

# Mão De Obra Digital: Aplicativo de Gestão para Trabalhadores Autônomos da Construção Civil

Bruno Dion Correa dos Santos<sup>1</sup>, Isabelle Cordova Gomes <sup>1</sup>, Enrique Augusto da Roza<sup>1</sup>

<sup>1</sup>Centro Universitário Campo Real  
Rua Comendador Norberto, 1299 - Santa Cruz – Guarapuava – PR – Brasil  
{engs-brunosantos@camporeal.edu.br, prof\_isabellecordova@camporeal.edu.br,  
prof\_enriqueroza@camporeal.edu.br}

## Resumo

O setor da construção civil no Brasil tem se destacado como área de grande investimento e oportunidades, gerando empregos e ampliando o número de microempreendedores individuais. Apesar desse cenário, ainda predominam práticas manuais para elaboração de orçamentos, registro de horas trabalhadas, controle de serviços e acompanhamento financeiro, o que compromete a organização e a profissionalização da atividade. Para enfrentar esse desafio, este trabalho apresenta o Mão de Obra Digital, um aplicativo modelo SaaS (*Software as a Service*) baseado em arquitetura *Mobile Cloud Computing* (MCC), desenvolvido para apoiar a gestão operacional de profissionais autônomos da construção civil. A pesquisa, de caráter qualitativo, utilizou princípios de *Design Thinking*, contemplando etapas de empatia, definição, ideação, prototipagem e testes com o público-alvo. A solução foi implementada com *back-end* em Laravel e PostgreSQL e *front-end* em Flutter, oferecendo funcionalidades como cadastro de clientes e colaboradores, acompanhamento de serviços em andamento, geração de orçamentos em PDF e controle de despesas. A validação inicial foi realizada com cinco potenciais usuários ao longo de dois meses, esses usuários relataram facilidade de uso e clareza das funcionalidades propostas, reconhecendo potencial da ferramenta para melhorar a organização do trabalho diário e apresentação profissional diante de clientes.

**Palavras-chave:** Construção civil. Gestão operacional. Aplicativo móvel. Mobile Cloud Computing. Profissionais autônomos.

## Abstract

The civil construction sector in Brazil has emerged as a field of significant investment and opportunities, generating jobs and increasing the number of individual micro-entrepreneurs. Despite this scenario, manual practices still prevail in the preparation of budgets, recording of working hours, service control, and financial monitoring, which compromises organization and professionalization. To address this challenge, this study presents Mão de Obra Digital, a SaaS (Software as a Service) application based on Mobile Cloud Computing (MCC) architecture, developed to support the operational management of self-employed construction professionals. The qualitative research adopted Design Thinking principles, encompassing stages of empathy, definition, ideation, prototyping, and testing with the target audience. The solution was implemented with a Laravel and PostgreSQL back-end and a Flutter front-end, offering

functionalities such as client and collaborator registration, monitoring of ongoing services, generation of PDF budgets, and expense control. Initial validation was conducted with five potential users over two months, who reported ease of use and clarity of the proposed functionalities, recognizing the tool's potential to improve daily work organization and professional presentation to clients.

**Keywords:** Civil construction. Operational management. Mobile application. Mobile Cloud Computing. Self-employed professionals.

## 1. Introdução

O setor da construção civil no Brasil registrou expansão de 4,3% em 2024, alcançando um Produto Interno Bruto de R\$359,523 bilhões, segundo dados do IBGE (CBIC, 2025). Esse crescimento reflete diretamente na oferta de empregos e no aumento do número de empreendedores na área, incluindo microempreendedores individuais (MEI). Das 874 mil novas microempresas abertas no Brasil em 2024, 283,7 mil estavam ligadas ao setor da construção civil (SEBRAE, 2025).

Apesar desse cenário favorável, ainda se observam lacunas em digitalização e automação de processos, especialmente em atividades rotineiras como elaboração de orçamentos, registros de horas trabalhadas e controle financeiro, que são fundamentais para a sustentabilidade econômica desses profissionais (PALFINGER, 2025). Em um levantamento exploratório realizado em uma loja de tintas, identificou-se que cerca de 90% dos profissionais consultados elaboram orçamentos manualmente com papel e caneta ou via mensagens em aplicativos de conversação, enquanto apenas uma minoria utiliza ferramentas digitais como Microsoft Word, para gerar documentos em PDF. Além disso, o registro de dias e horas trabalhadas, assim como o controle de ganhos dos serviços em andamento, em grande parte são feitos em cadernos, o que aumenta o risco de perda de informações e dificulta a gestão financeira.

Diante desse contexto, este trabalho propõe o desenvolvimento de um aplicativo modelo SaaS (*Software as a Service*) para dispositivos Android, baseado em arquitetura *Mobile Cloud Computing* (MCC), aproveitando o alto índice de uso de *Smartphone* no país, 1,2 aparelhos por habitante em 2024 totalizando 258 milhões de dispositivos (FGV, 2024). O aplicativo oferece funcionalidades como cadastro de colaboradores, registro de dias ou horas de trabalho, emissão de orçamentos de mão de obra em PDF, geração de recibos para seus clientes, exibição de relatórios

de lucros e despesas operacionais. A adoção da arquitetura MCC visa assegurar leveza e acessibilidade, permitindo que profissionais com aparelhos de *hardware* limitado utilizem o sistema com bom desempenho e usabilidade.

Este trabalho tem como objetivo apresentar o desenvolvimento e a validação do aplicativo Mão de Obra Digital como ferramenta de apoio à gestão operacional de profissionais autônomos da construção civil, com foco na organização de serviços, no controle financeiro e na formalização da atividade.

## **2. Referencial Teórico**

Esta seção apresenta conceitos que fundamentam o desenvolvimento do sistema Mão de Obra Digital. São abordados os temas de modelo de fornecimento de *software* como serviço (SaaS), arquitetura *Mobile Cloud Computing* (MCC), abordagem de desenvolvimento orientada por *Design Thinking* e a arquitetura técnica empregada, incluindo serviços web REST, Flutter, Laravel e PostgreSQL.

### **2.1 Software como Serviço (SaaS) e acessibilidade em aplicações móveis**

O mercado de desenvolvimento mobile tem crescido significativamente, impulsionado pela popularização de smartphones e tablets como ferramentas de acesso à informação. Aplicações móveis têm se tornado ferramentas fundamentais e essenciais para a vida cotidiana, facilitando o acesso a serviços, informações, comunicações e entretenimento (LEZCANO et al., 2023). Esse cenário é particularmente relevante em contextos nos quais os usuários não possuem infraestrutura computacional avançada, mas dependem do celular como principal recurso tecnológico.

Ao mesmo tempo, muitos desses dispositivos operam com *hardware* limitado (memória reduzida, processamento restrito, armazenamento escasso), o que exige soluções que não imponham alto custo computacional ao aparelho. A computação móvel, combinada ao uso da nuvem, viabiliza esse equilíbrio: parte do processamento e do armazenamento é transferida para servidores remotos, garantindo melhor desempenho, maior tolerância a falhas e acesso remoto mesmo em ambientes de infraestrutura limitada (CATUMBELA, 2019). Essas características tornam o desenvolvimento mobile uma alternativa eficiente e acessível para a criação de soluções digitais escaláveis e inclusivas.

O Mão de Obra Digital foi concebido como um sistema no modelo *Software as a Service* (SaaS). Nesse modelo, as funcionalidades do sistema são disponibilizadas ao usuário final como um serviço acessível a partir de dispositivos conectados, normalmente por meio de assinaturas periódicas ou cobranças por consumo, em vez de adquirir a propriedade do produto. Diferentemente de *software* tradicionais que exigem instalação e manutenção na infraestrutura do cliente, aplicações SaaS são hospedadas remotamente e atualizadas centralmente, permitindo que múltiplos clientes acessem a mesma base de código sem necessidade de implantações individuais.

Como ressaltam Correia, Barreto e Alves, "o Software como Serviço (SaaS) oferece um enfoque transformador para a entrega e consumo de aplicações de software" (CORREIA; BARRETO; ALVES, 2024, p. 1). Esse modelo proporciona escalabilidade, redução de custos operacionais e maior acessibilidade digital, possibilitando o uso por clientes com *hardware* limitado.

## **2.2 Arquitetura Mobile Cloud Computing (MCC)**

A arquitetura deste projeto baseia-se em *Mobile Cloud Computing* (MCC), que integra computação móvel com computação em nuvem, visando superar limitações de processamento, memória e armazenamento dos dispositivos móveis ao transferir parte das operações para servidores remotos (CATUMBELA, 2019). A adoção de MCC é particularmente relevante para o público-alvo do Mão de Obra Digital, composto por profissionais autônomos da construção civil que utilizam smartphones como principal ferramenta tecnológica.

Ao manter a maior parte da lógica de negócios e dos dados em servidores remotos, o aplicativo tende a operar de forma mais leve e estável mesmo em aparelhos de menor desempenho, possibilitando acesso aos recursos do sistema diretamente do local da obra. Além disso, a arquitetura facilita a escalabilidade do sistema, permitindo que múltiplos usuários, com dispositivos diferentes, acessem as mesmas funcionalidades sem necessidade de infraestrutura local complexa.

## **2.3 Design Thinking como abordagem de desenvolvimento**

O *Design Thinking* é uma abordagem de inovação centrada no ser humano, caracterizada por sua natureza não linear e interativa, que busca compreender profundamente os usuários e

propor soluções criativas para suas necessidades (APOCALYPSE; JORENTE, 2022). De modo geral, essa abordagem é descrita em cinco etapas principais: empatia, definição, ideação, prototipagem e teste.

A etapa de empatia consiste em identificar, conhecer e compreender o usuário e sua realidade de trabalho. No contexto deste estudo, essa etapa envolveu a observação direta das rotinas de profissionais autônomos da construção civil, evidenciando dificuldades como elaboração de orçamentos de forma manual, controle informal de horas trabalhadas pela equipe e baixa visibilidade sobre lucros e despesas de cada serviço. A etapa seguinte, de definição, organiza esses achados e os transforma em problemas claramente delimitados, como a necessidade de emitir orçamentos formais em PDF para enviar ao cliente, registrar jornadas de trabalho de colaboradores e visualizar se determinado serviço está, de fato, gerando lucro.

Com base nesses problemas priorizados, ocorre a fase de ideação, na qual são propostas e selecionadas soluções possíveis para atender às necessidades identificadas. Nessa fase foram elencadas funcionalidades consideradas essenciais para o uso cotidiano do público-alvo, como cadastro de clientes e colaboradores, registro de serviços em andamento, emissão de orçamentos e geração de relatórios financeiros. A etapa de prototipagem, por sua vez, consistiu na transformação dessas ideias em fluxos de tela e em representações funcionais do sistema, permitindo visualizar como cada ação seria executada dentro do aplicativo.

Por fim, a etapa de teste foi utilizada para validar o protótipo com potenciais usuários do setor da construção civil, com o objetivo de avaliar clareza de interface, utilidade prática no dia a dia e aderência às necessidades reais do trabalho em campo. Esse processo de validação forneceu insumos para ajustes de usabilidade e priorização de funcionalidades.

A adoção do *Design Thinking* mostrou-se adequada ao desenvolvimento do Mão de Obra Digital porque o público-alvo não é composto por usuários técnicos ou especializados em gestão digital, mas por profissionais que necessitam de uma solução direta, acessível e imediatamente aplicável. Dessa forma, a metodologia de pesquisa e desenvolvimento foi guiada a partir dessas etapas descritas.

## 2.4 Arquitetura técnica e tecnologias adotadas

A solução proposta neste trabalho foi desenvolvida com uma arquitetura voltada à disponibilidade contínua, segurança de acesso e escalabilidade, de forma a atender profissionais autônomos da construção civil que dependem do celular como principal ferramenta de trabalho. O aplicativo móvel foi implementado em Flutter, enquanto as regras de negócio e o gerenciamento de dados foram estruturados em uma API *RESTful* desenvolvido com o *framework* Laravel, em conjunto com um banco de dados PostgreSQL, protegidos por uma autenticação via *tokens*.

A camada de apresentação do sistema foi construída em Flutter, *framework* baseado na linguagem Dart voltado ao desenvolvimento de interfaces multiplataforma a partir de uma única base de código. O Flutter permite criar aplicações interativas e responsivas para Android (e, potencialmente, outras plataformas como iOS e web) com consistência visual e boa performance percebida pelo usuário final, além de acelerar o ciclo de prototipação e interação de telas (FLUTTER, 2025). Essa característica é relevante para este projeto porque o público-alvo demanda uma interface direta, de fácil leitura e baixo atrito de uso.

A camada de lógica de negócios foi implementada em Laravel, *framework* PHP amplamente utilizado para construção de aplicações web e serviços de *back-end*. Ele simplifica recursos como autenticação, definição de rotas, validação de dados e acesso estruturado ao banco de dados, permitindo organizar funcionalidades como cadastro de colaboradores, registro de serviços em andamento, controle de horas trabalhadas e geração de orçamentos e recibos (LARAVEL, 2025). Além disso, ele fornece suporte nativo para padrões *RESTful*, o que facilita a integração com clientes móveis.

Essa API em Laravel é o que garante a comunicação entre o aplicativo e o banco de dados, ela disponibiliza recursos através de *endpoints* HTTP seguido de métodos GET, POST, PUT e DELETE o que permite a troca de informações entre o cliente e o servidor. Na Integração entre Flutter e Laravel, o uso de *tokens* emitidos pelos mecanismos como Laravel *Sanctum* garante que apenas usuários autenticados com o *token*, credencial gerada pelo servidor, válido possam consumir *endpoints* sensíveis, tornando o fluxo seguro quando o *token* é armazenado de forma protegida no dispositivo, transmitido via HTTPS e validado pelo *middleware* no servidor (AL-ATRAQCHI, 2022).

O armazenamento persistente das informações do sistema, clientes, serviços realizados, colaboradores, registros de horas, despesas e valores recebidos é realizado em PostgreSQL. O PostgreSQL é um sistema gerenciador de banco de dados relacional, robusto e de código aberto, conhecido por sua confiabilidade, conformidade com SQL e capacidade de escalar com segurança para cargas de dados mais complexas (POSTGRESQL, 2025). A escolha por uma base relacional facilita a geração de relatórios financeiros e históricos de serviço, que são essenciais para a tomada de decisão do usuário sobre lucratividade e custos operacionais.

Essa combinação tecnológica (Flutter no *front-end*, Laravel como API *RESTful* com autenticação via Sanctum, e PostgreSQL como banco de dados) busca atender simultaneamente três requisitos centrais do projeto: (i) acessibilidade de uso em dispositivos móveis, inclusive de menor capacidade; (ii) segurança e controle de acesso às informações de negócio; e (iii) capacidade de escalar o sistema para múltiplos usuários sem perda de consistência dos dados.

### **3. Estado da arte**

O modelo SaaS tem se consolidado como alternativa eficaz para a gestão de negócios e, na construção civil, impulsiona plataformas digitais que conectam prestadores de serviço a clientes. Contudo, ainda são limitadas as soluções voltadas à gestão financeira, ao controle de equipes e à organização do cotidiano desses profissionais. Nesse contexto, este capítulo analisa estudos científicos sobre o uso de tecnologias na área, buscando identificar abordagens existentes, mapear lacunas e situar este projeto entre as soluções digitais atuais.

A pesquisa seguiu os princípios do Mapeamento Sistemático (MS), conforme estabelecido por Kitchenham e Charters (2007). O mapeamento sistemático tem como finalidade identificar, classificar e resumir a produção científica existente sobre um determinado tema, permitindo compreender não apenas quais soluções foram propostas, mas também quais problemas permanecem em aberto.

Para esse trabalho, foram definidas quatro Questões de Pesquisa (QP), elaboradas *strings* de busca e aplicados critérios de inclusão e exclusão. A busca foi realizada na plataforma Google Acadêmico, considerando publicações entre 2015 e 2025, nos idiomas português e inglês com o objetivo de contemplar estudos recentes e relevantes para o escopo desta pesquisa.

### 3.1 Questões de Pesquisa

A pesquisa foi guiada por quatro QP fundamentais, elaboradas a partir de uma revisão preliminar da literatura e da definição dos objetivos do estudo. Essas questões orientaram o processo de mapeamento sistemático e possibilitar uma análise crítica, sendo elas:

- **QP1:** Qual a motivação mercadológica, identificada no contexto da construção civil que impulsiona o desenvolvimento de soluções digitais voltadas à organização e gestão de profissionais autônomos?
- **QP2:** Quais benefícios associados à digitalização e automação de processos no que se refere à administração e ao controle das atividades desempenhadas por prestadores de serviço na construção civil?
- **QP3:** Quais são as principais barreiras enfrentadas por prestadores de serviço na construção civil para a adoção de tecnologias digitais em seu cotidiano profissional?
- **QP4:** Quais tecnologias são empregadas em aplicações já existentes destinadas a esse público, e como essas soluções contribuem para a melhoria da gestão de suas atividades?

### 3.2 Strings de Busca

Para a identificação de estudos relevantes à temática abordada, foram elaboradas *strings* de busca com o objetivo de filtrar os materiais de acordo com as palavras-chave previamente definidas, as quais são ilustradas no Quadro 1.

Quadro 1 - *Strings* de busca

<i>String</i>	<b>Expressão</b>
1	("App" OR "Plataform") and ("construction sector" AND "service providers") AND ("SaaS" OR "Software as a Service") AND ("challenges" OR "benefits")
2	("trabalhadores autônomos" OR "Construção Civil") AND ("Android" OR "plataforma digital") AND ("Orçamentos")

**Fonte: O autor, 2025.**

### 3.3 Análise dos Artigos

A busca por publicações científicas teve como objetivo identificar aplicações digitais voltadas à gestão de profissionais autônomos na construção civil. A primeira *string* retornou 108 artigos e a segunda 568, mas nenhum abordou modelo semelhante ao Mão de Obra Digital. Assim, foram selecionados trabalhos que, embora distintos, tratam de soluções aplicáveis ao mesmo público-alvo, além de um estudo sobre a arquitetura MCC pela sua relevância tecnológica. No total, quatro artigos foram escolhidos para análise, considerando critérios de inclusão e exclusão previamente definidos.

Para a análise e seleção dos artigos, foi estabelecido critérios de inclusão (CI) sendo:

- CI1: Artigos escritos em Português ou Inglês;
- CI2: Artigos publicados entre os anos de 2015 e 2025;
- CI3: Estudos cujo conteúdo esteja alinhado com a temática da pesquisa.

Os artigos que atenderem esses critérios, serão reavaliados sobre os critérios de exclusão (CE), trabalhos nos quais não atendam algum destes critérios são retirados do mapeamento. Sendo assim os critérios de exclusão adotados, foram:

- CE1: Artigos duplicados em diferentes plataformas ou bases de dados;
- CE2: Artigos que desviam o conteúdo para temas paralelos à pesquisa;
- CE3: Artigos que não sejam acessíveis.

### 3.4 Resultados e Comparações com Mapeamento Sistemático

O primeiro artigo selecionado para análise intitula-se “*Mobile Computing in the Construction Industry: Main Challenges and Solutions*” (SILVERIO, 2017), onde é abordado a escolha de MCC para aplicações voltadas ao setor de construção civil, pois sua arquitetura garante leveza e flexibilidade, garantindo que dispositivos de hardware limitado também tenham acesso remoto à informação. Embora não apresente uma solução SaaS como o Mão de Obra Digital, o estudo respalda a escolha da arquitetura MCC por garantir desempenho e escalabilidade superiores em relação a soluções tradicionais.

O artigo “Desenvolvimento de um software para auxílio em cálculos e orçamentos para serviços de construção civil” (TRAPP, 2017) apresenta um sistema web em PHP que gera orçamentos do custo de uma obra a partir das dimensões do projeto, estimando custos de mão de obra, prazo e materiais necessários. Embora o Mão de Obra Digital não tenha foco nesse tipo de orçamento, o estudo destaca dois pontos relevantes: a importância de interfaces simples e a padronização dos orçamentos, onde é destacado que muitos profissionais atuam de forma autônoma e informal, recorrendo a orçamentos manuscritos que comprometem a clareza e profissionalismo, aspecto diretamente relacionado à funcionalidade do Mão de Obra Digital de emitir orçamentos em PDF padronizados, reforçando a credibilidade do prestador.

O estudo “Aplicativo Android para auxiliar a confecção de orçamentos na área de construção civil” (ROCHA, 2016) descreve um aplicativo em Java para Android que calcula, a partir de medidas fornecidas, a quantidade de materiais necessários em partes específicas da obra, como lajes e paredes. A solução contribui para reduzir desperdícios e planejar melhor as etapas do serviço. Embora também voltado à construção civil e baseado em uma plataforma acessível (Android), seu foco difere do Mão de Obra Digital: enquanto o primeiro prioriza o cálculo de insumos, o segundo concentra-se na gestão do cotidiano profissional, abrangendo serviços, equipe, clientes e finanças

O artigo “Implementação de uma Plataforma Digital na Divulgação de Trabalhadores Autônomos da Região de Jundiaí” (GOUVEIA, 2024) apresenta a plataforma web JundTask, desenvolvida em PHP e JavaScript para facilitar a divulgação de serviços e permitir a avaliação de profissionais por clientes, ampliando sua visibilidade e confiabilidade no mercado local. Embora atenda ao mesmo público-alvo do Mão de Obra Digital, seu foco está na promoção dos trabalhadores, e não na gestão de atividades operacionais e administrativas. Ainda assim, o estudo reforça a relevância de soluções digitais e da tecnologia para a valorização e profissionalização dos profissionais autônomos.

De forma geral, os artigos analisados indicam que já existem iniciativas tecnológicas voltadas ao setor da construção civil, mas essas iniciativas tendem a se concentrar em dois eixos principais: (i) cálculo técnico de insumos e planejamento de materiais ou (ii) divulgação do serviço para fins de contratação. Observa-se uma lacuna no apoio direto à gestão operacional cotidiana do profissional autônomo como controle de serviços em andamento, registro de colaboradores,

emissão formal de orçamentos e acompanhamento financeiro. O Mão de Obra Digital posiciona-se exatamente nesse espaço, buscando fornecer uma ferramenta de organização de atividades e suporte administrativo para esses profissionais.

### **3.5 Respostas para as Questões**

Esta subseção apresenta as respostas às Questões de Pesquisa (QP) definidas na etapa metodológica, com base na análise dos estudos selecionados.

- QP1: Qual a motivação mercadológica, identificada no contexto da construção civil que impulsiona o desenvolvimento de soluções digitais voltadas à organização e gestão de profissionais autônomos?

A construção civil, embora em constante expansão, ainda carece de digitalização, sobretudo entre trabalhadores autônomos e informais. Silverio (2017) aponta que a arquitetura de MCC amplia eficiência e mobilidade; Trapp (2017) evidencia a persistência de orçamentos manuais e informais, que dificultam negociações; Rocha (2016) destaca a ampla adoção do Android no Brasil, tornando aplicativos mais acessíveis; e Gouveia (2024) ressalta que soluções digitais aumentam a credibilidade entre profissionais e clientes. Nesse contexto, a motivação mercadológica para ferramentas como o Mão de Obra Digital está na necessidade de facilitar a gestão de serviços, promovendo organização, profissionalismo e tecnologia acessível ao público-alvo

- QP2: Quais benefícios associados à digitalização e automação de processos no que se refere à administração e ao controle das atividades desempenhadas por prestadores de serviço na construção civil?

A digitalização e automação de processos trazem benefícios significativos para a construção civil, como a otimização das atividades e a escalabilidade, já que sistemas móveis baseados em MCC permitem acesso remoto às funcionalidades de qualquer local, sem necessidade de infraestrutura complexa. Além disso, a digitalização aumenta o profissionalismo e a confiabilidade, ao organizar orçamentos e fluxos de trabalho, fortalecendo a credibilidade junto aos clientes.

- QP3: Quais são as principais barreiras enfrentadas por prestadores de serviço na construção civil para a adoção de tecnologias digitais em seu cotidiano profissional?

Apesar de todos os benefícios trazidos pela digitalização, nos estudos de Silverio (2017) e Trapp (2017) apontam algumas barreiras que dificultam a modernização nessa área. Um dos principais é a falta de conhecimento técnico, já que muitos profissionais dessa área têm dificuldades com interfaces complexas, como sistemas ERP (*Enterprise Resource Planning*). Além disso, há uma certa resistência à mudança, pois alguns trabalhadores do setor ainda preferem métodos tradicionais como anotações com papel e caneta.

- QP4: Quais tecnologias utilizadas em aplicações já existentes voltados para esse público?

Os estudos selecionados descrevem diferentes abordagens tecnológicas aplicadas ao contexto da construção civil, incluindo aplicações móveis para cálculo de materiais, sistemas web para geração de orçamentos, plataformas de divulgação de profissionais autônomos e o uso de arquiteturas em nuvem para viabilizar acesso remoto. O Quadro 2 sintetiza essas tecnologias, destacando o foco de aplicação de cada solução identificada na literatura.

Quadro 2 - Análise das tecnologias já adotadas.

<b>Tecnologia</b>	<b>Uso</b>	<b>Referência</b>
MCC	Acesso Remoto à Informação	Silverio (2017).
PHP	Aplicação Web para orçamento de Custo de materiais, Tempo de execução e mão de obra	Trapp (2017).
Java	Aplicativo Android para cálculo de materiais	Rocha (2016).
PHP e JavaScript	Plataforma Web para Divulgação do trabalho de Prestadores de Serviço	Gouveia (2024).

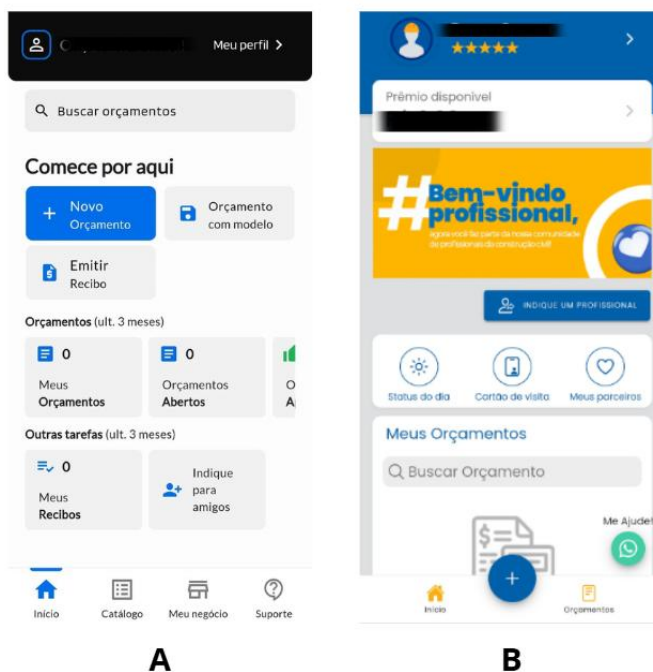
**Fonte: O autor, 2025.**

### **3.6 Análise de Mercado**

Além da revisão da literatura científica, realizou-se uma pesquisa exploratória na Play Store para identificar soluções digitais voltadas a profissionais da construção civil. A Figura 1

apresenta as telas iniciais de dois aplicativos encontrados: a Tela **A** referente ao Orçamento PRO: Orç. e Recibo, destinado à elaboração de orçamentos e geração de recibos para prestadores de serviços; e a Tela **B**, do aplicativo Obra Show para Profissionais, voltado à criação de orçamentos personalizados para profissionais da construção civil.

Figura 1 – Aplicativos encontrados na Play Store



**Fonte: Adaptado de Orçamento PRO e Dynaset Software LTDA, 2025.**

Assim como o Mão de Obra Digital, os dois aplicativos pesquisados são intuitivos e fáceis de usar, possibilitando a criação de orçamentos mais atrativos em PDF para encaminhar para os clientes. O atual projeto se destaca por, além de oferecer essa funcionalidade para ajudar profissionais a venderem seus serviços, possibilitar a gestão de sua equipe e o controle de seus serviços em andamento, além do gerenciamento das despesas operacionais.

### **3.7 Considerações do Mapeamento Sistemático**

Após a pesquisa de material científico utilizando a metodologia do mapeamento sistemático e a análise dos artigos selecionados, foi possível identificar que, com o crescimento do

setor da construção civil, algumas soluções foram desenvolvidas para prestadores de serviço. No entanto, grande parte dos materiais encontrados na literatura pesquisada focam em soluções para localizar profissionais via web, enquanto outra parcela significativa aborda estratégias para reduzir o desperdício de materiais em obras e calcular o quantitativo necessário.

O Mão de Obra Digital, projeto em estudo, é um aplicativo SaaS com foco exclusivo na administração e gestão do dia a dia do prestador de serviço. Os artigos selecionados buscam atender ao mesmo público-alvo, porém com propostas distintas. Embora não seja possível afirmar que não existam aplicações com objetivos semelhantes, pode-se concluir que nenhuma foi encontrada na literatura analisada.

#### **4. Metodologia de pesquisa**

Esta seção descreve os procedimentos metodológicos adotados nesta pesquisa, de caráter qualitativo, que buscou compreender em profundidade as necessidades do público estudado e transformar essas necessidades em requisitos funcionais para o desenvolvimento do aplicativo. A pesquisa foi conduzida com base nos princípios do *Design Thinking*, priorizando a empatia com os usuários e a validação contínua das soluções propostas. Esse processo garantiu que o sistema fosse projetado de forma alinhada às reais demandas práticas de profissionais de construção civil, especialmente microempreendedores e trabalhadores autônomos do setor.

##### **4.1 Empatia e Definição**

A etapa inicial do projeto consistiu em compreender o público-alvo por meio de observação empírica e escuta ativa em ambiente comercial do setor. Essa imersão permitiu registrar qualitativamente necessidades, dificuldades e expectativas dos profissionais. Entre os aspectos identificados destacam-se o controle de equipes, a elaboração de orçamentos mais profissionais e o acompanhamento simultâneo de diferentes serviços. Esse diagnóstico inicial serviu de base para definir os requisitos prioritários do sistema.

Com base nas informações coletadas na fase de empatia, foi elaborado um mapa mental de funcionalidades desejáveis para o aplicativo. Entre os itens mapeados estavam: cadastro de colaboradores, registro de dias e/ou horas trabalhadas, geração de orçamentos em PDF, gerenciamento de serviços ativos, relatórios financeiros do mês entre outros.

Esse artefato serviu tanto como síntese quanto como instrumento de validação das funcionalidades. O conjunto foi apresentado aos profissionais para obter *feedback* sobre utilidade, clareza e prioridade, permitindo identificar quais funções eram essenciais no cotidiano e quais poderiam ser consideradas secundárias ou desnecessárias, orientando as etapas seguintes do desenvolvimento.

## **4.2 Ideação e Fluxograma**

Após a validação inicial dos requisitos, as funcionalidades priorizadas foram traduzidas em representações visuais por meio de um fluxograma construído na plataforma Miro<sup>1</sup>. Esse fluxograma descreveu os principais fluxos de navegação e estados do sistema, permitindo visualizar como o usuário passaria por etapas como cadastrar um cliente, criar um orçamento ou registrar horas trabalhadas. O fluxograma funcionou como recurso de síntese e ideação colaborativa, permitindo validar sequências de uso, reduzir etapas desnecessárias e incorporar sugestões práticas dos profissionais envolvidos. Nesse contexto, as ideias de possíveis interfaces ficaram claras e norteadas para a etapa seguinte do projeto.

## **4.3 Prototipação**

Com os fluxos de uso definidos, foi desenvolvido um protótipo de interface do aplicativo na plataforma Figma<sup>2</sup>, transformando os requisitos e o fluxo de navegação em modelos visuais para a aprovação dos usuários. O protótipo contemplou as principais interfaces definidas no fluxograma e teve como objetivo não a criação de um artefato interativo, mas sim a validação do fluxo de navegação previamente estabelecido e da clareza da interface planejada para os profissionais.

As telas foram apresentadas em testes rápidos para analisar como o usuário interpretava as etapas de determinadas tarefas. O processo permitiu identificar dúvidas, ajustar nomenclaturas, simplificar componentes e reorganizar campos, refinando a interface até torná-la clara e funcional. Assim, obteve-se maior segurança para o desenvolvimento do código, com o protótipo servindo como planta do projeto.

---

<sup>1</sup> <https://miro.com/pt/>

<sup>2</sup> <https://www.figma.com/>

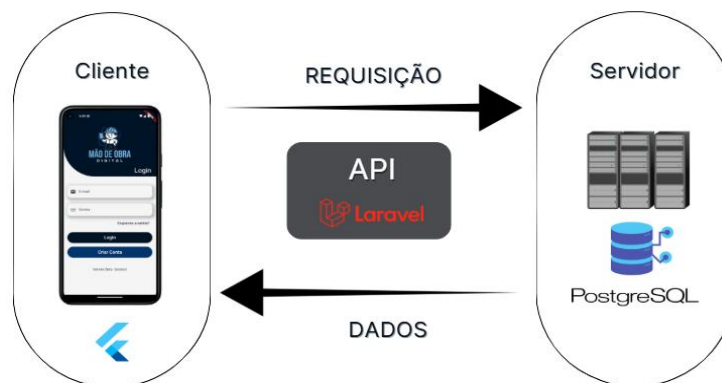
## 5. Metodologia de Desenvolvimento

Após a definição dos requisitos, validação inicial das funcionalidades e a elaboração do protótipo de interfaces, iniciou-se a etapa de desenvolvimento do sistema. Essa etapa foi orientada por boas práticas de engenharia de software, bem como a escolha da arquitetura, a definição das tecnologias que seriam utilizadas, a implementação das camadas *back-end* e *front-end* e a realização de testes de integração e qualidade, garantindo uma solução robusta e eficiente.

### 5.1 Arquitetura do Sistema

A arquitetura do projeto, conforme discutido no referencial teórico, adota o modelo de *Mobile Cloud Computing* (MCC), que integra os benefícios da computação móvel e da nuvem, permitindo que dispositivos com hardware limitado utilizem as funcionalidades sem perda de desempenho. Essa abordagem garante leveza, flexibilidade e escalabilidade, atendendo profissionais que utilizam smartphones com diferentes configurações. A Figura 2 apresenta, de forma conceitual, a comunicação entre o aplicativo móvel e os serviços em nuvem responsáveis pelo armazenamento e processamento dos dados.

Figura 2 – Ilustração de Arquitetura MCC



Fonte: O autor, 2025.

## 5.2 *Back-end* do projeto com Laravel e PostgreSQL

O desenvolvimento do *back-end* iniciou-se pela modelagem lógica do banco de dados, etapa fundamental para estruturar as tabelas e relacionamentos necessários para suprir a demanda do sistema. Em seguida, se deu início ao desenvolvimento da API em Laravel, responsável pela implementação das regras de negócio, como cadastro de colaboradores e clientes, registro de dias trabalhados, orçamentos e acompanhamento financeiro, além do controle de autenticação e acesso dos *tokens*, garantindo segurança no consumo dos *endpoints* sensíveis do servidor.

## 5.3 Desenvolvimento *Front-end* Multiplataforma com Flutter

A camada de interface foi desenvolvida em Flutter, *framework* que permite criar aplicações móveis com experiência nativa e que, futuramente, pode ser estendido para outras plataformas. A escolha por Flutter está associada à capacidade de produzir interfaces visuais consistentes, com rápida prototipação e boa responsividade em dispositivos Android, que são os mais utilizados pelo público-alvo.

A organização interna do código *front-end* foi estruturada em camadas para favorecer modularidade, clareza e manutenção. Entre as principais camadas estão:

- *Data*: responsável pela comunicação com a API e manipulação de dados;
- *Models*: definição das estruturas de dados utilizadas no sistema;
- *Widgets*: componentes reutilizáveis da interface;
- *Screens*: telas principais onde compõem a navegação do aplicativo;

Essa separação permitiu não apenas o desenvolvimento mais organizado, mas também facilitou ajustes visuais e de usabilidade conforme os retornos dos testes com usuários.

## 5.4 Integração e testes de qualidade

Após o desenvolvimento das camadas do *back-end* e do *front-end* da aplicação, foi realizada a integração entre as tecnologias a partir do pacote HTTP do Flutter responsável por enviar requisições à API e receber as respostas do servidor Laravel. A partir desse ponto, foram executados testes de qualidade fundamentais descritos abaixo:

- Testes unitários com widgets: utilizados para verificar o funcionamento de forma isolada os componentes da interface.
- Testes de integração manuais: realizados sem ferramentas automatizadas, consistiram em simulações de uso do aplicativo em ambiente local.
- Validação de requisições: realizado através do Postman e o próprio Flutter, com objetivo de testar as chamadas à API, garantindo correta comunicação entre cliente e servidor.
- Testes de Aceitação: consistiram em após o desenvolvimento do protótipo, disponibilizar para cinco possíveis clientes utilizarem. Essa fase corresponde à fase de Testes do *Design Thinking*, permitindo validar a solução junto ao público-alvo em condições reais de uso e coletar *feedbacks* para ajustes finais.

Esses testes foram fundamentais para identificar inconsistências, ajustar fluxos e validar a usabilidade do sistema antes da etapa de resultados. Os resultados dos testes de aceitação foram coletados por meio de formulários do Google, onde serão apresentados na seção de Resultados.

## **6. Resultados**

Esta seção apresenta os principais resultados obtidos a partir das etapas de pesquisa e desenvolvimento adotadas, bem como os materiais empregados no processo. Além da apresentação visual dos resultados por meio de figuras, busca-se analisar sua relevância e discutir suas implicações, reforçando a importância dos temas abordados nas seções anteriores. As figuras e representações completas encontram-se em apêndices, para consulta.

### **6.1 Mapa Mental**

O mapa mental elaborado na fase inicial do projeto atuou como instrumento de levantamento e validação preliminar de requisitos junto aos potenciais usuários. A primeira parte do mapa mental, apresentada no Apêndice A, reúne as funcionalidades consideradas essenciais para o funcionamento inicial do sistema, ou seja, aquelas percebidas como indispensáveis no dia a dia do profissional autônomo. Já a segunda parte do mapa, apresentada no Apêndice B, reúne funcionalidades complementares, identificadas como desejáveis, mas não obrigatórias na primeira

versão do aplicativo.

## **6.2 Fluxograma do Sistema**

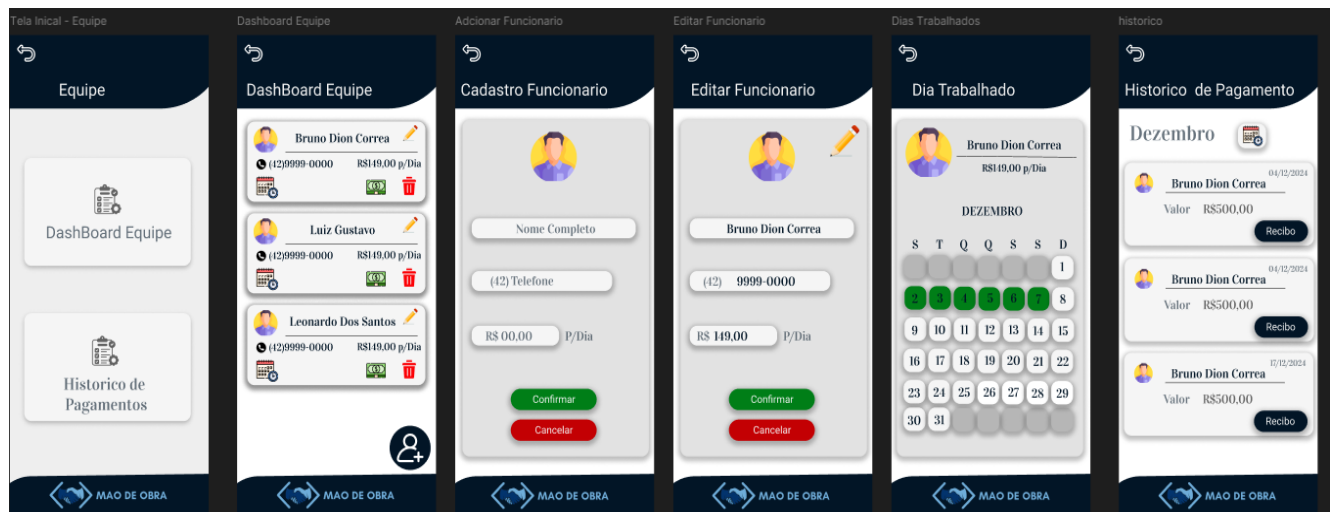
Com base nas funcionalidades priorizadas no mapa mental e validadas com os usuários, foi elaborado um fluxograma de navegação do sistema. Esse fluxograma, descrito na Seção 4 e apresentado integralmente no Apêndice C (com detalhamentos adicionais nos Apêndices D e E), representa o percurso esperado do usuário dentro do aplicativo, desde ações iniciais como cadastrar um cliente ou colaborador até tarefas mais avançadas, como gerar um orçamento em PDF ou consultar informações financeiras.

O fluxograma cumpriu dois papéis principais. Primeiro, permitiu transformar requisitos em rotas de uso concretas, evitando inconsistências entre telas e ações. Segundo, possibilitou eliminar etapas redundantes e ajustar a ordem das interações antes mesmo da construção do protótipo. Esse refinamento é consistente com a importância de interfaces claras e processos diretos para o público-alvo, já apontada por Trapp (2017), ao destacar que muitos profissionais desse setor tendem a rejeitar ferramentas complexas.

## **6.3 Protótipo de Interfaces no Figma**

Com o fluxo de navegação e funcionalidades definidos, foi desenvolvido um protótipo de interface do aplicativo utilizando a plataforma Figma. Esse protótipo ajudou a validar junto aos usuários se sua navegação proposta era clara e compatível com a forma como eles executam suas atividades no dia a dia e se as telas estavam com linguagem adequada e organização intuitiva dos campos. A Figura 8 ilustra um exemplo do protótipo da tela de gerenciamento de equipe, onde cada colaborador pode ser cadastrado e associado ao serviço em andamento.

Figura 8 – Ilustração do protótipo da Tela de Equipe no Figma



Fonte: O autor, 2025.

Esse processo permitiu identificar termos pouco claros, reorganizar campos e simplificar etapas. Como resultado, o protótipo em Figma funcionou como base visual para o desenvolvimento do *front-end* em Flutter e reduziu retrabalho na fase de implementação.

#### 6.4 Implementação técnica e segurança

A subseção 6.4 descreve os principais aspectos de implementação técnica do Mão de Obra Digital. O objetivo é demonstrar que o sistema foi desenvolvido não apenas como um protótipo conceitual, mas como uma aplicação funcional, escalável e segura, adequada ao contexto real de uso por profissionais autônomos da construção civil.

##### 6.4.1 Modelagem do banco de dados

A modelagem lógica do banco de dados orientou a definição de entidades e relacionamentos do sistema, guiando a estruturação de forma que se pudesse garantir integridade referencial e escalabilidade. O diagrama lógico do banco de dados do sistema está representado no Apêndice F, onde é possível visualizá-lo por completo e, devido ao seu tamanho, foi dividido em duas ilustrações para melhor visualização onde são citadas no Apêndice G e Apêndice H.

A modelagem orientou o desenvolvimento *back-end* do projeto, evitando manutenções desnecessárias. Nesse contexto de arquitetura em nuvem discutido anteriormente, para garantir que o sistema permaneça acessível e funcional mesmo diante de falhas ou picos de acesso e que o mesmo possa crescer e atender um número maior de usuários sem perda de desempenho, foi utilizado o serviço de hospedagem da Neon (NEON, 2025).

#### 6.4.2 Implementação em Laravel e proteção das rotas

Com a modelagem de banco de dados definida e o sistema gerenciador estabelecidos, foi desenvolvida a API responsável pelas funcionalidades do sistema. A API expõe endpoints que permitem operações como cadastrar e editar dados, listar registros, gerar orçamentos e excluir entradas quando necessário.

A Figura 12 apresenta um trecho do código responsável pela definição das rotas no Laravel, todas protegidas pelo *middleware auth:sanctum*. O *middleware* atua como uma camada intermediária que intercepta as requisições e aplica regras de segurança ou validação antes de permitir o acesso, enquanto o Laravel *Sanctum* é o mecanismo de autenticação baseado em *tokens* que garante que apenas usuários autenticados tenham acesso aos recursos do sistema. As rotas foram organizadas em grupos funcionais, como exemplo a rota de funcionários, onde há funções de adição, edição, visualização e exclusão, além de rotas extras adicionadas conforme a necessidade.

Figura 12– Trecho do código de definição das rotas da API

```
Route::middleware('auth:sanctum')->group(function (){
    //funcionarios
    Route::post('criarFuncionario',[funcionarioController::class,'criarFuncionario']);
    Route::get('buscarFuncionarios',[funcionarioController::class,'buscarFuncionarios']);
    Route::get('buscarFuncionariosArquivados',[funcionarioController::class,'buscarFuncionariosArquivados']);
    Route::post('deletarFuncionario',[funcionarioController::class,'deletarFuncionario']);
    Route::post('restaurarFuncionario',[funcionarioController::class,'restaurarFuncionario']);
    Route::post('updateFuncionario',[funcionarioController::class,'updateFuncionario']);
});
```

**Fonte: O autor, 2025.**

Essa estrutura modularizada favorece a manutenção e a escalabilidade da aplicação, ao

mesmo tempo em que assegura a integridade dos dados. O uso do Laravel *Sanctum* reforça a segurança, conforme discutido por Al-Atraqchi (2022), permitindo o consumo de endpoints sensíveis apenas por usuários autenticados. Para garantir a performance, a API foi hospedada em um serviço de VPS (Virtual Private Server) da Hostinger (2025), cuja infraestrutura em nuvem atende adequadamente às necessidades do sistema.

### 6.4.3 Integração Flutter-Laravel

Para assegurar a integração com o back-end, foram implementadas rotas de comunicação HTTP, permitindo que o aplicativo consumisse os serviços disponibilizados pela API em Laravel. A Figura 13 apresenta um exemplo desse processo por meio de um trecho de código em Flutter, no qual o aplicativo realiza uma requisição ao *endpoint* `/criarFuncionario`, utilizando o token de autenticação previamente obtido.

Figura 13– Trecho do código de comunicação em Flutter

```
Future criarFuncionario(Funcionario funcionario) async {
  final token = Global.usuario!.token;
  try {
    final response =
      await http.post(Uri.parse(Global.baseUrl + '/criarFuncionario'),
        headers: {
          'Content-Type': 'application/json',
          'Accept': 'application/json',
          'Authorization': "Bearer $token",
        },
        body: jsonEncode({
          'nome': funcionario.nome,
          'salario': funcionario.pagamento,
          'forma_trabalho': funcionario.formaTrabalho.toLowerCase(),
          'numero': funcionario.contato.replaceAll(RegExp(r'[^\\d,]'), ''),
        }));
    if (response.statusCode == 200) {
      return jsonDecode(response.body);
    }
  } catch (e) {
    print('erro -> $e');
    return e;
  }
}
```

**Fonte: O autor, 2025.**

Esse mecanismo de comunicação evidencia a integração entre Flutter e Laravel, assegurando consistência e segurança no tráfego de dados. Nesse contexto, reforça a proposta de oferecer uma solução SaaS acessível e confiável, alinhada às boas práticas de autenticação em

APIs *RESTful* (AL-ATRAQCHI, 2022).

#### 6.4.4 Autenticação em Duas Etapas (2FA)

Para reforçar a segurança do sistema, implementou-se a autenticação em duas etapas (2FA), que exige, além da senha, um código de verificação enviado por e-mail. O recurso é aplicado tanto no login quanto na recuperação de credenciais, garantindo que redefinições de senha ocorram apenas mediante autenticação. O envio dos códigos de verificação foi implementado utilizando a *facade Mail*, recurso nativo do Laravel que fornece uma interface simplificada para serviços internos do *framework*. Nesse caso, a *facade* atua como um atalho para o componente de envio de e-mails, que utiliza o protocolo SMTP (*Simple Mail Transfer Protocol*), responsável por transmitir mensagens entre servidores de e-mail de forma segura.

Essa abordagem permitiu disparar os códigos de autenticação de maneira integrada à aplicação, transmitindo maior confiabilidade aos profissionais que utilizam o aplicativo e assegurando que, mesmo em caso de comprometimento da senha, o acesso só seja liberado com a validação do segundo fator. A Figura 14 ilustra a tela de autenticação quando feito login no aplicativo pela primeira vez.

Figura 14 – Tela de autenticação de duas etapas



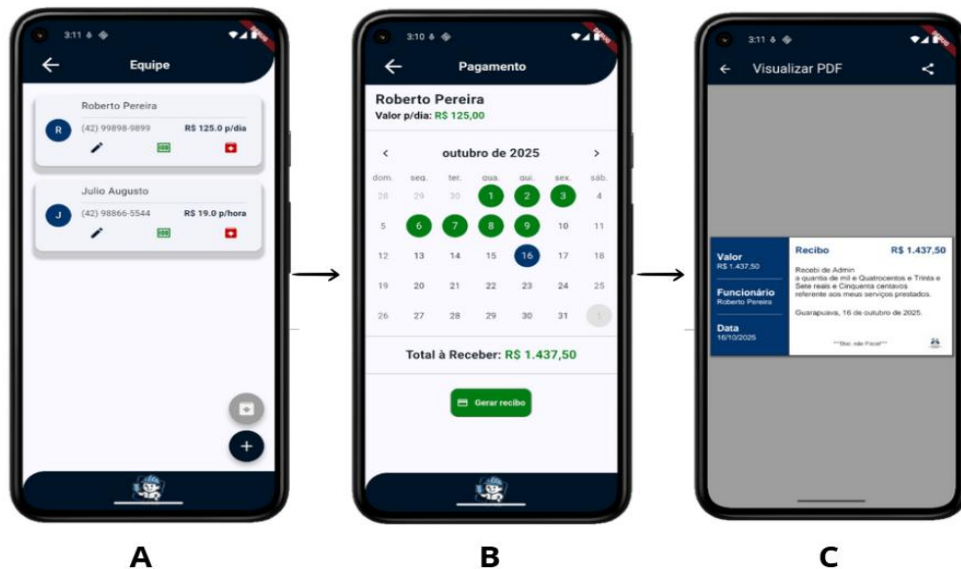
A imagem mostra uma interface de autenticação de duas etapas. No topo, há um ícone de envelope azul com folhas verdes decorativas. Abaixo, o texto indica: "Digite o código enviado para [e-mail oculto].com". Segue a informação "Código expira em 01:29". Abaixo disso, há seis caixas de entrada vazias para o código. Um botão verde com o texto "Validar e prosseguir" e um ícone de seta para a direita está posicionado abaixo das caixas. Na base da tela, há o link "Ainda não chegou? Reenviar código".

Fonte: O autor, 2025.

## 6.5 Mão de Obra Digital

Esta subseção apresenta as principais telas do sistema funcional, destacando como cada funcionalidade atende às necessidades identificadas. A Figura 15 demonstra o controle de gestão de equipe desenvolvido. Na **Tela A**, é exibida uma lista com os funcionários cadastrados no sistema com suas informações. A **Tela B** mostra a interface onde o usuário cadastra os dias trabalhados através de um calendário dinâmico. Por fim, a **Tela C** apresenta a visualização do recibo em PDF, onde contém dados como valor, nome do funcionário e período de referência.

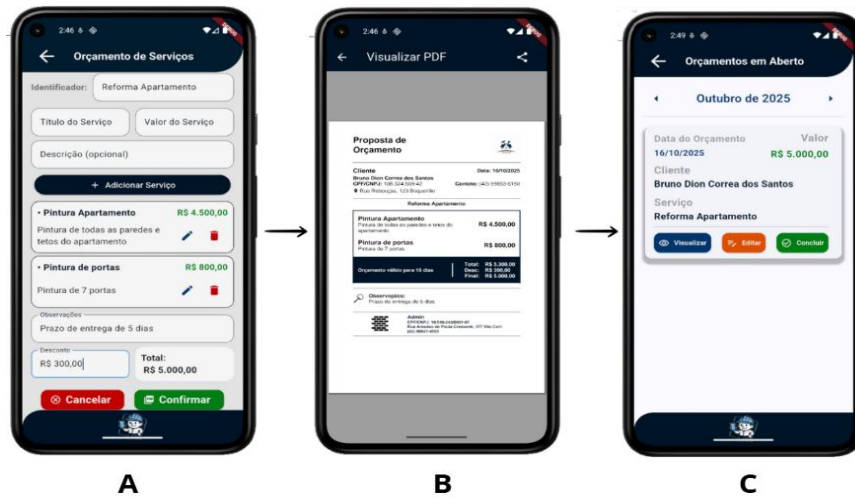
Figura 15 – Telas das Funcionalidades de equipe



Fonte: O autor, 2025.

A Figura 16 apresenta a tela de orçamento de serviços do aplicativo. Na **Tela A**, o usuário cria um orçamento, podendo adicionar múltiplos serviços, observações e descontos, com cálculo automático no total. A **Tela B** mostra a proposta em PDF, contendo os dados do cliente, do prestador e discriminação dos serviços incluídos. Já a **Tela C**, exibe a listagem de orçamentos em aberto, com opções de visualizar, editar ou concluir, transformando o em serviço em andamento. Essa funcionalidade responde diretamente a dificuldade apontada por Trapp (2017), que destacou a informalidade na elaboração de orçamentos como barreira para negociações mais profissionais.

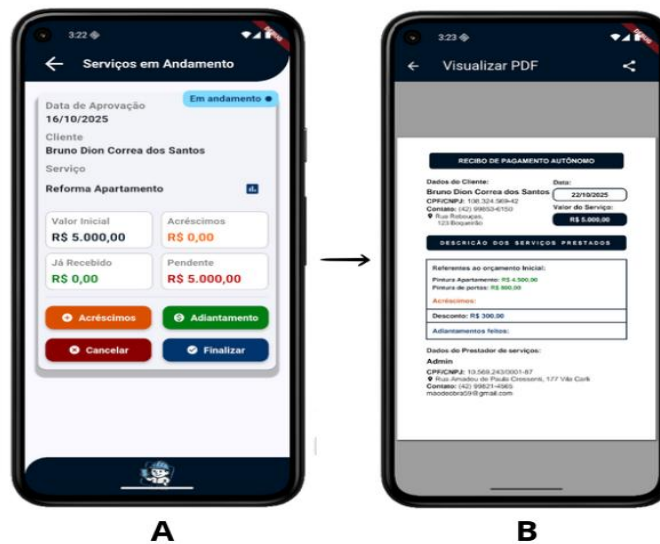
Figura 16 – Telas das Funcionalidades de Orçamento



Fonte: O autor, 2025.

A Figura 17 demonstra a dashboard de serviços em andamento do aplicativo onde, na Tela A, é exibido uma *dashboard* em formato de *cards*, que reúne informações essenciais de cada serviço já efetivado, facilitando a gestão dos mesmos. Já a Tela B mostra o recibo gerado para o cliente. Essa funcionalidade reforça a transparência e o profissionalismo no relacionamento entre o usuário e seus clientes.

Figura 17 – Telas das Funcionalidades de Serviços em Andamento



Fonte: O autor, 2025.

## 6.6 Aprovação do público alvo

Após a disponibilização do protótipo funcional para um grupo de cinco potenciais usuários do setor da construção civil, o sistema foi utilizado em ambiente real de trabalho durante um período de dois meses. Ao término desse período, aplicou-se um formulário elaborado no Google Forms, teve intuito de pesquisar e analisar como foi a experiência dos usuários com o aplicativo.

As respostas coletadas, demonstram uma avaliação positiva e grande satisfação com o aplicativo. Um dos participantes destacou que o sistema "*facilitou a forma de administrar minha rotina de trabalho*", evidenciando que as funcionalidades estão alinhadas às necessidades operacionais do público-alvo. Outro usuário afirmou que "*o aplicativo reduziu em pelo menos 50% o tempo que preciso para fazer um orçamento*", destacando ganhos de agilidade e padronização na etapa de negociação com clientes. Também foi mencionado que o aplicativo "*passa mais profissionalismo para o cliente*", indicando impacto direto na formalização da prestação de serviço e na imagem comercial do trabalhador.

Esses resultados validam tanto a viabilidade técnica quanto a pertinência mercadológica do Mão de Obra Digital. A avaliação positiva por parte de usuários reais, em contexto de uso real, consolida o sistema como uma solução prática e aplicável ao cotidiano de profissionais autônomos da construção civil.

## 7. Considerações Finais

O presente trabalho teve como objetivo principal desenvolver um aplicativo para auxiliar profissionais autônomos e microempreendedores da construção civil na gestão de suas atividades operacionais e administrativas. Para alcançar esse objetivo, foram aplicadas metodologias de pesquisa qualitativa e de desenvolvimento de *software*, com base nos princípios de *Design Thinking*, aliada a um processo de desenvolvimento orientado por boas práticas de engenharia de software.

Os objetivos definidos no início da pesquisa foram alcançados progressivamente. A etapa de empatia permitiu compreender necessidades reais do público-alvo, como a elaboração de orçamentos, controle de equipe e acompanhamento financeiro de serviços em andamento. A etapa de definição e priorização de requisitos transformou essas demandas em funcionalidades, enquanto

a prototipação e os testes garantiram soluções compreensíveis e relevantes para os usuários. A implementação técnica resultou em uma aplicação funcional baseada em arquitetura *Mobile Cloud Computing* (MCC), com *back-end* em Laravel, banco de dados PostgreSQL e *front-end* em Flutter, adequada ao perfil tecnológico do público por ser leve, escalável e acessível em sistemas operacionais Android.

Este trabalho apresenta três contribuições principais: (i) prática, ao oferecer uma solução móvel que apoia a gestão administrativa de profissionais autônomos; (ii) metodológica, ao demonstrar a eficácia do Design Thinking para levantar requisitos junto a usuários pouco familiarizados com sistemas complexos; e (iii) tecnológica, ao integrar Flutter, Laravel com *Sanctum*, PostgreSQL e autenticação em duas etapas (2FA), comprovando a viabilidade de uma solução SaaS segura, validada com usuários reais e preparada para escalar.

A avaliação de uso realizada com cinco potenciais usuários durante dois meses reforçou a pertinência do sistema. Os participantes relataram melhora na organização do trabalho diário, redução do tempo necessário para gerar orçamentos e aumento da percepção de profissionalismo perante os clientes. Esses resultados indicam que o Mão de Obra Digital atende a demandas concretas do setor e possui potencial de aplicação mercadológica.

Embora uma versão funcional já tenha sido validada, o sistema ainda prevê evoluções antes de um lançamento comercial. Entre elas, destacam-se: (i) inclusão de funcionalidade específica para registro e acompanhamento de empreiteiros; (ii) apoio à redação de orçamentos com auxílio de técnicas de Inteligência Artificial para melhoria gramatical e clareza do documento entregue ao cliente; (iii) emissão de listas de materiais em PDF; (iv) ampliação de recursos de acessibilidade, como opções de ajuste de fonte; (v) disponibilização de diferentes modelos de layout para orçamentos e recibos; e (vi) integração de provedores de pagamento para viabilizar controle de assinaturas e gestão do acesso como serviço.

Mais do que o desenvolvimento de um aplicativo, este trabalho representa uma jornada de aproximação entre teoria e prática em Engenharia de Software. Demonstra-se que a combinação entre metodologias centradas no usuário, arquitetura tecnológica adequada ao contexto de uso e validação contínua em campo pode gerar soluções com impacto direto em públicos que tradicionalmente não são atendidos por ferramentas digitais de gestão.

O desenvolvimento do Mão de Obra Digital demonstra cumprimento do objetivo proposto

e abre caminho para novas iterações técnicas e para aprofundamentos científicos. Além de responder a uma demanda imediata do setor da construção civil, o projeto reforça o papel da Engenharia de Software como instrumento de transformação social e de formalização de atividades produtivas em contextos informais.

## Referências

AL-ATRAQCHI, Osama M. A. A proposed model for building a secure restful API to connect between server-side and mobile applications using Laravel framework with Flutter toolkits. *Cihan University-Erbil Scientific Journal*, v. 6, n. 2, p. 28–35, 20 ago. 2022. DOI: 10.24086/cuesj.v6n2y2022.pp28-35.

APOCALYPSE, Simão Marcos; JORENTE, Maria José Vicentini. O método Design Thinking e a pesquisa em Ciência da Informação. *Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação*, v. 27, p. 1-21, 2022.

CATUMBELA, Euclides; PRATA, Paula. Estratégias de tolerância a falhas em computação móvel na nuvem. In: *Conferência de Ciências da Computação Tendências e Paradigmas (3CTP'2019)*.

CORREIA, Geovanni dos Santos; BARRETO, Gabriel Santos de Lima; ALVES, Nathalia de Meneses. Crescimento e expansão no uso de software como serviço (SaaS): estratégias e obstáculos para empresas de tecnologia. *Revista JRG de Estudos Acadêmicos*, v. 7, n. 13, e14902, 12 jan. 2024. DOI: 10.55892/jrg.v7i13.902.

CRESCIMENTO da construção civil. Disponível em: <https://cbic.org.br/construcao-civil-cresce-43-em-2024-e-impulsiona-economia-nacional/>. Acesso em: 2 jun. 2025.

CRESCIMENTO do MEI no setor. Disponível em: <https://agenciasebrae.com.br/economia-e-politica/microempresas-foram-segundo-que-mais-cresceu-entre-pequenos-negocios-abertos-em-2024/>. Acesso em: 2 jun. 2025.

DA SILVA, Rafael Trapp; CANTARELLI, Gustavo Stangherlin. Desenvolvimento de um software para auxílio em cálculos e orçamentos para serviços de construção civil.

FIGMA. Documentação oficial do Figma. 2025. Disponível em: <https://help.figma.com/hc/en-us>. Acesso em: 7 set. 2025.

FLUTTER. Documentação oficial do Flutter. 2025. Disponível em: <https://docs.flutter.dev/>. Acesso em: 7 set. 2025.

GOUVEIA, Giovana Neris Moraes; COSTA, Mateus Andrade da. Implementação de uma plataforma digital na divulgação de trabalhadores autônomos da região de Jundiá.

HOSTINGER. Documentação oficial da Hostinger. 2025. Disponível em: <https://www.hostinger.com/vps-hosting>. Acesso em: 20 set. 2025.

KITCHENHAM, Barbara; CHARTERS, Stuart. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. Technical Report EBSE 2007-001. Keele University; Durham University Joint Report, 2007.

LARAVEL. Documentação oficial do Laravel. 2025. Disponível em: <https://laravel.com/docs/12.x/installation>. Acesso em: 7 set. 2025.

LEZCANO, Ilsen Luján Miranda; ROMERO, Jonathan Daniel Lopez; GONZALEZ, Lucas Emanuel Cañete; ZACARÍAS, Fernando Gabriel; DOMINGUEZ, Myrian Graciela Ortega; CENTURIÓN, Cristhian Zarate. Mobile applications and their importance in the commercial world. *Revista Gênero e Interdisciplinaridade*, v. 4, n. 5, p. 797-811, 2023.

NEON. Documentação oficial do Neon. 2025. Disponível em: <https://neon.com/docs/introduction>. Acesso em: 20 set. 2025.

ORÇAMENTO PRO: Orç. e Recibo. Orçamento PRO: Orçamentos Profissionais. Disponível em: <https://play.google.com/store/apps/details?id=com.orcamento.pro>. Acesso em: 10 out. 2025.

OBRA SHOW para Profissionais. Dynaset Software LTDA. Disponível em: <https://play.google.com/store/apps/details?id=profissional.obrashow.dynaset>. Acesso em: 10 out. 2025.

PANORAMA da construção civil em 2025. Disponível em: [https://www.palfinger.com/pt-br/blog/qual-o-panorama-da-construcao-civil-no-brasil-em-2025\\_n\\_913064](https://www.palfinger.com/pt-br/blog/qual-o-panorama-da-construcao-civil-no-brasil-em-2025_n_913064). Acesso em: 2 jun. 2025.

POSTGRESQL GLOBAL DEVELOPMENT GROUP. Documentação oficial do PostgreSQL. 2025. Disponível em: <https://www.postgresql.org/docs/>. Acesso em: 7 set. 2025.

ROCHA, Renan da Silva. Aplicativo Android para auxiliar a confecção de orçamentos na área da construção civil. 2016.

SILVERIO, M. et al. Mobile computing in the construction industry: main challenges and solutions. In: *International Conference on Leadership, Innovation and Entrepreneurship (ICLIE)*, 2016. *Leadership, Innovation and Entrepreneurship as Driving Forces of the Global Economy: Proceedings*. Springer International Publishing, 2017.

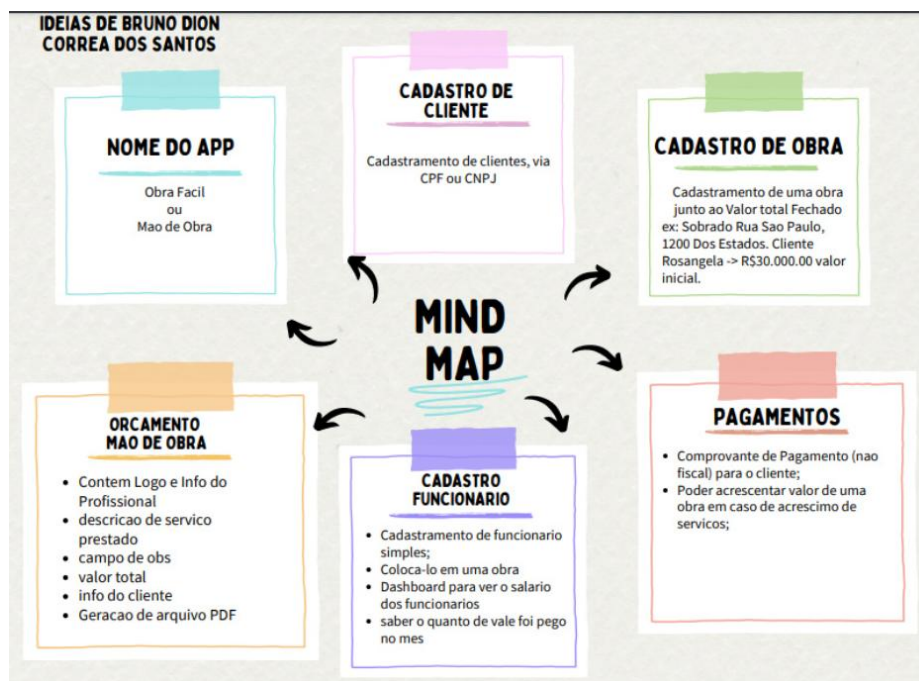
SMARTPHONES no Brasil. Disponível em: <https://portal.fgv.br/noticias/uso-ti-brasil-pais-tem-mais-dois-dispositivos-digitais-habitante-revela-pesquisa>. Acesso em: 2 jun. 2025.

## Apêndice

### Apêndice A - primeira parte do Mapa Mental

A Figura A.3 ilustra a primeira parte do mapa mental conforme mencionado na Seção 6.1. Essa primeira parte representa as ideias de funcionalidades principais que foram pensadas para o sistema, nas quais foram validadas junto ao público alvo para definir o que era realmente prioritário. Com a Ilustração é possível verificar o que cada funcionalidade traria de benefício para o usuário, como por exemplo a funcionalidade de orçamento de mão de obra, onde o documento seria em PDF com logo e informações do profissional junto com descrição dos serviços, campo de observação, valor total e informações do cliente.

Figura A.3 – primeira parte do Mapa Mental



Fonte: O autor, 2025.

Essa validação junto aos usuários permitiu lapidar algumas funcionalidades como a parte dos funcionários, onde não seria necessário especificar em qual obra ele está trabalhando, evitando possíveis manutenções no sistema.

## Apêndice B - Segunda parte do Mapa Mental

A Figura B.4 ilustra a segunda parte do mapa mental, onde é possível ver as funcionalidades extras que haveria no sistema. Essa parte do mapa mental foi importante pois eliminou três funcionalidades que foram consideradas como não necessárias pelo usuário, sendo elas Agenda, Geolocalização de Obras e Integração com Métodos de Pagamento.

Figura B.4 – Segunda parte do Mapa Mental



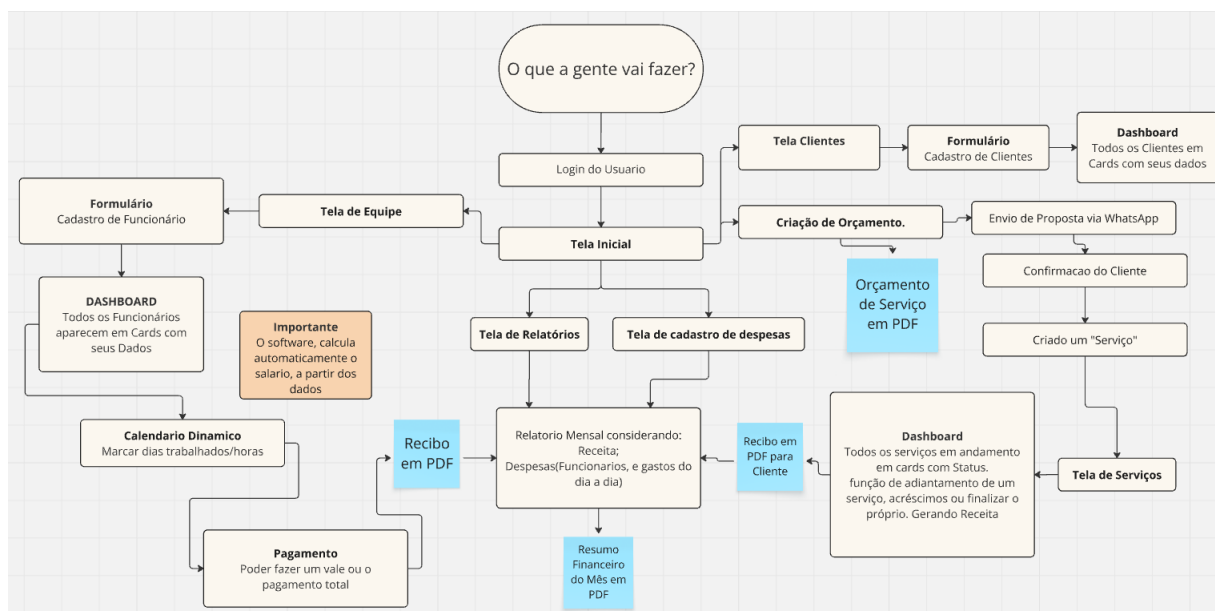
Fonte: O autor, 2025.

Relatórios financeiros foram aprovados e também sugerido a incrementação da funcionalidade de cadastro de despesas. Histórico de Obras sofreu algumas mudanças para entrar no sistema e orçamentos de materiais será uma futura implementação do projeto.

## Apêndice C - Ilustração Completa do Fluxograma

A Figura C.5 traz a ilustração completa do Fluxograma o qual foi mencionado anteriormente na Seção 6.1.2. Devido ao seu tamanho foi dividido em outras duas ilustrações para uma visualização mais detalhada. Nele é possível ver o fluxo de navegação de cada funcionalidade do sistema, e como se conectam umas às outras.

Figura C.5 – Ilustração Completa do Fluxograma

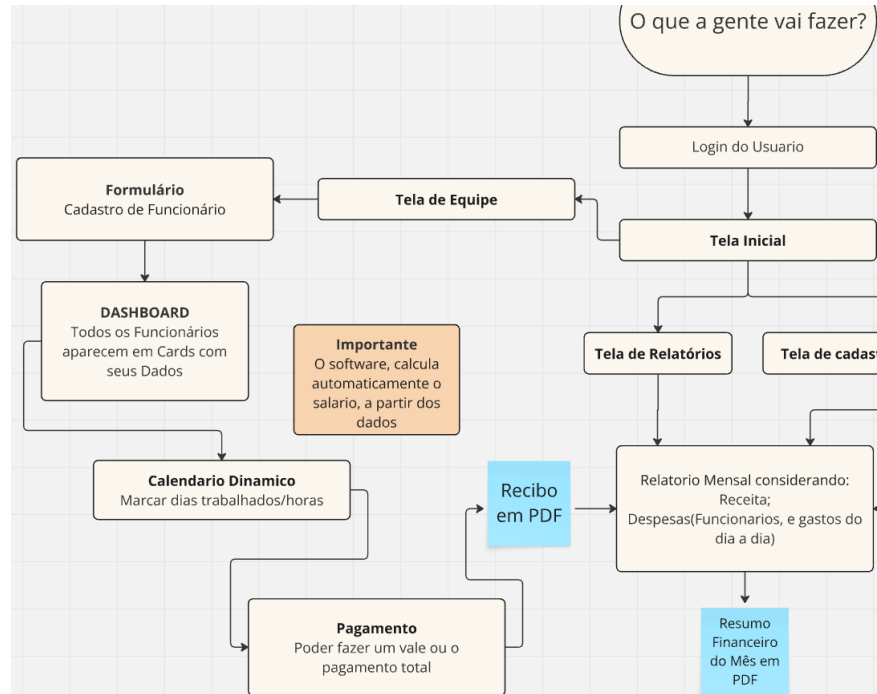


Fonte: O autor, 2025.

## Apêndice D - Primeira metade do Fluxograma

A Figura D.6 ilustra a primeira metade do fluxograma, onde é possível observar como o fluxo da Tela de Equipe iria se comportar no sistema. Essa ilustração permitiu alinhar como por exemplo a geração de um recibo para o funcionário iria afetar diretamente a Tela de relatórios, garantindo que o relatório financeiro do mês seja preciso e eficiente.

Figura D.6 – Primeira metade do Fluxograma

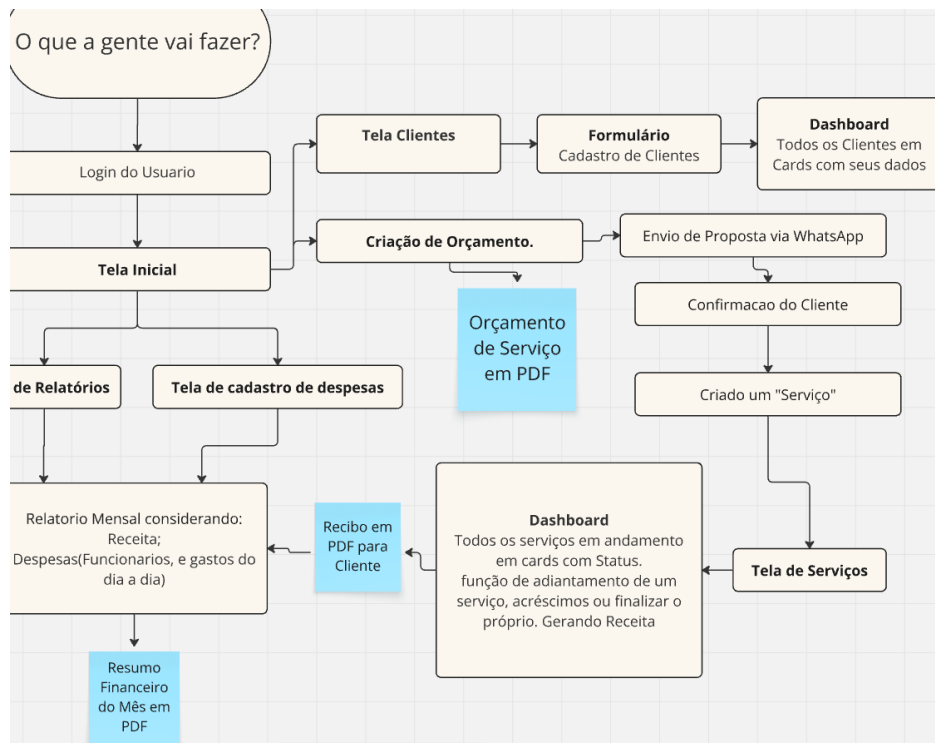


Fonte: O autor, 2025.

## Apêndice E - Segunda metade do Fluxograma

A Figura E.7 já traz detalhadamente um dos fluxos fundamentais do sistema que seria a criação de orçamentos e que serviços em andamentos dependem da geração de um orçamento e confirmação do mesmo para ser criado, além também da Tela de Clientes que é fundamental para geração de orçamentos. Nesse contexto as Telas de Serviços e despesas como é ilustrado abaixo, estão diretamente ligadas a Tela de Relatórios do sistema, assim como a Tela de Funcionários como foi mencionado anteriormente no Apêndice D.

Figura E.7 – Segunda metade do Fluxograma

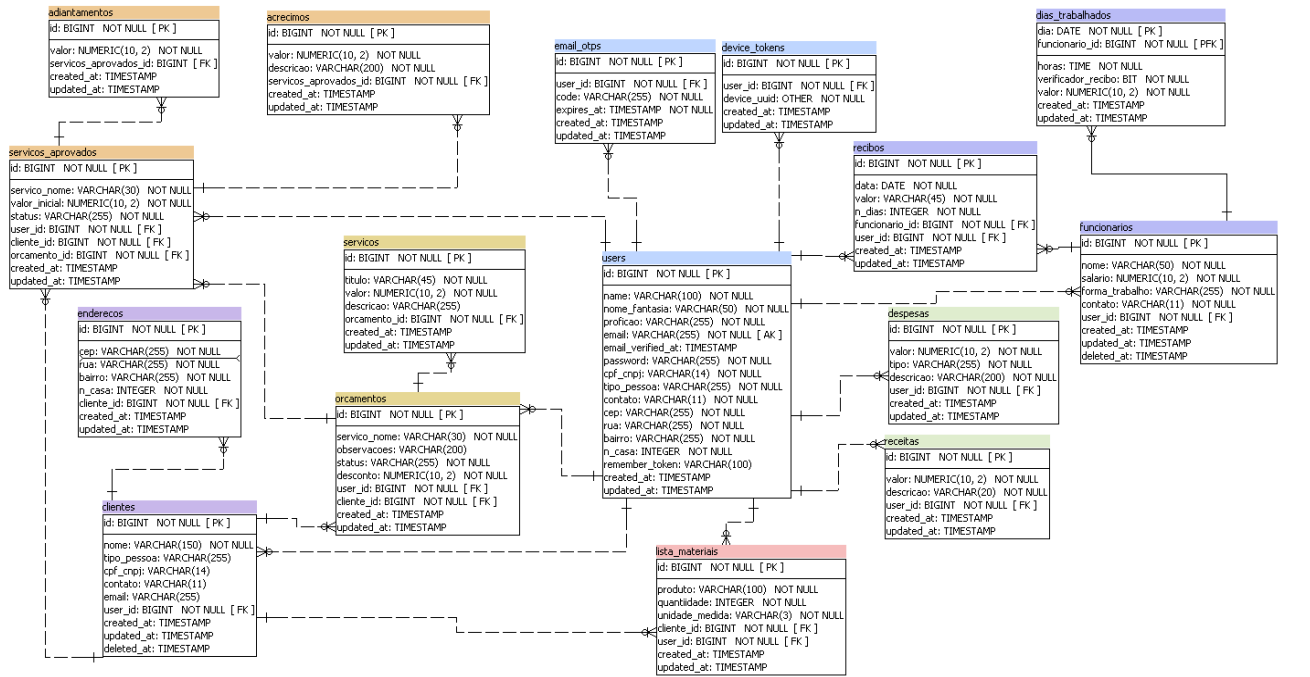


Fonte: O autor, 2025.

## Apêndice F - Ilustração completa do Modelo Lógico do Banco de Dados

A Figura F.9 traz a ilustração completa do modelo lógico do Banco de Dados do sistema, conforme mencionado anteriormente na Seção 6.1.3. Para atender a quantidade de funcionalidades e garantir integridade dos dados, o banco de dados acabou contendo em sua versão final um total de 16 tabelas para suprir as necessidades do sistema.

Figura F.9 – Ilustração completa do Modelo Lógico de Banco de Dados

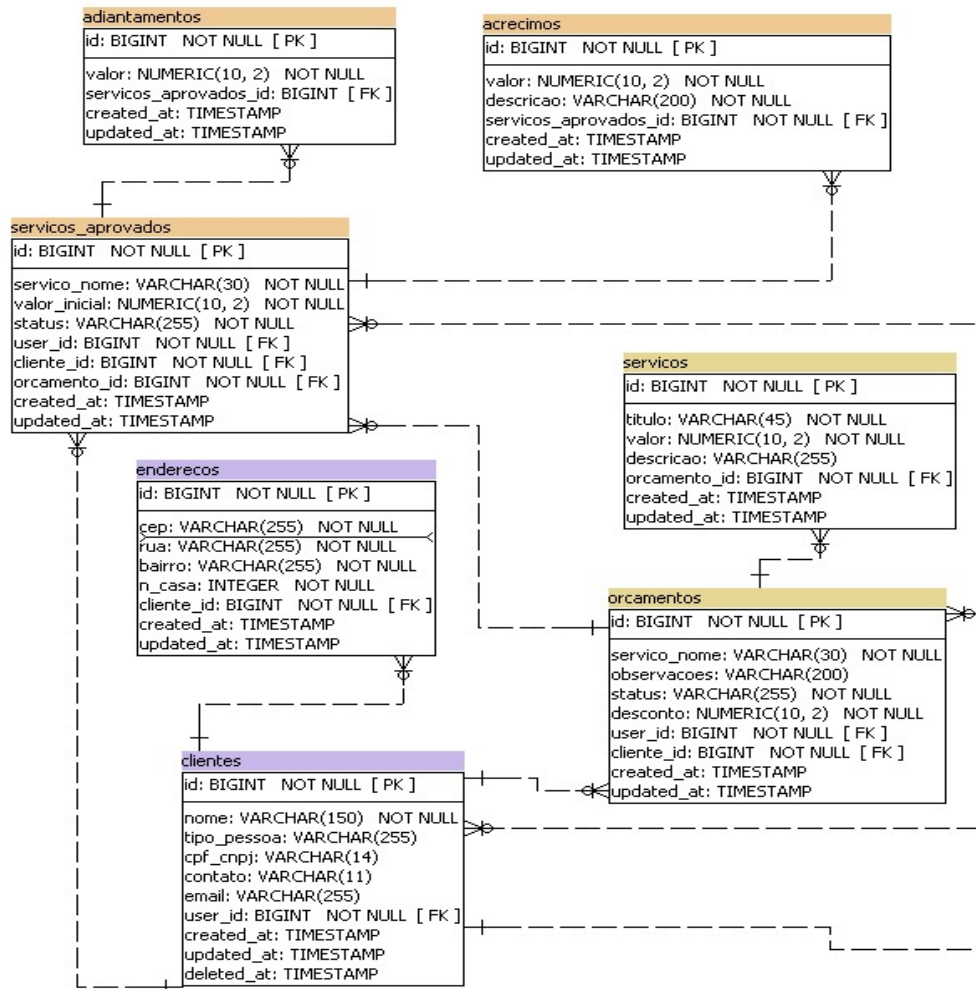


Fonte: O autor, 2025.

## Apêndice G - Primeira metade do Modelo Lógico

A Figura G.10 apresenta a primeira metade do modelo lógico do banco de dados, onde é demonstrado sete tabelas essenciais e suas respectivas relações. Com essa ilustração é possível identificar como certas funcionalidades têm tabelas próprias no banco, como acréscimos e adiantamentos, assim evitando tabelas muito grandes e facilitando a manutenção. Para evitar tabelas extensas, também foi criada uma tabela de endereços para relacionar com clientes, e uma tabela serviços para relacionar com orçamentos.

Figura G.10 – Primeira metade do Modelo Lógico

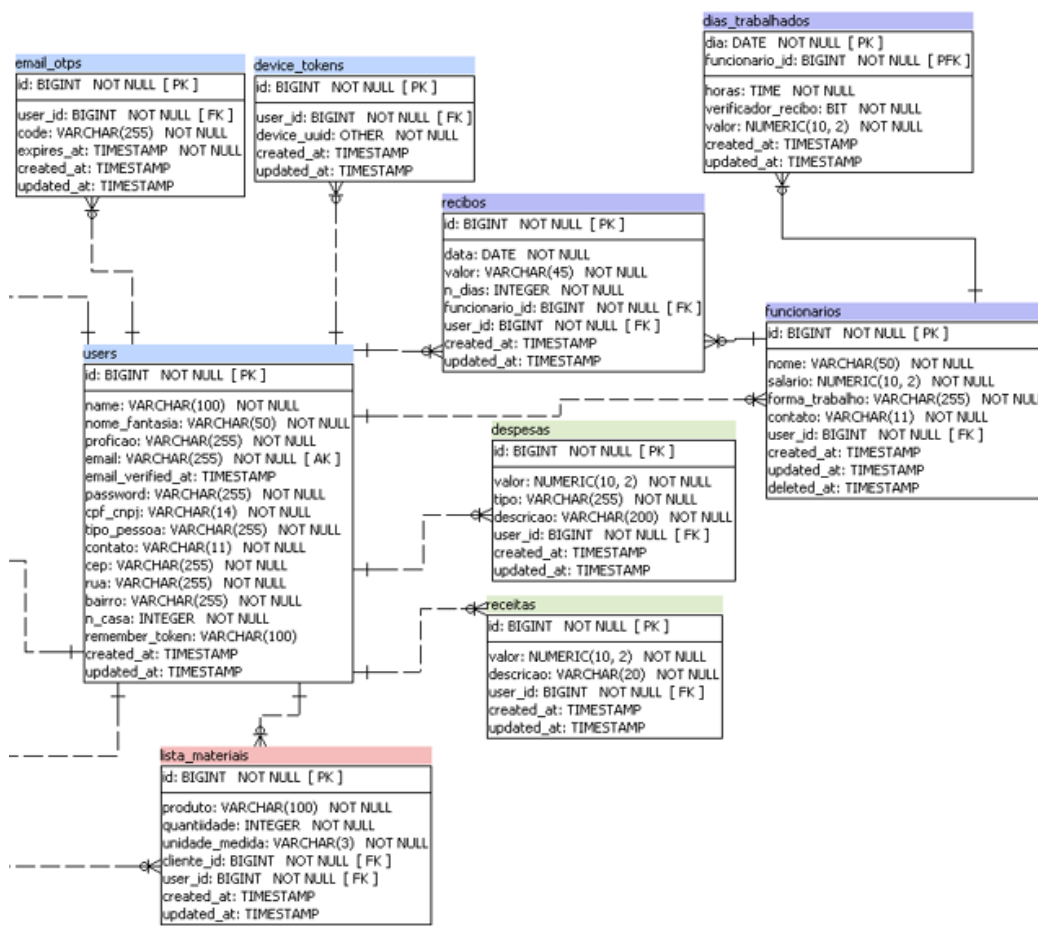


Fonte: O autor, 2025.

## Apêndice H - Segunda metade do modelo Lógico

A Figura H.11 traz a outra metade do modelo lógico, onde é possível visualizar as outras nove tabelas fundamentais. Na ilustração é possível visualizar a tabela central `users`, onde não utiliza a mesma tabela de endereços do cliente para evitar conflitos na lógica de negócios. Para que a tela de gestão de equipe mencionada na Figura 17 da Seção 6.2, tivesse integridade dos dados, foi necessário a criação de três tabelas, uma para funcionários, uma para dias trabalhados e outra dos recibos. A ilustração também demonstra outras tabelas fundamentais sendo elas receitas, despesas e as controladoras de seções `email_otps` e `device_tokens`, além de uma futura implementação já sendo mapeada que é a lista de materiais igual mencionado na Seção 7.

Figura H.11 – Segunda metade do modelo Lógico



Fonte: O autor, 2025.