

# **AGENDAGORA - SISTEMA DE AGENDAMENTO ONLINE PARA A BUSCA DE PRESTADORES DE SERVIÇOS**

**Bruna Eduarda Zuconelli e Matheus Krüger**

**Orientador: Roberto Luiz Debarba**

Curso Técnico em Informática - Ano/Semestre: 2023/3

Centro de Educação Profissional de Timbó (CEDUP) – Timbó, SC – Brasil

## **1 INTRODUÇÃO**

De acordo com Klaus Schwab, Fundador e Presidente Executivo do Fórum Econômico Mundial, no decorrer dos anos, a digitalização, a conectividade e a crescente dependência da tecnologia reconfigurou profundamente a maneira como interagimos com produtos e serviços, o que tem revolucionado a maneira como as pessoas realizam suas atividades cotidianas.

A Primeira Revolução Industrial utilizou água e energia a vapor para mecanizar a produção. A Segunda usou energia elétrica para criar produção em massa. O Terceiro usou eletrônica e tecnologia da informação para automatizar a produção. Agora, a Quarta Revolução Industrial baseia-se na Terceira, a revolução digital que tem ocorrido desde meados do século passado. É caracterizada por uma fusão de tecnologias que está confundindo os limites entre as esferas física, digital e biológica. Também estão a ocorrer grandes mudanças do lado da procura, à medida que a crescente transparência, o envolvimento do consumidor e os novos padrões de comportamento do consumidor (cada vez mais baseados no acesso a redes e dados móveis) forçam as empresas a adaptar a forma como concebem, comercializam e fornecem produtos e serviços. (The Fourth Industrial Revolution: what it means, how to respond - Klaus Schwab - 2023)

Nesse contexto, a prestação de serviços online tem se destacado como uma solução conveniente para atender às demandas do público. O desenvolvimento de um sistema de agendamento de serviços online emerge como uma resposta inovadora às demandas em constante evolução da sociedade moderna. O agendamento de serviços online, em particular, vem como um elemento fundamental dessa transformação, oferecendo a capacidade de agendar compromissos de maneira eficiente e acessível, redefinindo a maneira como consumidores e empresas organizam compromissos.

Este trabalho mostra os processos de concepção, criação e implementação deste sistema, explorando como essa solução tecnológica pode otimizar a forma de como clientes agendam compromissos e como empresas gerenciam suas operações, bem como os possíveis impactos no âmbito dos negócios e da experiência do usuário. Ao examinar os benefícios, desafios técnicos e

potenciais impactos sociais desse empreendimento, busca-se compreender de que forma essa plataforma pode transformar a maneira como interagimos com os serviços que utilizamos diariamente, e é possível abrir caminho para uma discussão sobre como as soluções digitais continuam a moldar nosso mundo cotidiano.

O sistema terá dois perfis de acesso, dos prestadores de serviços e dos contratantes, no qual será possível filtrar profissionais cadastrados pelos tipos de serviços prestados, qualificações, avaliações e formas de pagamento. Também será possível o agendamento online, podendo escolher data e horários disponíveis, para evitar conflitos. Com essas informações, é possível uma maior visibilidade e profissionalismo para o prestador e confiança para o contratante, podendo reduzir o não comparecimento, automatizando o processo de agendamento para os contratantes e o de gerenciamento de agenda para os prestadores, sem a necessidade de buscar em outras plataformas e entrar em contato com vários prestadores para confirmar com cada um os horários disponíveis.

Para a implementação *frontend* serão utilizadas as linguagens de programação HTML, Javascript e CSS na IDE Visual Studio, para o *backend* linguagem Java com a IDEA IntelliJ, junto ao recursos do framework Spring. Quanto ao banco de dados, será utilizado o banco de dados relacional MySQL. Como nossa plataforma possui um filtro de localização, será utilizada a API do Google Maps para Javascript.

## **2 OBJETIVOS**

### **2.1 OBJETIVO GERAL**

O sistema tem como objetivo a busca dos profissionais por parte do contratante, incluindo avaliações e gerenciamento de sua agenda por parte do prestador de serviços.

### **2.2 OBJETIVOS ESPECÍFICOS**

- Permitir ao contratante escolher o prestador de serviços e realizar o agendamento;
- Possibilitar ao contratante filtrar o prestador de serviços por localização e forma de pagamento;
- Permitir ao contratante avaliar o prestador de serviços;
- Permitir ao prestador de serviços o gerenciamento da sua agenda;

## **3 FUNDAMENTAÇÃO TEÓRICA**

### **3.1 Mercado de autônomos no Brasil e o impacto social de um sistema de agendamento online**

O mercado de autônomos no Brasil tem apresentado um crescimento significativo nos últimos anos. Segundo dados da Pesquisa Nacional por Amostra de Domicílio (Pnad), divulgada pelo IBGE (2022), o Brasil tem hoje mais de 25,7 milhões de trabalhadores atuando nessa modalidade. Os autônomos são profissionais que atuam de forma independente, sem vínculo empregatício, e oferecem serviços variados para atender às demandas dos consumidores.

De acordo com Sennett (1998), discute a natureza fragmentada e temporária do trabalho moderno, que frequentemente impede o desenvolvimento de habilidades especializadas e a formação de conexões significativas com colegas e comunidades de trabalho. Ele explora como essas mudanças afetam a forma como as pessoas se veem e como se sentem em relação ao trabalho e à sociedade. O trabalho autônomo pode ser visto como uma resposta às tendências descritas por Sennett, oferecendo aos profissionais a oportunidade de buscar uma maior autonomia, controle sobre suas atividades profissionais e a possibilidade de desenvolver relações de trabalho mais significativas. A prestação de serviços proporciona flexibilidade e autonomia aos profissionais, permitindo que eles escolham seus horários de trabalho, determinem os serviços oferecidos e estabeleçam suas próprias condições de atuação. Essa flexibilidade atrai tanto pessoas que buscam uma renda complementar como aquelas que desejam empreender por conta própria.

Um estudo feito pela Feijó (2022) e a Fundação Getúlio Vargas (FGV 2022) com base nos microdados da Pesquisa Nacional por Amostra de Domicílios (Pnad Contínua), divulgada pelo IBGE, que avaliou o panorama do mercado de trabalho no Brasil, diz que a retomada do emprego no Brasil tem ocorrido principalmente pelo crescimento do trabalho autônomo que está diretamente relacionado ao aumento do empreendedorismo no Brasil. Com a instabilidade econômica e a busca por autonomia profissional, muitas pessoas têm optado por iniciar seus próprios negócios e oferecer serviços como autônomos. Esse movimento é impulsionado por diversos fatores, como a busca por maior liberdade, a insatisfação com o mercado de trabalho tradicional, necessidade por conta da perda do emprego formal e a facilidade de acesso a tecnologias que permitem o trabalho remoto. Abrange uma ampla gama de segmentos e profissões. Desde profissionais de TI, marketing digital, design gráfico, consultoria, coaching, até profissionais da área de beleza, saúde, educação e construção civil, entre outros. Essa diversidade de segmentos oferece oportunidades para profissionais de diferentes áreas e permite que eles explorem suas habilidades e conhecimentos específicos.

Ainda baseado nesse estudo da FGV, observa-se que a evolução da tecnologia e o fácil acesso à informação têm sido fatores determinantes para o crescimento do mercado de autônomos.

Com o uso da internet e das redes sociais, os profissionais autônomos têm maior visibilidade e podem divulgar seus serviços de forma mais ampla. Além disso, as tecnologias digitais permitem a realização de trabalhos remotos, facilitam a comunicação com os clientes e permitem a gestão de tarefas e finanças de maneira mais eficiente. Apesar das oportunidades oferecidas pelo mercado de autônomos, é importante ressaltar que também existem desafios a serem enfrentados. Os autônomos precisam lidar com a busca constante por clientes, a instabilidade financeira, a falta de benefícios trabalhistas e a necessidade de desenvolver habilidades de gestão e marketing para se destacarem em meio à concorrência.

Em suma, os dados da PNAD mostram um crescimento expressivo do mercado de autônomos no Brasil, que está em ascensão, impulsionado pela busca por autonomia, flexibilidade e empreendedorismo. A diversidade de segmentos, o acesso à tecnologia e a facilidade de comunicação têm proporcionado oportunidades para profissionais autônomos se destacarem e conquistarem seu espaço no mercado. No entanto, é importante estar preparado para os desafios inerentes a essa modalidade de trabalho.

De acordo com um artigo do Sebrae (2023) sobre os negócios de impacto social, a expansão dos autônomos no mercado brasileiro realmente tem um impacto social notável. Esses profissionais têm a habilidade única de preencher nichos de mercado que muitas vezes passam despercebidos por empresas maiores. Eles oferecem uma ampla gama de serviços, que vão desde tarefas simples de reparos domésticos até serviços mais complexos de consultoria e design. Essa diversidade de opções beneficia diretamente a população, que pode escolher entre uma vasta lista de profissionais para atender às suas necessidades específicas. Além disso, a presença dos autônomos contribui de forma significativa para a geração de empregos, o que se torna ainda mais crucial durante períodos de instabilidade econômica. Esses profissionais autônomos não apenas preenchem lacunas de emprego, mas também fornecem um leque variado de serviços para os consumidores, aumentando a concorrência saudável e a qualidade dos serviços oferecidos no mercado.

“Causar um impacto positivo em uma comunidade, ampliar as perspectivas de pessoas marginalizadas pela sociedade, além de gerar renda compartilhada e autonomia financeira para os indivíduos de classe baixa: estes são alguns dos objetivos dos negócios de impacto social. Os negócios de impacto social buscam beneficiar diretamente pessoas de renda mais baixa, das chamadas classes C, D e E.” (SEBRAE, 2023)

Contudo, de acordo com o advogado Coriolano (2019), é importante reconhecer que, apesar dos benefícios, a falta de regulamentações claras pode acarretar desafios, como a ausência de benefícios trabalhistas, e de proteção social para esses profissionais. Esse é um aspecto crítico que

deve ser abordado para garantir que os autônomos possam desfrutar de condições de trabalho justas e sustentáveis.

### **3.2 Usabilidade e UI x UX**

Neste momento, iniciaremos uma discussão sobre UI x UX, que de acordo com Souza (2017), a User Experience (Experiência do Usuário) refere-se à experiência geral de um usuário ao interagir com um produto, sistema ou serviço. Ela abrange uma ampla gama de fatores que afetam as percepções e sentimentos dos usuários durante sua interação.

A seguir algumas noções fundamentais sobre UX:

- A usabilidade é um aspecto crucial da UX. Ela se concentra em quão fácil e eficaz é para os usuários realizar tarefas específicas dentro do sistema. Interfaces intuitivas, fluxos de trabalho claros e a minimização de obstáculos contribuem para uma boa usabilidade.
- A acessibilidade garante que o produto seja utilizável por pessoas com diferentes habilidades e necessidades. Isso envolve considerações para pessoas com deficiências visuais, auditivas, motoras e cognitivas, garantindo que todos possam interagir com o produto de maneira eficaz.
- O fluxo de usuário se refere à sequência de etapas que os usuários seguem ao realizar tarefas específicas. Um fluxo de usuário bem projetado leva em consideração a lógica e a sequência das ações do usuário, minimizando qualquer confusão ou complexidade desnecessária.
- Proporcionar feedback claro aos usuários sobre suas ações é fundamental. Isso ajuda os usuários a entender o que está acontecendo, reduzindo a incerteza e a frustração. A comunicação eficaz é essencial para uma boa experiência do usuário.

Segundo Nielsen (1990) as heurísticas são 10 princípios de avaliação da usabilidade de interfaces de softwares, são elas: Visibilidade do status do sistema, correspondência entre o sistema e o mundo real, controle e liberdade do usuário, consistência e padrões, prevenção de erros, reconhecimento em vez de lembrança, flexibilidade e eficiência de uso, estética e design minimalista, ajuda e documentação e diagnóstico e recuperação de erros, estes princípios definem pontos importantes da composição de interfaces e devem ser considerados no momento da criação dos layouts. Seguindo estas diretrizes, aprimora-se a usabilidade, e proporciona uma navegação muito mais agradável, eficiente e satisfatória aos seus usuários. A seguir são listadas as heurísticas de Nilsen:

- Visibilidade do status do sistema: O sistema deve sempre informar aos usuários sobre o que está acontecendo, por meio de feedback adequado, para que eles saibam o status atual das suas interações.
- Correspondência entre o sistema e o mundo real: O sistema deve usar linguagem e conceitos familiares aos usuários, seguindo convenções do mundo real, para que eles possam entender e interpretar facilmente as informações apresentadas.
- Controle e liberdade do usuário: Os usuários devem ter a opção de desfazer ações indesejadas ou sair de situações indesejáveis sem consequências graves. É importante fornecer uma saída clara das funcionalidades ou menus.
- Consistência e padrões: As convenções e a terminologia devem ser consistentes em todo o sistema para evitar confusão e facilitar o aprendizado do usuário. O uso de padrões de design reconhecidos também é recomendado.
- Prevenção de erros: É necessário projetar o sistema de forma a prevenir erros sempre que possível, seja através de mensagens de erro claras e informativas, confirmações antes de ações irreversíveis ou limitando as escolhas dos usuários para evitar opções prejudiciais.
- Reconhecimento em vez de lembrança: É preferível utilizar o reconhecimento em vez da lembrança. Ou seja, as informações e funcionalidades relevantes devem estar visíveis e disponíveis, em vez de depender da memória do usuário.
- Flexibilidade e eficiência de uso: O sistema deve atender tanto a usuários novatos quanto a usuários experientes, permitindo que eles executem tarefas com rapidez e eficiência.
- Estética e design minimalista: O design da interface deve ser limpo, simples e esteticamente agradável, para que os usuários possam se concentrar no conteúdo e nas tarefas, sem distrações desnecessárias.
- Ajuda e documentação: O sistema deve ser projetado de forma que seja possível encontrar ajuda e documentação facilmente, caso os usuários precisem. No entanto, a interface deve ser autoexplicativa o suficiente para minimizar a dependência desses recursos.
- Reconhecimento de erro: O sistema deve fornecer mensagens claras e construtivas para informar aos usuários quando ocorrerem erros, explicando o problema de maneira simples e indicando soluções ou caminhos alternativos.

Em resumo, a usabilidade do software está relacionada à sua capacidade de ser eficiente, de fácil aprendizado, visualização do status do sistema, de proporcionar familiaridade ao usuário, de fornecer feedback adequado, ter consistência e padronização, flexibilidade de uso, de minimização e prevenção de erros e ajuda. Ao atender a esses princípios de usabilidade, ele será mais intuitivo, facilitando a gestão do tempo e aumentando a produtividade do autônomo.

Ainda de acordo com Souza (2017), a User Interface (Interface do Usuário) é a parte tangível do produto com a qual os usuários interagem diretamente. Ela se concentra na apresentação visual e na organização dos elementos para tornar a interação mais eficaz e agradável. Algumas considerações importantes sobre UI incluem:

- O design visual engloba a aparência estética de um produto. Isso inclui a escolha de cores, tipografia, ícones e elementos visuais que contribuem para uma identidade visual coesa e agradável.
- A disposição dos elementos na tela afeta a clareza e a compreensão. Um layout organizado e lógico facilita a navegação e a interação.
- A consistência no design é crucial para criar uma experiência unificada. Elementos semelhantes devem ter aparência e comportamento consistentes em todo o produto.
- A interatividade envolve como os elementos respondem às ações do usuário. Botões, menus e elementos interativos devem responder de maneira previsível e visualmente reconhecível.

Em resumo, enquanto a UX se concentra na experiência emocional e funcional do usuário, a UI se concentra na apresentação visual e interativa dos elementos. Ambos são interdependentes e essenciais para criar produtos que atendam às necessidades dos usuários de forma eficaz e agradável.

### **3.3 Como as soluções digitais continuam a moldar nosso mundo cotidiano.**

De acordo com o artigo de Oliveira, Almeida e Trotta (2020), a evolução da terminologia "tecnologia" ao longo da história, enfatiza a importância das novas tecnologias na vida cotidiana. O artigo reflete também sobre os prós e contras da tecnologia na educação, abordando a transformação das relações sociais, a resistência cultural e os desafios da globalização. O artigo mostra que as soluções digitais continuam a ter um impacto profundo e transformador em nosso mundo cotidiano de várias maneiras:

- Conectividade Global: A internet e as redes sociais nos mantêm conectados, permitindo comunicações instantâneas, compartilhamento de informações e colaboração global em tempo real.
- Acesso à Informação: A internet oferece acesso a uma vasta quantidade de informações. Podemos aprender, pesquisar e obter notícias sobre qualquer tópico, a qualquer momento.
- Trabalho e Educação Remotos: A pandemia de COVID-19 destacou a importância das soluções digitais para o trabalho e a educação remotos. Videoconferências, plataformas de aprendizado on-line e ferramentas de colaboração permitem que as pessoas trabalhem e

estudem a partir de qualquer lugar.

- Comércio Eletrônico: As plataformas de comércio eletrônico tornaram as compras on-line uma parte essencial de nossas vidas, oferecendo conveniência e uma ampla variedade de produtos disponíveis para entrega em casa.
- Saúde Digital: Aplicativos de rastreamento de saúde, dispositivos vestíveis e telemedicina estão permitindo que as pessoas monitorem sua saúde e tenham acesso a cuidados médicos de forma mais eficiente.
- Entretenimento Sob Demanda: Plataformas de streaming oferecem entretenimento personalizado, permitindo que as pessoas assistam a filmes, séries e programas de acordo com seus horários e preferências.
- Mobilidade: Aplicativos de transporte compartilhado e navegação por GPS transformaram a forma como nos deslocamos nas cidades, tornando a mobilidade mais eficiente.
- Redes Sociais e Comunicação: As redes sociais continuam a ser uma maneira fundamental de nos conectarmos com amigos, familiares e colegas, compartilhando momentos e informações.
- Inteligência Artificial e Automatização: A IA está sendo usada em várias áreas, desde assistentes virtuais até automação de processos, otimizando eficiência e tomando decisões com base em dados.
- Acessibilidade e Inclusão: Soluções digitais estão permitindo maior acessibilidade para pessoas com deficiências, tornando o mundo digital mais inclusivo.
- Gestão de Finanças Pessoais: Aplicativos de finanças facilitam o monitoramento de gastos, investimentos e planejamento financeiro.
- Ativismo e Conscientização: As mídias sociais têm sido uma plataforma para conscientização e ativismo, permitindo que as vozes das pessoas sejam ouvidas globalmente.
- Agendamento online: Plataformas com serviços disponibilizados em que o usuário pode fazer esse agendamento diretamente, sem precisar ter o contato direto com o estabelecimento.

Em essência, segundo Holanda (2021), as soluções digitais moldaram nosso mundo cotidiano ao redefinir a maneira como nos comunicamos, trabalhamos, aprendemos, nos entretemos e interagimos. Elas oferecem conveniência, eficiência e oportunidades inovadoras, mas também levantam desafios relacionados à privacidade, segurança cibernética e o equilíbrio entre a vida on-line e off-line.



## 4 ESPECIFICAÇÃO

Um requisito funcional, de acordo com Sommerville (2011), é "uma declaração de um serviço que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações". É uma especificação que descreve uma função ou comportamento específico que um sistema deve ter para atender às necessidades dos usuários. Esses requisitos descrevem as ações que o sistema deve ser capaz de realizar, as entradas que ele deve aceitar e as saídas que deve produzir.

Os requisitos funcionais estão relacionados às tarefas que o sistema deve realizar e às funcionalidades específicas que ele deve suportar. Descrevem o que o sistema deve fazer em termos de processamento de dados, cálculos, manipulação de informações e interações com os usuários. Um requisito funcional geralmente é descrito de forma clara, para que possa ser compreendido tanto pelos desenvolvedores quanto pelos usuários. Ele deve ser testável, ou seja, deve ser possível verificar se o sistema atende ou não a esse requisito. Definimos os seguintes requisitos funcionais (RF) para o sistema, que estão apresentados no Quadro 1.

Quadro 1 – Requisitos funcionais (RF)

<b>Requisito funcional</b>
RF01 – O sistema deve gerenciar usuários;
RF02 – O sistema deve gerenciar a agenda;
RF03 – O sistema deve fazer a autenticação do usuário com login e senha;
RF04 – O sistema deve permitir que os contratantes avaliem os prestadores de serviço;
RF05 – O sistema deve permitir que os contratantes pesquisem prestadores de serviço com base em critérios como localização, tipo de serviço, avaliações e forma de pagamento;
RF06 – O sistema deve gerenciar o histórico de agendamentos;
RF07 – O sistema deve exibir os resultados da pesquisa de prestadores de serviços em um mapa.

Fonte: elaborado pelo autor.

### 4.1 DIAGRAMA DE CASOS DE USO

Um diagrama de caso de uso, de acordo com Rumbaugh, Jacobson e Booch (2005), é "uma representação gráfica das interações entre atores e casos de uso, mostrando como os atores estão envolvidos em certos comportamentos ao interagir com o sistema".

É uma ferramenta de modelagem usada no desenvolvimento de software para visualizar e descrever as interações entre atores (usuários externos, outros sistemas ou dispositivos) e um

sistema em termos de funcionalidades ou recursos oferecidos pelo sistema. É uma representação gráfica que ilustra como os usuários interagem com o sistema para realizar tarefas específicas. Mostra os atores como figuras externas ao sistema, geralmente representados por ícones ou símbolos, e os casos de uso como elipses. As setas entre os atores e os casos de uso indicam as interações ou ações que ocorrem entre eles.

Cada caso de uso representa uma funcionalidade ou uma tarefa específica que o software oferece. Ele descreve as ações que o sistema realiza em resposta às interações dos usuários. Os casos de uso são descritos em termos de cenários, descrevendo as etapas ou passos envolvidos na execução da funcionalidade.

Os diagramas de caso de uso ajudam a capturar os requisitos do sistema e a comunicação entre as partes interessadas, fornecendo uma visão geral do comportamento do sistema a partir da perspectiva do usuário. Eles são amplamente utilizados durante a fase de análise e design de um projeto de desenvolvimento de software, permitindo a identificação e o refinamento dos requisitos funcionais do sistema.

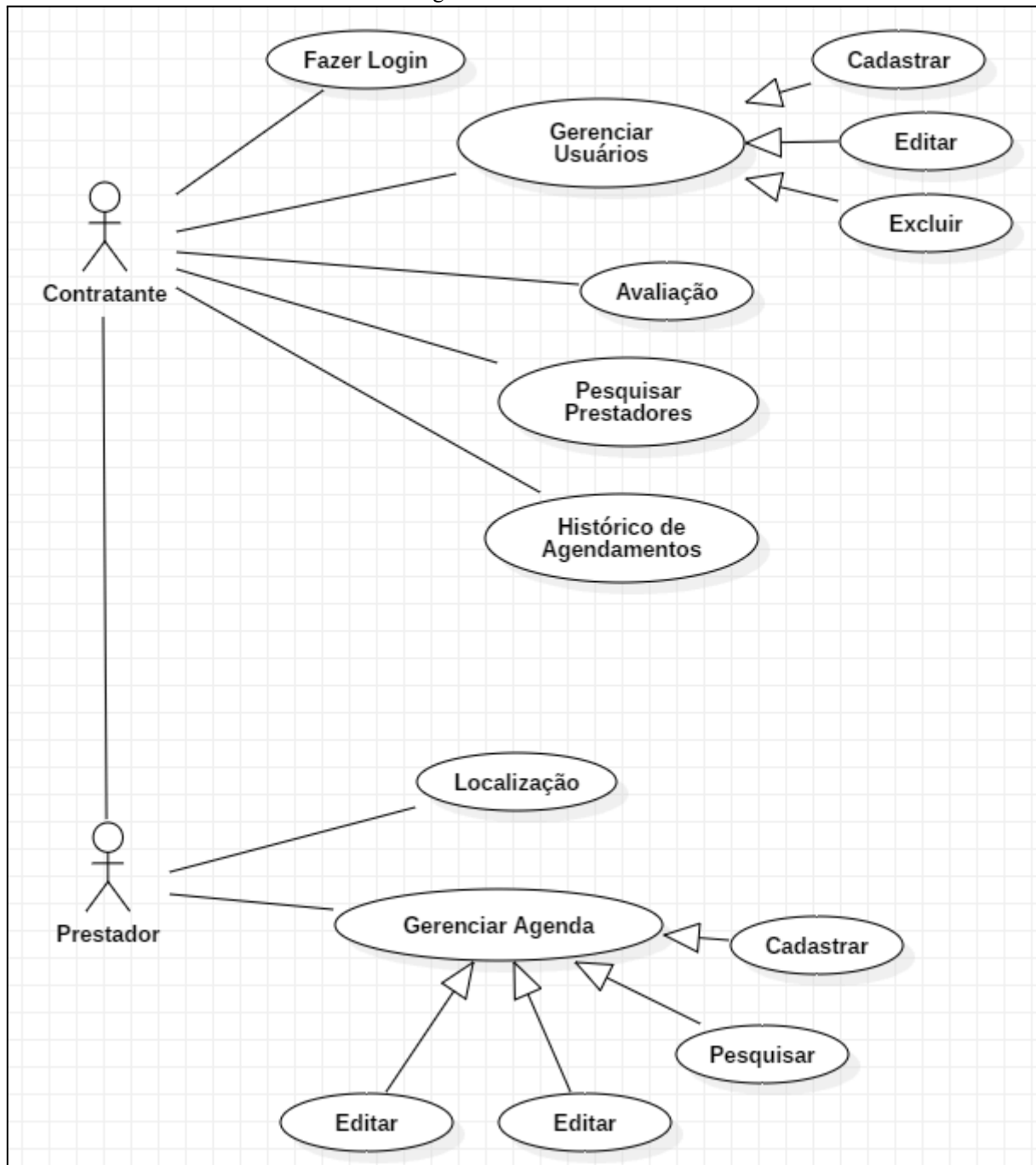
Neste trabalho, apresenta-se os casos de uso da seguinte forma: os atores: Representados por uma figura humana ou ícone relacionado, representam um papel desempenhado por um usuário externo, outro sistema ou dispositivo que interage com o sistema, descrito na Figura 1 como o “Prestador” e “Contratante”.

Linha de associação: Conecta um ator a um caso de uso, mostrando a interação entre eles. Indica que o ator está envolvido no caso de uso.

Relacionamento de generalização/especialização: Representado por uma seta sólida com um triângulo vazio no final, indica que um ator ou caso de uso é uma especialização de outro ator ou caso de uso. É usado quando há uma relação de herança entre atores ou casos de uso.

Essas notações ajudam a criar diagramas de caso de uso claros e compreensíveis, permitindo que os stakeholders visualizem e compreendam as interações entre os atores e os casos de uso em um sistema.

Figura 1– Casos de Uso

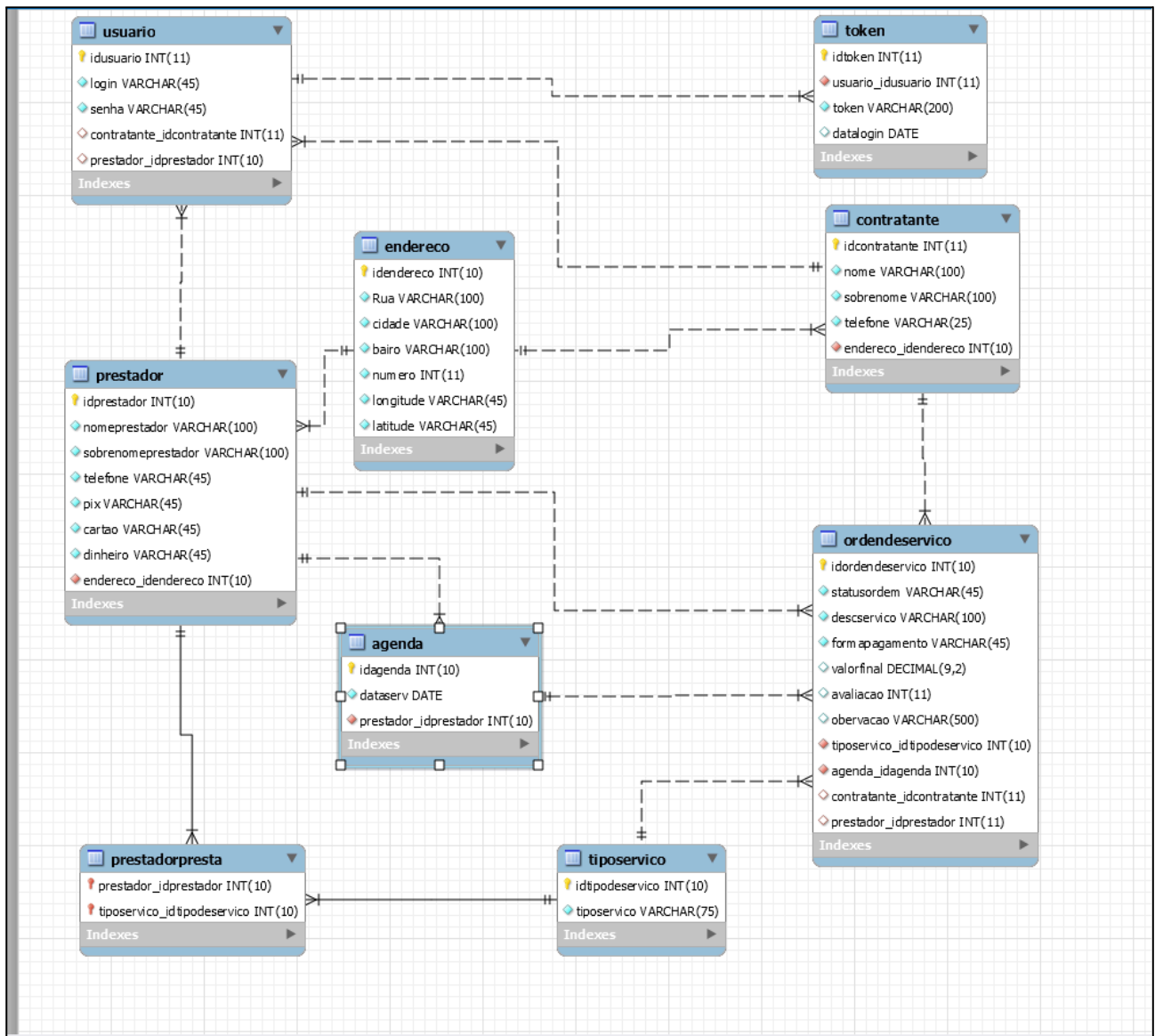


Fonte: elaborado pelo autor.

## 4.2 BANCO DE DADOS

Neste caso utilizado MySQL Workbench, banco de dados relacional, organiza e armazena dados em tabelas. Os dados são organizados em estruturas tabulares compostas por linhas (registros) e colunas (atributos ou campos). As tabelas em um banco de dados relacional são projetadas de forma a representar entidades do mundo real, e as relações entre as entidades são estabelecidas por meio de chaves primárias e chaves estrangeiras. Essas chaves são utilizadas para estabelecer vínculos entre tabelas, garantindo a integridade e consistência dos dados. Na Figura 2 é apresentado o modelo entidade relacional (MER) do banco de dados do sistema.

Figura 2 – Banco de dados



Fonte: elaborado pelo autor.

### 4.3 QUADRO DE MODELO DE NEGÓCIOS (CANVAS)

O Quadro de Modelo de Negócios, também conhecido como Business Model Canvas, é uma ferramenta visual desenvolvida pelo empreendedor suíço Alexander Osterwalder. O Sebrae utiliza essa metodologia para auxiliar empreendedores na construção e análise de modelos de negócios de forma estruturada.

O Canvas é composto por nove blocos, cada um representando um elemento chave do negócio, como segmento de clientes, proposta de valor, canais de distribuição, relacionamento com clientes, fontes de receita, recursos principais, atividades-chave, parcerias-chave e estrutura de custos. Essa abordagem visual permite uma compreensão holística do negócio, facilitando a identificação de oportunidades e desafios.

Ao utilizar o Quadro de Modelo de Negócios do SEBRAE, pode-se mapear e planejar elementos cruciais para o sucesso de empreendimentos, promovendo uma visão clara e integrada do funcionamento do negócio. Na Figura 3 é apresentado o quadro de modelo de negócios canvas do nosso sistema.

Figura 3 – Quadro de modelo de negócios canvas



Fonte: elaborado pelo autor.

#### 4.4 PESQUISA DE MERCADO

Foi realizada uma pesquisa qualitativa na sede do CEDUP Timbó, entre as datas de 20-10-2023 e 27-10-2023, que obteve duas respostas de Prestadores de Serviços. As informações estão sendo listadas apenas de caráter informativo pois devido a amostragem ser baixa, os dados não foram considerados no trabalho.

Das respostas obtidas um dos respondentes é fotógrafo e a outra pessoa trabalha como engenheira de segurança no trabalho.

O fotógrafo informou que programa seus compromissos mensalmente, sendo a maior dificuldade para agendar e a falta de praticidade, descreveu que utiliza o computador para fazer o gerenciamento da sua agenda. Ao ser perguntado sobre o que é mais importante para ele em um agenda online respondeu facilidade de uso, relatou que sua satisfação com as agendas online que já utilizou é alta, e acredita que uma agenda online melhora a sua produtividade, quanto a questão da segurança dos seus dados ela informa ser muito importante assim como telas fáceis de usar. Questionado se preferia ter uma versão paga ou uma gratuita com anúncios, escolheu a gratuita.

O engenheiro informou que programa seus compromissos diariamente, sendo a maior dificuldade a execução e conseguir realizar este agendamento, descreveu que utiliza o celular para fazer o gerenciamento da sua agenda, ao ser perguntado sobre o que é mais importante para ele em um agenda online respondeu personalização. Relatou que sua satisfação com as agendas online que já usou é bem baixa, e acredita que uma agenda online melhora a sua produtividade. Sobre a questão da segurança dos seus dados ela informa ser muito importante assim como telas fáceis de usar. Ao ser questionado se preferia ter uma versão paga ou uma gratuita com anúncios escolheu a gratuita.

Quanto a pesquisa quantitativa realizada na sede do CEDUP Timbó entre 20 e 27 de outubro de 2023, para quem busca o serviço, fornece insights valiosos sobre as preferências do público em relação ao agendamento de serviços online. Vamos analisar os principais pontos destacados pela pesquisa:

- **Faixa Etária Predominante:** A pesquisa identificou que a maioria dos respondentes (63,3%) está na faixa etária de 18 a 28 anos, e somando com a segunda colocada no ranking (29 a 38 anos), abrange 86,6% das respostas. Isso sugere que o público-alvo predominante para serviços de agendamento online é composto por jovens adultos.
- **Gênero:** O gênero feminino representa a maioria das respostas, com 58,3% das pessoas que responderam à pesquisa. Essa informação é relevante para a estratégia de marketing e comunicação, considerando as preferências de gênero.
- **Localização Geográfica:** A maioria dos entrevistados (75%) reside no Vale do Itajaí, com a maioria concentrada na cidade de Timbó (56,7%). Isso indica a importância de focar nas

necessidades e preferências dos moradores locais ao projetar serviços de agendamento online.

- Canais de Busca de Serviços: Cerca de 60% das pessoas procuram prestadores de serviços online por meio de plataformas como o Facebook. Além disso, 58,3% dos entrevistados buscam indicações de amigos ou familiares para contratar serviços. Isso sugere que a confiança e a recomendação boca a boca desempenham um papel significativo na escolha dos prestadores.
- Tipos de Serviços Contratados: Serviços gerais são os mais contratados, com 68,3% das respostas, seguidos por serviços de elétrica com 15%. Isso fornece informações valiosas para a seleção de serviços que podem ser oferecidos em uma plataforma de agendamento.
- Potencial de Mercado: Cerca de 60% das pessoas nunca utilizaram um site de agendamento. Isso sugere um grande mercado potencial a ser explorado, já que muitos ainda não estão familiarizados com esse método de agendamento.
- Canais de Agendamento Utilizados: A pesquisa identificou que os canais mais utilizados para agendamento online incluem WhatsApp, Facebook, e serviços como cabeleireiro, consultas médicas, eletricitista, aulas de inglês, cortador de grama e entrevistas de emprego. Isso mostra a diversidade de serviços que podem se beneficiar de uma plataforma de agendamento online.
- Frequência de Uso: A maioria das pessoas (55%) utiliza serviços de agendamento online raramente, enquanto 18,3% nunca usaram e 18,3% utilizam mensalmente. Isso destaca a necessidade de tornar o processo de agendamento mais acessível e atraente para todos os tipos de usuários.
- Fatores de Escolha: A facilidade de uso é o fator mais importante na escolha de um site de agendamento, citado por 63% dos entrevistados. Comentários de outras pessoas também são relevantes, com 60% mencionando-os como um fator de influência.
- Experiências Negativas: A maioria das pessoas (90%) não teve experiências negativas ao usar sites de agendamento. Para aqueles que tiveram, as principais queixas incluem complicações, dificuldades no entendimento e no agendamento, travamentos e dificuldades em desmarcar.
- Informações Relevantes: Os respondentes destacaram a disponibilidade do prestador e avaliações como as informações mais importantes na hora da contratação. O valor cobrado também é mencionado por 19 pessoas.
- Preferência no Agendamento: A maioria prefere agendar com base na facilidade de escolher data e horário (76,7%), enquanto 21% preferem a disponibilidade.



- Métodos de Pagamento: A pesquisa mostrou que 45% das pessoas preferem pagar por Pix, 26,7% no cartão de crédito e 20% no cartão de débito. Essa informação é útil para a integração de métodos de pagamento em uma plataforma de agendamento.
- Interesse em Participação Beta: 45% das pessoas expressaram interesse em participar de uma versão beta para fornecer feedback sobre o site, o que pode ser uma estratégia eficaz para o desenvolvimento de uma plataforma.
- Motivos para Usar Agendamento Online: A facilidade é o motivo principal para usar o agendamento online (39,4%), seguido pela praticidade (21,5%). Esses fatores podem ser enfatizados na promoção do serviço.
- Obstáculos para Não Usar Agendamento Online: Para aqueles que não utilizam o agendamento online, os principais obstáculos incluem comprometimento do prestador de serviço, dificuldade em encontrar serviços, falta de necessidade e preocupações com segurança.
- Importância das Avaliações: Um total de 20% das pessoas considera as avaliações dos prestadores como muito importantes, ressaltando a relevância da confiança na escolha de serviços.

## 5 DESENVOLVIMENTO

A crescente digitalização da economia e a transformação da maneira como os consumidores buscam e adquirem serviços têm impulsionado a necessidade de soluções tecnológicas que atendam a essa mudança de paradigma. Prestadores de serviços, que desempenham um papel vital na economia, muitas vezes encontram desafios ao promoverem suas ofertas e ao gerenciarem suas agendas de maneira eficiente. Ao mesmo tempo, os clientes que buscam por serviços específicos frequentemente enfrentam obstáculos na busca por profissionais qualificados e na realização de agendamentos convenientes.

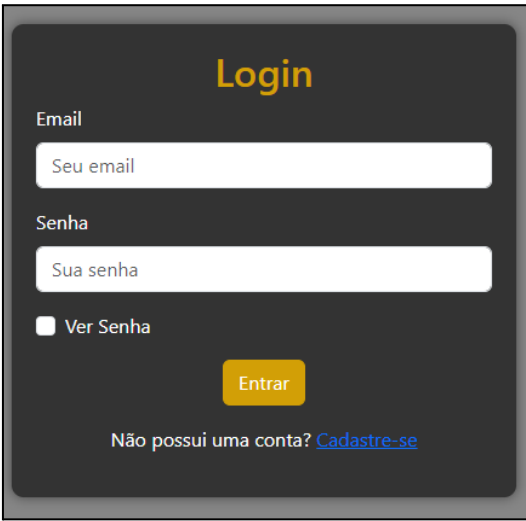
Nossa plataforma se concentra em atender a duas principais partes interessadas:

- Os Prestadores de Serviços poderão utilizar para:
  - Registrar e promover seus serviços, destacando suas habilidades e especializações.
  - Gerenciar suas agendas de forma eficiente, permitindo uma melhor alocação de tempo.
  - Manter um registro de clientes e histórico de serviços prestados.
- Os contratantes podem se beneficiar das seguintes funcionalidades:
  - Pesquisar e explorar uma variedade de serviços oferecidos por prestadores na plataforma.

- Filtrar resultados por distância geográfica, tornando mais fácil encontrar profissionais próximos.
- Escolher os profissionais com base nas preferências de pagamento, permitindo métodos como cartões de crédito, dinheiro, entre outros.
- Agendar compromissos por conveniência.

Para alcançar esses objetivos, a plataforma contará com um sistema de autenticação. Inicialmente na tela de Login, deve-se informar e-mail e senha cadastrados, demonstrado na Figura 4, permitindo que cada usuário acesse seu perfil e as funcionalidades relevantes.

Figura 4 – Login

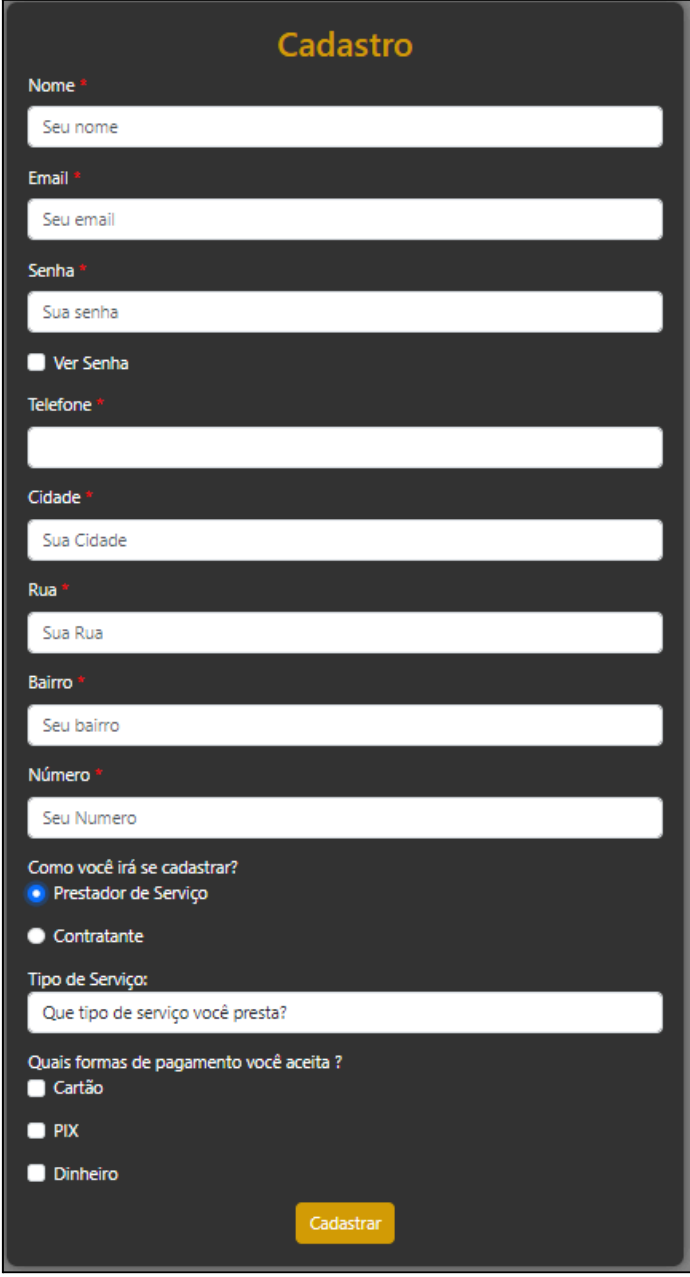


A imagem mostra uma interface de login com o título "Login" em amarelo. Abaixo dele, há dois campos de entrada: "Email" com o placeholder "Seu email" e "Senha" com o placeholder "Sua senha". Abaixo do campo de senha, há uma opção "Ver Senha" com um ícone de olho. Um botão amarelo "Entrar" está centralizado abaixo dos campos. No rodapé da interface, há o texto "Não possui uma conta? [Cadastre-se](#)".

Fonte: elaborado pelo autor.

Caso o usuário, seja ele contratante ou prestador não possua cadastro, visualiza-se na tela de login a seguinte frase “Não possui conta? Cadastre-se”, clicando na palavra “Cadastre-se” será redirecionado a Tela de cadastro. Nessa é necessário preencher todos os campos que possuem “\*” em vermelho, e por fim selecionar se for prestador de serviço. Ainda é necessário informar tipo de serviço que presta e formas de pagamento, demonstrado na Figura 5.

Figura 5 – Cadastro Prestador



O formulário de cadastro para prestador de serviço é apresentado em um layout vertical sobre um fundo escuro. No topo, o título "Cadastro" aparece em uma cor amarela vibrante. Abaixo dele, há uma série de campos de entrada brancos, cada um precedido por um rótulo em cinza claro. Os campos obrigatórios são marcados com um asterisco vermelho. Os campos incluem: Nome, Email, Senha (com uma opção para "Ver Senha"), Telefone, Cidade, Rua, Bairro e Número. Após os campos de endereço, há uma seção para "Como você irá se cadastrar?" com duas opções de rádio: "Prestador de Serviço" (selecionada) e "Contratante". Segue-se um campo para "Tipo de Serviço:" com o placeholder "Que tipo de serviço você presta?". Na base do formulário, há uma pergunta "Quais formas de pagamento você aceita?" com três opções de caixa de seleção: "Cartão", "PIX" e "Dinheiro". Um botão amarelo com o texto "Cadastrar" está posicionado no canto inferior direito.

**Cadastro**

Nome \*  
Seu nome

Email \*  
Seu email

Senha \*  
Sua senha  
☐ Ver Senha

Telefone \*

Cidade \*  
Sua Cidade

Rua \*  
Sua Rua

Bairro \*  
Seu bairro

Número \*  
Seu Numero

Como você irá se cadastrar?  
☒ Prestador de Serviço  
☐ Contratante

Tipo de Serviço:  
Que tipo de serviço você presta?

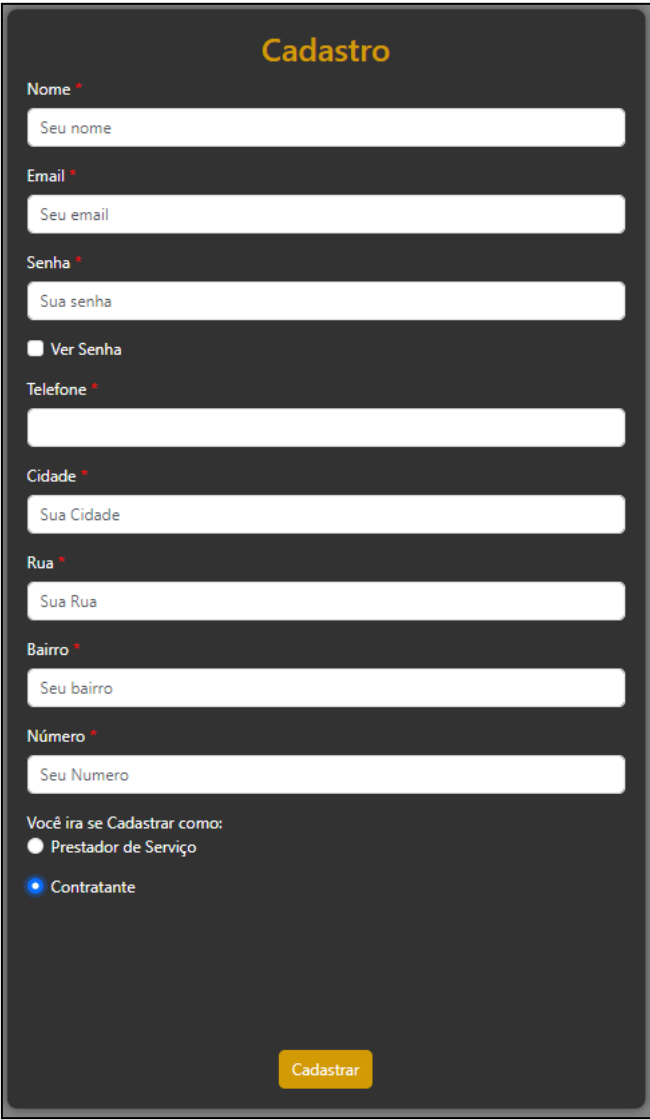
Quais formas de pagamento você aceita ?  
☐ Cartão  
☐ PIX  
☐ Dinheiro

Cadastrar

Fonte: elaborado pelo autor.

Ainda na tela de cadastro, caso for contratante, após o preenchimento de todos os dados solicitados obrigatórios (contém o “\*” vermelho), é necessário apenas clicar em cadastrar, demonstrado na Figura 6.

Figura 6 – Cadastro Contratante



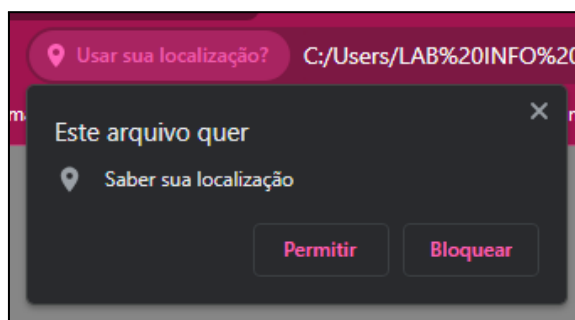
O formulário de cadastro para contratante possui o seguinte layout:

- Cadastro** (título em amarelo)
- Nome \*** (campo de texto com placeholder "Seu nome")
- Email \*** (campo de texto com placeholder "Seu email")
- Senha \*** (campo de texto com placeholder "Sua senha")
- ☐ Ver Senha
- Telefone \*** (campo de texto)
- Cidade \*** (campo de texto com placeholder "Sua Cidade")
- Rua \*** (campo de texto com placeholder "Sua Rua")
- Bairro \*** (campo de texto com placeholder "Seu bairro")
- Número \*** (campo de texto com placeholder "Seu Numero")
- Você ira se Cadastrar como:**
  - ☐ Prestador de Serviço
  - ☒ Contratante
- Cadastrar** (botão amarelo)

Fonte: elaborado pelo autor.

Para que o cadastro seja efetivado com sucesso, seja contratante ou prestador, é necessário além de preencher todos os campos com “\*” vermelho, permitir a localização, conforme Figura 7.

Figura 7 – Permitir localização



Fonte: elaborado pelo autor.

O Quadro 2 apresenta um código Java que implementa um controlador (*Controller*) em uma aplicação *Spring Boot*.

- *@RestController*: Indica que esta classe é um controlador Spring que lida com requisições HTTP e retorna respostas JSON.
- *@CrossOrigin("\*")*: Permite requisições de qualquer origem.
- *@PostMapping()*: Mapeia solicitações HTTP POST para este método.
- Verifica se o login fornecido já existe no banco de dados (*new UsuarioDAO().findbylogin(dto.login)*).
- Se não existir, adiciona um novo endereço, contratante e login associado no banco de dados.

No geral, este controlador adiciona novos contratantes verificando a existência de logins e gerenciando as operações no banco de dados por meio de classes DAO e conversores.

Quadro 2– Trecho do código fonte *backend* - Tela de cadastro contratante

```
public ResponseEntity<HttpStatus> addcontratante(@RequestBody ContratanteDTO dto)
throws SQLException {

    final EnderecoConverter converterE = new EnderecoConverter();
    final ContratanteConverter converterC = new ContratanteConverter();
    final UsuarioConverter converterU = new UsuarioConverter();
    if(!new UsuarioDAO().findbylogin(dto.login)){
        EnderecoEntity enderecoEntity =new
EnderecoDAO().addendereco(converterE.toEntity(dto));
        ContratanteEntity contratanteEntity= new
ContratanteDAO().addcontratante(converterC.toEntity(dto,enderecoEntity));
        new UsuarioDAO().addlongincont(converterU.toEntity(dto,contratanteEntity));
        return ResponseEntity.ok().body(HttpStatus.CREATED);
    }
    return ResponseEntity.ok().body(HttpStatus.UNPROCESSABLE_ENTITY);
}
```

Fonte: elaborado pelo autor.

O Quadro 3 apresenta um código Java que implementa um controlador (*Controller*) em uma aplicação *Spring Boot*, dedicado a operações relacionadas a prestadores de serviços. Em resumo, este controlador trata da adição de novos prestadores de serviços, gerenciando operações no banco de dados e verificando a existência de logins.

Quadro 3– Trecho do código fonte *backend* - Tela de cadastro prestador

```
public ResponseEntity<HttpStatus> addconprestador(@RequestBody PrestadorDTO dto)
throws SQLException {
```

Fonte: elaborado pelo autor.

O Quadro 4 apresenta um código *JavaScript* que lida com eventos associados ao formulário de cadastro em uma aplicação *web*.

- Verifica se o navegador suporta geolocalização (*"geolocation" in navigator*).
- Se suportado, obtém a localização atual do usuário (latitude e longitude).
- Exibe modais (janelas *pop-up*) informando sobre o sucesso ou falha do cadastro.

Em resumo, este *script* cuida da interação do usuário com o formulário de cadastro, realiza validações, modificações na página e envia dados ao servidor para o processo de cadastro.

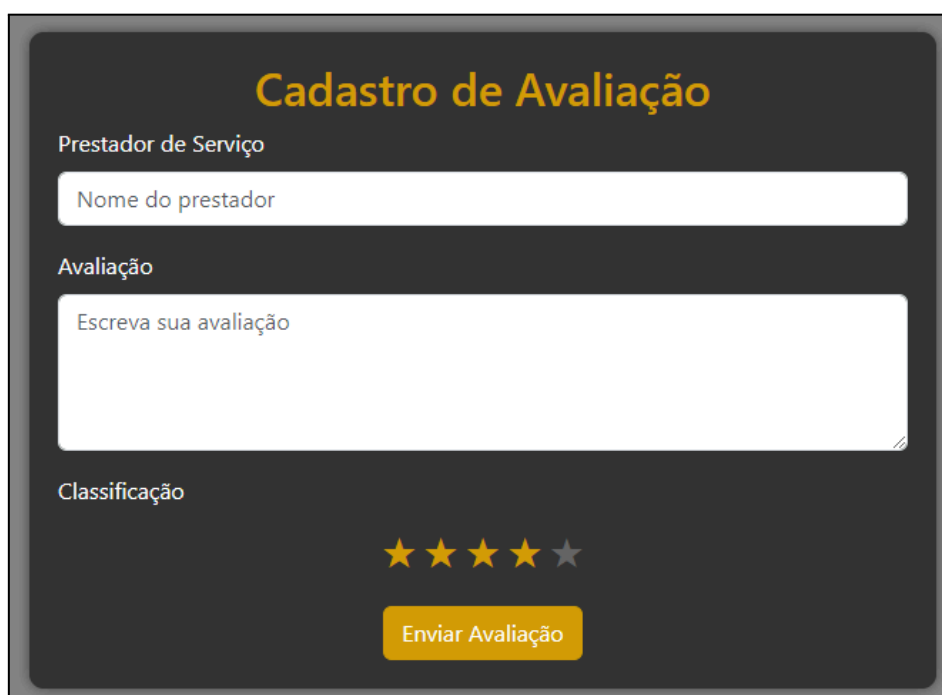
Quadro 4 – Trecho do código fonte *frontend*- Tela de cadastro

```
if ("geolocation" in navigator) {  
    navigator.geolocation.getCurrentPosition(function (position) {
```

Fonte: elaborado pelo autor.

Na Figura 8, para cumprir mais um objetivo, foi criado a tela de avaliação, na qual, quando o serviço é finalizado é possível avaliá-lo. O nome do prestador já é preenchido automaticamente, é opcional escrever a avaliação e de acordo com a quantidade de estrelas selecionadas informa-se a satisfação ou insatisfação com o serviço.

Figura 8– Cadastro de Avaliação



Fonte: elaborado pelo autor.

Para a contratação do serviço, o contratante terá a opção de escolha do serviço a ser realizado, fazer uma busca pela distância, data e forma de pagamento. Na Figura 9 demonstra-se a tela do agendamento, que é constituída de cinco *inputs*, que são os filtros para busca de um prestador e posteriormente agendamento. São eles: tipo de serviço, serviço a ser realizado, distância(km), data e forma de pagamento.

Figura 9 – Agendar Serviço



O formulário, intitulado "Agendar serviço", possui um fundo cinza escuro. Ele contém cinco campos de entrada brancos e um botão amarelo. Os campos são: 1. Um menu suspenso com o texto "Escolha o tipo de serviço" e uma seta para baixo. 2. Um campo de texto com o texto "Serviço à ser realizado". 3. Um campo de texto com o texto "Distância máxima". 4. Um campo de data com o formato "dd/mm/aaaa" e um ícone de calendário. 5. Um menu suspenso com o texto "Escolha uma forma de Pagamento" e uma seta para baixo. Abaixo dos campos, há um botão amarelo com o texto "Buscar prestadores".

## Agendar serviço

Escolha o tipo de serviço \*

Escolha o tipo de serviço

Serviço à ser realizado \*

Serviço a ser realizado

Agendar por distância (km) \*

Distância máxima

Escolha a data \*

dd/mm/aaaa

Escolha uma forma de Pagamento \*

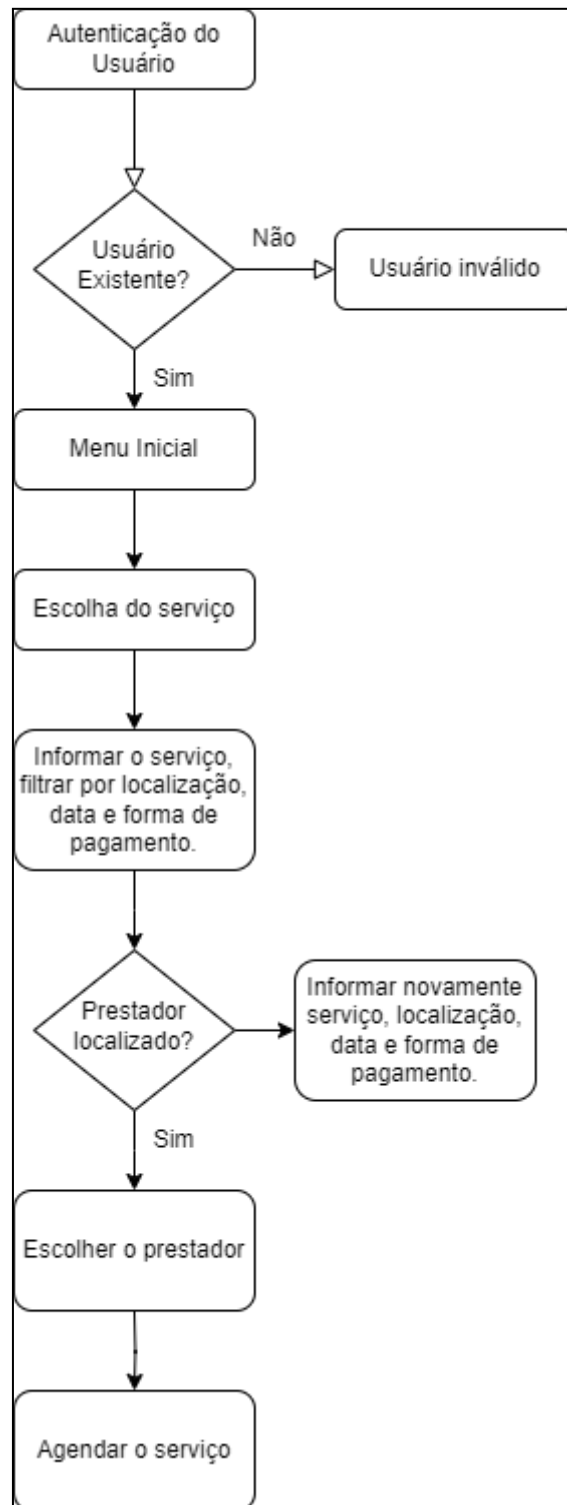
Escolha uma forma de Pagamento

Buscar prestadores

Fonte: elaborado pelo autor.

Na Figura 10, o fluxograma que explica o passo a passo para fazer essa contratação desde o login na plataforma. Primeiro a autenticação do usuário, se ele não existe é informado por uma mensagem, se ele existe prossegue-se para o menu inicial e já é possível agendar o serviço clicando em agendar serviço, e nessa tela os filtros para a busca do prestador.

Figura 10 – Fluxograma de Contratação



Fonte: elaborado pelo autor.

O Quadro 5 apresenta um código *JavaScript* que realiza as seguintes operações::

- **Inicialização do Mapa:**
  - Utiliza a *API* do *Google Maps* para inicializar um mapa na página, centrado nas coordenadas do contratante.



- Adiciona marcadores no mapa para os prestadores de serviço.
- Evento de Submissão do Formulário “agendarservico”:
  - Ao submeter o formulário para agendar um serviço, verifica se a data selecionada é posterior à data atual.
  - Chama a função “*getPretadores*” para obter prestadores disponíveis com base nos critérios especificados.
  - Exibe um modal informando sobre a disponibilidade de prestadores.
- Função “agendarservico”:
  - Chama a função “*postOrdendeservico*” para agendar um serviço com base nas informações fornecidas.
  - Exibe modais informando sobre o resultado da operação (sucesso ou falha).

Essencialmente, o código gerencia a interação do usuário com a página, realiza chamadas assíncronas para obter dados do contratante e prestadores, exibe informações no mapa e modais, e possibilita o agendamento de serviços.

Quadro 5– Trecho do código fonte *frontend*- Tela de agendar serviço

```
var map;

function initMap(latitudecontratante, longitudecontratante, prestadores) {
  if (latitudecontratante != "" || longitudecontratante != "") {

    var mapOptions = {
      center: { lat: parseFloat(latitudecontratante), lng:
parseFloat(longitudecontratante) },
      zoom: 13,
    };

    map = new google.maps.Map(document.getElementById('map'), mapOptions);

    if (prestadores && prestadores.length > 0) {
      prestadores.forEach(prestador => {
        const marker = new google.maps.Marker({
          position: { lat: parseFloat(prestador.lat), lng:
parseFloat(prestador.lng) },
          title: prestador.nome,
          map: map
        });

        let infoWindow = new google.maps.InfoWindow({
          content: '<div id="teste">' + prestador.id + '<h2>' +
prestador.nome + '</h2>' + '<h2>' + prestador.sobrenome + '</h2>' + '<h2>' +
prestador.nota + '</h2>' + ' <button type="button" data-bs-dismiss="modal"
aria-label="Close" onclick="selecionar(this)">Selecionar Prestador</button></div>'
        });
        marker.setMap(map);

        marker.addListener('click', () => {
          infoWindow.open(map, marker);
        });
      });
    }
  }
}
```

```

    });
  }
}

getPretadores(data, formadepagamento, tipoServico, distancia,
latitudecontratante, longitudecontratante).then(prestadores => {
  if (prestadores == null) {
    const myModal = new
bootstrap.Modal(document.getElementById('modalnadaencontrado'), {}));
    myModal.show();
    document.getElementById('Filtrar').disabled = false;
    document.getElementById("spinner").style.display = 'none';
  } else {
    const myModal = new bootstrap.Modal(document.getElementById('modalp'),
{}));
    myModal.show();
    initMap(latitudecontratante, longitudecontratante, prestadores);
  }
});
});
});
});

```

Fonte: elaborado pelo autor.

O Quadro 6 apresenta um código Java que implementa uma aplicação Spring Boot e realiza a pesquisa de prestadores de serviços com base em diversos parâmetros.

- DAO (“pesquisarprestadores”):
  - Executa uma chamada de procedimento armazenado (“BuscarPrestadores”) no banco de dados, passando os parâmetros fornecidos.
  - O procedimento armazenado retorna um conjunto de resultados com informações sobre os prestadores de serviços.
  - Para cada linha no conjunto de resultados, cria instâncias de “*PrestadorEntity*”, popula seus atributos com os dados do banco e adiciona à lista de resultados.
  - Retorna a lista de entidades como resultado da pesquisa.

Em resumo, esse código trata da pesquisa de prestadores de serviços com base em diferentes critérios, utilizando um procedimento armazenado no banco de dados para realizar a busca e retornando os resultados no formato de objetos DTO.

Quadro 6– Trecho do código fonte *backend* método DAO - Tela de agendar serviço

```

DAO
public List<PrestadorEntity> pesquisarprestadores(LocalDate date, int distancia, int
tiposervico, String pagamento, String lat, String lng) throws SQLException {
  String sql= "{call BuscarPrestadores(?, ?, ?, ?, ?, ?)}";
  try (final PreparedStatement preparedStatement =
connectionSingleton.getConnection().prepareStatement(sql)) {
    preparedStatement.setDouble(1, Double.parseDouble(lat));
    preparedStatement.setDouble(2, Double.parseDouble(lng));
    preparedStatement.setInt(3, distancia);
    preparedStatement.setInt(4, tiposervico);
  }
}

```

```

preparedStatement.setString(5,pagamento);
preparedStatement.setDate(6, Date.valueOf(date));
try (final ResultSet rs = preparedStatement.executeQuery()) {

    List<PrestadorEntity> resultado= new ArrayList<>();
    while (rs.next()){
        PrestadorEntity prestador = new PrestadorEntity();
        prestador.id=rs.getInt(1);
        prestador.nome= rs.getString(2);
        prestador.sobrenome= rs.getString(3);
        prestador.enderecoEntity= new EnderecoEntity();
        prestador.enderecoEntity.idendereco= rs.getInt(4);
        prestador.enderecoEntity.lng= rs.getString(5);
        prestador.enderecoEntity.lat= rs.getString(6);
        prestador.idagenda= rs.getInt(7);
        prestador.nota= rs.getInt(8);
        resultado.add(prestador);
    }
    return resultado;
}
}
}

```

Fonte: elaborado pelo autor.

No Quadro 7 a *procedure* do banco de dados que é um comando SQL responsável pelos filtros de tipo de serviço, serviço a ser realizado, distância, data e forma de pagamento.

Quadro 7 - Procedure

```

DELIMITER //

CREATE PROCEDURE BuscarPrestadores(
    IN latitude_usuario DECIMAL(10, 6),
    IN longitude_usuario DECIMAL(10, 6),
    IN distancia_maxima INT,
    IN tipo_servico_id INT,
    IN forma_pagamento VARCHAR(50),
    IN data_servico DATE
)
BEGIN
    SELECT
        p.idprestador,
        p.nomeprestador,
        p.sobrenomeprestador,
        e.idendereco,
        e.longitude,
        e.latitude,
        MIN(disponibilidade.idagenda) AS idagenda,
        COALESCE(ROUND(AVG(o_avaliacao.avaliacao)), 0) AS media_avaliacao
    FROM
        prestador p
    INNER JOIN
        prestadorpresta pp ON p.idprestador = pp.prestador_idprestador
    JOIN
        endereco e ON p.endereco_idendereco = e.idendereco
    LEFT JOIN (
        SELECT
            a.prestador_idprestador,
            a.dataserv,
            a.idagenda,
            COUNT(a.idagenda) AS total_agendas,
            COUNT(o.idordendeservico) AS total_ordens
        FROM

```

```

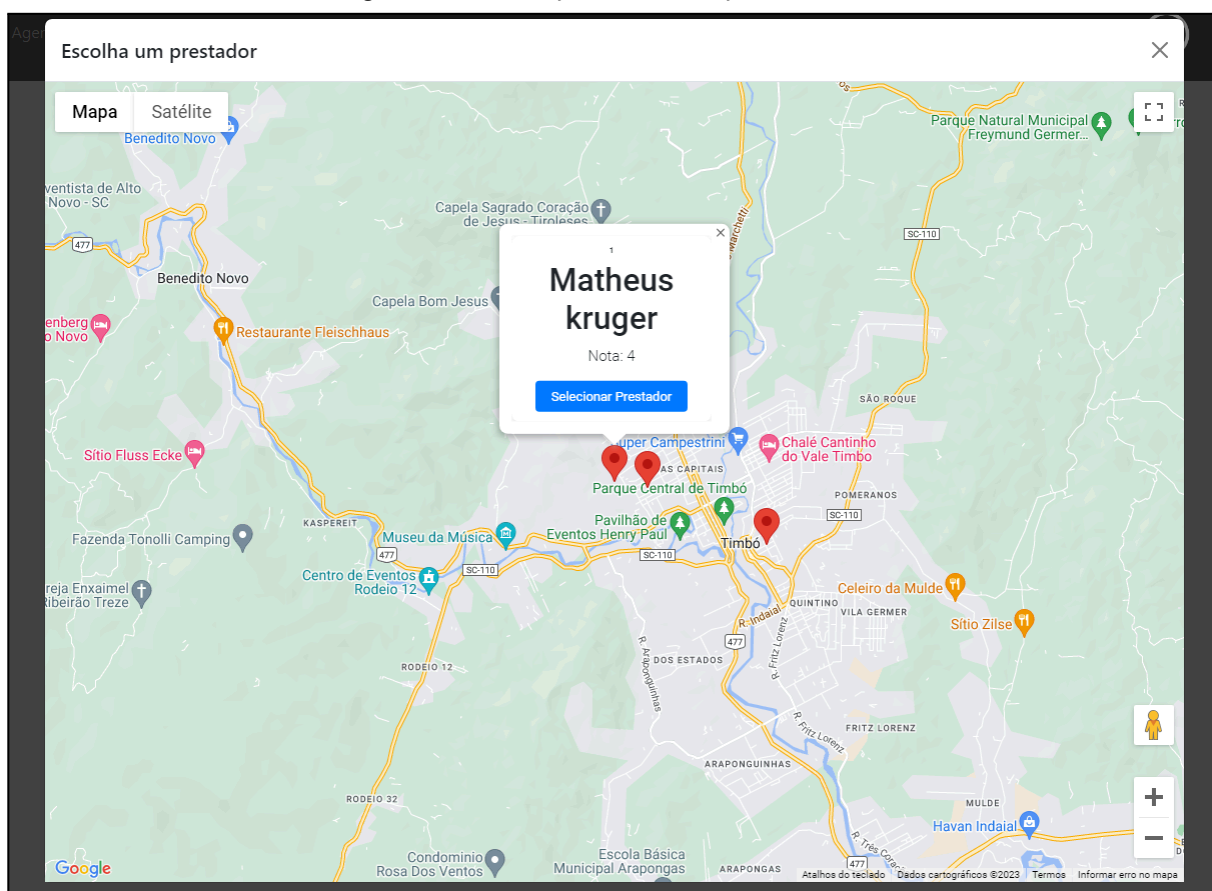
        agenda a
    LEFT JOIN
        ordendeservico o ON a.idagenda = o.agenda_idagenda
    WHERE
        a.dataserv = data_servico
    GROUP BY
        a.prestador_idprestador, a.dataserv, a.idagenda
) disponibilidade ON p.idprestador = disponibilidade.prestador_idprestador
LEFT JOIN ordendeservico o ON disponibilidade.idagenda = o.agenda_idagenda
LEFT JOIN (
    SELECT
        a.prestador_idprestador,
        AVG(o.avaliacao) AS avaliacao
    FROM
        agenda a
    LEFT JOIN
        ordendeservico o ON a.idagenda = o.agenda_idagenda
    GROUP BY
        a.prestador_idprestador
) o_avaliacao ON p.idprestador = o_avaliacao.prestador_idprestador
WHERE
    pp.tiposervico_idtipodeservico = tipo_servico_id AND
    (
        (forma_pagamento = 'dinheiro' AND p.dinheiro = 'sim') OR
        (forma_pagamento = 'pix' AND p.pix = 'sim') OR
        (forma_pagamento = 'cartao' AND p.cartao = 'sim')
    ) AND
    (
        disponibilidade.total_ordens < disponibilidade.total_agendas
    ) AND
    (
        6371 * ACOS(
            COS(RADIANS(latitude_usuario)) * COS(RADIANS(e.latitude)) *
COS(RADIANS(longitude_usuario) - RADIANS(e.longitude)) +
            SIN(RADIANS(latitude_usuario)) * SIN(RADIANS(e.latitude))
        )
    ) <= distancia_maxima
    GROUP BY
        p.idprestador, p.nomeprestador, p.sobrenomeprestador, e.idendereco,
e.longitude, e.latitude;
END //
DELIMITER ;

```

Fonte: elaborado pelo autor.

Na figura 11, como funcionará a API do Google Maps, que é utilizada no filtro de localização do prestador. Quando ele efetua o cadastro e informa o endereço, a API busca automaticamente esse endereço com as coordenadas de latitude e longitude, assim fixando um marcador. Clicando no marcador abrirá uma tela chamada de *infowindow* que mostrará o nome e avaliação do prestador.

Figura 11 – Informações de localização



Fonte: elaborado pelo autor.

Na figura 12, a tela Meus serviços contratados demonstra os serviços contratados pelo usuário, a tabela identifica o código, prestador solicitado, telefone, serviço solicitado, o status se está aberto ou concluído, data do serviço e também a forma de pagamento selecionada. Ainda conta com o botão para cancelar o serviço ou o redirecionar para o WhatsApp do prestador

Figura 12 – Informações de serviços contratados

Meus Serviços Contratados							
<input type="checkbox"/> Incluir serviços realizados							
Código	Prestador	Telefone	Serviço	Status	Data	Forma de pagamento	Ações
47	Matheus kruger	(47) 98859-0595	testep	aberto	28/11/2023	pix	<div>Cancelar</div> <div>WhatsApp</div>

Fonte: elaborado pelo autor.

O Quadro 8 apresenta um código *JavaScript* que realiza as seguintes operações:

- `let numeroFormatado = fone.replace(/\D/g, "");`: Remove todos os caracteres não numéricos da variável `fone`, armazenando apenas os dígitos em `numeroFormatado`.
- `window.open("https://wa.me/55"+numeroFormatado);`: Abre uma nova janela do navegador para a URL do WhatsApp, com o número de telefone formatado (contido em `numeroFormatado`), e prefixado com "55" (código de país do Brasil). O usuário será redirecionado para o WhatsApp com o número pré-preenchido.

Quadro 8 – Trecho do código fonte redirecionamento WhatsApp

```
let numeroFormatado = fone.replace(/\D/g, '');  
window.open("https://wa.me/55"+numeroFormatado)  
}
```

Fonte: elaborado pelo autor.

Na figura 13, a tela Calendário com vagas disponíveis no mês, demonstra os dias de cada mês e a possibilidade do prestador adicionar a quantidade de vagas que terá em cada um deles, e se necessário editar, também há a opção.

Figura 13 – Calendário com vagas disponíveis no mês

01	02	03	04	05	06	07
1	5	5	5	5	5	5
08	09	10	11	12	13	14
5	5	7	7	5	5	5
15	16	17	18	19	20	21
5	5	5	5	5	5	5
22	23	24	25	26	27	28
4	5	5	7	0	5	5
29	30	31				
5	5	5				

Editar

Fonte: elaborado pelo autor.

O Quadro 9 apresenta um código *JavaScript* que está criando dinamicamente a estrutura HTML para representar visualmente um dia no calendário, exibindo o dia formatado e um campo de entrada para a quantidade de atendimentos que o prestador consegue atender associada a esse dia. Essa estrutura HTML é então atribuída ao `innerHTML` do elemento `dayElement`, e isso será renderizado na interface do usuário.

Quadro 9 – Trecho do código fonte calendário.

```
dayElement.innerHTML = `
  <div class="day h2">
    <p><strong>${formattedDate}</strong></p>
    <input type="number" class="form-control day-text" min="0"
value="${valorVaga}">
  </div>`;
```

Fonte: elaborado pelo autor.

O Quadro 10 apresenta um código *JavaScript* implementa como funciona a parte da edição de dias trabalhados, o código itera sobre as agendas editadas e atuais, verifica as diferenças nas

quantidades de vagas e realiza as operações de *insert* e *delete* no banco de dados para refletir essas alterações

Quadro 10 – Trecho do código fonte edição de dias trabalhados.

```
dayElement.innerHTML = `
  <div class="day h2">
    <p><strong>${formattedDate}</strong></p>
    <input type="number" class="form-control day-text" min="0"
value="${valorVaga}">
  </div>`;
```

Fonte: elaborado pelo autor.

Na figura 14, temos uma amostra de como fica no banco de dados as inserções no calendário, como no exemplo o prestador informa que no dia 15-11 ele terá disponível apenas duas vagas e feito dois *inserts* na tabela com a data e o id do prestador

Figura 14 – Informações do banco de dados sobre inserções

	idagenda	dataserv	prestador_idprestador
	1	2023-11-15	1
	3	2023-11-15	1
	4	2023-11-16	1
	5	2023-11-16	1

Fonte: elaborado pelo autor.

## 5.1 TÉCNICAS E FERRAMENTAS UTILIZADAS

Utilizando linguagem de programação em Java, por ser uma plataforma independente, conhecida por sua portabilidade, inclusive destacada por Cay S. Horstmann (2018) em seu livro, “Core Java Volume I -- Fundamentals” como uma de suas principais características, o que significa que um software desenvolvido em Java pode ser executado em diferentes sistemas operacionais, como Windows, Linux, etc.

Java é conhecida por sua ênfase na segurança, de acordo com a a Oracle, a empresa que mantém Java, fornece extensa documentação sobre os recursos de segurança em Java. O Java Security Guide, disponível no site oficial da Oracle, aborda os princípios de segurança em Java e fornece orientações sobre como desenvolver aplicativos seguros utilizando a plataforma Java. A plataforma possui recursos integrados de segurança, como o controle de acesso a recursos do sistema, verificação de tipos e gerenciamento de exceções. Esses recursos ajudam a proteger o software contra ameaças e vulnerabilidades, garantindo a integridade dos dados dos usuários.



É uma linguagem orientada a objetos, o que permite que organizemos o código de forma modular, reutilizar componentes e facilitar a manutenção e evolução do software. Segundo Erich Gamma, Richard Helm, Ralph Johnson e John Vlissides (2015) em seu livro “Design Patterns: Elements of Reusable Object-Oriented Software” Java promove o uso de padrões de design, como o padrão de projeto MVC (Model-View-Controller) e o padrão Observer, que ajudam a separar a lógica de negócios da interface do usuário e melhoram a modularidade do código. Esses padrões facilitam a manutenção e a evolução do software, permitindo alterações em uma parte do sistema sem afetar outras partes. Essa abordagem é particularmente útil, onde há diferentes entidades, como compromissos, tarefas, categorias e usuários, que podem ser modeladas como objetos.

Possui um suporte robusto para APIs (Interface de Programação de Aplicação) e integrações com outros sistemas. Onde pode ser necessário integrar com serviços externos, como utilizado a API do Google Maps para o filtro de localização. Através do suporte a APIs, é possível estabelecer a comunicação e a troca de dados de forma eficiente e segura. De acordo com a Java API Documentation, a documentação oficial da Oracle para a plataforma Java fornece detalhes sobre as APIs disponíveis para comunicação com serviços externos. Através de APIs como Java API for RESTful Web Services (JAX-RS) e Java Message Service (JMS), é possível estabelecer comunicação eficiente e segura com serviços externos.

Já o banco de dados, utilizou-se MySQL Workbench, de acordo com Baron Schwartz, Peter Zaitsev, Vadim Tkachenko e Jeremy D. Zawodny (2012) em seu livro "High Performance MySQL" que aborda aspectos do MySQL, incluindo otimização de desempenho, escalabilidade, estabilidade e confiabilidade, também discute práticas e técnicas para garantir um desempenho sólido e consistente do banco de dados, visto isso, o MySQL Workbench é utilizado em diversas aplicações, o que contribui para a sua reputação como uma escolha segura e sólida para armazenar dados.

De acordo com Michael Kruckenberg e Jay Pipes (2006), em seu livro Pro MySQL, que o explora com profundidade o banco mencionado, mostrando aspectos avançados de design do banco de dados, consultas otimizadas, segurança, também aborda que MySQL é conhecido por seu bom desempenho em aplicações que exigem consultas rápidas e eficientes. Ele é projetado para otimizar a recuperação e manipulação de dados, o que é especialmente relevante, onde é necessário acessar e atualizar informações de forma ágil.

Possui recursos de escalabilidade que permitem lidar com um grande volume de dados e um alto número de usuários simultâneos. O MySQL oferece opções para dimensionar o banco de dados e garantir o bom desempenho mesmo em cenários de alta demanda, considerando um crescimento do trabalho proposto, o MySQL vai suportar a demanda.

Compatível com várias linguagens de programação, incluindo Java, muitas outras. Possui também uma grande comunidade de usuários e desenvolvedores, o que significa que há uma ampla gama de recursos, documentação e fóruns disponíveis para obter suporte e solucionar problemas. A comunidade ativa contribui para a evolução e melhoria contínua do MySQL, fornecendo atualizações e correções de segurança. É uma opção de banco de dados de código aberto, o que significa que é possível utilizá-lo gratuitamente.

Segundo José Carlos Macoratti, em seu artigo “Visual Studio Code - Apresentando o editor multiplataforma da Microsoft”, o Visual Studio Code, frequentemente abreviado como VS Code, é um popular editor de código-fonte desenvolvido pela Microsoft. Possui algumas características notáveis, citadas inclusive na documentação da Microsoft:

- O VS Code é um software de código aberto e pode ser usado gratuitamente. Isso o torna acessível a uma ampla gama de desenvolvedores.
- Compatível com Windows, macOS e Linux, tornando-o uma escolha versátil para desenvolvedores em diferentes sistemas operacionais.
- Uma das características mais destacadas do VS Code é seu sistema de extensões. Os desenvolvedores podem personalizar o editor com extensões para atender às suas necessidades. Existem extensões para várias linguagens de programação, depuração, controle de versão e muito mais.
- O VS Code oferece integração nativa com o Git, tornando a gestão de controle de versão e colaboração mais fácil.
- Ele suporta depuração para várias linguagens, o que é útil para encontrar e corrigir erros em seu código.
- O VS Code inclui um terminal integrado, permitindo que os desenvolvedores executem comandos diretamente no ambiente de desenvolvimento.
- O editor oferece destaque de sintaxe para muitas linguagens de programação e recursos de autocompletar.
- A configuração do VS Code é geralmente simples e acessível por meio de arquivos JSON. Isso permite que os desenvolvedores personalizem facilmente o ambiente de desenvolvimento de acordo com suas preferências.
- Há uma comunidade ativa em torno do VS Code, o que significa que há muitos recursos, tutoriais e suporte disponíveis.

Em resumo, o Visual Studio Code foi uma escolha devido à sua flexibilidade, extensibilidade e suporte a várias linguagens de programação.

De acordo com Maurício Samy Silva, em seu artigo acadêmico “JavaScript - Guia do

Programador”, o JavaScript é uma linguagem de programação amplamente utilizada no desenvolvimento web. Alguns pontos-chave sobre JavaScript:

- É principalmente uma linguagem de script do lado do cliente, o que significa que é executada no navegador da web do usuário. Ela é usada para tornar as páginas da web interativas, respondendo a eventos do usuário, manipulando o DOM (Modelo de Objeto do Documento) e realizando chamadas assíncronas para servidores.
- Possui uma sintaxe relativamente flexível e é fácil de aprender para iniciantes. É uma linguagem de tipagem dinâmica, o que significa que as variáveis não precisam ser declaradas com tipos específicos.
- Possui um vasto ecossistema de bibliotecas e frameworks, como React, Angular, Vue.js e Node.js, que permitem o desenvolvimento de aplicativos web altamente sofisticados e escaláveis.
- Lida bem com operações assíncronas, o que é fundamental para o desenvolvimento de aplicativos web modernos que dependem de chamadas de API, manipulação de eventos e atualizações de interface do usuário em tempo real.
- É amplamente usado para criar interatividade em páginas web, validação de formulários, animações e muito mais. Também é a linguagem por trás da maioria dos jogos e aplicativos web.
- JavaScript tem uma comunidade de desenvolvedores altamente ativa e recursos educacionais abundantes, o que facilita o aprendizado e a resolução de problemas.

Quanto a API do Google Maps, de acordo com a plataforma oficial Google Maps, é uma coleção de serviços e recursos fornecidos pela Google para integração de mapas, geolocalização e informações de lugares em aplicativos e websites. Ela oferece uma variedade de funcionalidades para desenvolvedores que desejam incorporar mapas e recursos de localização em seus aplicativos. Aqui estão alguns aspectos-chave da API do Google Maps:

- A API do Google Maps permite que você insira mapas interativos em seu aplicativo ou site. Isso inclui recursos como zoom, pan, marcadores, polígonos e sobreposições.
- Geolocalização: pode usar a API para obter a localização atual do usuário, rastrear movimentos e calcular direções entre pontos.
- Sugestões de Lugares: A API fornece sugestões de lugares, permitindo que os usuários pesquisem locais, endereços e empresas com base em consultas de texto.
- Informações de Lugares: Com a API, pode-se acessar detalhes sobre locais, como horários de funcionamento, avaliações, fotos e muito mais.
- Serviço de Rotas: A API permite que você calcule rotas entre locais e forneça instruções

passo a passo para navegação.

- Integração com Street View: Pode -se incorporar imagens panorâmicas do Google Street View em seu aplicativo para mostrar vistas em 360 graus das localizações.
- Personalização de Mapas: É possível personalizar a aparência dos mapas, escolhendo estilos de mapa, cores e ícones personalizados.
- A API do Google Maps é usada por muitas empresas para criar aplicativos que dependem de informações de localização, como serviços de entrega, aplicativos de compartilhamento de carros, aplicativos de rastreamento e muito mais. O uso comercial pode implicar custos, dependendo da escala e do uso.
- A Google fornece documentação detalhada e suporte técnico para ajudar os desenvolvedores a usar a API de maneira eficaz.

Quanto à metodologia utilizada enfatiza a colaboração, a flexibilidade e a entrega incremental, ou seja, com várias etapas de processo. Primeiramente foi identificado o problema e o software criado empiricamente para que fosse resolvido, após definimos os requisitos, onde conversamos com um prestador de serviços para entender os maiores problemas que ocorriam quando se tratava de agendamentos e elaborado uma pesquisa de mercado que buscava o entendimento dos desafios tanto do contratante quanto do prestador, também o desenvolvimento dos casos de uso, e sucessivamente o diagrama de atividades, diagrama de classes, modelo de entidade e relacionamento do banco de dados, a programação em Java e por fim a apresentação do software, as atividades foram divididas entre os membros da equipe, mas por fim, foram analisadas por todos, chegando a um denominador comum.

## **6. CONCLUSÃO**

Ao longo deste projeto, foi possível projetar e desenvolver uma plataforma de agendamento que abrangesse não apenas a funcionalidade de marcação de compromissos, mas também um sistema para que os prestadores de serviços e clientes possam buscar em apenas um lugar serviços que precisem solicitar sem a necessidade de uma pesquisa em outras plataformas. Durante o desenvolvimento, enfatiza-se a simplicidade e a usabilidade, reconhecendo que esses são fatores cruciais para a adoção e sucesso de qualquer sistema.

O objetivo geral foi alcançado totalmente, pois a plataforma busca os profissionais, tanto por parte do contratante quanto do prestador de serviço, e após a finalização do serviço, é possível a avaliação. Ainda, o prestador faz o gerenciamento de sua agenda, verificando os agendamentos

solicitados e agendando com outros prestadores e pode, inclusive, cancelar, editar e entrar em contato direto com o contratante.

Dos quatro objetivos específicos, o primeiro citado foi alcançado totalmente, pois a plataforma permite que o usuário que deseja contratar um serviço, possa buscá-lo e agendá-lo.

Quanto ao segundo objetivo específico citado, o mesmo foi alcançado totalmente, pois o contratante tem a tela de agendamento com a opção de tipo de serviço, descrever o que deve ser realizado, buscar o prestador pela distância solicitada, filtrar a data que deseja e a forma de pagamento que vai utilizar.

O terceiro objetivo específico foi atingido totalmente, que fala sobre a avaliação de serviços após a finalização dele. Na tela dos serviços contratados o contratante tem acesso à um botão que direciona para a tela de avaliações, preenchendo automaticamente o nome do prestador do serviço e dá a possibilidade da avaliação descritiva não obrigatória e a pontuação de 5 estrelas, que segue o princípio de uma para péssimo, duas para ruim, três para razoável, quatro para bom e cinco para excelente.

Para o último objetivo específico proposto, o mesmo foi alcançado totalmente, trata-se do gerenciamento da agenda do prestador, onde por meio do seu login, é possível além de contratar outros serviços de outros prestadores, visualizar, cancelar, finalizar, entrar em contato com o cliente via WhatsApp, cadastrar a quantidade de vagas disponíveis no dia por meio de *slot* de tempo e visualizar e alterar dados do seu perfil.

Uma forma de melhorar a plataforma é implementar um sistema que ofereça uma visão mais clara da disponibilidade do prestador em tempo real. Isso permite que os clientes vejam instantaneamente os horários disponíveis e façam agendamentos de acordo com sua conveniência, evitando conflitos de horários.

Outra sugestão de extensão para a plataforma é a incorporação de recursos de agendamento não apenas de serviços, mas também outras necessidades, como compromissos individuais, reservas de salas de reuniões, equipamentos ou outros recursos. Além disso, integrar um sistema de pagamento direto na plataforma facilitaria a cobrança por serviços agendados, tornando-o uma solução completa para empresas que oferecem serviços por agendamento.

## REFERÊNCIAS

AGENDAI. Como uma agenda online pode ajudar profissionais autônomos. Disponível em: <<https://blog.agendaai.com.br/como-uma-agenda-online-pode-ajudar-profissionais-autonomos/>>.

Acesso em: 16 ago. 2023.

BARON SCHWARTZ; ZAITSEV, P.; VADIM TKACHENKO. High performance MySQL. Beijing: Cambridge [Mass, 2012.

BRITTON, B.K & TESSER, A. "Time Management and Psychological Well-Being among College Students" (Gerenciamento do Tempo e Bem-Estar Psicológico entre Estudantes Universitários). Journal of Educational Psychology, 1991.

CORIOLANO, Thiago Gustavo. A (des)regulamentação do trabalho autônomo pela lei 13467/17 e o perigo do apego a forma. 2019. Disponível em: <https://www.migalhas.com.br/depeso/317683/a--des-regulamentacao-do-trabalho-autonomo-pela-lei-13467-17-e-o-perigo-do-apego-a-forma>. Acesso em: 23 ago. 2023.

COVEY, S. R. 7 Habits Of Highly Effective People. [s.l.] Simon & Schuster Ltd, 1989.

COVEY, S. R.; ALBERTO CABRAL FUSARO; MÁRCIA DO CARMO FELISMINO FUSARO. Os 7 hábitos das pessoas altamente eficazes. Rio De Janeiro: Best Seller, 2011.

Documentação da Plataforma Google Maps | API Maps JavaScript. Disponível em: <<https://developers.google.com/maps/documentation/javascript?hl=pt-br> >. Acesso em: 9 nov. 2023.

FEIJÓ, Janaína. A quantidade de trabalhadores nessa categoria já ultrapassa o valor registrado no 3º tri de 2019 em mais de 1,3 milhão. 2022. Disponível em: <https://portal.fgv.br/artigos/empreender-sobreviver-quem-sao-trabalhadores-conta-propria>. Acesso em: 02 set. 2023.

GAMMA, E. et al. Design patterns : elements of reusable object-oriented software. New Dehli: Pearson Education, 2015.

HOLANDA, Isabel. A influência das redes sociais na comunicação humana. 2021. Disponível em: <https://blog.fortestecnologia.com.br/tecnologia-e-inovacao/a-influencia-das-redes-sociais>. Acesso em: 24 set. 2023.

HORSTMANN, C. S. Core Java Volume I--Fundamentals. [s.l.] Prentice Hall, 2018.

JACOBSON, I. (1994). The Object Advantage: Business Process Reengineering with Object Technology. Addison-Wesley Professional.

Java Platform, Standard Edition Java API Reference. Disponível em: <https://docs.oracle.com/en/java/javase/14/docs/api/index.html>. Acesso em: 08 nov. 2023.

JONES, S., & FOX, S. "The Impact of Electronic Personal Calendar Use on Time Management Skills" (O Impacto do Uso de Calendário Eletrônico Pessoal nas Habilidades de Gerenciamento do Tempo) Pew Research Center, 2009.

NIELSEN, J. (1993). Usability Engineering. Morgan Kaufmann.

OLIVEIRA, Sabrina Guedes de; ALMEIDA, Veronica Eloi de; TROTTA, Leonardo Monteiro. As tecnologias e o mundo globalizado: reflexões sobre o cotidiano contemporâneo. 2020. Disponível em:

<https://educacaopublica.cecierj.edu.br/artigos/20/2/as-tecnologias-e-o-mundo-globalizado-reflexoes-sobre-o-cotidiano-contemporaneo>. Acesso em: 20 nov. 2023.

PIPES, J.; KRUCKENBERG, M. Pro MySQL. [s.l.] Apress, 2006.

REDAÇÃO DO JORNAL GRANDE BAHIA (Brasil). Número de trabalhadores autônomos atinge recorde e chega a 25,7 milhões no Brasil, aponta IBGE. 2022. Disponível em: <https://jornalgrandebahia.com.br/2022/10/numero-de-trabalhadores-autonomos-atinge-recorde-e-chega-a-257-milhoes-no-brasil-aponta-ibge/>. Acesso em: 16 ago. 2023.

RUMBAUGH, J., Jacobson, I., & Booch, G. (2005). The Unified Modeling Language Reference Manual (2nd ed.). Addison-Wesley.

RUMBAUGH, J., & Blaha, M. (1991). Object-Oriented Modeling and Design. Prentice Hall.

SENNETT, R. The corrosion of character : the personal consequences of work in the new capitalism. New York: W.W. Norton, 1998.

SCHWAB, K. The Fourth Industrial Revolution: what it means, how to respond. Disponível em: <<https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/>>.

SEBRAE (Brasil). O que são negócios de impacto social e como eles funcionam. 2023. Disponível em:

[https://sebrae.com.br/sites/PortalSebrae/sebraeaz/o-que-sao-negocios-de-impacto-social,1f4d9e5d32055410VgnVCM1000003b74010aRCRD?utm\\_source=link+interno&utm\\_medium=referral&utm\\_campaign=tofu+passa+para+mofu&u=undefined&jaytkMais+Campus+-+Uniarp=E6Ewkn0BAA\\_CmizGSfQEAAAdg2NDg4ZGM4LTgwNmUfNDNIOS05YzkxLTgwN2M0NjI3OTJINQ](https://sebrae.com.br/sites/PortalSebrae/sebraeaz/o-que-sao-negocios-de-impacto-social,1f4d9e5d32055410VgnVCM1000003b74010aRCRD?utm_source=link+interno&utm_medium=referral&utm_campaign=tofu+passa+para+mofu&u=undefined&jaytkMais+Campus+-+Uniarp=E6Ewkn0BAA_CmizGSfQEAAAdg2NDg4ZGM4LTgwNmUfNDNIOS05YzkxLTgwN2M0NjI3OTJINQ) . Acesso em: 20 nov. 2023.

SILVA, M. S. JavaScript - Guia do Programador: Guia completo das funcionalidades de linguagem JavaScript. [s.l.] Novatec Editora, 2010.

SOUZA, Natan. UX e UI: conheça as semelhanças e diferenças entre ambos. Disponível em: <https://www.alura.com.br/artigos/ux-e-ui-conheca-as-semelhancas-e-diferencas-entre-ambos>. Acesso em: 24 set. 2023.

Sommerville, I. (2011). Engenharia de Software. Pearson.

TERRYGLEE. Funcionalidades do editor de códigos - Visual Studio (Windows). Disponível em: <<https://learn.microsoft.com/pt-br/visualstudio/ide/writing-code-in-the-code-and-text-editor?view=vs-2022>>.

Visual Studio Code - O Editor multiplataforma da Microsoft. Disponível em: <[https://www.macoratti.net/16/05/vs\\_code1.htm](https://www.macoratti.net/16/05/vs_code1.htm)>.

.