacional.

### 6.3.1 Modelo Operacional

O modelo operacional é elaborado para mostrar dados a serem persistidos em um sistema de banco de dados, com tecnologia específica. No caso do sistema EzMart, trata-se de sistemas de bancos de dados relacionais.

O modelo operacional do banco de dados do sistema EzMart encontra-se disponível no Apêndice P deste documento.

### 6.3.2 Dicionário de Dados do Modelo Operacional

O dicionário de dados compõe-se de uma lista organizada de todos os elementos de dados pertinentes ao sistema, onde, são descritas suas entradas, saídas e composição de dados.

O dicionário do banco de dados do EzMart encontra-se disponível no Apêndice Q deste documento.

## 6.4 VISÃO FÍSICA

A visão física mostra os aspectos físicos de um sistema, como a configuração dos nós de processamento em tempo de execução e os artefatos que nele existem, moldando a topologia do *hardware* em que o sistema é executado. (BOOCH; RUMBAUGH; JACOBSON, 2006).

Desta maneira, nesta seção encontra-se o diagrama de componentes que permite observar uma visão física do sistema.

### 6.4.1 Diagrama de Componentes

O diagrama de componentes identifica os componentes que fazem parte de um sistema, um subsistema, ou até mesmo os componentes ou classes internas de um componente individual. Guedes (2009, p. 321) descreve que “um componente pode representar tanto um componente lógico (um componente de negócio ou de processo) ou um componente físico como arquivos contendo código-fonte, arquivos de ajuda, bibliotecas, ou arquivos executáveis”.

Os diagramas de componentes do sistema EzMart estão disponíveis no Apêndice R.

### 6.4.2 Estrutura Física do Banco de Dados

Segundo Lima (2016, p. 1), o projeto físico é “a descrição completa do projeto a ser implementado. Nesta fase do projeto não poderá haver lacunas quanto ao que será desenvolvido”, ou seja, é uma implementação do projeto fundamentado pelo projeto lógico com a finalidade de produzir uma representação física, sendo a implementação do banco de dados. O projeto físico representa os dados por meio da estrutura de armazenamento em uso, neste caso, o relacional.

A representação dos objetos do mundo real do projeto EzMart encontra-se disponível no Apêndice S anexado a este documento.

## 6.5 PADRÕES, CONVENÇÕES E GUIAS

Um padrão é compreendido como uma forma genérica de se obter uma solução para problemas específicos.

Nesta seção são apresentados os padrões de projeto adotados para o desenvolvimento do sistema EzMart, bem como as convenções para a criação do respectivo código-fonte.

### 6.5.1 *Design Patterns*

*Design Patterns* ou padrões de projeto devem ser compreendidos como uma solução geral para um problema que ocorre com frequência dentro de um determinado contexto no projeto de *software*. Pressman (2011) afirma:

O intuito de cada padrão de projeto é fornecer uma descrição que permite a um projetista determinar se o padrão se aplica ou não ao trabalho em questão, se o padrão pode ou não ser reutilizado (e, portanto, poupando tempo) e se o padrão pode servir como um guia para desenvolver um padrão similar, mas funcional ou estruturalmente diferente. (PRESSMAN, 2011, p. 214).

Com os avanços na programação surgiram abordagens com objetivo de simplificar a programação e garantir como resultado um *software* de fácil manutenção não apenas para quem o desenvolveu, mas para todos os que futuramente irão trabalhar diretamente com este *software.*

Assim como Pressman, Castilho (2011, p. 3) afirma que as principais vantagens em se utilizar *Design Patterns* são:

1. vocabulário comum entre desenvolvedores: os desenvolvedores que conhecem padrões podem conversar “dando nome aos bois”, ou seja, ao invés de explicar a solução contando uma história (muitas vezes confusa), pode-se dizer “utilizei padrão X”;
2. facilidade na aplicação de técnicas e princípios de orientação a objetos, como herança, composição e polimorfismo;
3. refatoração de um código ruim, altamente acoplado e de baixa coesão;
4. todos esses itens colaboram para melhor facilidade de extensão e manutenção do *software*.

Para facilitara visualização dos padrões aplicados no projeto EzMart, o Quadro 6 demonstra-os.

|  |  |  |
| --- | --- | --- |
| **Tipos GoF** | **Padrões utilizados** | **Implementações do padrão** |
| Criacional | *Singleton* | ConnectionManager.java |
| Comportamental | Iterator | java.util |
| Outros padrões | *Model-View-Controller* (MVC)  *Data Access Object* (DAO)  *Criteria* | *Spring* MVC |

Quadro 6- Padrões de projeto

Fonte: Equipe (2018)

O Quadro 5 mostra os padrões do tipo GoF, que foram devidamente abordados no livro *Design Patterns* lançado em 199, dentre os 23 padrões descritos neste livro. Os 4 autores ficaram conhecidos como a *Gang of Four* (GoF). Os tipos de padrões GoF podem ser divididos em categorias, são elas:

1. criacional;
2. estrutural;
3. comportamental.

Entretanto, utiliza-se no projeto EzMart a categoria criacional e comportamental, porém existem outros padrões que não foram catalogados dentre os 23 padrões, mas devem ser considerados como um. São os exemplos citados na terceira linha do Quadro 5.

O sistema EzMart em sua arquitetura, utiliza o padrão MVC. Model-View-Controller (MVC) é um padrão concebido no Centro de Pesquisa da XEROX PARC (EUA), entre os anos de 1978-79. O padrão MVC é um *framework* que permite o desenvolvimento de um *software* estabelecendo um padrão de qualidade, proporcionando facilidade de manutenção, fator devidamente importante quando o *software* ultrapassa a sua etapa de implantação, ou seja, quando que por algum tempo está em produção sendo utilizado por diversas pessoas.

O padrão MVC possui três camadas, que são responsáveis por permitir uma alta coesão entre o que é desenvolvido, ou seja, permite que cada uma das camadas exerça somente uma funcionalidade e que estas camadas possam se comunicar, trocando as informações necessárias. As camadas do padrão MVC são:

1. Model é o responsável pela manipulação dos dados, faz a leitura, escrita e validação dos dados contidos na aplicação;
2. View é a camada de interação com o usuário. Ela é responsável por apresentar aos usuários as informações por meio de uma interface;
3. Controller é o responsável por receber e interpretar todas as requisições dos usuários, encaminhando-as em forma de comandos para cada parte responsável.

Além disso, as três camadas do padrão MVC podem estender-se em outras subcamadas que desempenham papéis específicos dentro de um sistema. Em se tratar de um trabalho científico, é de extrema importância que estas características sejam abordadas de forma documental, por isso, a seguir, a imagem ilustra o funcionamento das três camadas e as subcamadas.

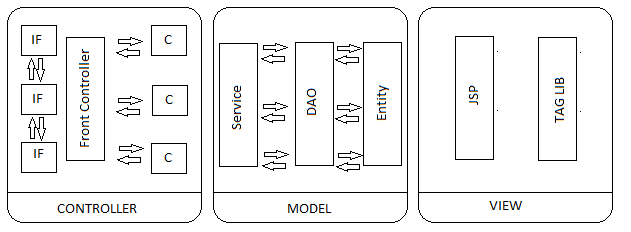


Figura 39- Padão MVC

Fonte: Equipe (2018)

Em termos técnicos, a primeira camada a receber quaisquer requisições dos usuários EzMart através de uma das interfaces, é a camada de *Controller.* As requisições do usuário passam por uma subcamada ainda dentro da camada de *Controler*, denominada *Intercept Filter (IF),* que faz as verificações necessárias no quesito de restrições em nível de acesso dos usuários, conhecido também como “cadeia de responsabilidade”. Além disso, o *Intercept Filter (IF)* também é considerado como um padrão não catalogado como GoF, porém, tão importante como os outros.

Conseguinte, o *Front Controller (FC),* localiza as classes de *Views*, ou melhor, as classes que contém a interface com o usuário, e comunica-se por fim com o *Controller*, que valida os passos e persistem as entidades da camada *Model* e comunica-se como DAO localizado no *Model*.

Após receber as requisições do usuário e passar pelas subcamadas do *Controller*, a camada seguinte é a *Model,* que tem como subcamadas: *Service*, DAO e *Entitys*. Os *Services* são os serviços prestados pela aplicação. O DAO (*Data Acess Object*) é quem recebe as especificações com os métodos e parâmetros da *Service* realizando a conexão com o Banco de Dados e ainda possibilita ao desenvolvedor alterar a forma de persistir os dados sem influenciar no restante do sistema. Por meio de objetos DAO, torna-se possível realizar operações de gravação, leitura, alteração, exclusão e pesquisa no banco de dados, e assim como os outros padrões, também é considerado um *design pattern* importante As *Entitys* são as entidades que prestam serviços para a aplicação e que guardam diversas informações pertinentes a ela de forma organizada.

Dentre o padrão MVC adotado para o desenvolvimento do sistema EzMart, também é adotado o padrão *Singleton*, que segundo Medeiros (2013, p. 2) “[...] tem como definição garantir que uma classe tenha apenas uma instância de si mesma e que forneça um ponto global de acesso a ela”. Em outras palavras, pode-se dizer que uma classe gerência sua própria e única instância, evitando que qualquer outra classe crie uma instância dela.

Em resumo, os padrões abordados nesta seção têm muitas vantagens, e essas justificam-se o uso, pois oferece uma solução para um problema de projeto que verdadeiramente funciona, além de favorecer o uso de um vocabulário comum entre os desenvolvedores, aumentando assim a capacidade de reuso do código.

### 6.5.2 Convenções e Guias para Codificação

As convenções de codificação definem um conjunto de regras que deve ser seguido durante a escrita do código fonte do sistema de *software*, tais regras têm por objetivo formatar o código, bem como auxiliar na interação da equipe de desenvolvimento. Seguindo essas regras, pode-se fazer com que o código se torne mais manutenível e com um melhor desempenho.

Diante disto, uma vez que a linguem de desenvolvimento do sistema é Java, foi utilizado a convenção *Java Code Conventions* fornecida pela *Sum MicroSystems*.

As especificações e convenções aplicadas no *Java Code Conventions* está relacionada com os seguintes elementos:

1. classes;
2. interfaces;
3. pacotes;
4. constantes;
5. métodos;
6. endentação;
7. atributos;
8. variáveis.

As classes e interfaces não devem ser nomeadas com verbos, ao invés disto, devem ser nomeadas com um ou mais substantivos. A forma pela qual este processo deve acontecer é por meio do método *Camel Case*, isto é, os substantivos devem estar juntos e a primeira letra de cada palavra deve ser maiúscula. Exemplo: “ListService.java”.

Os pacotes devem ser estruturados de acordo com a hierarquia em que os mesmos se encontram. Além disto, estes devem ser nomeados com letras minúsculas e separados por ponto. Exemplo: “br.com.ezmart.entity”.

Já os atributos e variáveis podem ser criados(as) utilizando uma ou mais palavras e convém que os mesmos sejam substantivos. Para a criação dos atributos e variáveis, é necessário que a primeira palavra comece com letra minúscula e as demais com letras maiúsculas. Exemplo: “private List userList”

Em relação aos métodos, convém que os mesmos sejam formados por verbos, ao serem criados, este processo deve acontecer da mesma forma que acontece com os atributos, isto é, a primeira palavra deve começar com letra minúscula e as demais com letras maiúsculas. Exemplo: “public List<E> readByCriteria()”

Em relação as constantes, é necessários que as sejam criadas utilizando letras maiúsculas, além disto, as mesmas devem ser separadas por *underline* ( \_ ). Exemplo: “public static Long NOME\_ILIKE”

A endentação diz respeito ao alinhamento do código, ou seja, a partir de um código alinhado e formatado, a criação, manutenção e abstração torna-se mais simples e organizada Sendo assim, é necessário que o código esteja alinhado quatro espaços do teclado (espaço correspondente à tecla *tab*).

## 6.6 ANÁLISE DE COMPLEXIDADE

## 6.7 PROJETO DE SISTEMAS DISTRIBUÍDOS

Coulouris, Dollimore e Kindberg (2007) em seu livro “Sistemas distribuídos – Conceitos e projeto” mostram de maneira clara o objetivo de um sistema distribuído, e propõem de maneira completa a caracterização de um projeto de sistemas distribuídos. Desta maneira, é possível projetar um sistema distribuído referente ao embasamento proposto por Coulouris, Dollimore e Kindberg (2007) em seu livro.

Coulouris, Dollimore e Kindberg (2007, p. 15) apresentam o projeto de sistemas distribuídos como sendo aquele no qual os componentes são localizados em computadores interligados em rede, e que se comunicam e coordenam suas ações apenas passando mensagens.

Desta maneira, compreende-se que o sistema EzMart é um sistema distribuído, pois todas as informações em fase de processamento são distribuídas por vários computadores, em vez de ficarem confinadas a uma única máquina, visto que há muitas partes interessadas neste projeto, e que de acordo com a sua finalidade e viabilidade deve ser utilizado por inúmeros usuários, que acessarão de diversas regiões geográficas para obter os mesmos serviços. De acordo com as afirmações de Coulouris, Dollimore e Kindberg (2007), nota-se a presença de características que identificam o projeto EzMart como sendo um sistema distribuído, por exemplo: há o compartilhamento de recursos; vários processos podem operar ao mesmo tempo em diferentes computadores na rede; a capacidade do sistema EzMart pode ser aumentada por acréscimo de novos recursos, e o usuário não sabe necessariamente da natureza distribuída do sistema.

O desafio para os projetistas do sistema EzMart está em projetar um *software* e um *hardware* para que ele forneça as características desejáveis dos sistemas distribuídos e ao mesmo tempo, minimizem os problemas inerentes a estes sistemas.

Portanto, esta seção tem como objetivo a apresentação dos procedimentos para tratamentos dos desafios de um sistema distribuído, onde é imprescindível a análise destes desafios para o projeto EzMart, e as tecnologias e arquiteturas de distribuição conforme é proposto por Coulouris, Dollimore e Kindberg (2007).

### 6.7.1 Procedimentos para Tratamento dos Desafios

Os desafios da construção de sistemas distribuídos são apresentados nesta seção, inclusive obtendo o foco necessário em todos os métodos para tratar cada um dos casos e como foram aplicados em especial no sistema EzMart.

Coulouris, Dollimore e Kindberg (2007) afirmam que muitos são os problemas que surgem em um sistema distribuído, advindo da abrangência que este tem em seu projeto. Desta maneira, todos os obstáculos e dificuldades encontrados na concepção de um sistema distribuído podem ser chamados de desafios. Os desafios de um sistema distribuído devem ser tratados para que o grau de qualidade esperado do sistema seja alcançado.

Entretanto, Coulouris, Dollimore e Kindberg (2007) afirmam que também é possível vencer estes desafios, e projetar de maneira simples sistema distribuído, porém se torna imprescindível conhecê-los e tratá-los, mesmo que muitos destes desafios tenham sido resolvidos.

6.7.1.1 Heterogeneidade

Coulouris, Dollimore e Kindberg (2007) afirmam que a Internet permite aos usuários acessarem serviços e executarem aplicativos por meio de um conjunto heterogêneo de computadores e redes, e complementam explicando que a heterogeneidade é nada mais do que a variedade ou diferença que pode ser aplicada em diversos aspectos: redes, *hardware,* sistemas operacionais, linguagens de programação e implementações de diferentes desenvolvedores.

Segundo Coulouris, Dollimore e Kindberg (2007, p. 28):

Embora a Internet seja composta de muitos tipos de redes, suas diferenças são mascaradas pelo fato de que todos os computadores ligados a elas utilizam protocolos Internet para se comunicar. Por exemplo, um computador que possui uma placa de Ethernet tem uma implementação dos protocolos Internet enquanto um computador em um tipo diferente de rede tem uma implementação dos protocolos Internet para essa rede.

Em relação aos tipos de dados trocados entre *hardware*s diferentes, Coulouris, Dollimore e Kindberg (2007), afirmam que podem ser apresentados de diversas maneiras em diferentes tipos de *hardware,* e essas diferenças na apresentação devem ser consideradas, caso mensagens sejam trocadas entre programas sendo executados por diferentes *hardwares.*

Embora os sistemas operacionais de todos os computadores na Internet precisem incluir uma implementação dos protocolos de Internet, nem todos fornecem a mesma interface de programação de aplicativos para estes protocolos, afirmam Coulouris, Dollimore e Kindberg (2007).

Eles também afirmam que as diferentes linguagens de programação usam diferentes representações para caracteres e estruturas de dados, como *arrays* e registros. Coulouris, Dollimore e Kindberg (2007) anunciam que estas diferenças devem ser levadas em consideração, caso programas escritos em diferentes linguagens precisem se comunicar. No entanto, os programas escritos por diferentes desenvolvedores não podem se comunicar, a menos que estes utilizem padrões comuns, como por exemplo, realizar a comunicação via rede utilizando a mesma representação de dados primitivos e estruturas de dados nas mensagens. Para que isso ocorra, Coulouris, Dollimore e Kindberg (2007), estabelecem padrões que precisam ser adotados.

Portanto, Coulouris, Dollimore e Kindberg (2007, p.29), afirmam que a utilização de *middleware* é uma forma de solucionar a heterogeneidade.

Além de resolver os problemas de heterogeneidade, o *middleware* fornece um modelo computacional uniforme para ser usado pelos programadores de serviços e de aplicativos distribuídos. Os modelos possíveis incluem a invocação remota de objetos, a notificação remota de eventos, o acesso remoto a banco de dados e o processamento de transação distribuído. Por exemplo, o CORBA fornece invocação remota de objetos, a qual permite que um objeto, em um programa sendo executado em um computador, invoque um método de um objeto em um programa executado em outro computador. Sua implementação oculta o fato de que apenas mensagens passam por uma rede para enviar o pedido de invocação e sua resposta.

Coulouris, Dollimore e Kindberg (2007, p.29), propõem ainda, outra maneira para que seja possível vencer a heterogeneidade de um sistema distribuído, a migração de código ou ainda código móvel.

O termo migração de código, ou ainda, código móvel, é usado para referir ao código que pode ser enviado de um computador para outro a ser executado no destino – os *applets* JAVA são um exemplo. [...]

A estratégia de máquina virtual oferece uma maneira de tornar um código executável em qualquer tipo de processador (*hardware*) ou sistema operacional. Nessa estratégia, o compilador de uma linguagem em particular gera código para uma máquina virtual, em vez de código para um processador e sistema operacional específicos: por exemplo, o compilador JAVA produz código para a máquina virtual JAVA (*JAVA Virtual Machine – JVM*). Para permitir a execução de programas JAVA, é necessário ter-se uma versão da JVM implementada para o processador e sistema operacional da máquina alvo.

Desta maneira, para que o desafio da heterogeneidade seja tratado de forma correta no sistema EzMart, pode ser executado em diferentes sistemas operacionais que possuem uma *JAVA Virtual Machine* compatível. A *JAVA Virtual Machine* possibilita a portabilidade do código. O que for escrito em um sistema operacional *Windows*, irá rodar em um sistema operacional *Linux*. Esse processo cria uma independência do Sistema Operacional, dando ao desenvolvedor uma liberdade de desenvolver para múltiplas plataformas sem a preocupação de se o código irá funcionar corretamente.

Uma outra tratativa para este desafio, é a arquitetura REST (*Representational State Transfer*), em português Transferência de Estado Representacional que consiste de regras e princípios que, quando seguidas, permitem a criação de um projeto com interfaces bem definidas. Essa arquitetura REST, portanto, é utilizada para dividir as responsabilidades do sistema entre clientes e servidores. Além disto, utiliza também do protocolo HTTP para vencer o desafio da heterogeneidade do sistema.

6.7.1.2 Sistemas abertos

Coulouris, Dollimore e Kindberg (2007) definem um sistema aberto como tendo a capacidade de ser estendido e reimplementado de várias maneiras. O fato de um sistema distribuído ser ou não aberto é determinado principalmente pelo grau com que os novos serviços podem ser adicionados e disponibilizados para o uso por uma variedade de programas clientes.

Coulouris, Dollimore e Kindberg (2007, p. 29) afirmam que:

A característica de um sistema aberto é obtida a partir do momento em que a especificação e a documentação das principais interfaces de software dos componentes de um sistema estão disponíveis para os desenvolvedores do *software*. Em uma palavra, as principais interfaces são publicadas. Esse processo é similar àquele realizado por organizações de padronização, porém, frequentemente, ignora os procedimentos oficiais, os quais normalmente são pesados e lentos.

Entretanto, a publicação de interfaces é apenas o ponto de partida para adicionar e estender serviços em um sistema distribuído. O maior desafio dos projetistas, afirmam Coulouris, Dollimore e Kindberg (2007) é encarar a complexidade de sistemas distribuídos compostos por muitos componentes e elaborados por diferentes pessoas.

Portanto, os sistemas projetados a partir de padrões públicos são chamados de sistemas distribuídos abertos, para reforçar o fato de que são extensíveis. Eles podem ser ampliados em nível de *hardware,* pela adição de computadores em uma rede, e nível de *software,* pela introdução de novos serviços ou pela reimplantação dos antigos, permitindo que os programas aplicativos compartilhem recursos. Coulouris, Dollimore e Kindberg (2007) afirmam que uma vantagem adicional, frequentemente mencionada paro sistemas abertos, é sua independência de fornecedores individuais.

Desta maneira, para que o desafio da abertura seja solucionado no projeto EzMart, é importante que sejam conhecidas as interfaces dos novos componentes através da publicação e da sua documentação, e uma das maneiras é utilizar formatos padrões para a publicação. Esse formato padrão consiste na documentação conhecida como *Swagger.*

Fazer uma boa documentação de uma API REST, é fundamental para que desenvolvedores, testadores e principalmente o usuário final tenham um entendimento claro do comportamento oferecido pelo serviço EzMart. O *Swagger* é uma ferramenta que permite criar a documentação para a API de três maneiras:

1. manualmente: Permite-se escrever manualmente a especificação dos serviço se publicá-los no seu próprio servidor ou em algum outro;
2. automaticamente: Permite-se que ao mesmo tempo que é criado a API, também seja gerado a documentação;
3. codegen: É um aplicativo que converte as anotações do próprio Swagger contidas no código fonte da API REST em documentação.

É esta tratativa que o sistema EzMart tem para que o problema da Abertura seja solucionado. Assim, os terceiros que acessam os serviços EzMart, possuem em sua posse a documentação destes serviços, como por exemplo, todos os métodos, como GET, POST, DELETE, e qual sua responsabilidade mediante à cada serviço.

6.7.1.3 Segurança

Muitos recursos, informações, e dados pessoais que se tornam disponíveis e são mantidos em sistemas distribuídos têm um alto valor para os usuários. Portanto Coulouris, Dollimore e Kindberg (2007) afirmam que a segurança é um considerável fator e que possue três componentes: confidencialidade (proteção contra pessoas não autorizadas), integridade (proteção contra alteração ou dano) e disponibilidade (proteção contra interferência com os meios de acesso aos recursos). Outro componente se trata da autenticação que tem o objetivo de identificar o usuário, ou seja, fornecer-lhe uma identidade, possibilitando, portanto, o serviço de autorização mediante à determinadas operações.

Coulouris, Dollimore e Kindberg (2007, p. 30) afirma que embora a Internet ou o *firewall* possam ser utilizados como mecanismos de segurança, não são suficientes para tratar o desafio de segurança de um projeto de sistemas distribuídos.

Embora a Internet permita que um programa em um computador se comunique com um programa em outro computador, independentemente de sua localização, existem riscos de segurança associados ao livre acesso a todos os recursos em uma Internet. Embora um *firewall* possa ser usado para formar uma barreira em torno de uma Intranet, restringindo o tráfego que pode entrar ou sair, isso não garante o uso apropriado dos recursos pelos usuários de dentro da intranet, nem o uso apropriado dos recursos da Internet, que não são protegidos por *firewalls.*

Em um projeto de sistemas distribuídos, portanto, o desafio é evitar que informações sigilosas em uma ou mais mensagens trafegue pela rede sem que sejam utilizados mecanismos de segurança, ou que ainda, as identidades de quem troca as mensagens sejam expostas sem que haja mecanismos de segurança que as protejam. Outro desafio consiste em identificar corretamente um usuário ou outro agente remoto. Ambos os desafios podem ser resolvidos com o uso de técnicas como criptografia, que são desenvolvidas para este propósito.

Entretanto, Coulouris, Dollimore e Kindberg (2007), explicam que há ainda, desafios de segurança que não foram realmente resolvidos, como o ataque de negação de serviço (*Denial of service*) que ocorre quando um usuário interrompe um serviço com um número alto de pedidos sem sentido, porém medidas para se opor a isto baseadas no gerenciamento das redes estão sendo desenvolvidas. Outro desafio ainda não resolvido é a segurança de código móvel, que precisa ser manipulado com cuidado, pois há possíveis efeitos da execução do programa que não são previstos.

No sistema EzMart, para tratar do desafio de segurança é utilizado para realizar a troca de dados na aplicação de forma segura, uma camada adicional de segurança, chamada *Hyper Text Transfer Protocol Secure* (HTTPS), que criptografa as mensagens.

Assim como a internet e toda a tecnologia a criptografia também evoluiu, sendo aplicada a diversas situações, se adaptando e tomando vertentes variadas. A criptografia possui dois tipos, a simétrica e a assimétrica. O SSL/TLS utiliza ambos: assimétrica para assegurar a autenticidade, e simétrica para a confidencialidade e integridade.

6.7.1.4 Escalabilidade

Os sistemas distribuídos devem funcionar de forma efetiva e eficaz em muitas escalas diferentes. Um sistema é descrito como escalável segundo os autores Coulouris, Dollimore e Kindberg (2007) se permanece eficiente quando há um aumento significativo no número de recursos e no número de usuários. A Figura 39 mostra o aumento no número de computadores na Internet durante 24 anos, até 2003, conforme trata Coulouris, Dollimore e Kindberg (2007), afirmando que a Internet é um exemplo de sistema distribuído. Já a Figura 40 mostra um número cada vez maior de computadores e servidores durante os 10 anos de história da *web*. Isso mostra a tendência até o ano atual, o que deve ser levado em consideração.

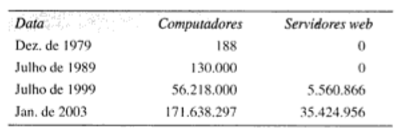


Figura 40- Computadores registrados na Internet

Fonte: Coulouris, Dollimore e Kindberg (2007, p. 31)

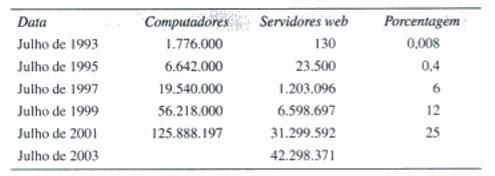


Figura 41-Computadores e servidores na Internet

Fonte: Coulouris, Dollimore e Kindberg (2007, p. 32)

Em complemento às afirmações de Coulouris, Dollimore e Kindberg (2007), a *International Telecommunication Union* (2016, p. 1) também afirma que 54% da população mundial têm acesso à Internet e 95% vivem em áreas com coberturas de rede móvel. No Brasil, segundo dados do Centro Regional de Estudos para o Desenvolvimento da Sociedade da Informação (Cetic.br), em 2015, 51% da população brasileira teve acesso à Internet. Mediante aos dados expostos por Coulouris, Dollimore e Kindberg (2007) e em comparação com os dados da União Internacional de Telecomunicações, nota-se que há uma tendência significativa no aumento da população em relação ao acesso à Internet e aos sistemas *web.* E por sua vez, estes sistemas precisam suportar cada vez mais o aumento de recursos e usuários, para isso os desenvolvedores devem garantir que este sistema seja escalável.

Segundo Coulouris, Dollimore e Kindberg (2007, p. 32) são apresentados os seguintes desafios referentes à escalabilidade.

1. controlar o custo aos recursos físicos: á medida que a demanda por recurso aumenta, deve ser possível, a um custo razoável, ampliar o sistema para atende-la.[...] Em geral, para que um sistema com n usuários seja escalável, a quantidade de recursos físicos seja exigida para suportá-los deve ser no máximo O(n) – isto é proporcional a n;
2. controlar a perda de desempenho: considere o gerenciamento de um conjunto de dados, cujo tamanho é proporcional ao número de usuários ou recursos presentes no sistema. [...] Os algoritmos que utilizam estruturas hierárquicas têm melhor escalabilidade do que aqueles que usam estruturas lineares. Mas mesmo com estruturas hierárquicas, um aumento no tamanho, resultará em alguma perda de desempenho: o tempo que leva para acessar dados hierarquicamente estruturados é O(log n), onde n é o conjunto de dados. Para que um sistema seja escalável, a perda de desempenho máxima não deve ser maior do que isso;
3. impedir que os recursos de software se esgotem: um exemplo de falta de escalabilidade é mostrada pelos números usados como endereços IP. [...] É difícil prever, com anos de antecedência, a demanda que será imposta sobre um sistema. Além disso superestimar o crescimento futuro pode ser pior do que adaptar para uma mudança quando formos obrigados a isso – por exemplo, endereços IP maiores ocupam espaço extra no armazenamento de mensagens e no computador;
4. evitar gargalos de desempenho: em geral os algoritmos devem ser descentralizados para evitar a existência de gargalos de desempenho. Ilustramos esse ponto como referência ao predecessor do *Domain Name System,* no qual a tabela de correspondência entre endereços IP e nomes era mantidas em um único arquivo central, cujo *download* podia ser feito em qualquer computador que precisasse dela. [...] O *Domain Name System* eliminou esse gargalo particionando a tabela de correspondência de nomes entre diversos servidores localizados em toda a Internet e administrados de forma local. Alguns recursos compartilhados são acessados com muita frequência, por exemplo, muitos usuários podem acessar a mesma página *web* causando perda no desempenho.

Portanto, é necessário que o sistema não mude quando a escala do sistema aumentar, entretanto isso é difícil de conseguir segundo Coulouris, Dollimore e Kindberg (2007).

No sistema EzMart não existem ajustes para tratar da escalabilidade, no entanto caso seja necessário, podem ser utilizadas as características escaláveis do processo servidor de aplicações denominado *Apache Tomcat*. Apesar disso, caso seja necessário, pode-se hospedar todo ou grande parte do serviço em nuvem, podendo aumentar a quantidade de recursos conforme o aumento de usuários.

6.7.1.5 Tratamento de falhas

Segundo Coulouris, Dollimore e Kindberg (2007), os sistemas de computador falham. Quando ocorrem falhas no *hardware* ou no *software* os programas podem produzir resultados incorretos, ou podem parar antes de terem concluído a computação pretendida. Ainda, afirmam que as falhas podem ser parciais, isto é, apenas alguns componentes falham, enquanto outros continuam funcionando. Portanto, o tratamento de falhas é particularmente difícil. As técnicas a seguir são discutidas por todo o livro de Coulouris, Dollimore e Kindberg (2007), porém isso é tratado com mais ênfase, na página 33.

1. detecção de falhas: algumas falhas podem ser detectadas. Por exemplo, somas de verificação podem ser usadas para detectar dados danificados em uma mensagem ou em um arquivo. [...] O desafio é de como gerenciar a ocorrência de falhas que não podem ser detectadas, mas que podem ser suspeitas;
2. mascaramento de falhas: algumas falhas podem ser ocultas ou se tornar menos sérias. Dois exemplos de ocultação de falhas: mensagens podem ser retransmitidas quando não chegam; dados de arquivos podem ser gravados em dois discos, para que, se um estiver danificado, o outro ainda possa estar correto. Simplesmente eliminar uma mensagem danificada é um exemplo de como tornar uma falha menos séria;
3. tolerância a falhas: a maioria dos serviços na Internet apresenta falhas – não seria prático para eles tentar detectar e mascarar tudo que possa ocorrer em uma rede grande assim, com tantos componentes. Seus clientes podem ser projetados de forma a tolerar falhas, o que geralmente envolve a tolerância também por parte dos usuários;
4. recuperação de falhas: a recuperação envolve projetar software de modo que o estado dos dados permanentes possa ser recuperado ou retrocedido, após a falha de um servidor. Em geral, as computações realizadas, por alguns programas ficarão incompletas quando ocorrer uma falha e os dados permanentes que eles atualizam (arquivos em disco) podem não estar em um estado consistente;
5. redundância: os serviços podem se tornar tolerantes a falhas com o uso de componentes redundantes. Por exemplo: sempre deve haver pelo menos duas rotas diferentes entre dois roteadores quaisquer na Internet, no *Domain Name System,* toda tabela de correspondência de nomes é replicada em pelo menos dois servidores diferentes, e um banco de dados pode ser replicado em vários servidores, para garantir que os dados permaneçam acessíveis após a falha de qualquer servidor. Os servidores podem ser projetados de forma a detectar falhas em seus pares; quando uma falha é detectada em um servidor, os clientes são redirecionados para os servidores restantes.

O projeto de técnicas eficazes para manter réplicas atualizadas de dados que mudam rapidamente, sem perda de desempenho excessiva, é um desafio.

Os sistemas distribuídos fornecem um alto grau de disponibilidade perante falhas de *hardware.* A disponibilidade de um sistema é a medida da proporção de tempo em que ele está pronto para uso segundo Coulouris, Dollimore e Kindberg (2007). Quando um dos componentes de um sistema falha, apenas o trabalho que estava usando o componente defeituoso é afetado. Portanto, um usuário pode passar para outro computador, caso aquele que estava usando falhe.

Como o código do sistema EzMart possui tratamento de exceções e já existem medidas de tratamento de falhas no servidor de aplicação *Tomcat* e no *Postgres,* não há maiores necessidades, em que haja relevância para a adoção de técnicas mais elaboradas para essa tratativa.

6.7.1.6 Concorrência

Todos os serviços como os aplicativos fornecem recursos que podem ser compartilhados pelos clientes em um sistema distribuído. Coulouris, Dollimore e Kindberg (2007) afirmam que existe a possibilidade de que vários clientes tentem acessar um recurso compartilhado ao mesmo tempo. Coulouris, Dollimore e Kindberg (2007, p. 34) afirmam que “para que um objeto mantenha coerência em um ambiente concorrente, suas operações devem ser sincronizadas de maneira que seus dados sempre permaneçam consistentes. ”. Isso pode ser obtido através de técnicas padrão, como semáforos, que são disponíveis na maioria dos sistemas operacionais.

Para vencer o desafio de concorrência e controlar o número de conexões no sistema EzMart são utilizados o servidor *Apache Tomcat* e o *pull* de conexões, ou seja, um conjunto de conexões, que são responsáveis por controlar o acesso simultâneo dos usuários no sistema. No mais, utiliza-se o servidor na nuvem para aumentar a escalabilidade *Web Service* do sistema EzMart.

6.7.1.7 Transparência

Coulouris, Dollimore e Kindberg (2007) definem a transparência como sendo a ocultação, para um usuário final ou para um programador de aplicativos, da separação dos componentes em um sistema distribuído de modo que o sistema seja percebido como um todo, em vez de uma coleção de componentes independentes. As implicações da transparência têm grande interferência sobre o projeto do sistema. Coulouris, Dollimore e Kindberg (2007, p. 34) ainda afirmam que o ANSA (*Reference Manual*) e o RM-ODP (*Reference Model Of Open Distributed Processing*) da ISO (*International Organization for Standardization*) identificam oito formas de transparência:

1. transparência de acesso permite que recursos locais e remotos sejam acessados com o uso de operações idênticas;
2. transparência de localização permite que os recursos sejam acessados sem conhecimento de sua localização física ou na rede;
3. transparência de replicação permite que várias instâncias dos recursos sejam usadas para aumentar a confiabilidade e o desempenho, sem conhecimento das réplicas por parte do usuário ou dos programadores do aplicativo;
4. transparência de falhas permite a ocultação de falhas, possibilitando que os usuários e programas aplicativos concluam suas tarefas a despeito da falha de componentes de *hardware* ou *software;*
5. transparência de mobilidade permite a movimentação de recursos e clientes dentro de um sistema, sem afetar a operação de usuários ou programas;
6. transparência de desempenho permite que o sistema seja reconfigurado para melhorar o desempenho à medida que as cargas variam;
7. transparência de escalabilidade permite que os sistemas e aplicativos se expandam em escala sem alterar a estrutura do sistema ou os algoritmos do aplicativo.

Dentre todos os tipos de transparência, Coulouris, Dollimore e Kindberg (2007) afirmam que as mais importantes são a de acesso e a de localização, pois sua presença ou ausência afeta fortemente a utilização de recursos distribuídos.

Compreende-se, portanto, que os sistemas distribuídos estão em toda parte. A Internet permite que os usuários de todo mundo acessem seus serviços onde quer que estejam. Cada organização gerencia uma Intranet, a qual fornece serviços locais e serviços Internet para usuários locais e remotos. Sistemas distribuídos de pequena escala podem ser construídos a partir de computadores móveis ou outros dispositivos computacionais interligados através de redes sem fio, afirmam Coulouris, Dollimore e Kindberg (2007).

Desta maneira, identifica-se que o compartilhamento de recursos é o principal fator de motivação para a construção de sistemas distribuídos. Todos os recursos disponíveis como, impressoras, arquivos ou páginas na Internet, são gerenciados por servidores de tipo apropriado e acessados por clientes específicos, e todo este projeto passa por inúmeros desafios, que foram tratados nesta seção. Coulouris, Dollimore e Kindberg (2007), apresentam inúmeros exemplos de projetos de sistemas distribuídos e a forma com que devem lidar com os desafios descritos por ele. No entanto, esta seção fornece também uma visão específica a respeito de cada tratamento de desafios referente ao sistema EzMart.

O sistema EzMart, para atender as necessidades dos usuários, realiza diversas operações no servidor de aplicação, banco de dados e *browser* sem o conhecimento do usuário, também possui processos que são executados concorrentemente utilizando recursos compartilhados sem sofrerem interferência entre eles, assegurando assim, um bom nível de transparência do sistema.

### 6.7.2 Tecnologias e Arquiteturas de Distribuição

Um modelo de arquitetura de um sistema distribuído envolve o posicionamento de suas partes e os relacionamentos entre elas. Coulouris, Dollimore e Kindberg (2007, p. 39) descrevem que um modelo de arquitetura “define a forma pela qual os componentes dos sistemas interagem e a maneira pela qual eles são mapeados em uma rede de computadores subjacentes. ”.

Desta maneira, passam a definir a arquitetura geral de um sistema, a partir do modelo de arquitetura. Coulouris, Dollimore e Kindberg (2007) definem que “a arquitetura de um sistema é a estrutura em termos de componentes especificados separadamente”, e apresentam um objetivo global para a arquitetura de um sistema distribuído. O objetivo global é garantir que a estrutura atenda as demandas atuais, e provavelmente, as futuras impostas sobre ela. Ainda, Coulouris, Dollimore e Kindberg (2007, p. 40) afirmam que há preocupações que rodeiam o projeto de arquitetura do sistema.

As maiores preocupações são tornar o sistema confiável, gerenciável, adaptável e rentável. O projeto arquitetônico de um prédio tem aspectos similares- ele determina não apenas sua aparência, mas também sua estrutura em geral e seu estilo arquitetônico (gótico, neoclássico, moderno) fornecendo um padrão de referência consistente para o seu projeto.

Desta maneira, é necessária que além da arquitetura física do sistema distribuído, seja projetado a arquitetura geral do sistema, assim como o estilo predominante neste sistema. Assim, é também possível construir o modelo de arquitetura do sistema distribuído EzMart em torno dos conceitos de processo e objeto apresentados por Coulouris, Dollimore e Kindberg (2007).

O modelo de arquitetura de um sistema distribuído primeiramente simplifica e abstrai as funções dos componentes individuais de um sistema distribuído e em seguida considera importantes pontos como abordado por Coulouris, Dollimore e Kindberg (2007).

O posicionamento dos componentes em uma rede de computadores, e os inter-relacionamentos entre estes componentes são as principais abstrações iniciais em um projeto de sistemas distribuídos. Coulouris, Dollimore e Kindberg (2007, p. 40) afirmam que “uma simplificação inicial é obtida por meio da classificação dos processos como processos servidores, processos clientes e processos *peer-to-peer -* sendo estes últimos os processos que colaboram e comunicam de maneira simétrica para realizar uma tarefa. ”. Essa classificação de processos identifica o impacto das falhas em cada um deles, como tratado na seção anterior. Resultantes desta análise, os pontos obtidos podem ser usados para especificar o posicionamento dos processos de uma forma que atenda os objetivos de desempenho e confiabilidade do sistema distribuído, conforme Coulouris, Dollimore e Kindberg (2007) descrevem.

Entretanto, existe a possibilidade de que um sistema possa ser dinâmico, e sua construção tem variações do modelo cliente-servidor, especificadas por Coulouris, Dollimore e Kindberg (2007, p. 40):

A possibilidade de mover o código de um processo para outro permite que um processo delegue tarefas: por exemplo, os clientes podem fazer o *download* de código a partir de servidores e executá-los localmente. Essa possibilidade reduz os atrasos de acesso a ele e minimiza o tráfego de comunicação.

Alguns sistemas distribuídos são projetados de forma que computadores e outros dispositivos móveis sejam adicionados ou removidos de forma transparente, permitindo que esses descubram os serviços disponíveis, ou que ofereçam seus serviços aos demais.

Desta forma, existem vários padrões que foram utilizados por Coulouris, Dollimore e Kindberg (2007) para a divisão de tarefas em um sistema distribuído, e que tem impacto direto sobre o desempenho e a eficiência do sistema resultante, neste caso o EzMart. Os modelos de arquitetura aqui descritos conforme Coulouris, Dollimore e Kindberg (2007) oferecem uma visão simplificada dos padrões mais importantes.

Coulouris, Dollimore e Kindberg (2007, p. 42) afirmam que “a divisão de responsabilidades entre os componentes de um sistema (aplicativos, servidores e outros processos) e a atribuição destes nos diversos computadores de uma rede, talvez seja o aspecto mais evidente do projeto de sistemas distribuídos. ”.

Em um sistema distribuído, os processos possuem responsabilidades bem definidas, e eles variam de acordo como modelo de arquitetura deste sistema. Por sua vez, Coulouris, Dollimore e Kindberg (2007) apresentam diversos modelos arquiteturais, por exemplo: cliente-servidor, *peer-to-peer,* servidores *proxies* e *caches*, código móvel, agentes móveis, computadores de rede e clientes “leves”. Porém o modelo arquitetural utilizado pelo sistema EzMart destaca-se por ser um dos mais conhecidos e utilizados por os sistemas distribuídos, e é o modelo cliente-servidor.

Coulouris, Dollimore e Kindberg (2007, p.42) afirmam que a arquitetura cliente-servidor é a mais citada quando os sistemas distribuídos são discutidos e historicamente é a mais importante e continua sendo amplamente utilizada. A Figura 41 ilustra essa arquitetura, na qual os processos clientes interagem com os processos servidores, localizados em distintos computadores hospedeiros para acessar os recursos compartilhados que eles gerenciam, segundo Coulouris, Dollimore e Kindberg (2007).

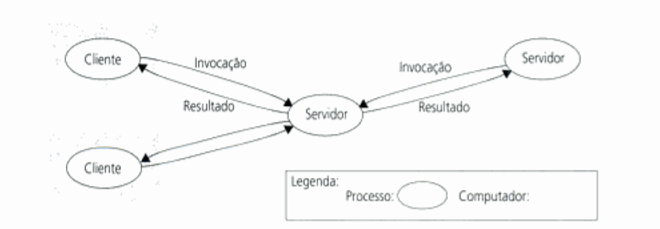


Figura 42- Exemplo da arquitetura cliente-servidor

Fonte: Coulouris, Dollimore e Kindberg (2007, p.42)

Os servidores podem ser clientes de outros servidores, conforme a Figura 63 indica. Coulouris, Dollimore e Kindberg (2007) exemplificam um servidor *web* é frequentemente um cliente de um servidor de arquivos local que gerencia os arquivos nos quais as páginas *web* estão armazenadas. Complementa, descrevendo que estes servidores e a maioria dos outros serviços da Internet, são clientes dos serviços DNS, que mapeia nomes de domínio Internet a endereços de rede IP. Outro exemplo é sobre os mecanismos de busca, as quais permitem os usuários pesquisar resumos de informações disponíveis em páginas na Internet. Coulouris, Dollimore e Kindberg (2007), afirmam que esses resumos são feitos por programas chamados *web crawlers* que são executados em segundo plano em um site de mecanismo de busca, usando pedidos HTT para acessar servidores *web* em toda Internet. Assim um mecanismo de busca pode ser tanto um servidor, quanto um cliente, respondendo às consultas de clientes e executando *web crawlers* que atuam como clientes de outros servidores. Neste caso, as tarefas do servidor e as tarefas dos *web crawlers* são totalmente independentes: há pouca necessidade de sincronizá-la e elas podem ser executadas concomitantemente.

Mediante a importantes notações referidas por Coulouris, Dollimore e Kindberg (2007), o sistema EzMart adota essa arquitetura pois todos os serviços, tarefas e processamentos que são divididos entre os servidores e os clientes acessam esses serviços mediante protocolos de rede. No lado servidor, é feito o atendimento de serviços *web*, enquanto o cliente faz as requisições por meio do protocolo HTTP.

A Figura 42 demonstra o diagrama de redes do sistema EzMart.

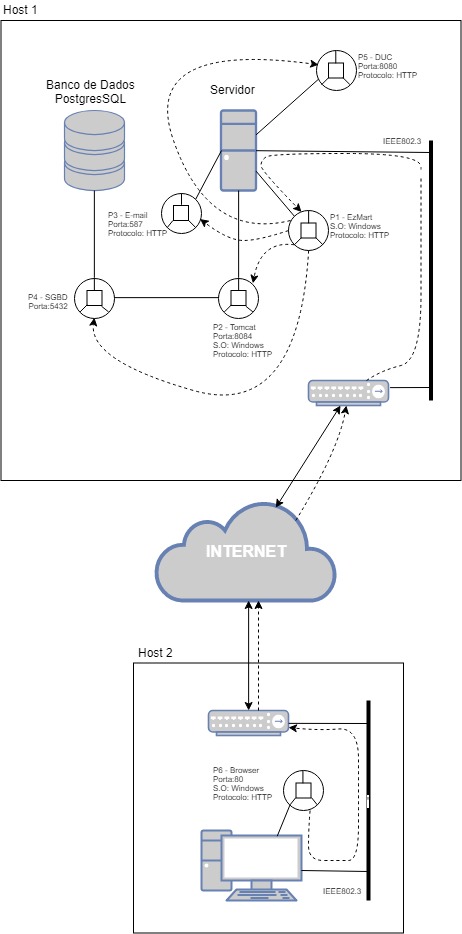


Figura 43- Diagrama de sistemas distribuídos

Fonte: Elaborado pela equipe (2018)

Em detalhes, o usuário se comunica com o sistema EzMart através do processo P6 por meio do *host* 2. A comunicação do usuário faz-se a partir de todas as funcionalidades que o sistema dispõe ao usuário através de sua interface. Todas as informações que são trocadas entre o usuário e o sistema EzMArt trafegam na rede de Internet, por meio do protocolo HTTP e estão armazenadas em uma máquina servidora, compreendido na Figura 64 como sendo o servidor. Por sua vez, o servidor contém os processos P1, P2, P3, P4 e P5. O processo P1 refere-se à aplicação EzMart em sua totalidade, contendo todas as funcionalidades e serviços que podem ser oferecidos aos usuários. O processo P2 refere-se ao servidor Tomcat que é servidor *web* Java, mais especificamente, um *container* de *servlets*. O Tomcat implementa, dentre outras de menor relevância, as tecnologias Java *Servlet* e Java *Server* *Pages* (JSP). Já o processo P3 é o servidor de *e-mail,* que é responsável por envio de *e-mails* para os usuários, por exemplo, confirmação de cadastro, alteração de senha, recebimento de mensagens de promoções, etc. O processo P4 é o SGBD, que gerencia o banco de dados da aplicação e tem como objetivo retirar da aplicação cliente a responsabilidade de gerenciar o acesso, a persistência, a manipulação e a organização dos dados. O SGBD disponibiliza uma interface para que seus clientes possam incluir, alterar ou consultar dados previamente armazenados. Por fim, o processo P6 é o processo responsável pelo DUC, que significa *Dynamic DNS Update Client,* responsável por verificar continuamente mudanças de endereço IP em segundo plano e atualiza automaticamente o DNS no NO-IP, sempre que houver mudanças.

## 6.8 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR

Os pesquisadores da área de Interação Humano-Computador (IHC) se empenham em estudar os fenômenos da comunicação entre pessoas e sistemas computacionais. O projeto de interação humano-computador inclui o projeto das interfaces do usuário, pois elas são um meio de comunicação entre o ser humano e o computador.

De acordo com Pressman (2011, p. 287), esse projeto identifica um *layout* de tela que forma a base para um protótipo de interface do usuário. Como consequência, o projeto de interfaces acrescenta uma medida qualitativa da facilidade e eficiência com a qual um ser humano consegue explorar as funções e recursos oferecidos pelo sistema. Tal medida qualitativa denomina-se usabilidade. Assim, com base no perfil dos usuários, o sistema deve oferecer uma interface que seja compreensível e agradável, propiciando uma boa experiência de uso.

Esta seção está dividida em três subseções, sendo elas: perfis de usuário, aspectos visuais da interface de usuário e heurísticas de usabilidade.

### 6.8.1 Perfil de Usuário

Observando o perfil dos possíveis usuários do EzMart, tendo em vista o planejamento de colocar o sistema em funcionamento e disponibilizar o acesso aos usuários, identificam-se conjuntos de usuário com diferentes perfis. A seguir são demonstrados três perfis que ilustram os diferentes tipos de usuários e suas características específicas.

1. jovens e adultos:

Faixa etária: 16 a 50 anos.

Sexo: indiferente.

Características: devido ao possível domínio de tecnologias dos navegadores web pelos usuários dessa faixa etária, pode-se observar que eles esperam que o sistema seja simples na utilização, mas que supra suas necessidades. Portanto, para que os usuários possam ter uma boa experiência na utilização do sistema, o EzMart deve prezar pela qualidade, interatividade e desempenho do sistema e que seja uma solução útil aos seus usuários.

1. idosos responsáveis pelas atividades do lar:

Faixa etária: a partir de 50 anos.

Sexo: indiferente.

Características: esses usuários podem não ter facilidade em utilizar as tecnologias dos navegadores web, no entanto, sabem da necessidade de um sistema para auxiliá-los no dia a dia. Portanto, para que eles possam ter uma boa experiência de utilização, o EzMart deve prezar pela simplicidade das funcionalidades e desempenho satisfatório. Os usuários desta categoria esperam que o sistema seja de fácil utilização e, para isso, deve-se desenvolver uma interface de usuário minimalista e bem explicativa visando suprir as necessidades deste grupo de usuários.

1. Estabelecimentos:

Características: os usuários deste perfil devem possuir conhecimentos suficientes para utilizar o sistema, por meio de treinamentos e manuais de utilização, a fim de que possam tomar decisões para melhoria na prestação de seus serviços. Esses usuários conhecem bem o domínio do problema e a necessidade de uma ferramenta de melhoria. Para atendê-los, o sistema EzMart deve fornecer todas as informações significativas das avaliações realizadas pelos consumidores para que possam ser utilizadas na tomada de decisão.

Desta maneira, é possível a obtenção de informações pertinentes aos perfis de possíveis consumidores ou partes interessadas do projeto EzMart por meio de entrevistas. Aas entrevistas com três perfis de usuários encontram-se disponíveis no Apêndice Y.

### 6.8.2 Aspectos Visuais da Interface de Usuário

Os aspectos visuais da interface representam a forma pela qual os usuários se comunicam com o *software* e podem ser classificados em 3 importantes tipos:

1. esquema de cores;
2. ícones;
3. fontes tipográficas.

Segundo Benyon (2011, p. 212), é preciso “elaborar um *design* de forma que as pessoas vejam e percebam coisas em uma tela. Os botões devem ser grandes o suficiente para serem vistos e precisam ser identificados de forma compreensível para as pessoas”.

Segundo Pressman (2010), é necessário que sejam compreendidos os seguintes pontos para que se obtenha um aspecto visual relevante e consistente da interface do usuário:

1. pessoas (usuários finais) que irão interagir com o sistema por meio da interface;
2. tarefas que devem ser realizadas pelos usuários finais para completar seus trabalhos;
3. conteúdo apresentado como parte da interface;
4. ambiente onde as tarefas serão conduzidas.

A partir da compreensão dos aspectos descritos nos itens “a” até “d”, é possível construir uma interface instintiva e confiável, com cores, ícones e outros aspectos visuais precisamente elaborados.

No entanto, para construir uma interface que contenha um aspecto visual que seja consistente para o usuário, é necessário o embasamento teórico científico, que explicita detalhadamente e de forma ampla o que o usuário apresenta de necessidades visuais para quaisquer sistemas de *software*.

Ao se tratar de cores, há inúmeros estudos que descrevem a importância das cores no segmento de *marketing* de um produto ou sistema, e o efeito que cada uma delas têm no cérebro humano.

Dentre a importância dada às cores, Soares (1991, p. 41) destaca as de “distinguir, unificar, chamar a atenção, camuflar, estruturar, codificar, indicar, simbolizar e estilizar”. A função de distinguir tem a finalidade de separar conceitos, de diferenciar. Já a função de unificar tende a agrupar conceitos em uma mesma gama de cores, como por exemplo, em uma empresa onde desde a pintura dos carros até os uniformes dos funcionários seguem uma padronagem tonal. O uso das cores para chamar a atenção é a forma de maior uso de contrastes citado, onde a função da cor é se destacar e atrair toda a atenção. Os itens de segurança normalmente usam cores destacadas a fim de ser o elemento mais visualizado no conjunto.

Goldman (1964, p. 145), complementa a importância das cores, afirmando que cada cor tem um estímulo no cérebro do usuário, e isso é responsável por atrair sua atenção. Ele destaca que “87% dos estímulos que chegam ao nosso cérebro vão através da visão, ficando a audição com 7%, o olfato 3%, o tato 1,5% e o paladar com 1,5%”. Ao receber o estímulo da luz, "os olhos alimentam o cérebro com informação codificada em atividade neural – cadeias de impulsos elétricos – a qual, pelo seu código e pelos padrões de atividade cerebral, representa objetos" (GREGORY,1979, p.77).

Ao se tratar de *marketing,* Carvalho (2013, p.2), 84,7% afirma quedos consumidores acham que a cor é um dos fatores mais importantes ao escolher um produto e a cor pode ser responsável por 60% da aceitação ou rejeição de um produto ou serviço. As cores em uma interface desempenham um papel importantíssimo ao despertar sentimentos diversos nos usuários, por isso, elas devem ser escolhidas cuidadosamente. A Figura 44 comprova a importância das cores em um produto por meio de dados extraídos de uma pesquisa.

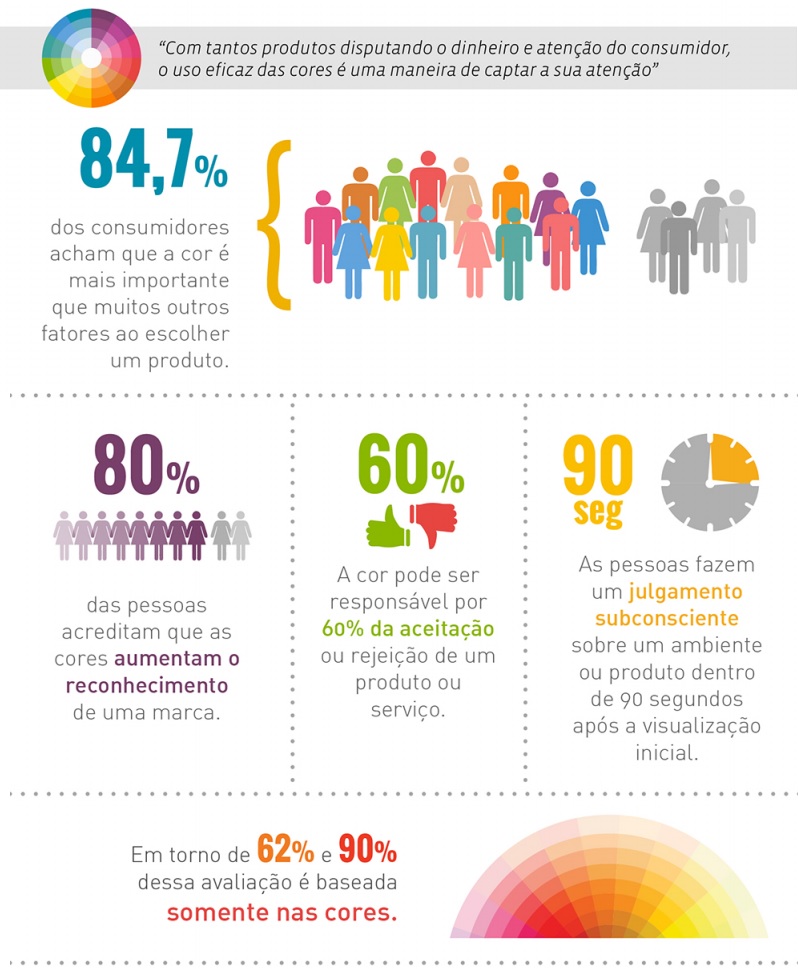


Figura 44- A importância das cores

Fonte: Carvalho (2013)

O corpo humano, ao receber pelas células foto-sensoras dos olhos as ondas de luz que definem as cores, repassa ao cérebro diversas sensações antes mesmo de uma interpretação destas. Algumas informações são transmitidas ao sistema límbico, às glândulas pituitária, hipotálamo e pinel, glândulas estas que controlam o sistema nervoso. Desta maneira, fica clara a interferência fisiológica e, por conseguinte psicológica pelas cores.

Assim, de acordo com um dos princípios fundamentais do sistema EzMart, é necessário que todas as cores que são empregadas neste sistema, estimulem visualmente o usuário. Portanto, de acordo com o embasamento científico teórico ao longo dessa seção, o sistema EzMart utiliza a cor vermelha e a amarela, assim como é exemplificado no logotipo exposto na Figura 45.



Figura 45- Logotipo do sistema EzMart

Fonte: Carvalho (2013)

Para justificar a escolha de cores tão significativas que compõem intimamente um dos aspectos visuais do sistema EzMart, os aspectos estimulantes dessas cores são tratados a seguir.

O vermelho possui um tempo menor de percepção, 0,02 segundos, com ondas de comprimento longo e de maior intensidade, excitando o cérebro, porém não deixando-o cansado. Neste caso, o vermelho eleva a pressão arterial do indivíduo, acelera os batimentos cardíacos e causa certa inquietação, interferindo ainda no sistema nervoso responsável pelo alerta, pelo ataque e pela defesa. Outra cor bastante interessante do ponto de vista fisiológico é o amarelo, quem tem o mais rápido tempo de percepção, 0,01 segundo, e atua nas funções metabólicas e hipotalâmicas, despertando a sensação de fome.

Em resumo, o primeiro aspecto da interface do usuário, tratado como cores, é um importante fator para o estímulo visual do usuário, assim como uma significativa ferramenta de *marketing* para comercialização ou propaganda do sistema ou produto. Dessa maneira, o *marketing* pode ser entendido como uma forma de persuasão e o que realmente importa é a percepção do produto com seus benefícios e características compreendidos pelo consumidor. O *marketing* busca tornar atraente, impulsionar o produto em que trabalha, impressionando, comovendo, emocionando, enfim, agregando valor, sendo muitas vezes utilizados recursos de sugestão, de alusão, de amplificação, da repetição, que amenizam ou exageram determinada expressão ou característica.

Outro aspecto importante são os ícones. Para Benyon (2011, p. 215) os ícones são usados para representar características e funções em aplicações de *software*. Eles são úteis para ajudar as pessoas a decidirem qual função ela deve acessar. Ainda segundo Benyon (2011, p. 216), há 3 tipos de representação de ícones: metáfora, mapeamento direto e convenção. A técnica mais simples é o mapeamento direto, que consiste em criar uma imagem daquilo que o ícone pretende representar.

Com o objetivo de auxiliar o usuário no manuseio do sistema, o EzMart possui diversos ícones que ajudam a reconhecer as suas funcionalidades. Esses ícones são tratados na Figura

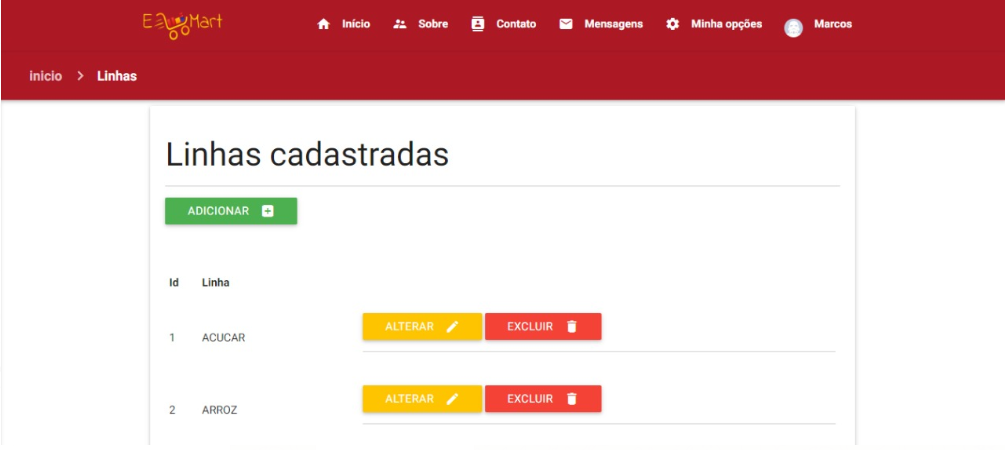


Figura 46- Página de listagem de linhas do sistema EzMart na versão V 1.0

Fonte: Carvalho (2013)

Em detalhamento sobre alguns ícones específicos:

1. ícone de início: Responsável por redirecionar o usuário para a tela inicial do sistema, sempre que o mesmo estiver localizado em outra página. Como no exemplo da Figura 46, o usuário está localizado na página de linhas, o que pode se notar pelo caminho início > Linhas. Portanto, caso o usuário queira se redirecionar para a página inicial/ principal da aplicação, basta que ele clique no ícone representado como uma casa, para que se obtenha essa funcionalidade;
2. ícone de opções: Responsável por exibir as opções referentes ao perfil de usuário que está autenticado no momento, sendo isso justificado pelo ícone de configurações comumente usados em vários sistemas;
3. ícone adicionar: Responsável por permitir que um usuário, que neste caso está autenticado como um estabelecimento, adicione uma linha à um produto, e neste caso justifica-se a utilização de um ícone representativo de “+”;
4. ícone alterar: Responsável por permitir que um usuário, que neste caso está autenticado como um estabelecimento, altere a linha a qual pertence um produto, sendo isso justificado pelo uso de um ícone representativo de lápis que remete a ideia de escrever/sobrescrever;
5. ícone excluir: Responsável por permitir que um usuário que neste caso está autenticado como um estabelecimento, exclua a linha a qual pertence um produto, sendo isso justificado pelo uso de um ícone representativo de lixeira, que remete a ideia de descartar quaisquer itens.

Desta maneira, todos os itens da aplicação, não só os mencionados nos itens “a” até “e”, seguem a mesma padronização, além das cores, em tamanhos, dimensões, e de forma a tornar a aplicação mais intuitiva.

Já, o terceiro e último aspecto visual se trata das fontes em que os textos gerais do sistema são escritos, no entanto, não só estes, mas o logotipo do sistema. Especificamente, apenas o logotipo não foi produzido utilizando os mesmos padrões e características aplicadas para os textos de botões, ícones ou títulos do sistema, pois foi projetado para atender o *marketing* geral do sistema, utilizando de letras que remetem significados e estão ilustradas, como no caso da letra Z, especificada na Figura 45. Já as demais características das letras do sistema, como textos de ícones, botões, ou títulos foram projetadas para atender quaisquer variedades de clientes consumidores, em todas as faixas etárias de idade, pois caso sejam idosos, eles devem se sentir agradáveis principalmente com o tamanho da fonte em que os textos estão escritos. Além de que as letras não são ornamentais, o que facilita o entendimento de quaisquer usuários.

### 6.8.3 Heurísticas de Usabilidade

Para Nielsen (1993) medir a usabilidade de uma interface envolve não apenas medir questões relativas às funcionalidades de um *software*, mas também a facilidade de seu uso como ferramenta de trabalho, tendo como um dos principais desafios a redução do tempo necessário para aprender a utilizar o sistema. Com isso, definiu as 10 heurísticas de usabilidade, a seguir:

1. visibilidade do estado atual no sistema;
2. correspondência entre o sistema e o mundo real;
3. liberdade de controle para o usuário;
4. consistência e padrões;
5. prevenções de erros;
6. reconhecimento em vez de memorização;
7. flexibilidade e eficiência de uso;
8. estética e design minimalista;
9. ajudar os usuários a identificarem, diagnosticarem e recuperarem-se de erros;
10. ajuda e documentação.

Para cada heurística deve-se ter atenção na análise, uma vez que o conjunto de todas elas devidamente implantado, fornece uma usabilidade adequada em relação ao *software*.

Em detalhamento, a primeira heurística descrita no item “a” corresponde à garantia que o usuário tenha plena compressão de qual estado no sistema o mesmo se encontra. Além disto, o sistema deve fornecer informações constantes ao usuário, de forma que dinamicamente, o mesmo esteja consciente das ações dele para com a aplicação, isto é, se os dados de entrada são consistentes e quais são as possíveis formas de interação com o sistema.

A Figura 47 ilustra um exemplo de visibilidade do sistema EzMart indicando que o usuário se encontra na página de cadastro, e os campos que devem ser preenchidos para que essa ação seja concluída.

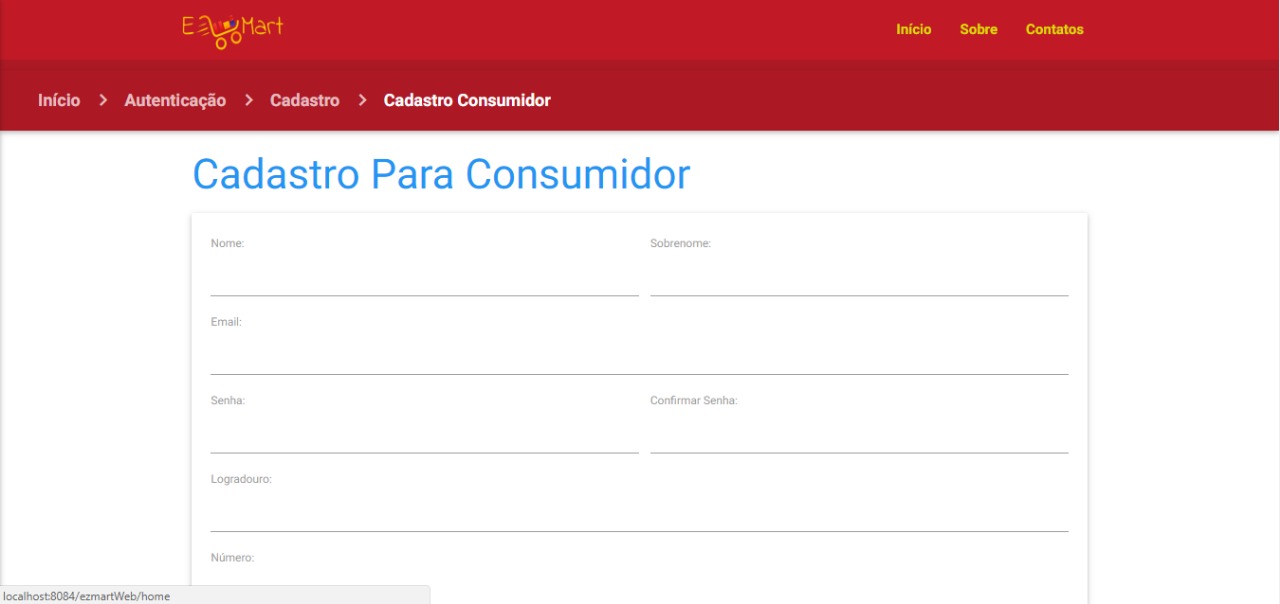


Figura 47- Página de cadastro para um usuário consumidor V 3.0

Fonte: Equipe (2018)

A segunda heurística descrita no item “b” possui como objetivo fornecer elementos na interface do sistema que se assemelham aos elementos do mundo real, de forma que a carga de memória do usuário seja reduzida, usando elementos e ícones contextualizados que sejam coerentes com o modelo mental do usuário. Conforme Nielsen explica (1994), o sistema deve falar a linguagem dos usuários, com palavras, frases e conceitos familiares ao usuário, em vez de termos orientados pelo sistema.

A Figura 48 fornece uma exemplificação deste conceito no sistema EzMart, a partir dos ícones utilizados nos botões “Alterar” e “Excluir”.

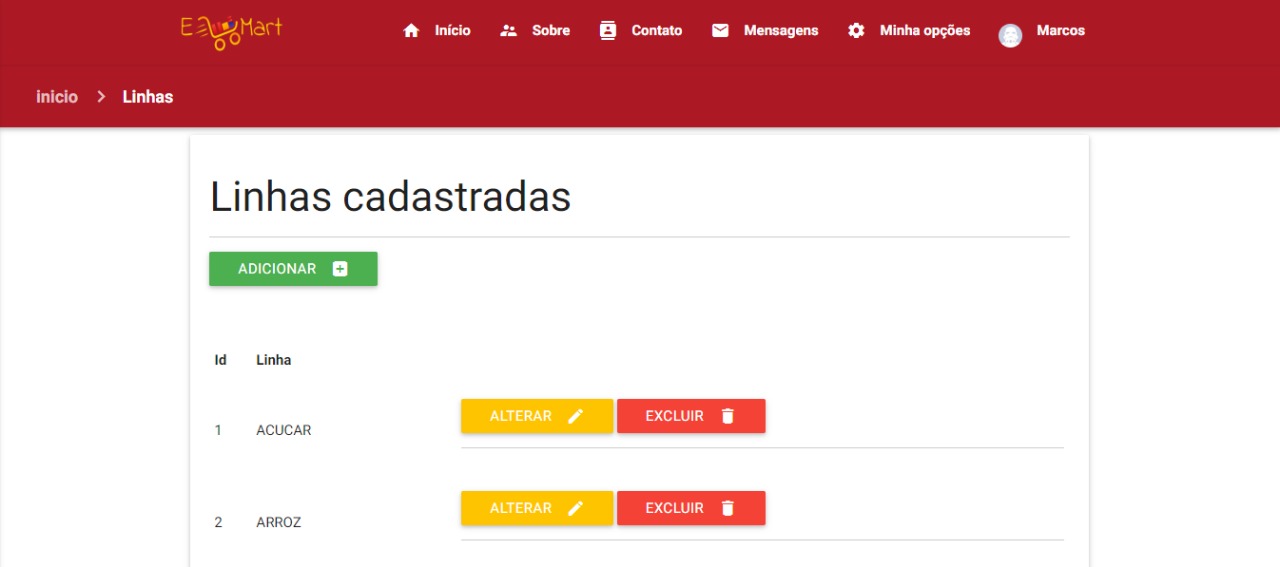


Figura 48- Página de listagem de linhas com opções de alterar e excluir V 3.0

**Fonte: Equipe (2018)**

Já a terceira heurística descrita no item “c”, tem por objetivo definir o controle e liberdade do usuário, que diz respeito a autonomia do mesmo para navegar no sistema de forma livre, isto é, de forma que possa desfazer operações, iniciar novas tarefas ou até mesmo desconectar do sistema a qualquer momento.

É na quarta heurística descrita no item “d”, que Nielsen (1995) propõe que os usuários não devem se perguntar se palavras, situações ou ações diferentes significam a mesma coisa. A padronização de um sistema diz respeito à diminuição de carga de memória do usuário. Isto é, diante de tantos sites e aplicações, os componentes e padrões de interação devem ser respeitadas no sistema para que o usuário se sinta mais à vontade e familiarizado com o mesmo.

Um exemplo deste conceito é a disposição dos elementos no sistema EzMart, visando a facilidade de navegação conforme descrito na seção Aspectos Visuais da Interface de Usuário

Sobre as prevenções de erros como quinta heurística, que é descrita no item “e”, Nielsen (1990) orienta que ainda melhor do que boas mensagens de erro é um projeto cuidadoso que impede que um problema ocorra em primeiro lugar. A interface deve fornecer interações com o usuário de forma que sejam prevenidos erros, e caso estes aconteçam que fique claro o motivo e de que maneira o usuário poderá reverte-lo.

O sistema EzMart, pretende viabilizar o conforto para o usuário ao utilizar a aplicação de forma intuitiva e simples através da sexta heurística, descrita no item “f”. É necessário que a interface forneça uma distribuição de elementos que intuitivamente indique ao usuário o contexto no qual ele se encontra e consequentemente, quais são as suas possibilidades de navegação.

Nielsen (1994) orienta que se deve minimizar a carga de memória do usuário, tornando os objetos, ações e opções visíveis. O usuário não deve ter que lembrar informações de uma parte do diálogo para outra. As instruções de uso do sistema devem ser visíveis ou facilmente recuperáveis ​​sempre que apropriado.

A aplicação deste conceito é mostrada por meio da Figura 49, cujo recurso de *Breadcrumb,* é utilizado, indicando o caminho percorrido pelo usuário.

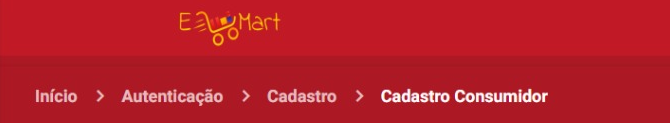


Figura 49- Recurso de Breadcrumb do sistema EzMart V 3.0

Fonte: Equipe (2018)

A sétima heurística, descrita no item “g” indica a necessidade da aplicação de permitir que o usuário iniciante ou avançado crie fluxos de acordo com a sua usabilidade, além disto, disponha de atalhos para auxiliar na eficiência de uso do sistema. Um sistema inflexível e ineficiente pode causar frustações ao usuário.

Já a oitava heurística descrita no item “h”, refere-se ao conceito da heurística de interface com o usuário, que é um dos mais importantes, uma vez que está relacionado com a facilidade de navegação pelo sistema e acesso às informações necessárias sem a necessidade de avançar muitas telas. O ideal é que seja utilizado somente o necessário no quesito imagens, texto e design, evitando-se informações irrelevantes que possam confundir o usuário.

Nesse sentido o sistema EzMart preza o uso de elementos adequados e minimalistas, evitando exageros e recursos irrelevantes.

Na nona heurística, descrita no item “i”, o suporte em reconhecimento, diagnóstico e recuperação de erros é expresso significativamente na medida em que o sistema fornece *feedbacks* e mensagens de erros claras, que indicam objetivamente o problema e o que deve ser corrigido para que o mesmo funcione de forma simples. Nielsen (1994) explica que as mensagens de erro devem ser expressas em linguagem simples (sem códigos), indicar precisamente o problema e sugerir construtivamente uma solução.

A Figura demonstra um exemplo no sistema EzMart, onde, caso o usuário consumidor não preencha devidamente os campos obrigatórios para criação de uma nova conta, não é possível que ele prossiga para etapas posteriores, e para que não haja chances de erros, o usuário é orientado por meio de mensagens informativas que os conduzem a uma melhor experiência com o sistema.

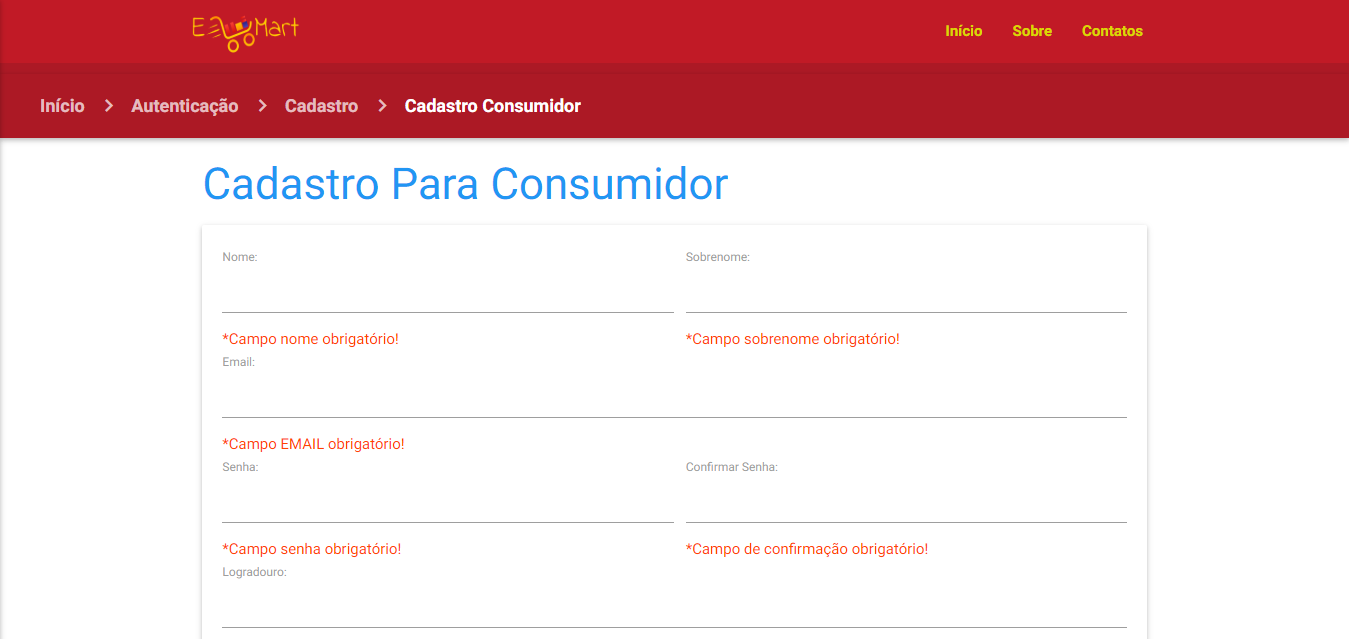


Figura 50- Mensagens informativas sobre campos obrigatórios

Fonte: Equipe (2018)

Na última heurística, descrita no item “j” sobre a ajuda ao usuário e documentação, denota que embora a utilização por parte do usuário deva ser evitada, utilizando-se de uma interface fácil de usar e intuitiva, a documentação deve ser facilmente acessível, nos casos em que ela for necessária. Deve ser escrita de forma clara, direta e em linguagem acessível aos usuários. Sendo assim, a partir da documentação, os usuários podem utilizar a aplicação de forma simples e carente de dúvidas excessivas.

# 7 PLANO DE TESTES

O teste de um sistema é uma das atividades do processo de desenvolvimento de *software* que tem por objetivo de verificar e validar o sistema, de modo sistemático, com a finalidade de encontrar falhas, defeitos e erros, bem como assegurar a qualidade do que é produzido como artefato, a

Este Capítulo apresenta o planejamento das atividades de teste de validação do sistema EzMart.

Nele consta a estratégia de teste adotada, os documentos relevantes para a elaboração do plano geral de testes, o ambiente para a realização e os itens que fazem parte deste plano, bem como o resultado dos testes.

## 7.1 Finalidade

Os testes apresentados neste Capítulo têm por finalidade verificar o grau com que os requisitos elaborados e descritos na seção 5.1.1 são atendidos. Segundo Sommerville (2008), a verificação consiste em avaliar se a aplicação atende, corretamente, aos requisitos que foram identificados inicialmente.

## 7.2 Escopo

Nesta seção são apresentados os documentos relevantes para a construção do plano de testes do sistema EzMart, bem como os itens e o ambiente a ser utilizado para a realização dos testes.

### 7.2.1 Referências a Documentos Relevantes

Os documentos relevantes para a realização deste plano de testes são apresentados no Quadro 7.

|  |  |
| --- | --- |
| Tipo de Material | Referência |
| Requisitos Funcionais | 5.1.1 deste documento |
| Diagramas de Casos de Uso | 5.2.1 deste documento |
| Descrição de Casos de Uso | Apêndice F anexado à este documento. |

Quadro 7-Documentos relevantes para os testes do sistema EzMart

Fonte: Equipe (2018)

### 7.2.2 Ambiente para a Realização dos Testes

O Quadro 8 explicita os equipamentos utilizados nos testes do sistema EzMart.

|  |  |  |
| --- | --- | --- |
| Equipamento | Marca/Modelo/Configuração | Finalidade |
| *Notebook* | Asus Processador Core™ I5 – 5 geração com 8 GB de RAM e 1 TB de HD | Executar a aplicação EzMart |
| *Desktop* Asus | Asus Processador AMD Phenon IIx4 com 12 GB de RAM e 500 TB de HD | Executar a aplicação EzMart |

Quadro 8- Equipamentos utilizados nos testes do sistema EzMart

Fonte: Equipe (2018)

O Quadro 9 explicita os recursos de *software* utilizados nos testes do sistema EzMart.

|  |  |  |
| --- | --- | --- |
| Software/Versão | Fabricante | Finalidade |
| Apache Tomcat V 8.0 | *Apache Software Foundation* | Executar o *container web* da aplicação. |
| NetBeans V 8.1 | *Oracle* | Gerenciador de serviços |
| Windows V 7.0 | *Microsoft* | Plataforma de Sistema Operacional. |
| PostgreSQL V 9.4 | *The PostgreSQL Global Development Group* | Acesso ao banco de dados |

Quadro 9- Recursos de software para os testes do sistema EzMart

Fonte: Equipe (2018)

## 7.3 Especificação dos casos de teste

Nesta subseção são apresentados os itens que são testados, a rastreabilidade entre os requisitos e os casos de testes e a descrição dos casos de testes.

### 7.3.1 Item a Testar

O Quadro 10 apresenta os Requisitos Funcionais do sistema EzMart que são submetidos aos testes.

|  |  |
| --- | --- |
| Identificação do Item | Descrição |
| 1 | [RF01] – Realizar autenticação |
| 2 | [RF04] – Cadastrar consumidor |
| 3 | [RF05] – Cadastrar estabelecimento |
| 4 | [RF10] – Gerenciar lista de compra |
| 5 | [RF14] – Pesquisar produto |

Quadro 10- Requisitos do sistema EzMart submetidos à testes

Fonte: Equipe (2018)

### 7.3.2 Rastreabilidade entre Requisitos e Casos de Teste

O Quadro 11 apresenta a rastreabilidade entre os requisitos e os casos de teste. Os casos de testes são os diversos cenários, ou seja, as diversas possibilidades que existem a partir de uma funcionalidade. É necessário e devidamente importante que todos os possíveis cenários sejam abordados, a fim de diminuir significativamente a probabilidade de erros, defeitos ou falhas.

|  |  |
| --- | --- |
| Identificação do Requisito | Caso(s) de teste(s) aplicável(eis) |
| [RF01] – Realizar autenticação | CT1 – Realizar autenticação com sucesso;  CT2 – Realizar autenticação com um usuário que não foi cadastrado;  CT3 – Realizar autenticação com campos vazios;  CT4 – Realizar autenticação sem conexão com a Internet;  CT5 – Realizar autenticação somente com o campo de usuário vazio;  CT6 – Realizar autenticação somente com o campo senha vazio;  CT7 – Realizar autenticação com uma senha inválida;  CT8- Realizar autenticação com um *e-mail* inválido. |
| [RF04] – Cadastrar consumidor | CT9 – Cadastrar consumidor com sucesso;  CT10 – Cadastrar consumidor com campos vazios;  CT11 – Cadastrar consumidor sem confirmar a senha;  CT12 – Cadastrar consumidor sem conexão com a Internet. |
| [RF05] – Cadastrar estabelecimento | CT13 – Cadastrar estabelecimento com sucesso;  CT14 – Cadastrar estabelecimento com campos vazios;  CT15 – Cadastrar estabelecimento com caracteres especiais;  CT16 – Cadastrar estabelecimento sem preencher os campos obrigatórios. |
| [RF10] – Gerenciar lista de compra | CT17 – Criar lista de compra com sucesso;  CT18 – Criar lista de compra vazia;  CT19 – Editar lista de compra com sucesso;  CT20 – Excluir lista de compra com sucesso. |
| [RF14] – Pesquisar produto | CT21 – Pesquisar por produto existente;  CT22 – Pesquisar por produto inexistente;  CT23 - Pesquisar produto sem conexão com a internet. |

Quadro 11- Rastreabilidade entre requisitos e casos de testes

Fonte: Equipe (2018)

### 7.3.3 Descrição dos Casos de Teste

Nesta subseção é apresentada a descrição dos casos de testes previstos neste plano, e que foram mencionados no Quadro 9. Vale ressaltar que apenas alguns deles encontram-se disponíveis neste documento. Porém testes exploratórios referentes à todos os requisitos implementados são executados em todas as entregas formais.

## 7.4 Resultado dos casos de teste

### 7.4.2 Resultados

# 8 PLANO DE IMPLANTAÇÃO

## 8.1 METODOLOGIA

### 8.1.1 Descrição da Metodologia

### 8.1.2 Matriz de Responsabilidades

## 8.2 TREINAMENTOS PREVISTOS

## 8.3 CRONOGRAMA DE IMPLANTAÇÃO

## 8.4 DOCUMENTO DE APOIO À IMPLANTAÇÃO

## 8.5 VISÃO DE IMPLANTAÇÃO

# 9 CONCLUSÃO

# REFERÊNCIAS

ALBRECHT, Mauro. **GASTOMETRO**: desenvolvimento de um aplicativo Android para auxiliar nas compras de supermercados. 2014. 49 p. Monografia de Especialização (Especialista em Desenvolvimento de Sistemas para Internet e Dispositivos Móveis)- Universidade Tecnológica Federal do Paraná, Francisco Beltrão - Brasil, 2014. Disponível em:<http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/6915/1/FB\_DESIDM\_I\_2014\_05.pdf>. Acesso em: 09 mar. 2018.

ALEXANDRUK, Marcos. **Modelagem de Banco de dados.** São Paulo: Unilivros, 2011. Disponível em: <http://www.unilivros.com.br/pdf/dbmod.pdf>. Acesso em: 10 fev. 2018.

APPLEBAUM, W. Methods for determining store trade areas, market penetration and potential sales. **Journal of Marketing Research**, Chicago, v. III, p. 127-141, May 1966.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS – ABNT. NBR ISO 10006:1997 (E). **Gestão** **da qualidade – Diretrizes para a qualidade no gerenciamento de Projetos.** Rio de Janeiro: ABNT, 1997.

ASSOCIAÇÃO BRASILEIRA DE SUPERMERCADOS (ABRAS). **Guia Abras de Marcas Próprias**. São Paulo: ABRAS, 2011.

BATISTA, Emerson de Oliveira. **Sistema de Informação**: o uso consciente da tecnologia para o gerenciamento. São Paulo: Saraiva, 2004.

BELMIRO, João; PASTORE, Ricardo. "**Pesquisa em Varejo Omnichannel: uma Revisão Sistemática e Análise de Conteúdo Quantitativo**" .*11º Congresso Latino-Americano de Varejo:"Engaging and Interactive Shopper Experience"* (2017): n. pág. Web. 10 Ago. 2018

BERRY, L.; PARASURAMAN, A. **Serviços de marketing:** competindo através da qualidade. São Paulo: Maltese, 1992.

BENYON, David. **Interação humano-computador.** 2. ed. São Paulo: Pearson Prentice Hall, 2011.

BISPO, C.A.F. **Uma análise da nova geração de sistemas de apoio à decisão.** 1998. 160p. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, 1998.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML**:Guia do Usuário. 2. ed. Rio de Janeiro: Elsevier, 2006.

BRADESCO. **Supermercados**. São Paulo: [S.n], 2017. Disponível em: <https://www.economiaemdia.com.br/EconomiaEmDia/pdf/infset\_supermercados.pdf>. Acesso em: Jul. 2018.

CARVALHO, M. M.; RABECHINI JUNIOR, R. **Fundamentos da Gestão de Projetos**: construindo competências para gerenciar projetos. 3. ed. São Paulo: Atlas, 2011

CASTILHO, Robson. **Conhecendo *Design Patterns* e o padrão *Strategy****.* 2011. Disponível em: <https://robsoncastilho.com.br/2011/06/25/conhecendo-design-patterns-e-o-padrao-strategy/>. Acesso em: 22 ago. 2018.

CHOO, C. W. A. **Organização do Conhecimento**: como as organizações usam a informação para criar significado, construir conhecimento e tomar decisões. São Paulo: Editora Senac São Paulo, 2003.

CHOPRA, Sumil; MEINDL, Peter. **Supply Chain Management:** Strategy, Planning, and Operation. New Jersey: Prentice-Hall, Inc., 2001, p. 354.

COULOURIS, George; DOLLIMORE, Jean; TIM, Kindberg. **Sistemas distribuídos - conceitos e projeto.** 4. ed. São Paulo: Bookman, 2007.

CUNICO, Malton William Machado. **Desenvolvimento de sistema de colaboração em massa para reduzir gastos dos consumidores com compras de supermercado**. 2012. 45 p. Trabalho de conclusão de curso (Tecnologia em Desenvolvimento de Sistemas Distribuídos)- Universidade Tecnológica Federal do Paraná – UTFPR; Curitiba-Brasil, 2012. Disponível em: <https://portaldeinformacao.utfpr.edu.br/Record/roca-1-1125>. Acesso em: 09 mar. 2018.

CLEMENTS, P. et al. **Documenting software architectures:** views and beyond.Pearson Education, 2002.

DONDIS, Donis A. Sintaxe da linguagem visual. São Paulo: Editora Martins Fontes, 2000.

DUBOC, L. et al. A framework for characterization and analysis of *software* system scalability. In: ESEC-FSE 07. **Proceedings of the the 6th joint meeting of the European *software* engineering conference.** USA: 2007. p. 375-384.

E-GOV. **Governo Eletrônico Brasileiro**. 2018. Disponível em: Acesso em: 23 Mai. 2018

FERREIRA, Marco Aurélio Marques; VENANCIO, Michele Moutinho; ABRANTES, Luiz Antônio. Análise da eficiência do setor de supermercados no Brasil.**Econ. Apl.**,  Ribeirão Preto ,  v. 13, n. 2, p. 333-347,  Jun. 2009 .   Disponível em:<http://www.scielo.br/scielo.php ?script =sci\_arttext&pid= S1413-80502009000200007&lng=en&nrm=iso>. Acesso em  21  Abr.  2018.

FLORENZANO, Cláudio. **Tipos de sistema de informação nas organizações.** 2015. Disponível em: < http://www.cbsi.net.br/2015/04/tipos-de-sistemas-de-informacao-nas-organizacoes.html>. Acesso em: 04 mar. 2018.

FREEMAN, R. E. **Strategic management:** a stakeholder approach. Boston: Pitman Publishing, 1984.

FISHER, A.; SHARKIE, C. **Jump Start Responsive Web Design**. Austrália: SitePoint, 2013. 145 p.

GUEDES, Gilleanes T. A. **UML - Uma Abordagem Prática**. 3. ed. São Paulo: Novatec Editora, 2009.

GODOY, Arilda Schimidt; RUGGIERO, Alberto Pirró. A influência da tecnologia da informação no trabalho gerencial: um estudo com gestores de recursos humanos. **Revista Eletrônica de Administração,** 49. ed., v. 12, n.1, 2006.

GOLDMAN, Simão. Psicodinâmica das cores. Porto Alegre: Ed. La Salle, 1964.

GREGORY, Richard L. Olho e Cérebro. Psicologia da Visão. Rio de Janeiro: Zahar

Editores, 1979.

HELDMAN, Kim. **Gerência de Projetos:** guia para o exame oficial do PMBoK. 5. ed. Rio de Janeiro: Elsevier Editora, 2009.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA (IBGE). **Pesquisa Anual do Comércio,** v. 28. Rio de Janeiro: IBGE, 2016.

KARNER, G. **Resource Estimation for Objectory Projects**. Sweden: Objetive System, 1993.

LIMA, Guilherme Pereira. **Gestão de projetos:** como estruturar logicamente as ações futuras. 1. ed. Rio de Janeiro: LTC, 2009.

LIMA, Luiz Augusto de Paula. **Projeto Físico**. 2016. Disponível em: <http://www.ppgia.pucpr.br/~laplima/ensino/pfec/arquivos/s1-doc4\_projetofisico.pdf>. Acesso em: 20 jun. 2017.

MARTIN, James. S**trategic data-planning methodologies**. 2. ed. Englewood Cliffs, New Jersey: Prentice Hall, 1982. 236 p.

MEDEIROS, Higor. **Padrão de Projeto Singleton em Java.** São Paulo: Devmedia, 2013. Disponível em: <http://www.devmedia.com.br/padrao-de-projeto-singleton-em-java/26392>. Acesso em: 22 ago. 2018.

MENDES, Antonio. **Engenharia de Software 3 – Requisitos Não Funcionais**. 2016. Disponível em: <http://www.devmedia.com.br/artigo-engenharia-de-*software*-3-requisitos-nao-funcionais/9525>. Acesso em: 28 mar. 2018.

NIELSEN, J. Ten Usability Heuristics. USA: [S.n], 1994. Disponível em: https://www.nngroup.com/articles/ten-usability-heuristics/. Acesso em: ago. 2018.

OLIVEIRA, Djalma de Pinho Rebouças de. **Sistemas de Informações Gerenciais**: estratégias e táticas operacionais. 8. ed., São Paulo: Atlas, 1992.

PANCHAL, Jiteshi H.; FATHIANATHAN, Mervyn. **Product realization in the age of mass collaboration**. In: ASME 2008. New York: 2008.

PARENTE, J.; KATO, T. H. Área de influência: um estudo no varejo de supermercados. **RAE - Revista de Administração de Empresas**, São Paulo, v. 41, n. 3, p. 46-53, Abr./Jun. 2001. Disponível em: <http://www.scielo.br/pdf/rae/v41n2/v41n2a05> Acesso: 21 abr. 2018.

PEREIRA, M. J. L de B.; FONSECA, J. G. M. **Faces da decisão**: Abordagem sistêmica do processo decisório. Rio de Janeiro: ITC, 2009.

PORTAL EDUCAÇÃO. **Administração – níveis hierárquicos.** 2013. Disponível em: < https://www.portaleducacao.com.br/conteudo/artigos/administracao/administracao-niveis-hierarquicos/47644>. Acesso em: 18 mar. 2018.

PROJECT MANAGEMENT INSTITUTE. **Um Guia do Conhecimento em Gerenciamento de Projetos (GUIA PMBOK)**. 5. ed. Pennsylvania: Project Management Institute, 2013.

PRESSMAN, Roger S. **Engenharia de software:** uma abordagem profissional.8. ed. Porto Alegre: Bookman, 2016.

ROBBINS, Stephen P. **Comportamento Organizacional**. 9. ed. São Paulo: Prentice Hall, 2002. 637 p.

RODRIGUES; M. A. Escavando dados no varejo. **Revista Tecnologística**, Set. 1998. Disponível em: http://www.tecnologistica.com.br/site/5%2C1%2C26%2C5480.asp. Acesso em: 22 abr. 2018.

ROJO, F. J. G. Pesquisa: o comportamento do consumidor nos supermercados. **ERA - Revista de Administração de Empresas**, São Paulo, v. 38, n. 3, p. 16-24, Jul./Set.1998. Disponível em: < http://www.scielo.br/pdf/rae/v38n3/a03v38n3.pdf> Acesso em: 21 abr. 2018.

ROSSETTI, Adroaldo Guimarães; MORALES, Aran Bey Tcholakian. O papel da tecnologia da informação na gestão do conhecimento. **Ciência da Informação**, [S.l.], v. 36, n. 1, dez. 2007. Disponível em: <<http://revista.ibict.br/ciinf/article/view/1191/1362>>. Acesso em: 21 abr. 2018.

SHETH, Jagdish N., MITTAL, Banwari, NEWMAN, Bruce I. **Customer Behavior**: consumer behavior and beyond. Fort Worth: The Dryden Press, 1999.

SILVA, M. C. M.; TEIXEIRA, R. M. Gerenciamento da tecnologia da informação para tomada de decisão em supermercados. **Revista de Ciências da Administração**, Florianópolis, v. 4, n. 6, p.69-80, jan/jun. 2002.

SIMON, H. A. Comportamento administrativo. Rio de Janeiro: Fundação Getúlio Vargas, 1965.

SOARES, Paulo T. O Mundo das Cores. São Paulo, Editora Moderna, 1991.

SOMMERVILLE, Ian. **Engenharia de Software.** 8. ed. Rio de Janeiro: Prentice-Hall, 2008.

VAZQUEZ,C. E.; SIMÕES, G. S.; ALBERT, R. M. **Análise de ponto de função medição, estimativa e gerenciamento de projetos de software.** São Paulo: Editora Érica, 2009.

YANES, B. The emerging role of electronic marketplaces on the Internet. **Communications of the ACM,** New York, v. 41, n. 8, p. 35-42, Aug. 1998.

ZANETI JUNIOR, Luiz A.; VIDAL, Antonio G. da Rocha. Construção de sistemas de informação baseados na tecnologia web. **R. Adm.,** São Paulo, v. 41, n. 3, p. 232-244, jul./ago./set. 2006. Disponível em: <https://www.revistas.usp.br/rausp/article/viewFile/44402/48022> . Acesso em: 21 abr. 2018.

ZEMEL, T. **Web Design Responsivo**: páginas adaptáveis para todos os dispositivos. São Paulo: Casa do Código, 2013. 150 p.

ZENONE, Luiz Claudio. **Marketing Estratégico e Competitividade Empresarial.** 1 ed. São Paulo: Novatec Editora Ltda, 2007**.**

# OBRAS CONSULTADAS

ADAMS, Elizabeth K.; WILLETTS, Keith J. **The Lean Communications Provider – Surviving the Shakeout through Service Management Excellence**. 1. ed. USA: McGraw-Hill, 1997.

ALMEIDA, R.; TERRA, J. C. C. **Varejo 2.0:** um guia para aplicar redes sociais aos negócios. Rio de Janeiro: Campus, 2011. 200 p.

ALVARENGA NETO, R.D.C. **Gestão do conhecimento em organizações:** proposta de mapeamento conceitual integrativo. Curitiba: Saraiva, 2005.

BARBOSA, Ricardo R.; SEPÛLVEDA, Maria I. M.; COSTA, Mateus U. P. da. **Gestão  
da informação e do conhecimento na era do compartilhamento e da colaboração.  
Informação & Sociedade:** Estudos, João Pessoa, v.19, n. 2, p. 13-24, maio/ago.  
2009.

BORBA, Gilmar. Sistemas Colaborativos – Introdução. 2010. Disponível em <http://gilmarborba.com.br/?p=243>. Acesso em 22 Abr 2018.

CARVALHO, Henrique. **A psicologia das cores no marketing e no dia-a-dia.** 2013. Disponível em: <<http://viverdeblog.com/psicologia-das-cores/>>. Acesso em: 13 fev 2018.

DI SÉRIO, Luiz Carlos; DUARTE, Luis de C. M. Competindo em tempo e flexibilidade – casos de empresas brasileiras. In: CONSELHO LATINO–AMERICANO DE ESCOLAS DE ADMINISTRAÇÃO. **Anais.** Porto Alegre: 2002.

FERREIRA, Edmar. Escolhendo entre escalabilidade horizontal e escalabilidade vertical. 2010. Disponível em: < http://escalabilidade.com/2010/09/21/ escolhendo-entre-escalabilidade-horizontal-eescalabilidade-vertical/>. Acesso em: 22 Abr .de 2018.

FREITAS, Gustavo André de. **Colaboração em massa.** 2008. Monografia (Graduação em Bacharelado em Sistemas de Informação) - Faculdade de Ciências Aplicadas "Sagrado Coração". Linhares, 2008.

GRODISKI, Henrique. A importância do planejamento e controle financeiro para o desempenho empresarial. 2008. Disponível em <http://www.artigonal.com/gestao-artigos/a-importancia-do-planejamento-e-controlefinanceiro-para-o-desempenho-empresarial-386410.html>. Acesso em 22 Abr. 2018.

LOPES, S. A. **Web Mobile:** Programe para um mundo de muitos dispositivos. São Paulo: Casa do Código, 2013. 217 p.

KARDEC, A; FLORES, J.; SEIXAS, E. **Gestão estratégica e indicadores de desempenho.** Rio de Janeiro: Qualitymark, Abraman, 2002.

MCKENNA, Mary. **Five benefits and five challenges of collaborative working**. 2011. Disponível em <http://www.learningpool.com/five-benefits-and-five-challenges-ofcollaborative-working/>. Acesso em 22 abr 2018.

MIRANDA, Candida Leonor. **Satisfação do cliente em supermercados:** a avaliação da qualidade dos serviços. 2001.182 p. Dissertação (Mestrado em Engenharia de Produção) -Universidade Federal de Santa Catarina, Florianópolis-Brasil, 24 de Outubro de 2001.

MONTES, Eduardo. **Estratégias para riscos negativos ou ameaças.** 2017. Disponível em: <https://escritoriodeprojetos.com.br/estrategias-para-riscos-negativos-ou-ameacas>. Acesso em: 24 mar. 2018.

\_\_\_\_\_. **Realizar o controle integrado de mudanças.** 2017.Disponível em: <https://escritoriodeprojetos.com.br/realizar-o-controle-integrado-de-mudancas>. Acesso em: 25 mar. 2018.

RODRIGUES, Elio. **Definição e importância do cronograma de projeto**. 2014. Disponível em: <http://www.elirodrigues.com/2014/09/12/definicao-e-importancia-do-cronograma-de-projeto/>. Acesso em: 23 mar. 2018.

RICARTE, Ivan L. M. **Codificação**.2003.Disponível em: <http://www.dca.fee.unicamp.br/cursos/EA876/apostila/HTML/node8.html>. Acesso em: 26 mar 2018.

SAVOIA, José R. F.; SAITO, André T.; SANTANA, Flávia de A. Paradigmas da educação financeira no Brasil. **Revista de Administração Pública**. Rio de Janeiro, v. 41, no. 6, p. 1121-1141, Nov-Dez, 2007.

SILVA, Taigra Maria da; SILVA, Nailde Gonçalves da; LIMA, Sávia Antonyelle Gomes de; MELO, Maria de Fatima Massena de; LIMA, Daisyvângela Eucrêmia da Silva. Orçamento doméstico e consumo: vamos comercializar. IN: X JORNADA DE ENSINO, PESQUISA E EXTENSÃO. **Anais...** Recife, out. 2010.

SILVEIRA, José Augusto G. da ; LEPSCH, Sérgio Luiz. Alterações recentes na economia do setor supermercadista brasileiro. **Revista de Administração,** v. 32, n. 2, p. 513, abr/jun., 1997.

SOUZA, C.S. et al. Projeto de interfaces de usuário: perspectivas cognitiva e semiótica. In: XIX Congresso da Sociedade Brasileira de Computação, 1999, Rio de Janeiro. **Anais...** Rio de Janeiro., jul. 1999.

SOTILLE, Mauro Afonso *et al.* **Gerenciamento do escopo em projetos.** 1. ed. Rio de Janeiro: Editora FGV, 2007.

TANENBAUM, Andrew S.; STEEN, Maarten Van. **Sistemas distribuídos:** princípios e paradigmas.2. ed. São Paulo: Pearson Prentice Hall, 2007.

THEODORO, Flávio R. F. **O uso da matemática para a educação financeira a partir do ensino fundamental.** Taubaté: 2008.

TRAMMELL, C. J. et al. The incremental development process in Cleanroom *software* engineering.  **Decision Support Systems**, v. 17, 1996, 55-71 p.

# APÊNDICE A – ESTRUTURA ANALÍTICA DE PROJETO (EAP)

A EAP do projeto encontra-se na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE B – CRONOGRAMA DO PROJETO

O cronograma de atividades do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE C – RELATÓRIO DE DESEMPENHO

O relatório de desempenho encontra-se disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE D – PLANILHA PARA ACOMPANHAMENTO DOS RISCOS DE AMEAÇA DO PROJETO

A planilha para acompanhamento dos riscos de ameaça do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE E – DIAGRAMAS DE CASOS DE USO

Os diagramas de casos de uso do projeto encontram-se disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE F – DESCRIÇÃO DOS CASOS DE USO

A descrição dos casos de uso está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE G – DIAGRAMA ENTIDADE RELACIONAMENTO

O diagrama entidade relacionamento está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE H – ESTIMATIVA DE PONTOS DE FUNÇÃO E PONTOS DE CASO DE USO

As estimativas de Pontos de Função (PF) e de Pontos de Caso de Uso (PCU) do projeto estão disponíveis na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE I – LISTA DE VERIFICAÇÃO DA QUALIDADE

A lista de verificações de qualidade do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE J – DIAGRAMA DE PACOTES

O diagrama de pacotes do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE K – DIAGRAMA DE CLASSES

O diagrama de classes do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE L – DIAGRAMA DE OBJETOS

Um diagrama de objetos do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE M – DIAGRAMA DE SEQUÊNCIA

Os diagramas de sequência estão disponíveis na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE N – DIAGRAMA DE VISÃO GERAL DA INTERAÇÃO

O diagrama de visão geral da interação está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE O – DIAGRAMA DE ATIVIDADES

Um diagrama de atividades do projeto está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE P – MODELO OPERACIONAL DOS DADOS

O modelo operacional dos dados está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE Q – DICIONÁRIO DE DADOS

O dicionário de dados está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE R – DIAGRAMA DE COMPONENTES

O diagrama de componentes está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE S – ESTRUTURA FÍSICA DO BANCO DE DADOS

A estrutura física do banco de dados está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE T – MANUAIS DE INSTALAÇÃO E CONFIGURAÇÃO

Os manuais de instalação e configuração estão disponíveis na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE U – MANUAL DO USUÁRIO

O manual do usuário está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE V – DIAGRAMA DE IMPLANTAÇÃO

O diagrama de implantação está disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE X – DIAGRAMAS UML UTILIZADOS NESTE PROJETO

A descrição completa dos diagramas UML que foram utilizados neste projeto encontra-se disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE W– HISTÓRICO DE REALIZAÇÃO

O histórico de realização encontra-se disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE Y– ENTREVISTA COM O USUÁRIO

A entrevista com os perfis de consumidores do projeto EzMart encontra-se disponível na pasta APÊNDICE do arquivo anexado a este documento.

# APÊNDICE Z– SOLICITAÇÕES DE MUDANÇAS NO PROJETO

Todas as solicitações de mudança que ocorreram no projeto EzMart encontram-se disponíveis na pasta APÊNDICE do arquivo anexado a este documento.

**APÊNDICE Z – SOLICITAÇÕES DE MUDANÇAS DO PROJETO**

As solicitações de mudança que ocorreram ao longo do projeto estão disponíveis na pasta APÊNDICE do arquivo anexado a este documento.