

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DO RIO GRANDE DO SUL
CAMPUS RESTINGA**

GUILHERME MÜLLER SEVERO

SISTEMA DE GESTÃO PARA OFICINAS DE MOTOCICLETAS

**Porto Alegre
2023**

GUILHERME MÜLLER SEVERO

SISTEMA DE GESTÃO PARA OFICINAS DE MOTOCICLETAS

Trabalho de conclusão de curso apresentado ao Curso de análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, como requisito parcial para a Obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Gleison Samuel do Nascimento

**Porto Alegre
2023**

SISTEMA DE GESTÃO PARA OFICINAS DE MOTOCICLETAS

Trabalho de conclusão de curso apresentado ao Curso de análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, como requisito parcial para a Obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Gleison Samuel do Nascimento

Aprovado em 11 de setembro, 2023

Professor Gleison Samuel do Nascimento

Membro da Banca – Professor Rafael Esteves – IFRS – Campus Restinga

Membro da Banca – Professora Sandra Rovena Frigeri – IFRS – Campus Restinga

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO SUL

Reitor: Prof. Júlio Xandro Heck

Pró-Reitora de Ensino: Prof. Lucas Coradini

Diretor-geral do *Campus* Restinga: Prof. Rudinei Müller

Coordenador do CST em Análise e Desenvolvimento de Sistemas: Prof. Roben Castagna Lunardi

Bibliotecária-chefe do *Campus* Restinga: Paula Porto Pedone

RESUMO

Esta pesquisa tem como objetivo desenvolver uma aplicação para gerenciamento e fidelização de clientes para oficinas de motocicletas. Com a pandemia e abertura de empregos em aplicativos de delivery, houve um aumento na utilização de oficinas mecânicas devido a crescente comercialização de motocicletas. Assim, o uso da tecnologia pode ser um diferencial entre diferentes estabelecimentos. A solução desenvolvida neste trabalho propõe uma plataforma web para o gerenciamento de uma oficina mecânica especializada em motocicleta localizada na cidade de Viamão. Para auxiliar no processo de gestão da oficina, o sistema terá como funcionalidade o gerenciamento e orçamento de peças, cadastro e histórico de manutenções, comunicação com os clientes, dashboard para gestão do gerente e geração de relatórios. Mostra-se que o uso da plataforma para o gerenciamento é de extrema importância para a oficina, pois permite que o cliente possa acessar e acompanhar o histórico de serviços realizados em sua motocicleta e o profissional terá o controle do seu estoque de produtos.

Palavra-chave: Sistema web, usuário e processo.

ABSTRACT

This research aims to develop an application for management and customer loyalty for motorcycle workshops. With the pandemic and the opening of jobs in delivery apps, there was an increase in the use of mechanic workshops due to the growing sale of motorcycles. Thus, the use of technology can be a differentiator between different establishments. The solution developed in this work proposes a web platform for managing a mechanical workshop specializing in motorcycles located in the city of Viamão. To assist in the workshop management process, the system will have the functionality of parts management and budget, maintenance registration and history, communication with customers, dashboard for manager management and generation of reports. It is shown that the use of the platform for management is extremely important for the workshop, as it allows the customer to access and monitor the history of services carried out on their motorcycle and the professional will have control of their product stock.

Keywords: *Web system, user and process.*

LISTA DE ILUSTRAÇÕES

Figura 1 - Proprietários de motocicletas	21
Figura 2 - Procedimentos	21
Figura 3 - Manutenção periódica	22
Figura 4 - Prevenção	22
Figura 5 - Oficinas	23
Figura 6 - Melhor oficina	23
Figura 7 - Aplicativos para gerenciamento	24
Figura 8 - Utilização	24
Figura 9 - Funcionalidades do sistema	25
Figura 10 - Diagrama Casos de uso	28
Figura 11 - Diagrama Casos de uso	29
Figura 12 - Diagrama de Classes (Produto, Serviço, Motocicleta, Usuário, Gerente, Cliente, Papel, Permissão)	31
Figura 13 - Diagrama ER (Produto, Serviço, Motocicleta, Usuário, Papel, Status, Permissão)	32
Figura 14 - Protótipos	33
Figura 15 - Arquitetura cliente servidor	34
Figura 16 - Estrutura da aplicação	35
Figura 17 - Estrutura de pastas Nestjs	36
Figura 18 - Estrutura de pastas Nextjs	37
Figura 19 - Estrutura REST API	38
Figura 20 - Modelo usuário	39
Figura 21 - Rotas de usuários	40
Figura 22 - Serviço do usuário	41
Figura 23 - Página principal	42
Figura 24 - Tela de login	43
Figura 25 - Tela de cadastro	43
Figura 26 - Dashboard com resumo de dados	44
Figura 27 - Tela de listagem de produtos	44
Figura 28 - Tela de listagem de motos	45
Figura 29 - Tela de listagem de serviços	46

LISTA DE TABELAS

Tabela 1 - Comparativo entre os sistemas	18
Tabela 2 - Comparativo entre os sistemas	19
Tabela 3 - Resultado das perguntas feitas a oficina	20
Tabela 4 - Detalhamento do caso de uso Realizar Cadastro	30
Tabela 4 - Detalhamento do caso de uso Realizar Cadastro	31

LISTA DE ABREVIATURAS E SIGLAS

CSS	Cascading Style Sheets
HTML	Hyper Text Markup Language
IFRS	Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul
UML	Linguagem de Modelagem Unificada
PostgreSQL	Sistema gerenciador de banco de dados objeto relacional

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Objetivo Geral	14
1.2	Objetivos Específicos.....	14
2	TRABALHOS RELACIONADOS	16
2.1	GESTMEC: SISTEMA DE GESTÃO DE OFICINA MECÂNICA	17
2.2	SISTEMA INFORMATIZADO PARA MECÂNICAS	17
2.3	Comparativo dos trabalhos	18
3	LEVANTAMENTO DE REQUISITOS	20
3.1	Resultado da coleta de informações na oficina	20
3.2	Resultado da coleta de informações dos usuários.....	21
4	SOLUÇÃO CONCEITUAL	27
4.1	Diagrama de casos de uso	27
4.2	Diagrama de Classes	31
4.3	Diagrama Entidade Relacionamento.....	32
4.4	Prototipagem.....	32
5	DESENVOLVIMENTO	34
5.1	Implementação da camada de modelo.....	38
5.2	Implementação da camada de controle	39
5.3	Implementação da camada de visualização	41
5.4	Execução do sistema.....	42
5.5	Segurança	46
6	TESTES	47
7	CONCLUSÃO	48
	REFERÊNCIAS	49
	APÊNDICE A – DETALHAMENTO DAS TECNOLOGIAS UTILIZADAS...51	
A.1	HTML.....	51
A.2	Javascript	51
A.3	Typescript.....	51
A.4	React	52
A.5	Nextjs	52
A.6	Vercel.....	52
A.7	Nestjs.....	52

A.8	Postgres.....	53
A.9	Ngrok	53
A.10	Styled Components.....	53
	APÊNDICE B – DETALHAMENTO DOS CASOS DE USO	54
	ANEXO A – DOCUMENTO DE TESTE COM USUÁRIOS.....	68

1 INTRODUÇÃO

Com o crescimento constante da tecnologia, houve muitos pontos de inovação em praticamente todas as áreas. Entre elas, o uso de aplicativos de entregas por decorrência da pandemia disparou e com isso o mercado de dispositivos digitais (computadores, celulares) tem um forte crescimento. Esse aumento tem superado o número de habitantes. São 477 milhões de dispositivos em uso, sendo eles corporativos e domésticos, um aumento de mais de 26% nas vendas de smartphones em 2021. Esses números que tendem a crescer em 2022.

O aumento do mercado de aplicativos por entregas obteve uma escala de crescimento anual, chegando a um faturamento de R\$ 40 bilhões em 2021, 24% acima do ano anterior (MERCADO & CONSUMO, 2022).

Com o aumento das vendas por aplicativos delivery outras áreas foram beneficiadas, abrindo novas oportunidades de trabalho para as pessoas. Uma destas áreas está relacionada ao mercado de motocicletas.

A moto, é um veículo indispensável para executar diversas atividades de delivery, sendo um dos principais meios de locomoção para esse segmento de mercado.

De acordo com a Abraciclo (representante brasileira das fabricantes de motocicletas e bicicletas), a produção de motocicletas no Brasil deve ser de aproximadamente 1,29 milhão de motocicletas produzidas neste ano de 2022. Mesmo com a inflação em alta, afetando vários setores, um nicho que está se recuperando e que já obteve números melhores de venda em 2022, em relação ao ano de 2021, foi o mercado automobilístico. Outro ponto expressivo desse mercado foi a evolução das exportações em alta de 1%. Esse crescimento foi impulsionado pela pandemia, pois as motos desempenharam um papel importante, pois muitos negócios operaram através de serviços de delivery.

Para este ano de 2022 está previsto um aumento na produção de motos. Isto significa que será necessário aumentar a quantidade de oficinas mecânicas para suprir o aumento da quantidade de motos em circulação. Logo, o usuário terá mais opções de lugares para levar seu veículo ao reparo. Um ponto importante para quem necessita da moto para suas atividades, é averiguar sempre com antecedência e regularidade alguns pontos como:

- Gastos
 - Combustível
 - Oficina
 - Motopeças
 - Documentação
 - Seguro

Outro fator que impulsiona o mercado de motocicletas é o aumento substancial dos combustíveis. Com os sucessivos aumentos um dos meios de transportes que está cada vez mais em alta dentro do mercado automobilístico são as motocicletas, se comparadas ao carro, que exigem mais consumo e os gastos que são maiores. As motos por se tornarem versáteis dentro do trânsito brasileiro, acabam sendo mais econômicas desde sua compra até os gastos com combustível e manutenção.

Com a expansão do mercado de aplicativos de entrega durante a pandemia e o aumento nas vendas de motocicletas, proprietários de motocicletas são obrigados a levar o seu veículo com mais frequência às oficinas, ocasionando um crescimento no número de oficinas mecânicas que realizam diversos tipos de serviços de reparo e manutenção. Houve crescimento de pessoas contratadas, tanto para realizar estas manutenções, como também para trabalhar como motoboy. Esse fator acabou contribuindo para uma recolocação de pessoas no mercado de trabalho, diminuindo o déficit de pessoas desempregadas (SBT BRASIL, 2022).

Com o aumento da concorrência entre as oficinas, a tecnologia pode se tornar um destaque na gestão de grandes, médias e pequenas empresas, trazendo inúmeros benefícios, automatizando tarefas comuns e minimizando possíveis erros. A tecnologia pode atuar em vários pontos do negócio, como por exemplo: Controle de estoque, controle fiscal, controle de contas e controle de produtos.

Atualmente as oficinas mecânicas de motocicletas são impactadas pela falta de informatização no gerenciamento do seu negócio, desde o controle de estoque, clientes e motos, com isso impactando diretamente na qualidade do serviço e atendimento aos seus clientes. Portanto, ter uma plataforma para fazer este tipo de controle para uma oficina, acaba

resultando em melhorias de velocidade, atendimento, suporte e gestão do seu ambiente de trabalho, aumentando a satisfação de seus clientes.

Em Viamão, por exemplo, há dezenas de oficinas especializadas, que se situam nas mais diversas propriedades, sendo desde lugares grandes até menores, com grandes capacidades e qualidade no atendimento. Entregando um serviço de referência especializada, atribuindo uma excelente qualidade de trabalho. Mesmo com mão de obra especializada, muitas vezes o gerenciamento de uma oficina acaba sendo impactado por ser controlado de uma forma manual e mecânica, isso devido à falta de uma plataforma para auxiliar nos processos do dia a dia.

Sendo assim, esse trabalho de conclusão pergunta se é possível fidelizar clientes através da informatização de alguns serviços prestados por oficinas mecânicas de motocicletas?

Nossa hipótese constitui-se na construção de um sistema para ter todo o gerenciamento necessário para o controle destas informações, além de permitir a comunicação direta com os clientes. Acreditamos que um sistema, baseado em uma aplicação compatível com celular, melhore a interação e a fidelização dos clientes e oficinas, aumentando o rendimento de dinheiro e de tempo, tanto para o dono da oficina como para o cliente.

Com o sistema se saberá quem é o cliente, qual motocicleta ele possui e o que ele fez em todas as vezes em que ele visitou a oficina. Com essas informações a mecânica poderá ofertar ou notificar a necessidade de realização de manutenções preventivas, vender peças com trocas contínuas, entre outros serviços, que aumentarão o faturamento da oficina.

Isso resolve o problema de mecânicas tradicionais que escrevem tudo no papel, perdendo a maioria do seu tempo procurando onde foram feitas as anotações. Esses dados ficam salvos na nuvem sem se preocupar em perder informações, isso quer dizer que só será necessário um navegador para acessar a internet.

Inicialmente o sistema abrangerá apenas oficinas mecânicas em que o mecânico autorize a implementação do sistema e que seja de pequeno porte, ou seja, aquelas que tem uma baixa cartela de clientes fixos e que não tenham uma alta rotatividade de estoque. Com

isso o sistema poderá ter um produto mínimo viável (MVP) para poder suprir as necessidades básicas, como o gerenciamento de motocicletas, estoque e clientes, e a partir disso, deverá ser analisado o quão este sistema poderá funcionar em oficinas de portes maiores.

Para consolidarmos os dados deste trabalho, será necessário um estudo de caso para comprovar que o caminho que está sendo seguido está de acordo com o mercado de oficinas mecânicas. Assim, foi escolhido uma oficina de motocicletas para que não seja apenas um usuário de teste para a plataforma, mas sim que ajude a orientar a construção do sistema, para torná-lo fácil de utilizar e que tenha as funcionalidades mínimas para que uma oficina mecânica possa aproveitar suas vantagens.

A oficina que servirá de estudo de caso está situada na cidade de Viamão, cidade vizinha da capital do Rio Grande do Sul, Porto Alegre. Uma oficina de pequeno porte, com apenas três funcionários que trabalham no local, fazendo manutenções, reparos e a venda de peças de reposição para seus clientes. No momento todo seu gerenciamento de peças, clientes e motos, é apenas feito em anotações físicas e manuais, que acaba dependendo de um alto controle na organização e muita atenção para que certos serviços não acabem se misturando com o de outros clientes e que se percam durante o passar do tempo. Com essa questão, esta oficina abriu suas portas para usarmos o seu modelo de negócio da oficina, com isso se tem o foco nas funcionalidades necessárias para que esta oficina se desligue totalmente de algo físico e que não possui algum backup ou até mesmo a segurança dos dados salvos.

1.1 Objetivo Geral

Desenvolver um sistema de gerenciamento de oficinas mecânicas de motocicletas, contando com suporte para o gerenciamento de peças, serviços e rotinas de fidelização de seus clientes.

1.2 Objetivos Específicos

1. Aumento da fidelização de clientes para a oficina.
2. Aumentar o faturamento.
3. Reduzir o tempo buscas manuais do estoque de produtos, motos e clientes.
4. Melhorar a geração e autorização de orçamentos.
5. Formação no curso Análise e Desenvolvimento de Sistemas.

O restante deste trabalho de conclusão de curso está organizado em uma estrutura, que segue o seguinte formato: o capítulo 2 apresenta os trabalhos relacionados; o capítulo 3 apresenta os levantamentos de requisitos; o capítulo 4 cita o modelo conceitual; seguido das referências e os apêndices deste trabalho.

2 TRABALHOS RELACIONADOS

Para contextualização do assunto que o trabalho irá abordar, foi realizado um estudo para encontrar referências bibliográficas. A revisão da literatura buscou trabalhos relacionados sobre plataformas de gerenciamento para oficinas mecânicas, procurando relacionar estes trabalhos com a questão de pesquisa elaborada para este trabalho. A busca foi realizada no sistema de pesquisa Google Acadêmico.

O primeiro trabalho foi elaborado com o objetivo de desenvolver um *software* para a empresa Auto Mecânica Fortaleza, localizada na cidade de Curitiba, estado do Paraná, a qual atua no ramo da manutenção mecânica em veículos leves, que venha a facilitar de forma efetiva o gerenciamento daquele negócio. Ou seja, possibilitar ao usuário do sistema cadastrar dados de seus clientes, colaboradores, parceiros, fornecedores e veículos recolhidos para manutenção, além de permitir o controle dos serviços realizados e de pagamentos recebidos. Além disso, haverá a oportunidade de ter os dados armazenados num banco de dados e poder realizar o *backup* dele. Assim, com este trabalho eles pretendiam demonstrar que pequenas oficinas mecânicas, diferentemente de grandes centros automotivos, também podem dispor de um sistema de gestão prático (pela facilidade de interação com o usuário), gratuito ou de custo extremamente reduzido (pelo uso de tecnologias *open source*) e eficiente (pela máxima de se fazer mais com menos). Ou seja, mais controle, qualidade e produtividade no negócio com menos pessoal, tempo, tecnologia e recursos financeiros. Já no segundo trabalho selecionado é apresentado o desenvolvimento de um *software* para gerenciamento dos serviços oferecidos ou prestados em uma oficina mecânica de pequeno porte. Esse *software* possui funcionalidades para controle dos serviços prestados, do cadastro de clientes e respectivos veículos pertencentes, dos relatórios, entre outros. Ambos os trabalhos foram selecionados dentre tantos outros pela sua similaridade com o assunto que será abordado no trabalho de conclusão a ser desenvolvido.

2.1 GESTMEC: SISTEMA DE GESTÃO DE OFICINA MECÂNICA

O primeiro trabalho foi elaborado com o objetivo de desenvolver um *software* para a empresa Auto Mecânica Fortaleza, localizada na cidade de Curitiba, estado do Paraná, a qual atua no ramo da manutenção mecânica em veículos leves.

O *software* facilita de forma efetiva o gerenciamento do negócio, possibilitando ao usuário cadastrar dados de seus clientes, colaboradores, parceiros, fornecedores e veículos recolhidos para manutenção, além de permitir o controle dos serviços realizados e de pagamentos recebidos. Além disso, a solução permite que os dados sejam armazenados em um banco de dados, podendo realizar os *backups* deles.

Com este trabalho os autores demonstram que pequenas oficinas mecânicas, diferentemente de grandes centros automotivos, também podem dispor de um sistema de gestão prático (pela facilidade de interação com o usuário), gratuito ou de custo extremamente reduzido (pelo uso de tecnologias open source) e eficiente (pela máxima de se fazer mais com menos). Ou seja, mais controle, qualidade e produtividade no negócio com menos pessoal, tempo, tecnologia e recursos financeiros.

O sistema chama-se "GESTMEC" e foi disponibilizado somente para a plataforma *desktop*, sendo necessário apenas um computador para acessar, não necessitando de conexão com a internet.

O sistema foi desenvolvido na linguagem Java, tanto para interface como para gerenciamento dos dados, e para persistência dos dados foi utilizado o banco de dados relacional MySQL. O sistema terá como funcionalidades o gerenciamento de clientes, colaboradores, parceiros, fornecedores e veículos.

Como resultado do desenvolvimento deste *software*, foi de extrema importância para o aprendizado do aluno responsável, além de ser um software que será altamente utilizado dentro da oficina mecânica, com segurança, gerenciamento de entidades e *backups* de dados. Para os próximos passos de desenvolvimento da plataforma seria a migração para uma plataforma web e *mobile*, com outras funcionalidades como o agendamento de atendimentos e a solicitação de orçamentos.

2.2 SISTEMA INFORMATIZADO PARA MECÂNICAS

Já no segundo trabalho selecionado é apresentado o desenvolvimento de um *software* para gerenciamento dos serviços oferecidos ou prestados em uma oficina mecânica de pequeno porte. Esse *software* possui funcionalidades para controle dos serviços prestados, do cadastro de clientes e respectivos veículos pertencentes, dos relatórios, entre outros.

O projeto apresenta um sistema para o gerenciamento de oficinas mecânicas de pequeno porte, conhecido como "S.I.M". Ele está disponível somente para a plataforma web, necessitando ter acesso a um computador e conseqüentemente a conexão com a internet.

O sistema foi desenvolvido na linguagem C#, para a interface da plataforma junto do HTML e para o gerenciamento dos dados do sistema, e para persistência dos dados foi utilizado o banco de dados relacional MySQL. O sistema tem como funcionalidades o gerenciamento de clientes, veículos, orçamentos, ordens de serviço e emissão de relatórios.

Como resultado do desenvolvimento do projeto mostrou que oficinas de pequeno porte, que ainda trabalham com uma forma tradicional, ou seja, usam de planilhas para o gerenciamento da oficina, que acaba impactando na hora de gerar relatórios, gerenciar clientes e peças, perdem eficiência e tempo, permitindo que a falha e a falta de atenção em outros pontos dentro da oficina acabam sendo afetadas. Portanto o desenvolvimento deste *software* satisfaz todas as necessidades de clientes que possuam oficinas de pequeno porte.

2.3 Comparativo dos trabalhos

Ambos os trabalhos foram selecionados dentre tantos outros pela sua similaridade com o assunto que será abordado neste trabalho de conclusão. A Tabela 1 apresenta um comparativo entre os trabalhos encontrados na literatura e a solução que será proposta neste trabalho de conclusão.

Tabela 1 - Comparativo entre os sistemas

	André Andrade dos Santos, 2014	Fábio Paula de Morais, Huéltton Aparecido da Silva, João Paulo Gomes da Costa, Lucas Cardoso Sarvuchi, 2017	Projeto desenvolvido
Existem sistemas de informação criados para oficinas mecânicas de motocicletas?	sim	sim	sim
Quais plataformas (web, desktop, celular etc.) são empregadas para sistemas de informação criados para oficinas mecânicas de motocicletas?	Desktop	Web	Web/mobile
Quais tecnologias (banco de dados, linguagem de programação etc.) são mais empregadas nos sistemas de informação criados para oficinas mecânicas de motocicletas?	Java e MySQL	C# e MySQL	Javascript, Postgres
Qual o público atendido com as soluções implementadas para oficinas mecânicas de motocicletas?	Oficina mecânica de linhas leves	Oficina mecânica de pequeno porte	Oficina mecânica de pequeno porte

Tabela 2 - Comparativo entre os sistemas

	André Andrade dos Santos, 2014	Fábio Paula de Moraes, Huélton Aparecido da Silva, João Paulo Gomes da Costa, Lucas Cardoso Sarvuchi, 2017	Projeto a ser desenvolvido
Gerenciamento de clientes	sim	sim	sim
Gerenciamento de veículos	sim	sim	Sim
Gerenciamento de serviços	não	sim	sim

Fonte: Elaborado pelo autor (2022).

3 LEVANTAMENTO DE REQUISITOS

Para levantamento dos requisitos, foram construídos dois formulários utilizando a ferramenta Google Forms. O primeiro formulário, foi para coletar informações dos clientes, ou seja, usuários das motocicletas. O segundo, destinou-se a coletar informações do proprietário da oficina mecânica. Através das informações obtidas, que foi realizada para os usuários, foi disponibilizado para os alunos do Instituto Federal do Rio Grande do Sul no mês de dezembro de 2021. No entanto, a oficina mecânica recebeu o formulário para conceder os dados e informações relevantes, o formulário ficou disponível para responder pelo período de duas semanas.

O objetivo dos formulários foi encontrar as funcionalidades necessárias, de acordo com os requisitos da aplicabilidade do sistema. O formulário dos usuários obteve o número de 17 respostas, porém, os resultados obtidos da oficina mecânica foram de apenas uma resposta, em comparação ao proprietário da oficina usada no exemplo do estudo de caso.

3.1 Resultado da coleta de informações na oficina

Tabela 3 - Resultado das perguntas feitas a oficina

Quantidade de funcionários na oficina	3
Já utiliza de alguma ferramenta para o gerenciamento?	Não
Utilizaria de alguma ferramenta?	Sim
Que dispositivos costuma utilizar no dia a dia?	Smartphone
Que funcionalidades deverá ter no sistema?	Gerenciamento de peças, motos, financeiro, clientes e geração de relatórios.

Fonte: Elaborado pelo autor (2022).

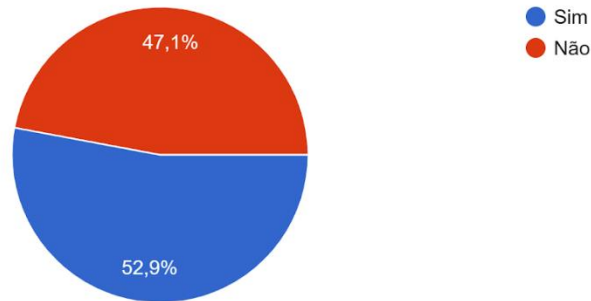
Com base na Tabela 2, pode se analisar que é uma oficina de pequeno porte e que ainda não possui alguma ferramenta para automatizar os processos manuais que ainda são realizados. A utilização do software deste projeto será através de um smartphone e um navegador *web*. E de funcionalidades mínimas para a utilização do sistema: gerenciamento de peças, motos, financeiro, clientes e geração de relatórios.

3.2 Resultado da coleta de informações dos usuários

Figura 1- Proprietários de motocicletas

Possui uma motocicleta?

17 respostas

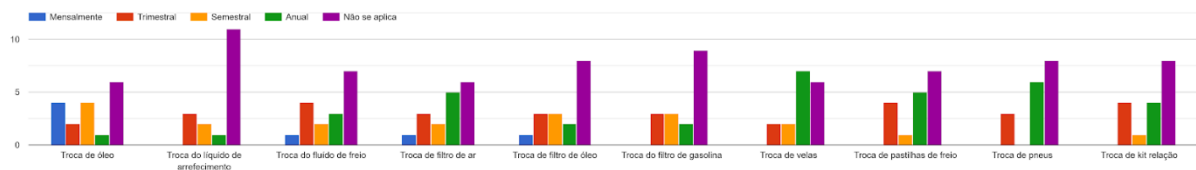


Fonte: Elaborado pelo autor (2022).

Com base no gráfico, Figura 1, é apresentado que o número de usuários que utilizam motocicletas se sobressai em mais de 5% em relação aos usuários que não utilizam.

Figura 2 - Procedimentos

Com que frequência realiza os procedimentos abaixo



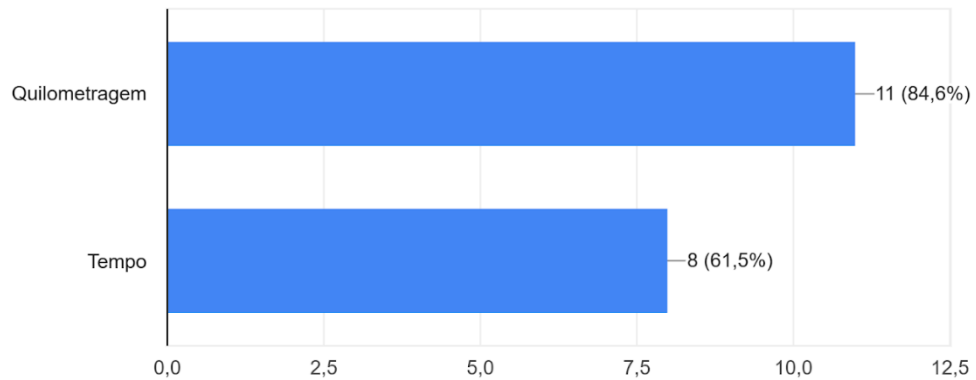
Fonte: Elaborado pelo autor (2022).

No que diz respeito ao gráfico Figura 2, é apresentado que cada manutenção realizada em uma motocicleta é realizada em diferentes tempos, porém o tempo por trimestre é o que mais se mantém estável em todos os procedimentos.

Figura 3 - Manutenção periódica

Costuma realizar as manutenções acima por:

13 respostas



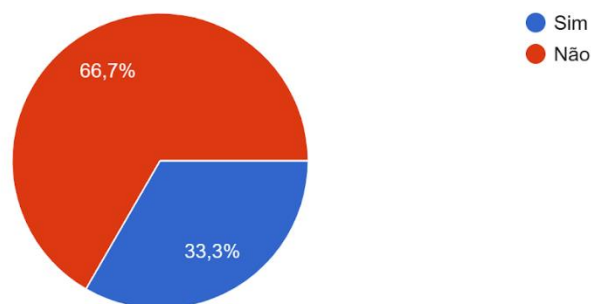
Fonte: Elaborado pelo autor (2022).

No que diz respeito ao gráfico Figura 3, é apresentado que os usuários que possuem motocicleta, realizam a manutenção quando atingem uma certa quilometragem, ao invés do tempo de uso do veículo.

Figura 4 - Prevenção

Realiza manutenção preventiva na parte elétrica da motocicleta?

15 respostas

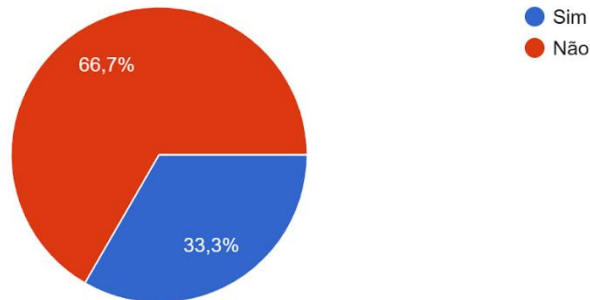


Fonte: Elaborado pelo autor (2022).

Mais do 60% das pessoas não realizam uma manutenção preventiva na parte elétrica da sua motocicleta, de acordo com o gráfico, Figura 4.

Figura 5 - Oficinas

Você frequenta mais de uma oficina?
15 respostas

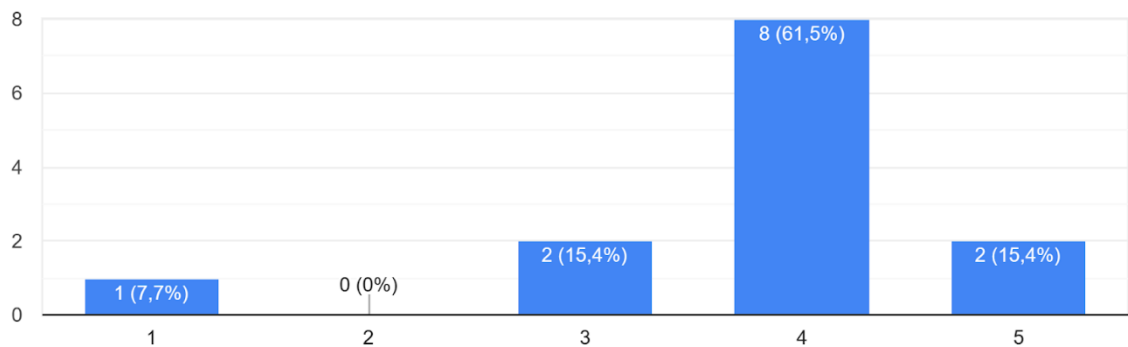


Fonte: Elaborado pelo autor (2022).

Mais de 60% dos usuários não frequentam mais de uma oficina, isso muitas vezes por causa da confiança e qualidade do local, Figura 5.

Figura 6 - Melhor oficina

Avalie a melhor oficina que costuma frequentar
13 respostas

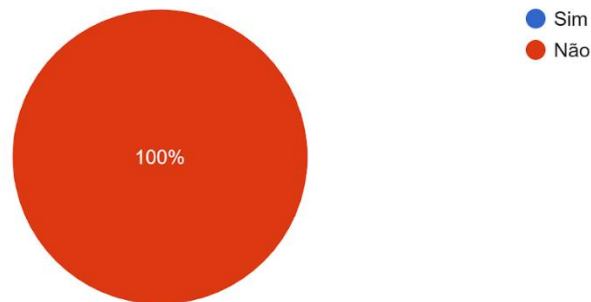


Fonte: Elaborado pelo autor (2022).

Mesmo sendo a oficina que o usuário mais frequenta, ela acaba recebendo no máximo uma nota 8 e ficando com mais de 60% dos votos de acordo com a Figura 6.

Figura 7 - Aplicativos para gerenciamento

Você utiliza de algum aplicativo para gerenciar sua moto?(controle de manutenção, histórico)
14 respostas

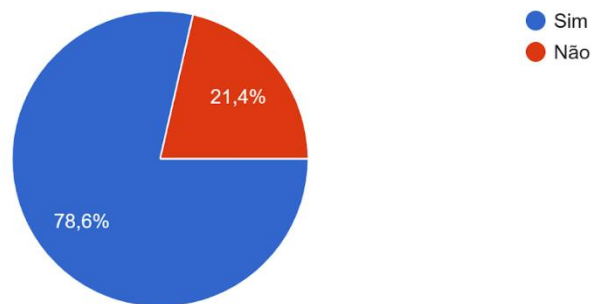


Fonte: Elaborado pelo autor (2022).

No que diz respeito ao gráfico Figura 7, é apresentado que todos os usuários não utilizam de algum aplicativo para fazer o gerenciamento de sua motocicleta.

Figura 8 - Utilização

Você utilizaria de algum aplicativo?
14 respostas



Fonte: Elaborado pelo autor (2022).

De acordo com a Figura 8, é apresentado que quase 80% das pessoas, seriam usuários de algum aplicativo para o gerenciamento de sua motocicleta.

Figura 9 - Funcionalidades do sistema

Fonte: Elaborado pelo autor (2022).

No que diz respeito ao gráfico Figura 9, são apresentadas funcionalidades em que um aplicativo (teria de auxiliar um usuário) na gestão de sua motocicleta, e as funcionalidades com mais votos são: relatório de gastos em gasolina, notificações de troca de óleo e a visualização do histórico de manutenções.

Com base nos gráficos, apontamentos feitos sobre a pesquisa realizada em uma oficina, juntamente com seus usuários. Analisou-se que os resultados obtidos da oficina constariam como um possível sucesso a utilização de uma ferramenta, o qual tivesse a capacidade de ser utilizada em um smartphone, junto com funcionalidades para o gerenciamento de peças, motocicletas, parte financeira, clientes e relatórios em geral. Através dos resultados da pesquisa dos usuários, o relato foi de que eles utilizaram de uma aplicação, a qual possuía mais como foco o aviso prévio de certas manutenções, troca de óleo, relatório de gastos de gasolina e histórico de manutenções da motocicleta.

No Gráfico 14 são apresentadas todas as respostas das possíveis soluções que visam poder integrar no sistema da oficina mecânica. Das 14 respostas das funcionalidades apuradas, a pesquisa foi realizada e revelou que, a notificação de troca de óleo, relatório de gastos em gasolina e histórico de manutenção teve taxa de 92,9% dos dados. As taxas de 71,4% representaram as funcionalidades de agendar revisão e relatório de gastos em manutenção, com as funcionalidades de acompanhar o andamento da manutenção e realizar

o pagamento da manutenção a taxa foi de 64,3%. Uma parcela de 57,1 % prefere as notificações de acompanhar o andamento da manutenção e autorizar orçamento.

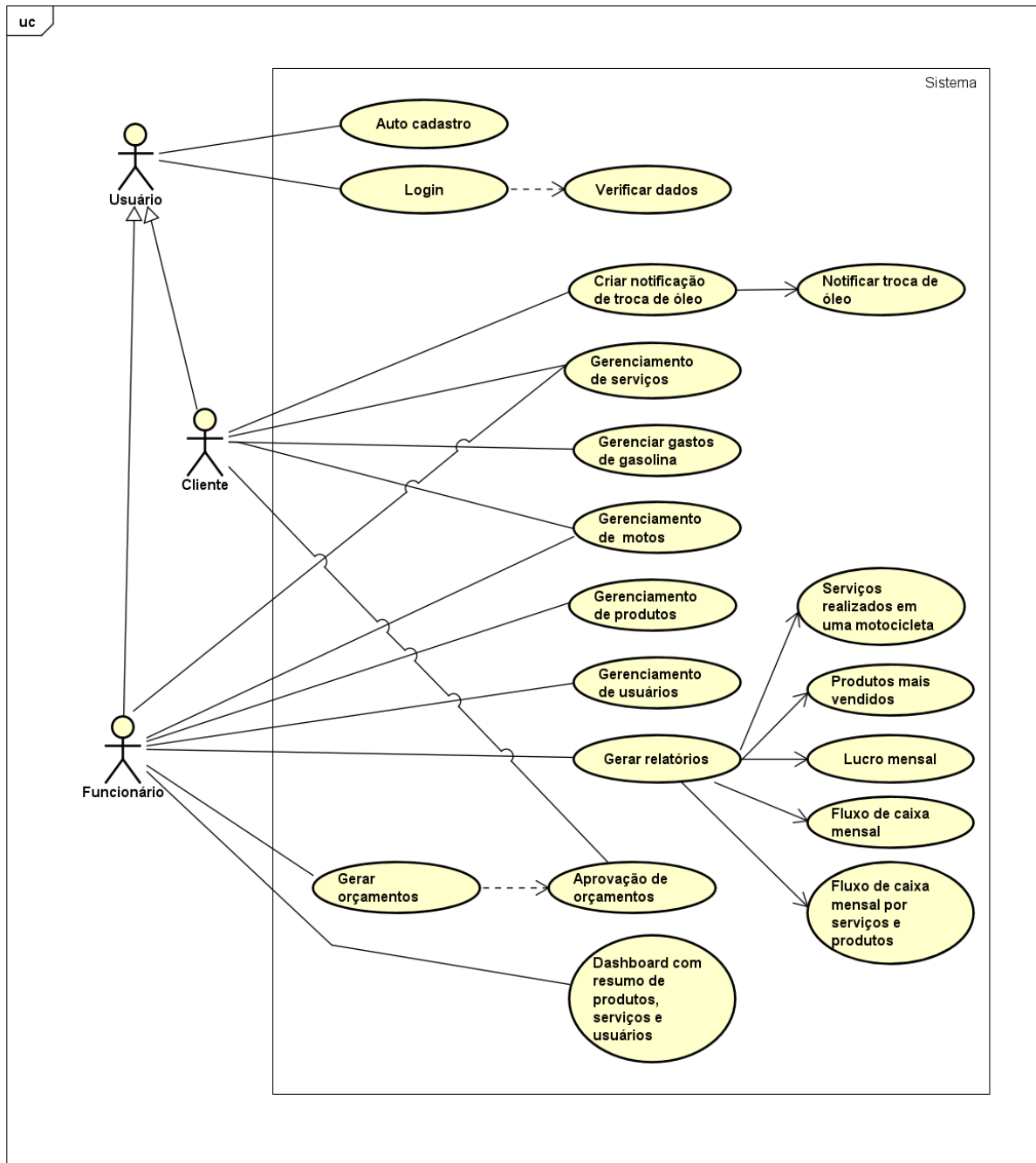
4 SOLUÇÃO CONCEITUAL

O levantamento de requisitos funcionais e não funcionais, tem como objetivo analisar as funcionalidades que um sistema terá, para entender melhor o posicionamento destes requisitos. A modelagem do diagrama de casos de uso será representada como uma associação UML, onde é exibido os diagramas e demais informações, deixando todos os envolvidos no projeto, cientes de suas aplicabilidades.

4.1 Diagrama de casos de uso

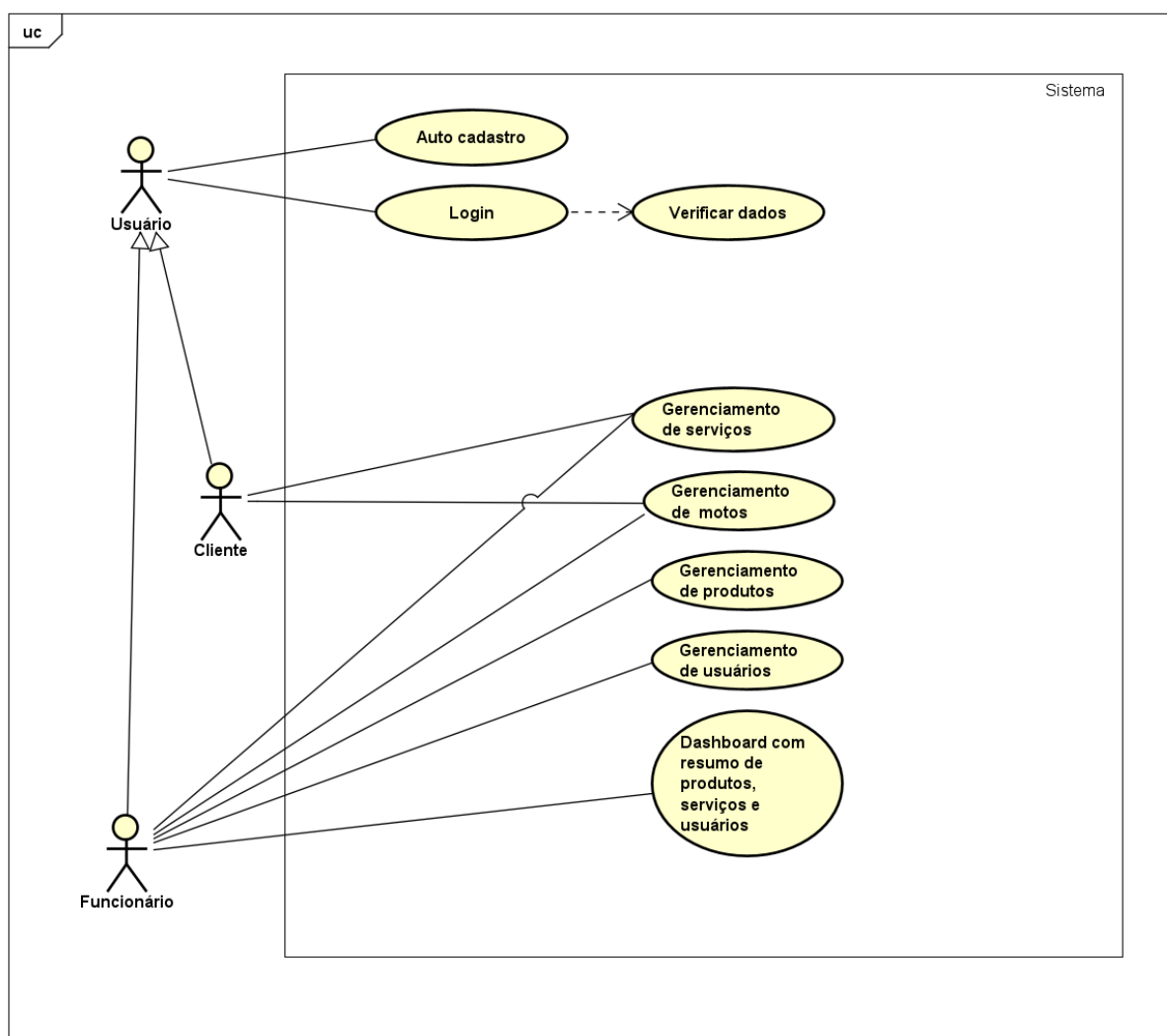
Na linguagem de modelagem unificada (UML), o diagrama de caso de uso demonstra as interações que um usuário poderá realizar em um sistema. Este diagrama não possui detalhes precisos de fluxo, ele tem como foco ampliar a visão sobre os casos de uso, atores e o sistema. Então, ele será utilizado para mapear os cenários e funcionalidades do sistema.

Figura 10 - Diagrama Casos de uso



Fonte: Elaborado pelo autor (2023).

Figura 11 - Diagrama Casos de uso



Fonte: Elaborado pelo autor (2023).

Com base na Figura 10, o diagrama dos casos de usos, demonstra todas as funcionalidades deste sistema, que foi estruturado com base nos formulários e outras funcionalidades que demonstram que o projeto após ser finalizado poderá dar continuidade. Porém verificando que o tempo para o desenvolvimento do software é limitado, definiu-se que seria necessário reduzir o número de funcionalidades, isso levando em consideração as funcionalidades base dos requisitos funcionais que a oficina de estudo de caso apresentou. Por meio de informações dos casos de uso, concretizou que algumas das ações dos usuários no sistema, possuem o mesmo tipo de execução. Portanto, foi criado um ator que tivesse atributos em comum, e com isso se divergiu em mais dois atores, o cliente e o funcionário, que mesmo que tenham ações diferentes, eles são reconhecidos como atores usuários dentro do sistema.

Conforme a Figura 11, ao definir as funcionalidades também foi realizado um estudo mais preciso. Dessa maneira, pode se verificar e entender o funcionamento real de uma oficina mecânica, a partir dos seus serviços e suas regras de negócio. Igualmente foi levado em consideração o fator tempo ao desenvolver todas as funcionalidades anteriormente citadas, sendo necessário escolher alguns requisitos que seriam utilizados.

Após a construção do diagrama, os casos de uso foram detalhados textualmente apresentado o fluxo de execução da funcionalidade e as regras de negócio necessárias para sua implementação. A Tabela 2 mostra um exemplo da descrição do caso de uso ‘Realizar Cadastro’. A documentação contempla dos demais casos de uso do sistema de intranet estão disponíveis no Apêndice A.

Tabela 4 - Detalhamento do caso de uso Realizar Cadastro

Nome do Caso de Uso	Cadastrar serviço
Resumo (Objetivo)	Permitir que o usuário cadastre seus(s) serviço(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de serviços.</p> <p>P2. O usuário abre o formulário de cadastro.</p> <p>P3. O sistema apresenta o formulário.</p> <p>P4. O usuário insere os dados.</p> <p>P5. O usuário seleciona a opção de tela “Salvar”.</p> <p>P6. O sistema valida e salva os dados.</p> <p>P7. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	<p>FA1. No P6 quando usuário não informa os dados obrigatórios</p> <p>A. O sistema exibe a seguinte msg “Informe os campos obrigatórios”</p> <p>B. O sistema retorna ao P2</p>
Pós-condições	O sistema redireciona para a listagem dos serviços.

Tabela 5 - Detalhamento do caso de uso Realizar Cadastro

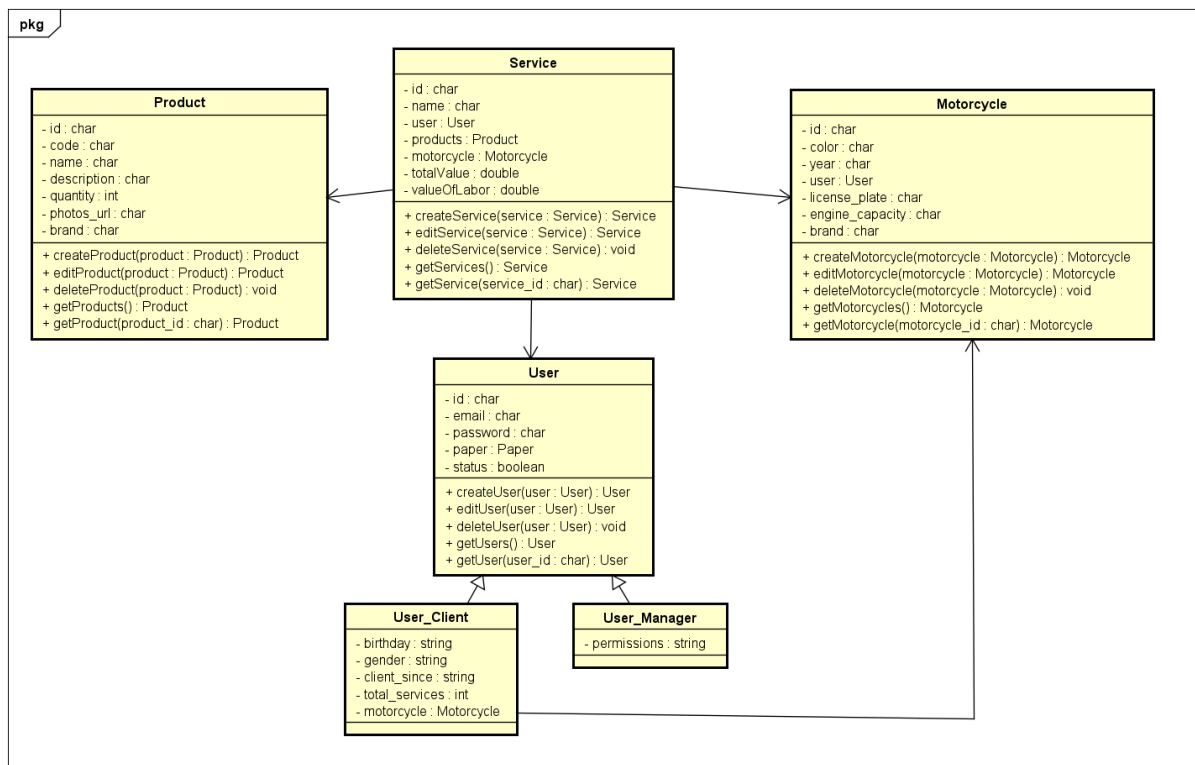
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	Nome: campo alfanumérico. Campo obrigatório Descrição: campo alfanumérico. Campo obrigatório
Observações	-

Fonte: Elaborado pelo autor (2022).

4.2 Diagrama de Classes

A partir das descrições dos casos de uso, foi idealizado o diagrama de classes da aplicação. A Figura 12 mostra o diagrama de classes implementado no sistema.

Figura 12 - Diagrama de Classes (Produto, Serviço, Motocicleta, Usuário, Gerente, Cliente, Papel, Permissão).

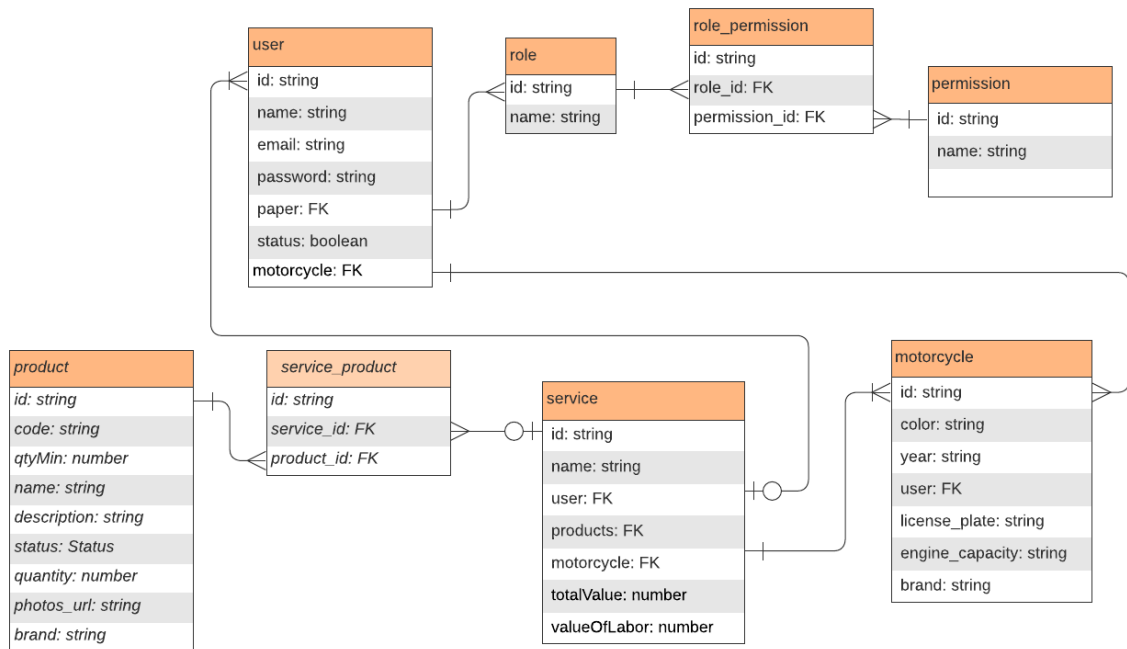


Fonte: Elaborado pelo autor (2023).

4.3 Diagrama Entidade Relacionamento

Este diagrama refere-se as ligações definidas para as classes mapeadas, a Figura 13 demonstra o relacionamento entre as tabelas do banco de dados.

Figura 13 - Diagrama ER (Produto, Serviço, Motocicleta, Usuário, Papel, Status, Permissão).



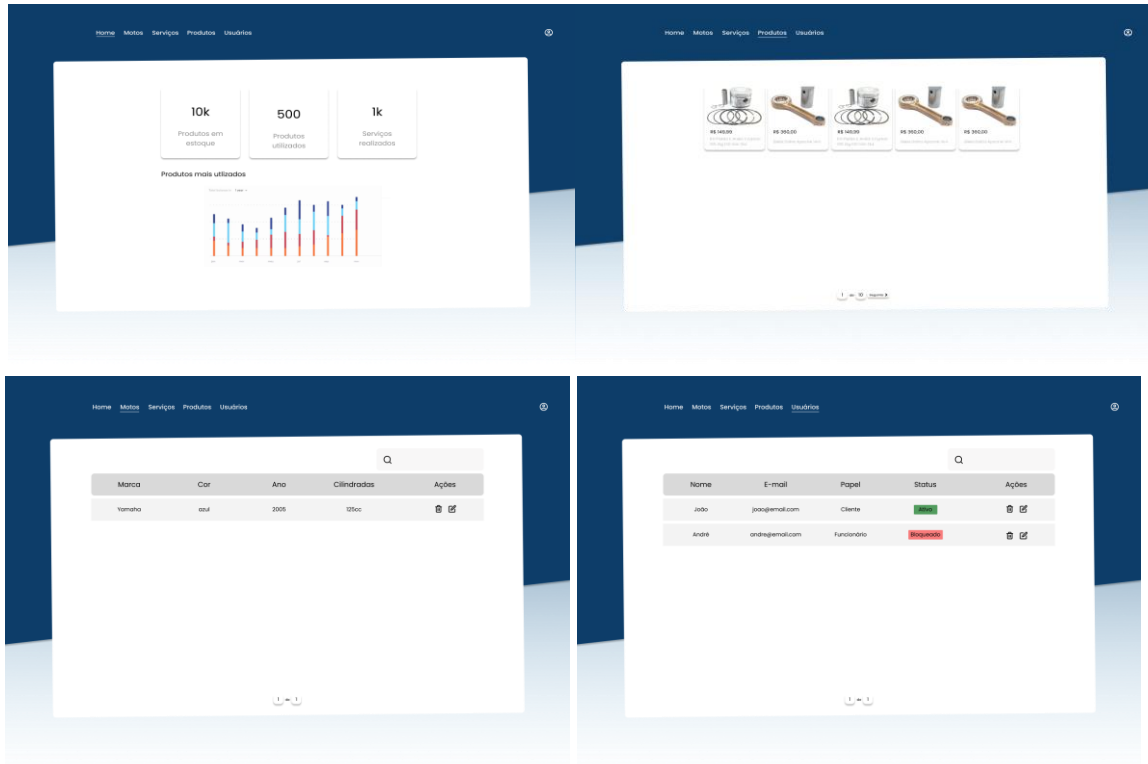
Fonte: Elaborado pelo autor (2023).

4.4 Prototipagem

A fim de validar as funcionalidades e visual do sistema, sem ter a necessidade a necessidade desenvolver um projeto em HTML, impactando no aumento do tempo, foi realizado a prototipação do sistema em uma plataforma chamada Figma, com isso foi possível construir todas as telas principais que o sistema irá possuir.

A seguir é conferido as telas construídas Figura 14, dando noção de como supostamente o sistema ficará no final de seu desenvolvimento final.

Figura 14 - Protótipos

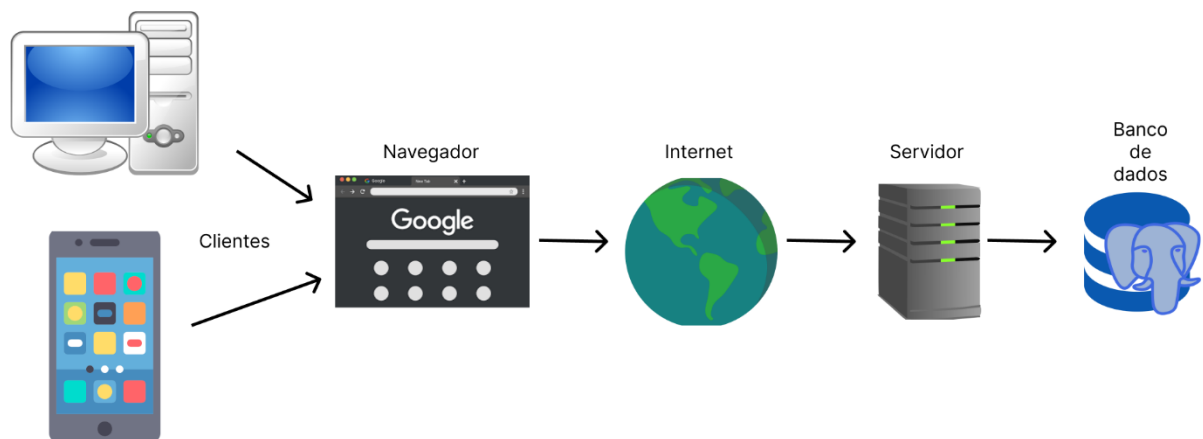


Fonte: Elaborado pelo autor (2023).

5 DESENVOLVIMENTO

Devido ao cenário no aumento do número de utilização de motocicletas no mercado brasileiro em 2022, a utilização de alguma ferramenta para controle de oficinas se demonstrou indispensável, por questões de portabilidade se optou pela arquitetura *web* Figura 15, para o desenvolvimento deste projeto. A escolha do desenvolvimento *web* se deu pelo princípio de estar disponível em qualquer lugar através de um navegador, sem ter a necessidade da instalação do software em alguma máquina. O requisito para uso do sistema é apenas o acesso à internet, com computador ou *smartphone*.

Figura 15 - Arquitetura cliente servidor

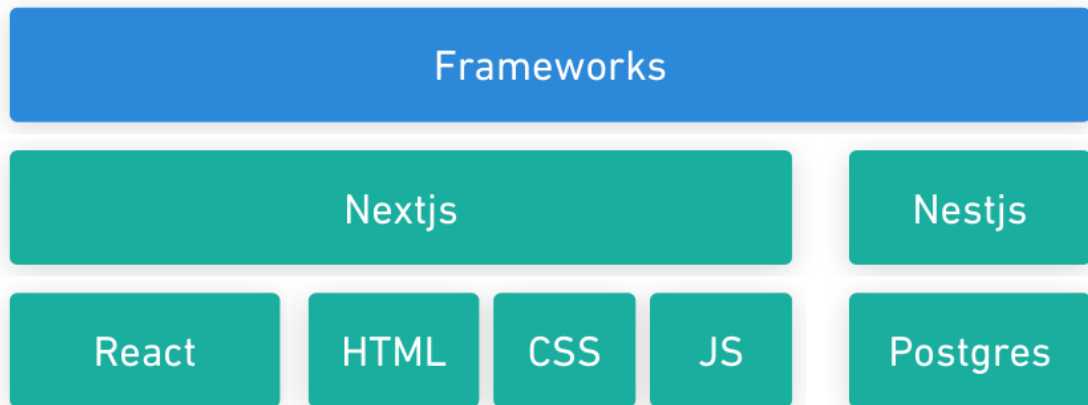


Fonte: Elaborado pelo autor (2023).

Para o desenvolvimento deste projeto se envolveu uma série de tecnologias. A descrição e explicações destas tecnologias se encontram no Apêndice A.

A escolha da linguagem de programação Javascript (A.2) para o desenvolvimento da camada de *backend* e *frontend*, se deu após analisar que sua utilização no mercado é de grande valia para a evolução do sistema, por se tratar de uma linguagem de fácil aprendizado e ser indispensável no desenvolvimento *web*.

Figura 16 - Estrutura da aplicação

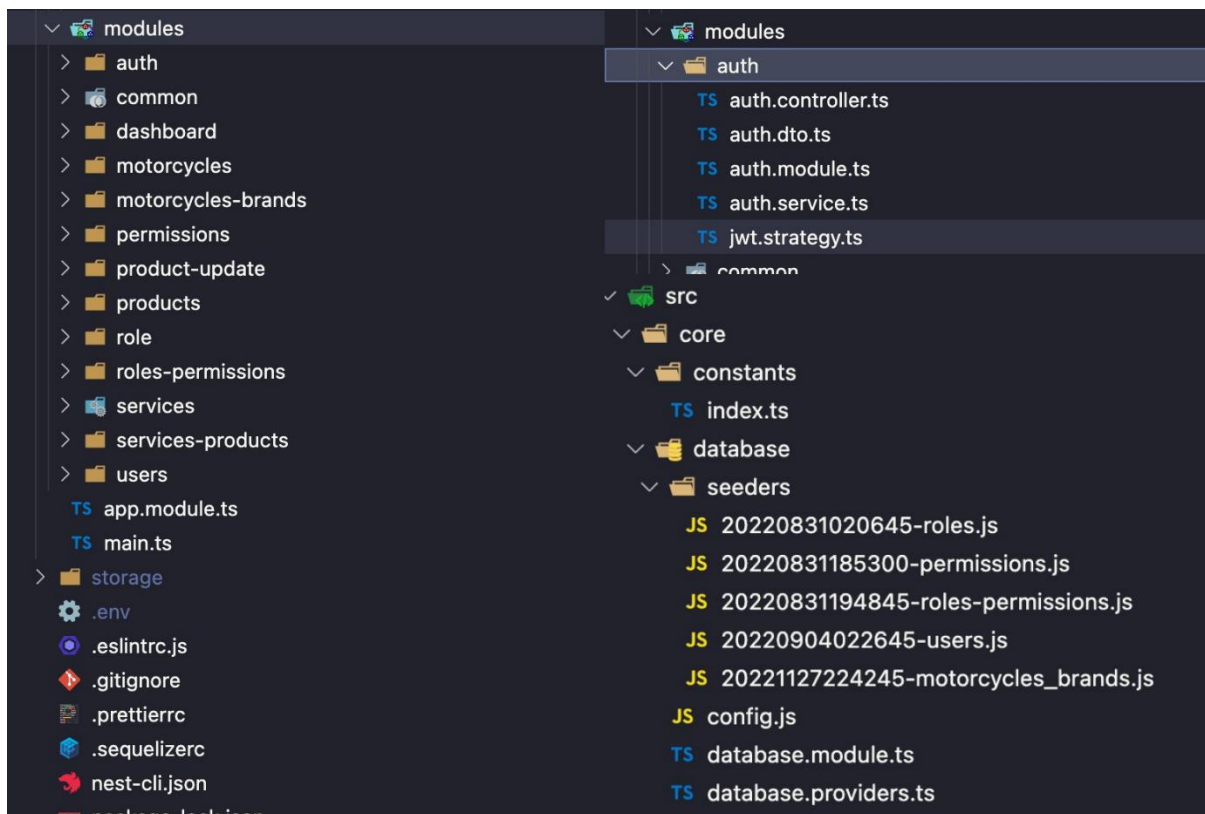


Fonte: Elaborado pelo autor (2023).

Por não ter a necessidade de um alto processamento de dados na camada do backend, foi escolhido como framework, o Nestjs (A.7), seu objetivo é abstrair os obstáculos de estrutura de um projeto, permitindo criar uma aplicação escalável, testável e de fácil manutenção. Além disso, tem baixa curva de aprendizado e em pouco tempo pode ser aplicado na prática.

- A estruturação e organização de pastas do código fonte segue na Figura 17: Na pasta *modules* ficam os serviços responsáveis por cada funcionalidade do sistema, ou seja, por exemplo no module *auth*, fica a responsabilidade pela autenticação do sistema;
- Na pasta *core* ficam as configurações principais, como constantes e dados do banco de dados;
- Na *storage*, ficam as imagens dos produtos que foram anexadas.

Figura 17 - Estrutura de pastas Nestjs



Fonte: Elaborado pelo autor (2023).

O *framework* Nexjs (A.5) foi escolhido por ter em sua base no React (A.4), auxiliando no desenvolvimento de interfaces e melhorando a performance em produção. Sua integração com a Vercel (A.6) facilita colocar em produção o sistema. Por se tratar de um *framework*, ele remove certas configurações de rotas, páginas e consultas ao *backend*, ajudando no desenvolvimento da aplicação.

A estruturação e organização de pastas do código fonte da camada do *frontend* é apresentada na Figura 18. Nesta camada os nomes e responsabilidades mudam. Na pasta *components*, é onde ficam os elementos que serão utilizados e reaproveitados em alguma tela. Um exemplo disso, é um botão onde ele vai ter o comportamento e padrão em todas as telas, não tendo a necessidade de repeti-los ao longo do código. Na pasta *pages* se encontram as páginas do sistema, desde o login, até a página de listagem dos produtos. Nas demais pastas como *services*, ficam funcionalidades como a configuração de requisição com o *backend*. Na pasta *styles*, ficam o estilo que será aplicado na interface, cores, fontes, tamanhos.

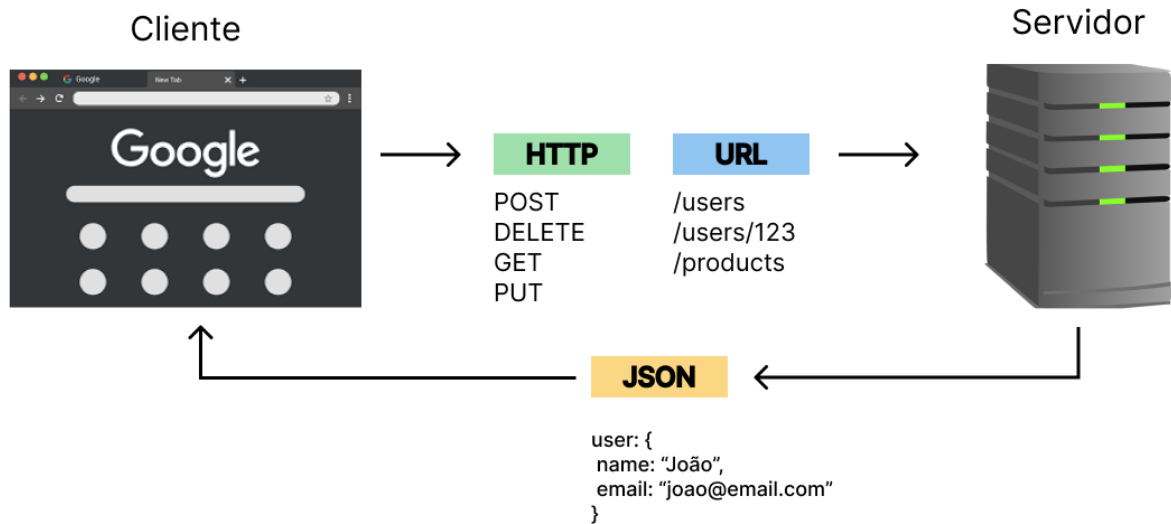
Figura 18 - Estrutura de pastas Nextjs



Fonte: Elaborado pelo autor (2023).

Para realizar a comunicação entre estas duas camadas e trocas informações, do tipo autenticação do usuário, listagem de dados, validações e regras de negócio, foi seguido o padrão REST API, conforme mostra a Figura 19. REST API é um padrão de desenvolvimento que consiste em requisição do cliente e resposta do servidor, porém seguindo certas regras. O cliente, que é a camada do *frontend*, quando for realizar uma requisição ao servidor deve seguir um padrão, acessando uma URL que solicita uma informação e como ele quer acessar a informação. Ou seja, o *frontend* faz uma requisição do tipo GET para a URL /users e o servidor retorna a lista dos usuários cadastrados no banco de dados. Na camada do *backend*, a aplicação responde a requisição usando o formato de dados chamado JSON (Notação de objeto Javascript).

Figura 19 - Estrutura REST API



Fonte: Elaborado pelo autor (2023).

5.1 Implementação da camada de modelo

Na sequência são apresentados fragmentos de códigos, demonstrando o fluxo de desenvolvimento empregado pelo *Nestjs* (A.7). Fragmento de classe de modelo Usuário Figura 20. Desta forma, representamos a tabela do banco de dados através de uma classe Javascript.

E para a persistência deste modelo no banco de dados, se deu a escolha do Postgres (A.8), um banco de dados de código aberto, ou seja, sua utilização não tem preço, além disso é um banco relacional que se identifica com a necessidade do projeto.

Figura 20 - Modelo usuário

```
@Table({ tableName: 'users' })
export class UserEntity extends AbstractEntity {
  @Column({ unique: true })
  @IsEmail()
  email: string;

  @Column
  @Exclude()
  password: string;

  @Column
  username: string;

  @Column
  status: boolean;

  @ForeignKey(() => RolesEntity)
  @Column
  role_id: number;

  @BelongsTo(() => RolesEntity)
  role: RolesEntity;

  @BeforeCreate
  static async hashPasswordBeforeUpdate(user: UserEntity) {
    user.password = await hash(user.password, 10);
  }

  comparePassword(attempt: string) {
    return compareSync(attempt, this.password);
  }

  toJSON() {
    return instanceToPlain(this);
  }
}
```

Fonte: Elaborado pelo autor (2023).

5.2 Implementação da camada de controle

O fragmento, Figura 21, demonstra o arquivo de rotas da backend, seguindo o padrão REST API. As rotas são responsáveis por receberem os dados de uma requisição, transportarem para uma camada de serviço, Figura 22 e responderem com o dado que foi tratado neste serviço.

Figura 21 - Rotas de usuários

```
@Controller('users')
export class UsersController {
  constructor(private userService: UsersService) {}

  @Get()
  @UseGuards(AuthGuard())
  findAll() {
    return this.userService.findAll();
  }

  @Get('/:id')
  @UseGuards(AuthGuard())
  findOne(@Param('id') id: string) {
    return this.userService.findOne(+id);
  }

  @Put('/:id')
  @UseGuards(AuthGuard())
  update(
    @Param('id') userId: number,
    @Body(ValidationPipe)
    data: UpdateUserDTO,
  ) {
    return this.userService.updateUser(+userId, data);
  }
}
```

Fonte: Elaborado pelo autor (2023).

Figura 22 - Serviço do usuário

```
@Injectable()
export class UsersService {
  constructor(
    @Inject(USERS_REPOSITORY)
    private userRepository: typeof UserEntity,
  ) {}

  async findAll(): Promise<UserEntity[]> {
    return await this.userRepository.findAll<UserEntity>();
  }

  async findOne(userId: number): Promise<UserEntity> { ...
  }

  updateUser(userId: number, data: UpdateUserDTO) { ...
  }

  async createUser(data: CreateUserDTO): Promise<UserEntity> { ...
  }

  async deleteUser(userId: number) { ...
  }
}
```

Fonte: Elaborado pelo autor (2023).

5.3 Implementação da camada de visualização

Representação da página principal do sistema, a dashboard Figura 23. Neste arquivo é construído a interface da dashboard com os dados que são requisitados ao backend. Para a construção dos elementos foi utilizado *Styled Components* (A.10), é possível perceber a utilização de componentes, isso devido ao modo de desenvolvimento do *React* (A.4).

Figura 23 - Página principal

```

const Index = ({ session, dashboard }: HomeTemplateProps) => {
  ChartJS.register(ArcElement, Tooltip, Legend, ChartDataLabels);

  const { ...
} = dashboard;

  const data = { ...
};

  return (
    <Base session={session}>
      <Container>
        <S.Wrapper>
          <S.Cards>
            <CardDashboard
              description="Produtos em estoque"
              title={totalProducts > 1000 ? `${totalProducts}k` : totalProducts}
            />
            <CardDashboard
              description="Produtos utilizados"
              title={
                totalProductsUsed > 1000
                  ? `${totalProductsUsed}k`
                  : totalProductsUsed
              }
            />
            <CardDashboard
              description="Serviços realizados"
              title={totalServices > 1000 ? `${totalServices}k` : totalServices}
            />
          </S.Cards>
          <S.WrapperDataGraphics>
            <S.Graphic>
              <S.GraphicTitle>Produtos mais utilizados</S.GraphicTitle>
              <Pie data={data} />
            </S.Graphic>
            <S.ListUsers>
              <div>
                <S.GraphicTitle>Últimos usuários cadastrados</S.GraphicTitle>
                <TableUsers data={lastUsers} />
              </div>
              <div style={{ marginTop: "2rem" }}>
                <S.GraphicTitle>Produtos com baixa em estoque</S.GraphicTitle>
                <TableStockProducts data={lowStockProducts} />
              </div>
            </S.ListUsers>
          </S.WrapperDataGraphics>
        </S.Wrapper>
      </Container>
    </Base>
  );
};

```

Fonte: Elaborado pelo autor (2023).

5.4 Execução do sistema

Nesta seção é apresentado o funcionamento de cada tela dentro do sistema. Demonstrando a autenticação, *dashboard*, listagem de motos, produtos, serviços e usuários.

Tela de *login*, Figura 24, para acesso ao sistema;

Figura 24 - Tela de login

Login

Email

Senha

Entrar

Não possui uma conta? [Cadastrar](#)

Fonte: Elaborado pelo autor (2023).

Tela de *cadastro* Figura 25, esta página é utilizada para clientes do sistema que ainda não estão previamente cadastrado no sistema, se cadastrarem.

Figura 25 - Tela de cadastro

Cadastrar

Nome

Email

Senha

Confirmação de senha

Entrar

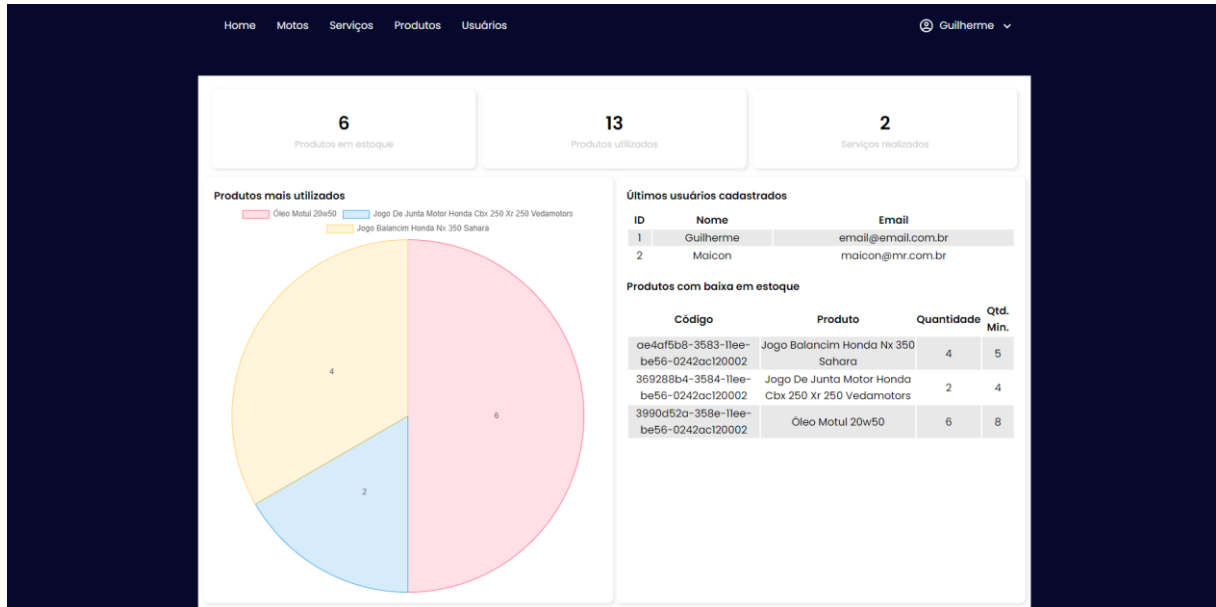
Já possui uma conta? [Login](#)

Fonte: Elaborado pelo autor (2023).

Após realizar o login no sistema, o administrador verá como sua primeira página uma dashboard, onde terá o resumo de certos dados cadastrados Figura 26, como a quantidade de produtos em estoque, serviços realizados, produtos com baixa em estoque. Esses dados são

relevantes para o administrador por não ter a necessidade de ficar navegando em outras páginas.

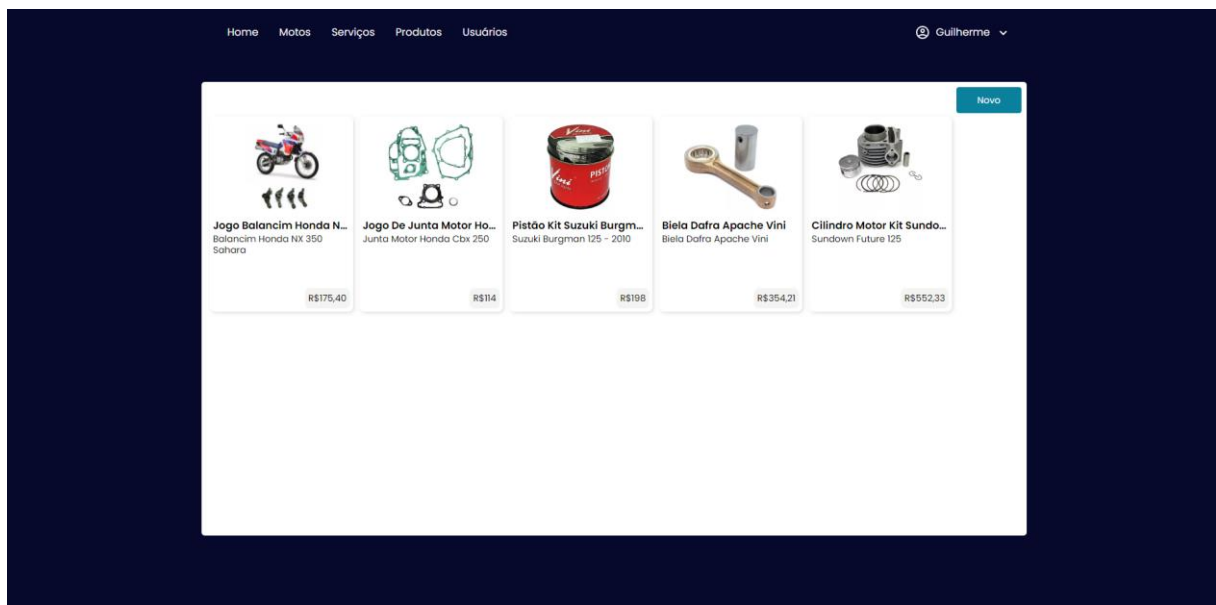
Figura 26 - Dashboard com resumo de dados



Fonte: Elaborado pelo autor (2023).

Gerenciamento de produtos, nesta página o administrador poderá criar, editar, visualizar e remover um produto do seu estoque Figura 27.

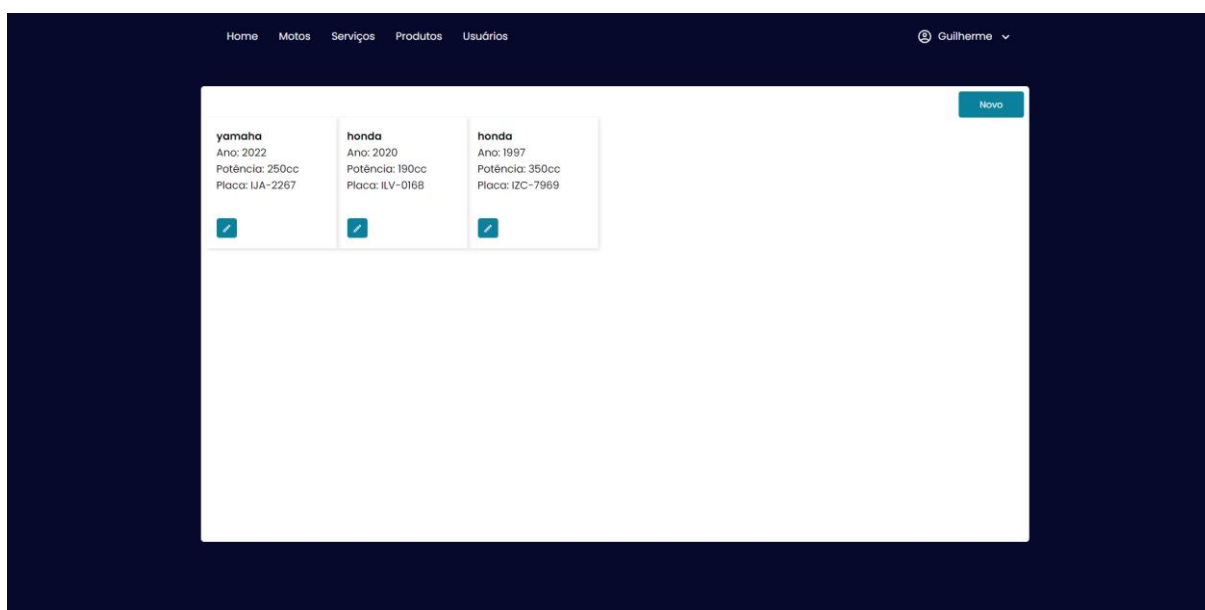
Figura 27 - Tela de listagem de produtos



Fonte: Elaborado pelo autor (2023).

Gerenciamento de motos, nesta página o administrador poderá criar, editar, visualizar e remover uma moto Figura 28.

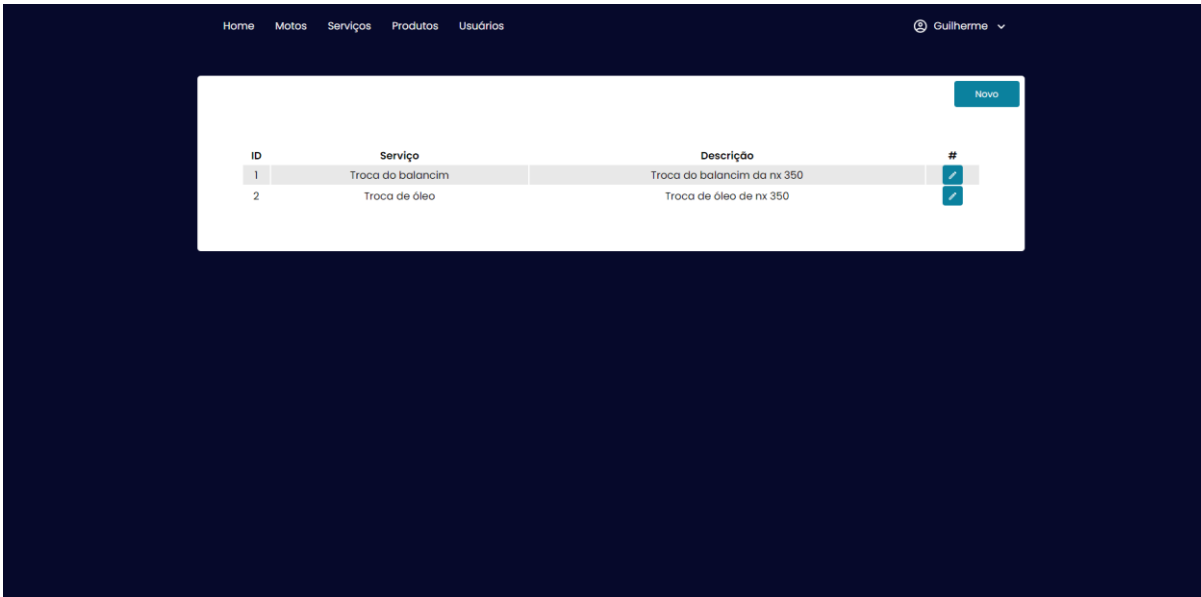
Figura 28 - Tela de listagem de motos





Fonte: Elaborado pelo autor (2023).

Gerenciamento de serviços, nesta página o administrador poderá criar, editar, visualizar um serviço Figura 29. Sendo o pilar de gerenciamento mais importante, pois une os modelos de produto, moto e usuário. O administrador poderá criar seus serviços realizados em motocicletas neste menu, tendo o histórico de informações sobre o que foi realizado e o que foi utilizado de produtos.

Figura 29 - Tela de listagem de serviços



ID	Serviço	Descrição	#
1	Troca do balancim	Troca do balancim da rx 350	
2	Troca de óleo	Troca de óleo de rx 350	

Fonte: Elaborado pelo autor (2023).

5.5 Segurança

Para garantir a segurança do sistema, na camada *backend*, por se tratar de uma API privada, é necessária uma autorização para acessar os serviços. Para isso, foi implementado um *token* de autenticação, que deve ser enviado pelo *frontend* para fazer requisição de dados. Antes da API responder as requisições, é validado o *token*, evitando que qualquer script malicioso possa solicitar informações a camada de *backend*. Na camada *frontend*, é necessário fazer um login, este é feito com os dados do usuário, e-mail e senha, essa senha está previamente persistida no banco de dados, em formato de *hash*, ou seja, somente o *backend* para pode fazer a comparação de senhas para efetuação do *login*.

A troca de dados entre as duas camadas está sendo feito pelo protocolo [https](https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview)¹, ou seja, todo dado é encriptado no transporte. O sistema possui dados sensíveis, portanto um dos desafios para tornar uma aplicação segura está na própria infraestrutura, onde o sistema ficara implantado.

¹ <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview>

6 TESTES

A fim de corrigir algumas possíveis falhas na programação de ambas a camada da aplicação, validar o sistema, foram idealizados alguns testes com a oficina de caso de uso.

O sistema foi disponibilizado ao usuário para acesso do sistema, e com isso recebeu orientações. A coleta das informações foi realizada durante os testes na oficina, foram anotados pontos de melhorias e feedbacks do usuário. Para a seção de testes, foi elaborado um roteiro com tarefas, orientando e colhendo informações específicas sobre as funcionalidades do sistema.

Exemplo de tarefa:

Página: Página de gerenciamento de usuários

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Cadastrar um novo usuário no sistema.**

Usuário conseguiu realizar a tarefa proposta?

- Sim, com facilidade
- Sim, mas com dificuldade
- Não conseguiu realizar a tarefa.

Impressões gerais:

O teste teve uma duração de seis horas, realizado localmente na oficina de teste. O usuário pode testar o sistema com dados reais, isso trouxe mais experiência durante os testes, sobre as funcionalidades e o comportamento da ferramenta. Os dados inseridos foram com base na demanda do dia do teste. Então foram cadastrados motos, produtos e serviços. O documento resultante do teste está disponível no Anexo A.

Como feedback do usuário durante o uso da ferramenta, foi de melhorias que o sistema poderá ter com novas funcionalidades, estas por exemplo, seria o envio de notificações de serviço ou orçamentos para o cliente, filtros na tabela de serviços, e melhoria na visualização do histórico de serviço de uma motocicleta.

7 CONCLUSÃO

O Sistema de Gestão para Oficinas de Motocicletas, permite ao gerente da oficina realizar tarefas de forma simples e com velocidade, todo seu gerenciamento antes realizado de forma manual se tornou ineficaz, com o sistema que pode ser acessado de qualquer lugar, tem todo o controle de produtos, motos e serviços na palma da sua mão. Com base na avaliação realizada pelo usuário, durante a fase de testes, demonstra que o sistema tem potencial para proporcionar ao gerente da oficina uma eficiência nos seus processos internos.

Para trabalhos futuros deverá ser seguido o planejamento de funcionalidades do caso de uso, Figura 10:

- Gerenciamento de gastos de gasolina;
- Geração de relatórios;
- Envio de orçamentos para os clientes.

Tais funcionalidades agregarão mais valor para os clientes da oficina.

A elaboração deste projeto, foi construído por dois pilares principais, o conhecimento adquirido no decorrer do curso e capacidade de resolver problemas em um ambiente profissional, demonstrando que o curso em questão, ensina, aplica e disponibiliza conhecimentos suficientes para preparar profissionais com boas qualificações para resolução de problemas.

REFERÊNCIAS

Produção de motocicletas no Brasil deve crescer. Disponível em: <https://diariodocomercio.com.br/economia/producao-de-motocicletas-no-brasil-deve-crescer-79-em-2022-projeta-a-abraciclo/>. Acesso em: 18 jul. 2022.

O potencial das oficinas mecânicas com o boom do app. Disponível em: <https://tribunapr.uol.com.br/conteudo-publicitario/magnetron/o-potencial-das-oficinas-mecanicas-com-o-boom-dos-apps/>. Acesso em: 18 jul. 2022.

Venda de motos cresce e empresas de motopeças apostam na estabilização do setor em 2022. Disponível em: <https://www.digddv.cefetmg.br/wp-content/uploads/sites/52/2018/02/Meu-Possante.pdf>. Acesso em: 20 jun. 2022.

Meu possante: sistema web para gerenciamento de gastos de carros e motocicletas. Disponível em: <https://mobilidadecuritiba.com.br/venda-de-motos-empresas-motopecas-stabilizacao-setor-2022/>. Acesso em: 11 jul. 2022.

Food at home: desenvolvimento de um sistema delivery responsivo web. Disponível em: https://www.unirios.edu.br/revistarios/media/revistas/2021/31/food_at_home.pdf. Acesso em: 11 jul. 2022.

Gastos com delivery com alimentação fora do lar somam R\$ 40,5 bilhões em 2021. Disponível em: <https://mercadoconsumo.com.br/2022/03/24/gastos-com-delivery-no-foodservice-somaram-r-405-bilhoes-em-2021/#:~:text=Um%20dos%20setores%20mais%20impactados,acima%20do%20que%20em%202020>. Acesso em: 11 jul. 2022.

Brasil tem mais smartphones que habitantes, aponta FGV. Disponível em: <https://www.cnnbrasil.com.br/business/brasil-tem-mais-smartphones-que-habitantes-aponta-fgv/#:~:text=Compartilhe%3A,de%20acordo%20com%20o%20IBGE>. Acesso em: 11 jul. 2022.

Oficinas mecânicas abrem 2,2 milhões de vagas de trabalho em 2022. Disponível em: <https://www.sbtnews.com.br/noticia/sbt-brasil/211158-oficinas-mecanicas-abriram-22-milhoes-de-vagas-de-trabalho-em-2022>. Acesso em: 12 jul. 2022.

DE OLIVEIRA, André Luis Belini; CARREIRA, Marcio Luis; MORETI, Thiago Moura. Aprimorando a Gestão de Negócios com a utilização de Tecnologia de Informação. **Revista de Ciências Gerenciais**, v. 13, n. 17, p. 141-160, 2009.

Apêndices

APÊNDICE A – DETALHAMENTO DAS TECNOLOGIAS UTILIZADAS

A.1 HTML

O HTML² é uma linguagem de marcação, seu uso é destinado a páginas web, tem como objetivo dar a estrutura de uma página seguindo um padrão de sintaxe, estes chamados de elementos, que podem ser desde uma tabela, parágrafos, imagens, vídeos, campos de texto. Quando o navegador lê um documento deste tipo, ele interpreta e apresenta o conteúdo descrito em um navegador web.

A.2 Javascript

O Javascript³ é uma linguagem de programação que consiste em ser leve e interpretada. Pode ser utilizada no lado do cliente, ou seja, o navegador, onde ela terá o papel de manipular os dados de um formulário, de um parágrafo, de mostrar ou ocultar algum elemento na tela, mas ela também pode ser utilizada no lado do servidor, que em conjunto com algum framework, ela pode inicializar a instancia de um servidor que com isso, em conjunto com o navegador poderá receber dados através de uma requisição e então podendo persistir estes em um banco de dados. Sua escolha foi devida a sua curva de aprendizado, tem um bom suporte para tantos os lados trabalhados e uma boa gama de exemplos para sua utilização.

A.3 Typescript

O Typescript⁴ é uma linguagem de programação, sua utilização é feita para adicionar sintaxe a um código Javascript, isso significa que ela é só utilizada durante o desenvolvimento de um projeto, dando segurança e auxiliando na prevenção de erros. Ela pode ser utilizada no servidor como também no navegador, porém quando em produção é necessário que essa linguagem seja compilada para Javascript que somente assim um navegador poderá interpretá-la.

² https://developer.mozilla.org/pt-BR/docs/learn/getting_started_with_the_web/html_basics

³ <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>

⁴ <https://www.typescriptlang.org/pt/>

A.4 React

O React⁵ é uma biblioteca que auxilia na criação de interfaces, sua sintaxe é escrita em jsx, uma mistura de xml e Javascript. Seu principal conceito é componentizar tudo aquilo que pode repetir em uma interface, desde um botão, a uma tabela, ou seja, estes pequenos elementos separados e que tem vida independentes, podem ser utilizados em qualquer lugar e ter comportamentos diferentes, portanto, se for alterado a cor ou alguma propriedade deste componente, todos serão afetados. Sua escolha foi feita com base neste conceito, uma boa documentação e por ter uma alta performance em produção.

A.5 Nextjs

O Nextjs⁶ é um framework, que utiliza em sua base o React, contendo outras funcionalidades que auxiliam durante a configuração do projeto, otimização de imagens, renderização de páginas e componentes no lado do servidor.

A.6 Vercel

A Vercel⁷ é uma plataforma como serviço (PaaS) público ou privado, seu propósito é criar um ambiente de produção para poder hospedar um site, como eles sendo os criadores do Nextjs, colocar em produção o projeto acaba sendo fácil e sem nenhuma dificuldade. Após colocado em produção eles dispõem de uma url para poder acessar a interface.

A.7 Nestjs

O Nestjs⁸ é um framework, ou seja, sua construção é feita a partir de um conjunto de bibliotecas. Seu uso em um projeto é na camada do backend, ele ficará responsável por criar regras de negócio, criar entidades, classes, conexão com o banco de dados, persistência dos dados.

⁵ <https://react.dev/>

⁶ <https://nextjs.org/>

⁷ <https://vercel.com/>

⁸ <https://docs.nestjs.com/>

A.8 Postgres

O Postgres⁹ é um banco de dados relacional, possui os princípios de ter relações entre entidades através de chaves estrangeiras, um banco confiável, eficiente e rápido em suas consultas.

A.9 Ngrok

O Ngrok¹⁰ é responsável por criar um túnel com segurança de uma máquina para a internet, ao invés de criar um DNS, hospedar o backend ou alguma aplicação que deseje expor na internet, como ele é possível abstrair isso e então expor esse serviço de uma máquina qualquer, disponibilizando então uma url para poder disponibilizar para a camada do frontend conseguir consultar os dados persistidos.

A.10 Styled Components

O Styled Components¹¹ permite criar estilo para uma interface misturando o poder do css¹² junto com Javascript, ele tem como objetivo ajudar na construção de componentes com estilos próprios para serem reutilizados novamente.

⁹ <https://www.postgresql.org/docs/current/intro-what-is.html>

¹⁰ <https://ngrok.com/>

¹¹ <https://styled-components.com/docs>

¹² <https://developer.mozilla.org/pt-BR/docs/Web/CSS>

APÊNDICE B – DETALHAMENTO DOS CASOS DE USO

Nome do Caso de Uso	Realizar Autenticação
Resumo (Objetivo)	Permitir que o usuário se autentique no sistema para acessar as funcionalidades.
Atores	Cliente, funcionário.
Pré-condições	Possuir e-mail e senha (estar cadastrado no sistema)
Fluxo Principal	<p>P1. O usuário acessa a URL do sistema.</p> <p>P2. O sistema apresenta o formulário de autenticação.</p> <p>P3. O usuário informa os dados (e-mail e senha).</p> <p>P4. O usuário seleciona a opção de tela “Entrar”</p> <p>P5. O sistema valida os dados.</p> <p>P6. O sistema mostra a tela principal do sistema.</p>
Fluxo Alternativo	<p>FA1. No P5 quando usuário informado não existe na base de dados</p> <ul style="list-style-type: none"> A. O sistema exibe a seguinte msg “Usuário não cadastrado no sistema” B. O sistema retorna ao P2. <p>FA2. No P5 quando usuário informa senha errada</p> <ul style="list-style-type: none"> A. O sistema exibe a seguinte msg “Dados incorretos” B. O sistema retorna ao P2 <p>FA3. No P5 quando usuário não informa os dados obrigatórios</p> <ul style="list-style-type: none"> A. O sistema exibe a seguinte msg “Informe os campos obrigatórios” B. O sistema retorna ao P2
Pós-condições	O usuário é autenticado no sistema.
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	E-mail: campo alfanumérico. Campo obrigatório Senha: campo alfanumérico. Campo obrigatório
Observações	-

Nome do Caso de Uso	Visualizar perfil
Resumo (Objetivo)	Permitir que o usuário entre em seu perfil e edite ele.
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	P1. Navegar até a opção de perfil. P2. Visualizar perfil. P3. Editar perfil se necessário.
Fluxo Alternativo	-
Pós-condições	-
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/01/2022
Regras de Validação	-
Observações	-

Nome do Caso de Uso	Cadastrar moto
Resumo (Objetivo)	Permitir que o usuário cadastre sua(s) moto(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de motos.</p> <p>P2. O usuário abre o formulário de cadastro.</p> <p>P3. O sistema apresenta o formulário.</p> <p>P4. O usuário insere os dados.</p> <p>P5. O usuário seleciona a opção de tela “Salvar”.</p> <p>P6. O sistema valida e salva os dados.</p> <p>P7. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	<p>FA1. No P6 quando usuário não informa os dados obrigatórios</p> <p style="padding-left: 40px;">A. O sistema exibe a seguinte msg “Informe os campos obrigatórios”</p> <p style="padding-left: 40px;">B. O sistema retorna ao P2</p>
Pós-condições	O sistema redireciona para a listagem das motos.
Regras de Negócio	A moto será única no sistema.
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	<p>Modelo: campo alfanumérico. Campo obrigatório</p> <p>Placa: campo alfanumérico. Campo obrigatório</p> <p>Cilindradas: campo alfanumérico. Campo obrigatório</p> <p>Ano: campo alfanumérico. Campo obrigatório</p>
Observações	-

Nome do Caso de Uso	Visualizar moto
Resumo (Objetivo)	Permitir que o usuário visualize sua(s) moto(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de motos.</p> <p>P2. O sistema mostra a listagem de suas motos.</p> <p>P3. O usuário pode selecionar uma moto para ver seus dados.</p> <p>p4. O sistema mostra os dados da moto.</p>
Fluxo Alternativo	-
Pós-condições	-
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	-
Observações	-

Nome do Caso de Uso	Editar moto
Resumo (Objetivo)	Permitir que o usuário edite sua(s) moto(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de motos.</p> <p>P2. O sistema mostra a listagem de suas motos.</p> <p>P3. O usuário seleciona uma moto.</p> <p>P4. O sistema mostra o formulário para editar a moto.</p> <p>P5. O usuário edita os dados.</p> <p>P6. O usuário seleciona a opção de tela “Salvar”.</p> <p>P7. O sistema valida e salva os dados.</p> <p>P8. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	<p>FA1. No P6 quando usuário não informa os dados obrigatórios</p> <p style="padding-left: 40px;">C. O sistema exibe a seguinte msg “Informe os campos obrigatórios”</p> <p style="padding-left: 40px;">D. O sistema retorna ao P2</p>
Pós-condições	O sistema redireciona para a listagem das motos.
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	<p>Modelo: campo alfanumérico. Campo obrigatório</p> <p>Placa: campo alfanumérico. Campo obrigatório</p> <p>Cilindradas: campo alfanumérico. Campo obrigatório</p> <p>Ano: campo alfanumérico. Campo obrigatório</p>
Observações	-

Nome do Caso de Uso	Excluir moto
Resumo (Objetivo)	Permitir que o usuário exclua sua(s) moto(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de motos.</p> <p>P2. O sistema mostra a listagem de suas motos.</p> <p>P3. O usuário seleciona a opção de tela "Excluir".</p> <p>P4. O sistema mostra uma mensagem para confirmar a exclusão.</p> <p>P5. O usuário confirma a exclusão.</p> <p>P6. O sistema exclui a moto.</p> <p>P7. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	-
Pós-condições	O sistema redireciona para a listagem das motos.
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	-
Observações	-

Nome do Caso de Uso	Visualizar serviços
Resumo (Objetivo)	Permitir que o usuário visualize seu(s) serviço(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de serviços.</p> <p>P2. O sistema mostra a listagem de serviços.</p> <p>P3. O usuário pode selecionar um serviço para ver seus dados.</p> <p>p4. O sistema mostra os dados do serviço.</p>
Fluxo Alternativo	-
Pós-condições	-
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	-
Observações	-

Nome do Caso de Uso	Editar serviço
Resumo (Objetivo)	Permitir que o usuário edite seu(s) serviço(s).
Atores	Cliente, funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de serviços.</p> <p>P2. O sistema mostra a listagem de seus serviços.</p> <p>P3. O usuário seleciona um serviço.</p> <p>P4. O sistema mostra o formulário para editar o serviço.</p> <p>P5. O usuário edita os dados.</p> <p>P6. O usuário seleciona a opção de tela “Salvar”.</p> <p>P7. O sistema valida e salva os dados.</p> <p>P8. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	<p>FA1. No P6 quando usuário não informa os dados obrigatórios</p> <p>A. O sistema exibe a seguinte msg “Informe os campos obrigatórios”</p> <p>B. O sistema retorna ao P2</p>
Pós-condições	O sistema redireciona para a listagem dos serviços.
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	<p>Nome: campo alfanumérico. Campo obrigatório</p> <p>Descrição: campo alfanumérico. Campo obrigatório</p>
Observações	-

Nome do Caso de Uso	Cadastrar produto
Resumo (Objetivo)	Permitir que o usuário cadastre seu(s) produto(s).
Atores	Funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de produtos.</p> <p>P2. O usuário abre o formulário de cadastro.</p> <p>P3. O sistema apresenta o formulário.</p> <p>P4. O usuário insere os dados.</p> <p>P5. O usuário seleciona a opção de tela “Salvar”.</p> <p>P6. O sistema valida e salva os dados.</p> <p>P7. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	<p>FA1. No P6 quando usuário não informa os dados obrigatórios</p> <p>A. O sistema exibe a seguinte msg “Informe os campos obrigatórios”</p> <p>B. O sistema retorna ao P2</p>
Pós-condições	O sistema redireciona para a listagem dos produtos.
Regras de Negócio	O produto será único no sistema.
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	
Observações	-

Nome do Caso de Uso	Visualizar produto
Resumo (Objetivo)	Permitir que o usuário visualize seu(s) produto(s).
Atores	Funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de produtos.</p> <p>P2. O sistema mostra a listagem dos seus produtos.</p> <p>P3. O usuário pode selecionar um produto para ver seus dados.</p> <p>p4. O sistema mostra os dados do produto.</p>
Fluxo Alternativo	-
Pós-condições	-
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	-
Observações	-

Nome do Caso de Uso	Editar produto
Resumo (Objetivo)	Permitir que o usuário edite seu(s) produto(s).
Atores	Funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de produtos.</p> <p>P2. O sistema mostra a listagem de seus produtos.</p> <p>P3. O usuário seleciona um produto.</p> <p>P4. O sistema mostra o formulário para editar a moto.</p> <p>P5. O usuário edita os dados.</p> <p>P6. O usuário seleciona a opção de tela “Salvar”.</p> <p>P7. O sistema valida e salva os dados.</p> <p>P8. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	<p>FA1. No P6 quando usuário não informa os dados obrigatórios</p> <p>C. O sistema exibe a seguinte msg “Informe os campos obrigatórios”</p> <p>D. O sistema retorna ao P2</p>
Pós-condições	O sistema redireciona para a listagem dos produtos.
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	<p>Nome: campo alfanumérico. Campo obrigatório</p> <p>Preço: campo alfanumérico. Campo obrigatório</p> <p>Código: campo alfanumérico. Campo obrigatório</p> <p>Marca: campo alfanumérico. Campo obrigatório</p> <p>Descrição: campo alfanumérico. Campo obrigatório</p>
Observações	-

Nome do Caso de Uso	Excluir produto
Resumo (Objetivo)	Permitir que o usuário exclua seu(s) produto(s).
Atores	Funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a opção de produtos.</p> <p>P2. O sistema mostra a listagem de seus produtos.</p> <p>P3. O usuário seleciona a opção de tela "Excluir".</p> <p>P4. O sistema mostra uma mensagem para confirmar a exclusão.</p> <p>P5. O usuário confirma a exclusão.</p> <p>P6. O sistema exclui o produto.</p> <p>P7. O sistema mostra uma mensagem para o usuário.</p>
Fluxo Alternativo	-
Pós-condições	O sistema redireciona para a listagem dos produtos.
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	-
Observações	-

Nome do Caso de Uso	Visualizar dashboard
Resumo (Objetivo)	Permitir que o usuário visualize o dashboard.
Atores	Funcionário.
Pré-condições	Estar logado no sistema
Fluxo Principal	<p>P1. O usuário navega até a página principal.</p> <p>P2. O sistema mostra a listagem de dados no formato de um dashboard.</p> <p>P3. O usuário pode visualizar a informação de resumo de produtos e usuários.</p>
Fluxo Alternativo	-
Pós-condições	-
Regras de Negócio	-
Requisitos Não Funcionais	-
Autor	Guilherme
Data	18/06/2023
Regras de Validação	-
Observações	-

Anexos

ANEXO A – DOCUMENTO DE TESTE COM USUÁRIOS

Neste teste de usabilidade iremos testar o Sistema de Gestão para Oficinas de Motocicletas.

Anotações gerais

Produto:

Teste n°: 1

Data: 14/08/2023

Início da sessão: 9:00 - 12:00

Término da sessão: 14:00 - 17:00

Local: Oficina mecânica

Entrevistador: Guilherme Müller Severo

Usuário: Maicon Roberto

Realização de tarefas

O principal objetivo é a avaliação do sistema por parte da oficina, se o sistema atende às mínimas expectativas para poder ser utilizado e identificar os pontos de melhoria.

Tarefa A

Página: Página de gerenciamento de usuários

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Cadastrar um novo usuário no sistema.**

Usuário conseguiu realizar a tarefa proposta?

- Sim, com facilidade
- Sim, mas com dificuldade
- Não conseguiu realizar a tarefa.

Impressões gerais: Achou fácil de gerenciar os usuários, atendendo o propósito, mas de melhorias acha que se tiver mais dados sobre o cliente, seria bem importante.

Tarefa B

Página: Página de gerenciamento de motos

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Cadastrar uma nova moto no sistema.**

Usuário conseguiu realizar a tarefa proposta?

- Sim, com facilidade
- Sim, mas com dificuldade
- Não conseguiu realizar a tarefa

Impressões gerais: Achou fácil de gerenciar as motos, salientando que vai ser importante ter estes dados na plataforma.

Tarefa C

Página: Página de gerenciamento de produtos

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Cadastrar um novo produto no sistema.**

Usuário conseguiu realizar a tarefa proposta?

- Sim, com facilidade
- Sim, mas com dificuldade
- Não conseguiu realizar a tarefa

Impressões gerais: A funcionalidade atendeu à finalidade, mas pontuou que seria útil a colocação de um campo com a listagem das motos que são compatíveis com esta moto.

Tarefa D

Página: Página de gerenciamento de serviços

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Cadastrar um novo serviço no sistema.**

Usuário conseguiu realizar a tarefa proposta?

- Sim, com facilidade
- Sim, mas com dificuldade
- Não conseguiu realizar a tarefa

Impressões gerais: A funcionalidade cumpre com o propósito, porém para o modelo de negócio da oficina, seria importante ter mais de um orçamento dentro do serviço e não apenas um como, foi proposto.

Tarefa E

Página: Página principal

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Conferir se os dados apresentados na dashboard são dados relevantes para o gerente.**

Usuário conseguiu visualizar se os dados estão de acordo com a tarefa proposta?

- Sim, são dados relevantes
- Sim, mas poderia ter mais dados
- Não possui dados relevantes

Impressões gerais: Achou relevantes os dados exibidos, e que o propósito da dashboard é de grande importância para o gerenciamento da oficina.

Tarefa F

Página: Geral

Objetivo: Entender se esta página cumpre com os requisitos.

Descrição da tarefa: **Navegando pela interface do sistema e sabendo que é um sistema que teve como proposta algumas das funcionalidades mapeadas. Você acha que ele pode ser um produto viável com estas funcionalidades?**

Usuário conseguiu visualizar que o sistema proposto poderá ser utilizado?

- Sim, poderia ser utilizado
- Sim, mas como primeira versão ainda falta algumas funcionalidades.
- Não seria um sistema viável

Impressões gerais: De modo geral o sistema atende como propósito para realizar o gerenciamento de uma oficina mecânica. Adicionando as outras funcionalidades que foram mapeadas para uma próxima versão se tornaria uma ferramenta indispensável em uma oficina. Portanto o gerente vê como um produto de uso real para o gerenciamento.

Perguntas:

O sistema atende ao propósito de poder ajudar a gerenciar uma oficina?

- Sim

- Não

Tendo em vista que o sistema terá melhorias futuras, na sua visão, ele se tornará um bom produto?

- Sim

- Não

Com o número atual de funcionalidades, o quão satisfeito você ficou com o sistema?

- Muito satisfeito

- Satisfeito

- Nada satisfeito

Conclusão

Momento para rever como o usuário se sentiu realizando as tarefas, se tem algo a adicionar para contribuir com a melhoria do projeto.

Se não tiver tocado nos seguintes assuntos, é possível perguntar diretamente ao usuário o que ele achou sobre:

- Dar sua opinião sobre o sistema
- Comentar sobre quaisquer pontos que não foram comentados
- Sobre gerar orçamentos para os usuários
- Notificações para os clientes
- Uso da ferramenta pelos clientes