UNIVERSIDADE FEDERAL DE JUIZ DE FORA
FACULDADE DE ENGENHARIA
GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

Gabriela da Silva Lima

Proposta para agilidade nas compras de supermercado por encomenda ou por aplicativo de entrega: planejamento de aplicativo

Juiz de Fora
2020

Gabriela da Silva Lima

Proposta para agilidade nas compras de supermercado por encomenda ou por aplicativo de entrega: planejamento de aplicativo

Trabalho de Conclusão de Curso apresentado à Faculdade de Engenharia da Universidade Federal de Juiz de Fora, como requisito parcial para a obtenção do grau de bacharel em Engenharia de Produção.

Orientadora: Prof.ª Dr.ª Roberta Pereira Cavalcanti Nunes.

Juiz de Fora

2020

Ficha catalográfica elaborada através do programa de geração
automática da Biblioteca Universitária da UFJF,
com os dados fornecidos pelo(a) autor(a)

Lima, Gabriela da Silva.
Proposta para agilidade nas compras de supermercado por encomenda ou por aplicativo de entrega : planejamento de aplicativo / Gabriela da Silva Lima. -- 2020.
57 f.

Orientadora: Roberta Pereira Cavalcanti Nunes
Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Juiz de Fora, Faculdade de Engenharia, 2020.

1. Supermercado. 2. Aplicativo. 3. Planejamento. I. Nunes, Roberta Pereira Cavalcanti, orient. II. Título.

Gabriela da Silva Lima

Proposta para agilidade nas compras de supermercado por encomenda ou por aplicativo de entrega: planejamento de aplicativo

Trabalho de Conclusão de Curso apresentado à Faculdade de Engenharia da Universidade Federal de Juiz de Fora, como requisito parcial para a obtenção do grau de bacharel em Engenharia de Produção.

Aprovada em 16 de dezembro de 2020.

BANCA EXAMINADORA

_____

Prof.ª Dr.ª Roberta Pereira Cavalcanti Nunes – Orientadora

Universidade Federal de Juiz de Fora

_____

Prof.ª Dr.ª Cristina Márcia Barros de Castro

Universidade Federal de Juiz de Fora

_____

Prof.ª Dr.ª Clarice Breviglieri Porto

Universidade Federal de Juiz de Fora

AGRADECIMENTOS

Agradeço a Deus, por me permitir viver tudo isso e sempre guiar o meu caminho.

Agradeço aos meus pais, Simone e Júlio, e meu irmão, Gustavo, que durante toda essa trajetória estiveram ao meu lado e deram-me todo o suporte que eu precisei e mais um pouco. Tenho certeza de que devo essa conquista a eles.

Agradeço aos amigos que fizeram com que esses anos na faculdade fossem leves e gostosos de serem vividos.

Agradeço aos professores, por todo o aprendizado transmitido. Em especial, à Roberta, que diante de um grande desafio fez com que este trabalho fosse possível. Eterna gratidão.

Agradeço às instituições das quais fiz parte durante a faculdade, por terem sido peças fundamentais no meu desenvolvimento e aprendizado – SEEPRO, GET Produção, Mais Consultoria JR – e pelas empresas nas quais fiz estágio.

RESUMO

O presente texto tem como objetivo o planejamento de um aplicativo com a função de reduzir o tempo despendido no processo de coleta de itens em supermercados dada uma lista de compras. Nessa aplicação o processo de coleta de itens é realizado por um colaborador de uma empresa de entrega ou por encomenda. A demanda por compras em supermercados via *delivery* cresceu de forma substancial nos últimos tempos e, principalmente, durante a pandemia do Covid-19, no ano de 2020, uma vez que o isolamento social levou ao aumento desse setor. Com isso, houve uma mudança no comportamento do consumidor, o qual pós-pandemia continuará a demandar por esse serviço. As atividades de planejamento são compostas da especificação, com o estudo de viabilidade e engenharia de requisitos, e projeto, com a projeto arquitetural, projeto de interface e projeto de componente. Dessa forma, essas etapas foram desenvolvidas chegando-se à etapa final de construção da proposta de interfaces e descrição de todos os componentes necessários para que os requisitos dos usuários fossem atendidos, com posterior avaliação do resultado alcançado através de 10 Heurísticas de Usabilidade de Nielsen. Assim, como próximos passos, tem-se a implementação do que foi planejado e proposto. A metodologia de estudo foi de natureza aplicada, objetivo exploratório e possui uma abordagem quantitativa.

Palavras-chave: Supermercado. Aplicativo. Planejamento.

ABSTRACT

The aim of this text is to plan an application with the goal to reduce the time consumed in the process of gathering items in supermarkets when one is given a shopping list. In this application, the aforementioned process is carried out by an employee of a delivery company or by customer ordering. The demand for grocery delivery services has grown substantially in recent times, especially, during the Covid-19 pandemic, in the year 2020, since social isolation has led to an increase in the sector. As a result, there has been a change in consumer behavior, which will continue to demand for this particular service. The planning activities are composed of the specification, with the feasibility study and requirements engineering, and design, with the architectural design, interface design and component design. In this manner, these steps were developed in order to reach the final stage of construction of the proposed interfaces and description of all the necessary components so that the users' requests were met, with subsequent evaluation of the result achieved through Nielsen's 10 Usability Heuristics. Therefore, as the next step, there is the implementation of what was planned and proposed. The research methodology used in this study include the applied research methodology, exploratory objective and quantitative approach.

Keywords: Supermarket. App. Planning.

LISTA DE ILUSTRAÇÕES

Figura 1 – A organização do MVC 17
Figura 2 – Exemplo do Problema do Caixeiro Viajante 22
Figura 3 – Requisitos das partes envolvidas 26
Figura 4 – Comunicação de dados entre partes envolvidas 26
Figura 5 – Tela Inicial – Módulo Supermercado 32
Figura 6 – Cadastrar *Layout* – Módulo Supermercado 32
Figura 7 – Cadastrar Produto – Módulo Supermercado 33
Figura 8 – Cadastrar Colaborador – Módulo Supermercado 34
Figura 9 – Cadastrar Colaborador – Módulo Supermercado 35
Figura 10 – Relatórios – Módulo Supermercado 36
Figura 11 – Tela Inicial – Módulo Colaborador 37
Figura 12 – Pedido – Módulo Colaborador 38
Figura 13 – Relatar Anomalia – Módulo Colaborador 39
Figura 14 – Relatórios – Módulo Colaborador 40
Figura 15 – Tela Inicial – Módulo Empresa de Entrega 41
Figura 16 – Cadastrar Supermercado – Módulo Empresa de Entrega 42
Figura 17 – Cadastrar Colaborador – Módulo Empresa de Entrega 43
Figura 18 – Relatório – Módulo Empresa de Entrega 44
Figura 19 – Tela Inicial – Módulo Colaborador 45
Figura 20 – Cadastrar Lista de Compras 46
Figura 21 – Acompanhar Pedidos 47
Figura 22 – Exemplo: arranjo físico com demandas assinaladas 50
Figura 23 – Exemplo: arranjo físico com demandas assinaladas e rota definida 50

LISTA DE QUADROS

Quadro 1 – O padrão MVC 16
Quadro 2 – Requisitos Funcionais Gerais 28
Quadro 3 – Requisitos Funcionais para os Supermercados 29
Quadro 4 – Requisitos Funcionais para os Colaboradores 29
Quadro 5 – Requisitos Funcionais para as Empresas de Entrega 30
Quadro 6 – Requisitos Funcionais para os Clientes 30
Quadro 7 – Requisitos Não-Funcionais 30

LISTA DE ABREVIATURAS E SIGLAS

CNPJ	Cadastro Nacional da Pessoa Jurídica
CPF	Cadastro de Pessoal Física
ERP	Enterprise Resource Planning
MVC	Model Veiw Controller
NP	Non-Deterministic Polynomial Time
PVC	Problema do Caixeiro Viajante
QR	Quick Response

SUMÁRIO

1 INTRODUÇÃO 10
1.1 CONSIDERAÇÕES INICIAIS 10
1.2 JUSTIFICATIVA 11
1.3 ESCOPO DO TRABALHO 11
1.4 ELABORAÇÃO DOS OBJETIVOS 12
1.5 DEFINIÇÃO DA METODOLOGIA 12
1.6 ESTRUTURA DO TRABALHO 13
2 REVISÃO DA LITERATURA 14
2.1 PLANEJAMENTO DE UM APLICATIVO 14
2.1.1 Especificação 15
2.1.2 Projeto e implementação 16
2.1.3 Validação 18
2.1.4 Evolução 18
2.2 ARRANJO FÍSICO 19
2.3 PROBLEMA DO CAIXEIRO VIAJANTE (PCV) 21
3 DESENVOLVIMENTO 25
3.1 PROPOSTA 25
3.2 ESTUDO DE VIABILIDADE 26
3.3 ENGENHARIA DE REQUISITOS 28
3.4 PROJETO DE INTERFACE E PROJETO DE COMPONENTE 31
3.4.1 Módulo Supermercado 31
3.4.2 Módulo Colaborador 36
3.4.3 Módulo Empresa de Entrega 41
3.4.4 Módulo cliente 44
3.5 HEURÍSTICAS DE USABILIDADE DE NIELSEN 47
3.6 APLICAÇÃO DO CAIXEIRO VIAJANTE EM UMA ÁREA DE SUPERMERCADO 49
4 CONSIDERAÇÕES FINAIS 52
REFERÊNCIAS 53

1 INTRODUÇÃO

1.1 CONSIDERAÇÕES INICIAIS

A *web* tornou-se uma tecnologia indispensável em uma diversidade de áreas e modificou a forma como diversas atividades eram executas (PRESSMAN, 2015). Portanto, é uma ferramenta de suporte relevante devido à necessidade de uma rápida disseminação de informações e um novo padrão de relacionamento através dela. Os aplicativos são meios que permitem essa disseminação de informação. Para Sommerville (2008), não existe um processo ideal para o desenvolvimento de aplicativo. Cada empresa utiliza os recursos que desejar e desenvolve seu processo conforme especificações do sistema a ser desenvolvido e interesse da organização. De acordo com o objetivo em questão, há diversas formas de proceder esse desenvolvimento.

Segundo Tompkins (1996), 50% do tempo total de coleta de um pedido no armazém é gasto com viagens, assim, ao se reduzir as distâncias há uma melhora considerável na eficiência da coleta de pedidos. Da mesma forma ocorre durante a coleta de itens em um supermercado. Os supermercados apresentam um arranjo físico funcional, o qual também pode ser chamado de arranjo físico por processo, e tem como objetivo o grupamento de itens com funções ou processos similares (CORRÊA; CORRÊA, 2009).

Dessa forma, diante de uma lista de compras ampla, diversos setores devem ser percorridos para coleta de itens da lista. Esse problema assemelha-se ao Problema do Caixeiro Viajante, o qual consiste na determinação do circuito mais curto para percorrer um determinado número de pontos, passando uma vez em cada ponto e saindo e retornando à origem (HILLIER; LIEBERMAN, 2006).

No ano de 2020, com a pandemia do Covid-19, a realização de compras em supermercados por *delivery*, seja por aplicativo, seja por encomenda, cresceu de forma substancial com a necessidade do distanciamento social. Dentre as funções do Engenheiro de Produção dado o desenvolvimento de um aplicativo, tem-se a garantia de que todos os requisitos das partes interessadas sejam atendidos no projeto e implementação e aplicação de conhecimento sobre processos, a fim de criar um fluxo de trabalho eficiente, reduzindo custos e ganhando produtividade.

1.2 JUSTIFICATIVA

Em uma pesquisa realizada no final do mês de março de 2020, de acordo com reportagem do **Portal G1**, a consultoria Ebt/Nielsen constatou que, após o início da pandemia do Covid-19, as compras em varejo de autosserviço, ou seja, de supermercado, realizadas de forma *online* cresceram 96%. Além disso, na mesma reportagem, em uma entrevista com Bernardo Ouro Preto, executivo do St, Marche, ele afirma "Nós temos dois grandes modelos no varejo: centro de distribuição e loja mais próxima. Para nossa empresa, o modelo da loja mais próxima tem sido o mais vitorioso". Como em diversas outras empresas, o modelo de entrega pode partir de uma loja ou de um centro de distribuição. Com o aumento da demanda, também houve um aumento relevante nos atrasos das entregas desse setor (TOOGE, 2020).

O principal motivo desse atraso é a complexidade do processo de realizar compras, selecionando item a item de uma lista de compra do cliente. Apesar de ser um processo simples, torna-se complexo a partir da consideração de todas a variáveis envolvidas no problema. Assim, as empresas que fornecem esse serviço necessitam de meios para agilizar o processo e garantir a qualidade.

Em pesquisa realizada pela Social Miner, em parceria com a Opinion Box, **O futuro do consumo num cenário pós-Covid-19**, 62,7% dos entrevistados irão fazer compras de mercado/ feira tanto em lojas físicas, quanto em lojas *online*, e 10,9% irão consumir apenas em lojas *online* (SOCIAL MINER, 2020). A pretensão é que com o crescimento do setor e o ganho de representatividade deste haja a otimização dos processos que envolvem esse serviço para garantia de melhorias e qualidade do serviço.

O tema foi escolhido pelo crescimento do setor pós pandemia da Covid-19, em 2020, e por um interesse pessoal pela otimização de processos para melhoria do desempenho das companhias. A resposta do aplicativo proposto seria a ordem de coleta para redução do tempo despendido no processo de compras, indo de encontro com a otimização dos processos.

1.3 ESCOPO DO TRABALHO

Para Toth e Vigo (2002), o uso de procedimentos computadorizados para o planejamento da distribuição geram grandes economias para as corporações. Dessa forma, em um ambiente onde a coleta, no lugar da distribuição, é representativa, o uso desses procedimentos também pode gerar grandes economias. Dessa forma, o presente trabalho aborda a proposta de planejamento de um aplicativo, o qual terá como finalidade a definição

da ordem de coleta de itens em um supermercado, dada uma lista de compras. As etapas de desenvolvimento de um modelo piloto e validação do modelo não compõem o escopo do estudo.

O método usado para o desenvolvimento do aplicativo foi o Método Incremental, derivado do Método de Cascata e baseia-se em entregas parciais que incrementam a entrega anterior e a entrega final. Durante o desenvolvimento do estudo foram usados os conhecimentos da autora sobre o setor e técnicas observacionais dos processos, sem colaboração prática das empresas envolvidas. O projeto arquitetural será baseado em um padrão já existente, o padrão MVC (*Model View Controller*), o qual é dividido em três componentes lógicos, a saber: o Modelo, responsável pelo sistema de dados e comunicação com o banco de dados; a Visão, responsável pela apresentação dos dados na interface com o usuário; o Controlador, responsável pela entrada de dados pelo usuário e interações com o Modelo e Visão (SOMMERVILLE, 2011).

1.4 ELABORAÇÃO DOS OBJETIVOS

O objetivo geral do presente estudo é a proposta de planejamento de um aplicativo com a função de reduzir o tempo despendido no processo de coleta de itens em um supermercado, dada uma lista de compras, quando esse processo é realizado por um colaborador de uma empresa de entrega ou por um colaborador do próprio supermercado. Dessa maneira, atividades para o planejamento de um aplicativo foram desenvolvidas com o intuito de garantir uma proposta que possa embasar o desenvolvimento piloto do aplicativo.

1.5 DEFINIÇÃO DA METODOLOGIA

O estudo tem natureza aplicada, uma vez que é direcionado para um setor específico, o setor de varejo de autosserviço, e objetiva gerar soluções para aplicações práticas em um problema real (GERHARDT; SILVEIRA, 2009). Além disso, há um objetivo exploratório, uma vez que usa como base o referencial bibliográfico para o desenvolvimento do presente estudo. Por fim, possui uma abordagem qualitativa, pois faz uma análise qualitativas dos ganhos obtidos através do presente estudo (GERHARDT; SILVEIRA, 2009).

Com a proposta desta pesquisa, buscou-se a aplicação do conhecimento científico na prática. Assim, tem-se como referência bibliográfica para o desenvolvimento do trabalho o Problema do Caixeiro Viajante, o qual determina a melhor rota para visitar *n* pontos, saindo e

retornando à origem, levantando-se os métodos de solução do problema, com Goldbarg e Luna (2005) e Hillier e Lieberan (2006) como principais autores. Ademais, apresentou-se informações acerca de arranjo físico, levantando características inerentes ao arranjo físico de um supermercado, o qual possui um arranjo físico funcional, com Corrêa e Corrêa (2009) e Peinado e Graeml (2007) como principais autores. Por fim, foram usadas metodologias para o desenvolvimento de um aplicativo, levantando-se métodos e atividades que compõem esse processo, com Sommerville (2011) e Pressman (2015) como principais autores.

1.6 ESTRUTURA DO TRABALHO

O primeiro capítulo do trabalho versa sobre a introdução do estudo, contendo as considerações iniciais, a justificativa da definição do presente tema, o escopo do trabalho, os objetivos do trabalho e a metodologia de estudo. O segundo capítulo apresenta a revisão bibliográfica sobre o tema de estudo, na qual é abordada a teoria sobre o Problema do Caixeiro Viajante, Arranjo Físico e Desenvolvimento de um aplicativo. O terceiro capítulo apresenta o desenvolvimento do trabalho, seguindo a metodologia de trabalho com fases do desenvolvimento do aplicativo objeto desse estudo. E, por fim, serão apresentadas as considerações finais.

2 REVISÃO DA LITERATURA

2.1 PLANEJAMENTO DE UM APLICATIVO

O desenvolvimento de um aplicativo deve ocorrer por meio de uma equipe multidisciplinar a fim de atender o objetivo final e o problema que se deseja solucionar de forma clara e completa. De acordo com Laudon e Laudon (2007), para se identificar um problema, é importante a existência de um consenso com relação às causas e possíveis soluções, considerando os recursos limitados disponíveis. Os autores consideram as três dimensões mais comuns de problemas em empresas: a organização, a tecnologia e as pessoas. Essas dimensões são bons guias para o entendimento dos problemas observados. Após o entendimento do problema ou questão-chave, deve-se definir as informações necessárias para solucioná-lo, ter clareza acerca de qual o objetivo que se deseja alcançar ao solucionar o problema e definir o que pode ser feito para solucioná-lo.

A Engenharia de *software* surgiu com o intuito de estabelecer métodos, ferramentas e técnicas a fim de tornar o processo de desenvolvimento mais econômico e eficiente (SEABRA, 2013). Para Vasconcelos (2006), o desenvolvimento de *software* tem como objetivo o atendimento das necessidades de clientes e usuários através da criação de um sistema, sendo essencial a definição clara dos requisitos das partes interessadas.

Para Pressman (2015), *software* são instruções executadas que fornecem o desempenho e o resultado desejado, dados estruturados que possibilitam a manipulação das informações e documentos que definem o uso do programa. Além disso, ainda segundo o autor, é sempre feito sobre medida, ou seja, é sempre feito para atender uma demanda bem específica. Sommerville (2011) ainda ressalta que não existe um processo ideal para o desenvolvimento de um *software*, mas cada empresa utiliza dos recursos que desejar e desenvolve o seu processo de acordo com especificações do sistema a ser desenvolvido. Todavia, no geral, é possível encontrar atividades em comum em todos os casos, são elas: Especificação, com o Estudo de Viabilidade e a Engenharia de Requisitos; Projeto e Implementação, com o Projeto Arquitetural, o Projeto de Interface, o Projeto de Componentes e a Codificação; Validação, com o Teste de desenvolvimento, os Testes de Sistema e o Teste de Aceitação; Evolução.

2.1.1 Especificação

Processo de definição das funcionalidades necessárias no sistema, sendo assim um processo crítico, pois um erro nessa etapa corrobora para futuros problemas no projeto e na implementação. Tem por finalidade definir um documento de requisitos que atendam a todas as partes interessadas no sistema. Há a definição das funcionalidades e suas constantes de operação. Composto pelas seguintes etapas:

- Estudo de Viabilidade:

Considera se as necessidades dos usuários serão atendidas dadas as tecnologias disponíveis de *software* e *hardware*. O estudo tem como resposta se o projeto deve seguir ou não de acordo com o escopo do projeto. Sommerville (2008, p. 97) considera que, para se obter as informações sobre escopo do projeto, faz-se necessário que algumas perguntas sejam respondidas pelo cliente, por exemplo:

1. Como a organização se comportaria se esse sistema não fosse implementado?
2. Quais são os problemas com os processos atuais e como o novo sistema ajudaria a reduzir esses problemas?
3. Qual será a contribuição direta do sistema para os objetivos e requisitos da empresa?
4. As informações podem ser transferidas e recebidas de outros sistemas da organização?
5. O sistema requer tecnologia que ainda não foi usada na organização?

Assim, caso o sistema contribua com os objetivos da organização, deve-se seguir seu desenvolvimento.

- Engenharia de Requisitos:

Objetiva especificar os requisitos do projeto. De acordo com Young (2003), a Engenharia de Requisitos consiste na definição dos objetivos de um sistema, considerando suas características, atributos, capacidade e fatores de qualidade. Nessa fase, diversas técnicas de elicitação podem ser utilizadas: técnicas de elicitação conversacionais, que considera uma conversa direta como forma básica de interação (SERNA, 2012); técnica de elicitação observacionais, que é a observação de processos como forma de auxiliar na descoberta de requisitos (SUTCLIFFE; SAWYER, 2013); técnicas de elicitação sintéticas, que combinam canais de comunicação diferentes (SERNA, 2012). Sommerville (2011) afirma que os requisitos de um *software* são subdivididos em funcionais e não funcionais. Os requisitos funcionais determinam o que o *software* deve ser capaz de fazer, determinando as funções

presentes no sistema, suas entradas e saídas. Já os não funcionais determinam as propriedades e restrições da operação. Após o levantamento dos requisitos, deve-se criar o documento de requisitos. Na etapa de validação dos requisitos, estes devem ser analisados para que sejam encontradas inconsistências, ambiguidades e omissões, podendo ser feita de forma formal ou informal.

2.1.2 Projeto e implementação

- Projeto Arquitetural:

Consiste no desenvolvimento do conceito para o sistema com a descrição dos componentes e como os componentes trabalham em conjunto. De acordo com Pressman (2006), um estilo arquitetural é o modo como um sistema é estruturado e organizado. Assim, existem algumas taxonomias de estilos arquiteturais, são eles: arquitetura centrada em dados, arquitetura de fluxo de dados, arquitetura de chamada e retorno (arquitetura orientada a objetos e arquitetura em camadas). Padrões arquiteturais tem como base o tripé problema-solução-contexto e são considerados uma solução bem estabelecida para problemas arquiteturais. Para Sommerville (2011), um padrão de arquitetura pode ser definido como uma descrição estilizada, abstrata e de boas práticas fazendo com que seja um sistema bem-sucedido em outros sistemas. O padrão MVC (*Model View Controller*) está definido no Quadro 1.

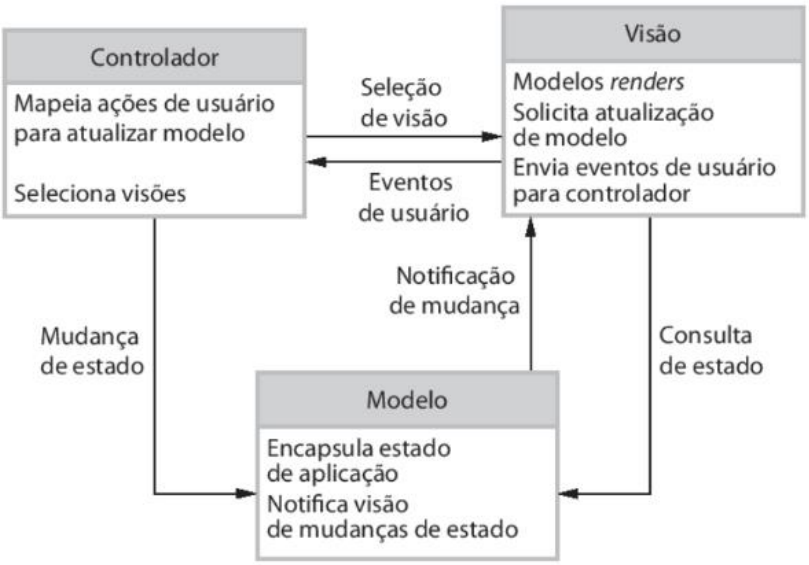
Quadro 1 – O padrão MVC

Nome	**MVC (Modelo-Visão-Controlador)**
Descrição	Separa a apresentação e a integração dos dados do sistema. O sistema é estruturado em três componentes lógicos que interagem entre si. O componente Modelo gerencia o sistema de dados e as operações associadas a esses dados. O componente Visão define e gerencia como os dados são apresentados ao usuário. O componente Controlador gerencia a interação do usuário (por exemplo, teclas, cliques do mouse etc.) e passa essas interações para a Visão e o Modelo.
Quando é usado	É usado quando existem várias maneiras de se visualizar e interagir com dados. Também quando são desconhecidos os futuros requisitos de interação e apresentação de dados.
Vantagens	Permite que os dados sejam alterados de forma independente de sua representação, e vice-versa. Apoia a apresentação dos mesmos dados de maneiras diferentes, com as alterações feitas em uma representação aparecendo em todas elas.
Desvantagens	Quando o modelo de dados e as interações são simples, pode envolver código adicional e complexidade de código.

Fonte: Sommerville (2011).

Ainda de acordo com Sommerville (2011), a independência e separação são essenciais para padrões de arquitetura, pois permitem alterações localizadas quando necessário. Padrão de arquitetura em camadas auxilia nessa independência entre os módulos do sistema; o MVC é um exemplo de um estilo de arquitetura em camadas. A Figura 1 salienta essa independência entre as funcionalidades.

Figura 1 – A organização do MVC



Fonte: Sommerville (2011).

- Projeto de Interface:

Definição das interfaces do sistema, o meio pelo qual ocorre a diálogo entre uma aplicação e o usuário. Pressman (2011) define interface como o elemento mais importante de um produto. Uma interface mal projetada interfere na capacidade do usuário utilizar todo o potencial do sistema. Nielsen (1994) apresenta as 10 Heurísticas de Usabilidade de Nilesen que levanta alguns pontos a serem avaliados com relação ao *design*: visibilidade do estado do sistema, correspondência entre sistemas e mundo real, controle e liberdade para o usuário, consistência e padronização, prevenção de erros, reconhecimento em vez de memorização, flexibilidade e eficiência de utilização, *design* estético e minimalista, que ajude os usuários a reconhecer, diagnosticar e recuperar erros e ajuda e documentação.

- Projeto De Componente:

Considera-se cada componente do sistema e define-se como será seu funcionamento.

- Codificação:

Implementação do sistema com o código de programação. Nessa etapa, conhecimentos sobre modelos de otimização devem ser utilizados se necessário para o objetivo em questão. No presente estudo conhecimentos a certa do Problema do Caixeiro Viajante serão utilizados a fim de definir a melhor ordem de coleta dos itens. O PCV será abordado posteriormente.

2.1.3 Validação

Há a verificação do atendimento das necessidades dos usuários.

- Testes de Desenvolvimento:

As pessoas que fizeram o desenvolvimento do *software* devem testar cada um dos componentes de forma separada e/ou grupamento podem ser feitos a fim de serem testados.

- Testes do Sistema:

Consiste na integração dos módulos de forma homogênea, a fim de encontrar erros decorrentes da integração entre os componentes. Valida se os requisitos funcionais e não funcionais foram atendidos.

- Teste de Aceitação:

Os testes, nessa etapa, são realizados com dados reais e não com dados simulados. Esses testes podem revelar falhas ou erros de requisitos não observados anteriormente.

2.1.4 Evolução

Há a adaptação de acordo com a mudanças demandadas pelo usuário, com a presença de um ciclo interativo.

O desenvolvimento de um *software* pode ser dividido, segundo Souza Neto (2004) e Soares (2004), em duas possíveis áreas de atuação. São elas:

a. Desenvolvimento tradicional: pautada na análise e no projeto, caracterizada por grande quantidade documentação e padronização, o que favorece a ocorrência de mudanças futuras;

b. Desenvolvimento ágil: pautada na adaptação constante de acordo com requisitos do usuário, porém, como ponto negativo tem-se a deficiência em documentos e contratos claros.

De acordo com Pressman (2015), os métodos prescritos ou modelos clássicos buscam garantir uma estrutura bem definida no desenvolvimento do sistema e têm interesse em definir os limites do sistema antes de iniciar a fase desenvolvimento. O Modelo em Cascata é um exemplo desse método. A partir de uma evolução do Modelo de Cascata surgiu o Modelo Incremental. De acordo com Sommerville (2008), nesse modelo, as funcionalidades são liberadas de forma mais rápida, uma vez que entregas parciais são planejadas, as quais vão incrementando a entrega final periodicamente. As necessidades do usuário são priorizadas a fim de definir a ordem das entregas.

Como método mais atual há as Metodologias Ágeis. Segundo Pressman (2015), essa metodologia tem foco na entrega rápida de uma aplicação e faz com que o cliente se torne parte da equipe de desenvolvimento. Com essa metodologia, há uma maior valorização das pessoas e suas relações, do funcionamento do sistema, da colaboração do cliente e da agilidade na resposta a mudanças (SABBAGH, 2013). Assim, para o planejamento de um aplicativo é importante considerar todas as variáveis relacionadas com o objetivo que se deseja alcançar. Dessa forma, para o presente estudo conhecimentos acerca de arranjo físico são necessários para atingir o objetivo proposto.

2.2 ARRANJO FÍSICO

Arranjo físico é a disposição de equipamentos em uma área, com o objetivo de equilibrar a ambientação, movimentação e produção (ANTON; EIDELWEIN; DIEDRICH, 2012). O arranjo físico é de extrema importância para um processo de produção e para a armazenagem, pois a organização do espaço faz com que ele seja usado de forma mais eficiente. O *layout* ou arranjo físico diz respeito ao planejamento da ocupação dos espaços físicos e é composto pela forma como as máquinas e equipamentos estão dispostas na empresa (CHIAVENATO, 2005). Um arranjo físico adequado apoia a estratégia competitiva da operação, melhora os níveis de eficiência e eficácia (CORRÊA; CORRÊA, 2009).

Segundo Wanke e Magalhães (2012), as escolhas de *layout* devem gerar a maior eficiência possível na movimentação de materiais, o que pode ser percebido através de menores necessidades, espaço e mão de obra, flexibilidade em caso de eventuais mudanças no

layout e de deslocamentos e menores despesas com equipamentos. De acordo com Dias (1993), em um ambiente de armazenamento, para que a armazenagem seja feita de forma mais econômica e com maior rendimento é necessário a integração do fluxo de materiais e de equipamentos de movimentação contando com a disposição de homens, máquinas e materiais.

Para um local de armazenagem, Moura (1998) salienta que é a demanda dos materiais estocados e suas possíveis flutuações que determinam os espaços necessários para cada item, bem como a ocupação alternativa para espaços ociosos. Já para Castiglioni (2010), o *layout* para um local de armazenagem é determinante para saber o grau de acessibilidade do material, as áreas obstruídas, o fluxo de material, eficiência da mão de obra, entre outros.

Segundo Tompkins (1996), 50% do tempo total de coleta de um pedido no armazém é gasto com viagens, reduzindo-se as distâncias percorridas há uma melhora considerável na eficiência da coleta de pedidos. Dukic e Opetuk (2014) afirmam que há uma influência não desprezível do *layout* na eficiência na coleta de pedidos.

Slack (2013) salienta que o arranjo físico funcional está ligado às funções que o arranjo exercer e qual sua finalidade. O autor usa como exemplo um supermercado, em que é mais conveniente o agrupamento de alguns produtos, como vegetais congelados e enlatados, já que podem exigir tecnologias e gabinetes específicos.

O arranjo físico funcional também pode ser chamado de arranjo físico por processo, o qual tem como objetivo o grupamento de itens com funções ou processos similares (CORRÊA; CORRÊA, 2009). Os autores citados ainda exemplificam esse arranjo através de lojas de departamento, supermercados e hospitais, em que há o agrupamento de itens por setores similares (feminino, masculino, material de limpeza, congelados, ortopedia, radiologia). Entre as características presentes no arranjo físico funcional estão as seguintes: alto manuseio de materiais, controle de produção complexo, presença de máquinas de uso geral, grande quantidade de material em processo, baixo investimento inicial, *layout* flexível (GORGULHO, 2010).

São consideradas por Peinado e Graeml (2007) vantagens do arranjo físico funcional:

- Flexibilidade para atender mudanças do mercado;
- Maior margem do produto;
- Consegue atender uma diversidade de produtos e quantidade ao mesmo tempo;
- Baixo investimento para instalação do parque industrial;
- Bom nível de motivação.

Para os autores há os seguintes pontos como desvantagem do arranjo físico funcional:

- Longos deslocamentos dentro da fábrica;
- Dificuldade de balanceamento;
- Baixa diluição de custo fixo em função de menor expectativa de produção;
- Necessidade elevada de *setup* de máquina;
- Necessidade de mão-de-obra qualificada.

Assim, dado o arranjo físico funcional, pode-se destacar como uma característica importante os longos deslocamento dentro de supermercados, o que gera um elevado tempo despendido com deslocamentos nesse tipo de arranjo físico. Dessa forma, o uso métodos de otimização devem ser levado em consideração, como o Problema do Caixeiro Viajante.

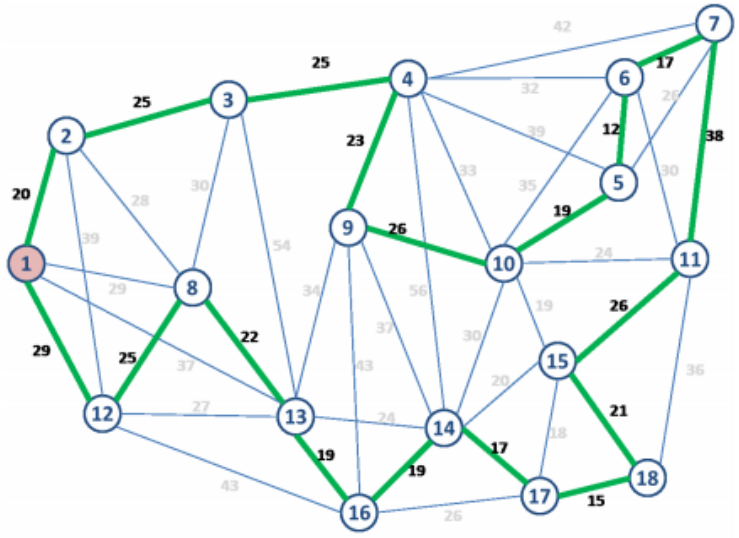
2.3 PROBLEMA DO CAIXEIRO VIAJANTE (PCV)

O Problema do Caixeiro Viajante consiste na determinação do circuito mais curto para percorrer um determinado número de pontos, passando uma vez em cada ponto e saindo e retornando à origem (HILLIER; LIEBERMAN, 2006). Esse problema pode ser modelado como um grafo G = (V, A), no qual deseja-se determinar um caminho hamiltoniano de custo mínimo. Um caminho hamiltoniano é um circuito que passa por todos os nós de um grafo uma única vez (GOLDBARG; LUNA, 2005).

A metodologia para solução desse problema de forma simplificada seria definir todas as possíveis soluções e, assim, determinar qual tem o menor custo. Nessa situação, com o auxílio do computador, seriam definidas todas as rotas possíveis e seus respectivos comprimentos; o menor valor seria selecionado com solução ótima. Assim, ao considerar todas as rotas, é possível afirmar que um problema de otimização está reduzido a um problema de enumeração (TAUFER; PEREIRA, 2011).

A Figura 2, a seguir, é um exemplo para o Problema do Caixeiro Viajante (PCV) não direcionado, ou seja, o custo de ir do nó 1 para o nó 2 é o mesmo de ir do nó 2 para o nó 1. O valor do arco pode representar o tempo, o custo ou a distância entre os nós, dependendo a aplicação desejada.

Figura 2 – Exemplo do Problema do Caixeiro Viajante



Fonte: Póvoa (2001).

Por meio desse problema, é possível modelar diversas situações, como: planejamento e logística e desenvolvimento de *chips*. Além disso, metáforas podem ser usadas, considerando os nós como cidades, pontos de solda ou fragmentos de DNA (FREITAS, 2009). Portanto, sua aplicação é vasta e variada. Goldbarg e Luna (2005) apresentam o modelo de programação abaixo para o Problema do Caixeiro Viajante, o qual é modelado como um problema de programação linear inteira. Assim, dado um grafo G = (V, A), temos:

Minimizar $$z = \sum_{j=1}^{n} \sum_{i=1}^{n} c_{ij} x_{ij} \quad (1)$$

Sujeito a $$\sum_{i=1}^{n} x_{ij} = 1, \forall\, j \in V \quad (2)$$

$$\sum_{j=1}^{n} x_{ij} = 1, \forall\, i \in V \quad (3)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1, \forall\, S \subset V \quad (4)$$

$$x_{ij} \in \{1,0\}, \forall\, i,j \in V \quad (5)$$

No qual, c_{ij} é o custo para percorrer o arco (i,j); x_{ij} é uma variável binária e assume valor igual a 1, se o arco $(i,j) \in A$ for escolhido para compor a solução, caso contrário,

assume valor igual a 0; S é um subgrafo de G, no qual $|S|$ representa a quantidade de vértices que compõe esse subgrafo. No problema a restrição (2) obriga que exatamente um arco chegue no vértice j; (3) obriga que exatamente um arco saia no vértice i; (4) impedem a existência de subciclos no problema, além de definir que $x_{ii} = 0$ quando $|S| = 1$; (5) garante que as variáveis x_{ij} são binárias e não negativas.

O PCV é um problema NP-difícil, ou seja, é um problema com tempo polinomial não determinístico. Assim, demanda o uso de heurísticas ou meta-heurísticas para solução do problema. Os algoritmos exatos são capazes de garantir uma solução ótima para o problema, mas, normalmente, possuem um alto custo computacional. Já os métodos heurísticos e meta-heurísticos não são capazes de garantir uma solução ótima, mas resultam uma solução sub-ótima, com um esforço computacional menor que nos modelos exatos. Essa solução sub-ótima, geralmente, é de qualidade e próxima da solução ótima.

De acordo com Cunha (2000), a importância do estudo e aplicação do roteamento se dá pela importância da logística para o mercado e pela sua complexidade, o que torna impossível a definição da solução ótima em problemas reais. Isso faz com que cada vez mais heurísticas e meta-heurísticas sejam estudadas. Segundo Silva (2010), heurísticas são procedimentos de aproximação, sendo um algoritmo de busca capaz de localizar soluções aproximadas de boa qualidade e em tempo computacional razoável. No geral, não é possível determinar quão longe a solução está em relação à solução ótima do problema.

Para Reeves (1993), um benefício da aplicação das heurísticas é a possibilidade de modelar problemas mais próximos da realidade, algo que muitas vezes não é possível em algoritmos exatos, dado que são capazes de operar restrições e função objetivo mais complexas. Os métodos heurísticos podem ser divididos em dois métodos: os métodos clássico, capazes de encontrar uma boa solução em um tempo reduzido, porém não explora de forma profunda todo o espaço de busca; as meta-heurísticas, que, em sua maioria, incorporam métodos clássicos, mas diferente dos métodos clássico, faz uma exploração aprofundada no espaço de busca.

Como já mencionado, a principal diferença entre as heurísticas e as meta-heurísticas é que a última realiza uma exploração aprofundada no espaço de busca. Dessa forma, o surgimento das meta-heurísticas se deu através da necessidade da diminuição dos riscos de paradas em ótimos locais que estejam distantes do ótimo global. Em geral, componentes aleatórios são utilizados durante a execução do algoritmo, o que faz com que outras soluções

sejam exploradas. As meta-heurísticas têm sido usadas de forma ampla para problemas NP-difícil, como é o caso do PCV, por encontrarem resultados satisfatórios na busca pela solução próxima da ótima. Segundo Reeves (1993), as meta-heurística são eficientes na busca por soluções de qualidade quando há um espaço de solução muito grande.

3 DESENVOLVIMENTO

3.1 PROPOSTA

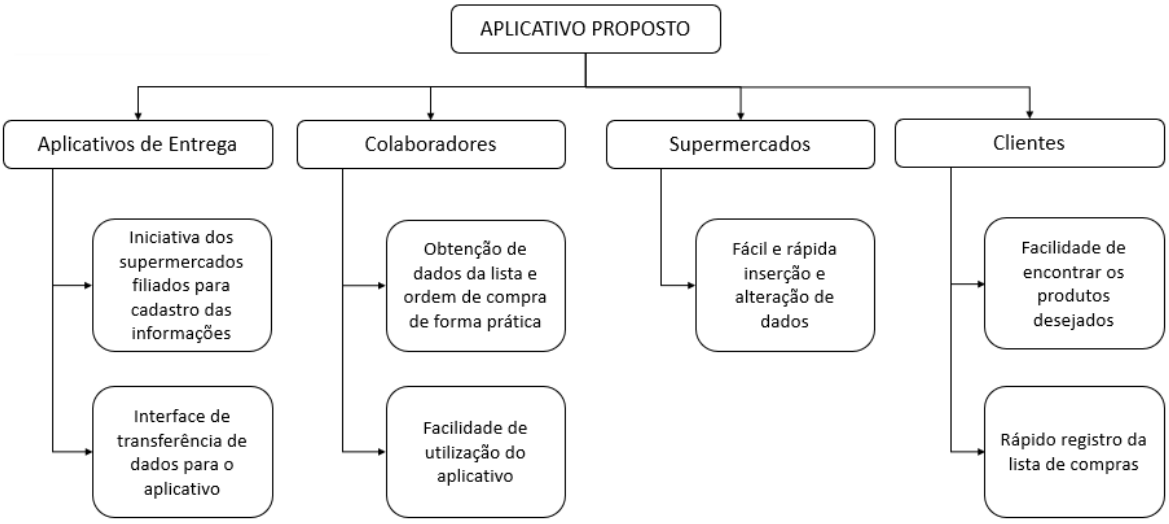
O problema a ser solucionado é a demora na realização do processo de compras em supermercados, dada uma lista de compras de um cliente, realizada por um colaborador de uma empresa de entregas ou de um supermercado. Essa demora, em geral, é devida aos seguintes fatores:

- Variedade de supermercados que oferecem esse serviço, principalmente nos grandes centros. Nesse caso, os colaboradores das empresas de entrega têm dificuldade de conhecer todos os arranjos físicos;
- Dificuldade de localização dos produtos dentro do supermercado;
- Dificuldade de identificar os produtos de forma correta, o que gera um erro nos itens com relação a lista de compras solicitada.

Dessa maneira, a solução geral para o problema é o planejamento de um aplicativo que oriente os colaborados dentro dos problemas supracitados, com o desenvolvimento do aplicativo a ser feito em trabalhos futuros. Deve haver um suporte para os colaboradores, a fim de saber qual a melhor ordem de coleta dos itens, de acordo com suas localizações no supermercado, para cada um dos supermercados visitados por ele, e a conferência dos itens através de código de barra ou *QR Code* a fim de que erros referentes à lista de compras não ocorram. Os colaboradores envolvidos podem ser *motoboys,* funcionários da empresa de entrega ou funcionários do supermercado.

Para que a proposta gere frutos de melhoria de processos, a iniciativa deve partir dos supermercados, levando em consideração os benefícios para seus clientes, uma vez que há necessidade de cadastro de diversas informações essenciais por parte dos supermercados. Assim, deve haver o interesse de todas as partes com o benefício que se propõe alcançar. Na Figura 3 estão os principais requisitos das partes envolvidas no aplicativo proposto.

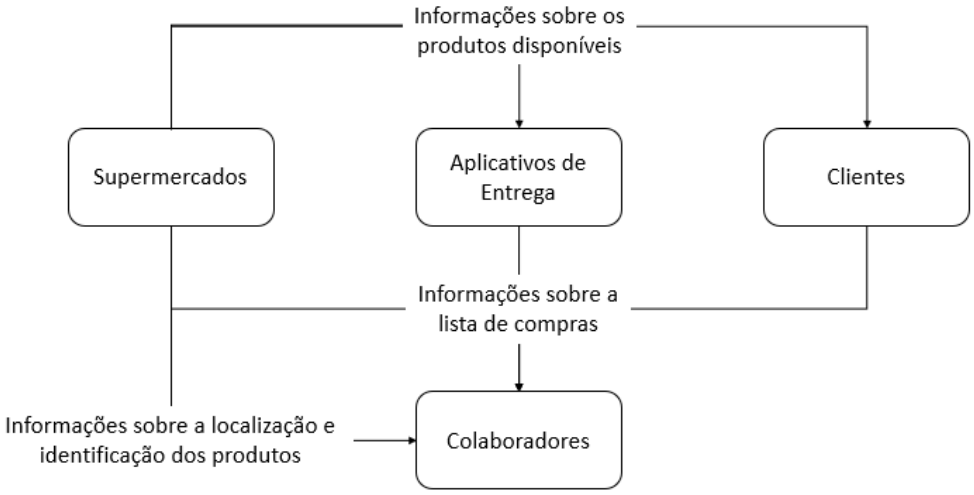
Figura 3 – Requisitos das partes envolvidas



Fonte: Elaborada pela autora (2020).

A Figura 4, abaixo, salienta as informações transferidas entre os módulos. Cada parte envolvida contribui com uma informação para o atingimento do objetivo final.

Figura 4 – Comunicação de dados entre partes envolvidas



Fonte: Elaborada pela autora (2020).

3.2 ESTUDO DE VIABILIDADE

O estudo de viabilidade tem como resposta se o projeto deve seguir ou não de acordo com o escopo do projeto. Para essa análise, as perguntas propostas por Sommerville (2008, p. 97) para o entendimento completo acerca do escopo do projeto serão respondidas. Após a

resposta das questões, caso o sistema contribua com os objetivos que se propõe e estejam alinhadas com os processos que se deseja melhorar, deve-se seguir seu desenvolvimento. Uma vez que o aplicativo tem como objetivo servir à empresa de prestação de serviço de entrega por aplicativo e supermercados, esses serão usadas como referência para as respostas das perguntas a seguir. As respostas foram feitas pautadas no conhecimento da autora acerca do processo das empresas e dos processos dessas empresas.

1- Como a organização se comportaria se esse sistema não fosse implementado?

   Sem a utilização desse sistema, a organização permaneceria com um processo sem o ganho de produtividade, com demora nas entregas e insatisfação dos clientes decorrentes da entrega de pedidos com itens errados.

2- Quais são os problemas com os processos atuais e como o novo sistema ajudaria a reduzir esses problemas?

   Os problemas com o processo atual são, como supracitados, a demora na entrega e a entrega de pedidos com itens errados. Assim, para cada um dos problemas tem-se os seguintes métodos para solução. Para a demora na entrega, tem-se como foco principal a utilização de modelos de otimização de rotas, o Problema do Caixeiro Viajante, o qual tem como objetivo determinar o caminho mais curto para a coleta dos itens demandados pelo cliente. Para a entrega de pedidos com itens errados, o sistema contará com um método de conferência dos itens coletados através de código de barras do produto ou *QR Code*.

3- Qual será a contribuição direta do sistema para os objetivos e requisitos da empresa?

   Os serviços de entrega têm como objetivo ofertar uma alta variedade de produtos e com qualidade no serviço prestado. Com relação à qualidade no serviço prestado, a solução apresentada neste estudo atua na rapidez da entrega e na garantia da entrega de todos os itens solicitados pelo cliente e, assim, maior controle do estoque.

   Analisando com relação ao cliente, ele também busca por qualidade e, além disso, agilidade no serviço de entrega. Já os colaboradores, buscam por facilidade na realização do processo de coleta de itens em um supermercado.

4- As informações podem ser transferidas e recebidas de outros sistemas da organização?

   A proposta é que as informações necessárias para o aplicativo em desenvolvimento sejam recebidas através do próprio aplicativo, sem necessidade de recebimento e transferência de informações de outros sistemas. Apenas as informações referentes à lista de compras deveriam ser recebidas do aplicativo de entrega.

5- O sistema requer tecnologia que ainda não foi usada na organização?

Tem-se como previsão que as tecnologias necessárias para a implementação do aplicativo, de programação de aplicativos e mecanismos de otimização já são utilizados pelas empresas do setor, não necessitando de novos investimentos.

Sendo assim, após as respostas das perguntas propostas por Sommerville (2008) observa-se que o aplicativo está alinhado aos objetivos e necessidades das empresas e proporcionará um melhor atendimento e maior satisfação do cliente final. Portanto, deve-se seguir com o projeto de acordo com o escopo pensado.

3.3 ENGENHARIA DE REQUISITOS

Os requisitos de um aplicativo são subdivididos em requisitos funcionais e requisitos não funcionais. No presente estudo, a técnica de elicitação observacional, a qual usa a observação dos processos como forma de auxiliar na descoberta dos requisitos, foi utilizada para identificação dos requisitos do sistema. Dessa forma, a observação dos processos foi o principal método para identificação e descoberta de requisitos. Além disso, a perspectiva como cliente dos aplicativos de *delivery* também foi considerada.

Os requisitos funcionais estão apresentados nos quadros a seguir, são eles: Quadro 2, com os requisitos gerais; Quadro 3, com os requisitos para os supermercados; Quadro 4, com os requisitos para os colaboradores; Quadro 5, com os requisitos para as empresas de entrega; Quadro 6, com os requisitos dos clientes. Os requisitos não-funcionais estão apresentados no Quadro 7. Os requisitos servirão de base para o planejamento das interfaces e componentes.

Quadro 2 – Requisitos Funcionais Gerais

GERAL	RF001	Variedade de acessos	O aplicativo deve contar com três tipos de acesso. Um acesso para os supermercados, outro para os colaboradores e outro para a empresa prestadora do serviço de entregar. Cada um deve atender as demandas especificas dos usuários.
	RF002	Base de dados de localização gerida por código de barras	O aplicativo deve contar com informações sobre a posição dos itens no arranjo físico do supermercado através de referências por código de barras. Vale ressaltar a dificuldade na obtenção desse tipo de informação em alguns supermercados.

Fonte: Elaborado pela autora (2020).

Quadro 3 – Requisitos Funcionais para os Supermercados

PARA OS SUPERMERCADOS	RF003	*Login*	Todo usuário do aplicativo deve contar com seu registro de *Login*. Deve-se contar com a funcionalidade "Esqueci minha senha".
	RF005	Cadastrar *Layout*	Cada supermercado deve cadastrar seu *layout* para que seja possível calcular as distâncias entre gôndolas. Também deve-se coletar as informações sobre os a quantidade de andares das prateleiras.
	RF004	Cadastrar Produtos	Cada supermercado deve cadastrar sua lista de produtos. Assim, deve conter além de dados acerca do produto, dados acerca da localização do produto dentro do espaço físico do supermercado. Deve-se ainda contar com a opção de alterar o cadastro e deletar o cadastro.
	RF006	Lista de Compras	Para o caso de o supermercado ter o desejo de utilizar o aplicativo sem a participação do aplicativo de entrega, deve-se inserir a lista de compra que se deseja coletar, a qual será enviada para um colaborador selecionado nessa etapa.
	RF007	Registrar Colaborador	Realizar o cadastro dos colaboradores vinculados ao supermercado. Um colaborador pode estar vinculado a mais de uma empresa. Deve haver a opções de alterar e excluir cadastro.
	RF008	Relatórios	Relatórios de desempenho devem ser fornecido para os supermercados a fim de compreenderem seu desempenho e seu desempenho comparado com outros supermercados.

Fonte: Elaborado pela autora (2020).

Quadro 4 – Requisitos Funcionais para os Colaboradores

PARA OS COLABORADORES	RF009	*Login*	Todo usuário do aplicativo deve contar com seu registro de *Login*. Deve-se contar com a funcionalidade "Esqueci minha senha".
	RF010	Pedido	Página com informações referentes ao pedido que está em andamento. Informações sobre o cliente, supermercado e lista de compras. Nessa função, o aplicativo deve receber da empresa prestadora do serviço de entrega ou do supermercado a lista de compras. O usuário deve ser capaz de alterar a lista de compras.
	RF011	Relatar anomalia	Possibilidade de relatar anomalias em caso de problemas durante a realização da compra. Assim, problemas como item no local errado ou itens faltantes podem ser relatados e posteriormente tratados.
	RF012	Relatórios	Relatórios de desempenho devem ser fornecido para os colaboradores a fim de compreenderem seu desempenho e seu desempenho comparado com outros colaboradores.

Fonte: Elaborado pela autora (2020).

Quadro 5 – Requisitos Funcionais para as Empresas de Entrega

PARA A EMPRESA DE ENTREGA	RF013	*Login*	Todo usuário do aplicativo deve contar com seu registro de *Login*. Deve-se contar com a funcionalidade "Esqueci minha senha".
	RF014	Cadastrar supermercados	Realizar o cadastro dos supermercados a fim de vincular o mesmo com a empresa de entrega. Um supermercado pode estar vinculado a mais de uma empresa ou não estar vinculado à nenhuma. Deve haver a opção de alterar e excluir cadastro.
	RF015	Cadastrar colaborador	Realizar o cadastro dos colaboradores a fim de vinculá-lo com a empresa de entrega no aplicativo. Um colaborador pode estar vinculado a mais de uma empresa. Sendo assim, o colaborador pode ser vinculado ao supermercado e/ou ao aplicativo de entrega. Deve haver a opções de alterar e excluir cadastro.
	RF016	Relatórios	Relatórios de desempenho devem ser fornecidos para as empresas de entrega para compreender seu desempenho.

Fonte: Elaborado pela autora (2020).

Quadro 6 – Requisitos Funcionais para os Clientes

PARA A EMPRESA DE ENTREGA	RF013	*Login*	Todo usuário do aplicativo deve contar com seu registro de *Login*. Deve-se contar com a funcionalidade "Esqueci minha senha".
	RF014	Lista de compras	Para o caso de o cliente desejar registrar a lista de compras, deve-se inserir a lista de compra que se deseja coletar com todas as informações sobre ele para o um colaborador faço o serviço.
	RF015	Acompanhamento dos Pedidos	O cliente deve poder acompanhar os pedidos em andamento e já realizados.

Fonte: Elaborado pela autora (2020).

Quadro 7 – Requisitos Não-Funcionais

REQUISITOS NÃO-FUNCIONAIS	NF001	Usabilidade	A interface deve ser de fácil utilização e de rápido entendimento, uma vez que atenderá a diversas pessoas. Além disso, como possui a finalidade de agilizar o processo de compras, não pode gerar consumo excessivo de tempo para o uso.
	NF002	Desempenho	Possui relação direta com a qualidade do aplicativo. Além disso, será utilizado por diversas pessoas ao mesmo tempo, o que demanda um bom desempenho.
	NF003	Segurança	Uma vez que o aplicativo conta com informações de empresas e pessoas, a segurança é de extrema importância. Os dados fornecidos pelos usuários devem estar seguros.
	NF004	Manutenção	Faz-se necessário um suporte ao usuário constante em caso de possíveis problemas com o aplicativo. Esses devem ser resolvidos o mais rápido possível para não atrapalhar o dia a dia dos colaboradores e resultados das empresas de entrega.

Fonte: Elaborado pela autora (2020).

3.4 PROJETO DE INTERFACE E PROJETO DE COMPONENTE

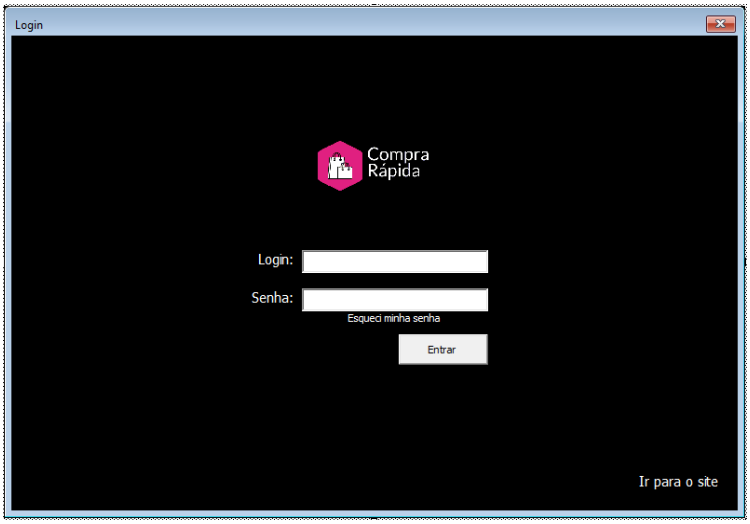
As interfaces do sistema são o meio de diálogo da aplicação com o usuário. A interface deve contar com um bom *design* a fim de garantir a usabilidade do sistema e o uso completo, por parte do usuário, de todo o potencial do sistema. Juntamente com a interface, o projeto de componente será desenvolvido com a descrição dos componentes presente nas interfaces desenvolvidas e seu funcionamento. O uso por parte dos supermercados e por parte das empresas de entrega se dará por meio de computador, já o uso por parte do colaborador será por meio de celular. As interfaces que seguem são apenas propostas, considerando as necessidades dos usuários e não um modelo final a ser desenvolvido.

3.4.1 Módulo Supermercado

O módulo tem como objetivo atender ao supermercado de acordo com seus requisitos, os quais englobam, primordialmente, a facilidade na inserção de dados e alteração desses. Além disso, os requisitos funcionais do aplicativo serviram como guia para formulação das interfaces. Assim, esse módulo é composto pela Tela Inicial (Figura 5), Cadastro de *Layout* (Figura 6) e Cadastro de Produto (Figura 7), que servem como base para a definição da ordem de coleta dos itens; Lista de Compras (Figura 8), que coleta informações sobre a lista a ser coletada pelos colaboradores quando não há a participação da empresa de entrega; Cadastro de Colaborador (Figura 9) e Relatórios (Figura 10).

- Tela inicial:

Figura 5 – Tela Inicial – Módulo Supermercado

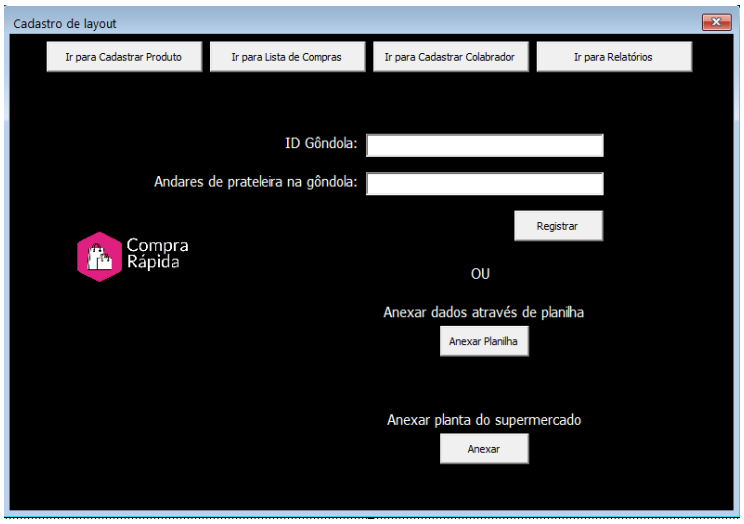


Fonte: Elaborada pela autora (2020).

A tela inicial tem por finalidade solicitar ao usuário suas informações para que o *login* seja feito. Além disso, opções de alterar senha, em caso de esquecimento, e ir para o *site* do aplicativo estariam disponíveis. No *site* o usuário teria a opção de conhecer mais sobre a empresa, consultar as funcionalidades do aplicativo, consultar dúvidas frequentes, solicitar suporte, entre outros.

- Cadastrar *Layout*:

Figura 6 – Cadastrar Layout – Módulo Supermercado

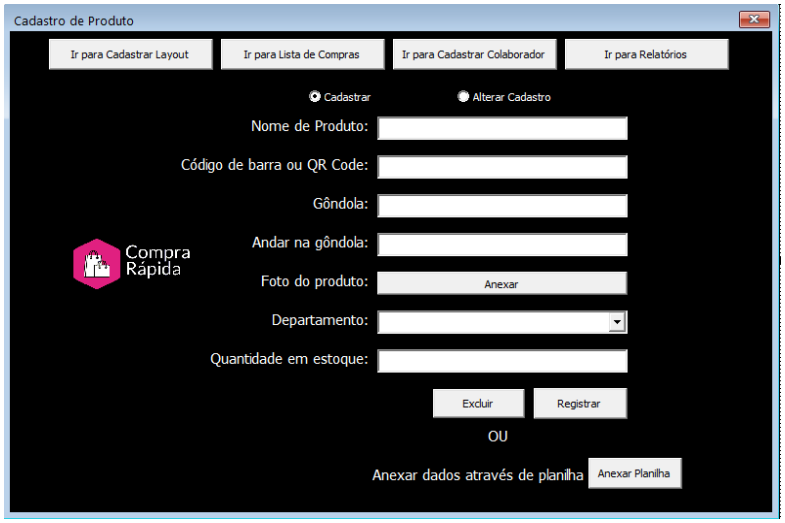


Fonte: Elaborada pela autora (2020).

Os dados das gôndolas devem ser inseridos, diretamente no sistema ou utilizando planilha no Excel. Além disso, há a necessidade de anexar a planta do supermercado a fim fazer adaptações no modelo de cálculo das distâncias entre pontos. As duas informações são necessárias. Em todas as páginas, os botões superiores têm a finalidade de migrar para as outras telas no aplicativo quando desejado.

- Cadastrar Produto:

Figura 7 – Cadastrar Produto – Módulo Supermercado



Fonte: Elaborada pela autora (2020).

Os produtos devem ser cadastrados a fim de coletar, principalmente, os dados de código de barra ou *QR Code* e local, gôndola, andar na gôndola e quantidades em estoque. Essas informações serão utilizadas para melhorar a qualidade das compras e diminuir o tempo despendido com essa atividade. Para anexar a foto, o usuário deve clicar em “Anexar” e será dado a ele a opções de tirar a foto ou de escolher uma foto no dispositivo utilizado. Além disso, deve-se inserir a quantidade de itens em estoque para que seja possível saber quais produtos estão disponíveis para os clientes, a atualização da quantidade de itens em estoque deve ser feita através de vínculo com o ERP da empresa.

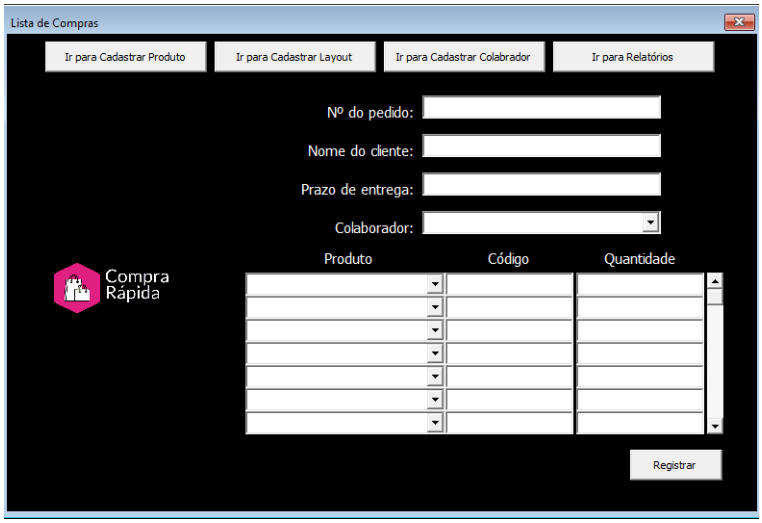
O cadastro também pode ser feito através de uma planilha com os dados na mesma ordem que solicitada pelo sistema. Quando muitos produtos forem cadastrados essa alternativa será melhor, uma vez que despende menos tempo. Para alterar cadastro, basta

inserir o nome do produto ou código de barras ou *QR Code* que as outras informações serão atualizadas e poderão, então, serem alteradas.

Para excluir o cadastro, o botão “Excluir” deve ser usado. Se esse for usado antes do registro, as informações presentes nos campos serão apagadas. Se for usado após o registro, quando o usuário estiver alterando um cadastro, o produto será excluído do aplicativo. Para que um item seja excluído a quantidade em estoque deve ser igual a zero.

- Lista de Compras:

Figura 8 – Cadastrar Colaborador – Módulo Supermercado



Fonte: Elaborada pela autora (2020).

O cadastro de “Lista de Compras” deve ser feito quando o supermercado tiver uma lista de compras a ser entregue por encomenda e desejar utilizar das funcionalidade dos aplicativo sem a participação da empresa de entrega. Assim, o usuário deve fazer o registro das informações referentes a lista que se deseja coletar. Após, os dados serão enviados para o colaborador selecionado para que faça a coleta do itens.

- Cadastrar Colaborador:

Figura 9 – Cadastrar Colaborador – Módulo Supermercado

Fonte: Elaborada pela autora (2020).

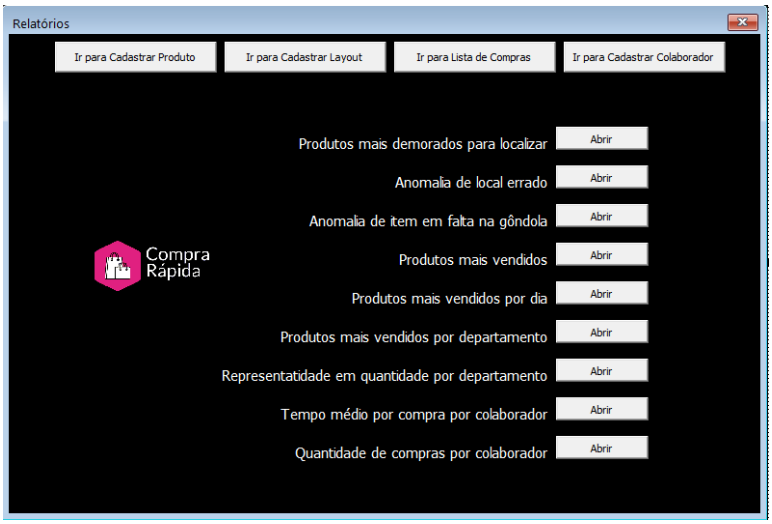
Os supermercados devem cadastrar as informações dos colaboradores para fins de controle interno e, principalmente, para que um *login* seja criado para cada colaborador novo que ingressar no aplicativo. Em “Foto”, o usuário tem a opção de tirar a foto ou selecionar uma foto do dispositivo. Deve-se inserir a origem dos colaboradores, ou seja, se ele é um funcionário do supermercado, um funcionário terceirizado ou um colaborador da uma empresa de entrega.

O usuário ainda tem opções de Cadastrar ou Alterar Cadastro já existente. No caso de “Alterar Cadastro”, deve-se pesquisar os cadastros existentes pelo Nome do Colaborador ou pelo CPF. Caso o usuário tenha o desejo de excluir um cadastro, deve pesquisar em “Alterar Cadastro” e selecionar o botão “Excluir”.

- Relatórios:

Nesta página, o usuário conta com alguns relatórios com informações interessantes para a empresa acompanhar. Relatórios customizados também podem ser solicitados pela empresa.

Figura 10 – Relatórios – Módulo Supermercado



Fonte: Elaborada pela autora (2020).

As informações inseridas no módulo do supermercado servirão para o funcionamento completo dos outros módulos, uma vez que todos os dados coletados nesse módulo compõem os dados necessários para a solução final de definição das rotas de coleta. Os dados relativos à lista de compras serão combinados com os dados de *layout* a fim de formular a rota de coleta ideal a ser seguida. Os dados de *layout* são registrados no módulo do supermercado e os dados de lista de compras registradas no módulo de supermercado, módulo cliente ou enviados pela empresa de entrega. Com esses dados, o colaborador recebe as informações acerca da ordem de coletas dos itens através do módulo do colaborador, como segue na próxima subseção.

3.4.2 Módulo Colaborador

O módulo do colaborador é composto pela resposta final do problema que se deseja solucionar, a ordem de coleta de itens, sendo essa a função principal do módulo. Para isso, o aplicativo deve contar uma interface de fácil utilização, composta por todas as informações que podem ser necessárias nesse processo, usando-se assim, todo o espaço de tela do *smartphone* de forma otimizada. Assim, esse módulo é composto pela Tela Inicial (Figura 11), Pedido (Figura 12), com a resposta final do aplicativo, pela página de Registro de Anomalias (Figura 13) e Relatórios (Figura 14).

- Tela inicial:

Figura 11 – Tela Inicial – Módulo Colaborador

Fonte: Elaborada pela autora (2020).

A tela inicial tem por finalidade solicitar ao usuário suas informações para que o *login* seja feito. Opções de alterar senha, em caso de esquecimento de senha, e ir para o *site* para conhecer mais sobre a empresa também estão disponíveis.

- Pedido:

Figura 12 – Pedido – Módulo Colaborador

Fonte: Elaborada pela autora (2020).

O pedido em andamento será mostrado para o colaborador na ordem correta de coleta, de acordo com os itens solicitados na lista de compra do cliente. O aplicativo deve possuir interligação com o aplicativo da empresa de entrega, a fim de receber a informações do pedido de forma automática, quando a lista de compras partir do aplicativo de entrega. Assim, todas as informações necessárias serão mostradas. O local do produto será sinalizado e deve servir para orientação do usuário. O local do produto será composto pelo local no supermercado e pelo andar da prateleira que se encontra o produto, sendo assim em uma perspectiva 3D.

Durante a compra, o colaborador deve conferir o Código de Barras ou *QR Code* do produto, devendo preencher o campo de verificação, escaneando o código de barras ou *QR Code* com a câmera do celular ou digitando o código para validação da coleta do item. Em caso de dúvidas, o colaborador também pode ampliar a imagem do produto para facilitar a identificação.

Para alterar a lista de compras, existem três opções. Para excluir o produto, o colaborador deve clicar nele e a opção de excluir será mostrada. Para alterar a quantidade do item, o colaborador deve clicar nele e a opção de alterar quantidade será mostrada, bem como a opção de excluir. Para inserir itens, o colaborador tem a opção “Inserir Item” e, assim, poderá escolher um item e a quantidade deseja. Essas opções devem ser usadas apenas com o conhecimento do cliente final e, de preferência, no início das compras, para que o trajeto de coleta seja calculado gerando um maior ganho de produtividade. Além disso, a cada inserção ou exclusão de um item, a ordem de coleta dos itens é alterada para gerar um maior ganho.

- Relatar Anomalia:

Figura 13 – Relatar Anomalia – Módulo Colaborador

Fonte: Elaborada pela autora (2020).

O local para registro de anomalias é extremamente importante para que erros nas informações fornecidas pelo supermercado sejam corrigidos. Erros de localização, por exemplo, podem fazer com que toda a funcionalidade da ferramenta não gere grandes proveitos. Dessa forma, sempre que alguma informação do produto estiver errada, deve ser sinalizada para, posteriormente, ser tratada. Além disso, quando algum item da lista de

compras não estiver disponível no supermercado, uma anomalia também deve ser registrada para ser tratada posteriormente.

- Relatórios:

Nesta página, o usuário conta com alguns relatórios com informações interessantes para o colaborador acompanhar.

Figura 14 – Relatórios – Módulo Colaborador

Fonte: Elaborada pela autora (2020).

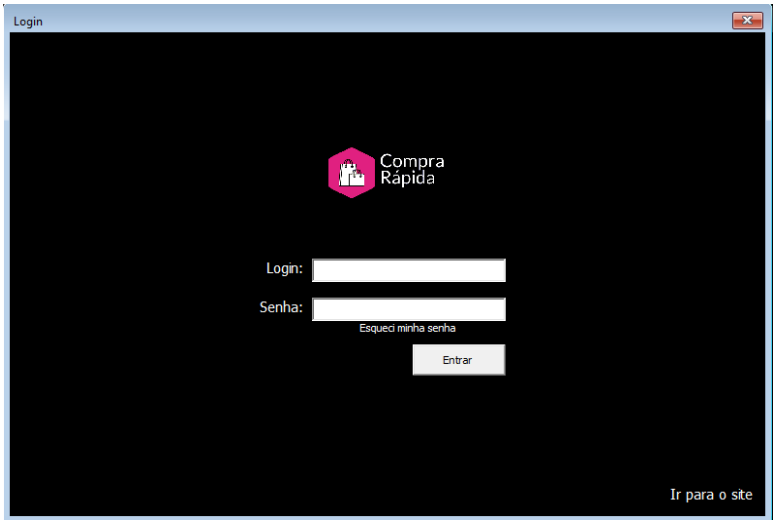
O colaborador pode receber as informações da lista de compras através de um registro do supermercado, cliente ou através da empresa de entrega, como já citado. No caso da empresa de entrega, os dados seriam recebidos de forma automática, sem necessidade de cadastro. Nessa situação, a empresa de entrega precisa registrar os colaboradores envolvidos nesse processo. Além disso, deve registrar supermercados vinculados e acompanhar relatórios de desempenho, como segue na próxima subseção.

3.4.3 Módulo Empresa de Entrega

O módulo da empresa de entrega faz a inserção de dados das partes envolvidas no processo do funcionamento do aplicativo. Assim, após o registro dos colaboradores e supermercados, o aplicativo deve contar com uma comunicação com o aplicativo da empresa de entrega para recebimento das informações de lista de compra e encaminhá-la para um colaborador registrado. Esse módulo possui Tela Inicial (Figura 15), Cadastro de Supermercado (Figura 16), Cadastro de Colaborador (Figura 17) e Relatórios (Figura 18).

- Tela inicial:

Figura 15 – Tela Inicial – Módulo Empresa de Entrega

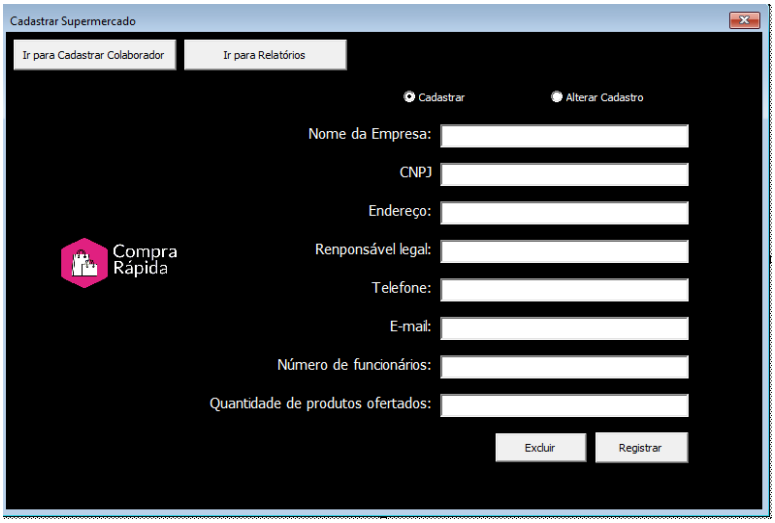


Fonte: Elaborada pela autora (2020).

A tela inicial tem por finalidade solicitar ao usuário suas informações para que o login seja feito. Opções de alterar senha em caso de esquecimento de senha e se ir para o site para conhecer mais sobre a empresa também estão disponíveis.

- Cadastrar Supermercado:

Figura 16 – Cadastrar Supermercado – Módulo Empresa de Entrega

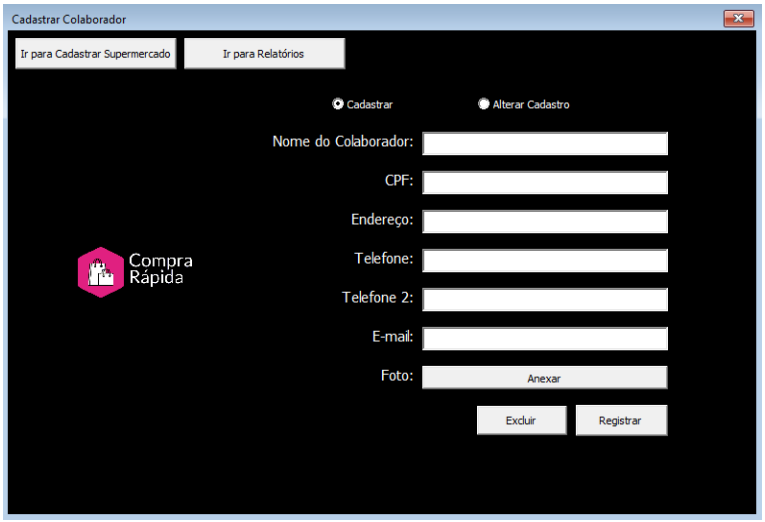


Fonte: Elaborada pela autora (2020).

As empresas de entrega devem cadastrar as informações dos supermercados para fins de controle interno e, principalmente, para vinculá-la com o aplicativo de entrega. O usuário tem opções de Cadastrar ou Alterar Cadastro já existente. No caso de “Alterar Cadastro”, deve-se pesquisar os cadastros existentes pelo Nome da Empresa ou pelo CNPJ. Caso o usuário tenha o desejo de excluir um cadastro, deve pesquisá-lo em “Alterar Cadastro” e selecionar o botão “Excluir”.

- Cadastrar Colaborador:

Figura 17 – Cadastrar Colaborador – Módulo Empresa de Entrega



Fonte: Elaborada pela autora (2020).

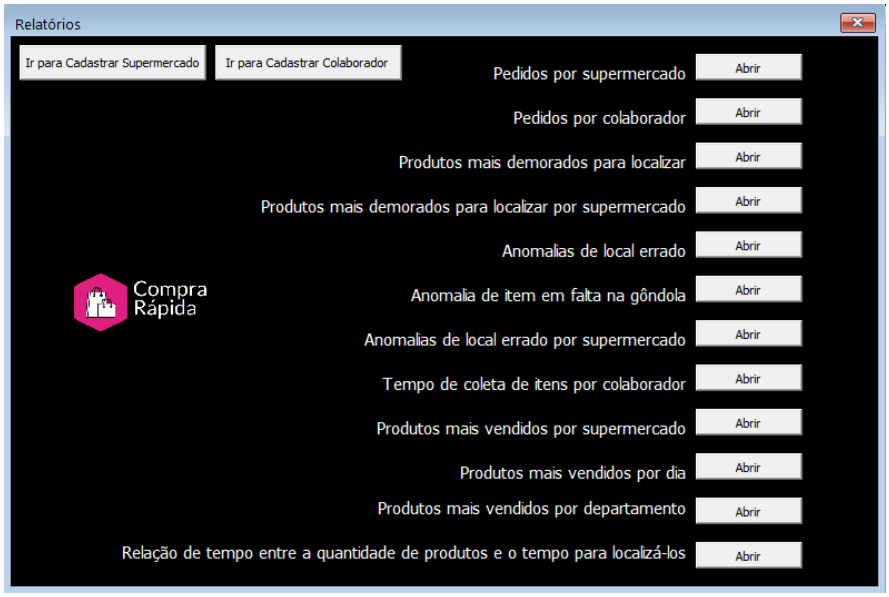
As empresas de entrega devem cadastrar as informações dos colaboradores para fins de controle interno e, principalmente, para que um *login* seja criado para cada colaborador novo que ingressar no aplicativo.

Em “Foto”, o usuário tem a opção de tirar a foto ou selecionar uma foto do dispositivo. O usuário ainda tem opções de Cadastrar ou Alterar Cadastro já existente. No caso de “Alterar Cadastro”, deve-se pesquisar os cadastros existentes pelo Nome do Colaborador ou pelo CPF. Caso o usuário tenha o desejo de excluir um cadastro, deve pesquisá-lo em “Alterar Cadastro” e selecionar o botão “Excluir”.

- Relatório:

Nesta página, o usuário conta com alguns relatórios com informações interessantes para a empresa acompanhar. Relatórios customizados também podem ser solicitados pela empresa.

Figura 18 – Relatório – Módulo Empresa de Entrega



Fonte: Elaborada pela autora (2020).

3.4.4 Módulo cliente

Esse módulo tem como objetivo possibilitar ao cliente registra sua lista de compras direto no aplicativo e assim será enviada para o supermercado selecionado. Para isso, o módulo é composto pelas seguintes telas: Tela Inicial (Figura 19), Cadastrar Lista de Compras (Figura 20) e Acompanhar Pedidos (Figura 21).

- Tela inicial:

Figura 19 – Tela Inicial – Módulo Colaborador

Fonte: Elaborada pela autora (2020).

A tela inicial tem por finalidade solicitar ao usuário suas informações para que o *login* seja feito. Opções de alterar senha, em caso de esquecimento de senha, e ir para o *site* para conhecer mais sobre a empresa também estão disponíveis.

Caso o usuário ainda não tenha cadastro no aplicativo, ele deve selecionar ‘Não sou Cadastrado’ e realizar o cadastro de acordo com as informações solicitadas.

- Cadastrar Lista de Compras

Figura 20 – Cadastrar Lista de Compras

Fonte: Elaborada pela autora (2020).

Para o cadastro da lista de compras o cliente deve inicialmente selecionar o supermercado que se deseja realizar a compras, as opções de supermercado serão mostradas para os usuários ao selecionar o campo de supermercado. Assim, a lista de produtos disponíveis será atualizada e com isso o usuário pode prosseguir com a escolha dos produtos. Ao clicar no campo de produtos a lista de produtos será mostrada ao cliente para que faça seleção dos itens e quantidades.

Por fim, o cliente deve inserir o endereço de entrega e a forma de pagamento. Ao clicar no campo de ‘Forma de Pagamento’, as opções serão mostradas e as informações referentes à forma de pagamento solicitadas.

Ao finalizar o pedido, o supermercado receberá um alerta. Assim, deve aceitar o pedido e selecionar o colaborador que deve realizar a coleta dos itens da lista de compras.

- Acompanhar Pedidos

Figura 21 – Acompanhar Pedidos

Fonte: Elaborada pela autora (2020).

Nessa tela o usuário pode acompanhar todos os pedidos já realizados. Inicialmente as informações gerais serão apresentadas e se o usuário desejar ver as informações sobre o pedido basta selecionar ‘Abrir’ e todas as informações sobre aquele pedido serão apresentadas. Para procurar o pedido desejado, basta deslizar a tela que outros pedidos serão mostrados.

As interfaces foram desenvolvidas de acordo com os requisitos de cada uma das partes envolvidas. Todavia, é importante uma avaliação da usabilidade e *design* da interface a fim de garantir o atendimento de todas as necessidades em termos de usabilidade. Assim, a avaliação por meio das 10 Heurísticas de Usabilidade de Nielsen segue na próxima sessão.

3.5 HEURÍSTICAS DE USABILIDADE DE NIELSEN

Nielsen (1994) apresenta as 10 Heurísticas de Usabilidade de Nielsen, que levantam alguns pontos a serem avaliados com relação ao *design* da interface, em um projeto de

aplicativo. Assim, a avaliação segundo esses pontos será feita a fim de identificar se o *design* feito atende às necessidades do aplicativo.

- Visibilidade do estado do sistema: o sistema irá contar com mensagens de informação para o usuário, quando necessário. No caso de cadastro de informações, quando este for feito o sistema irá mostrar uma mensagem com essa informação. Quando o sistema estiver carregando informações, essa mensagem também será mostrada. Dessa forma, os usuários estarão sempre informados do que está acontecendo.
- Correspondência entre sistemas e mundo real: o sistema apresenta termos e *design* de fácil entendimento para o usuário e conformidade com a realidade.
- Controle e liberdade para o usuário: com a interface desenvolvida, os usuários têm clareza das funcionalidades de cada parte do sistema e podem entrar e sair de forma fácil de cada uma das páginas. O usuário tem a liberdade para cancelar operações indesejadas antes que elas sejam finalizadas.
- Consistência e padronização: existe uma consistência entre as páginas do sistema, as quais apresentam uma padronização com relação aos termos utilizados, *design* e lógica de organização das informações.
- Prevenção de erros: com *design* simples e intuitivo, o sistema minimiza a possibilidade de erros. Além disso, mensagens de erro serão mostradas ao usuário quando uma informação for inserida de forma diferente do solicitado, como por exemplo, escrever um número em um campo de *e-mail*.
- Reconhecimento em vez de memorização: todos os objetivo e funcionalidade do sistema estão claros e intuitivos, não necessitando de memorização para o uso.
- Flexibilidade e eficiência de utilização: o sistema atende de forma simples e eficiente tanto o usuário experiente, quanto o inexperiente no uso do aplicativo.
- *Design* estético e minimalista: o *design* conta com apenas funcionalidade e informações extremamente necessárias para o uso do aplicativo, sem informações irrelevantes ou desnecessárias.
- Ajuda os usuários para reconhecer, diagnosticar e recuperar erros: as mensagens de erro elencadas anteriormente irão contar com a descrição do problema e da solução do problema através de uma linguagem simples.

- Ajuda e documentação: a documentação e registro das informações inseridas e acessos ao sistema serão feitas. O acesso a essas informações será restrito aos responsáveis de TI, caso necessário.

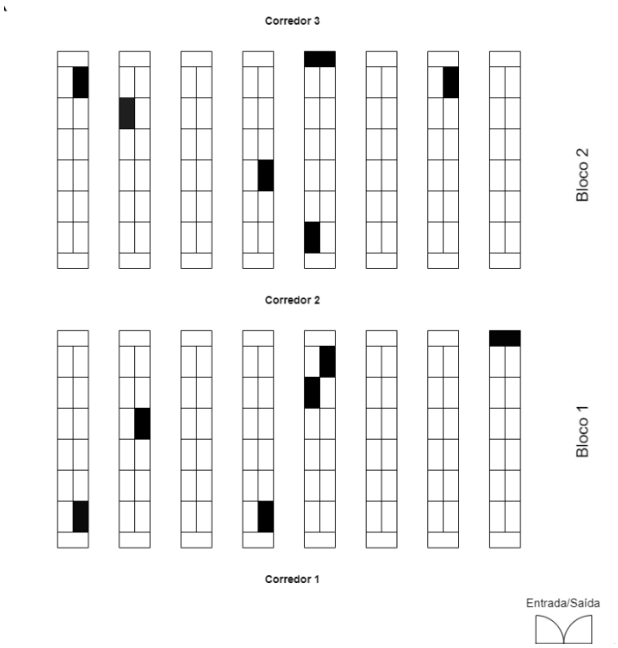
Dessa forma, as 10 Heurísticas de Usabilidade de Nielsen serviram como uma importante ferramenta para levantar demandas de um aplicativo e pensar em soluções para essas demandas. As soluções vão além das interfaces desenvolvidas e, com isso, o aplicativo apresentará uma maior facilidade no uso por parte do usuário, contando com prevenção de erros e melhor eficiência e eficácia em sua utilização, melhorando a experiência do usuário, colaboradores, empresas de entrega ou supermercados. Dados o planejamento de interface e as funcionalidades do aplicativo de acordo com o problema que se deseja solucionar, na próxima sessão, segue um exemplo prático para clareza da solução pretendida.

3.6 APLICAÇÃO DO CAIXEIRO VIAJANTE EM UMA ÁREA DE SUPERMERCADO

Supermercados contam com um arranjo físico funcional, com o objetivo de grupamento de itens com funções ou processos similares (CORRÊA; CORRÊA, 2009). Considerando as características do arranjo físico funcional levantadas por Peinado e Graeml (2007), algumas são diretamente relacionadas com o problema que se deseja solucionar. Assim, esse tipo de arranjo consegue atender uma diversidade de produtos e quantidade ao mesmo tempo (considerada uma vantagem) e longos deslocamentos dentro da fábrica (uma desvantagem desse arranjo). Essas características estão correlacionadas e são características que reforçam a necessidade do sistema, atuando para reduzir os deslocamentos decorrentes da grande diversidade de produtos.

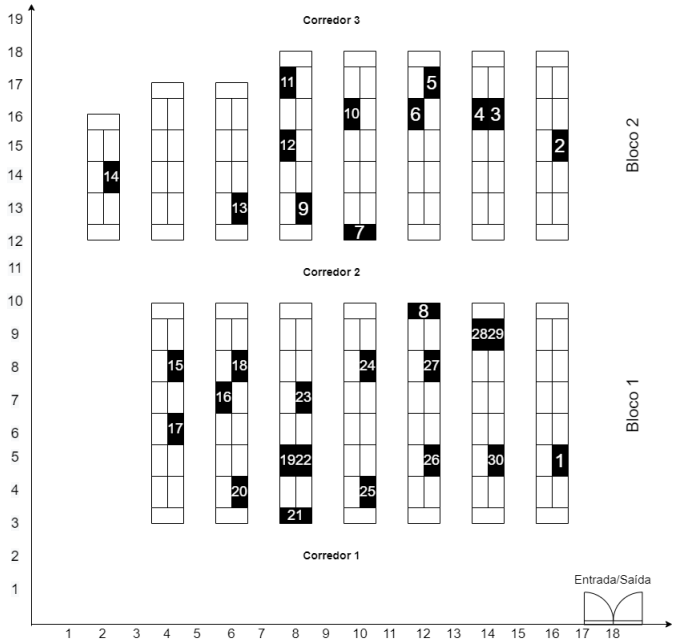
A Figura 22, a seguir, exemplifica o *layout* de um supermercado, na qual os pontos em preto representam a localização dos produtos presentes em uma lista de compras. Como é possível perceber, o cálculo das distâncias de ponto a ponto deve considerar a existência dos corredores e gôndolas. Assim, deve-se ter clareza no *layout* do supermercado a fim de escolher o melhor método para cálculo das distâncias para o percurso a ser determinado. Por essa razão, o percurso deve ser anexado no sistema.

Figura 22 – Exemplo: arranjo físico com demandas assinaladas



Fonte: Elaborada pela autora (2020).

Figura 23 – Exemplo: arranjo físico com demandas assinaladas e rota definida



Fonte: Elaborada pela autora (2020).

Por fim, a Figura 23 salienta o objetivo final do trabalho. Assim, dado um supermercado com um *layout* específico e uma determinada localização para cada item e dada uma lista de compras de um cliente, qual é a melhor ordem de coleta de itens? A Figura 23 em

questão mostra-nos um exemplo para essa ordem. No aplicativo em desenvolvimento neste estudo, a lista de produtos seria dada em forma de tabela com a ordem de coleta dos itens.

4 CONSIDERAÇÕES FINAIS

Quanto ao objetivo do presente estudo, tem-se a proposta de planejamento de um aplicativo com a função de reduzir o tempo despendido no processo de coleta de itens em um supermercado, dada uma lista de compras tal objetivo foi atingido.

Nesse contexto, a execução das atividades de desenvolvimento de aplicativo foi de extrema importância para o detalhamento das necessidades e características do projeto. Durante a execução do estudo detalhado, diversas demandas foram identificadas para que fosse entregue o Projeto de Interface e Componentes ao final do trabalho. O modelo desenvolvido foi um modelo inicial e suficiente para aplicação prática do dia a dia de uma empresa. Todavia, depois de testes e validação do modelo, *upgrades* podem ser realizados a fim de aumentar a complexidade e aplicação do aplicativo.

A utilização da solução proposta seria de extrema importância para melhoria dos serviços de entrega de compras de supermercado, uma vez que conta com uma melhoria concreta no processo em termos de tempo despendido, conta com uma interface minimalista e eficiente para o atingimento do objetivo, proporcionando assim melhoria concreta no dia a dia dos usuários e colaboradores que realizam a atividade de coleta de itens

Assim, para trabalhos futuros tem-se o desenvolvimento do aplicativo a partir da etapa de codificação para posterior aplicação no mercado. Além disso, como possibilidade de futura aplicação, tem-se a utilização dessa funcionalidade, de redução do deslocamento no processo de compras, para o cliente final, ou seja, possibilitando qualquer cliente formular sua lista de compras em um supermercado e ter em mãos a melhor rota para isso. Dessa forma, o cliente final economizaria tempo, principalmente, se desejar utilizar essa funcionalidade em um supermercado que não está acostumado a frequentar e não sabe a localização dos itens.

REFERÊNCIAS

ANTON, C. I.; EIDELWEIN, H.; DIEDRICH, H. Proposta de melhoria no *layout* da produção de uma empresa do vale do Taquari. **Revista Destaques Acadêmicos**, v. 4, n. 1, 2012.

ASKIN, Ronald G.; STANDRIDGE, Charles, R. **Modeling and analysis of manufacturing systems**. Nova Iorque: John Wiley and Sons, 1993.

CASTIGLIONI, José Antônio de Mattos. **Logística Operacional**. 2. ed. São Paulo: Érica, 2010.

CHASE, R. B.; AQUILANO, N. J.; JACOBS, F. R**. Administração da Produção para a vantagem competitiva**. 10. ed. Porto Alegre: Bookman, 2006.

CHIAVENATO, I. **Administração da produção: uma abordagem introdutória**. Rio de Janeiro: Elsevier, 2005.

CORRÊA, H. L.; CORRÊA, C. A. **Administração de Produção e Operações**. 2. ed. São Paulo: Atlas, 2009.

CUNHA, C.B. Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais. **Transportes**, v. 8, n. 2 p. 51-74, 2000.

DIAS, Marcos. **Administração de materiais**: uma abordagem logística. 4. ed. São Paulo: Atlas, 1993.

DUKIC Goran; OPETUK Tihomir. **Warehouse Layouts**. Zagreb (Croácia): Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, 2014.

FREITAS, Alan Robert Resende de. **Resolvendo o problema do caixeiro viajante via procedimento de busca adaptativa aleatória gulosa com construção baseada em redes neurais auto-organizáveis**. 2009. Monografia (Bacharelado em Ciência da Computação) – Universidade Federal de Ouro Preto, Ouro Preto, 2009.

GERHARDT, T.; SILVEIRA, D. (orgs.). **Métodos de pesquisa**. Porto Alegre: Ed. UFRGS, 2009.

GOLDBARG, Marco Cesar; LUNA, Henrique Pacca L. **Otimização combinatória e programação linea**r: modelos e algoritmos. 2. ed. Rio de Janeiro: Elsevier, 2005.

GORGULHO, J. H. C. Jr. **Arranjos Físicos**. Itajubá: Universidade Federal de Itajubá, 2010.

HILLIER, Frederick S.; LIEBERMAN, Gerald J. **Introdução à pesquisa operacional**. 8. ed. São Paulo: McGraw-Hill, 2006

LAUDON, Kenneth C.; LAUDON, Jane P. **Sistemas de informação gerenciais**. 7. ed. São Paulo: Pearson, 2007.

LAUDON, Kenneth C.; LOWE, David. **Web engineering**: a practitioner's approach. Nova Iorque: McGraw-Hill, 2009.

MOURA, Reinaldo A. **Sistema e técnicas de movimentação e armazenagem de materiais**. 4. ed. São Paulo: IMAM, 1998.

NIELSEN, Jakob; MACK, Robert L**. Usability Inspection Methods Computer**. [S.l.]: John Wiley & Sons, 1994.

PEINADO, J.; GRAEML, A.R. **Administração da Produção**: operações industriais e de serviços. Curitiba: UnicenP, 2007.

PÓVOA, C. L. R. Geoinformação e logística: caminho mais curto, roteamento de veículos e o caixeiro viajante. **MundoGEO**, abr. 2007.

PRESSMAN, Roger S. **Software Engineering:** a practitioner's approach. 8. ed. Nova Iorque: McGrawHill, 2015.

PRESSMAN, Roger S. **Engenharia de Software.** 6. ed. Rio de Janeiro: McGraw-Hill, 2006.

REEVES C. R. **Modern Heuristic Techniques for Combinatorial Problems.** Nova Iorque: John Wiley& Sons, 1993.

SABBAGH, Rafael. **Scrum**: gestão ágil para projetos de sucesso. São Paulo: Casa do Código, 2013.

SEABRA, João. UML: **Uma ferramenta para o *design* de *software***. Rio de Janeiro: Ciência Moderna, 2013.

SERNA, M., E. Analysis and Selection to Requirements Elicitation Techniques. *In*: COLOMBIAN COMPUTING CONGRESS (CCC), 7., 2012, Medellín, Colômbia. **Anais...** Medellín: IEEE, 2012. p. 1-7.

SILVA. G. L. da. **Uma nova abordagem para o problema de roteirização de veículos com restrições operacionais**. 2010. Tese (Doutorado em Transportes) – Universidade de Brasília, Brasília, 2010.

SLACK, N.; CHAMBERS, S.; JOHNSTON, R**. Administração da Produção**. 3. ed. São Paulo: Atlas, 2009.

SLACK, N.; CHAMBERS, S.; JOHNSTON, R. **Administração da Produção**. Tradução A. B. Brandão. 5. ed. São Paulo: Atlas, 2013.

SOARES, Michel dos S. Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de *Software*. **Infocomp – Revista de Ciência da Computação**, Lavras, v. 2, p. 8-13, 2004.

SOCIAL MINER. **Pesquisa – O futuro do consumo pós-Covid-19**. São Paulo: Social Miner; Opinion Box, 2020.

SOMMERVILLE, I. **Engenharia de *software***. Tradução Ivan Bosnic e Kalinka G. O. Gonçalves. Revisão técnica Kechi Hirama. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

SOMMERVILLE, Ian. **Software Engineering**. 8. ed. Harlow: Pearson Education, 2008.

SOUZA NETO, Oscar Nogueira de. **Análise Comparativa das Metodologias de Desenvolvimento de Softwares Tradicionais e Ágeis**. 2004. Monografia (Bacharelado em Ciência da Computação) – Universidade da Amazônia, Belém, 2004.

SUTCLIFFE, A.; SAWYER, P. **Requirements Elicitation**: Towards the Unknown Unknowns. Requirements Engineering Conference. Rio de Janeiro: [s.n.], 2013.

TAUFER, Fernando Soares Gomes; PEREIRA, Elaine Correa. Aplicação do problema do caixeiro viajante na otimização de roteiros. *In*: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 21., 2011, Belo Horizonte. **Anais...** Belo Horizonte: UFMG, 2011.

TOMPKINS, J. A.; WHITE, J. A.; BOZER, Y. A.; FRAZELLE, E. H.; TANCHOCO, J. M. A.; TREVINO, J. **Facilities Planning**. 2. ed. Nova Iorque: John Wiley & Sons, 1996.

TOOGE, Rikardy. Vendas online de supermercados quase dobram após o coronavírus, entregas atrasam e exigem investimentos das empresas. **Portal G1,** 07 de abr. de 2020. Disponível em: https://g1.globo.com/economia/noticia/2020/04/07/vendas-online-de-supermercados-quase-dobram-apos-o-coronavirus-entregas-atrasam-e-exigem-investimentos-das-empresas.ghtml. Acesso em: 02 nov. 2020.

TOTH, P.; VIGO, D. The Vehicle Routing Problem. SIAM. **Monographs on Discrete Mathematics and Applications**, Filadélfia, 2002.

VASCONCELOS, Alexandre Marcos Lins de; ROUILLER, Ana Cristina; MACHADO, Cristina Ângela Filipak; MEDEIROS, Teresa Maria Maciel de. **Introdução à Engenharia de Software e à Qualidade de Software**. Lavras: UFLA/FAEPE, 2006.

YOUNG, R. R. **Requirements Engineering Handbook**. Norwood: Artech House, 2003

WANKE, Peter; MAGALHÃES, Andréia. **Logística para Micro e Pequenas Empresas**. São Paulo: Atlas, 2012.

ANEXO A

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
FACULDADE DE ENGENHARIA

Termo de Declaração de Autenticidade de Autoria

Declaro, sob as penas da lei e para os devidos fins, junto à Universidade Federal de Juiz de Fora, que meu Trabalho de Conclusão de Curso do Curso de Graduação em Engenharia de Produção é original, de minha única e exclusiva autoria. E não se trata de cópia integral ou parcial de textos e trabalhos de autoria de outrem, seja em formato de papel, eletrônico, digital, áudio-visual ou qualquer outro meio.
Declaro ainda ter total conhecimento e compreensão do que é considerado plágio, não apenas a cópia integral do trabalho, mas também de parte dele, inclusive de artigos e/ou parágrafos, sem citação do autor ou de sua fonte.
Declaro, por fim, ter total conhecimento e compreensão das punições decorrentes da prática de plágio, através das sanções civis previstas na lei do direito autoral[1] e criminais previstas no Código Penal[2], além das cominações administrativas e acadêmicas que poderão resultar em reprovação no Trabalho de Conclusão de Curso.

Juiz de Fora, 18 de dezembro de 2020.

Gabriela da Silva Lima	201549003
NOME LEGÍVEL DO ALUNO (A)	Matrícula
Gabriela Lima	106.266.536-81
ASSINATURA	CPF

[1] LEI N° 9.610, DE 19 DE FEVEREIRO DE 1998. Altera, atualiza e consolida a legislação sobre direitos autorais e dá outras providências.
[2] Art. 184. Violar direitos de autor e os que lhe são conexos: Pena - detenção, de 3 (três) meses a 1 (um) ano, ou multa.