

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

BRUNNO MARCEL FERREIRA MARINS
LUCAS HIRO ONO DA FONSECA

VENDAS MOBILE: UMA APLICAÇÃO MÓVEL PARA LANÇAMENTOS DE
VENDAS DE FORMA OTIMIZADA

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA

2022

BRUNNO MARCEL FERREIRA MARINS

LUCAS HIRO ONO DA FONSECA

**VENDAS MOBILE: UMA APLICAÇÃO MÓVEL PARA LANÇAMENTOS DE
VENDAS DE FORMA OTIMIZADA**

MOBILE SALES: A MOBILE APP FOR OPTIMIZED SALES LAUNCHES

Trabalho de Conclusão de Curso
apresentado como requisito parcial à
obtenção do título de Tecnólogo, do
Departamento de Informática, da
Universidade Tecnológica Federal do
Paraná.

Orientadora: Profa. Dra. Eliana Cláudia
Mayumi Ishikawa

PONTA GROSSA

2022



[4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/)

Esta licença permite remixe, adaptação e criação a partir do trabalho, para fins não comerciais, desde que sejam atribuídos créditos ao(s) autor(es) e que licenciem as novas criações sob termos idênticos. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



TERMO DE APROVAÇÃO

VENDAS MOBILE: UMA APLICAÇÃO MÓVEL PARA LANÇAMENTOS DE VENDAS DE FORMA OTIMIZADA

por

BRUNNO MARCEL FERREIRA MARINS
LUCAS HIRO ONO DA FONSECA

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 31 de Outubro de 2022 como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas. Os candidatos foram arguidos pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof^a. Dra. Eliana Cláudia Mayumi Ishikawa
Orientadora

Prof. Dr. Lourival A. de Góis
Membro titular

Prof. MSc. Geraldo Ranthum
Membro titular

Prof. MSc. Geraldo Ranthum
Responsável pelo Trabalho de
Conclusão de Curso

Prof^a. Dra. Mônica H. Pietruchinski
Coordenadora do curso

AGRADECIMENTOS

Agradecemos aos nossos amigos e familiares por sempre estarem ao nosso lado, nos apoiando e incentivando com sabedoria.

Agradecemos a nossa orientadora Prof^a. Dra. Eliana Cláudia Mayumi Ishikawa pela motivação, orientação e por não medir esforços para esclarecer nossas dúvidas no decorrer desse trabalho.

Aos professores, pelas correções e ensinamentos que nos permitiram apresentar um melhor desempenho no nosso processo de formação profissional ao longo do curso.

RESUMO

MARINS, Brunno Marcel Ferreira; FONSECA, Lucas Hiro Ono. **VENDAS MOBILE: UMA APLICAÇÃO MÓVEL PARA LANÇAMENTOS DE VENDAS DE FORMA OTIMIZADA.** 2022. 107 f. Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistemas) - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2022.

A utilização dos *smartphones* no nosso cotidiano tornou-se uma constante, diariamente utilizamos para nos auxiliar em algum processo, para obter informações ou até mesmo para o lazer. Com essa percepção, várias empresas estão apostando em recursos de mobilidade para o aumento do desempenho em seus processos. Na empresa Insumos LTDA, os consultores ainda precisam anotar em blocos de pedidos os pedidos solicitados pelos clientes. Após a solicitação do cliente, o consultor precisa encaminhar as anotações para o faturista de sua filial para ele poder cadastrar os pedidos no sistema da empresa. Esse processo não é eficaz, pois os pedidos são encaminhados no final do expediente e apenas no dia seguinte é cadastrado no sistema. Buscando a solução para esse problema, este trabalho propôs o desenvolvimento de um aplicativo móvel capaz de cadastrar os pedidos através de um *smartphone* e sincronizar os dados com o sistema da empresa através da internet. Para tal feito, a sincronização dos dados do aplicativo com o sistema da empresa ocorre através de *web service* que foi desenvolvido utilizando a arquitetura REST. O aplicativo foi desenvolvido utilizando o *framework* Flutter que disponibiliza a aplicação para sistema Android e IOS através de um único código fonte.

Palavras-chave: Criação de pedidos. Smartphone. Aplicativo móvel. Integração de Sistemas. Web Service. REST. Flutter.

ABSTRACT

MARINS, Brunno Marcel Ferreira; FONSECA, Lucas Hiro Ono. **MOBILE SALES: A MOBILE APP FOR OPTIMIZED SALES LAUNCHES**. 2022. 107 p. Work of Conclusion Course (Graduation in Development and Analysis Technology of the Systems) - Federal Technology University - Parana. Ponta Grossa, 2022.

The use of smartphones in our daily lives has become a constant, daily used to assist us in some process, to obtain or even for leisure. With this perception, several companies are betting on mobility resources to increase performance in their processes. In the Insumos LTDA company, the consultants still need to write down the orders requested by the clients in order blocks. After the customer's request, the consultant needs his orders for the invoice to elect the company. This process is efficient, as the orders sent are not sent at the end of the day and only on the day they are registered in the next system. Seeking the solution to the problem, this work for the development of a mobile application capable of registering orders through a smartphone and synchronizing the data with the internet system. For this, a system of synchronization of the company's data takes place from the application to the web service that was developed using a REST architecture. The application was developed using the Flutter framework that provides an application for the Android and IOS system through a single source code.

Keywords: Order creation. Smartphones. Mobile app. Systems integration. Web Service. REST Flutter.

LISTA DE ILUSTRAÇÕES

Figura 1 - Visão geral da adoção e uso de dispositivos e serviços conectados	17
Figura 2 - Representação visual da equipe Scrum	23
Figura 3 - Representação visual do Product Backlog	24
Figura 4 - Representação visual dos principais artefatos do Framework Scrum	25
Figura 5 - Popularização dos frameworks híbridos 2019 - 2021	27
Figura 6 - Representação gráfica do fluxo de comunicação de um Web Service	30
Figura 7 – Etapas do desenvolvimento do projeto	38
Figura 8 - Representação visual dos principais artefatos do SCRUM Framework ...	43
Figura 9 – Estrutura do Sistema Web	48
Figura 10 – Estrutura do Sistema Móvel	50
Figura 11 - Verificação de conexão com a web service e versão (Parte 1)	51
Figura 12 - Verificação de conexão com a web service e versão (Parte 2)	52
Figura 13 - Tela de Login do Aplicativo	53
Figura 14 - Menu Principal do Vendas Mobile	54
Figura 15 - Requisição para atualização de dados do estoque, lado cliente	55
Figura 16 - Estrutura interna para armazenamento de dados do estoque	56
Figura 17 - Exemplo de requisição para atualização de dados, lado servidor	57
Figura 18 - Tela de Lista de Pedidos do Vendas Mobile	58
Figura 19 - Tela de Lançamento de Pedido do Vendas Mobile	59
Figura 20 - Validação de dados do Lançamento de Pedido do Aplicativo	60
Figura 21 - Armazenamento do pedido internamente e mensagens de erro	60
Figura 22 - Tela de Inclusão de Produto no Pedido	61
Figura 23 - Parte do código de busca e cálculo do valor do produto	62
Figura 24 - Tela de Lista de Pedidos do Vendas Mobile	63
Figura 25 - Parte do processo de envio de pedidos pendentes	64
Figura 26 - Tela de Consulta de Estoque no Aplicativo	65
Figura 27 - Tela de Liberações do Vendas Mobile	66
Figura 28 - Lista de Liberação de Pedidos de Venda	67
Figura 29 - Visualização de Pedido para Liberação	68
Figura 30 - Processo de liberação de pedidos	68
Figura 31 - Gráfico referente ao Cargo e Função	72

LISTA DE QUADROS

Quadro 1 - Características de avaliação de qualidade ISO/IEC 9126	34
Quadro 2 - Subcaracterísticas de avaliação de qualidade ISO/IEC 9126	35
Quadro 3 - Lista de requisitos do Product Backlog	45
Quadro 4 - Lista de Categorias de <i>Sprints</i> e Tempos de Execução	46
Quadro 5 - Característica, quantidade de perguntas e número da questão	71
Quadro 6 - Perguntas referentes a característica de funcionalidade. Parte 1	73
Quadro 7 - Perguntas referentes a característica de funcionalidade. Parte 2	75
Quadro 8 - Perguntas referentes a característica de usabilidade	78

LISTA DE SIGLAS

ERP	Enterprise Resource Planning
FTP	File Transfer Protocol
HTTP	HyperText Transfer Protocol
IOS	Iphone Operating System
LTDA	Limitada
PHP	Personal Home Page
SGBD	Sistema Gerenciador de Banco de Dados
SMTP	Simple Mail Transfer Protocol
SQL	Structured Query Language
WSDL	Web Services Description Language
XML	Extensible Markup Language

LISTA DE ACRÔNIMOS

ISO	International Organization for Standardization
JSON	JavaScript Object Notation
SOAP	Simple Object Access Protocol
REST	Representational State Transfer
UDDI	Universal Description, Discovery and Integration
URI	Uniform Resource Identifier

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 JUSTIFICATIVA.....	13
1.2 OBJETIVO GERAL.....	14
1.3 OBJETIVO ESPECÍFICOS.....	14
1.4 ORGANIZAÇÃO DO TRABALHO.....	14
2 REFERENCIAL TEÓRICO.....	16
2.1 APLICAÇÃO MOBILE.....	16
2.2 METODOLOGIA ÁGEIS DE DESENVOLVIMENTO DE SOFTWARE.....	19
2.2.1 Modelo Scrum.....	21
2.3 FLUTTER.....	25
2.4 INTEGRAÇÃO ENTRE SISTEMAS.....	28
2.4.1 Web Services.....	29
2.4.2 Arquitetura REST.....	31
2.5 BANCO DE DADOS.....	32
2.5.1 PostgreSQL.....	32
2.6 AVALIAÇÃO DA QUALIDADE DE PRODUTO DE SOFTWARE.....	33
2.6.1 Norma ISO/IEC 9126.....	34
3 METODOLOGIA.....	37
3.1 ORGANIZAÇÃO E DESENVOLVIMENTO DO TRABALHO.....	38
3.1.1 Desenvolvimento da aplicação.....	39
3.1.2 Avaliação do Aplicativo Vendas Mobile.....	39
3.1.2.1 Localização e participantes da pesquisa.....	40
3.1.2.2 Coleta de dados.....	41
4 ANÁLISE E RESULTADOS DA PESQUISA.....	42
4.1 DESENVOLVIMENTO DA APLICAÇÃO.....	42
4.1.1 Equipe.....	44
4.1.2 Product Backlog.....	44
4.1.3 Sprint Backlog.....	46
4.1.4 Daily Scrum.....	69
4.2 AVALIAÇÃO DO APLICATIVO VENDAS MOBILE.....	70
4.2.1 Análise dos Resultados.....	72
4.2.1.1 Avaliação da característica funcionalidade.....	73
4.2.1.2 Avaliação da característica confiabilidade.....	78
4.2.1.3 Avaliação da característica usabilidade.....	78
4.2.1.4 Avaliação da característica eficiência.....	80
5 CONCLUSÃO.....	81
5.1 TRABALHOS FUTUROS.....	83
REFERÊNCIAS.....	85

APÊNDICE A - Pesquisa de Satisfação do Aplicativo Vendas Mobile	90
APÊNDICE B - Gráficos da Respostas da Pesquisa de Satisfação.....	99

1 INTRODUÇÃO

Dentre vários aspectos do desenvolvimento da tecnologia, é perceptível o aumento exponencial dos dispositivos móveis no mercado e isso se deve ao fato da aceitação dessa tecnologia pela sociedade como um todo.

Nota-se que esse instrumento tornou-se uma ferramenta importante no contexto de negócios, pois permite a otimização da gestão empresarial em todos os setores, seja por meio do controle automático de processos, indicadores de metas, tomada de decisão ou comunicação organizacional interna ou externa. Utilizada de maneira estratégica, as tecnologias de informação facilitam o acesso às informações e permitem uma maior mobilidade, possibilitando aos usuários enviar e receber dados a qualquer momento e em qualquer lugar.

Tendo em vista a sintetização de processos rotineiros, principalmente de trabalho, tanto da parte do dono de uma empresa, seus funcionários e clientes, há cada vez mais a necessidade de uma alternativa móvel para a execução dessas tarefas presentes no ambiente de trabalho.

Com a evolução dos dispositivos móveis, surgiu a necessidade de criar aplicações móveis, ou seja, *software* para *smartphones* que permitem a integração entre *software* e mobilidade. Assim como existem sites para vários setores, também existem aplicações móveis com diversas finalidades, como por exemplo: aplicativos de comércio, de prestação de serviços ou até mesmo aplicativos de uso exclusivo de colaboradores que agilizam processos internos de uma empresa (COSTA, 2018).

Bisol (2019) define que as empresas que utilizam aplicativos móveis, se beneficiam de aumento dos lucros, da otimização do tempo e conseguem um melhor gerenciamento dos processos interno pelo fato de que as ações podem ser realizadas em qualquer lugar e a qualquer momento, pois os aplicativos móveis tornam-se uma extensão dos sistemas da empresa.

Pode-se verificar que essa necessidade tem o potencial de melhorar o cotidiano dos usuários e da empresa. Tanto é, em um estudo feito para verificar a correlação do desenvolvimento econômico de países africanos com o aumento do uso de dispositivos móveis mostrou que, a facilidade de acesso à informação e a

funcionalidades providas pelo uso desses dispositivos, realmente teve impacto no crescimento econômico desses países em desenvolvimento (AKER; MBITI, 2010).

Com isso em mente, notou-se a necessidade em dar mobilidade nos processos do módulo de vendas da empresa Insumos LTDA, principalmente nos processos de lançamento de pedidos e de liberação de pedidos. O primeiro trata-se da inclusão do pedido feito pelo cliente através de um vendedor e a segunda trata-se da liberação dos pedidos criados com um valor abaixo do indicado, onde eles devem ser encaminhados para análise e liberação.

Esse trabalho de conclusão de curso tem como objetivo desenvolver um aplicativo móvel que flexibilize e dê maior eficiência para os processos do módulo de vendas da Insumos LTDA permitindo aumentar a precisão nos dados adquiridos e facilitar o acesso de informações.

1.1 JUSTIFICATIVA

O desenvolvimento da aplicação Vendas Mobile se deu diante da necessidade de facilitar o acesso ao sistema em situações diversas onde o consultor de vendas não possui seu computador disponível para o lançamento e consulta de seus pedidos.

A utilização dos blocos de pedidos de papel resultam em alguns problemas como perdas ou realocação sem a ciência de todos os funcionários, gerando conflitos durante a busca do histórico. Com o desenvolvimento do aplicativo esses problemas também serão solucionados, pois todos os pedidos gerados serão centralizados no sistema *web*, possibilitando que os funcionários tenham o acesso a eles a qualquer momento.

Com o uso do Vendas Mobile controle dos pedidos será de responsabilidade apenas dos vendedores e não mais dos faturistas de cada filial resultando no aumento da produtividade do mesmo.

Tornando móvel a liberação de pedidos do sistema, os gestores poderão tomar decisão em qualquer lugar em que estiverem, sem precisar se preocupar em acessar o sistema *web*. De forma discreta podem estar tomando decisões sobre os

negócios da empresa durante reuniões, viagens, almoço ou em qualquer situação em que o acesso a um desktop ou notebook não é viável.

As empresas que utilizam de aplicativos móveis demonstram que estão atualizadas com a modernidade e as tecnologias existentes tornando a marca cada vez mais visível aos seus clientes (COSTA, 2018).

Integrar um sistema corporativo com um aplicativo móvel faz com que o aplicativo torne-se uma extensão do sistema que está presente na empresa. A integração desses elementos cria mobilidade gerando mais liberdade aos colaboradores da organização, pelo fato de que as ações podem ser realizadas de qualquer lugar e a qualquer momento (BISOL, 2019).

1.2 OBJETIVO GERAL

O objetivo geral deste trabalho é desenvolver um aplicativo mobile que permita dar maior mobilidade e flexibilidade aos processos de consulta de estoque, lançamento e liberação de pedidos da empresa Insumos LTDA¹.

1.3 OBJETIVO ESPECÍFICOS

- Utilizar a ferramenta de metodologia ágil de desenvolvimento *Scrum* no processo de desenvolvimento do aplicativo Vendas Mobile para a empresa Insumos LTDA;
- Avaliar a aplicação na perspectiva do usuário por meio das características da norma ISO/IEC 9126.

1.4 ORGANIZAÇÃO DO TRABALHO

O trabalho está organizado em quatro capítulos. O Capítulo 1 (um) apresenta a motivação e problema em questão que se pretende resolver. O Capítulo

¹ Nome fictício da empresa

2 (dois) aborda sobre as metodologias ágeis, principalmente sobre a utilizada no trabalho, o *Scrum*, também as ferramentas utilizadas para o desenvolvimento da aplicação e sobre a avaliação de qualidade de produto de *software*.

O Capítulo 3 (três) trata da metodologia a ser aplicada no desenvolvimento, comentado dos papéis de trabalho trazidos pelo metodologia *Scrum*, como funciona o gerenciamento de atividades na metodologia e as etapas em que o projeto foi dividido.

O Capítulo 4 (quatro) mostra detalhadamente a aplicação da metodologia nas etapas citadas anteriormente, como o desenvolvimento do aplicativo, criação e aplicação do formulário que avalia a qualidade de produto de *software* e a análise dos resultados das respostas do formulário.

2 REFERENCIAL TEÓRICO

Neste capítulo serão apresentadas abordagens e características sobre o desenvolvimento de aplicações móveis e como elas podem impactar na gestão de processos internos entre empresas.

2.1 APLICAÇÃO MOBILE

Segundo Vilcek e Jakopiec (2017), Que, Guo e Zhu (2016), o crescimento da aceitação de *smartphones* é devido a grande quantidade de aplicativos que oferecem uma variedade de recursos trazendo comodidade e mobilidade para as pessoas. Devido a aceitação do aparelho, no mercado há diversos modelos de dispositivos que executam diferentes sistemas operacionais (QUE; GUO; ZHU, 2016). O autor Brito (2018) reforça as afirmações anteriores e explica a importância dos dispositivos móveis na seguinte citação:

“Os dispositivos móveis estão cada vez mais avançados e eficientes, estimando-se que até 2020 existam cerca de 2,87 mil milhões de utilizadores de *smartphones*. O número de utilizadores de Internet a nível global previsto para 2020 é de 52,4%, julgando-se que cerca de 34% da população mundial terá pelo menos um *smartphone*. Isto vem reforçar a importância dos dispositivos móveis no dia a dia de toda a sociedade, pois aliam o poder de computação à portabilidade”. (BRITO, 2018, p. 1).

Kemp (2022) relata que entre o início de 2021 até o início de 2022, houve um aumento de 95 (noventa e cinco) milhões de novos usuários de *smartphones*, atingindo um total de 5,31 (cinco vírgula trinta e um) bilhões, o equivalente a 67,1% (sessenta e sete vírgula um por cento) da população mundial.

Também houve aumento na quantidade de usuários da internet, no mesmo período houve um acréscimo de 4% (quatro por cento) o equivalente a 192 (cento e noventa e dois) milhões de pessoas conforme ilustra a Figura 1.

Figura 1 - Visão geral da adoção e uso de dispositivos e serviços conectados



Fonte: Adaptado de Kemp (2022)

Os *smartphones* tornam-se cada vez mais populares devido às diversas funcionalidades que neles existem e também pela quantidade de aplicações desenvolvidas para a comodidade dos usuários em suas atividades cotidianas, lazer e trabalho (QUE; GUO; ZHU, 2016).

Com a evolução desses aparelhos, surgiu a necessidade de criar aplicativos *mobile*, ou seja, *softwares* para *smartphones* que permitem a integração entre sistemas e mobilidade. Assim como existem sites para vários setores, também existem aplicações móveis com diversas finalidades, como por exemplo, aplicativos de comércio, de prestação de serviços ou até mesmo aplicativos de uso exclusivo de colaboradores que agilizam processos internos de uma empresa (COSTA, 2018).

Para Costa (2018), a portabilidade é uma das principais vantagens de se utilizar aplicativos móveis, uma vez que os serviços disponíveis na internet que são acessados por meio de um computador, podem ser utilizados no *smartphone* em qualquer lugar.

Além da portabilidade, é possível citar algumas outras vantagens, conforme descritas por Porto (2012), tais como:

- Facilidade de uso: Como trata-se de um dispositivo compacto, a interface é adaptada fazendo com que os recursos sejam otimizados para agilidade na navegação.
- Custo de acesso: Por se tratar de uma interface minimalista e adaptada para o dispositivo, a quantidade de dados trafegados é menor em relação a navegadores dos computadores.
- Recursos disponíveis: Como o aparelho possui diversos recursos, os aplicativos podem possibilitar uma melhor experiência de uso.
- Acesso sem rede de internet: Caso não tenha acesso a internet, o aplicativo pode armazenar as informações e sincronizar quando o acesso for disponível.

Em contrapartida, o autor também descreve o versionamento e os diversos sistemas operacionais existentes como sendo algumas das principais desvantagens às aplicações *mobile*.

- Versionamento: Como existe a atualização constante dos modelos dos *smartphones*, os sistemas operacionais também são atualizados possibilitando novos recursos e caso a aplicação utilize algum desses recursos, os usuários com os aparelhos antigos não poderão utilizar a aplicação.
- Diferentes sistemas operacionais: Alguns fabricantes de *smartphones* possuem seu próprio sistema operacional, tornando necessário o desenvolvimento de aplicações para cada tipo de sistema. Neste caso o desenvolvedor terá de escolher se irá desenvolver para um sistema operacional específico ou híbrido.

Bisol (2019) define que as empresas que utilizam aplicativos móveis, se beneficiam de aumento dos lucros, da otimização do tempo e conseguem um melhor gerenciamento dos processos interno pelo fato de que as ações podem ser realizadas em qualquer lugar e a qualquer momento, pois os aplicativos móveis tornam-se uma extensão dos sistemas da empresa.

Tendo em vista a importância competitiva que os aplicativos trazem através da otimização do tempo no processo, na flexibilidade da utilização dos recursos *offline*, na mobilidade, na comodidade entre outros já citados, torna-se necessário que as empresas invistam no desenvolvimento de tecnologia móvel customizada que atendam suas necessidades específicas.

2.2 METODOLOGIA ÁGEIS DE DESENVOLVIMENTO DE SOFTWARE

O desenvolvimento de *software* é uma tarefa complexa que exige um processo metódico para que seja gerado um produto de qualidade dentro do prazo e do orçamento previsto, atendendo as necessidades e interesses do cliente.

Utilizar de metodologias é uma maneira de evitar situações caóticas em atividades complexas no desenvolvimento de *software*. Nas metodologias são definidas as atividades, tarefas, ações entre outros elementos necessários para o desenvolvimento com alta qualidade (LEITÃO, 2010).

As diversas metodologias de desenvolvimento de *software* podem, ocasionalmente, adotar as mesmas atividades padrões, mas dando ênfase em outras atividades, além de possuírem ciclos de vidas diferentes (PRESSMAN, 2006).

No período de 1980 e 1990, acreditava-se que a melhor maneira de desenvolver um *software* era através de um planejamento com um processo de desenvolvimento minucioso e controlado. Esse tipo de abordagem se justifica para sistemas críticos ou projetos de grande proporção com várias equipes distintas que precisam ser gerenciadas (SOMMERVILLE, 2011).

Para Sommerville (2011), esse tipo de abordagem raramente trazia resultados satisfatórios quando aplicada para sistemas de pequeno e médio porte. O tempo gasto com a elaboração do projeto e com a documentação era muito superior ao tempo gasto com o desenvolvimento.

Com a insatisfação dos resultados que as abordagens tradicionais traziam, em 2001 um grupo de profissionais de desenvolvimento de *software* se reuniram para discutir ideias e conceitos em que tornassem o desenvolvimento leve e assim

criaram o Manifesto Ágil. O Manifesto descreve quatro valores centrais do Agile (SAVOINE et al., 2009):

- Interações individuais são mais importantes do que processos e ferramentas.
- Foco no *software* funcionando em vez de documentação completa.
- Colaboração em vez de negociações contratuais.
- Respostas rápidas a mudanças ao invés de seguir planos.

O Manifesto Ágil também apresenta 12 (doze) princípios básicos e fundamentais para o processo de desenvolvimento:

1. Satisfazer os clientes através da entrega antecipada e contínua de *software* com valor agregado.
2. Dividir tarefas extensas em tarefas menores que podem ser concluídas rapidamente.
3. Reconhecer que o melhor trabalho emerge de equipes auto-organizadas.
4. Fornecer o ambiente e o apoio que a equipe precisa para mantê-los motivados e confiar na capacidade deles para realizar o trabalho.
5. Criar processos que promovam esforços sustentáveis. Todos os envolvidos no projeto devem ser capazes de manter uma paz constante indefinidamente.
6. Entregar o *software* funcionando frequentemente no menor tempo possível.
7. Mudança de requisitos sempre é bem vindo, mesmo no final do projeto. A mudança faz parte da vantagem competitiva de seus clientes.
8. Equipe de desenvolvimento e empresários devem trabalhar juntos durante todo o projeto.
9. Fazer com que a equipe reflita em intervalos regulares sobre como se tornar mais eficaz e, em seguida, ajuste o comportamento de acordo.
10. Meça o progresso pela quantidade de trabalho concluído.
11. Buscar continuamente a excelência.
12. Aproveite a mudança para obter uma vantagem competitiva.

Sommerville (2011) reforça que ao contrário das metodologias tradicionais, as metodologias ágeis tem como objetivo diminuir a burocracia dos processos a fim de evitar o tempo gasto desnecessariamente com documentação que possivelmente não serão utilizadas pelo cliente, dessa maneira leva menos tempo para a entrega do *software*.

Esse tipo de abordagem é mais flexível a mudanças durante o desenvolvimento do projeto, fazendo com que os investimentos sejam melhores aproveitados em comparação com as metodologias tradicionais que utilizam muito tempo em análise e planejamento da extensa regulamentação, regimentação e micro gerenciamento (SAVOINE et al., 2009).

Projetos podem sofrer mudanças e os requisitos estão em constante evolução, por isso surgiu a necessidade de metodologias que se adequem a esse paradigma. Diferente das abordagens tradicionais que são pouco flexíveis à mudança, a metodologia ágil recebe e se adapta facilmente na ocorrência de novas alterações no projeto, pois todos os participantes estão bem informados e prontos para lidar com mudanças (LEITÃO, 2010).

No capítulo seguinte será apresentado a metodologia de desenvolvimento ágil chamada *Scrum* que foi escolhida para o desenvolvimento do aplicativo proposto neste trabalho. O *Scrum* é a metodologia utilizada para gerenciar o desenvolvimento de um produto, onde é possível utilizar diferentes técnicas e processos (SCHWABER; SUTHERLAND, 2020).

2.2.1 Modelo Scrum

O *Scrum* é uma ferramenta de metodologia ágil para desenvolvimento de produtos, que permite tratar e resolver problemas com alto grau de complexidade entregando produtos de alto valor. Esta ferramenta não é um processo padronizado com uma série de passos onde deve ser seguido metodicamente, mas sim um conjunto de valores, princípios e práticas que podem ser usados de acordo com a necessidade e relevância para cada organização, resultando em sua própria versão do *Scrum* (RUBIN, 2018).

O objetivo do *Scrum* é fazer com que um projeto seja concluído com sucesso através de gerenciamento e criação de regras. Dessa forma os envolvidos no desenvolvimento do projeto sabem em qual estágio o projeto se encontra (ANDRADE et al., 2012).

Os processos dessa metodologia ágil são divididos em ciclos e cada divisão representa um conjunto de atividades que devem ser executadas. Nessa metodologia são definidos os padrões de gestão e planejamento do desenvolvimento (SCHWABER; SUTHERLAND, 2020).

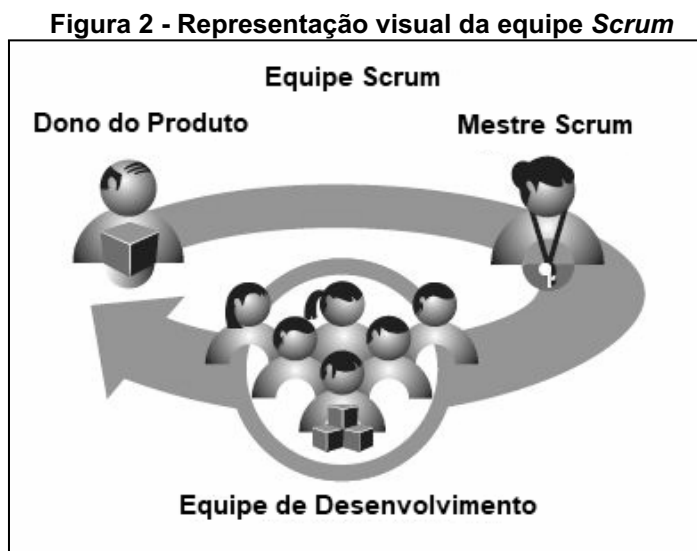
Os autores descrevem a metodologia como sendo uma ferramenta baseada em times associados a papéis, eventos, artefatos e regras com um propósito específico e são essenciais para uso e sucesso do *Scrum*.

Esta ferramenta é focada no aperfeiçoamento de previsibilidade e controle de riscos, para isso, os processos são iterativos e incrementais. A metodologia utiliza de uma lista de requisitos organizada, categorizada e que pode ser alterada para garantir uma entrega rápida e de acordo com o que foi previsto (SCHWABER; SUTHERLAND, 2020).

Segundo Rubin (2018), uma das práticas do *Scrum* é a definição da Equipe *Scrum* que é dividida em três grupos, sendo eles:

- **Dono do Produto:** Responsável pela gestão do projeto, decidindo o que deve ser priorizado dentro da lista de pendências, atribuindo tarefas de acordo com o desenvolvimento do trabalho de forma a otimizar as etapas.
- **Mestre *Scrum*:** Líder de equipe responsável em certificar-se de que todo o processo é eficaz, facilitando as reuniões, avaliando a transparência, esclarecendo o que for necessário para a equipe atingir os objetivos guiando em direção ao aprimoramento contínuo.
- **Time de Desenvolvimento:** São responsáveis por desenvolver o produto cumprindo as funções que foram delegadas pelo Mestre *Scrum* e Dono do Produto, de forma incremental e ágil.

A Figura 2 representa visualmente a equipe *Scrum* e o ciclo de interação que ocorre dentro da metodologia.



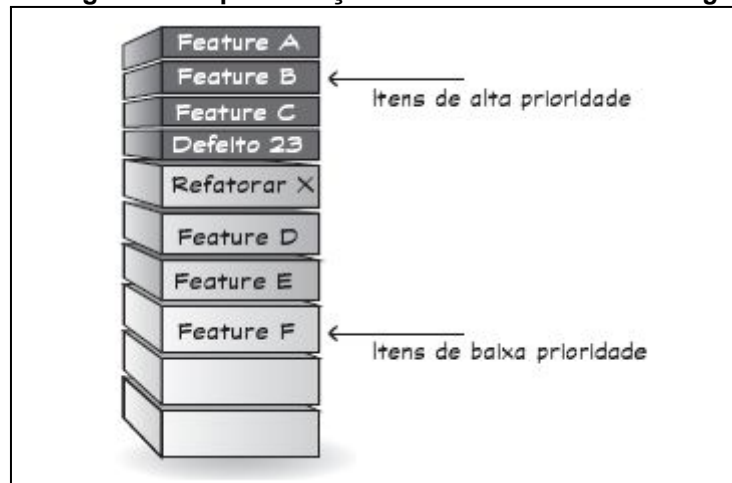
Fonte: Adaptado de Rubin (2018).

Segundo Albino, Souza e Prado (2013), quando já existe o conhecimento do que deve ser feito para atingir o resultado final do produto que o cliente espera, então significa que o projeto *Scrum* foi iniciado. Esse conhecimento deve ser documentado com todas as informações dos requisitos levantados, esse documento é denominado como *Product Backlog*.

O *Product Backlog* é uma lista que contém todas as etapas, elementos e requisitos necessários para o desenvolvimento do produto, organizada de forma que prioriza os itens mais importantes e detalhados até itens que não estão bem definidos. Essa lista pode sofrer alterações a medida que cada tópico seja concluído conforme novas necessidades sejam levantadas. As alterações podem ser de novos requisitos, mudanças em requisitos já existentes, defeitos necessitando de reparos, melhorias técnicas e assim por diante (RUBIN, 2018).

A Figura 3 apresenta uma ilustração da ordem de prioridade dos itens em um *Product Backlog*.

Figura 3 - Representação visual do *Product Backlog*



Fonte: Rubin (2018).

Após a criação do *Product Backlog*, uma reunião de planejamento conhecida como *Sprint Planning*, deve ser iniciada onde o Dono do Produto juntamente com o Mestre *Scrum* e o Time de Desenvolvimento definem o que deverá ser desenvolvido nas *sprints*. Essa categorização de cada tópico da lista definida pelo time de desenvolvimento recebe o nome de *Sprint Backlog* (SCHWABER; SUTHERLAND, 2020).

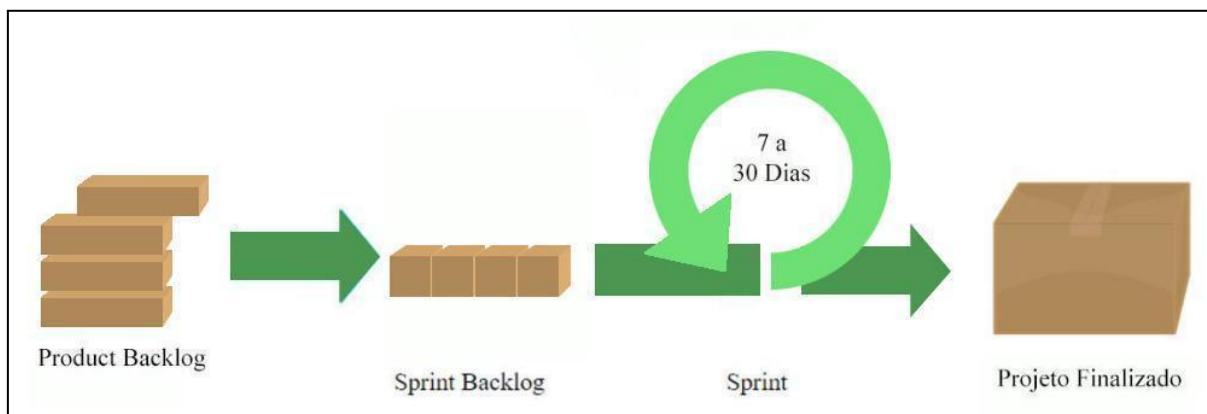
Rubin (2018) explica que as *Sprints* são períodos utilizados para execução e conclusão de uma parte do projeto e nelas deve haver diariamente uma reunião de acompanhamento do progresso do trabalho da equipe e essa reunião recebe o nome de *Daily Scrum*. O tempo da reunião deve ser de no máximo 15 (quinze) minutos, e nela cada membro da equipe deve responder três perguntas, que são:

- O que eu realizei desde a última *Daily Scrum*?
- O que eu planejo trabalhar para o próximo *Daily Scrum*?
- Quais são os obstáculos ou impedimentos que estão evitando que eu progrida?

O objetivo desta reunião é para que todos entendam o quadro geral do que está acontecendo, como estão progredindo na *Sprint*. Nela não devem ser discutidas soluções de problemas, mas após a sua conclusão, os interessados podem ajudar uns aos outros (RUBIN, 2018).

Uma *Sprint Review* ou reunião de revisão, deve ocorrer ao final de cada *Sprint* como o objetivo de validar as funcionalidades, adaptações podem ser realizadas se forem necessárias. Todo o processo se repete até a conclusão do *Product Backlog* e a entrega do produto ao cliente (SCHWABER; SUTHERLAND, 2020). De forma simplificada, a Figura 4 representa as etapas do *Scrum* descritas anteriormente.

Figura 4 - Representação visual dos principais artefatos do Framework *Scrum*



Fonte: Adaptado de Sutherland (2020).

O projeto foi desenvolvido utilizando *Scrum* por esse modelo conter essas características promovendo um desenvolvimento otimizado com entregas constantes de qualidade.

No próximo capítulo serão apresentadas algumas tecnologias disponíveis para desenvolvimento da aplicação móvel, assim como a qual foi escolhida para o desenvolvimento do projeto.

2.3 FLUTTER

Com o objetivo de facilitar o desenvolvimento de aplicativos móveis, a Google criou uma ferramenta de código aberto e multiplataforma, ou seja, ele possibilita o desenvolvimento de aplicativos que executam em diferentes sistemas operacionais por meio de um único código fonte. O *Flutter* é capaz de produzir aplicativos fiéis ao estilo de interface nativa com alto desempenho, fornecendo uma experiência nativa tanto para os usuários do sistema *Android* quanto para os do sistema *IOS* (BELLINASSO, 2018).

A linguagem de programação utilizada nessa ferramenta é a *Dart* e sua sintaxe assimila-se com a de linguagem *JavaScript*. Uma das características do *Flutter* é a sua estratégia de renderizar os elementos de interface do aplicativo em uma tela canvas, assim como é feita em motores de jogos, esse fluxo de desenvolvimento é orientado ao *design*. Essa característica de desenhar o que está presente na tela em um canvas do dispositivo, promete a entrega de animações mais fluídas e mais responsivas (LELER, 2017).

Segundo Bellinaso (2018) um dos pontos positivos do *Flutter* é o fato de que desenhar os componentes torna previsível a visualização de como será a interface nos dispositivos, uma vez que o que é visto nos teste imediatamente será a visão final nos dispositivos. Outro ponto é a funcionalidade *Hot Reload*² que permite instantaneamente a pré-visualização das alterações do aplicativo, sem precisar reiniciá-lo completamente.

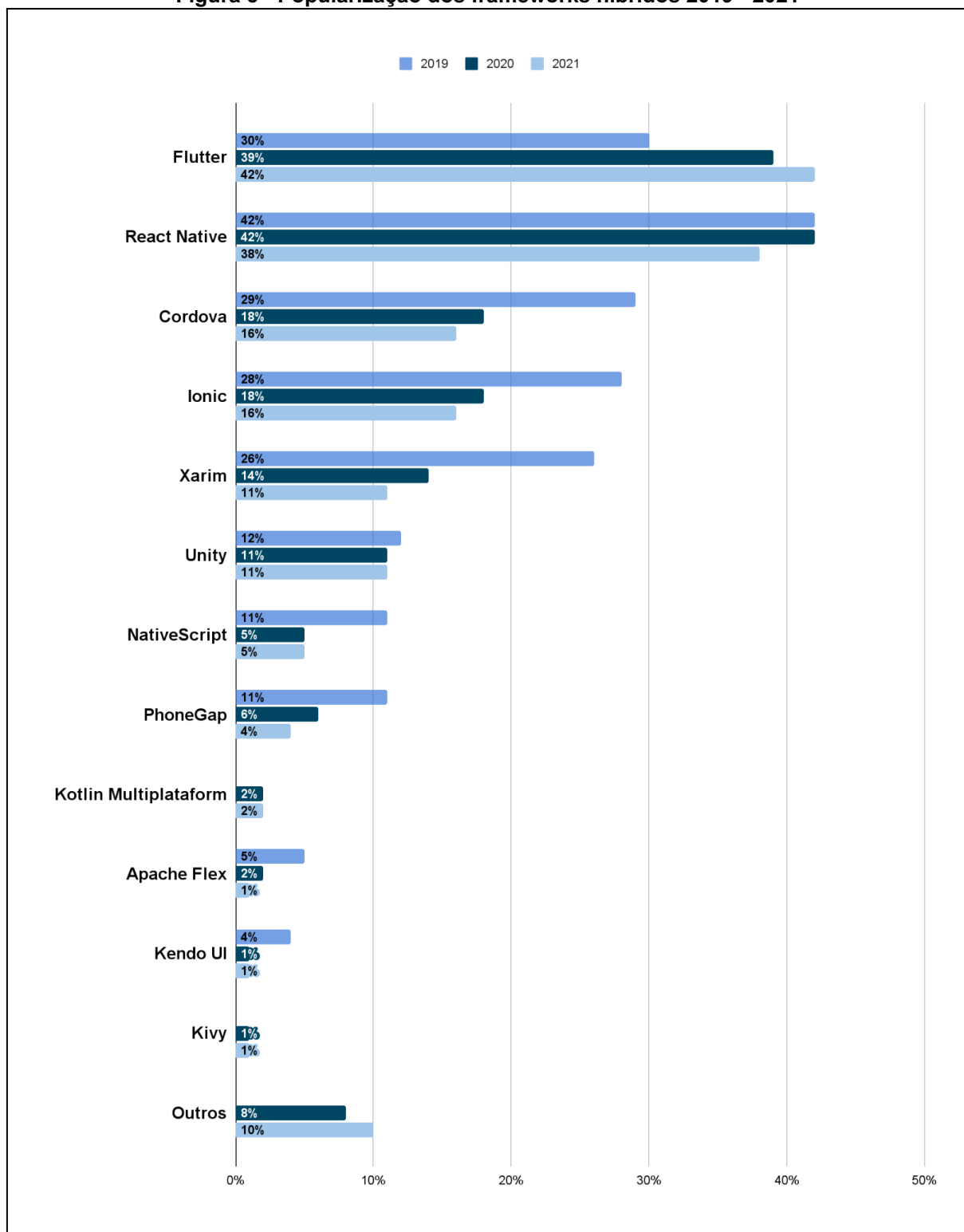
Ainda sobre a análise do autor, alguns pontos negativos são citados como, caso algum sistema operacional atualize a aparência de um componente, um aplicativo criado através do *Flutter* estará automaticamente desatualizado até que seja desenvolvido uma nova versão do aplicativo. Em comparação a comunidade de outras ferramentas, a quantidade de plugins é muito baixa. Caso a interface do aplicativo necessite ser diferente entre as plataformas, será passível de duplicidade de código.

No período de 2019 até 2021, a organização Statista (2021) fez uma pesquisa global perguntando para 31.743 (trinta e um mil setecentos e quarenta e três) desenvolvedores qual o *framework* multi plataforma mais utilizado por eles. Com base nas respostas, 42% (quarenta e dois por cento) dos desenvolvedores responderam que utilizavam *Flutter* para desenvolver seus projetos híbridos.

² <https://docs.flutter.dev/development/tools/hot-reload>

Na Figura 5 é apresentado a popularização crescente do *Flutter* e de outros *frameworks* no período de 2019 até 2021.

Figura 5 - Popularização dos frameworks híbridos 2019 - 2021



Fonte: Adaptado de Statista (2021).

Pela popularidade crescente e pelos pontos positivos se sobreporem aos pontos negativos, o *Flutter* foi a ferramenta escolhida para o desenvolvimento do aplicativo.

2.4 INTEGRAÇÃO ENTRE SISTEMAS

Diferentes tipos de sistemas são encontrados em organizações públicas e privadas das mais diversas áreas e eles atuam na gerência do negócio e na automação de tarefas rotineiras. Esses sistemas podem ser de nível operacional, gerencial ou estratégico (GARCIA; ABILIO; MALHEIROS, 2015).

Para organizações que possuem diferentes tipos de sistema e as informações entre eles precisam ser compartilhadas, é essencial que exista uma integração entre os sistemas, pois dessa maneira os processos internos serão mais eficientes com redução no tempo de execução de determinadas atividades (BISOL, 2019).

Martins (2005) define que esse tipo de abordagem é uma expansão da informação e dos processos em uma organização, seja ela através de base de dados ou aplicativos.

Segundo Garcia, Abilio e Malheiros (2015), existem razões que podem levar uma empresa a contratar ou desenvolver soluções de integração de sistema que são elas:

- Utilizar uma tecnologia já existente para reduzir custos na implantação de novos serviços.
- Permitir a integração com *stakeholders*, expandindo o alcance dos serviços.
- Concentrar na mesma plataforma as informações comuns vindas de diferentes sistemas.

O objetivo da integração de sistemas é facilitar a comunicação entre os dados e os processos otimizando o trabalho das equipes no ambiente corporativo (MARTINS, 2005).

A seguir serão apresentados os conceitos necessários para o para que haja a comunicação entre o aplicativo e o sistema legado da empresa.

2.4.1 Web Services

Com o crescimento da quantidade de aplicações existentes e da quantidade das novas linguagens de programação que surgiram, encontrou-se a necessidade de integrar sistemas. Para solucionar estes problemas de integração entre sistemas distintos foi criada a tecnologia dos *web services* (MENÉNDEZ, 2002).

Desenvolver aplicativos móveis tornou-se uma tarefa mais simples com a disponibilização dos serviços dos *web services*, pois eles ficam responsáveis por fornecer dados enquanto os aplicativos ficam responsáveis por solicitar e apresentar os dados para os usuários (MURAKAMI; MARQUES; LIMA, 2013).

O conceito cliente servidor foi introduzido a partir do momento em que surgiram as primeiras redes de computadores. Os servidores são responsáveis por disponibilizar serviços para a rede de computadores ao qual está inserida e quem utiliza esses serviços são denominados de clientes (OLIVEIRA, 2011).

Web service é um sistema desenvolvido com o objetivo de permitir que outros sistemas interajam entre si, de diferentes linguagens de programação através da internet. Esse sistema permite que aplicações enviem e recebam dados em formatos variados. Essa troca de dados ocorre por formatos como XML³ e JSON⁴. Os meios de transmissão de dados podem ser através de protocolos de rede como o HTTP, SMTP, FTP ou protocolos de mensagem proprietários (MORO; DORNELES; REBONATTO, 2011).

O HTTP é um protocolo de comunicação referenciado por uma URI⁵. Ele é o principal protocolo da internet e a sua função é realizar requisições e receber resposta entre um cliente e servidor (EMER, 2014).

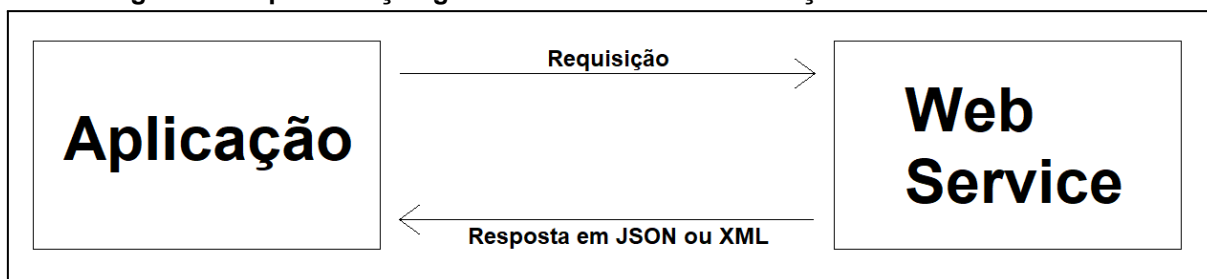
³ www.w3schools.com/xml/xml_what_is.asp

⁴ www.w3schools.com/js/js_json_intro.asp

⁵ <https://developer.mozilla.org/pt-BR/docs/Glossary/URI>

Para haver uma comunicação, é necessário um emissor e um receptor. A Figura 6 representa o fluxo da troca de dados entre um *web service* e uma aplicação (OLIVEIRA, 2011).

Figura 6 - Representação gráfica do fluxo de comunicação de um *Web Service*.



Fonte: Adaptado de Oliveira (2011).

Uma das vantagens da utilização de *web service* é de que o cliente não precisa ser da mesma linguagem de programação do servidor. Outra vantagem é da capacidade de prover serviços que podem ser usados internamente como, por exemplo, na sincronização de dados entre base de dados que estejam em locais diferentes e em dispositivos diferentes, dessa maneira não existe a necessidade de ser um serviço aberto ao público, mas também, esses serviços podem ser abertos a consultas externas que podem ser usado por qualquer aplicativo (COSTA et al., 2010).

A composição da estrutura de um *web service* é feita por um serviço e a descrição do serviço. O serviço é um módulo desenvolvido e disponível em um servidor que tem acesso a rede e a descrição do serviço é um conjunto de informações com detalhes do serviço, como as operações permitidas, os tipos de dados, informações de aquisições do serviço e a localização na rede.

Oliveira (2011) descreve que os *web services* são *softwares* criados para apoiar interações entre máquinas em uma rede, fornecendo uma interface descrita processável por máquina.

Richardson (2007) relata que os *web services* cresceram muito na quantidade de uso e na sua complexidade fazendo com que surgissem uma grande quantidade de padrões. Um dos padrões que surgiu foi o SOAP⁶ que para desenvolver um serviço utilizando essa arquitetura é necessário conhecer, além da

⁶ www.w3schools.com/xml/xml_soap.asp

própria arquitetura, padrões como XML, WSDL⁷, UDDI⁸, WS-Policy⁹ entre outros elementos. Em contrapartida, a arquitetura REST surgiu com o objetivo de facilitar a criação de *web service* através da utilização de padrões básicos como HTTP, URI, XML e JSON.

No próximo capítulo será apresentado o estilo arquitetural REST que foi escolhido para disponibilizar os serviços necessários para que haja a comunicação do aplicativo com o banco de dados da organização.

2.4.2 Arquitetura REST

REST é um estilo arquitetural de *software* voltado para comunicação entre *web services* com funções simples e acessíveis que podem ser executadas de forma síncrona ou assíncrona. As URIs e seus serviços são os elementos principais dessa arquitetura (MURAKAMI; MARQUES; LIMA, 2013). Os serviços são identificados pela URI (GARCIA; ABILIO; MALHEIROS, 2015).

A comunicação entre o *web service* REST e o cliente ocorre de maneira que o cliente realiza o envio de uma requisição HTTP, com as informações necessárias, para o servidor de aplicação, o servidor processa a requisição e devolve uma resposta contendo dados (BISOL, 2019).

A troca de dados ocorre através de padrões XML ou JSON e a escolha de qual padrão usar deve ser definida de acordo com a necessidade do ambiente. O protocolo de transporte HTTP contém métodos, também chamados de verbos, que são utilizados para a manipulação de recursos (KALIN, 2010). São eles:

- POST: Esse método é utilizado quando existe a necessidade de criar um registro em uma coleção.
- GET: Método utilizado para operações somente de leitura que obtém informações de um determinado registro no servidor.
- PUT: O verbo Put é utilizado para atualizar registros de uma coleção do servidor.

⁷ www.w3schools.com/xml/xml_wsdl.asp

⁸ www.ibm.com/docs/pt-br/rsas/7.5.0?topic=standards-universal-description-discovery-integration-uddi

⁹ <https://www.ibm.com/docs/pt-br/was-nd/9.0.5?topic=services-ws-policy>

- DELETE: Esse método é utilizado para apagar registros da coleção.

É recomendado a utilização de cabeçalhos para a transmissão de dados relacionados a uma requisição ou resposta HTTP. O cabeçalho é um campo que passa informações adicionais melhorando a precisão da semântica da mensagem ou do corpo. Por exemplo, é através do cabeçalho que é informado o formato de resposta esperado em um resposta HTTP (FELLER, 2010).

Em uma resposta HTTP, deve-se utilizar de forma correta o código de status adequado à semântica do resultado. Por exemplo, em uma requisição GET onde o recurso foi acessado corretamente, a resposta HTTP deve retornar com status 200 (Ok), caso o recurso não exista o status deve ser o 404 (*Not Found*) (FELLER, 2010).

Segundo FELLER (2010), a arquitetura REST é consideravelmente mais simples de ser utilizada em relação a outras existentes e isso é apresentando dado a quantidade de *web services* que o utilizam.

2.5 BANCO DE DADOS

Para Date (2004), banco de dados é um sistema que tem o objetivo de registrar e manter informações, ou seja, um sistema capaz de armazenar, gerenciar e realizar manutenção de registros eletrônicos. Esse sistema disponibiliza as informações armazenadas para serem pesquisadas, atualizadas e filtradas pelos usuários ou pela a organização que necessite desses registros.

Para gerenciar os registros eletrônicos foi utilizado o SGDB (Sistema Gerenciador de Banco de Dados) PostgreSQL que já estava em uso pela empresa e já continham registros armazenados.

2.5.1 PostgreSQL

PostgreSQL é um sistema de banco de dados relacional de objetos que utiliza a linguagem SQL (*Structured Query Language*). Esse sistema surgiu em 1986

na Universidade da Califórnia em Berkeley em um projeto chamado POSTGRES e tem mais de 30 anos de desenvolvimento ativo na plataforma principal (POSTGRESQL, 2022).

Esse sistema é muito popular, pois é gratuito, compatível com todos os principais sistemas operacionais e também possui código aberto com comunidade ativa que se dedica a mantê-lo atualizado fornecendo soluções inovadoras e de desempenho consistentes (POSTGRESQL, 2022).

O *PostgreSQL* está em conformidade com 170 dos 179 recursos obrigatórios para conformidade com SQL 2016 *Core* nas versões a partir da 14. Até o momento, nenhum banco de dados relacional atende totalmente a esse padrão (POSTGRESQL, 2022).

Nesse projeto, foi utilizado o *PostgreSQL* na versão 14.5, o qual é responsável pelo armazenamento de registros assim como sua consulta, interagindo diretamente com o *web service*. O aplicativo não se comunica diretamente com a base de dados real da organização.

2.6 AVALIAÇÃO DA QUALIDADE DE PRODUTO DE SOFTWARE

A competitividade entre as empresas ficou mais acirrada com o processo de globalização da economia e o caminho encontrado por essas empresas para se manter no mercado foi oferecer serviços de qualidade (STORCH, 2000).

Por muito tempo, para que um *software* fosse considerado de boa qualidade, bastava apenas garantir que o sistema não contivesse erros (CÔRTES; CHIOSSI, 2001). O desenvolvedor desenvolvia o sistema e também testava, pois não existiam pessoas dedicadas apenas para testar o sistema. A validação do *software* ocorria quando o desenvolvimento estava quase ou totalmente concluído. Essa abordagem passou a ser considerada má prática no processo de desenvolvimento de *software* (BARTIÉ, 2002).

Com o decorrer do tempo o cliente e o mercado começaram a ter maior impacto nas definições de características desejáveis em um *software*. A tecnologia também evoluiu e assim surgiu uma grande quantidade de recursos e características

disponíveis em um *software*. Com o surgimento desses novos elementos, eles passam a ser importantes componentes na avaliação de qualidade do produto (CÔRTEZ; CHIOSSI, 2001).

Segundo Sibisi (2007) qualificar um *software* é uma tarefa difícil, pois as definições de qualidade variam e depende do ponto de vista do avaliador e dos envolvidos. De forma geral, pode ser entendida que a qualidade de *software* é determinada pela quantidade de características atendidas as necessidades implícitas e explícitas de seus usuários (DUARTE; FALBO, 2006).

Para Alsultanny e Wohaiishi (2009), aplicar métricas de qualidade para produto e processo de *software* é uma tarefa complexa e exige conhecimento dos objetivos que se pretende alcançar. O efeito da falta de qualidade em um *software* faz com que as empresas de *software* percam credibilidade no mercado causando prejuízo tanto para a empresa quanto para os clientes insatisfeitos.

2.6.1 Norma ISO/IEC 9126

O Organismo Internacional de Padronização, conhecido como ISO (*International Organization for Standardization*) criou a norma ISO/IEC 9126 com o objetivo de auxiliar, padronizar e aprimorar a avaliação da qualidade de um produto de *software* (ROCHA; MALDONADO; WEBER, 2001).

Segundo a norma ISO/IEC 9126, a qualidade de um *software* pode ser avaliada através de um conjunto de seis características (ROCHA; MALDONADO; WEBER, 2001).

O Quadro 1 apresenta as características de avaliação de qualidade de software da norma ISO/IEC 9126 e suas respectivas descrições.

Quadro 1 - Características de avaliação de qualidade ISO/IEC 9126

Funcionalidade	Característica responsável por avaliar a existência de um conjunto de funções que atendam as necessidades explícitas ou implícitas do <i>software</i> .
Confiabilidade	Condiz com a capacidade do <i>software</i> manter seu desempenho sob condições específicas para um determinado período de tempo.

Usabilidade	Esta característica analisa o esforço necessário do usuário para o aprendizado e utilização do sistema, assim como o julgamento individual do uso por um conjunto de usuários.
Eficiência	Afere a relação entre o desempenho do <i>software</i> e a quantidade de recursos utilizados sob condições específicas.
Manutenibilidade	Característica que avalia o esforço necessário para modificar, localizar e reparar erros no <i>software</i> .
Portabilidade	Responsável por avaliar a capacidade de um <i>software</i> ser transferido de um ambiente para outro.

Fonte: Autoria própria (2022).

Ainda segundo a norma, cada característica é dividida em subcaracterística onde é apresentado no Quadro 2, junto de sua descrição com sua pergunta chave;

Quadro 2 - Subcaracterísticas de avaliação de qualidade ISO/IEC 9126

FUNCIONALIDADE		
Adequação	Avalia a presença de adequação de um conjunto de funções para tarefas específicas e objetivos do uso.	Propõe-se a fazer o que é apropriado?
Acurácia	Avalia a precisão de resultados ou efeitos corretos ou conformidade acordados.	Faz o que foi proposto de forma correta?
Interoperabilidade	Avalia a capacidade do <i>software</i> em interagir com um ou mais sistemas específicos.	Interage com os sistemas especificados?
Conformidade	Avalia a conformidade do <i>software</i> em relação a convenções ou regulamentações legais e prescrições similares.	Está de acordo com as normas, leis e etc.?
Segurança de acesso	Avalia a habilidade de qualquer <i>software</i> com relação a proteção de dados na prevenção de acesso não autorizado a programas e dados.	Evita o acesso não autorizado aos dados?
CONFIABILIDADE		
Maturidade	Avalia a capacidade do <i>software</i> em manter seu nível de performance, a frequência de erros devido à falhas no <i>software</i> . Aborda a precisão dos dados, a ocorrência de erros durante a utilização do <i>software</i> e controle de integridade.	Com que frequência apresenta falhas?
Tolerância a falhas	Avalia a capacidade de qualquer <i>software</i> em manter seu nível de performance quando da ocorrência de falhas no <i>software</i> . Avalia a capacidade do <i>software</i> controlar as ações do usuário, tratando de situações anormais e o desempenho do sistema com relação ao volume de dados.	Ocorrendo falhas, como ele reage?
Recuperabilidade	Avalia a capacidade do <i>software</i> de restabelecer seu nível de desempenho e recuperar os dados	É capaz de recuperar dados em caso de falha?

	diretamente afetados em caso da ocorrência de falhas, no prazo e esforço necessário para isto.	
USABILIDADE		
Inteligibilidade	Avalia o esforço do usuário para conhecer os conceitos lógicos do <i>software</i> e sua aplicabilidade.	É fácil entender o conceito e a aplicação?
Apreensibilidade	Avalia o esforço requerido pelo usuário para operar o <i>software</i> .	É fácil aprender a usar?
Operacionalidade	Avalia o esforço do usuário para a operação e controle de operação das funções do <i>software</i> .	É fácil de operar e controlar?
EFICIÊNCIA		
Tempo	Avalia o tempo de processamento e respostas e a taxa de tempo no desempenho de sua função.	Qual é o tempo de resposta, a velocidade de execução?
Recursos	Avalia a quantidade de recursos utilizados para atingir o nível de desempenho de uma função desejada pelo <i>software</i> .	Quanto recurso usa? Durante quanto tempo?
MANUTENIBILIDADE		
Analisabilidade	Avalia o esforço necessário para diagnosticar deficiências ou causas de falhas, ou identificação de partes a serem modificadas.	É fácil de encontrar uma falha, quando ocorre?
Modificabilidade	Avalia-se o esforço necessário para fazer modificações, remoção de erros, ou para modificações de ambientes.	É fácil modificar e adaptar?
Estabilidade	Avalia os riscos de efeitos inesperados devido a modificações. Avalia os erros que podem surgir com as modificações.	Há grande risco quando se fazem alterações?
Testabilidade	Avalia o esforço necessário para validação do <i>software</i> modificado.	É fácil testar quando se fizer alterações?
PORTABILIDADE		
Adaptabilidade	Avalia os atributos do sistema que suportam a habilidade de adaptação do <i>software</i> para diferentes ambientes sem aplicar outras ações ou meios do que provêm os propósitos do <i>software</i> considerado.	É fácil adaptar a outros ambientes?
Capacidade para ser instalado	Avalia o esforço necessário para instalar o <i>software</i> em ambiente específico. Inclui os recursos de auxílio fornecidos pelo <i>software</i> .	É fácil instalar em outros ambientes?
Conformidade	Avalia os atributos do <i>software</i> em manter padrões ou convenções relativas à portabilidade.	Está de acordo com padrões de portabilidade?

Capacidade para substituir	Avalia a habilidade do <i>software</i> ser usado no lugar de outro <i>software</i> especificado no ambiente deste <i>software</i> .	É fácil usar para substituir outro?
----------------------------	---	-------------------------------------

Fonte: Aatoria própria (2022).

Segundo Storch (2000) a norma ISO/IEC 9126, a avaliação da qualidade de produto de *software* define-se à medição de quanto esse produto, em uso, atende às necessidades implícitas e explícitas. Não existe um sistema de classificação de *software* amplamente aceito, o que existe são classes de *softwares* com características com maior importância em relação a outras. Um *software* de missão crítica, por exemplo, a característica de confiabilidade é a mais importante, para *softwares* de tempo real a característica de eficiência é a mais importante, já a característica de usabilidade é a mais importante para *softwares* interativos em relação ao usuário final. A característica de qualidade depende do ponto de vista considerado.

O autor ainda explica que para os usuários, a qualidade de um sistema é definida principalmente pela confiabilidade, eficiência e facilidade de uso. São levantadas questões como:

- O *software* é confiável?
- O *software* é eficiente?
- É fácil usar o *software*?

Considerando que o foco do trabalho é agilizar e dar mobilidade no processo de lançamentos de pedidos utilizado pelos vendedores, as características e subcaracterísticas utilizadas para criar o formulário de avaliação de qualidade foram baseados nas métricas que avaliam o nível de satisfação dos usuários.

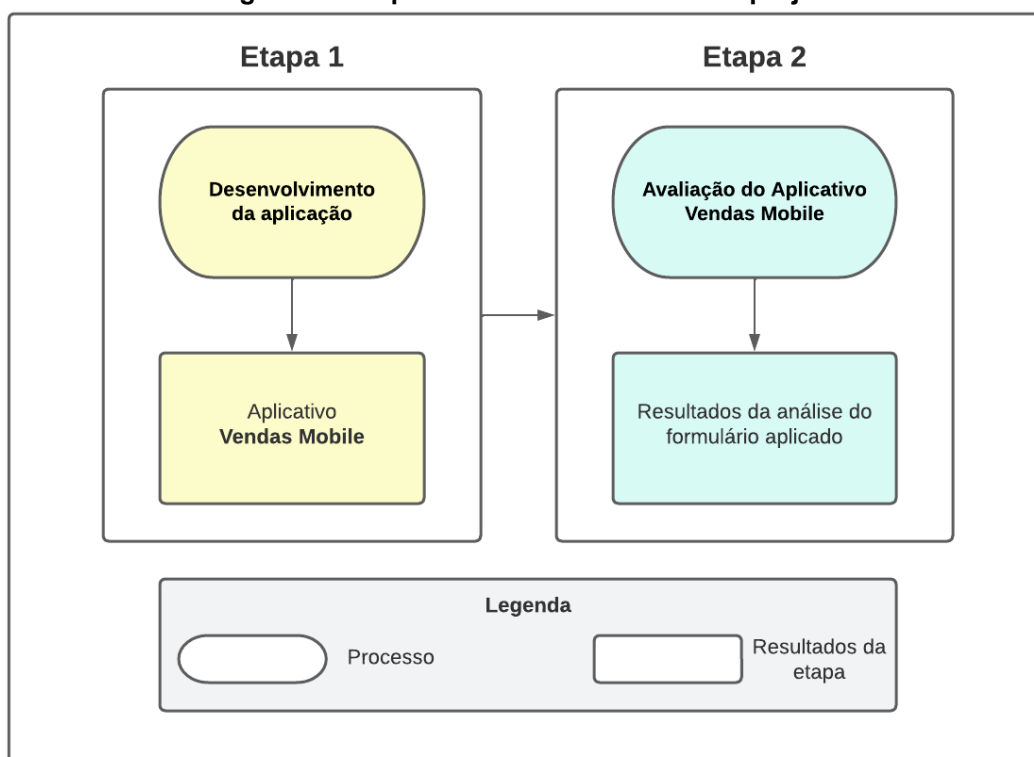
3 METODOLOGIA

Este capítulo apresenta os encaminhamentos metodológicos da pesquisa. A seção 3.1, apresenta a organização e o desenvolvimento do trabalho e a descrição das etapas do processo de realização do trabalho, sendo elas: Desenvolvimento da aplicação, criação e aplicação do formulário de avaliação do aplicativo e análise dos resultados. A seção 3.2 discorre sobre o processo de coleta de dados da pesquisa.

3.1 ORGANIZAÇÃO E DESENVOLVIMENTO DO TRABALHO

O trabalho foi operacionalizado por meio de duas etapas, a primeira diz respeito ao desenvolvimento do aplicativo móvel para o módulo de vendas e a segunda a avaliação do produto pelos usuários e análise dos resultados. A Figura 7 ilustra as etapas para a realização do trabalho.

Figura 7 – Etapas do desenvolvimento do projeto



Fonte: Autoria própria (2022)

A seguir segue a descrição das etapas aplicadas na realização deste trabalho.

3.1.1 Desenvolvimento da aplicação

A primeira etapa teve como objetivo utilizar a metodologia ágil *Scrum* para o desenvolvimento do aplicativo intitulado como Vendas Mobile¹⁰, tendo referência o módulo de vendas do sistema *web* utilizado pela empresa Insumos LTDA.

¹⁰ Nome fictício do aplicativo

Foram consideradas as informações referente as necessidades, restrições dos usuários do sistema, as particularidades do módulo de vendas dos sistema *web* utilizado pela empresa e as concepções teóricas levantadas sobre o *framework* de desenvolvimento ágil *Scrum* levando-se em consideração a criação do *Product Backlog*, que consiste em uma listagem de necessidades do *software* organizado em ordem de execução. As *Sprints Backlog* foram criadas e para cada uma delas foi definida uma lista agrupada de atividades a serem finalizadas em um período de tempo bem como as *Daily Scrum*, sendo essas definidas como reuniões realizadas pela equipe de forma adaptada.

Para a criação do produto foi utilizado o *framework Flutter* por ser muito utilizado entre a comunidade de desenvolvedores (STATISTA, 2021) e por ter capacidade de criar versões para os sistemas operacionais *Android* e *IOS* (BELLINASSO, 2018). Para o desenvolvimento dos serviços, foram utilizados os conceitos da arquitetura *REST* e a linguagem *PHP*, escolhida por já ser utilizada pela empresa. A última *Sprint* que foi denominada de “Avaliação da Aplicação” tem por objetivo avaliar o aplicativo criado e os resultados dessa fase são apresentados na etapa 2, descrita a seguir.

3.1.2 Avaliação do Aplicativo Vendas Mobile

Esta etapa teve como objetivo verificar se foram solucionados os problemas acarretados pela falta de mobilidade do módulo de vendas da empresa Insumos LTDA. Para isso foi criado um formulário para avaliar a qualidade do aplicativo Vendas Mobile.

O aplicativo já estava em uso há um ano, porém as funcionalidades foram liberadas para o uso de forma gradativa, conforme as sprints iam sendo finalizadas. O produto foi entregue no início de setembro de 2022. Inicialmente a liberação do uso do aplicativo foi feita apenas na filial de Castro - PR, pois um dos integrantes do time de desenvolvimento trabalhava na filial e isso poderia possibilitar que os usuários contribuíssem com alguma observação em relação às funcionalidades do aplicativo.

Após o fim da implantação, foram disponibilizadas 21 (vinte e uma) questões, onde 17 (dezessete) foram desenvolvidas tendo como premissa as características de qualidade previstas na norma ISO/IEC 9126 e 4 (quatro) foram desenvolvidas para identificar o perfil do usuário. Para as dezessete questões foram utilizadas as características de: funcionalidade, confiabilidade, usabilidade e eficácia. O tempo disponibilizado para o preenchimento do formulário foi de uma semana, entre os dias 3 de outubro até o dia 9 de outubro de 2022.

3.1.2.1 Localização e participantes da pesquisa

Inicialmente o formulário da pesquisa foi apresentado para os gerentes de cada unidade em uma reunião na filial da empresa na cidade de Itapeva - SP, no dia 1 de outubro de 2022, com o objetivo de explicar a finalidade de aplicar a pesquisa para os funcionários que utilizam o aplicativo. Foi exposto que os principais objetivos da pesquisa eram de levantar informações a respeito do aplicativo Vendas Mobile.

Na reunião com os cinco gerentes, cada um de uma das filiais, foi realizada uma explicação item a item do formulário de avaliação. Foram sanadas dúvidas sobre algumas das perguntas e prazo máximo de preenchimento. Os gerentes ficaram encarregados de repassar o formulário para os seus respectivos consultores de vendas e auxiliá-los em caso de eventuais dúvidas no momento do preenchimento.

O formulário foi aplicado para 25 (vinte e cinco) usuários do Vendas Mobile, sendo eles 5 (cinco) gerentes e 20 (vinte) vendedores, todos com acesso às mesmas funcionalidades, portanto o formulário para ambas as funções é o mesmo.

3.1.2.2 Coleta de dados

Para a realização da coleta de dados para o desenvolvimento do projeto, foram consideradas todas as etapas do desenvolvimento da pesquisa, conforme descrito na seção 3.1. Os dados para a primeira etapa foram coletados por meio de

conversas informais com os consultores de venda e reuniões com os gestores e diretores da empresa.

Nas reuniões foram ponderadas as necessidades que os funcionários não levaram em consideração, avaliação dos dados necessários para preenchimento no banco de dados e também as telas do sistema *web*.

O levantamento de dados da segunda etapa se deu por meio de pesquisa sobre o tema de avaliação de qualidade da ISO/IEC 9126 em artigos acadêmicos e também em outros trabalhos de conclusão de curso. Nas pesquisas em outros trabalhos, foi verificado o método de aplicação de avaliação de qualidade, que serviram como base para o desenvolvimento do formulário aplicado. Os dados coletados foram validados pela equipe *Scrum*, para o alinhamento das etapas a serem desenvolvidas.

No capítulo seguinte, serão apresentadas as análises e os resultados obtidos em cada uma das 2 (duas) etapas da pesquisa.

4 ANÁLISE E RESULTADOS DA PESQUISA

Neste capítulo, são discutidos os resultados e a análise dos dados de cada uma das duas etapas previstas na pesquisa. A seção 4.1 apresenta os resultados da etapa um, na qual é feita a descrição da utilização da *framework Scrum* no desenvolvimento da aplicação, descrevendo cada uma das *sprints* e seus produtos. A seção 4.2 traz os resultados da etapa dois da pesquisa, que relata a criação e aplicação do formulário de avaliação do produto.

4.1 DESENVOLVIMENTO DA APLICAÇÃO

A primeira etapa da pesquisa foi desenvolvida uma aplicação móvel para a Insumos LTDA. A empresa foi criada em 1965 (mil novecentos e sessenta e cinco) na cidade de Castro, Paraná e atua no ramo de revenda de insumos agrícolas. A matriz encontra-se na cidade de Castro, e quatro outras filiais que estão localizadas nas cidades de Irati e Ponta Grossa, ambas no estado do Paraná e outras duas, são no estado de São Paulo, nas cidades de Itapeva e Taquarituba. O quadro de colaboradores, conta com 61 (sessenta) funcionários, dentre eles 25 (vinte e cinco) são vendedores e 36 (trinta e seis) são da parte administrativa em geral.

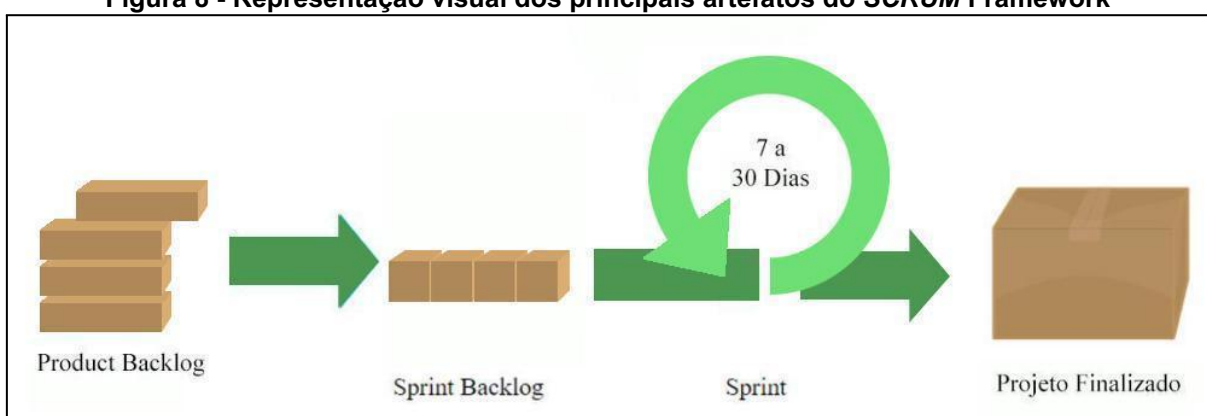
Para auxiliar nos seus processos gerenciais, durante os cinquenta e sete anos de existência da empresa, foram utilizados três sistemas informatizados, os quais foram sendo migrados até a versão atual, sendo eles: dois sistemas em servidor local e um sistema *web*. Esse último, foi implantado em 2014 (dois mil e quatorze) e vem sendo utilizado até os dias atuais, ou seja, há mais de oito anos.

O sistema *web* proporciona facilidade de uso aos seus usuários como a possibilidade de acesso em qualquer lugar com internet através de computadores de mesa ou notebooks, porém quando a utilização é através de *smartphones* o sistema torna-se de difícil utilização, pois ele não dispõe de interface responsiva para telas menores. A adesão do módulo de vendas como aplicativo móvel se deu pela possibilidade de criar uma interface “amigável” para a utilização do usuário.

Para auxiliar no processo de desenvolvimento do Vendas Mobile, optou-se pelo uso do *Scrum*, que conforme descrito na seção 2.2.1, trata-se de uma metodologia ágil de desenvolvimento de software, no qual os processos são divididos em ciclos, que facilita o processo de validação de cada parte desenvolvida e cada divisão representa um conjunto de atividades que devem ser executadas, direcionando o desenvolvimento de maneira clara e objetiva.

A Figura 8 apresenta a adaptação feita na ferramenta de metodologia ágil *Scrum* para o desenvolvimento da aplicação proposta.

Figura 8 - Representação visual dos principais artefatos do SCRUM Framework



Fonte: Adaptado de Sutherland (2010)

O aplicativo móvel foi desenvolvido com a linguagem de programação *Dart* por meio do *framework Flutter*, que trabalha com os componentes nativos de cada plataforma. Ela é de fácil entendimento e produz aplicações intuitivas, tendo como maior atrativo a portabilidade do código entre os sistemas: *IOS*, *Android*, *Web* e *Desktop*.

O *web service* para integração entre o sistema móvel e o sistema desktop, foi desenvolvida em *PHP*, linguagem de programação muito utilizada para desenvolvimento *web*, pois nela existe a possibilidade de mesclar com código *HTML*. O modelo de arquitetura *REST* foi utilizado para disponibilização de acesso a recursos do servidor e o acesso aos registros se deu através do banco de dados objeto-relacional *PostgreSQL*.

A seguir serão descritos os papéis da equipe *Scrum*, juntamente dos responsáveis por cada um deles no desenvolvimento do aplicativo Vendas Mobile.

4.1.1 Equipe

Para construção da equipe *Scrum*, foram utilizados os papéis descritos no capítulo 2.2.1. Existem três papéis principais na equipe, sendo elas o Dono do Produto, o Mestre *Scrum* e o time de desenvolvimento.

Como descrito no capítulo 2.2.1, o papel do Dono do Produto é representar os interesses do cliente, ele é responsável pela gestão do projeto, atribuindo tarefas de acordo com o desenvolvimento do trabalho de forma a otimizar as etapas. O encarregado deste papel foi o orientado Lucas Hiro, pois ele era o funcionário que tinha contato direto com a empresa e com o sistema *web* e dessa forma poderia verificar qual a ordem das tarefas a serem realizadas.

O Mestre *Scrum* é o responsável pela aplicação das regras do *Scrum*, ele se faz necessário por possuir o conhecimento dos processos do *Scrum* e gerencia os envolvidos no desenvolvimento do projeto. Esse papel foi atribuído à orientadora professora Dra. Eliana C. M. Ishikawa.

Por fim, o papel de time de desenvolvimento foi atribuído aos orientados Lucas Hiro e Brunno Marcel, onde foram responsáveis por aplicar as funções delegadas pelo Mestre *Scrum* e Dono do Produto.

A seguir serão descritas a execução das etapas previstas na metodologia ágil *Scrum* para o desenvolvimento do aplicativo Vendas Mobile.

4.1.2 Product Backlog

Como descrito no capítulo 2.2.1, o *Product Backlog* é uma lista criada pelo Dono do Produto que contém todas as etapas, elementos e requisitos necessários para o desenvolvimento do produto. Nele é possível verificar quais os itens mais importantes que devem ser priorizados na execução do projeto.

O Quadro 3 apresenta a lista de requisitos levantados e enumerados em ordem de execução e prioridade.

Quadro 3 - Lista de requisitos do Product Backlog.

Nº	Lista de requisitos
1	Levantamento das necessidades do usuário
2	Levantamento dos requisitos
3	Instalação e uso de aplicativos de lançamento de pedidos e de compras na internet
4	Avaliação dos métodos utilizados pelas aplicações e selecionar os itens interessantes
5	Tela com dados necessários para cadastro do aparelho no sistema
6	Processo no <i>web service</i> para verificar se o cadastro esta correto
7	Processo de validação do usuário no aplicativo
8	Telas de aviso em caso de erro
9	Processo no <i>web service</i> para buscar os dados a serem armazenados
10	Desenvolvimento das classes dos objetos a serem usados no aplicativo
11	Processo no aplicativo para armazenar os dados resgatados
12	Busca dos clientes e endereços dentro do aplicativo
13	Desenvolvimento da tela de lançamento de produto
14	Desenvolvimento da tela de escolha do produto (lista dos produtos armazenados no aplicativo, cálculo do valor com base na data de pagamento escolhida, escolha da cultura e quantidade)
15	Desenvolvimento do processo de armazenamento do pedido dentro do aplicativo
16	Desenvolvimento de lista de pedidos armazenados internamente pelo processo de inclusão
17	Edição da tela de lançamento do pedido para comportar operação de edição e visualização
18	Desenvolvimento de função dentro do <i>web service</i> para armazenamento de pedidos dentro do sistema
19	Tela de listagem de todos os produtos presentes na tabela de preço com quantidade maior que 0 no estoque
20	Desenvolvimento de lista de lotes vs quantidade ao clicar no produto
21	Desenvolvimento de tela de listagem de pedidos com pendência de liberação no sistema (busca pelo <i>web service</i> no banco de dados)
22	Desenvolvimento de tela e Desenvolvimento de tela de listagem de faturamento com pendência de liberação no sistema (busca pelo <i>web service</i> no banco de dados)
23	Avaliação da Aplicação

Fonte: Autoria própria (2022).

Há itens listados na coluna de requisitos que foram elencados levando em consideração o conhecimento prévio dos desenvolvedores sobre o domínio da aplicação atual, troca de informações nas reuniões com a professora orientadora, sugestões da diretoria da empresa e gestores comerciais que discutiram a possibilidade do desenvolvimento de uma aplicação móvel para o módulo de vendas a fim de substituir o pedido em papel.

4.1.3 Sprint Backlog

A *Sprint Backlog* foi definida pelo Dono do Produto juntamente com o time de desenvolvimento. Nela foram definidas as etapas do desenvolvimento, agrupando as tarefas em tópicos para poder quantificar o tempo previsto que cada *Sprint* levará para ser finalizada, seguindo o modelo de planejamento do *Scrum*.

O Quadro 4 apresenta a lista de *Sprints* levantadas enumeradas em ordem de execução e prioridade e também apresenta o tempo de execução para desenvolvimento de cada uma delas.

Quadro 4 - Lista de Categorias de *Sprints* e Tempos de Execução.

Nº da Sprint	Categoria	Nº do Requisito do <i>Product Backlog</i>	Tempo de Execução (dias)
1	Levantamento de Necessidades e Restrições	1, 2, 3 e 4	30
2	Desenvolvimento do módulo de <i>login</i>	5, 6, 7 e 8	15
3	Desenvolvimento de Armazenamento Interno de Dados	9, 10 e 11	15
4	Desenvolvimento da tela de lançamento de pedidos e produtos e funções para salvar dados internamente	12, 13, 14, 15, 16, 17 e 18	30
5	Desenvolvimento do sistema de envio de pedidos (aplicativo e API)	19	7
6	Desenvolvimento de tela de demonstração de produtos em estoque e quantidades no lote por filial	20	7
7	Desenvolvimento de tela / processo de liberação de pedidos com desconto e tela / processo de liberação de faturamento	21 e 22	15
8	Avaliação da Aplicação	23	30

Fonte: Autoria própria (2022).

Conforme ilustrado, foram criadas 8 (oito) *Sprints*, tendo como premissa os requisitos levantados no *Product Backlog* (Quadro 3 - Lista de requisitos do *Product Backlog*) e o tempo de execução de cada cada *Sprint*, totalizando 149 dias para o desenvolvimento do projeto.

- ***Sprint 1*** - Levantamento de Necessidades e Restrições:

Na primeira *Sprint* foram realizadas as pesquisas iniciais quanto às restrições do módulo de vendas do sistema *web* existente, bem como as reais

necessidades dos gestores e vendedores em relação às funcionalidades desejadas para a nova aplicação. Nesse período foram realizadas reuniões que possibilitaram ao Time *Scrum* levantar sugestões de melhorias das funcionalidades com base na experiência dos usuários e de com outras aplicações e no cotidiano.

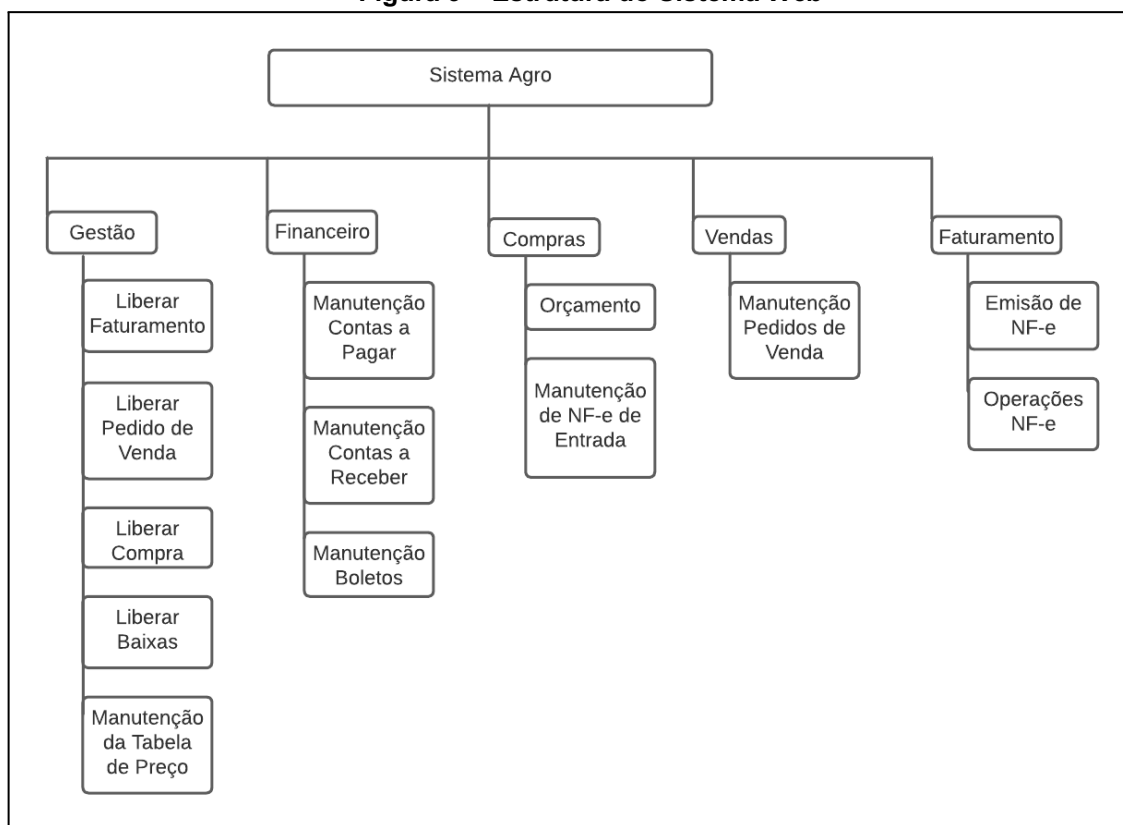
As funcionalidades levantadas levaram em consideração o sistema *web* vigente da empresa, verificando os módulos utilizados para o lançamento completo do pedido, e quais ações seriam realmente necessárias para integração entre o sistema e o aplicativo proposto, visando diminuir a quantidade de campos de inserção de dados sem comprometer as informações do pedido.

Conforme já relatado, a empresa já fez uso de três sistemas informatizados, os quais foram sendo migrados até a versão atual. O sistema *web* foi desenvolvido no framework *GeneXus*, uma ferramenta de desenvolvimento com linguagem própria, que converte o código escrito para linguagens como *Java*, *Swift*, *Javascript* e *Angular* (GENEXUS, 2022).

O sistema possui cinco módulos, sendo eles: Gestão, Financeiro, Compras, Vendas e Faturamento, dessa forma, contempla todas as funcionalidades operacionais como: controle financeiro, lançamento de notas de entrada, faturamento, consulta e cancelamento de notas, ajuste de preço de venda de produtos, conferência de estoque, emissão de relatórios e lançamento de pedidos de venda.

A Figura 9, apresenta uma ilustração das funcionalidades presentes no sistema *web* da empresa.

Figura 9 – Estrutura do Sistema Web



Fonte: Autoria própria (2022)

Cada um dos cinco módulos apresentados na Figura 9 exercem funções específicas, sendo elas:

- **Gestão:** responsável por realizar operações de gerenciamento de certos tipos de atividades, tais como: liberação de diferentes tipos ou precificação dos produtos.
- **Financeiro:** responsável por realizar as operações de controle, baixa e estorno de lançamentos financeiros, e manutenção dos boletos gerados pelo sistema.
- **Compras:** responsável por realizar todo o processo de compra, desde o orçamento até a entrada das notas fiscais dentro do sistema, monitorando e mobilizando o estoque.
- **Vendas:** responsável por realizar o lançamento dos pedidos e controle das quantidades de produtos a serem entregues ao cliente.

- Faturamento: Responsável por realizar as operações de emissão, cancelamento e alteração de diferentes tipos de notas.

Apesar de todos os pontos positivos advindos de um sistema *web*, os usuários da área de vendas, identificaram alguns pontos que restringiam a utilização de forma satisfatória do sistema, principalmente no que diz respeito ao uso das funcionalidades do módulo e na efetivação de pedidos em campo.

O primeiro ponto dizia respeito à dificuldade de uso do sistema *web* em *smartphones*. Mesmo sendo possível acessar o sistema, não existia a responsividade para o uso em telas menores, forçando o usuário a acionar o modo “para computador”, que como os usuários relataram, a utilização era inviável nas situações de emergência, forçando o uso de outros meios como: envio de foto do pedido de venda em papel para outro funcionário lançar pelo computador.

O segundo ponto é a perda dos pedidos físicos, pedidos de venda em papel, que poderia acontecer em vários pontos do processo interno da empresa. Havia a possibilidade de ser perdido enquanto estava com o vendedor, em consultoria com clientes, dentro da empresa, no traslado do pedido entre funcionários para realização das tarefas, e até mesmo na consulta dos pedidos com eles já armazenados para histórico.

Diante das restrições e dificuldades descritas anteriormente, a diretoria da empresa, juntamente dos gestores comerciais, discutiram a possibilidade do desenvolvimento de uma aplicação móvel para o módulo de vendas a fim de substituir o pedido em papel.

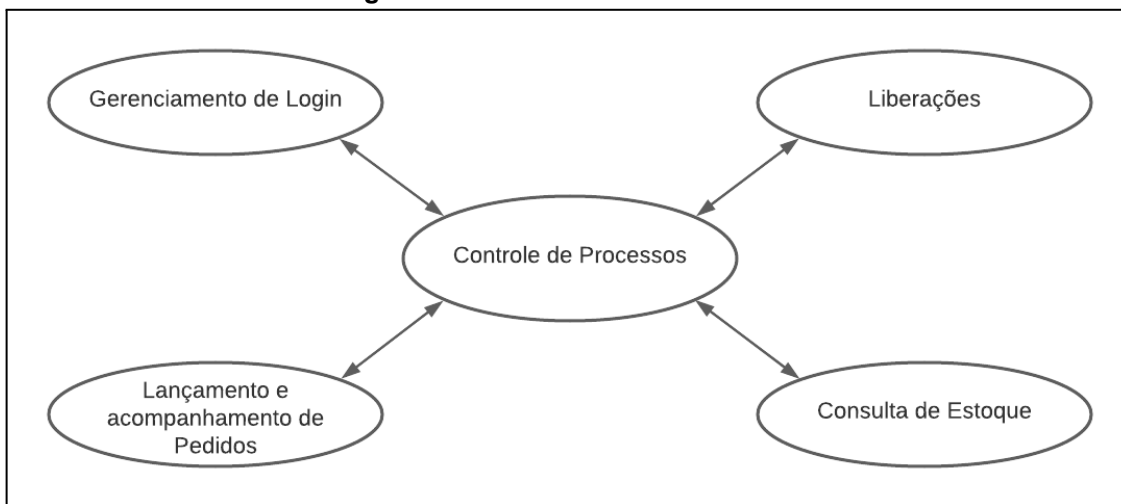
Além da necessidade de aumentar a flexibilidade e mobilidade dos vendedores, o crescimento da empresa e o retrabalho que os pedidos em papel viriam a gerar ao longo dos anos, ratificaram a necessidade de mudar o processo de lançamento de pedidos, do manual em papel para o digital gerando assim a solicitação do desenvolvimento de a aplicação móvel para o módulo de vendas do sistema *web*.

A ideia principal do Vendas Mobile surgiu com a intenção de fornecer aos usuários uma aplicação atualizada, funcional e que facilitasse suas atividades cotidianas, sem depender de um computador ou *notebook* para realizá-las. O aplicativo deveria possuir apenas o módulo de vendas, no qual consiste as

funcionalidades de consulta de estoque e liberação de alçada¹¹. Como a necessidade de liberação de alçada não era apenas para o setor de vendas, foi desenvolvida também a liberação para o setor de compras e financeiro.

Ao final da *Sprint 1*, foi criado um diagrama de Análise de Domínio, conforme ilustrado pela Figura 10 que apresenta os contextos da aplicação do aplicativo de vendas proposto.

Figura 10 – Estrutura do Sistema Móvel



Fonte: Autoria própria (2022)

Por meio de análise foi possível identificar as características do módulo de vendas para a implementação. Verificou-se que há a necessidade do *login* ser vinculado apenas a um dispositivo. Também foi possível identificar quais os dados dos clientes, da tabela de preço e do estoque devem ser apresentados e por fim, foi identificado quais os campos necessários para o lançamento de pedido e quais são os dados necessários para avaliação do pedido para liberação.

- ***Sprint 2*** - Desenvolvimento do Sistema de Login:

Nessa etapa, foram realizados desenvolvimentos tanto na aplicação quanto no *web service*. Na aplicação foram criadas as telas para efetuar o login e no *web service* foram criados e disponibilizados os serviços para as requisições do aplicativo.

Para o desenvolvimento do login foi utilizado um método de autenticação no qual o celular era vinculado ao usuário do sistema ERP¹² (*Enterprise Resource*

¹¹ Liberação por alçada é uma ferramenta pela qual o gestor do departamento determina o limite que cada funcionário pode dar de desconto para determinado material, suprimento ou serviço.

¹² Sistema de software que auxilia na administração de toda a empresa

Planning), tornando assim o uso do sistema restrito, não permitindo a utilização por terceiros.

A validação é realizada no primeiro login no Vendas Mobile, que deve ser feita com acesso a internet, e somente após essa validação será possível utilizar o aplicativo no modo *offline*. A aplicação também verifica se a sua versão é a mesma do *web service*. Para o acesso ao serviço foram definidos dois endereços, caso um deles esteja fora do ar o acesso pode ocorrer através do outro. A aplicação testa o primeiro endereço e caso não receba uma resposta ela testa o segundo. A Figura 11 apresenta parte do código que verifica a versão do aplicativo.

Figura 11 - Verificação de conexão com a *web service* e versão (Parte 1)

```
Future getVersao(String url, String url2) async {
  Map<String, dynamic> data;

  try {
    http.Response response = await http.get(url + 'versao.php');

    if (response != null && response.statusCode == 200) {
      DatabaseHelper dbHelper = DatabaseHelper.instance;

      var versao = await dbHelper.queryRow('dadosAPP', 'id', 1).then((result) {
        return result[0]['versaoAPP'];
      });

      data = jsonDecode(response.body);

      if (versao == data['versao']) {
        return 'Correto';
      } else {
        return 'Incorreto';
      }
    }
  }
}
```

Fonte: Autoria própria (2022)

Através do primeiro endereço o aplicativo envia uma requisição ao *web service* para verificar se eles estão na mesma versão e caso a resposta for positiva, é feito o acesso aos dados.

Em caso de falta de conexão é executado o mesmo processo, porém com o segundo endereço no qual também é verificado se existe conexão e se a versão retornada pelo *web service* é a mesma do aplicativo. Em caso de falta de conexão com o segundo endereço, é concluído pela aplicação que os serviços não estão no ar.

A Figura 12 apresenta parte do código que verifica a versão do aplicativo no caso de falha do primeiro endereço.

Figura 12 - Verificação de conexão com a *web service* e versão (Parte 2)

```
} catch (e) {
  try {
    http.Response response = await http.get(url2 + 'versao_app.php');

    if (response != null && response.statusCode == 200) {
      DatabaseHelper dbHelper = DatabaseHelper.instance;

      var versao = await dbHelper.queryRow('dadosAPP', 'id', 1).then((result) {
        return result[0]['versaoAPP'];
      });

      data = jsonDecode(response.body);

      if (versao == data['versao']) {
        return 'Correto1';
      } else {
        return 'Incorreto';
      }
    }
  } catch (e) {
    return 'Offline';
  }
}
```

Fonte: Autoria própria (2022)

Na tela de *login* ficam disponibilizadas as filiais em que o funcionário pode ter acesso, sendo esse impossibilitado de utilizar o Vendas Mobile caso seja selecionada uma filial na qual seu usuário não esteja cadastrado.

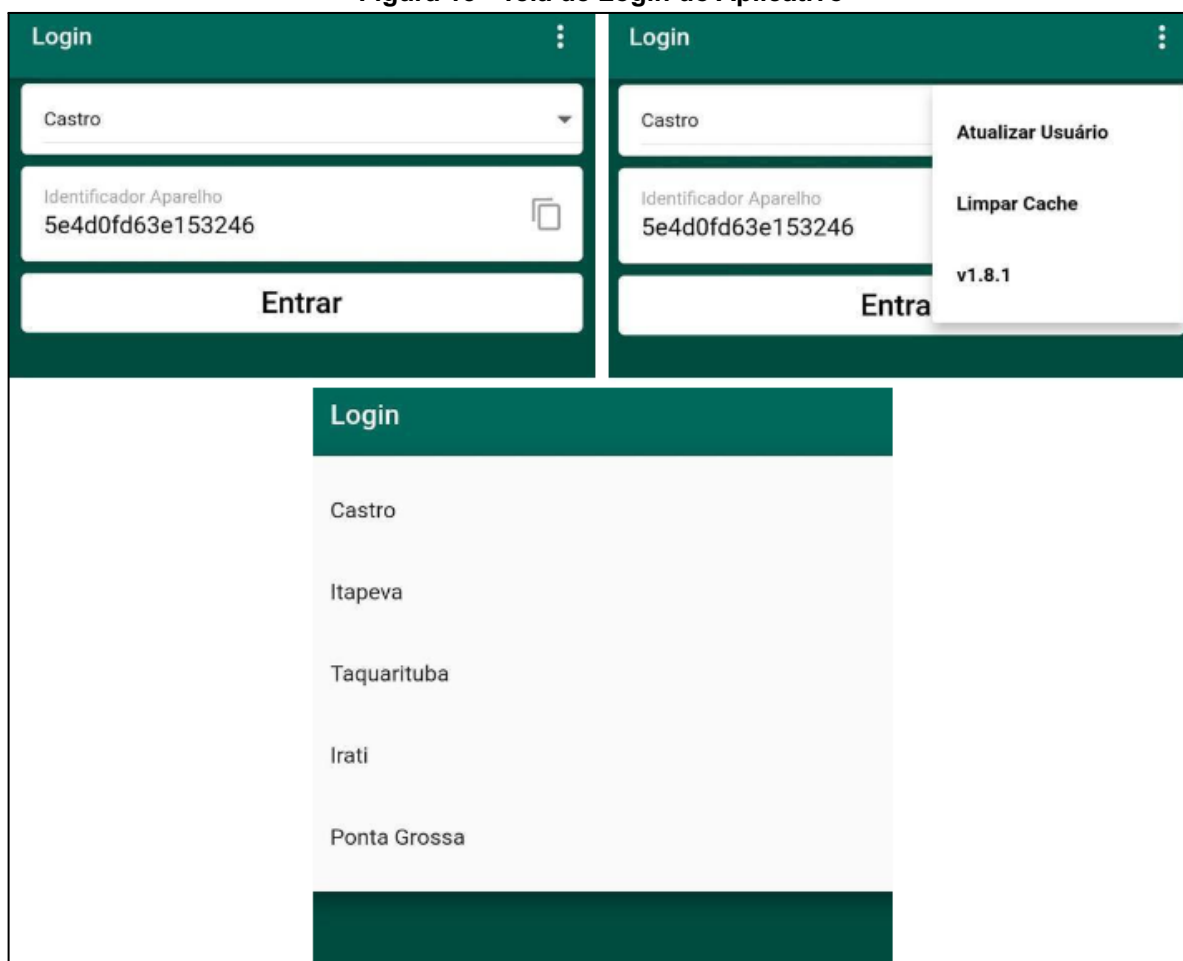
No canto superior direito da tela de login, encontra-se o ícone que dá acesso a três funcionalidades, sendo elas:

- Atualizar usuário: permite que a aplicação consulte os dados no banco para verificar se existe algum usuário com o identificador atrelado;
- Limpar *cache*: dados da aplicação são apagadas;
- Versão da aplicação: acesso a página da aplicação na *Google Play Store*.

Por fim, o botão de entrar possibilita o acesso à aplicação caso o usuário esteja cadastrado e a filial correta seja selecionada.

A Figura 13 apresenta a interface da tela de *login* assim como as opções disponíveis nela.

Figura 13 - Tela de Login do Aplicativo



Fonte: Autoria própria (2022)

- **Sprint 3** - Desenvolvimento de Armazenamento Interno de Dados:

No desenvolvimento do armazenamento interno foi utilizado o levantamento de requisitos para elaboração do menu principal e das funções de atualização de dados. Além de todos os campos utilizados pelo sistema ERP, foi também implementado campos para validação dos dados e caso estejam desatualizados, ao serem enviados ao sistema o pedido será bloqueado para liberação, contendo o motivo do bloqueio na descrição do pedido no sistema *web*.

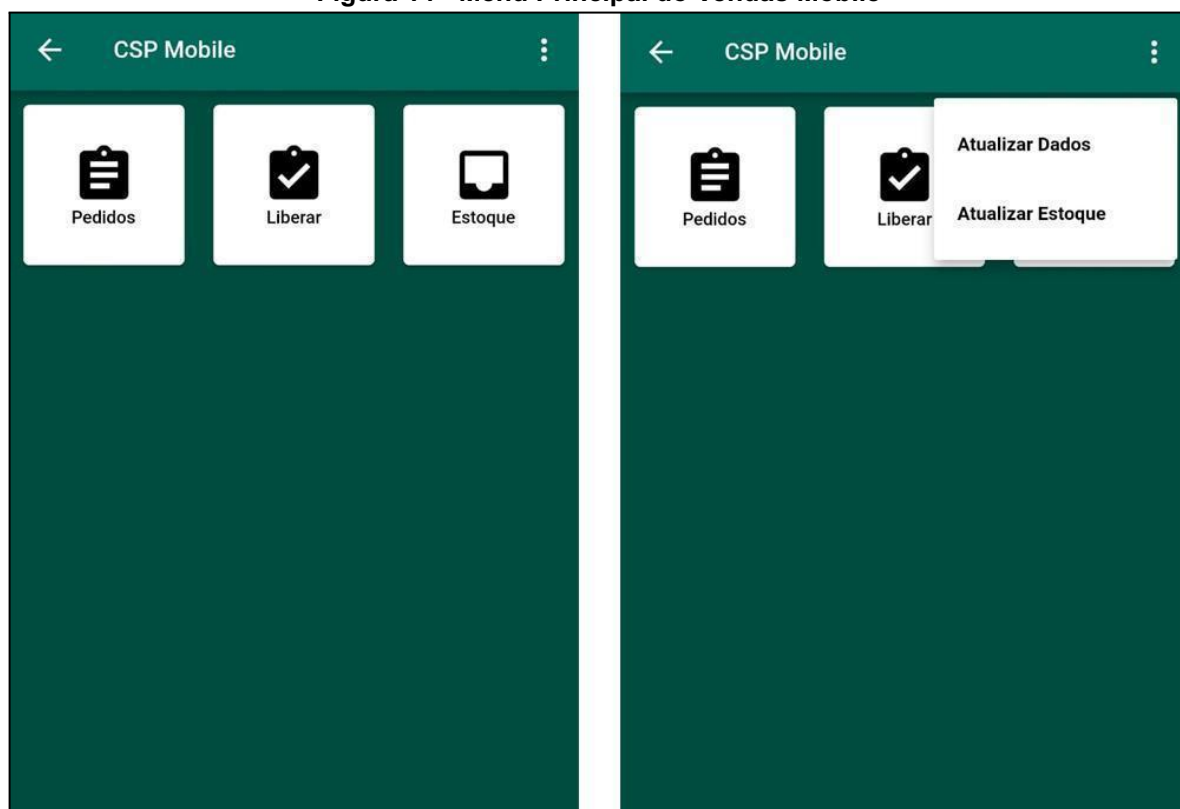
O menu principal da aplicação, no qual é possível acessar os seus três principais módulos que são:

- Pedidos: consultar os pedidos lançados e lançar novos pedidos no sistema;

- Liberar: realizar liberação por alçada;
- Estoque: verificar o preço do produto e sua disponibilidade no sistema ERP.

A Figura 14 apresenta a interface do menu principal da aplicação, no qual é possível acessar os seus três principais módulos.

Figura 14 - Menu Principal do Vendas Mobile



Fonte: Autoria própria (2022)

No canto superior direito, encontra-se o ícone que dá acesso a duas funcionalidades, sendo elas:

- Atualizar dados: clientes e culturas são atualizadas na aplicação;
- Atualizar estoque: estoque e tabela de preço do sistema são atualizados na aplicação.

No código apresentado na Figura 15 é possível observar que, após a primeira tentativa de login, um endereço é salvo como principal para uso no restante da aplicação. Esse endereço é utilizado, por exemplo, para a atualização do estoque.

Figura 15 - Requisição para atualização de dados do estoque, lado cliente

```

Future getEstoque(String url) async {
  Map data;
  try {
    (1) http.Response response = await http.get(url + 'read_estoque.php');
    if (response.statusCode == 200) {
      data = jsonDecode(response.body);
      DatabaseHelper dbHelper = DatabaseHelper.instance;
      dbHelper.queryRowCount('estoque').then((result) {
        (2) if (result > 0) {
          dbHelper.delete('estoque').then((aux) {
            data.forEach((key, values) {
              for (int i = 0; i < values.length; i++)
                dbHelper.insert("estoque", values[i]);
            });
          });
        } else {
          data.forEach((key, values) {
            for (int i = 0; i < values.length; i++) {
              dbHelper.insert("estoque", values[i]);
            }
          });
        }
      });
      return data['estoque']; (3)
    } else {
      throw Exception();
    }
  } on Exception catch (error) {
    return error; (4)
  }
}

```

Fonte: Autoria própria (2022)

Na primeira parte do código (1) é possível verificar a tentativa de conseguir uma respostas no serviço do *web service*, que ao retornar os dados a armazena em uma variável local.

Em seguida, é verificado se já existem dados de estoque armazenados (2), em caso positivo todos os dados são excluídos e os dados da requisição são armazenados. Em caso negativo os dados são apenas armazenados.

Na terceira parte do código são retornados os dados de estoque (3), mostrando que a operação foi bem sucedida, caso ocorra alguma exceção na resposta do serviço, o código retorna o erro (4).

A estrutura utilizada para armazenar os dados é muito parecida com a de um banco de dados relacional convencional. A Figura 16 representa a criação da tabela que armazena os dados do estoque no Vendas Mobile.

Figura 16 - Estrutura interna para armazenamento de dados do estoque

```
await db.execute(`
CREATE TABLE estoque (
  vestoqueempresaid INTEGER,
  vestoqueprodutoid INTEGER,
  vestoquecontrolepeid INTEGER,
  vestoqueinventarioloteid INTEGER,
  vestoquesaldo INTEGER,
  vestoquelotenome TEXT,
  vestoquecontrolepenome TEXT,
  vestoqueprodutonome TEXT
)`);
```

Fonte: Autoria própria (2022)

A Figura 17 mostra parte da *web service*, no qual os retornos das operações de busca realizadas no banco de dados (1), é possível verificar que os nomes das chaves da lista retornada, no caso como *JSON*, são os mesmos armazenadas no aplicativo, justamente para facilitar o processo de identificação dos dados durante o desenvolvimento.

Figura 17 - Exemplo de requisição para atualização de dados, lado servidor

```

<?php
header('Access-Control-Allow-Origin: *');
header('Content-Type: application/json');
include_once '../config/Database.php';
include_once '../models/post_estoque.php';

$databse = new Database();
$db = $databse->connect();
$post = new PostEstoque($db);
$result = $post->read();
$num = $result->rowCount();

if($num > 0){
(1) $posts_arr = array();
    $posts_arr['estoque'] = array();
    while($row = $result->fetch(PDO::FETCH_ASSOC)){
        extract($row);
        $post_item = array(
            'vestoquesaldo' => $vestoquesaldo,
            'vestoqueinventarioloteid' => $vestoqueinventarioloteid,
            'vestoquelotenome' => $vestoquelotenome,
            'vestoquecontrolepeid' => $vestoquecontrolepeid,
            'vestoquecontrolepenome' => $vestoquecontrolepenome,
            'vestoqueprodutoid' => $vestoqueprodutoid,
            'vestoqueprodutonome' => $vestoqueprodutonome,
            'vestoqueempresaid' => $vestoqueempresaid
        );
        array_push($posts_arr['estoque'], $post_item);
    }
    echo json_encode($posts_arr);
} else {
(2) echo json_encode(
    array('message' => 'No post found')
);
}
?>

```

Fonte: Autoria própria (2022)

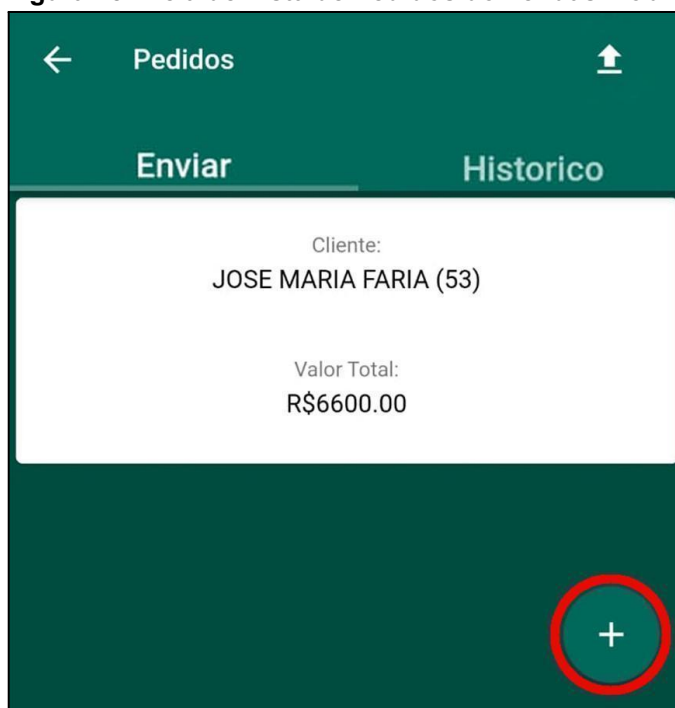
É possível verificar também que, em caso da consulta não retornar nenhum resultado, uma mensagem de erro é retornada no lugar dos dados (2).

- **Sprint 4** - Desenvolvimento da Tela de Lançamento de Pedidos e Produtos e funções para salvar dados internamente:

No desenvolvimento do processo de lançamento dos pedidos, foi utilizado como referência a tela e a modelagem do banco do sistema ERP, contendo assim todos os campos necessários para a utilização do pedido no sistema *web*.

Na tela de pedidos, demonstrada na Figura 18, encontram-se todos os pedidos lançados pelo funcionário, podendo esse realizar o gerenciamento dos pedidos nas ações de adicionar, excluir e editar, mas apenas os pedidos que ainda não foram enviados ao sistema são listados na tela Enviar. Só é possível adicionar pedidos na tela Enviar, clicando no ícone localizado no canto inferior direito. É possível visualizar os pedidos que já foram lançados no sistema acessando a tela Histórico.

Figura 18 - Tela de Lista de Pedidos do Vendas Mobile



Fonte: Autoria própria (2022)

Para realizar o Lançamento de Pedidos, na aba Dados (Figura 19), é necessário realizar o preenchimento de três principais campos na tela de inclusão, sendo eles:

- Cliente, que ao clicar carrega uma tela de busca de clientes e ao selecionar o cliente é retornado a tela o nome e o logradouro;
- Previsão de entrega, usado para controle de compras e previsão de saída do produto do estoque;
- Vencimento, utilizado para cálculo do preço do produto.

Os demais campos são ou preenchidos automaticamente, como Logradouro, Valor Total e Quantidade, ou são opcionais, como Observação.

Para finalizar o pedido, é necessário clicar no botão localizado no canto superior direito, a aplicação verifica os campos necessários para validação do pedido.

Caso falte algum campo a ser preenchido, será destacado em vermelho ou sinaliza a falta de produtos no pedido.

Figura 19 - Tela de Lançamento de Pedido do Vendas Mobile

The image displays two side-by-side screenshots of the 'Criar Pedido' (Create Order) screen in a mobile application. The left screenshot shows the 'Dados' (Data) tab, and the right screenshot shows the 'Produtos' (Products) tab.

Left Screenshot (Dados Tab):

- Header: Criar Pedido (with a checkmark icon circled in red)
- Client: JOSE MARIA FARIA (53)
- Logradouro: RUA DESEMBARGADOR LAURO LOPES
- Previsão de entrega: 30/06/2022
- Vencimento: 31/07/2022
- Observação: teste
- Valor Total: 6.600,00
- Quantidade Total: 12

Right Screenshot (Produtos Tab):

- Header: Criar Pedido (with a checkmark icon)
- Product 1: ROUNDUP ORIGINAL MAIS - FR 1 LT

Sistema:	Venda:	QTD:	Total:
R\$ 94.88	R\$ 100.00	6	R\$600.00
- Product 2: +AERIS PROTECT - BD 20L

Sistema:	Venda:	QTD:	Total:
R\$ 803.59	R\$ 1000.00	6	R\$6000.00
- Bottom right: A green circular button with a white plus sign (+).

Fonte: Autoria própria (2022)

Cabe ressaltar que, só será possível realizar um pedido, quando houver pelo menos um produto na tela Produtos. Nessa tela também é possível realizar o gerenciamento de produtos do pedido.

No processo de validação dos campos da tela de Pedido primeiramente os campos são validados e verifica-se a existência de produtos, na sequência os dados dos campos são armazenados na variável pedido, que está formatada com os nomes iguais aos do banco de dados do sistema *web*, facilitando assim a identificação das mesmas.

Parte do código do processo de validação dos campos da tela de Pedido é mostrada na Figura 20.

Figura 20 - Validação de dados do Lançamento de Pedido do Aplicativo

```

_confirmaFormulario(BuildContext context) {
  final form = _formKey.currentState;
  if (form.validate() && listaProdutosAdicionados.length > 0) {
    form.save();
    pedido.prepedidousuarioid = usuario.usuarioid;
    pedido.usuarionome = usuario.usuarionome;
    pedido.prepedidovendedorid = usuario.usuariovendedorid;
    pedido.prepedidoempresaid = usuario.empresaid;
    pedido.prepedidoprodutolinhas = listaProdutosAdicionados.length;
    pedido.gtabprecoid = listaProdutosAdicionados[0].tabprecoid;
    pedido.prepedidoemissao = DateTime.now().toString();
    pedido.prepedidostatus = 'AGUARDANDO FATURAMENTO';
    pedido.prepedidoautorizadovenda = true;
    pedido.prepedidoenviado = 0;
    for (int i = 0; i < listaProdutosAdicionados.length; i++) {
      if (listaProdutosAdicionados[i].prepedidoprodutoautorizado == false) {
        pedido.prepedidoautorizadovenda = false;
        pedido.prepedidostatus = 'AGUARDANDO AUTORIZAÇÃO';
      }
    }
  }
}

```

Fonte: Autoria própria (2022)

A Figura 21 mostra a continuação da função “_confirmaFormulario” onde o pedido e os produtos são armazenados internamente no Vendas Mobile. Caso a validação dos campos falhe, é verificado quais campos não foram preenchidos e uma mensagem de erro é retornada para auxiliar o usuário.

Figura 21 - Armazenamento do pedido internamente e mensagens de erro

```

pedido.armazenaPrePedido(pedido);
for (int i = 0; i < listaProdutosAdicionados.length; i++) {
  listaProdutosAdicionados[i].prepedidoprodutoid = i + 1;
  listaProdutosAdicionados[i].armazenaProduto(listaProdutosAdicionados[i]);
}
Navigator.pop(context, true);
} else {
  if (listaProdutosAdicionados.length <= 0 && !form.validate())
    helper.showSnackBar("Preencha os campos e adicione produtos antes de confirmar", _scaffoldKey);
  else if (listaProdutosAdicionados.length <= 0 && form.validate())
    helper.showSnackBar("Inclua produtos no pedido", _scaffoldKey);
  if (listaProdutosAdicionados.length > 0 && !form.validate())
    helper.showSnackBar("Preencha todos os campos antes de confirmar", _scaffoldKey);
}
}

```

Fonte: Autoria própria (2022)

O gerenciamento dos produtos do pedido é realizado por meio da tela de Produtos (Figura 22). O botão do canto inferior direito da tela abre a tela de inclusão de produtos no pedido e nela será exigido que o usuário preencha os campos:

- Produto: Ao clicar nessa opção, o aplicativo abrirá uma tela de seleção de produtos.

- Valor da Venda: Qualquer valor pode ser informado neste campo, mas caso o valor da venda esteja abaixo do valor do sistema, automaticamente o pedido fica com status "autorizado" e não será necessária a liberação.
- Quantidade: Nesta opção deve ser informado a quantidade do produto.
- Cultura: campo inicialmente preenchido com "selecione".

Tendo os campos obrigatórios preenchidos será possível finalizar a inclusão do produto clicando no ícone localizado no canto superior direito, como ilustrado na Figura 22.

Figura 22 - Tela de Inclusão de Produto no Pedido



A imagem mostra a interface de usuário para a inclusão de um produto em um pedido. O formulário é composto por vários campos empilhados verticalmente:

- Um campo de texto para "Produto:" com um ícone de lupa à direita.
- Um campo de texto para "Valor do Sistema:" com o valor "0,00" exibido.
- Um campo de texto para "Valor da Venda:" com o valor "0,00" exibido.
- Um campo de texto para "Quantidade:".
- Um campo de texto para "Total:" com o valor "0,00" exibido.
- Um campo de texto para "Autirizado:".
- Um campo de lista suspensa com o texto "SELECIONE" e uma seta para baixo.

No canto superior direito da tela, há um ícone de checkmark verde circulado em vermelho, indicando a opção de salvar ou finalizar a inclusão.

Fonte: Autoria própria (2022)

Os cálculos dos preços dos produtos foram validados com o diretor comercial da empresa, mantendo a integridade com os valores do sistema ERP.

Na Figura 23, é possível verificar a principal função da inclusão de produtos. Após a busca do produto desejado, a função verifica a quantidade de dias entre a data do dia do lançamento e a data de vencimento do pedido. Após verificar a quantidade de dias, é realizado o cálculo de juros compostos, levando em

consideração a taxa e a margem aplicada ao produto. Esses dados provêm da tabela de preço.

Figura 23 - Parte do código de busca e cálculo do valor do produto

```
GestureDetector(
  onTap: () async {
    if (tela == 'E' || tela == 'A') {
      produto = await Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => ListaProdutos(usuario: usuario),
        ), // MaterialPageRoute
      );
    }
    if (produto != null) {
      List<String> dataList = produto.tabprecodata.split('-');
      var iAno = int.parse(dataList[0]);
      var iMes = int.parse(dataList[1]);
      var iDia = int.parse(dataList[2].split(' ')[0]);
      var dataInicio = DateTime(iAno, iMes, iDia);
      List<String> venDataList = dataVencimento.split('-');
      var vAno = int.parse(venDataList[0]);
      var vMes = int.parse(venDataList[1]);
      var vDia = int.parse(venDataList[2]);
      var dataFim = DateTime(vAno, vMes, vDia);
      num dias = int.parse(dataFim.difference(dataInicio).inDays.toString());
      num taxaDia = produto.tabprecoprodutotaxa / 30 / 100;
      _controllerValorSistema.text = (produto.tabprecoprodutogerencial * pow(1.0 + taxaDia, dias)).toStringAsFixed(2);
      _controllerProduto.text = produto.tabprecoprodutoprodutonome;
    }
  }
)
```

Fonte: Autoria própria (2022)

Após o cálculo do valor de venda do produto, eles são salvos na variável do produto, que será retornado à Tela de Criação do Pedido, e em variáveis para demonstração em tela, para confirmação dos valores durante a inclusão do produto.

- **Sprint 5** - Desenvolvimento do sistema de envio de pedidos:

A modelagem do banco do sistema ERP serviu de referência para o desenvolvimento dessa *sprint*. Além dos dados padrões existentes, tais como: “prepedidoid”, “prepedidoemissao”, “clienteid”, foi também utilizado o campo já existente no banco chamado “prepedidodatasincro” para identificar que o envio foi feito pelo Vendas Mobile, contendo a data do envio.

Apenas na tela Enviar é possível lançar os pedidos no sistema, clicando no botão localizado no canto superior direito, como demonstra a Figura 24.

Figura 24 - Tela de Lista de Pedidos do Vendas Mobile



Fonte: Autoria própria (2022)

No momento do envio dos pedidos, esses não são enviados em lote e sim de forma sequencial em fila, tirando proveito do retorno do código identificador que o banco de dados provê após a inclusão, no qual é utilizado para vincular os produtos. Em seguida seus produtos são enviados junto com o identificador do pedido.

Na primeira parte, os dados do pedido são transformados em JSON, possibilitando o envio ao *web service*. Na segunda parte, caso o pedido já tenha sido enviado para o sistema *web*, a variável booleana “duplicou” será verdadeira, impedindo o envio do pedido. Essa verificação é feita anteriormente no código, consultado os pedidos armazenados no aplicativo.

A Figura 25 apresenta trecho do código onde ocorre o processo de envio dos pedidos pendentes.

Figura 25 - Parte do processo de envio de pedidos pendentes

```

try {
    jsonString =
    (1)  '{"prepedidoemissao":"'${prepedidos[i].prepedidoemissao}',"prepedidopreventrega":"'${prepedidos[i].pre
        "prepedidoavencimento":"'${prepedidos[i].prepedidoavencimento}',"prepedidoautorizadovenda":"'${prepedid
        "prepedidoprodutolinhas":"'${prepedidos[i].prepedidoprodutolinhas}',"pessoaid":"'${prepedidos[i].pess
        "prepedidoempresaid":"'${prepedidos[i].prepedidoempresaid}',"prepedidoobservacao":"'${prepedidos[i].p
        "prepedidostatus":"'${prepedidos[i].prepedidostatus}',"prepedidonumero":"'${prepedidos[i].prepedidonu
        "prepedidousuarioid":"'${prepedidos[i].prepedidousuarioid}',"prepedidovendedorid":"'${prepedidos[i].p

    if (!duplicou) { (2)
        response = await http.post(url + 'create_prepedido.php',
    (3)  headers: headers, body: jsonString);
        duplicou = true;
    }
    if (response.statusCode == 200) {
        dados = jsonDecode(response.body);

        if (dados["id"][0]['prepedidoid'] is int) {
    (4)  prepedidos[i].prepedidoenviado = dados["id"][0]['prepedidoid'];
        prepedidos[i].prepedidonumero = dados["id"][0]['prepedidoid'];
        } else {
            prepedidos[i].prepedidoenviado = int.parse(dados["id"][0]['prepedidoid']);
            prepedidos[i].prepedidonumero = int.parse(dados["id"][0]['prepedidoid']);
        }
        try {
    (5)  jsonString =
            '{"contasreceberdescricao":"'Parcela(s)Pré-Pedido${prepedidos[i].prepedidonumero}',"contasreceber
            "contasreceberdataavencimento":"'${prepedidos[i].prepedidoavencimento}',"contasrecebervalor":"'${pr
            "contasreceberprepedidoid":"'${dados["id"][0]['prepedidoid']}',"empresaid":"'${prepedidos[i].prep

            response = await http.post(url + 'create_titulo_pedido.php',headers: headers, body: jsonString);
        } on Exception catch (error) {}
        return error;
    }
}

```

Fonte: Autoria própria (2022)

Na terceira parte está destacado o envio de cabeçalho do pedido sendo realizado e na quarta parte verifica o armazenamento no produto do pedido o identificador único do pedido provido pelo retorno do *web service*. Esse identificador é criado pelo banco de dados do sistema *web* e por essa razão é necessário enviar o produto depois do sucesso da inclusão do cabeçalho do pedido.

O envio do produto é realizado da mesma maneira que o pedido, mostrado na quinta parte. Primeiro o *JSON* é criado e depois o envio é realizado. Em caso de erro no envio, tanto do pedido quanto dos produtos, uma mensagem de erro é retornada pelo código.

- **Sprint 6** - Desenvolvimento de Tela de Demonstração de Produtos em Estoque e Quantidades no Lote por Filial:

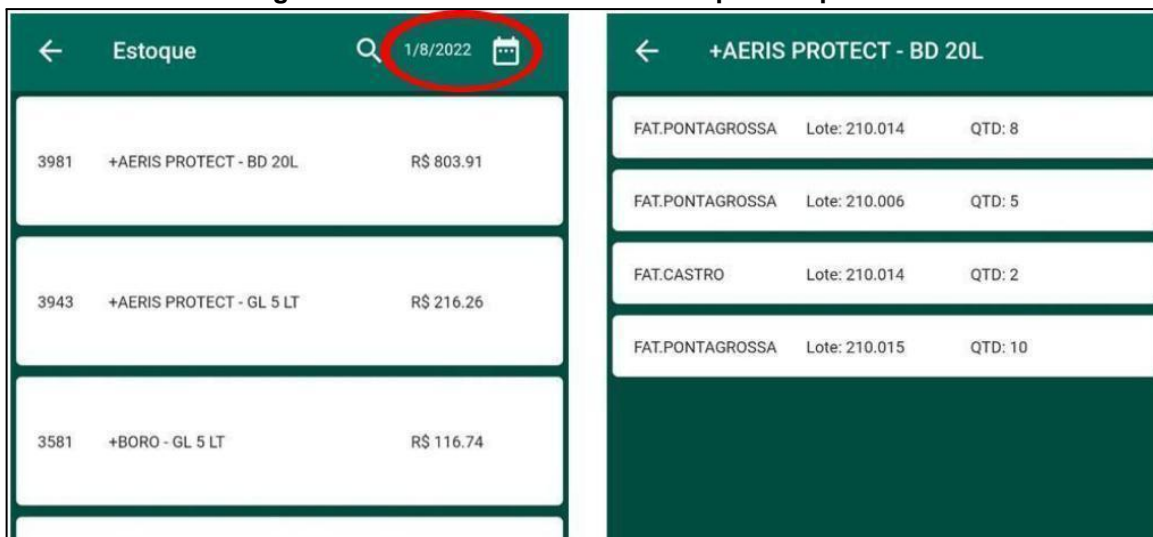
No desenvolvimento da tela de Consulta do Estoque, foi utilizado um processo diferente do sistema *web*, nele foi implantada a funcionalidade de consulta de todas as filiais, sendo esse um dos requisitos solicitados pelos vendedores e gestores. Pelo fato do lançamento do pedido não exigir a existência do produto na filial, era necessário que fosse possível existir a visibilidade do estoque da empresa como um todo, possibilitando assim a negociação de tempo de entrega do produto com o cliente e a transferência entre filiais entre os gestores.

Ao selecionar a opção Estoque no menu principal é possível ter acesso a Tela de Consulta de Estoque, que apresenta os valores dos produtos disponíveis.

No entanto, cabe ressaltar que as atualizações dos dados do estoque não ocorrem em tempo real, dessa forma, os usuários do aplicativo Venda Mobile precisam realizar a ação de atualização dos dados constantemente para manter a consistência entre o aplicativo e o sistema ERP.

A Figura 26 demonstra a tela de Consulta de Estoque no aplicativo, no qual é possível visualizar apenas os produtos presentes na tabela de preço e caso existam unidades do produto no estoque mas não esteja precificado na tabela, o produto não será listado.

Figura 26 - Tela de Consulta de Estoque no Aplicativo



ID	Nome do Produto	Preço
3981	+AERIS PROTECT - BD 20L	R\$ 803,91
3943	+AERIS PROTECT - GL 5 LT	R\$ 216,26
3581	+BORO - GL 5 LT	R\$ 116,74

Fornecedor	Lote	QTD
FAT.PONTAGROSSA	210.014	8
FAT.PONTAGROSSA	210.006	5
FAT.CASTRO	210.014	2
FAT.PONTAGROSSA	210.015	10

Fonte: Autoria própria (2022)

Além da consulta de produtos também é possível consultar o valor do produto da data que for necessário, informando a data desejada no ícone de calendário no canto superior direito. O cálculo do valor do produto é o mesmo encontrado na criação do pedido de venda, no qual o valor é baseado no custo

gerencial, a margem e a taxa de juros informado pela tabela de preço e também na data do prazo de vencimento do pedido. Informando a data no ícone do canto superior direito o cálculo é realizado em todos os produtos da lista.

Ao selecionar um produto, será possível verificar em quais filiais o produto está disponível, qual lote e a quantidade.

- **Sprint 7** - Desenvolvimento de tela e processo de liberação de pedidos com desconto e desenvolvimento de tela e processo de liberação de faturamento:

Para o desenvolvimento das telas de Liberação de Alçada, foram utilizadas as informações referentes às necessidades dos usuários relatadas nas reuniões, tendo como base o processo existente no sistema ERP, uma vez que é necessário que os dados sejam consistentes entre ambas as aplicações.

Nela foi desenvolvida, assim como no sistema *web*, a validação das autonomias de liberação de desconto nos pedidos e controle de acesso de liberação de cada tipo de bloqueio restrito à cada setor.

Selecionando a opção Liberar no menu principal, o usuário é encaminhado a tela que contém os tipos de liberações possíveis de serem realizadas. A Figura 27 mostra todas as opções da tela Liberações.

Figura 27 - Tela de Liberações do Vendas Mobile



Fonte: Autoria própria (2022)

As opções disponíveis são apresentadas conforme o perfil do usuário e suas permissões. Os perfis e permissões são:

- **Vendedor:** Com permissão apenas para liberar pedidos lançados por ele, com alçada de quatro por cento de desconto nos produtos.

- Gerente Comercial: Com permissão apenas para liberar pedidos lançados por ele e vendedores de sua filial, com alçada de dez por cento de desconto nos produtos.
- Financeiro: Com permissão apenas para liberar o financeiro.
- Compras: Com permissão apenas para autorizar solicitação de compra.
- Diretor Comercial: Com permissão para liberar pedidos com alçada de cem por cento, liberar margem, liberar devolução e liberar bonificação.
- Admin: Com permissão para acessar todas as telas de liberação.

Todas as telas de liberação seguem o mesmo padrão (Figura 28), tendo inicialmente uma lista de pedidos para escolher.

Figura 28 - Lista de Liberação de Pedidos de Venda

← Liberar Pedidos		
Pedido: 64759		
Cliente: OLIDIO APARECIDO DE OLIVEIRA E	Quantidade: 60	Total: R\$ 42420.00
Pedido: 67356		
Cliente: FRANCISCO SCHEBELSKI	Quantidade: 5	Total: R\$ 2150.00
Pedido: 64561		
Cliente: ANGELO SCHEBELSKI	Quantidade: 44	Total: R\$ 18400.00

Fonte: Autoria própria (2022)

Ao selecionar um pedido, é possível visualizar os dados do pedido e os produtos lançados. A liberação de um pedido é realizada por meio do botão Liberar, no caso da liberação de pedidos existe um limite de desconto vinculado ao cargo do usuário, sendo 4% (quatro por cento) de desconto disponível para o consultor e venda e 8% (oito por cento) disponível para o gerente comercial, valores definidos pelo diretor e aplicados ao perfil do usuário no sistema *web*, caso o usuário não permita o desconto ou outro tipo de liberação, é necessário selecionar o botão

Cancelar. Os produtos marcados em vermelho (Figura 29) são os que causaram o bloqueio.

Figura 29 - Visualização de Pedido para Liberação

Sistema	Venda	QTD	Total
R\$ 707,40	R\$ 707,00	60	R\$42420,00

Fonte: Autoria própria (2022)

Após verificar o tipo de liberação a ser realizada, é montado o JSON a ser enviado à *web service*. São poucos dados a serem enviados por ser apenas uma alteração de dois campos do pedido no banco de dados. A Figura 30 demonstra o processo para liberação.

Figura 30 - Processo de liberação de pedidos

```
Future _autorizaPedido() async {
  String json;
  http.Response response;

  if (tela == "M") {
    try {
      json = '{"pedidooid":"${pedido.pedidooid}","pedidoautorizadousuario2id":"${usuario.usuarioid}"}';

      response = await http.post(url + 'update_pedido_authorized_margem.php', headers: headers, body: json);

      if (response.statusCode != 200) {
        print(response.body);
        throw Exception('Erro ao autorizar pedido!');
      } else {
        json = '{"pedidooid":"${pedido.pedidooid}","pedidojustobservacao":"${_controllerJustificativa.text.toString()}"}';

        response = await http.post(url + 'create_pedidojust.php', headers: headers, body: json);

        Navigator.pop(context,true);
      }
    } on Exception catch (error) {
      return error;
    }
  }
}
```

Fonte: Autoria própria (2022)

Quando a liberação não é realizada com sucesso, uma mensagem de erro é apresentada ao usuário. No caso de liberação com sucesso, outro dado é salvo no banco, sendo esse a justificativa da liberação.

As *Sprints* foram executadas e concluídas, sendo essas uma parte do projeto do aplicativo Vendas Mobile. O acompanhamento do progresso do trabalho foi feito por reuniões denominadas *Daily Scrum Meeting* (SCHWABER; SUTHERLAND, 2020) as quais serão descritas a seguir.

4.1.4 Daily Scrum

A aplicação da *Daily Scrum* foi adaptada do modelo apresentado no capítulo 2.2.1, pois em consideração a limitação de tempo dos membros da equipe *Scrum*, as reuniões foram realizadas semanalmente. O Mestre *Scrum* acompanhou o progresso do time de desenvolvimento e orientou a execução das *sprints* durante as reuniões semanais.

Visto que o Dono do Produto é um dos integrantes do time de desenvolvimento, o alinhamento dos interesses da empresa com relação às funcionalidades do Vendas Mobile se mantiveram fiéis durante todas as *Sprints*. A ordem de execução das *sprints* foram planejadas de acordo com a prioridade de uso das funcionalidades, orientado pelo Dono do Produto.

Durante as reuniões realizadas semanalmente, eram ponderados os resultados do progresso do time de desenvolvimento, e caso alguma *sprint* tenha sido finalizada durante a semana, era planejada a aplicação dos testes da funcionalidade desenvolvida e a execução da *sprint* seguinte. Também era realizada a validação dos testes, partindo para ajuste em caso de falha em alguma funcionalidade.

4.2 AVALIAÇÃO DO APLICATIVO VENDAS MOBILE

A segunda etapa do projeto teve como objetivo avaliar a qualidade do aplicativo Vendas Mobile. Esta etapa foi dividida em três atividades: criação, aplicação e análise dos dados.

A criação consiste na elaboração do formulário de avaliação e para isso teve como base a norma ISO/IEC 9126, no qual foram utilizadas as características de: funcionalidade, confiabilidade, usabilidade e eficiência. O formulário foi composto por 17 questões objetivas (ANEXO A).

A característica de funcionalidade teve como objetivo avaliar as principais funções do aplicativo e verificar se as mesmas atendem as necessidades do usuário, como por exemplo o lançamento e liberação de pedidos. No que se diz respeito a funcionalidade foi questionado ao funcionário se o aplicativo aponta quais são os dados inseridos incorretamente, se as funções de lançamento e liberação de pedidos funcionam corretamente e se os dados apresentados são suficientes na consulta de estoque ou para liberação de um pedido.

A característica de confiabilidade avalia a capacidade do *software* de manter um processo em execução em caso de ocorrência de erros. Tem-se como exemplo: a capacidade de finalizar o processo após falha na execução da aplicação durante a operação.

A característica de usabilidade teve como objetivo, avaliar se a aplicação possui fácil entendimento na execução de suas funcionalidades, como por exemplo verificar se existe, por parte dos usuários, alguma dificuldade para executar um lançamento de pedidos, se o uso das funcionalidades são intuitivas ou se existia a necessidade de explicação detalhada.

Foi questionado ao funcionário qual o grau de dificuldade de cada uma das funcionalidades e também foi disponibilizado um espaço para argumentar quais foram as dificuldades encontradas.

Por fim, a característica de eficiência avaliou o desempenho do aplicativo na execução de suas funções, tal como, o carregamento e envio de dados a partir do sistema *web*, tendo em vista a complexidade da ação executada.

A seguir no Quadro 5 é apresentado o número de cada questão e a quantidade total delas separadas por características de qualidade.

Quadro 5 - Característica, quantidade de perguntas e número da questão

Característica da Qualidade	quantidade de perguntas	Nº da questão
Funcionalidade	11	1, 2, 3, 4, 5, 6, 7, 8, 9,10 e 11
Confiabilidade	1	12
Usabilidade	4	13, 14, 15 e 16
Eficiência	1	17

Fonte: Autoria própria (2022)

Conforme apresentado no Quadro 5, foram formuladas ao todo 17 (dezessete) perguntas referentes às características citadas, sendo elas 11 (onze) sobre funcionalidade, 1 (uma) sobre confiabilidade, 4 (quatro) sobre usabilidade e 1 (uma) sobre eficiência.

Também foram desenvolvidas 4 (quatro) perguntas não enumeradas com o objetivo de identificar o perfil do usuário. As perguntas solicitavam que os usuários informassem o nome, idade, cargo e filial em qual trabalha.

A avaliação do formulário ocorreu no dia 20 (vinte) de setembro de 2022 e foi realizada por dois funcionários da filial de Castro - PR. A escolha desses avaliadores se deu pelo motivo de trabalharem na mesma filial de um dos integrantes do time de desenvolvimento e por não saírem a campo.

A avaliação durou cerca de 30 (trinta) minutos, onde ambos os funcionários realizaram a leitura do questionário e fizeram uma análise sobre as questões. Após a análise algumas questões foram reformuladas para melhor entendimento de todos os participantes.

No dia primeiro de Outubro de 2022 foi realizada uma reunião com os gerentes de cada filial para apresentar a pesquisa de satisfação. Nessa reunião foram sanadas as dúvidas sobre algumas das perguntas e prazo máximo de preenchimento. Também foi apresentada as datas para preenchimento do formulário, que deveriam ocorrer entre o dia 3 (três) até o dia 9 (nove) de outubro, totalizando sete dias.

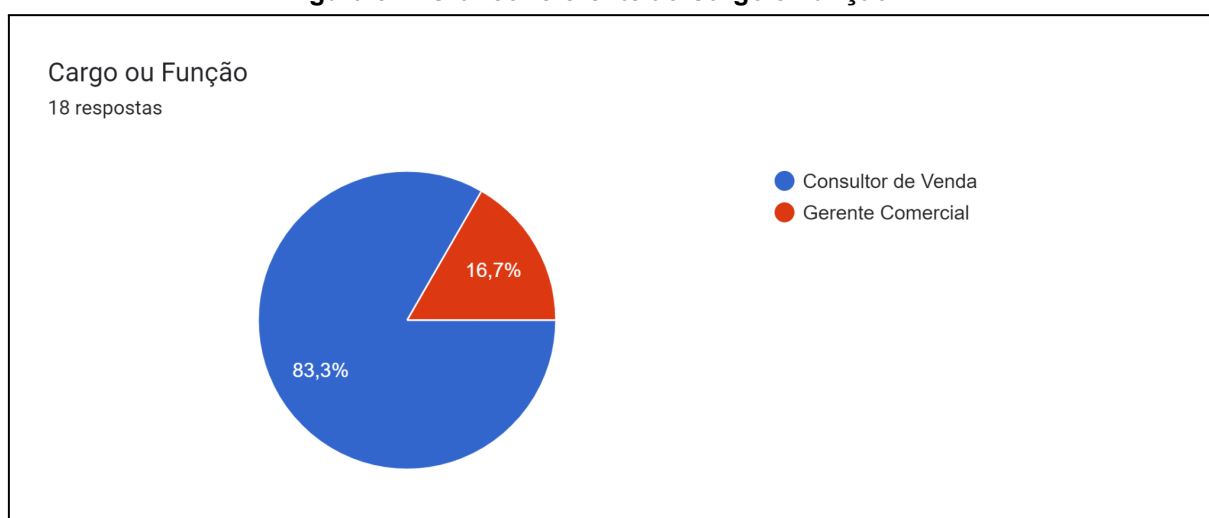
A aplicação do formulário ocorreu no dia 3 (três) de outubro de 2022 e foi aplicada pelos gerentes de cada filial para seus respectivos funcionários. A quantidade total de entrevistados foi de 25 (vinte e cinco) e o total de pessoas que finalizaram a pesquisa foi de 18 (dezoito). Os gráficos que ilustram os resultados da aplicação encontram-se no ANEXO B.

4.2.1 Análise dos Resultados

A análise dos resultados foi realizada por meio da análise dos dados quantitativos obtidos das 17 (dezesete) perguntas do formulário, bem como da interpretação das observações descritas pelos entrevistados e do conhecimento prévio dos desenvolvedores, no que se refere às funcionalidades da aplicação, restrições e necessidades técnicas.

Conforme já informado, o formulário foi aplicado para 25 (vinte e cinco) usuários, sendo eles 5 (cinco) gerentes e 20 (vinte) vendedores, todos com acesso às mesmas funcionalidades. No total, 18 (dezoito) usuários responderam ao questionário o que corresponde a 72% (setenta e dois por cento) da amostra. Dessa amostra, 3 (três) funcionários obtêm cargo de gerente comercial e 15 (quinze) são consultores de vendas, como mostra a Figura 31.

Figura 31 - Gráfico referente ao Cargo e Função



Fonte: Autoria própria (2022)

A seguir serão apresentadas as análises e interpretação dos dados, sendo esses classificados de acordo com as características de funcionalidade, confiabilidade, usabilidade e eficiência.

4.2.1.1 Avaliação da característica funcionalidade

Conforme já relatado, uma das características da ISO/IEC 9126 diz respeito à funcionalidade que será o tema abordado neste tópico. Para essa característica, os usuários tiveram que responder a 11 (onze) perguntas e para cada uma delas foram definidas diferentes alternativas. O Quadro 6 apresenta as perguntas de 1 (um) a 5 (cinco), com suas respectivas alternativas e a quantidade de usuários que a escolheram.

Quadro 6 - Perguntas referentes a característica de funcionalidade. Parte 1

Nº	Pergunta	Resposta	Quantidade
1	Realiza o lançamento de pedidos corretamente	Sim	16
		Não	2
2	Realiza a liberação de pedido corretamente	Sim	13
		Não	3
		Às Vezes	2
3	Dispõe de todas as funções necessárias para o lançamento de pedidos	Sim	18
		Nenhuma	0
		Algumas	0
4	Número de vezes por dia que atualiza os dados do aplicativo	Apenas uma	6
		Duas a cinco	7
		Mais que cinco vezes	5
5	Calcula o valor dos produtos de forma correta	Sim	15
		Nunca	3
		Às Vezes	0

Fonte: Autoria própria (2022).

A pergunta de número 1 (um) que questiona se o aplicativo realiza o lançamento de pedido corretamente, 88,9% (oitenta e oito vírgula nove por cento) dos usuários afirmaram que a funcionalidade executa corretamente. Para os que avaliaram negativamente foi habilitada no formulário a opção para justificar sua

avaliação. Os usuários que utilizaram dessa opção, argumentaram que às vezes o aplicativo não envia o pedido para o sistema.

Quando questionados se o aplicativo realiza a liberação de pedidos corretamente, referente a pergunta 2 (dois), a maioria avaliou positivamente sendo ao total 72,2% (setenta e dois vírgula dois por cento) de aprovação. O restante das avaliações ficaram divididas entre 11,1% (onze vírgula um por cento) com resposta "às vezes" e 16,7% (dezesseis vírgula sete por cento) responderam que "não". Na justificativa os argumentos foram de que o aplicativo não permite a liberação de pedidos.

Essa funcionalidade do Vendas Mobile está diretamente ligada ao cargo do funcionário, sendo assim os consultores possuem autonomia limitada e só podem liberar pedidos com no máximo 4% (quatro por cento) de desconto. Quando o desconto ultrapassa esse valor, somente os gestores podem fazer a liberação. Pode-se verificar que os funcionários, que não aprovaram a funcionalidade, são novos na empresa e possivelmente houve falta de treinamento para explicar essa condição.

Na pergunta de número 3 (três) referente a disponibilidade de funcionalidades necessárias para lançamento de pedidos, as respostas foram unânimes, todos avaliaram positivamente a funcionalidade, o que significa que o aplicativo tem todas as funções necessárias para poder efetuar um pedido. Porém um usuário deixou uma sugestão de implementar a visualização do valor do pedido a prazo.

As respostas da pergunta número 4 (quatro) referente a quantas vezes o usuário atualiza os dados do aplicativo durante o dia, 39,9% (trinta e nove vírgula nove por cento) responderam que atualizam de duas a cinco vezes ao dia, 33,3% (trinta e três vírgula três por cento) dizem que apenas uma vez ao dia os dados são atualizados e 27,8% (vinte e sete vírgula oito por cento) informam que têm o hábito de atualizar mais de cinco vezes. Essa informação indica que nos casos onde o usuário não tem o hábito de realizar a atualização pode fazer com que a integridade dos dados fique comprometida, ou seja, podem existir cenários em que as informações são divergentes com o sistema *web*. Essa informação é muito

relevante, pois as funcionalidades que serão avaliadas a seguir dependem que os dados estejam atualizados.

A questão referente ao cálculo correto dos valores dos produtos, pergunta 5 (cinco), obteve 83,3% (oitenta e três vírgula três por cento) de aprovação e 16,7% (dezesseis vírgula sete por cento) dos usuários relataram que às vezes os valores do produto são apresentados incorretamente, porém não houve justificativa das respostas. Dos usuários que responderam “Às Vezes”, identificou-se que todos eles fazem parte do grupo que atualiza os dados apenas uma vez ao dia. Essa funcionalidade depende de que os dados estejam atualizados e essa pode ser a causa do problema.

A seguir o Quadro 7 apresenta a parte 2 (dois) das perguntas das características de funcionalidade.

Quadro 7 - Perguntas referentes a característica de funcionalidade. Parte 2

Nº	Pergunta	Resposta	Quantidade
6	Informa corretamente os saldos do estoque	Sim	12
		Nunca	2
		Às Vezes	4
7	Informa todos os dados dos pedidos a serem liberados	Sim	12
		Dados principais são apresentados	3
		Sim, mas faltam dados	2
		Os pedidos não são apresentados	1
8	Pedido criado pelo aplicativo é enviado corretamente para o sistema <i>web</i> ?	Sim	14
		Nunca	4
		Às Vezes	0
9	Informa os campos obrigatórios que não foram preenchidos ao tentar finalizar o pedido	Sempre	15
		Nunca	0
		Às Vezes	3
10	Informa quando o pedido não foi enviado ao sistema <i>web</i>	Sim	10
		Não	8
		Às Vezes	0
11	Informa quando é realizada uma entrada de dados é inválida	Sim	8
		Não	5
		Às Vezes	5

Fonte: Autoria própria (2022).

A aprovação da questão de número 6 (seis) referente a demonstração correta dos saldos do estoque foi de 66,7% (sessenta e seis vírgula sete por cento). Somando as avaliações que responderam negativamente e as que responderam que às vezes, obtém-se o total de 33,3% (trinta e três vírgula três por cento). Nota-se que há uma quantidade significativa de pessoas que já encontraram pelo menos uma vez uma informação incorreta.

Entende-se que nesse último caso, pode estar relacionado ao fato do usuário estar em modo *offline* durante a verificação e consulta dos estoque na base de dados, possibilitando a inconsistência da informação. Outro cenário a considerar, é a possibilidade de um usuário dar baixa no estoque no mesmo momento em que outro usuário está no processo de finalização do pedido, causando novamente inconsistência da informação do estoque.

Referente a pergunta 7 (sete), que questiona se todos os dados são apresentados para a liberação de um pedido, o percentual de aprovação foi de 66,7% (sessenta e seis vírgula sete por cento), 16,7% (dezesesseis vírgula sete por cento) responderam que os principais dados são apresentados, 11,1% (onze vírgula um por cento) responderam que faltam dados e 5,6% (cinco vírgula seis por cento) informaram que os pedidos não são apresentados para a liberação. O total percentual das avaliações que afirmaram que todos os dados e os dados principais são apresentados é de 83,3% (oitenta e três vírgula três por cento) indicando alto grau de aprovação. Dos 16,7% (dezesesseis vírgula sete por cento) dos usuários que afirmaram que faltam dados na funcionalidade, nenhum argumentou sobre quais dados não estavam presentes. Com a ausência desses argumentos impossibilita identificar a causa do problema.

A pergunta referente a integridade dos dados do pedido quando são enviados ao sistema *web*, pergunta 8 (oito), 77,8% (setenta e sete vírgula oito por cento) dos usuários avaliaram positivamente e 22,2% (vinte e dois vírgula dois por cento) afirmaram que às vezes o pedido é criado incorretamente. Acredita-se que as respostas negativas estejam relacionadas a detalhes que passam despercebidos durante o lançamento do pedido, visto que é uma pequena parcela dos entrevistados e que não são todos os pedidos lançados que apresentam incongruência.

A pergunta 9 (nove) questiona se o aplicativo informa os campos obrigatórios que não foram preenchidos, 83,3% (oitenta e três vírgula três por cento) dos usuários aprovaram a funcionalidade, porém 16,7% (dezesesseis vírgula sete por cento) dos usuários afirmaram que em algum momento o aplicativo deixou de informar quais campos são obrigatórios para finalizar o pedido. Acredita-se que o processo de lançamento de pedidos está bem estabelecido, visto que apenas 3 (três) usuários responderam que às vezes o aplicativo informa os campos obrigatórios.

Já na pergunta referente a informação de que o aplicativo não foi capaz de enviar pedidos para o sistema *web*, pergunta 10 (dez), pouco mais da metade dos usuários, 55,6% (cinquenta e cinco vírgula seis por cento) afirmam que o processo funciona corretamente. Nesse caso, não chega a ser a maioria, mas há uma grande rejeição por parte dos usuários, 44,4% (quarenta e quatro vírgula quatro por cento) afirma que o aplicativo não informa quando o pedido não é enviado para o sistema, sendo necessária uma averiguação do aplicativo para identificar as possibilidades que levam a essa ocorrência.

Em relação à pergunta 11 (onze) onde é questionado se o aplicativo informa que algum tipo de dado foi inserido incorretamente, 44,4% (quarenta e quatro vírgula quatro por cento) afirmaram que esse processo sempre ocorre e 55,6% (cinquenta e cinco vírgula seis por cento) dos usuários relataram que às vezes ou nunca viram a mensagem. A avaliação dessa questão tornou-se desnecessária, pois os campos não permitem a inserção de dados diferente do previsto.

Como foi visto neste capítulo, apenas uma das onze avaliações obteve uma quantidade de rejeição significativa, que é o caso da pergunta 10 (dez) onde 44,4% (quarenta e quatro vírgula quatro por cento) dos usuários afirmam que o aplicativo não informa quando o pedido não é enviado para o sistema. Para todas as outras avaliações da característica de funcionalidade, observou-se que o aplicativo possui um grau de aceitação significativo, todas obtendo resultado superior a 55% (cinquenta e cinco por cento) e em vários casos ultrapassando 75% (setenta e cinco por cento), demonstrando que o Venda Mobile atende aos requisitos funcionais definidos para o produto criado. Contudo, as avaliações negativas não serão desconsideradas, existem pontos a serem verificados e melhorados para aumentar a

qualidade do aplicativo. É possível que apenas uma orientação direcionada aos funcionários seja efetiva na resolução desses casos.

4.2.1.2 Avaliação da característica confiabilidade

Para avaliar a característica de confiabilidade, a questão “O aplicativo é capaz de dar continuidade em um processo em andamento em caso de falha ?” foi formulada com 3 (três) alternativas para resposta, sendo elas sim, não e às vezes. Apenas 33,3% (trinta e três vírgula três por cento) dos usuários aprovaram o processo. Nessa pergunta, ficou disponível um campo descritivo, para os usuários explicarem as ocorrências em que o aplicativo falhou. A maioria que usou do artifício, explicou que é necessário refazer todo o processo quando o aplicativo trava e fecha sozinho.

Na avaliação da característica, o aplicativo não obteve resultados satisfatórios, demonstrando que existe a necessidade de efetuar melhorias para que os usuários não sofram com as falhas relatadas.

4.2.1.3 Avaliação da característica usabilidade

Neste capítulo serão apresentadas as perguntas e avaliações da característica de usabilidade. Esta característica tem o objetivo de analisar o esforço necessário do usuário para o aprendizado e utilização do sistema. Foram abordadas questões referentes à facilidade de uso, e a apresentação do sistema e para isso foram formuladas 4 (quatro) perguntas, conforme o Quadro 8, referente às informações do apêndice B.

Quadro 8 - Perguntas referentes a característica de usabilidade

Nº	Pergunta	Sim	Não
13	É fácil lançar pedidos no aplicativo	18	0
14	É fácil de realizar a consulta de estoque	15	3
15	É fácil de realizar a liberação de pedidos	11	7
16	É fácil de navegar entre as funcionalidades do aplicativo	17	1

Fonte: Autoria própria (2022).

Na pergunta 13 (treze) que questiona a facilidade de realizar o lançamento de pedidos, as respostas foram unânimes, todos afirmaram não encontraram dificuldades em lançar pedidos no aplicativo. Com isso, é possível afirmar que o usuário não necessita despender esforços para aprender a utilizar o processo de lançamento de pedidos, a funcionalidade é intuitiva.

A questão referente a facilidade de realizar a consulta de estoque, pergunta 14 (quatorze), 83,3% (oitenta e três vírgula três por cento) avaliaram positivamente, ou seja, os usuários realizaram a consulta de estoque facilmente. Foi verificado que o percentual de usuários que avaliaram negativamente fazem parte do grupo que responderam atualizar os dados do aplicativo de uma a duas vezes por dia, com isso, acredita-se que os usuários com dificuldade podem ter dados desatualizados na consulta de estoque.

Quando questionados sobre a facilidade de realizar a liberação de pedidos com relação à pergunta 15 (quinze), da avaliação da confiabilidade teve 61,1% (sessenta e um vírgula um por cento) de aprovação. Na opção aberta para argumentação da avaliação negativa, todos apontaram o mesmo problema como na questão 2 (dois) da avaliação de funcionalidade, que no caso os consultores não conseguem liberar o pedido. Como foi dito anteriormente, possivelmente os consultores de venda não tiveram o treinamento adequado para utilizar a funcionalidade.

A pergunta 16 (dezesseis) teve o objetivo de identificar se os usuários tinham a facilidade na navegação entre funcionalidades da aplicação. Essa avaliação obteve 94,4% (noventa e quatro vírgula quatro por cento) de aprovação. Apenas um usuário afirmou ter dificuldades na navegação do aplicativo e por se tratar de um caso isolado, os motivos da avaliação negativa pode ser por falta de treinamento ou por uma questão pessoal de adequação à aplicação.

Na avaliação da característica de usabilidade a questão 15 (quinze) obteve 61,1% (sessenta e um vírgula um por cento) de aprovação sendo ela a mais baixa. Para todas as outras, o índice de aprovação foi superior a 80% (oitenta por cento), ou seja, os usuários tiveram facilidade para interagir com a interface e de uma forma eficiente. Existem pontos a serem verificados e melhorados que podem aumentar

ainda mais a aprovação dos usuários melhorando a qualidade da característica de usabilidade.

4.2.1.4 Avaliação da característica eficiência

Na última pergunta do formulário e única referente a avaliação da eficiência, “O aplicativo é rápido na execução das tarefas?” obteve 61,1% (sessenta e um vírgula um por cento) de aprovação. As respostas da questão aberta para argumentação, os usuários informaram que existe instabilidade e demora para carregar informações. Essa ocorrência pode estar relacionada a diversos motivos técnicos, tais como limite de memória disponível no aparelho, tarefas em execução em segundo plano e velocidade da conexão com a internet, sendo que esses fogem ao escopo da aplicação.

Visto que 61,1% (sessenta e um vírgula um por cento) dos usuários avaliaram positivamente a questão e os outros 38,9% (trinta e oito vírgula nove) informaram que “às vezes” o aplicativo é rápido na execução das tarefas, mas os motivos fogem do escopo da aplicação, conclui-se que a característica de eficiência possui um grau aceitável, demonstrando que o aplicativo é eficaz no que se propõe a fazer.

5 CONCLUSÃO

A elaboração deste trabalho resultou em um aplicativo móvel denominado Vendas Mobile, que disponibiliza os mesmos recursos que já existem no sistema *web* da empresa Insumos LTDA. Porém esses recursos não possuíam a responsividade adequada para o uso diário dos funcionários. O objetivo do aplicativo foi agilizar os processos do módulo de vendas, melhorando a acessibilidade dos recursos de lançamento de pedidos com a possibilidade de sincronização dos dados que foram inseridos sem conexão com a internet.

O desenvolvimento do presente trabalho se deu em 2 (duas) etapas, sendo a primeira Desenvolvimento da Aplicação e a segunda Avaliação do Aplicativo Vendas Mobile.

Para a elaboração da primeira etapa, foi necessário realizar um estudo referente ao processo de lançamento de pedidos utilizado pela empresa. Também foi realizada uma consulta com os gestores para verificar se os mesmos tinham alguma sugestão de implementações de novas funcionalidades ou melhorias a serem feitas. Com isso foi possível levantar os requisitos funcionais mínimos necessários para consultar o estoque, lançar e liberar os pedidos pelo aplicativo.

Antes de iniciar o processo de desenvolvimento, foi pesquisado e utilizado os recursos da metodologia ágil de desenvolvimento *Scrum*, com o objetivo de organizar o desenvolvimento do projeto e dar velocidade na entrega das funcionalidades do aplicativo. A partir disso, iniciou-se o desenvolvimento do aplicativo com a utilização da tecnologia *Flutter*, que contribuiu de forma positiva na velocidade do desenvolvimento. Para o desenvolvimento do serviço que se tornaria disponível para a integração dos sistemas, foram utilizadas as ferramentas que já estavam em uso no sistema *web*, tais como o banco de dados PostgreSQL, linguagem de programação PHP e o serviço com arquitetura REST.

A utilização da aplicação requer o cadastro de um identificador único gerado pelo aplicativo que em seguida é vinculado ao registro do usuário no sistema *web*, permitindo assim a utilização do aplicativo. O usuário passa a ter acesso ao lançamento de pedidos com a possibilidade de realizar a inserção de dados de forma *offline* para posteriormente sincronizar com o sistema, dessa forma permitindo

a utilização em campo. Também passa ter acesso à consulta de estoque, podendo verificar a disponibilidade dos produtos de todas as filiais. Por último, o funcionário passa a ter acesso a liberação de pedidos que não ultrapassem o desconto máximo de 4% (quatro por cento) para consultores e 8% (oito por cento) para os gestores.

Para o desenvolvimento da segunda etapa, foi elaborado um formulário baseado nas diretrizes da ISO/IEC 9126 que avaliam a qualidade de um *software*. Com a intenção de verificar a aprovação do aplicativo pela visão dos usuários, foram utilizadas as características de funcionalidade, confiabilidade, usabilidade e eficiência para criar o questionário.

Na avaliação da característica de funcionalidade, dez das onze avaliações obtiveram resultados de aprovação superior a 55% (cinquenta e cinco por cento) e na maioria dos casos ultrapassando 75% (setenta e cinco por cento), demonstrando que o Venda Mobile atende aos requisitos definidos para o produto criado. Existem pontos a serem verificados e melhorados para aumentar a qualidade do aplicativo. Em pequena proporção, existem casos a serem investigados onde os usuários afirmam que as informações não estão corretas.

No que se diz respeito à característica de confiabilidade, os resultados da pesquisa não obteve resultados satisfatórios, concluindo-se que existe a necessidade de efetuar melhorias para que os usuários possam retornar ao ponto em que estavam antes de ocorrer o erro, evitando retrabalhos.

Na avaliação da característica de usabilidade, uma das quatro questões obteve o índice de aprovação superior a 61,1% (sessenta e um por cento) para todas as outras o índice de aprovação foi superior a 80% (oitenta por cento), ou seja, os usuários tiveram facilidade para interagir com a interface e não precisam despende grandes esforços para entender como usar as funcionalidades. Das avaliações negativas, notou-se que os usuários têm dificuldades no entendimento da regra de negócio da empresa para a liberação de pedidos.

Na avaliação da característica de eficiência, quando perguntado aos usuários se o aplicativo é rápido na execução das tarefas, 38,9% (trinta e oito vírgula nove) deles informaram que “às vezes” o aplicativo é rápido. Como dito anteriormente, essa ocorrência pode estar relacionada a diversos motivos técnicos, tais como limite de memória disponível no aparelho, tarefas em execução em

segundo plano e velocidade da conexão com a internet, sendo que essas condições fogem ao escopo da aplicação. Considerando as limitações técnicas citadas e considerando que 61,1% (sessenta e um vírgula um por cento) dos usuários afirmam que o aplicativo é rápido na execução das tarefas, conclui-se que a característica de eficiência obteve grande aceitação entre os usuários.

No geral, os resultados dessa avaliação foram considerados positivos e o aplicativo obteve sucesso em seu objetivo, dando mobilidade e flexibilidade aos processos de consulta de estoque, lançamento e liberação de pedidos da empresa Insumos LTDA. Também com a conclusão deste trabalho não existe mais a necessidade da utilização e armazenamento dos blocos de pedidos.

5.1 TRABALHOS FUTUROS

Correções e melhorias que foram apontadas nas avaliações neste projeto podem ser aplicadas pensando em manter a satisfação dos usuários do aplicativo. Das melhorias que podem ser feitas em projetos futuros:

- Manter localmente no aplicativo o pedido que está sendo criado para que no caso de falha o usuário possa retornar ao processo com os dados já preenchidos anteriormente, evitando retrabalhos.
- Alterar o processo manual de atualização dos dados do aplicativo para modo automático quando houver acesso a internet, dessa forma evita que os usuários tenham acesso a informações incorretas.
- Reforçar a regra de percentual máximo de desconto por meio de mensagem informativa ao usuário na funcionalidade da liberação de pedido.

A avaliação que ocorreu neste projeto deu ênfase apenas na satisfação dos usuários em relação ao aplicativo, pois existia a necessidade imediata de sua utilização, tornando curto o tempo para o desenvolvimento do projeto. Sendo assim, não foi aplicado um formulário para as características de manutenibilidade e portabilidade que avaliam a qualidade da parte técnica do projeto. Em trabalho futuro

pode ser realizado a avaliação dessas características, melhorando o *software* não apenas para o usuário final, mas também para a equipe de desenvolvimento.

O presente trabalho não utilizou de um recurso disponível na ferramenta Flutter, que possibilita gerar uma versão do aplicativo para *iPhone Operating System* (iOS). Para gerar essa versão é necessário de um recurso que a empresa não tinha disponível no momento, que seria um computador com sistema operacional da empresa *Apple*. Além do computador, também existe a necessidade de possuir uma licença de desenvolvedor.

Por outro lado, não houve interesse por parte da empresa em disponibilizar essa versão, pois seus colaboradores não possuíam um *smartphone* com esse sistema operacional. Caso a empresa adquira os recursos necessários para esse versionamento, em trabalhos futuros, pode ser criada essa versão para caso um colaborador necessite utilizar em um aparelho com sistema operacional IOS.

REFERÊNCIAS

ALBINO, Raphael Donaire; SOUZA, Cesar Alexandre de; PRADO, Edmir Parada Vasques. Benefícios alcançados através de um modelo de Gestão Ágil de Projeto em uma empresa de jogos eletrônicos. **Revista de Gestão e Projetos- GeP**. São Paulo(SP). v. 5, n. 1, p. 15-17, abr. 2014. Disponível em: <<https://periodicos.uninove.br/gep/article/view/9585/4329>>. Acesso em: 20 ago. 2022.

ALSULTANNY, Yas A.; WOHAISHI, Ahmed M.. Requirements of Software quality Assurance Model. In: INTERNATIONAL CONFERENCE ON ENVIRONMENTAL AND COMPUTER. 2., 2009, Dubai. **Proceedings...** Institute of Electrical and Electronics Engineers (IEEE), 2009. p. 19-23.

AKER, Jenny; MBITI, Isaac M.. Mobile Phones and Economic Development in Africa. **Journal of Economic Perspectives**, v. 24, n. 3, p. 207-232, jun. 2010.

ANDRADE, Antonio J. F.; et al. **Gestão de Projeto com Scrum: Um Estudo de Caso**. ENUCOMP – Encontro Unificado de Computação em Parnaíba, 2012. Disponível em: <<https://www.enucomp.com.br/2012/conteudos/artigos/scrum.pdf>>. Acesso em: 12 set. 2022.

BARTIÉ, Alexandre. **Garantia de qualidade de software: adquirindo maturidade organizacional**. 1. ed. Rio de Janeiro: GEN LTC, 2002.

BELLINASSO, Marco. **Flutter: the good, the bad and the ugly**. 2018. Disponível em: <<https://medium.com/asos-techblog/flutter-vs-react-native-for-ios-android-app-development-c41b4e038db9>>. Acesso em: 24 set. 2022.

BISOL, Gabriel A.. **DESENVOLVIMENTO DE UMA SOLUÇÃO PARA TROCA DE DADOS ENTRE UM ERP E UM APLICATIVO MÓVEL**. 2019. 82 f. Dissertação (Trabalho de Conclusão de Curso) – Programa de Graduação de Bacharel em Sistemas de Informação, Universidade de Caxias do Sul. Caxias do Sul, 2019.

BRITO, Hugo J. B.. **JavaScript para aplicações móveis**. 2019. 11 f. Dissertação (Mestrado) - Especialização em Engenharia de Software, Instituto Superior de Contabilidade e Administração de Coimbra. Coimbra, 2019.

CÔRTEZ, Mario L.; CHIOSSI, Thelma C. dos S. **Modelos de Qualidade de Software**. São Paulo: Editora da Unicamp, Instituto de Computação, 2001.

COSTA, Carlos; et al. Integração de Sistemas de Informação Universitários via Web Services. In: ACTAS DA 5ª CONFERENCIA IBÉRICA DE SISTEMAS Y TECNOLOGIAS DA INFORMACIÓN. 5., 2010, Santiago de Compostela. **Anais...** Braga: APPACDM, 2010. p. 290-295.

COSTA, Mirlanda S.. Sistemas web e mobile: uma visão geral para negócios empresariais. **Revista Científica Multidisciplinar Núcleo do Conhecimento**. São Paulo(SP). v. 9, n. 8, p. 82-99, ago. 2018. Disponível em: <<https://www.nucleodoconhecimento.com.br/ciencia-da-computacao/sistemas-web-e-mobile>>. Acesso em: 07 set. 2022.

DATE, C. J. **Introdução a sistemas de bancos de dados**. 1. ed. Rio de Janeiro: GEN LTC, 2004.

DUARTE, Katia C.; FALBO, Ricardo A.. **Uma ontologia de qualidade de software**. 2006. 11 f. Dissertação (Mestrado), Universidade Federal do Espírito Santo. Vitória, 2006.

EMER, Jean C.. **O grande desencontro do HTTP com o HTML**, 2014. Disponível em: <<http://tableless.com.br/o-grande-desencontro-http-com-o-html/>>. Acesso em: 10 ago. 2022.

FELLER, Nadja J.. **Estendendo Rest-Unit: Geração Baseada em U2T` de Drivers de Teste para RESTful Web Services**. 2010. 97 f. Dissertação (Trabalho de Conclusão de Curso) – Programa de Graduação de Bacharel em Ciência da Computação, Universidade Federal do Rio Grande do Sul. Porto Alegre, 2010.

GARCIA, Cristiano. M.; ABILIO, Ramon; MALHEIROS, Neumar. **Abordagens e Tecnologias para Integração de Sistemas: Um Estudo de Caso na Universidade Federal de Lavras**, 2015. Disponível em: <http://www.fsma.edu.br/si/edicao15/FSMA_SI_2015_1_Principal_5.pdf>. Acesso em: 18 set. 2022.

GENEXUS. **GeneXus by Globand**, 2022. Disponível em: <<https://www.genexus.com/pt/produtos/genexus/tecnologias-suportadas>>. Acesso em: 26 set. 2022.

KALIN, Martin. Scrum Essencial: **Java web services implementando**. 1. ed. Rio de Janeiro: Alta Books, 2010.

KEMP, Simon. **DIGITAL 2022: GLOBAL OVERVIEW REPORTI**, 2022. Disponível em: <<https://datareportal.com/reports/digital-2022-global-overview-report>>. Acesso em: 27 ago. 2022.

LEITÃO, Michele de V.. **Aplicação de Scrum em Ambiente de Desenvolvimento de Software Educativo**. 2010. 97 f. Monografia (Trabalho de Conclusão de Curso) – Programa de Graduação de Bacharel em Engenharia da Computação, Universidade de Pernambuco. Recife, 2010.

LELER, W. **What's Revolutionary about Flutter**, 2017. Disponível em: <<https://hackernoon.com/whats-revolutionary-about-flutter-946915b09514>>. Acesso em: 11 set. 2022.

MARTINS, Victor Manuel Moreira. **Integração de sistemas de informação: Perspectivas, normas e abordagens**. 2005. 201 f. Dissertação (Mestrado). Universidade do Minho. Guimarães, 2005.

MENÉNDEZ, Andrés I. J.. **Uma ferramenta de apoio ao desenvolvimento de Web Services**. 2002. 97 f. Dissertação (Trabalho de Conclusão de Curso) – Programa de Graduação de Bacharel em Ciência da Computação, Universidade Federal de Campina Grande. Campina Grande, 2002.

MORO, Tharcis D.; DORNELES, Carina; REBONATTO, Marcelo T. Web services WS-* versus Web Services REST. **Revista de Iniciação Científica**, Universidade de Passo Fundo (RS), v. 11, n. 1, p. 36-51, jun. 2011.

MURAKAMI, Ives S.; MARQUES, João J.; LIMA, Wilian. **Análise de bibliotecas para web services no desenvolvimento em smartphones baseado no sistema operacional Microsoft Windows Phone**. 2013. 62 f. Monografia (Trabalho de Conclusão de Curso) – Programa de Graduação de Engenheiro de Computação, Universidade São Francisco. Itatiba, 2013.

RICHARDSON, Leonard; RUBY, Sam. **RESTful Web Services**. 1 ed. Sebastopol: O'Reilly Media, Inc, 2007.

ROCHA, A. R. Cavalcanti da; MALDONADO, J. Carlos; WEBER, K. Chaves. **Qualidade de software: Teoria e Prática**. 1. ed. São Paulo: Prentice Hall, 2001.

RUBIN, Kenneth S.. **Scrum Essencial: Um guia prático para o mais popular processo ágil**. 1. ed. Rio de Janeiro: Alta Books, 2018.

OLIVEIRA, Ricardo R.. **Desenvolvimento de Web Services RESTful e SOAP-WSDL utilizando as Implementações de Referência JAX-RS e JAX-WS**. 2011. 82f. Dissertação (Mestrado) – Curso de Web Services do Instituto de Ciências Matemáticas e de Computação ICMC/USP, Universidade de São Paulo. São Carlos, 2011.

POCIOT, Marcel. **What's Revolutionary about Flutter**, 2017. Disponível em: <<https://hackernoon.com/whats-revolutionary-about-flutter-946915b09514>> . Acesso em: 09 out. 2022.

PORTO, Fabiano. **Aplicativos mobile: definições, histórias e previsões**, 2012. Disponível em: <<http://tectriadebrasil.com.br/blog/2012/11/07/aplicativos-mobile-definicoes-historia-e-previsoes/>>. Acesso em: 12 set. 2022.

POSTGRESQL. **About PostgreSQL**, 2022. Disponível em: <<https://www.postgresql.org/about/>>. Acesso em: 5 out. 2022.

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

QUE, Peixin; GUO, Xiao; ZHU, Maokun. A Comprehensive Comparison Between Hybrid and Native App. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE AND COMMUNICATION NETWORKS. 8., 2016, Tehri. **Proceedings...** Tehri: IEEE, 2017. p. 611-614.

SAVOINE, M.; et al. Análise de Gerenciamento de Projeto de Software Utilizando Metodologia Ágil XP e Scrum: Um Estudo de Caso Prático. In: XI Encontro de Estudantes de Informática do Tocantins. 11., 2009, Palmas. **Anais...** Palmas: Centro Universitário Luterano de Palmas, 2009. p. 93-102.

SCHWABER, Ken; SUTHERLAND, Jeff. O guia definitivo para o Scrum: As regras do jogo. **Guia do Scrum**, 2020. Disponível em: <<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Portuguese-BR-3.0.pdf>>. Acesso em: 15 set. 2022.

SIBISI, M; WAVEREN, C. c. van. A process framework for customising software quality models. In: AFRICON 2007. 3., 2007, Windhoek. **Anais...** Institute of Electrical and Electronics Engineers (IEEE), 2007. p. 1-8.

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. São Paulo: Pearson Education do Brasil, 2011.

STATISTA. **Cross-platform mobile frameworks used by software developers worldwide from 2019 to 2021**, 2022. Disponível em: <<https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/#main-content>> 12 set. 2022.

STORCH, Mirian M.. **Proposta de avaliação da qualidade de produtos de software utilizando a norma ISO/IEC 9126**. 2000. 97 f. Monografia (Trabalho de Conclusão de Curso) – Programa de Graduação de Bacharelado em Ciências da Computação, Universidade Regional de Blumenau. Blumenau, 2000.

SUTHERLAND, Jeff. **Guia do Scrum**, 2010. Disponível em: <https://www.researchgate.net/publication/301685699_Jeff_Sutherland%27s_Scrum_Handbook>. Acesso em: 15 set. 2022.

VILCEK, Tena; JAKOPEC, Tomislav. Comparative analysis of tools for development of native and hybrid mobile applications. In: International Convention on Information and Communication Technology, Electronics and Microelectronics. 40., 2017, Opatija. **Proceedings...** Opatija: IEEE, 2017. p. 1516-1521.

APÊNDICE A - Pesquisa de Satisfação do Aplicativo Vendas Mobile

PESQUISA DE SATISFAÇÃO DO APLICATIVO VENDAS MOBILE

O presente formulário tem a intenção de avaliar algumas funcionalidades e características do Aplicativo que vem sendo utilizado atualmente, a fim de promover a melhoria contínua, levando em consideração a avaliação dos usuários.

Figura 1 - Perguntas referentes ao perfil do usuário

<p>Nome</p> <p>Sua resposta _____</p>
<p>Cargo ou Função *</p> <p><input type="radio"/> Consultor de Venda</p> <p><input type="radio"/> Gerente Comercial</p>
<p>Idade *</p> <p><input type="radio"/> 18 - 25</p> <p><input type="radio"/> 26 - 30</p> <p><input type="radio"/> 31 - 35</p> <p><input type="radio"/> 36 - 40</p> <p><input type="radio"/> 41 - 45</p> <p><input type="radio"/> Mais de 46</p>
<p>Filial *</p> <p><input type="radio"/> Castro - PR</p> <p><input type="radio"/> Irati - PR</p> <p><input type="radio"/> Ponta Grossa - PR</p> <p><input type="radio"/> Itapeva - PR</p> <p><input type="radio"/> Taquarituba - PR</p>

Fonte: Autoria própria (2022).

Figura 2 - Pergunta 1 referente a funcionalidade

1 - O aplicativo realiza o lançamento de pedidos corretamente ? *

Sim

Não

Em caso negativo apresente uma justificativa.

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 3 - Pergunta 2 referente a funcionalidade

2 - O aplicativo/sistema realiza a liberação de pedido corretamente ? *

Sim

As Vezes

Não

Em caso não libere corretamente apresente uma justificativa.

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 4 - Pergunta 3 referente a funcionalidade

3 - O aplicativo dispõe de todas as funções necessárias para o lançamento de pedidos ? *

Sim

Algumas

Nenhuma

Caso você acredite que esteja faltando alguma funcionalidade, deixe sua sugestão.

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 5 - Pergunta 4 referente a funcionalidade

4 - Quantas vezes por dia você costuma atualizar os dados do aplicativo?

Apenas uma

Duas a cinco

Mais que cinco vezes

Fonte: Autoria própria (2022).

Figura 6 - Pergunta 5 referente a funcionalidade

5 - O aplicativo calcula o valor dos produtos de forma correta ? *

Sim

As Vezes

Nunca

Fonte: Autoria própria (2022).

Figura 7 - Pergunta 6 referente a funcionalidade

6 - O aplicativo informa corretamente os saldos do estoque ? *

Sim

As Vezes

Nunca

Fonte: Autoria própria (2022).

Figura 8 - Pergunta 7 referente a funcionalidade

7 - O aplicativo informa corretamente os dados dos pedidos a serem liberados ? *

Sim, todos dos dados são apresentados

Sim, os dados principais são apresentados (Ex: Cliente, Data Pagamento, Valor Venda)

Sim, mas faltam dados para avaliação da liberação

Não, os pedidos não são apresentados para liberação

Caso você acredite que esteja faltando algum dado, deixe sua sugestão.

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 9 - Pergunta 8 referente a funcionalidade

8 - O pedido criado pelo aplicativo é enviado corretamente para o sistema web ? *

Sim

As Vezes

Nunca

Fonte: Autoria própria (2022).

Figura 10 - Pergunta 9 referente a confiabilidade

9 - O aplicativo sempre informa os campos obrigatórios que não foram preenchidos ao finalizar o pedido? (Ex. Cliente, Data Vencimento, Produto, Quantidade, Valor) *

Sempre

As Vezes

Nunca

Fonte: Autoria própria (2022).

Figura 11 - Pergunta 10 referente a confiabilidade

10 - O aplicativo informa quando o pedido não foi enviado ao sistema web? *

Sim

As Vezes

Não

Fonte: Autoria própria (2022).

Figura 12 - Pergunta 11 referente a confiabilidade

11 - O aplicativo informa quando é realizada uma entrada de dados é inválida (Ex: * quando insere letra no lugar de numero) ?

Sempre

As Vezes

Nunca

Fonte: Autoria própria (2022).

Figura 13 - Pergunta 11 referente a confiabilidade

12 - O aplicativo é capaz de dar continuidade em um processo em andamento em * caso de falha ? (Ex: aplicativo fecha durante o processo de lançamento de um pedido)

OBS: Entende-se como processo: o Lançamento e Liberação de Pedido e Carregamento de Dados

Sim

As Vezes

Nunca

Em quais situações já ocorreu uma falha e foi necessário recomeçar o processo?

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 14 - Pergunta 13 referente a usabilidade

13 - É fácil lançar pedidos no aplicativo ? *

Sim

Não

Qual foi sua dificuldade?

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 15 - Pergunta 14 referente a usabilidade

14 - É fácil de realizar a consulta de estoque ? *

Sim

Não

Qual foi sua dificuldade?

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 16 - Pergunta 15 referente a usabilidade

15 - É fácil de realizar a liberação de pedidos ? *

Sim

Não

Qual foi sua dificuldade?

Sua resposta _____

Fonte: Autoria própria (2022).

Figura 17 - Pergunta 16 referente a usabilidade

16 - É fácil de navegar entre as funcionalidades do aplicativo ? *

Sim

Não

Fonte: Autoria própria (2022).

Figura 18 - Pergunta 17 referente a eficiência

17 - O aplicativo é rápido na execução das tarefas? *

Sim

As Vezes

Não

Caso a resposta seja negativa, quais situações e tarefas?

Sua resposta _____

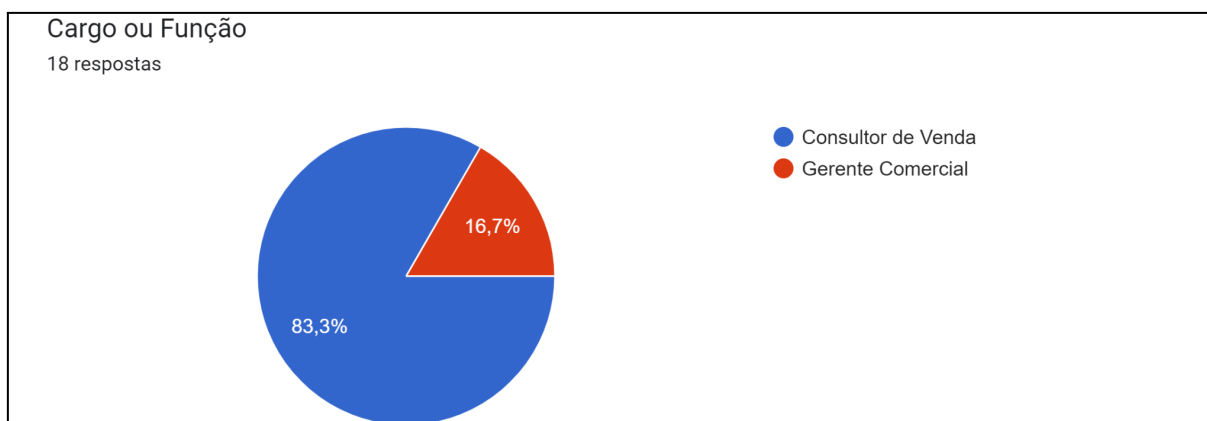
Fonte: Autoria própria (2022).

APÊNDICE B - Gráficos da Respostas da Pesquisa de Satisfação

GRÁFICOS REFERENTES A RESPOSTAS DA PESQUISA DE SATISFAÇÃO

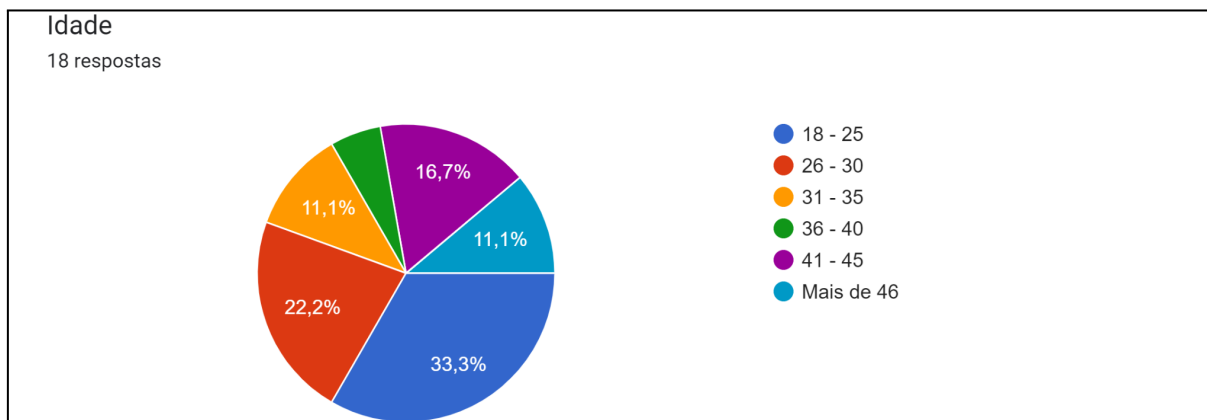
Este documento apresenta a visualização gráfica do formulário de validação da aplicação desenvolvida, aplicada a 18 funcionários da empresa Insumos LTDA, nas cidades de Castro - PR, Irati - PR, Ponta Grossa - PR, Itapeva - SP e Taquarituba - SP, servindo como fonte de dados para o Trabalho de Conclusão de Curso da UTFPR.

Figura 1 - Gráfico referente aos cargos



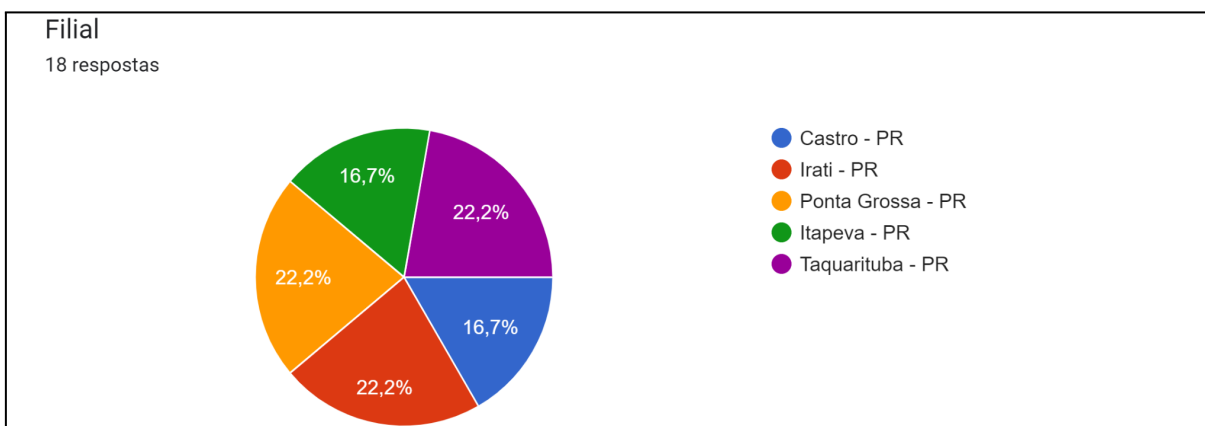
Fonte: Autoria própria (2022).

Figura 2 - Gráfico referente às idades



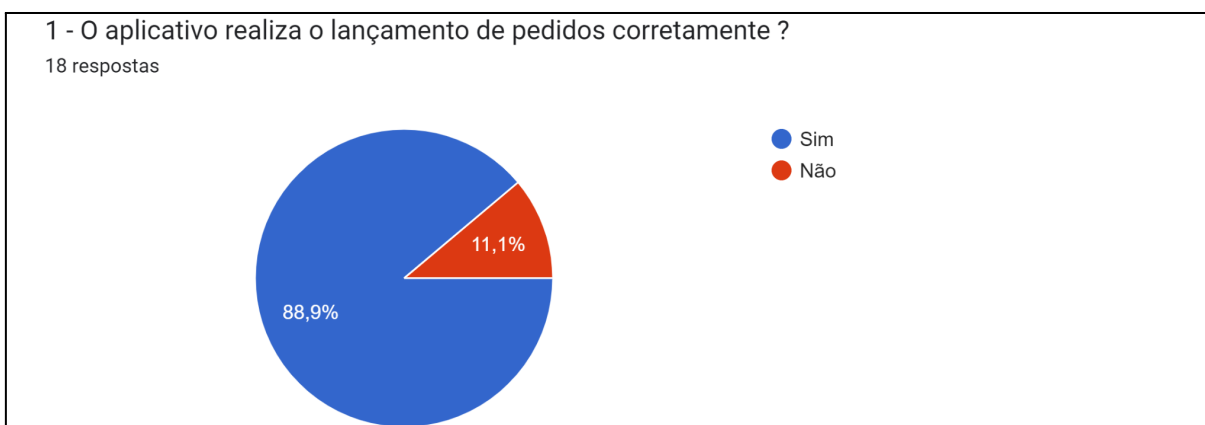
Fonte: Autoria própria (2022).

Figura 3 - Gráfico referente às filiais



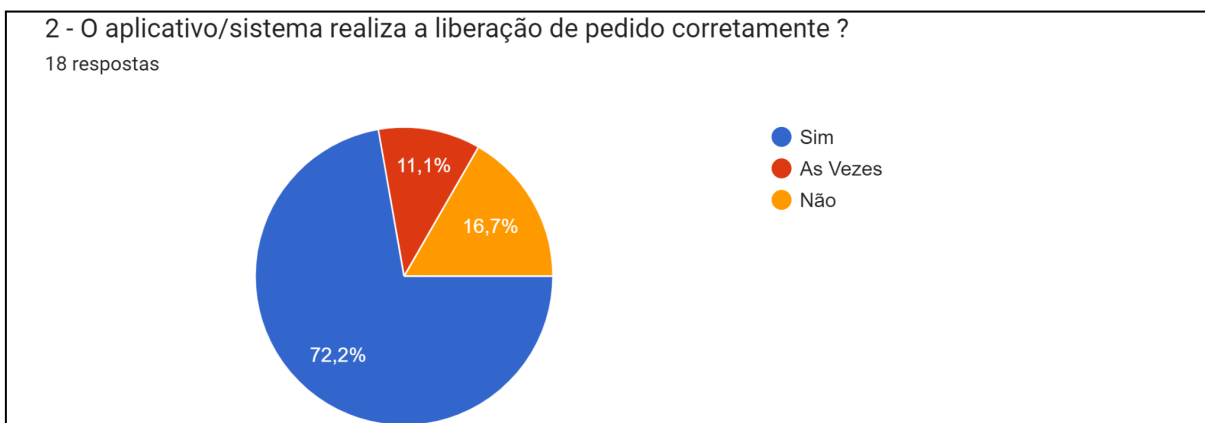
Fonte: Autoria própria (2022).

Figura 4 - Gráfico referente à questão 1 sobre funcionalidade



Fonte: Autoria própria (2022).

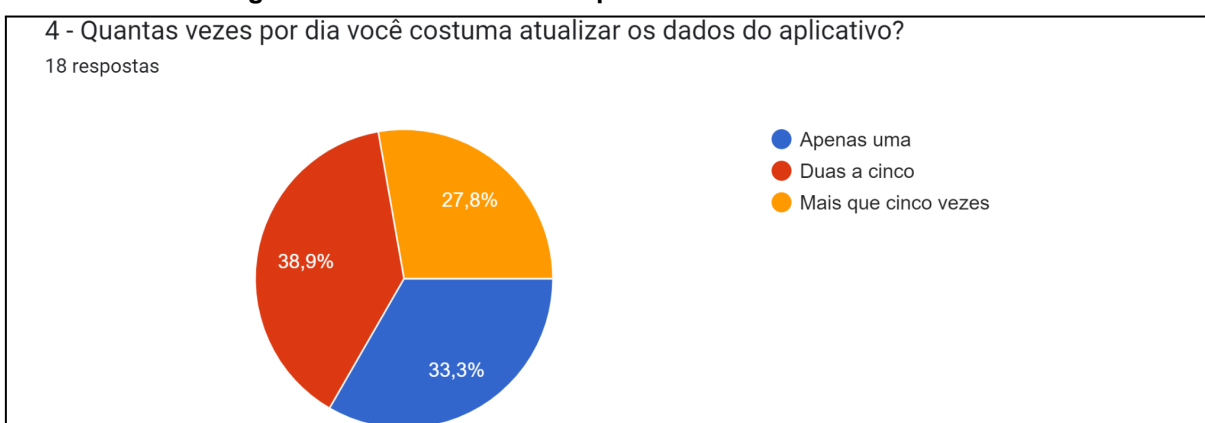
Figura 5 - Gráfico referente à questão 2 sobre funcionalidade



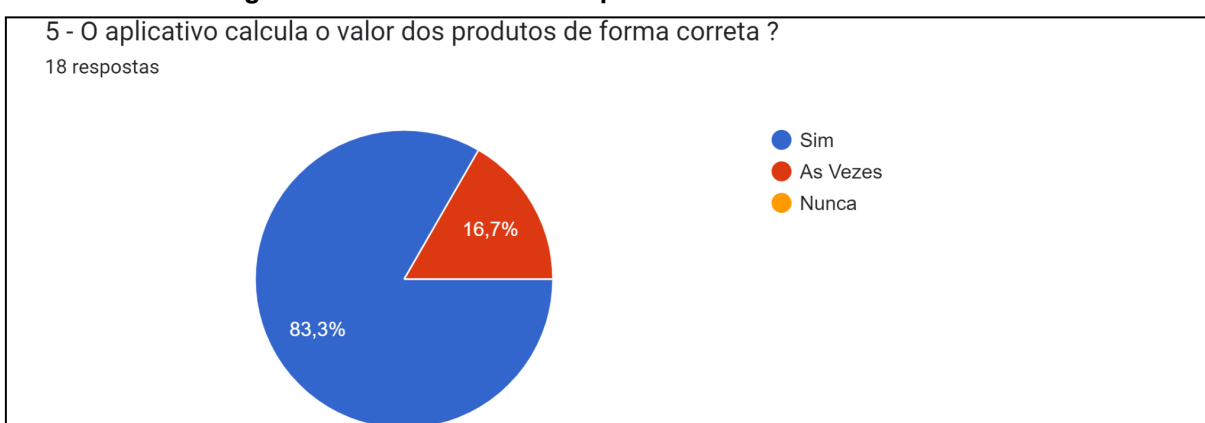
Fonte: Autoria própria (2022).

Figura 6 - Gráfico referente à questão 3 sobre funcionalidade

Fonte: Autoria própria (2022).

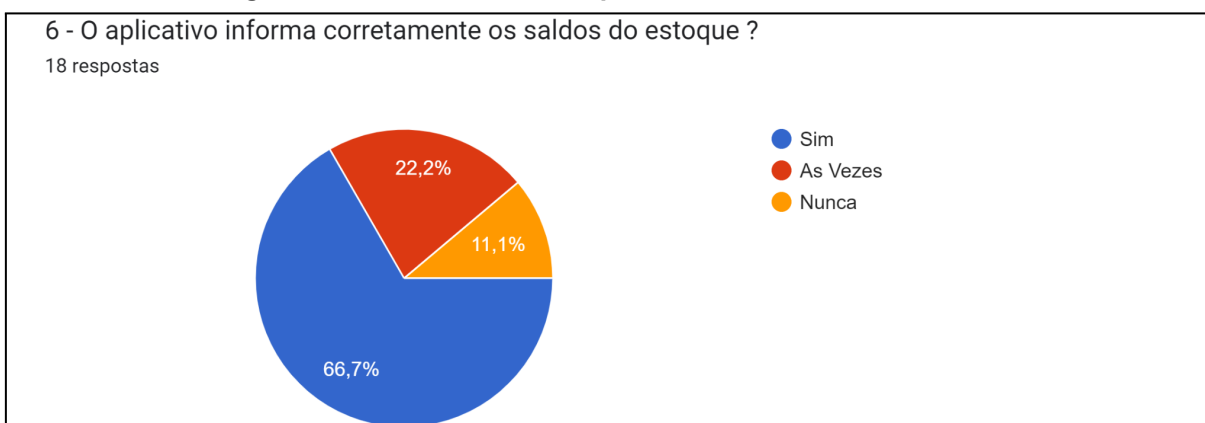
Figura 7 - Gráfico referente à questão 4 sobre funcionalidade

Fonte: Autoria própria (2022).

Figura 8 - Gráfico referente à questão 5 sobre funcionalidade

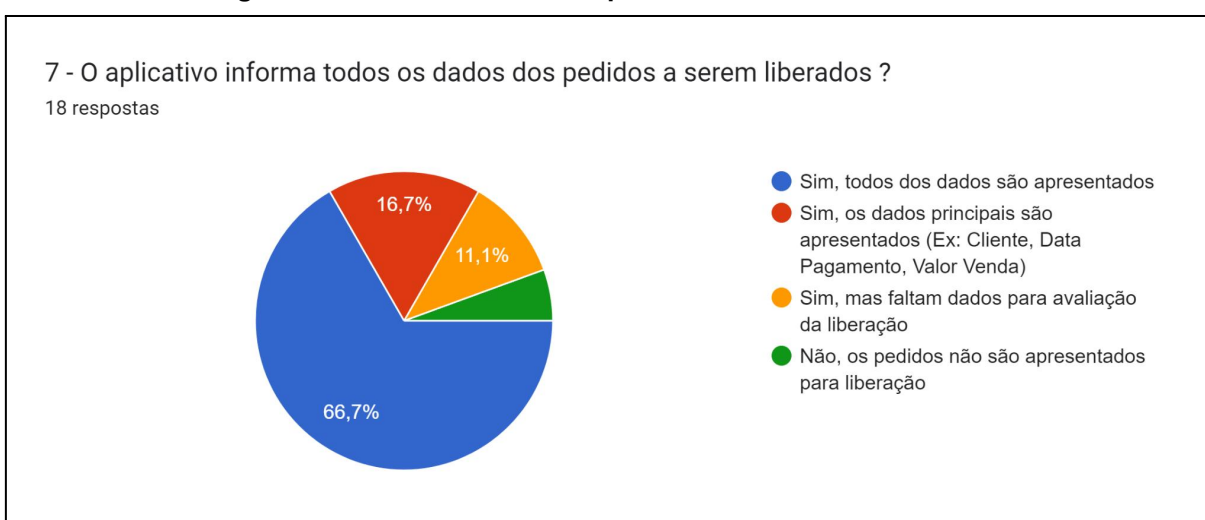
Fonte: Autoria própria (2022).

Figura 9 - Gráfico referente à questão 6 sobre funcionalidade



Fonte: Autoria própria (2022).

Figura 10 - Gráfico referente à questão 7 sobre funcionalidade

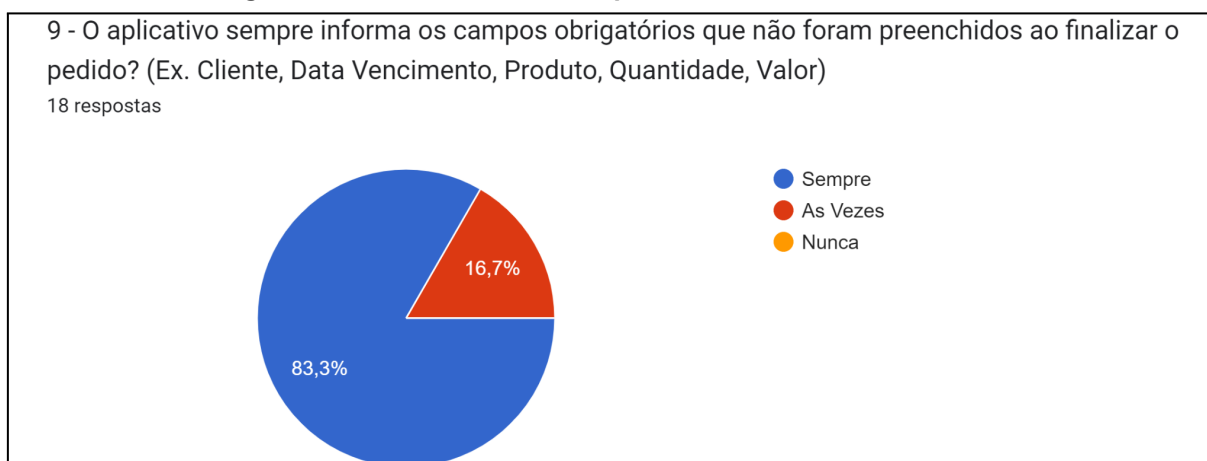


Fonte: Autoria própria (2022).

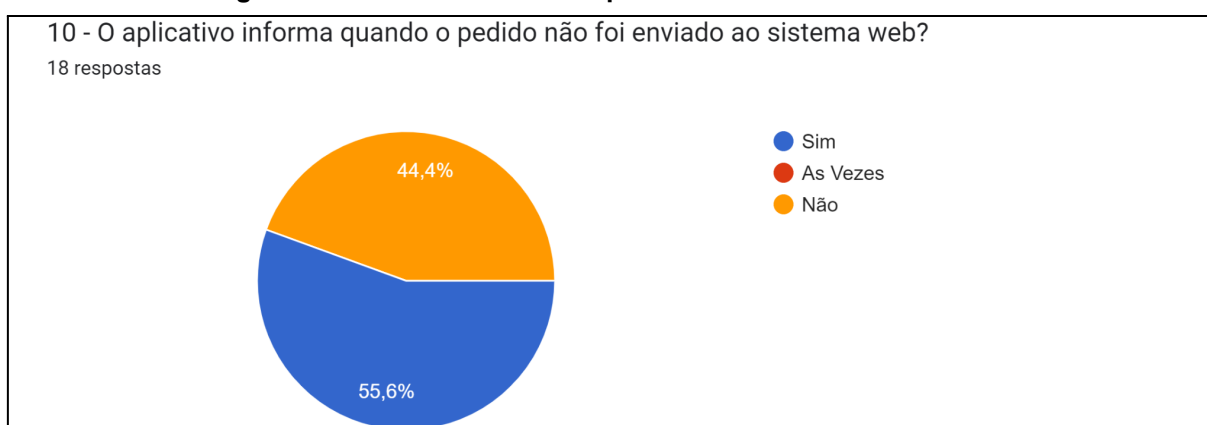
Figura 11 - Gráfico referente à questão 8 sobre funcionalidade



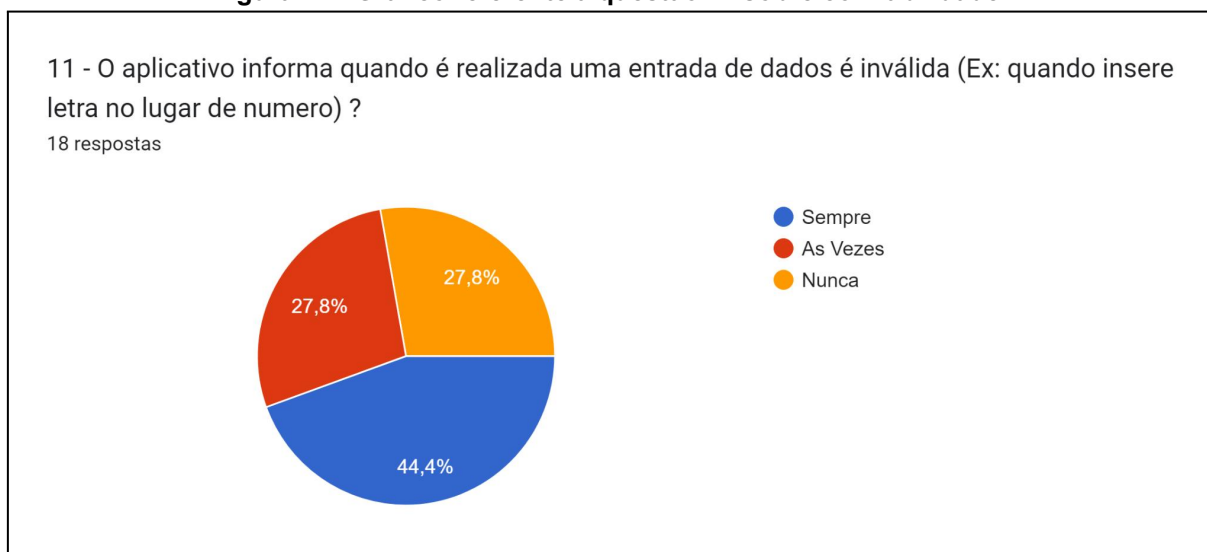
Fonte: Autoria própria (2022).

Figura 12 - Gráfico referente à questão 9 sobre confiabilidade

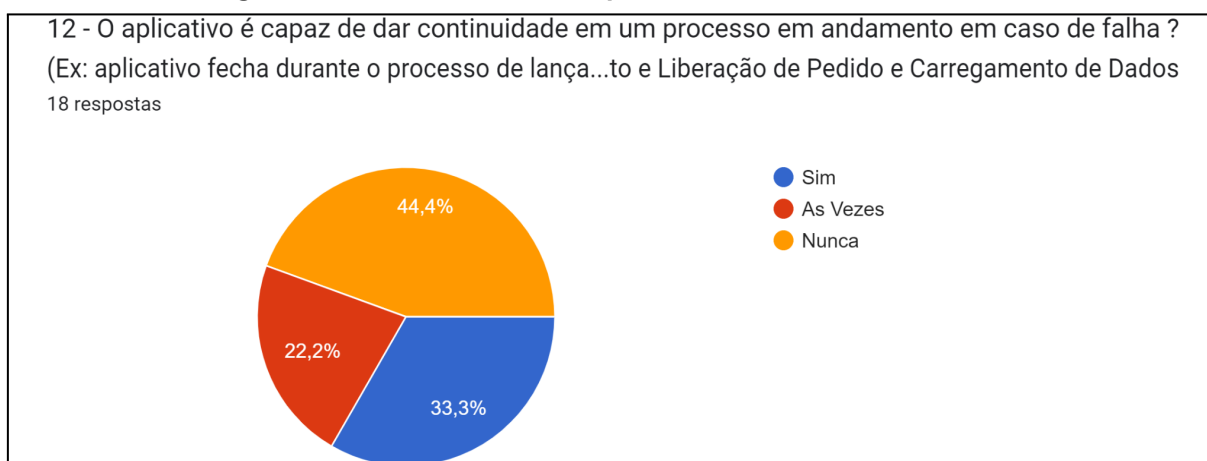
Fonte: Autoria própria (2022).

Figura 13 - Gráfico referente à questão 10 sobre confiabilidade

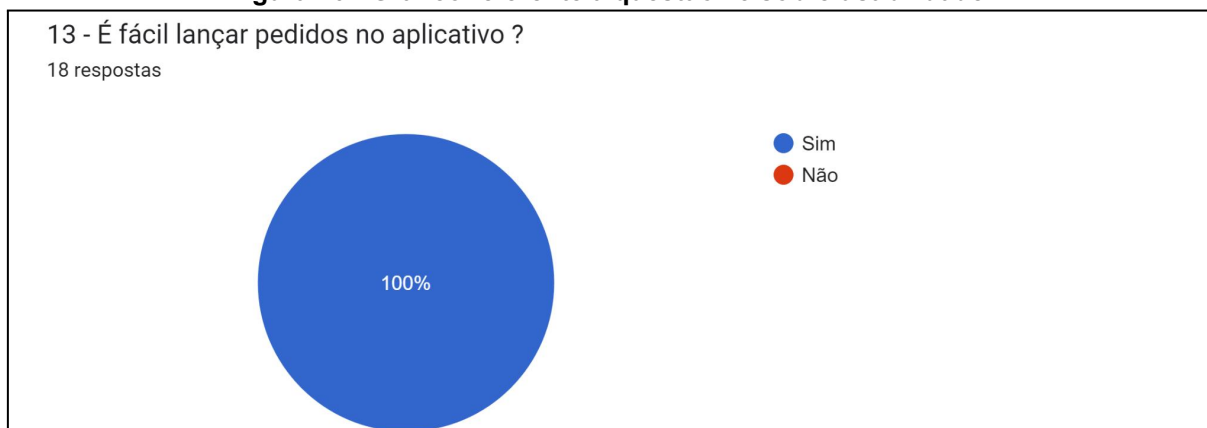
Fonte: Autoria própria (2022).

Figura 14 - Gráfico referente à questão 11 sobre confiabilidade

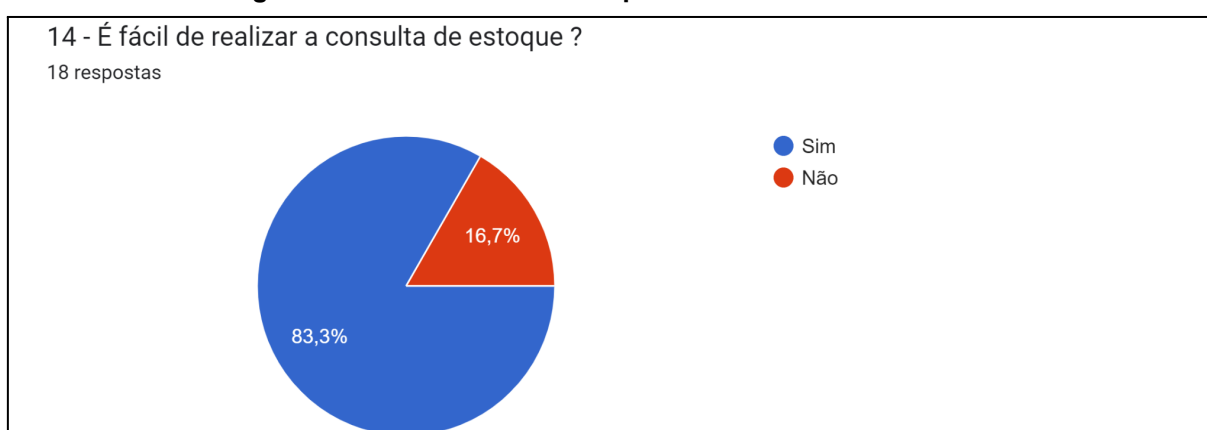
Fonte: Autoria própria (2022).

Figura 15 - Gráfico referente à questão 12 sobre confiabilidade

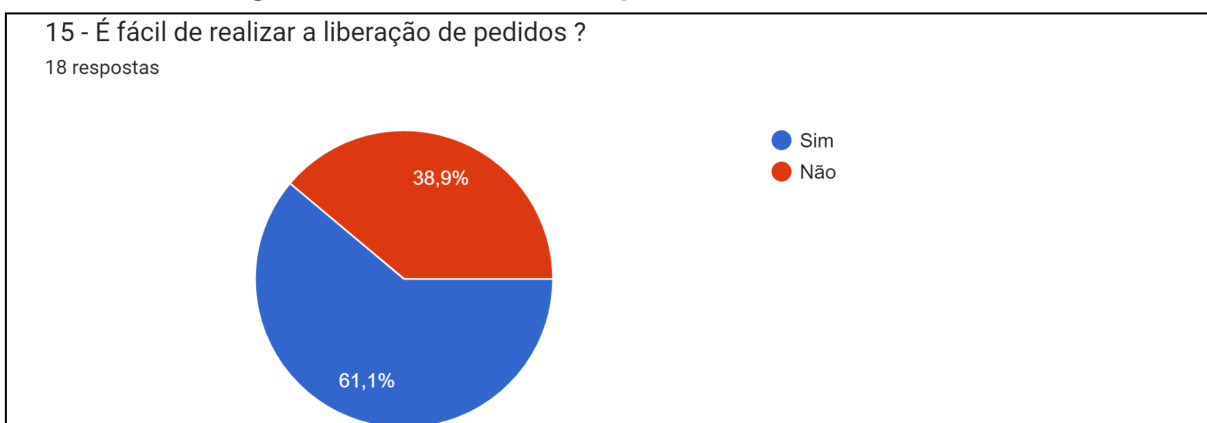
Fonte: Autoria própria (2022).

Figura 16 - Gráfico referente à questão 13 sobre usabilidade

Fonte: Autoria própria (2022).

Figura 17 - Gráfico referente à questão 14 sobre usabilidade

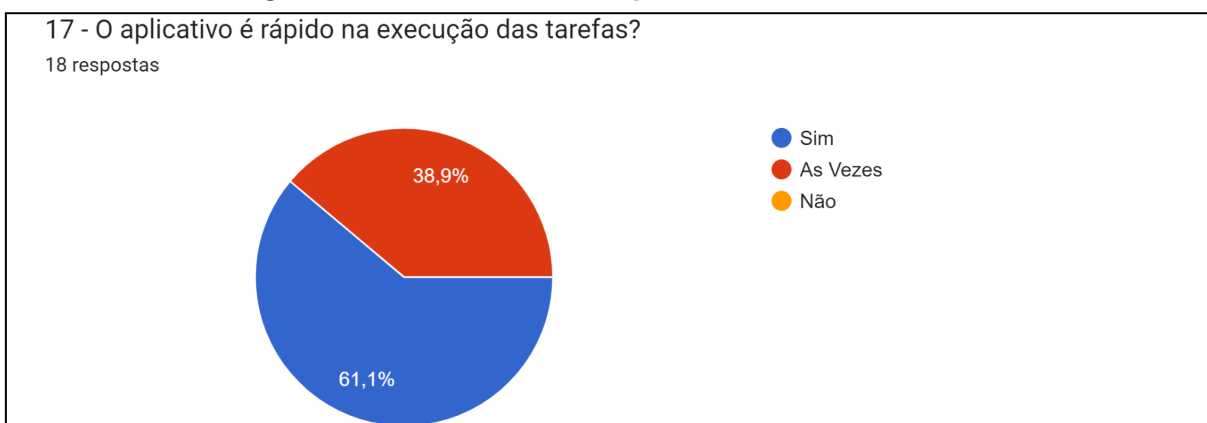
Fonte: Autoria própria (2022).

Figura 18 - Gráfico referente à questão 15 sobre usabilidade

Fonte: Autoria própria (2022).

Figura 19 - Gráfico referente à questão 16 sobre usabilidade

Fonte: Autoria própria (2022).

Figura 20 - Gráfico referente à questão 17 sobre eficiência

Fonte: Autoria própria (2022).