



CENTRO UNIVERSITÁRIO BRASILEIRO - UNIBRA

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS



Recife

2025

**CENTRO UNIVERSITÁRIO BRASILEIRO - UNIBRA**  
**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**PROJETO DE STARTUP – BICOO MAPS**

Trabalho apresentado ao Centro Universitário Brasileiro - UNIBRA, como requisito para a conclusão do curso de Análise e Desenvolvimento de Sistemas.

Fomentador: Profº.: Ismael Rodrigues.

Sócios: Alan Hugo Lins Barbosa

Alesson Felipe Crispiniano de Barros

Heleno José da Silva Junior

Lukas Rodrigues da Silva

Mateus Paz de Oliveira

Natan José de Araújo

Thaís Mirelle Silva

Victor Alexandre Hermínio de Lima

Willian Silva

Recife

2025

## SUMÁRIO

<b>Histórico das Alterações.....</b>	<b>5</b>
<b>1 INTRODUÇÃO .....</b>	<b>6</b>
1.1 Problema.....	6
1.2 Solução .....	6
<b>2 PLANO DE GERENCIAMENTO .....</b>	<b>8</b>
2.1 Escopo do projeto .....	8
2.1.1 Descrição do Escopo .....	8
2.1.2 Entregas .....	9
2.1.3 Critérios de Aceitação .....	9
2.2 Cronograma .....	10
2.2.1 Equipe de Desenvolvimento.....	11
<b>3 PLANO DE NEGÓCIOS .....</b>	<b>12</b>
3.1 Caracterização do Empreendimento.....	12
3.2 PÚBLICO-ALVO.....	12
3.3 Canais de Comunicação e Estratégia de Marketing .....	13
3.4 Proposta de Valor .....	14
3.5 Análise de SWOT .....	15
3.7 Parcerias.....	16
3.8 Estrutura de Custos .....	17
3.8.1 Resumo de Custos .....	18
3.9 Monetização.....	18
<b>4 REQUISITOS FUNCIONAIS (CASOS DE USO) .....</b>	<b>19</b>
4.1 Glossário.....	19
4.2 Requisitos Funcionais (Casos de Uso) .....	20
4.3 Requisitos não funcionais.....	50

<b>5 TESTES DE QUALIDADE .....</b>	<b>52</b>
<b>6 TECNOLOGIAS UTILIZADAS.....</b>	<b>55</b>
6.1 Tecnologias do projeto .....	55
6.2 Visão geral.....	55
6.3 Back-end.....	55
6.4 Banco de dados.....	56
6.5 Front-end .....	56
6.6 Tempo real, chat e notificações .....	57
6.7 Mapas e geolocalização .....	57
6.8 Armazenamento de mídia.....	58
6.9 Pagamentos.....	58
<b>7 INFRAESTRUTURA E IMPLEMENTAÇÃO .....</b>	<b>59</b>
7.1 Segurança e conformidade (tecnologias aplicadas).....	59
<b>8 ANEXOS.....</b>	<b>60</b>
8.1 Telas do prestador.....	60
8.2 Telas do solicitante (cliente).....	60

## Histórico das Alterações

DATA	DESCRIÇÃO	RESPONSÁVEIS
<b>11/09/2025</b>	Início do levantamento de requisitos e registro das primeiras ideias do projeto.	Willian Silva e Heleno José
<b>13/09/2025</b>	Organização inicial dos requisitos funcionais e começo do mapeamento dos fluxos entre cliente e prestador.	Willian Silva e Heleno José
<b>16/09/2025</b>	Adição dos requisitos: Cancelamento por falta de acordo no reajuste (RFP07) e Recuperação de senha (RFS07).	Willian Silva
<b>20/09/2025</b>	Padronização dos critérios de aceite no formato Given/When/Then e revisão dos cenários BDD.	Willian Silva e Alesson Felipe
<b>23/09/2025</b>	Reordenação dos requisitos por perfil (Solicitante, Prestador, Comum) e detalhamento das telas principais.	Willian Silva e Heleno José
<b>25/09/2025</b>	Redefinição do escopo: plataforma com dois módulos web (prestashop e solicitante). Atualização dos fluxos.	Heleno José e Willian Silva
<b>27/09/2025</b>	Definição e documentação das telas Home (cliente e prestador), com navegação e opções principais.	Willian Silva e Heleno José
<b>01/10/2025</b>	Consolidação dos requisitos de cadastro e autenticação para ambos os perfis; inclusão de critérios de segurança.	Willian Silva
<b>06/10/2025</b>	Levantamento completo dos requisitos não funcionais (segurança, desempenho, disponibilidade e acessibilidade).	Willian Silva
<b>09/10/2025</b>	Elaboração do glossário técnico com termos essenciais usados no documento de requisitos.	Willian Silva
<b>13/10/2025</b>	Reformulação do requisito “Execução do Serviço” com detalhamento de estados, evidências e regras operacionais.	Victor Alexandre e Willian Silva
<b>17/10/2025</b>	Documentação das tecnologias escolhidas, com justificativa técnica e notas de implantação.	Heleno José
<b>18/10/2025</b>	Ajustes finais na redação das seções principais (plano de gerenciamento e tecnologias); preparação dos arquivos finais.	Willian Silva

# 1 INTRODUÇÃO

## 1.1 Problema

A contratação de serviços autônomos no Brasil, em especial nas áreas de manutenção doméstica, reparos emergenciais e serviços gerais, enfrenta três grandes dificuldades principais: Dificuldade em encontrar prestadores confiáveis: Muitos clientes não possuem indicações de profissionais qualificados, dependendo de buscas informais em redes sociais ou anúncios pouco seguros.

Falta de transparência na negociação: A definição do valor do serviço geralmente ocorre de forma verbal, sem registro formal, o que gera riscos de cobranças abusivas, má prestação de serviço e ausência de garantias para o cliente.

Insegurança nos pagamentos e comunicação: A ausência de plataformas digitais confiáveis leva a transações informais (dinheiro em espécie ou transferências pessoais), que não oferecem segurança, rastreabilidade ou garantia de recebimento para o prestador.

Segundo dados da PNAD Contínua/IBGE (2023), mais de 25 milhões de brasileiros atuam como trabalhadores por conta própria, sendo grande parte em atividades de prestação de serviços. Apesar desse contingente, a contratação de tais profissionais ainda se dá de forma fragmentada, pouco estruturada e com baixa confiança mútua.

## 1.2 Solução

Para mitigar esses problemas, propõe-se o desenvolvimento de uma plataforma web responsiva, composta por dois módulos principais:

- **Portal do Solicitante (Cliente):** possibilitará a criação de solicitações de serviço, definição de prazos, negociação de valores, acompanhamento em tempo real e avaliação de prestadores.
- **Portal do Prestador de Serviço:** permitirá que profissionais autônomos ativem disponibilidade, visualizem solicitações, negociem valores, executem atendimentos e recebam pagamentos de forma segura por meio de uma carteira digital.

**As principais funcionalidades incluem:**

- Cadastro diferenciado para clientes e prestadores.
- Solicitação e aceite de serviços em tempo real.
- Chat integrado para comunicação entre as partes.
- Acompanhamento do status e da execução do serviço.
- Pagamento em custódia, liberado apenas após a finalização confirmada.
- Negociação de reajuste de valores dentro da própria plataforma.
- Avaliações públicas de prestadores, garantindo confiabilidade.

**Essa solução atua como um intermediador digital seguro, promovendo:**

- Transparéncia: registros formais de solicitações, valores e histórico.
- Segurança: pagamentos intermediados e dados criptografados.
- Praticidade: solicitação de serviços em tempo real, com interface simples e responsiva.
- Escalabilidade: modelo replicável para diferentes cidades e regiões, com potencial de expansão como startup nacional.

## **2 PLANO DE GERENCIAMENTO**

### **2.1 Escopo do projeto**

#### **2.1.1 Descrição do Escopo**

O projeto Bicoo Maps surge como uma proposta de solução tecnológica inovadora voltada para o mercado de serviços sob demanda, um setor que cresce de forma acelerada devido à digitalização e à busca por conveniência e autonomia profissional.

A ideia central é criar uma plataforma capaz de conectar pessoas que precisam de serviços pontuais — como reparos, jardinagem, limpeza ou assistência técnica — com profissionais autônomos disponíveis em suas proximidades, promovendo praticidade, segurança e formalização do trabalho informal.

O Bicoo Maps busca preencher uma lacuna existente entre grandes plataformas genéricas (como aplicativos de transporte ou entregas) e o público autônomo local, que muitas vezes carece de ferramentas acessíveis para divulgação, negociação e recebimento de serviços. O projeto pretende oferecer um ecossistema digital humanizado, no qual o prestador e o cliente possam interagir de forma justa, transparente e direta.

O escopo deste projeto inclui todo o ciclo de desenvolvimento do projeto, abrangendo desde o levantamento de requisitos, prototipagem, modelagem de dados, até a implantação e documentação final.

Em termos de resultado, espera-se entregar um sistema funcional, validado por testes com usuários reais, capaz de demonstrar:

- A viabilidade operacional da plataforma no contexto urbano;
- A potencial escalabilidade do modelo de negócio;
- E a capacidade de transformação social e econômica para trabalhadores autônomos.

#### **Delimitação do escopo:**

O escopo contempla as etapas de planejamento, concepção, desenvolvimento, testes e implantação piloto, não abrangendo fases de marketing, expansão comercial.

## 2.1.2 Entregas

As entregas do projeto são planejadas para representar marcos de avanço técnico e organizacional, evidenciando o amadurecimento do produto e a capacidade da equipe de gerir o desenvolvimento de forma estruturada.

Etapa / Fase	Descrição Geral	Produto Entregável
<b>1. Planejamento e Elicitação</b>	Compreensão do problema, definição de público-alvo, análise de mercado e levantamento de requisitos.	Documento de requisitos e visão do produto.
<b>2. Prototipagem e Design de Experiência</b>	Criação das interfaces e definição da jornada do usuário.	Protótipos navegáveis e mapa de interação.
<b>3. Modelagem e Arquitetura do Sistema</b>	Estruturação do banco de dados, definição da arquitetura e infraestrutura web.	Documento técnico e modelo de dados.
<b>4. Desenvolvimento do MVP</b>	Implementação incremental das funcionalidades básicas da plataforma.	Versão funcional (protótipo navegável).
<b>5. Testes e Validação</b>	Verificação da usabilidade, desempenho e confiabilidade da aplicação.	Relatórios de teste e ajustes de melhoria.
<b>6. Implantação e Apresentação Final</b>	Disponibilização em ambiente de hospedagem e preparação para avaliação.	Sistema implantado, documentação e apresentação final.

Cada entrega será acompanhada de relatórios de progresso, reuniões de validação e revisões de sprint, seguindo práticas de metodologia ágil. O foco não é apenas o código desenvolvido, mas também a gestão do processo de construção e aprendizado.

## 2.1.3 Critérios de Aceitação

O sucesso do projeto será avaliado por meio de critérios qualitativos e quantitativos que refletem a efetividade do produto, a qualidade técnica e o cumprimento dos objetivos propostos.

Dimensão	Critério de Aceitação	Indicadores de Avaliação
<b>Execução do Projeto</b>	Cumprimento dos prazos e entregas estabelecidos no cronograma.	Percentual de tarefas concluídas em cada sprint.

<b>Qualidade do Produto</b>	Sistema funcional, com navegação fluida e baixo índice de falhas.	Testes de usabilidade, tempo médio de resposta e feedback dos usuários.
<b>Confiabilidade</b>	Estabilidade do sistema durante testes e disponibilidade contínua.	Logs de teste e uptime monitorado.
<b>Relevância Social e Técnica</b>	Clareza do propósito, inovação na proposta e aplicabilidade real.	Avaliação da banca e percepção de usuários durante o piloto.
<b>Documentação</b>	Entrega completa do documento técnico e manual do sistema.	Conformidade com o modelo acadêmico e normas ABNT.
<b>Conformidade Legal e Ética</b>	Respeito à LGPD e boas práticas de segurança de dados.	Checklist de privacidade e termos de uso documentados.

Esses critérios buscam equilibrar aspectos técnicos e gerenciais, assegurando que o projeto seja avaliado não apenas pelo funcionamento, mas também pelo valor gerado e pela organização do processo.

## 2.2 Cronograma

O cronograma foi estruturado de forma a permitir gestão iterativa e incremental, seguindo a filosofia ágil. Cada fase representa uma etapa de evolução do produto e aprendizagem da equipe, contemplando entregas intermediárias que fortalecem a coesão e o controle do projeto.

Fase	Período Estimado	Duração	Objetivos Principais
<b>1. Planejamento e Elicitação</b>	11/09/2025 – 25/09/2025	2 semanas	Definir público-alvo, objetivos e requisitos iniciais do sistema.
<b>2. Design e Prototipagem</b>	26/09/2025 – 10/10/2025	2 semanas	Elaborar fluxos de navegação e layout visual.
<b>3. Modelagem e Arquitetura</b>	11/10/2025 – 24/10/2025	2 semanas	Estruturar o banco de dados e a arquitetura do sistema.
<b>4. Desenvolvimento do MVP (Fase 1)</b>	25/10/2025 – 15/11/2025	3 semanas	Construir a base do sistema e validar os principais fluxos.
<b>5. Desenvolvimento do MVP (Fase 2)</b>	16/11/2025 – 06/12/2025	3 semanas	Ampliar funcionalidades e consolidar integração.
<b>6. Testes e Homologação</b>	07/12/2025 – 21/12/2025	2 semanas	Executar testes, corrigir falhas e revisar documentação.
<b>7. Implantação e Entrega Final</b>	22/12/2025 – 05/01/2026	2 semanas	Implantar o sistema em ambiente de produção e apresentar resultados.

## **Método de acompanhamento:**

O progresso será monitorado por meio de um quadro Kanban e reuniões de revisão quinzenais, permitindo visibilidade sobre o andamento das tarefas e tomada de decisão rápida. A equipe utilizará ferramentas como Jira, GitHub e plataformas de comunicação (Discord e Whatsapp) para comunicação, versionamento e armazenamento de artefatos.

Indicadores de controle:

- Percentual de conclusão de tarefas planejadas por sprint.
- Quantidade de erros encontrados em testes por módulo.
- Tempo médio entre identificação e correção de falhas.
- Grau de satisfação dos usuários nos testes de usabilidade.

### **2.2.1 Equipe de Desenvolvimento**

<b>Alunos</b>	<b>Atuação</b>
Willian Silva	Analista de Requisitos
Natan José de Araújo	Desenvolvedor de Banco de Dados
Mateus Paz de Oliveira	Desenvolvedor de Banco de Dados
Thaís Mirelle Silva	UX Designer
Victor Alexandre Hermínio de Lima	UX Designer
Lukas Rodrigues da Silva	Desenvolvedor Front-End
Heleno José da Silva Junior	Desenvolvedor Front-End
Alan Hugo Lins Barbosa	Desenvolvedor Back-End
Alesson Felipe Crispiniano de Barros	Quality Assurance

## **3 PLANO DE NEGÓCIOS**

### **3.1 Caracterização do Empreendimento**

O Bicoo Maps é uma startup digital voltada ao ramo de tecnologia e serviços, com foco na intermediação de prestação de serviços sob demanda. Seu propósito é conectar clientes que necessitam de serviços pontuais (como reparos, jardinagem, manutenção, pintura, limpeza, entre outros) a prestadores autônomos locais, de maneira rápida, segura e acessível. O sistema funciona por meio de uma plataforma web responsiva, que utiliza geolocalização e filtros inteligentes para aproximar as duas partes conforme critérios de distância, tipo de serviço e avaliação de reputação. A proposta visa digitalizar e profissionalizar o trabalho informal, oferecendo uma solução que une conveniência, transparência e geração de renda. O empreendimento posiciona-se como um ecossistema de conexão direta, eliminando intermediários e oferecendo autonomia para o prestador definir sua disponibilidade e preço, enquanto o cliente tem liberdade de escolha, acompanhamento do serviço e avaliação de qualidade. Dessa forma, o Bicoo Maps se insere no setor de tecnologia aplicada ao mercado de serviços locais (ServiceTech), com um modelo escalável e de baixo custo operacional.

### **3.2 PÚBLICO-ALVO**

O público-alvo do Bicoo Maps é dividido em dois grupos principais:

#### **a) Solicitantes (clientes):**

- Faixa etária: maiores de 18 anos;
- Localização: áreas urbanas e metropolitanas;
- Perfil: pessoas com rotina dinâmica, que valorizam praticidade e rapidez na contratação de serviços;
- Estilo de vida: público digitalizado, habituado a aplicativos de mobilidade, delivery e marketplaces;
- Motivação: necessidade de serviços domésticos e comerciais eventuais com segurança e agilidade.

**b) Prestadores de serviço (autônomos):**

- Profissionais autônomos ou informais de diversas áreas (eletricistas, encanadores, diaristas, pintores, jardineiros, técnicos, pedreiros, cuidadores etc.);
- Renda variável e busca por maior visibilidade e estabilidade de demanda;
- Região de atuação urbana, com foco inicial em Recife e Região Metropolitana;
- Perfil empreendedor, aberto à digitalização e ao uso de ferramentas tecnológicas simples.

O mercado-alvo inicial contempla mais de 40 mil trabalhadores informais na região metropolitana, conforme dados do IBGE (PNAD Contínua, 2024), e um público crescente de consumidores conectados.

### **3.3 Canais de Comunicação e Estratégia de Marketing**

O plano de marketing do Bicoo Maps combina estratégias digitais de baixo custo com ações locais de aproximação entre prestadores e clientes, buscando construir confiança e visibilidade da marca.

#### **Canais de divulgação:**

- Redes sociais (Instagram, Facebook, TikTok, LinkedIn): campanhas orgânicas e anúncios segmentados;
- Google Ads e SEO: indexação para busca por serviços locais (“pedreiro perto de mim”, “diarista em Recife” etc.);
- Parcerias com associações e ONGs de apoio a trabalhadores autônomos, oferecendo cadastro gratuito na fase piloto;
- Marketing de indicação (referral program): recompensas para prestadores e clientes que convidarem novos usuários;
- Conteúdo educativo: postagens com dicas sobre formalização, precificação e boas práticas de atendimento.

## **Estratégia inicial de marketing:**

- Lançamento regional com foco em Recife e Olinda;
- Campanha de validação (piloto) com influenciadores locais e grupos de Facebook/WhatsApp;
- Construção de autoridade por meio de avaliações positivas e depoimentos reais;
- Fidelização com notificações personalizadas, descontos e selos de qualidade para prestadores bem avaliados.

## **3.4 Proposta de Valor**

A proposta de valor do Bicoo Maps é simplificar o acesso a serviços e gerar oportunidades reais de renda.

A plataforma entrega ao cliente uma experiência completa e segura, desde a busca pelo prestador até a finalização do serviço, e ao profissional, visibilidade, autonomia e gestão digital do seu trabalho.

### **Benefícios para o cliente:**

- ✓ Economia de tempo e praticidade na busca por profissionais;
- ✓ Segurança e confiança através de avaliações e perfis verificados;
- ✓ Comunicação direta e acompanhamento do serviço em tempo real;
- ✓ Facilidade de pagamento e histórico de contratações.

### **Benefícios para o prestador:**

- ✓ Divulgação gratuita e segmentada de seus serviços;
- ✓ Acesso a um fluxo constante de clientes;
- ✓ Autonomia para definir valores e horários;
- ✓ Histórico e reputação digital que aumentam credibilidade e chances de contratação.

**Em síntese:** “O Bicoo Maps resolve o problema da falta de confiança e da dificuldade de encontrar prestadores de forma rápida, enquanto oferece visibilidade e oportunidades reais a quem vive de serviços locais.”

### 3.5 Análise de SWOT



### 3.6 Produtos e Serviços

O Bicoo Maps oferece dois módulos principais de serviço:

#### 1. Módulo Solicitante (Cliente):

- Cadastro e login;
- Solicitação de serviço com descrição e localização;
- Acompanhamento em tempo real;
- Avaliação e histórico de contratações.

## **2. Módulo Prestador (Autônomo):**

- Cadastro de perfil profissional e portfólio;
- Recebimento e aceitação de solicitações;
- Chat integrado com clientes;
- Geolocalização em tempo real;
- Carteira digital com saldo de serviços finalizados;
- Avaliação de clientes após conclusão.

Além disso, a plataforma conta com módulo administrativo, destinado ao gerenciamento de usuários, categorias, métricas e relatórios de desempenho.

### **3.7 Parcerias**

O modelo de negócio prevê parcerias estratégicas com instituições e serviços complementares:

- ONGs e cooperativas de trabalho informal, para captação de prestadores e treinamentos básicos de uso da plataforma;
- Instituições financeiras e fintechs, para implementação de pagamento e repasse digital;
- Empresas de tecnologia e hospedagem cloud, visando infraestrutura estável e segura;
- Prefeituras e secretarias municipais, para projetos de incentivo à formalização e economia solidária.

Essas alianças fortalecem a credibilidade da marca e aceleram a entrada em novos mercados.

### 3.8 Estrutura de Custos

Item	Descrição	Periodicidade	Custo Estimado (R\$)
Hospedagem Web / Servidor Cloud (RedHosting)	VPS Linux com 4 vCPUs, 8 GB RAM, 100 GB SSD, backup automático (AWS Lightsail, render ou Hostinger Cloud). Inclui MySQL e API hospedados.	Mensal	R\$ 35,00
Domínio e Certificado SSL	Registro de domínio .com.br e certificado SSL (Let's Encrypt gratuito ou Cloudflare Pro).	Anual	R\$ 40,00
Serviços de Mapas e Geolocalização (Google Maps API)	Consumo de API para rotas e localização (até 10.000 requisições/mês).	Mensal	R\$ 150,00
Serviços de Mensageria / Notificações (Firebase)	Chat em tempo real e push notifications via Firebase Cloud Messaging.	Mensal	R\$ 50,00
Gateway de Pagamento / Escrow Simulado	Taxas por transação (ex.: Pagar.me, Mercado Pago ou Stripe). Simulação de 3% por transação média.	Variável (5% da receita)	≈ R\$ 300,00
Ferramentas de Gestão e Desenvolvimento	Licenças Jira Premium, Canva Pro, GitHub e Figma.	Mensal	R\$ 180,00
Desenvolvimento e Manutenção Técnica	Atualizações de código, correção de bugs, novas funcionalidades (freelancer ou equipe interna parcial).	Mensal	R\$ 1.200,00
Suporte e Atendimento ao Usuário	Atendimento via WhatsApp Business, e-mail e chatbot (horário comercial).	Mensal	R\$ 400,00
Marketing Digital	Anúncios pagos em redes sociais, campanhas regionais e impulsionamentos.	Mensal	R\$ 800,00
Infraestrutura de Backup e Monitoramento	Backup automatizado diário e monitoramento de uptime/logs (Grafana + Loki + PM2).	Mensal	R\$ 120,00
Custos Administrativos	Internet, energia, transporte, coworking e reuniões presenciais.	Mensal	R\$ 300,00
Regularização e Conformidade (LGPD / Jurídico)	Registro de marca, adequação LGPD e consultoria jurídica inicial.	Anual	R\$ 1.500,00

### 3.8.1 Resumo de Custos

Categoria	Custo Mensal (R\$)	Custo Anual (R\$)
Infraestrutura Técnica (servidor, APIs, monitoramento)	R\$ 315,00	R\$ 3.780,00
Ferramentas e Licenças	R\$ 180,00	R\$ 2.160,00
Desenvolvimento / Manutenção	R\$ 1.200,00	R\$ 14.400,00
Atendimento e Suporte	R\$ 400,00	R\$ 4.800,00
Marketing e Divulgação	R\$ 800,00	R\$ 9.600,00
Custos Administrativos	R\$ 300,00	R\$ 3.600,00
Outros (LGPD, jurídico, registro, domínio)	R\$ 1.590,00 (anual)	R\$ 1.590,00
<b>Total Geral Estimado</b>	<b>≈ R\$ 3.460,00 / mês</b>	<b>≈ R\$ 40.000,00 / ano</b>

### 3.9 Monetização

O modelo de monetização foi definido com base em escalabilidade e sustentabilidade financeira, buscando equilíbrio entre receita e atratividade para usuários.

#### Fontes de receita previstas:

- **Comissão por intermediação:** percentual variado (entre 5% a 40%) sobre cada serviço concluído e pago via plataforma.
- **Planos premium para solicitante(clientes) (opcional):** assinatura mensal para destaque, prioridade em listagem e acesso a métricas exclusivas.
- **Publicidade segmentada:** espaços para anúncios de ferramentas, materiais ou serviços correlatos.
- **Marketplace de produtos de apoio (futuro):** venda de kits e equipamentos profissionais.

O modelo é híbrido, com variável (comissões) e receita recorrente (assinaturas) de forma opcional, permitindo crescimento sustentável e escalonável

## 4 REQUISITOS FUNCIONAIS (CASOS DE USO)

### 4.1 Glossário

Termo	Definição
API RESTful	Conjunto de endpoints HTTP que expõe funcionalidades do servidor (CRUD, consultas) seguindo princípios REST, permitindo que front-end e serviços externos se comuniquem com o back-end.
JWT (JSON Web Token)	Token digital assinado usado para autenticar usuários em chamadas à API; carrega informações do usuário e tem tempo de validade.
Refresh token	Token de longa duração usado para obter um novo JWT quando o JWT expirou, sem exigir novo login do usuário.
Hash (bcrypt / Argon2)	Função criptográfica que converte senhas em um valor irreversível; bcrypt e Argon2 são algoritmos recomendados para armazenar senhas com segurança.
TLS 1.3 (HTTPS)	Protocolo de criptografia que protege a comunicação entre cliente e servidor — obrigatório para tráfego seguro na web.
WebSocket	Protocolo que permite comunicação bidirecional em tempo real entre cliente e servidor (usado para chat e atualizações de rota).
Escrow (Carteira digital em custódia)	Mecanismo onde o pagamento fica retido (custodiado) pela plataforma até a confirmação da conclusão do serviço; evita repasses indevidos antes da entrega.
Gateway de pagamento / PCI DSS	Serviço que processa pagamentos. PCI DSS são normas de segurança que provedores de pagamento devem seguir para proteger dados de cartão.
KYC (Know Your Customer)	Conjunto de procedimentos para verificar a identidade de um usuário (necessário quando há saques e movimentações financeiras).
Cache (Redis / Memcached)	Camada de armazenamento temporário de dados em memória para acelerar respostas a operações frequentes (por exemplo, listagens).
Balanceamento de carga	Técnica/serviço que distribui tráfego entre múltiplos servidores para melhorar desempenho e disponibilidade.
CI/CD (Integração e Entrega Contínua)	Conjunto de práticas e ferramentas que automatizam build, testes e deploy do código, reduzindo erros manuais e acelerando entregas.
Swagger / OpenAPI	Ferramenta/formato de documentação automática de APIs que descreve endpoints, parâmetros e respostas de forma padronizada.
Firebase Cloud Messaging (Push Notifications)	Serviço para envio de notificações push a aplicativos/web browsers (usado para avisos em tempo real).
WCAG 2.1	Conjunto de diretrizes para garantir acessibilidade digital (leitores de tela, contraste, navegação por teclado etc.).
RBAC (Role-Based Access Control)	Controle de acesso baseado em papéis (por exemplo: administrador, prestador, solicitante), que restringe ações conforme permissão do papel.

Log / Auditoria	Registros detalhados de ações e eventos do sistema (login, cancelamento, transações) usados para rastreabilidade e investigação.
RTO (Recovery Time Objective)	Tempo máximo aceitável para recuperação do sistema após uma falha crítica; usado em planos de backup/recuperação.

Os Requisitos Organizacionais listado neste documento foram atrelados a um Identificador Único possibilitado ser referenciado e imediatamente identificado em todo o contexto. Deste modo, estes podem ser representados **RFSxx** e **RFPxx**, no instante em que se referir a um **Requisito Funcional Solicitante (Cliente) e Requisito funcional Prestador**. Sendo que, o **xx** representa um número sequência para representar os requisitos. Assim como para cada requisito foi relatado uma descrição e um nível de prioridade (**ESSENCIAL** - requisito extremamente necessário para o bom funcionamento do sistema; **IMPORTANTE** - os requisitos que o cliente considera importante estarem presente no sistema; **DESEJADO** - requisitos não urgentes que podem ser implementados futuramente).

Os **Requisitos não Funcionais** são identificados usando a referência **NFxx**, onde o **xx** representa um número sequência para representar os requisitos. Dos requisitos não funcionais sejam apresentados o tipo em que são classificados: **REQUISITOS DE PERFORMANCE, DE USABILIDADE, DE SEGURANÇA E DE HARDWARE E SOFTWARE**.

#### 4.2 Requisitos Funcionais (Casos de Uso)

##### Portal do Solicitante (Cliente)

###### [RFS01] Cadastro de Solicitante (Cliente)

Prioridade:  Essencial  Importante  Desejável

**Descrição:** O sistema deve permitir que o solicitante realize seu cadastro em duas etapas, informando suas informações pessoais e de endereço. Os campos obrigatórios são:

1. **Informações pessoais:** Nome completo, E-mail, Senha + confirmação de senha, CPF (11 dígitos), Data de nascimento, Sexo, Número de telefone, Foto de perfil (opcional).
2. **Endereço completo:** CEP, Rua, Número, Bairro, Cidade, Estado.

Após o preenchimento e validação, o sistema deve salvar os dados e disponibilizar acesso ao login.

### Critérios de Aceite:

Given que o cliente acessa a tela de cadastro, When informa seus dados pessoais obrigatórios, Then o sistema deve validar os campos e permitir o avanço para a próxima etapa.

Given que o cliente preenche os dados de endereço, When clica em Concluir Cadastro, Then o sistema deve validar os dados e salvar o cadastro.

Given que o cliente insere informações inválidas (ex.: CPF incorreto, e-mail sem formato válido, senha não confirmada), When tenta avançar, Then o sistema deve bloquear a ação e exibir mensagens de erro.

Given que o cliente não preenche todos os campos obrigatórios, When tenta concluir, Then o sistema deve alertar quais campos estão pendentes.

### Fluxo:



## [RFS02] Login

Prioridade:  Essencial  Importante  Desejável

**Descrição:** O sistema deve permitir que solicitantes de serviços realizem login com suas credenciais cadastradas (e-mail e senha).

### Critérios de Aceite:

Given que o cliente já possui cadastro, When acessa a tela de login e insere e-mail e senha válidos, Then o sistema deve autenticar e redirecionar para a Home.

Given que o cliente insere senha incorreta, When tenta acessar, Then o sistema deve exibir mensagem de erro.

Given que o cliente esqueceu a senha, When clica em Esqueci minha senha, Then o sistema deve redirecionar para o fluxo de recuperação de senha.

Given que o cliente não possui cadastro, When clica em Cadastrar, Then o sistema deve abrir o fluxo de cadastro.

### Fluxo:



## [RFS03] Tela Home – Navegação Principal

Prioridade:  Essencial  Importante  Desejável

**Descrição:** A tela inicial do sistema deve funcionar como hub principal de navegação, apresentando ao usuário os seguintes elementos:

Barra de navegação inferior com quatro opções fixas:

**Home:** retorna à tela inicial.

**Perfil:** acessa os dados pessoais e configurações do usuário.

**Chat:** abre as conversas entre prestadores e solicitantes.

Notificações: exibe alertas sobre solicitações, mensagens e atualizações de status. (implantação futura)

**Menu lateral (ícone de três barras no canto superior direito):**

Opções de configuração do sistema.

Função para deslogar (encerrar sessão).

**Área central da tela (conteúdo principal):**

Solicitar Serviço – botão para criação de um novo pedido.

Serviço Ativo – botão que lista o serviço em andamento e suas informações.

Serviços Concluídos – botão que lista o histórico de serviços finalizados.

Foto de perfil centralizado com seu respectivo nome.

**Critérios de Aceite:**

Given que o usuário acessa o sistema, When a tela Home é carregada, Then o sistema deve exibir foto do perfil do usuário e seu nome, juntamente com uma mensagem (ex.: bem vindo de volta!), botões centrais, barra inferior e menu superior.

Given que o usuário clica em Solicitar Serviço, When a ação é confirmada, Then o sistema deve redirecionar para o formulário de criação de serviço.

Given que o usuário clica em Serviço Ativo, When a ação é confirmada, Then o sistema deve listar o serviço em andamento e suas informações.

Given que o usuário clica em Serviços Concluídos, When a ação é confirmada, Then o sistema deve exibir o histórico de serviços finalizados.

Given que o usuário clica em Perfil na barra inferior, When a ação é confirmada, Then o sistema deve abrir os dados e configurações do usuário.

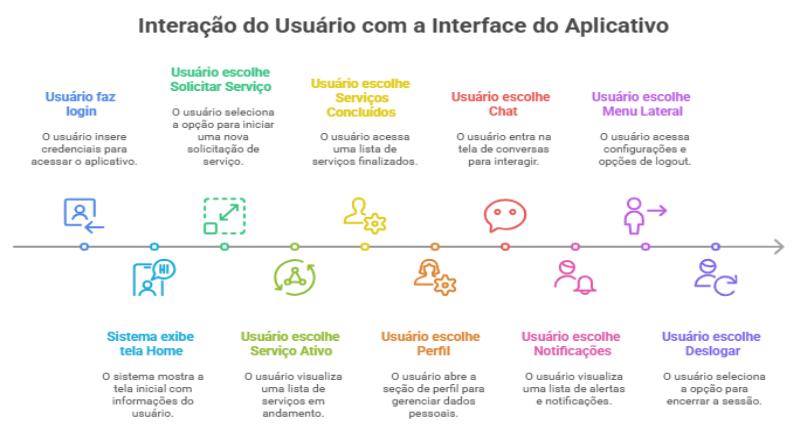
Given que o usuário clica em Chat na barra inferior, When a ação é confirmada, Then o sistema deve abrir a lista de conversas.

Given que o usuário clica em Notificações, When a ação é confirmada, Then o sistema deve exibir os alertas pendentes.

Given que o usuário acessa o menu lateral (três barras), When abre as opções, Then deve visualizar as configurações e a opção de deslogar.

Given que o usuário clica em Deslogar, When a ação é confirmada, Then o sistema deve encerrar a sessão e redirecionar para a tela de login.

## Fluxo:



## [RFS04] Exibir e Editar Perfil do Usuário

Prioridade:  Essencial  Importante  Desejável

**Descrição:** O sistema deve permitir que o usuário acesse a tela de Perfil, onde serão exibidas suas informações pessoais cadastradas, incluindo: Nome completo, e-mail, CPF, data de nascimento, Sexo, Número de telefone e endereço completo (CEP, rua, número, bairro, cidade, estado)

Na mesma tela, o usuário deve ter a opção de editar seus dados, acessando o modo de edição ao clicar no ícone de lápis. Após realizar alterações, deve haver a opção de salvar as informações (ícone de disquete).

### Critérios de Aceite:

Given que o usuário acessa o menu Perfil, When a tela é carregada, Then o sistema deve exibir todas as informações cadastradas.

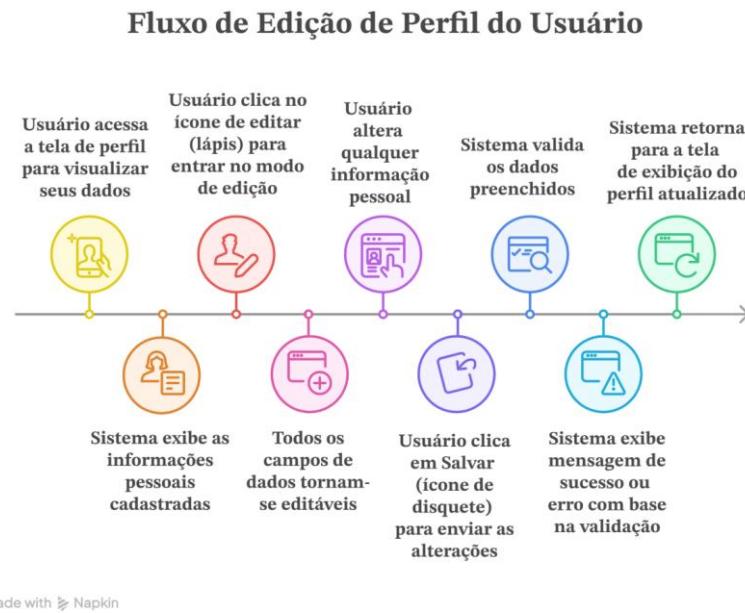
Given que o usuário clica no ícone de lápis, When a tela muda para modo de edição, Then todos os campos devem se tornar editáveis.

Given que o usuário altera informações e clica no ícone de salvar, When os dados são validados, Then o sistema deve atualizar o perfil e exibir mensagem de confirmação.

Given que o usuário altera informações inválidas (ex.: e-mail incorreto, CPF em formato inválido), When tenta salvar, Then o sistema deve impedir a ação e exibir mensagem de erro.

Given que o usuário não deseja salvar as alterações, When volta à tela anterior, Then os dados devem permanecer inalterados.

## Fluxo:



## [RFS05] Serviços finalizados

Prioridade: ● Essencial ○ Importante ○ Desejável

**Especificação:** O sistema deve permitir que o solicitante visualize o histórico de serviços finalizados, exibindo informações resumidas para referência futura.

### Critérios de Aceite:

Given que o cliente possui serviços finalizados, When acessa a tela Histórico de Serviços, Then deve visualizar a lista com resumo de cada serviço concluído.

Given que o cliente seleciona um serviço finalizado, When abre os detalhes, Then deve visualizar todas as informações completas do serviço.

Given que não há serviços finalizados, When acessa a tela, Then o sistema deve exibir a mensagem “Você ainda não possui serviços finalizados”.

## Fluxo:



## [RFS06] Criar Solicitação de Serviço

**Prioridade:**  Essencial  Importante  Desejável

**Especificação:** O cliente deve poder solicitar um serviço, informando detalhes e urgência (**será cobrado uma porcentagem por nível de urgência**), forma de pagamento.

### Critérios de Aceite:

Given que o cliente preencheu descrição, local, valor, forma de pagamento e urgência, When confirma a solicitação, Then o serviço deve ser registrado e enviado para prestadores disponíveis.

Given que o cliente não preencheu todos os campos obrigatórios, When tenta confirmar, Then o sistema deve impedir e exibir mensagem de erro.

## Fluxo:



## [RFS07] Selecionar Prestador

Prioridade:  Essencial  Importante  Desejável

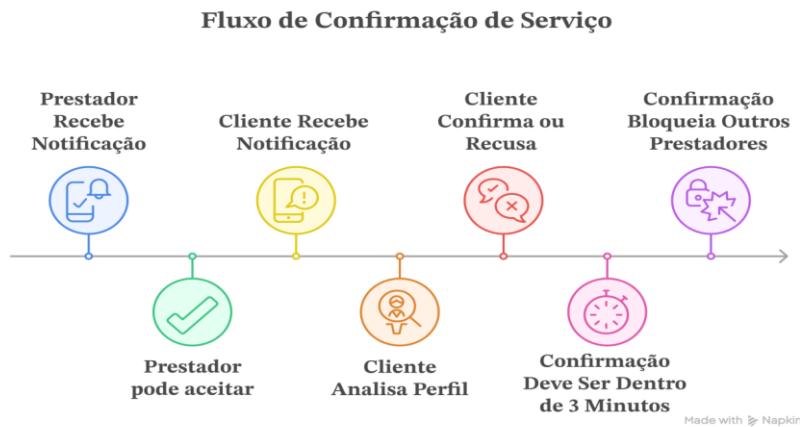
Especificação: O cliente deve poder escolher entre os prestadores que aceitaram a solicitação.

### Critérios de Aceite:

Given que um prestador aceitou, When o cliente visualiza o perfil, Then deve conseguir confirmar ou recusar.

Given que o cliente confirmou, When a confirmação é registrada, Then apenas esse prestador segue no fluxo.

## Fluxo:



## [RFS08] Acompanhamento de Serviços Ativos

Prioridade: ● Essencial ○ Importante ○ Desejável

**Especificação:** O sistema deve permitir que o solicitante acompanhe em tempo real o serviço ativo, com informações detalhadas sobre o status da execução, a localização do prestador e a comunicação via chat.

### Critérios de Aceite:

Given que o cliente possui serviço em andamento, When acessa a tela Serviços Ativos, Then deve visualizar a lista com resumo do(s) serviço(s).

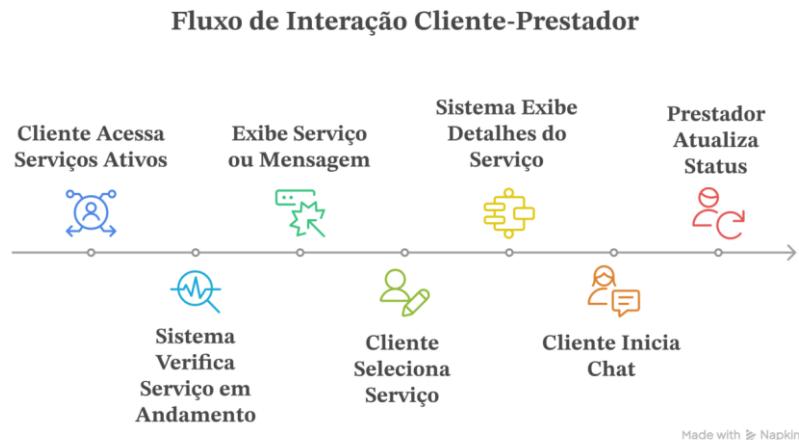
Given que o cliente seleciona um serviço ativo, When abre os detalhes, Then deve visualizar todas as informações completas relacionadas ao serviço, incluindo mapa em tempo real.

Given que não há serviços ativos, When acessa a tela, Then o sistema deve exibir a mensagem “Você não possui serviço ativo no momento”.

Given que o prestador foi confirmado, When inicia deslocamento, Then o cliente deve visualizar sua rota no mapa em tempo real.

Given que o cliente abre o chat, When envia mensagem, Then o prestador deve recebê-la instantaneamente.

### Fluxo:



### [RFS09] Finalização do Serviço

Prioridade: ● Essencial ○ Importante ○ Desejável

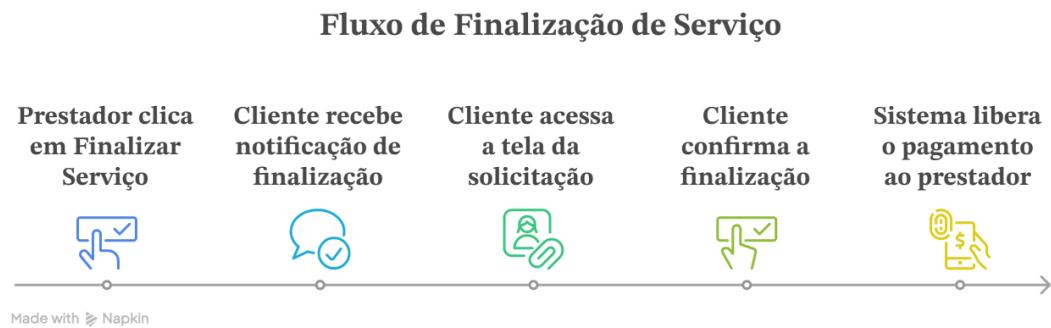
**Especificação:** O cliente deve confirmar a conclusão do serviço após a finalização feita pelo prestador, para que o pagamento seja liberado.

### Critérios de Aceite:

Given que o prestador finalizou, When o cliente confirma, Then o pagamento deve ser liberado ao prestador.

Given que o cliente não confirma em até X horas, When o prazo expira, Then o sistema pode liberar o pagamento automaticamente (configuração da plataforma).

## Fluxo:



## [RFS10] Avaliação do Prestador

Prioridade:  Essencial  Importante  Desejável

**Especificação:** Após a conclusão do serviço, o cliente deve avaliar o prestador por meio de nota e comentário, para alimentar o histórico e credibilidade da plataforma.

### Critérios de Aceite:

Given que o cliente avaliou, When envia nota/comentário, Then deve ficar registrado no perfil do prestador.

Given que o cliente não avalia no momento da finalização, When acessar o histórico, Then deve ter a opção de avaliar posteriormente.

## Fluxo:



## [RFS11] Cancelamento de serviço

Prioridade:  Essencial  Importante  Desejável

**Especificação:** Permitir que o solicitante cancele um serviço a qualquer momento, desde que informe o motivo.

O sistema deve diferenciar o tipo de cancelamento (antes da confirmação, após a confirmação e após o início da rota do prestador) e aplicar as regras adequadas:

- Reembolso integral em todos os casos;
- Registro do motivo do cancelamento;
- Restrição temporária para nova solicitação, se o prestador já estiver em rota.

### Critérios de Aceite:

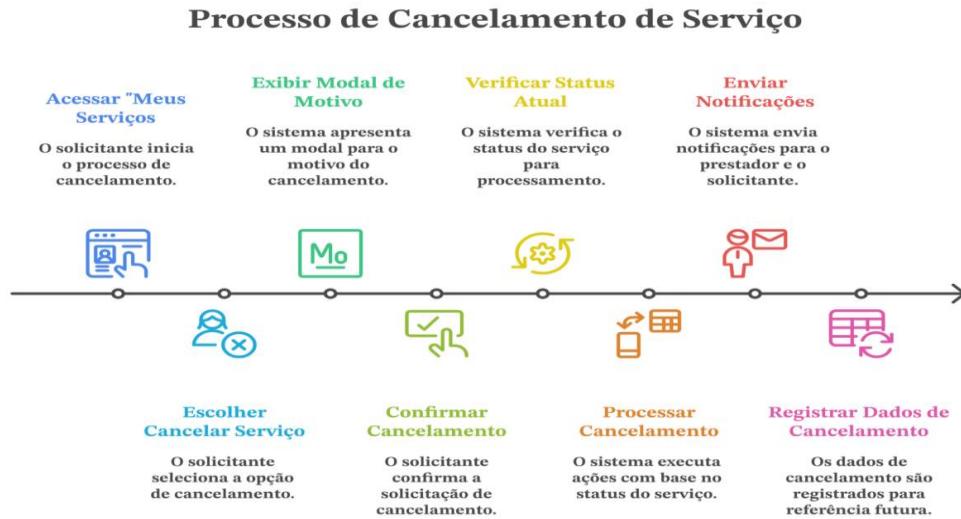
Given que o solicitante cancelou antes do prestador confirmar, When solicita o cancelamento, Then o serviço é excluído da fila de solicitações e não há qualquer bloqueio ou penalidade.

Given que o solicitante cancela após o prestador confirmar, mas antes do início da rota, When confirma o cancelamento, Then o valor é reembolsado integralmente e o serviço é removido da listagem.

Given que o solicitante cancela após o prestador confirmar e já estar em rota, When executa o cancelamento e informa o motivo, Then o cliente recebe o reembolso total e fica temporariamente bloqueado para criar novas solicitações por 2 horas.

Given que o cancelamento é recorrente (ex.: mais de 3 vezes em curto intervalo), When o sistema detecta o padrão, Then deve notificar o cliente e aplicar monitoramento preventivo (registro no histórico e alerta **administrativo**).

## Fluxo



## Portal do Prestador

### [RFP01] Cadastro de Prestador de Serviços

Prioridade:  Essencial  Importante  Desejável

**Especificação:** O sistema deve permitir que um novo prestador de serviços realize seu cadastro em três etapas sequenciais, informando:

- 1. Informações pessoais:** Nome completo, E-mail, Senha + confirmação de senha, CPF, data de nascimento, sexo, número de telefone, foto de perfil (Obrigatório)
- 2. Endereço completo:** CEP, rua, número, bairro, cidade, estado

- 3. Informações profissionais:** Descrição sobre si (resumo do perfil), até 3 habilidades principais, Upload de documentos (certificações e currículo)

O cadastro só será concluído se todas as etapas obrigatórias forem preenchidas corretamente.

### Critérios de Aceite:

Given que o prestador acessa a tela de cadastro, When informa seus dados pessoais obrigatórios, Then o sistema deve validar os campos e permitir o avanço para a próxima etapa.

Given que o prestador preenche as informações de endereço, When clica em "Próximo", Then o sistema deve validar os campos e avançar para a etapa profissional.

Given que o prestador adiciona suas habilidades e anexa documentos, When clica em "Concluir Cadastro", Then o sistema deve salvar todas as informações e exibir mensagem de sucesso.

Given que o prestador insere informações inválidas (ex.: e-mail sem formato válido, CPF incorreto, senha não confirmada), When tenta avançar, Then o sistema deve bloquear a ação e exibir mensagens de erro específicas.

Given que o prestador não preenche os campos obrigatórios, When tenta concluir, Then o sistema deve alertar quais campos estão pendentes.

### Fluxo:



## [RFP02] Login de Prestador

Prioridade:  Essencial  Importante  Desejável

**Descrição:** O sistema deve permitir que prestadores de serviços realizem login com as credenciais cadastradas (e-mail e senha).

### Critérios de Aceite:

Given que o prestador já possui cadastro, When acessa a tela de login e insere e-mail e senha válidos, Then o sistema deve autenticar e redirecionar para a tela Home.

Given que o prestador insere senha incorreta, When tenta acessar, Then o sistema deve exibir mensagem de erro.

Given que o prestador esqueceu a senha, When clica em "Esqueci minha senha", Then o sistema deve redirecionar para o fluxo de recuperação de senha.

Given que o prestador não possui cadastro, When clica em "Cadastrar", Then o sistema deve abrir o fluxo de cadastro.

### Fluxo:



## [RFP03] Tela Home – Prestador de Serviços

Prioridade:  Essencial  Importante  Desejável

**Descrição:** A tela inicial do prestador deve apresentar as informações principais relacionadas à sua disponibilidade e ao seu perfil. O prestador pode alternar entre modo online e modo offline.

### Modo Online

- Exibir mapa em tempo real, centralizado na localização do prestador.
- Mostrar solicitações de serviço próximas representadas por marcadores.
- Para cada solicitação, exibir as seguintes informações resumidas (Urgência, ícone e nome do cliente, tipo de serviço, descrição, data preferencial, horário e valor final).
- Oferecer botões de ação Aceitar e Recusar → descritos em detalhe no requisito [RFP05].

### Modo Offline

- Exibir resumo do perfil do prestador:
- Foto de perfil.
- Mensagem de boas-vindas personalizada.
- Lista de habilidades cadastradas.
- Avaliações recentes com comentários.
- Saldo na carteira digital.

### Critérios de Aceite:

Given que o prestador está online, When acessa a Home, Then deve visualizar o mapa com solicitações próximas.

Given que o prestador está offline, When acessa a Home, Then deve visualizar o resumo de perfil.

Given que o prestador clica em uma solicitação exibida no mapa, When seleciona, Then deve visualizar o resumo e ter as opções de Aceitar ou Recusar (detalhes no [RFP05]).

Given que o prestador altera o status (online/offline), When confirma a ação, Then o sistema deve atualizar a exibição correspondente.

### Fluxo:



### [RFP04] Exibir e Editar Perfil do Prestador

Prioridade:  Essencial  Importante  Desejável

**Descrição:** O sistema deve permitir que o prestador de serviços accesse a tela de Perfil, onde serão exibidas suas informações pessoais e profissionais cadastradas, incluindo: Nome completo, e-mail, CPF, data de nascimento, sexo, número de telefone, endereço completo (CEP, rua, número, bairro, cidade, estado), principais habilidades (até 3), documentos anexados (certificações, currículo, comprovações) e descrição.

Na mesma tela, o prestador deve ter a opção de editar seus dados pessoais e profissionais, acessando o modo de edição ao clicar no ícone de lápis. Após realizar alterações, deve haver a opção de salvar as informações (ícone de disquete).

### Critérios de Aceite:

Given que o prestador acessa o menu Perfil, When a tela é carregada, Then o sistema deve exibir todas as informações pessoais e profissionais cadastradas.

Given que o prestador clica no ícone de lápis, When a tela muda para modo de edição, Then todos os campos devem se tornar editáveis.

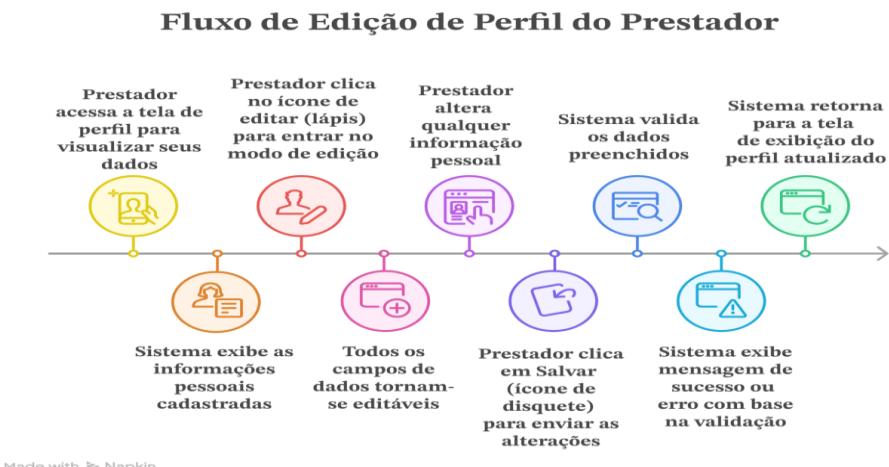
Given que o prestador altera informações e clica no ícone de salvar, When os dados são validados, Then o sistema deve atualizar o perfil e exibir mensagem de confirmação.

Given que o prestador anexa um documento (certificação/currículo), When o upload é concluído, Then o documento deve ser vinculado ao perfil.

Given que o prestador informa dados inválidos (ex.: e-mail incorreto, CPF inválido), When tenta salvar, Then o sistema deve impedir a ação e exibir mensagem de erro.

Given que o prestador não deseja salvar as alterações, When volta à tela anterior, Then os dados devem permanecer inalterados.

### Fluxo:



## [RFP05] Visualizar, Aceitar ou Recusar Serviço

Prioridade:  Essencial  Importante  Desejável

**Especificação:** O sistema deve permitir que o prestador visualize em tempo real as solicitações de serviço disponíveis no mapa e/ou lista. O prestador pode analisar os detalhes do serviço antes de decidir aceitar ou recusar.

- **Aceitar Serviço:** quando aceita, o sistema deve vincular o serviço ao prestador e aguardar a confirmação do cliente. Após a confirmação, habilitar o chat e a rota no mapa para acompanhamento do deslocamento até o local.
- **Recusar Serviço:** quando recusa, o sistema deve descartar aquela oferta, retornando o prestador ao mapa para continuar disponível a outros serviços sem penalidade.

### Critérios de Aceite:

Given que o prestador acessa o mapa de serviços, When seleciona uma solicitação, Then deve visualizar o resumo completo (tipo, endereço, descrição, valor, urgência, cliente).

Given que o prestador clica em Aceitar, When o cliente confirma, Then o sistema deve vincular o serviço ao prestador, habilitar o chat e liberar a rota no mapa.

Given que o prestador clica em Recusar, When volta ao mapa, Then o serviço deve ser descartado para ele e o sistema deve mantê-lo disponível para receber novas solicitações.

Given que o prestador já aceitou um serviço, When tenta aceitar outro, Then o sistema deve impedir a ação até a finalização ou cancelamento do serviço em andamento.

Given que o prestador recusa repetidas vezes, When permanece online, Then o sistema deve seguir oferecendo outros serviços sem bloqueio.

## Fluxo:

### Fluxo de Seleção de Serviço do Prestador



Made with Napkin

## [RFP06] Execução do Serviço

Prioridade: ● Essencial ○ Importante ○ Desejável

**Especificação:** O sistema deve controlar e expor todo o ciclo de execução do serviço, desde o aceite do prestador até a finalização e liberação do pagamento. O controle deve contemplar os estados do serviço, atualização em tempo real (rota/GPS), confirmações de chegada, início e término da execução, regras de tempo de espera (no-show), tratamento de cancelamentos e disputas, e integração com o módulo financeiro (escrow/carteira) para retenção e liberação de valores. A implementação deve priorizar evidências automáticas (timestamps, coordenadas GPS, logs de chat) para suportar decisões automáticas e manuais em disputas.

### Estados do Serviço (mínimos)

- Publicado (visível no mapa)
- Aceito (prestador clicou aceitar — pendente confirmação do cliente)
- Confirmado (cliente confirmou o prestador)
- EmRota (prestador iniciou deslocamento / rota ativa)
- NoLocal (prestador marcou “Cheguei”)
- EmExecucao (prestador declarou início efetivo da execução)
- Finalizado (prestador marcou finalização — pendente confirmação do cliente)
- Concluido (cliente confirmou finalização; pagamento liberado)

- Cancelado (cancelamento por cliente ou prestador; sub-motivo registrado)
- EmDisputa (valor/conduta sob análise administrativa)

## **Critérios de Aceite:**

### **1. Aceite e confirmação**

Given que o prestador clica em Aceitar, When o cliente confirma em até 3 minutos, Then o serviço passa para confirmado, o chat é habilitado e a rota é mostrada ao cliente e prestador.

### **2. Inicio de deslocamento / rota**

Given que o prestador confirmou saída para o serviço, When inicia deslocamento (evento de rota iniciado ou envio do comando “Iniciar Rota”), Then o serviço passa para Em Rota e o cliente visualiza o trajeto em tempo real.

### **3. Confirmação de chegada**

Given que o prestador marca Cheguei no app, When o evento é recebido pelo backend, Then o status atualiza para NoLocal e o cliente é notificado; as coordenadas GPS e timestamp são registradas.

### **4. Início da execução**

Given que o prestador clicou para iniciar o trabalho, When o prestador sinaliza “Iniciar Execução”, Then o serviço passa a EmExecucao e o tempo de execução começa a ser medido.

### **5. Finalização e liberação do pagamento**

Given que o prestador marcou Finalizar, When o cliente confirma a conclusão (ou o prazo de confirmação expira conforme política), Then o serviço passa a Concluido e o valor é liberado do escrow para a carteira do prestador.

## 6. No-show do prestador

Given que o prestador não inicia deslocamento dentro do prazo aceitável após confirmação (configurável), When o cliente reporta “Prestador não iniciou deslocamento” ou sistema detecta ausência de movimento GPS significativo, Then o cliente pode cancelar com justificativa automática e abrir disputa; o sistema registra evidências e aplica regras financeiras conforme política de cancelamento. (Ver motivos de cancelamento para detalhamento de penalidades).

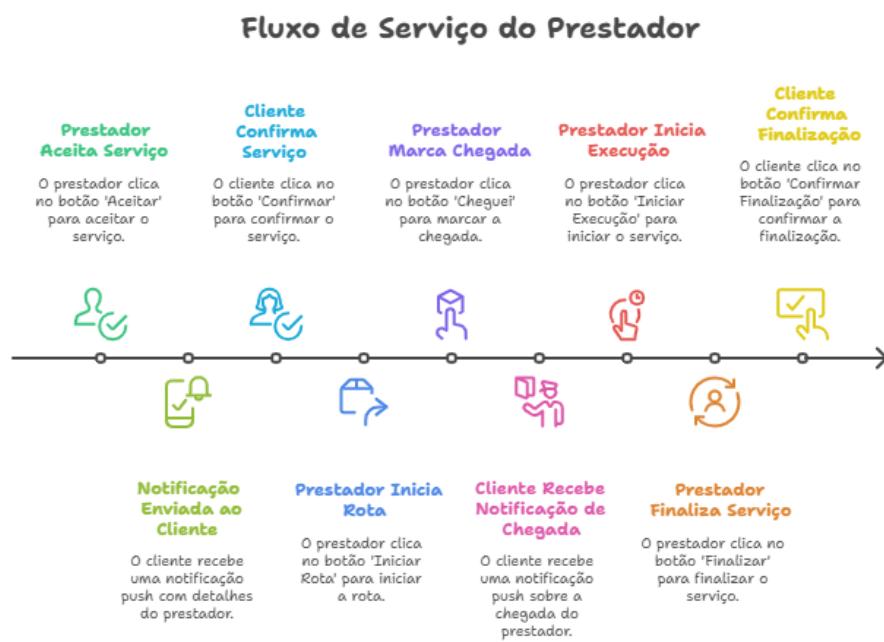
## 7. Ausência / não comparecimento do cliente

Given que o prestador marcou Chegou e aguardou o tempo de espera padrão (p.ex. 15 minutos — parametrizável), When o cliente não comparece, Then o prestador pode cancelar por “Contratante ausente”; o sistema deve registrar evidências (timestamp, GPS, print chat) e aplicar regras de compensação/penalidade conforme política.

## 8. Casos de disputa/conduta inadequada

Given que há alegação de conduta inadequada (cliente ou prestador), When uma das partes anexa evidências, Then o sistema muda o status para Em Disputa, retém o valor no escrow e notifica a equipe administrativa para análise.

**Fluxo:**



## [RFP07] Negociação/Reajuste de Valor

Prioridade:  Essencial  Importante  Desejável

**Especificação:** Em caso de escopo maior, prestador pode solicitar reajuste de valor dentro do app antes da conclusão.

### Critérios de Aceite:

Given que prestador solicitou reajuste, When cliente aceita, Then valor final é atualizado e reautorizado.

Given que cliente recusa, When rejeita a proposta, Then valor original permanece.

Given que cliente não responde em X min, When expira prazo, Then serviço segue com valor original.

### Fluxo:

#### Processo de Solicitação de Reajuste de Preço



## [RFP08] Cancelamento de serviço por desacordo de valor

Prioridade:  Essencial  Importante  Desejável

**Especificação:** O prestador deve poder cancelar o serviço dentro do sistema caso não haja acordo sobre o reajuste de valor solicitado, evitando a execução sem concordância.

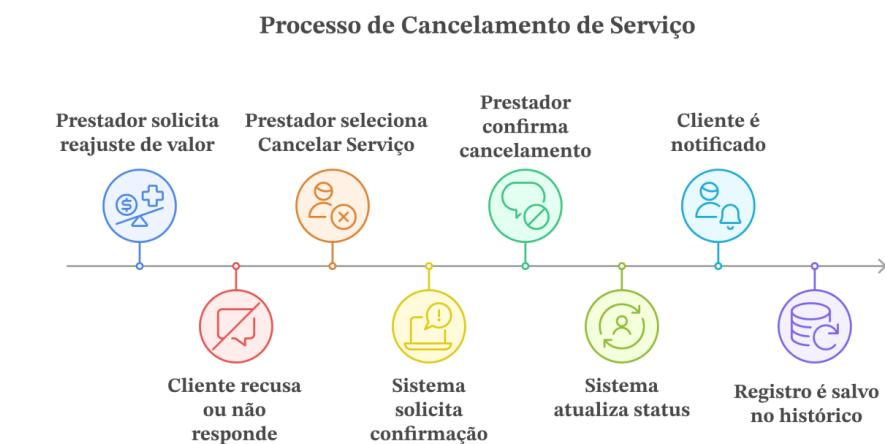
### Critérios de Aceite:

Given que o prestador solicitou reajuste, When o cliente recusa ou não responde, Then o prestador deve poder cancelar o serviço.

Given que o prestador cancelou, When a ação é confirmada, Then o cliente deve ser notificado imediatamente sobre o cancelamento.

Given que houve cancelamento, When registrado no sistema, Then deve ser armazenado no histórico do serviço (com motivo: “cancelado por falta de acordo”).

### Fluxo:



Made with Napkin

## [RFP09] Pagamentos

Prioridade:  Essencial  Importante  Desejável

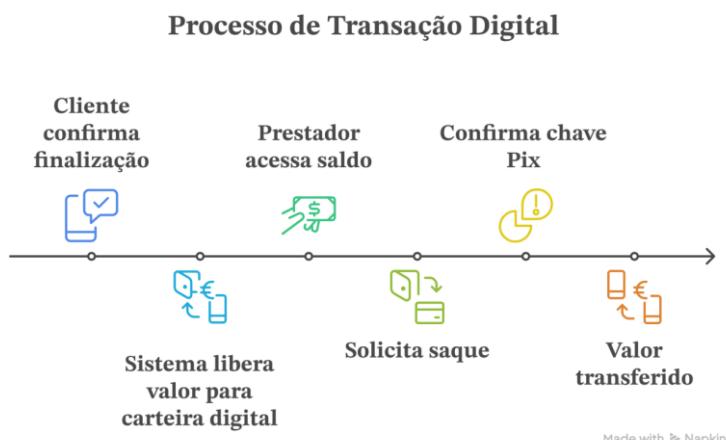
Especificação: O prestador deve receber o pagamento em carteira digital com opção de saque.

### Critérios de Aceite:

Given que cliente confirmou finalização, When sistema libera saldo, Then prestador deve vê-lo na carteira.

Given que prestador solicitou saque, When confirma chave Pix, Then valor deve ser transferido.

### Fluxo:



## [RFP10] Avaliação do Cliente pelo Prestador

Prioridade:  Essencial  Importante  Desejável

Descrição: O sistema deve permitir que o prestador de serviços avalie o cliente ao final da execução do serviço, registrando uma nota e um comentário opcional. O objetivo é criar um histórico de confiabilidade também para os clientes, auxiliando prestadores em futuras escolhas de atendimento.

## Critérios de Aceite:

Given que o serviço foi concluído, When o prestador acessa a tela de avaliação, Then deve visualizar o formulário para avaliar o cliente.

Given que o prestador atribui uma nota e comentário, When envia a avaliação, Then o sistema deve registrar os dados vinculados ao perfil do cliente.

Given que o prestador não deseja adicionar comentário, When envia apenas a nota, Then o sistema deve registrar a avaliação normalmente.

Given que o prestador já avaliou o cliente em um serviço, When tenta avaliar novamente, Then o sistema deve impedir a duplicidade.

Given que o prestador não envia nenhuma avaliação, When o serviço for encerrado, Then o sistema deve permitir que prossiga, mas marcar o status como “não avaliado”.

## Fluxo:



Made with Napkin

## [RFP11] Cancelamento pelo Prestador

Prioridade:  Essencial  Importante  Desejável

**Especificação:** Permitir que o prestador cancele um serviço por motivos operacionais, técnicos ou de segurança, registrando a causa e informando o cliente. O sistema deve aplicar bloqueios temporários progressivos em casos de cancelamentos frequentes por perda de conectividade ou transporte, garantindo controle de qualidade e confiabilidade da plataforma.

### Critérios de Aceite (BDD):

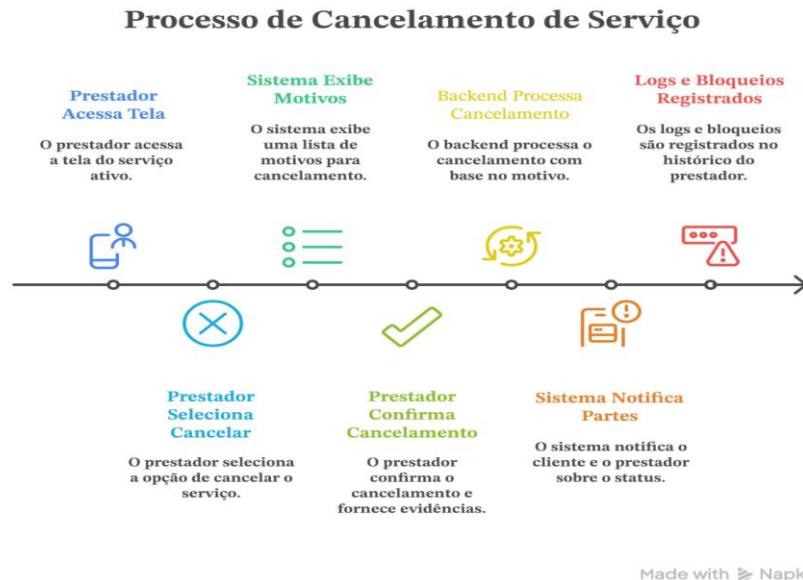
Given que o prestador cancela por endereço incorreto, inacessível ou cliente ausente, When registra o motivo e evidência, Then o serviço é encerrado sem penalidade e o cliente é notificado.

Given que o prestador cancela por motivos de conectividade ou transporte, When informa o cancelamento, Then o serviço volta a ficar disponível no mapa e o prestador é bloqueado por 30 minutos antes de aceitar novos serviços.

Given que o prestador cancela novamente por conectividade ou transporte após o primeiro bloqueio, When o sistema detecta reincidência, Then o bloqueio aumenta progressivamente para 1h, 1h30min, 2h e assim por diante.

Given que o prestador cancela por motivos de segurança, When anexa evidência, Then o cancelamento ocorre sem penalidade e o caso é registrado como prioridade para análise da administração.

## Fluxo:



## Observações / Regras Adicionais:

- Bloqueios temporários não afetam reputação, apenas disponibilidade para novos serviços.
- Se o prestador atingir 3 cancelamentos consecutivos por motivos operacionais, o sistema gera alerta para supervisão administrativa.
- Os bloqueios são calculados de forma automática e progressiva:
  - 1<sup>a</sup> ocorrência → 30 min
  - 2<sup>a</sup> ocorrência → 1 h
  - 3<sup>a</sup> ocorrência → 1 h 30 min
  - 4<sup>a</sup> ocorrência → 2 h
- Caso o prestador finalize serviços com frequência normal após bloqueios, o tempo volta ao padrão (30 min).
- Todas as ações são registradas com timestamp, coordenadas e logs de status (“Em rota”, “Cheguei”, “Cancelado”).
- Fluxo vinculado a RFP08 (Carteira e Pagamentos) e RFP06 (Aceitar/Recusar).

## Requisitos em comum

### [RFC12] Recuperação de Senha

Prioridade:  Essencial  Importante  Desejável

**Especificação:** O sistema deve permitir que usuários (clientes ou prestadores) recuperem o acesso em caso de esquecimento da senha, por meio de envio de link de redefinição via e-mail ou código temporário via SMS.

#### Critérios de Aceite:

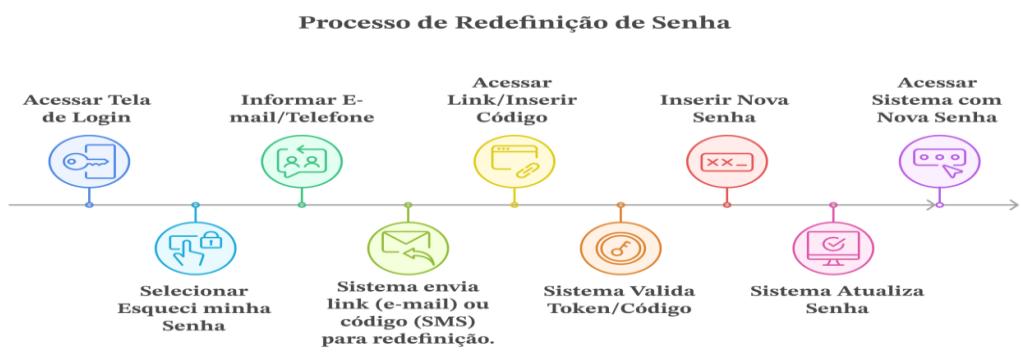
Given que o usuário esqueceu a senha, When solicita recuperação, Then o sistema deve enviar um link ou código de verificação para o contato cadastrado.

Given que o usuário recebeu o link/código, When valida corretamente, Then o sistema deve permitir cadastrar uma nova senha.

Given que a senha foi redefinida, When a alteração é confirmada, Then o usuário deve conseguir acessar normalmente com a nova credencial.

Given que o link/código expirou, When o usuário tenta usar, Then o sistema deve exibir mensagem de erro e solicitar novo processo de recuperação.

#### Fluxo:

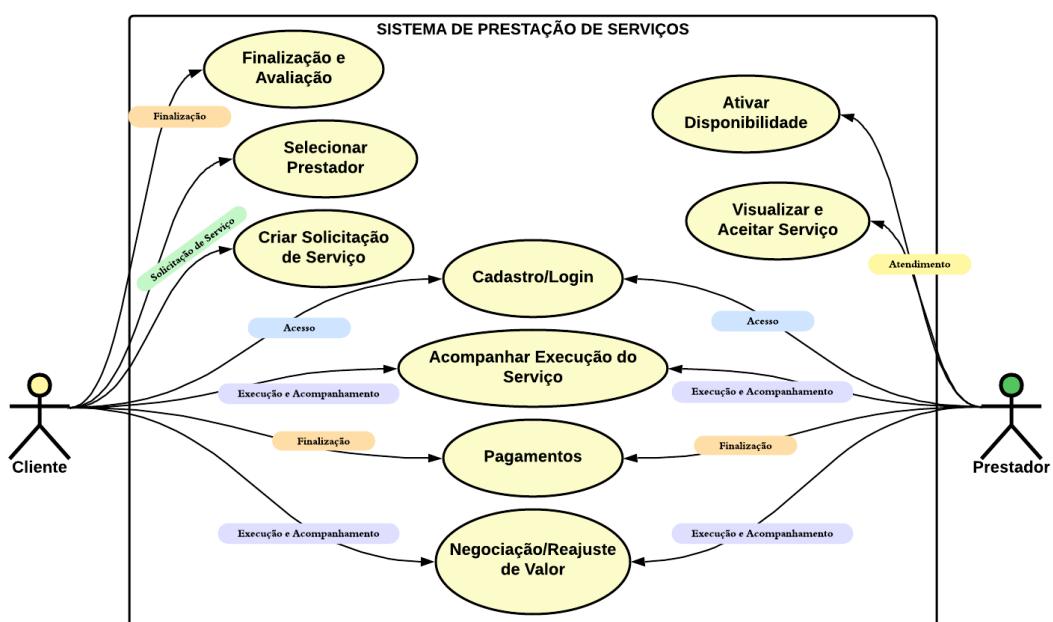


### 4.3 Requisitos não funcionais

Código	tipo requisito	Requisito	descrição
NF01	Segurança de Dados	Segurança de Dados	A confidencialidade dos dados deve ser garantida por meio de criptografia de ponta a ponta (TLS 1.3). Dados sensíveis como senhas devem ser armazenados utilizando algoritmos de hash seguros (bcrypt/Argon2). Todo o sistema deve estar em conformidade com a LGPD.
NF02	Segurança de Dados	Autenticação e Autorização	O sistema deve utilizar autenticação baseada em tokens JWT e aplicar controle de acesso baseado em papéis (clientes, prestadores, administradores). Cada ator terá acesso apenas às funcionalidades autorizadas ao seu perfil.
NF03	Usabilidade	Responsividade	A aplicação web deve ser responsiva, permitindo acesso em dispositivos desktop, tablets e smartphones, mantendo boa legibilidade e usabilidade em diferentes resoluções de tela.
NF04	Usabilidade	Usabilidade	A interface deve ser intuitiva e simples, permitindo que um usuário leigo consiga solicitar um serviço em até três passos principais. Devem ser aplicados testes de usabilidade para validar a experiência.
NF05	Desempenho	Desempenho	Operações críticas como login, criação de solicitação, chat e confirmação de pagamento devem ser processadas em até 3 segundos. O sistema deve suportar pelo menos 1.000 usuários simultâneos no ambiente MVP.
NF06	Escalabilidade	Desempenho/ Escalabilidade	O sistema deve ser implantável em nuvem (AWS, Azure ou GCP), com suporte a balanceamento de carga, containers (Docker/Kubernetes) e replicação de banco de dados, permitindo expansão sem reescrita da arquitetura.
NF07	Confiabilidade	Disponibilidade	O sistema deve estar disponível 24h por dia, 7 dias por semana, com uptime mínimo de 99% mensal. Períodos de manutenção programada devem ser comunicados previamente aos usuários.

NF08	Compatibilidade	Compatibilidade	O sistema deve ser compatível com os principais navegadores modernos (Google Chrome, Mozilla Firefox, Microsoft Edge e Safari), em suas versões atualizadas.
NF09	Segurança/ Manutenibilidade	Logs e Auditoria	O sistema deve registrar em logs as ações críticas (login, criação de solicitação, pagamentos, cancelamentos e reajustes). Cada registro deve conter usuário, data, hora, IP e ação executada, com retenção mínima de 12 meses.
NF10	Segurança/ Confiabilidade	Backup e Recuperação	O sistema deve realizar backups automáticos do banco de dados diariamente. Os arquivos devem ser armazenados em ambiente seguro na nuvem, com retenção de no mínimo 30 dias, e recuperação em até 24h em caso de falhas críticas.

## Modelagem de caso de uso



## 5 TESTES DE QUALIDADE

<b>Título:</b>	Fluxo completo de execução do serviço (aceitar → rota → chegada → execução → finalização → liberação de pagamento)
<b>Identificação:</b>	<b>TST001</b>
<b>Objetivo:</b>	Validar o fluxo completo do serviço entre prestador e cliente, incluindo atualização de estados, registro de GPS e liberação do pagamento via carteira/escrow.
<b>Pré-condições:</b>	Cliente (cliente01) e prestador (prestashop01) cadastrados e validados. Serviço S3001 criado pelo cliente com valor R\$100,00; pagamento pré-autorizado (escrow). Ambiente: homologação.
<b>Passos de Execução:</b>	<ol style="list-style-type: none"> <li>Realizar login como prestador (prestashop01).</li> <li>No mapa/lista, localizar a solicitação S3001 e clicar Aceitar.</li> <li>Fazer login como cliente (cliente01) e Confirmar o prestador.</li> <li>Como prestador, clicar Iniciar Rota (ou iniciar navegação) — enviar posições via WebSocket a cada 3–5s.</li> <li>Ao chegar, prestador clica Cheguei.</li> <li>Prestador clica Iniciar Execução.</li> <li>Após término, prestador clica Finalizar.</li> <li>Cliente recebe notificação e clica Confirmar Finalização.</li> </ol>
<b>Dados de Entrada:</b>	serviceId = S3001 valor = R\$100,00 credenciais de cliente01 / prestador01
<b>Resultado Esperado:</b>	Estados transitam: Aceito → Confirmado → EmRota → NoLocal → EmExecucao → Finalizado → Concluído. GPS/routeSession gravados no banco (gpsLogs). Transação gera liberação do valor: R\$90 para prestador, R\$10 de comissão. Registros de log (statusHistory, transaction) criados.
<b>Critérios de Sucesso:</b>	Cliente e prestador visualizaram os status corretamente; pagamento liberado e registrado; evidências (logs/GPS) salvas.
<b>Critérios de Falha:</b>	Qualquer transição de estado não realizada; ausência de logs/GPS; valor não liberado; erro 5xx no endpoint.

<b>Título:</b>	Cancelamento do prestador por perda de conectividade e aplicação de bloqueio progressivo
<b>Identificação:</b>	<b>TST002</b>
<b>Objetivo:</b>	Validar que cancelamentos do prestador por “Perda de conectividade / transporte” aplicam bloqueio temporário progressivo (30min → 1h → 1h30, etc.) conforme número de ocorrências.
<b>Pré-condições:</b>	Prestador prestador03 cadastrado e apto a receber solicitações. Três serviços distintos (S3021, S3022, S3023) ocorrerão com cancelamento por conectividade em sequência dentro da janela de contagem definida (ex.: 30 dias).
<b>Passos de Execução:</b>	<ol style="list-style-type: none"> <li>1. Para cada serviço (repetir passos 1–3 três vezes, com intervalo curto entre ocorrências):</li> <li>2. Logar como prestador03.</li> <li>3. Acessar serviço (ex.: S3021) e clicar Cancelar.</li> <li>4. Selecionar motivo “Perda de conectividade” e confirmar, anexando justificativa curta se solicitado.</li> <li>5. Verificar registro de bloqueio aplicado ao prestador após cada ocorrência.</li> </ol>
<b>Dados de Entrada:</b>	serviceIds = S3021, S3022, S3023 motivo = “Perda de conectividade”
<b>Resultado Esperado:</b>	<p>1<sup>a</sup> ocorrência → bloqueio de 30 minutos registrado (provider_blocks).</p> <p>2<sup>a</sup> ocorrência → bloqueio de 1 hora registrado.</p> <p>3<sup>a</sup> ocorrência → bloqueio de 1 hora e 30 minutos registrado.</p> <p>Histórico de ocorrências (provider cancels) atualizado.</p>
<b>Critérios de Sucesso:</b>	Duração do bloqueio aplicada conforme tabela progressiva; sistema registra contagem e timestamps; prestador fica indisponível durante o bloqueio.
<b>Critérios de Falha:</b>	Duração incorreta aplicada; contagem de ocorrências não atualizada; prestador continua disponível antes do fim do bloqueio.

<b>Título:</b>	Cancelamento pelo cliente com prestador em rota (reembolso integral + bloqueio do cliente por 2 horas)
<b>Identificação:</b>	<b>TST003</b>
<b>Objetivo:</b>	Verificar se, quando o cliente cancela após o prestador estar em rota, o sistema processa reembolso integral e aplica bloqueio de 2 horas para novas solicitações do cliente.
<b>Pré-condições:</b>	<p>Serviço S3010 em estado EmRota (prestashop iniciou deslocamento).</p> <p>Pagamento em escrow (E4010).</p> <p>Cliente cliente02 logado.</p>
<b>Passos de Execução:</b>	<ol style="list-style-type: none"> <li>1. Logar como cliente02.</li> <li>2. Acessar detalhes do serviço S3010.</li> <li>3. Clicar Cancelar Serviço.</li> <li>4. Selecionar motivo (ex.: “Mudança de planos”) e confirmar.</li> <li>5. Verificar notificações e tentar abrir nova solicitação (após cancelamento).</li> </ol>
<b>Dados de Entrada:</b>	<p>serviceId = S3010</p> <p>motivo = “Mudança de planos”</p>
<b>Resultado Esperado:</b>	<p>Serviço marcado como Cancelado pelo cliente.</p> <p>Reembolso integral processado (registro de refund vinculado ao escrow).</p> <p>Registro de bloqueio do cliente (user_blocks ou blocked_until) com expiração em now + 2h.</p> <p>Notificações enviadas ao prestador e cliente.</p>
<b>Critérios de Sucesso:</b>	Reembolso registrado; cliente impedido de criar nova solicitação dentro do período de bloqueio; logs de cancelamento gerados.
<b>Critérios de Falha:</b>	Reembolso não processado; bloqueio não aplicado ou expiração incorreta; ausência de notificações; inconsistência em service status.

## 6 TECNOLOGIAS UTILIZADAS

### 6.1 Tecnologias do projeto

### 6.2 Visão geral

O Bicoo Maps foi implementado com um conjunto tecnológico pensado para atender requisitos de segurança, disponibilidade, performance e tempo real, priorizando ferramentas consolidadas e de baixo custo operacional para a fase inicial. A arquitetura segue o padrão front-end (SPA) → API REST (back-end) → banco relacional, com serviços externos para mapas, notificações e armazenamento de mídia.

### 6.3 Back-end

#### Java 21 + Spring Boot

- **O que é:** plataforma Java moderna (JDK 21) com Spring Boot para criação rápida de APIs REST.
- **Por que:** robustez, maturidade do ecossistema Spring (Security, Data, Test), facilidade para testes e requisitos transacionais (financeiro / escrow).
- **Impacto nos requisitos:** suporta autenticação segura, transações ACID e testes automatizados (NF01, NF08, NF21).
- **Nota prática:** empacotar como JAR; usar spring.profiles para separar dev/homolog/prod; versão da JVM alinhada ao provedor.

#### Spring Security + JWT / OAuth2

- **O que é:** módulo de segurança do Spring para autenticação/autorização; tokens JWT para sessões sem estado.
- **Por que:** permite RBAC por papéis (prestashop/cliente/admin) e integração com login social (Google).
- **Impacto:** atende controle de acesso e proteção de endpoints sensíveis (RNF segurança).
- **Configuração:** chaves RSA para assinatura de JWT; refresh token para renovação.

#### Spring Data JPA (Hibernate) + MySQL 8.0.37

- **O que é:** JPA/Hibernate para ORM; MySQL como BD relacional.

- **Por que:** consistência transacional e facilidade de modelagem (serviços, transações, carteira).
- **Impacto:** garante integridade de dados em operações financeiras (escrow) e rastreabilidade (logs).
- **Nota prática:** backups diários; índices em colunas usadas em buscas por localização e histórico.

### Lombok, Maven

- **O que é / por que:** Lombok reduz boilerplate; Maven gerencia dependências e build.
- **Nota:** manter pom.xml limpo e perfis de build para CI.

## 6.4 Banco de dados

### MySQL 8.0.37

- **O que é:** SGBD relacional maduro.
- **Por que:** ACID necessário para transações de pagamento; fácil hospedagem em VPS.
- **Impacto:** suporte a transações e integridade referencial (requisito financeiro).
- **Nota prática:** executar export diário (mysqldump) + snapshots; retenção mínima 30 dias.

## 6.5 Front-end

### Vue 3 + Vite

- **O que é:** framework (Vue 3) com bundler rápido (Vite) para SPA.
- **Por que:** produtividade, reatividade e build muito rápidos (bom para entregas ágeis).
- **Impacto:** permite interfaces reativas (mapa em tempo real, atualização de status) atendendo RNF de usabilidade/performance.
- **Configuração:** dividir em módulos (cliente / prestador); lazy loading para rotas pesadas (mapa).

## Tailwind CSS + PrimeVue

- **O que é:** Tailwind (utilitário CSS) para estilo; PrimeVue para componentes prontos.
- **Por que:** rapidez de implementação e consistência visual.
- **Impacto:** facilita conformidade com padrões de UI/UX e acessibilidade (RNF17, RNF07).

## Axios

- **O que é:** cliente HTTP para consumir API REST.
- **Uso:** chamadas ao back-end, handling de tokens JWT e erros.

## 6.6 Tempo real, chat e notificações

### WebSocket / Spring WebSocket (para chat/rota)

- **O que é:** canal bidirecional para transmitir posição e mensagens em tempo real.
- **Por que:** latência baixa para atualização de rota e chat instantâneo (requisito RFS09).
- **Configuração prática:** mensagens compactas (lat, lon, ts); heartbeat e reconnection.

### Firebase Cloud Messaging (FCM)

- **O que é:** serviço para envio de push notifications.
- **Por que:** entrega confiável de notificações em background (alertas, aceitação, chegada).
- **Impacto:** melhora UX com avisos imediatos (RNF20).
- **Nota:** usar FCM para push; WebSocket para canal ativo de interação.

## 6.7 Mapas e geolocalização

### Google Maps API

- **O que é:** APIs para map display, rotas e cálculo de distâncias.
- **Por que:** precisão e funcionalidades prontas (directions, distance matrix).
- **Impacto:** requisito de acompanhamento/rota (RFS09).
- **Configuração:** chave API armazenada em variável segura; monitorar consumo (custo).

## 6.8 Armazenamento de mídia

### Cloudinary

- **O que é:** serviço de hospedagem e otimização de imagens.
- **Por que:** descarrega o servidor principal, CDN e transformação automática (thumbnails).
- **Impacto:** reduz latência do front-end e espaço em disco do VPS.  
**Nota:** usar upload direto do front para Cloudinary (signed/upload preset).

## 6.9 Pagamentos

### Gateway de pagamento (integração com provider)

- **O que é:** serviço externo para processar cartões, PIX e repasses (sandbox em dev).
- **Por que:** conformidade PCI DSS do provedor e facilidade de integração para escrow/simulações.
- **Impacto:** requisito financeiro crítico (RFP08), cobrança de taxas e liberação de valores.
- **Configuração:** trabalhar sempre em modo sandbox para testes; armazenar apenas identificadores, não dados de cartão.

## 7.0 INFRAESTRUTURA E IMPLANTAÇÃO

### RedHosting (VPS) + Ubuntu 22.04 + Docker

- **O que é:** provedor de VPS (RedHosting); SO Ubuntu; containerização com Docker.
- **Por que:** custo x benefício, portabilidade entre ambientes e facilidade de deploy.
- **Impacto:** atende disponibilidade e facilidade de gestão (RNF05, NF14).
- **Configuração sugestiva:** VPS 4 vCPU / 8 GB RAM / 100 GB SSD; usar docker-compose para back/front/db; snapshots diários.

### PM2 + Grafana / Loki (monitoramento e logs)

- **O que é:** PM2 para processos node/monitor; Grafana/Loki para métricas e logs.
- **Por que:** observabilidade para detectar incidentes e monitorar SLAs.
- **Nota:** centralizar logs e configurar alertas básicos.

### GitHub Actions (CI/CD)

- **O que é:** automação de build/test/deploy.
- **Por que:** garante qualidade e entrega contínua (RNF22).
- **Configuração:** pipelines para testes unitários, build de imagens Docker e deploy automatizado para VPS/homolog.

## 7.1 Segurança e conformidade (tecnologias aplicadas)

TLS 1.3 (HTTPS) — proteger tráfego; certificado via Let's Encrypt.

bcrypt — hashing de senhas; recomendado pelo Spring Security.

AES / criptografia em repouso — para dados sensíveis se necessário (ex.: tokens).

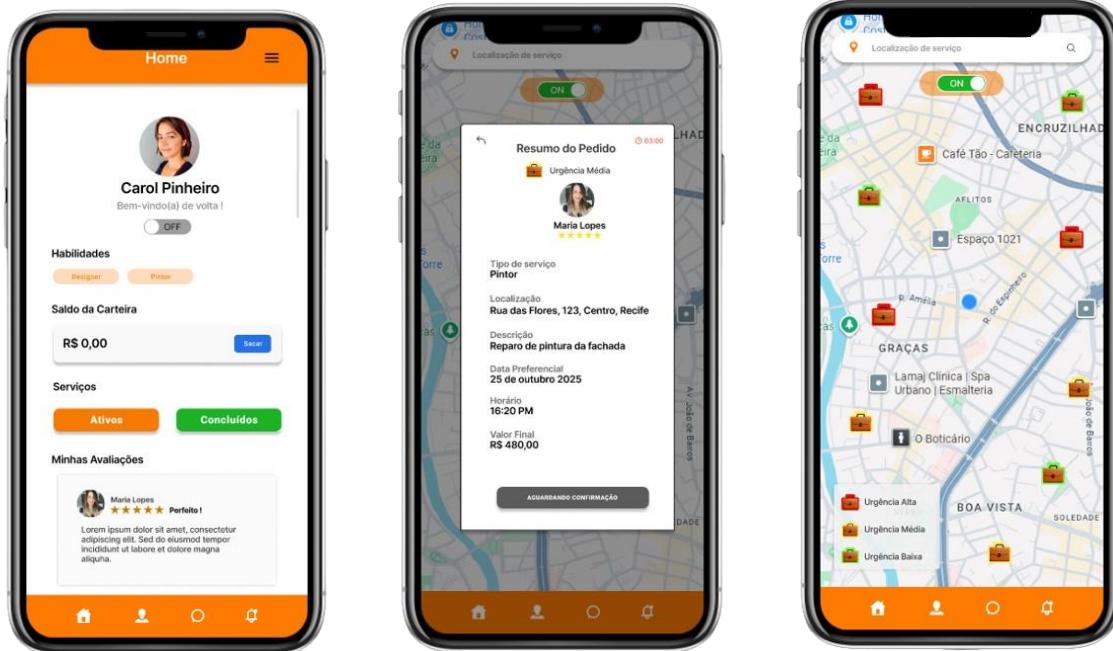
RBAC via Spring Security — papéis e permissões.

Logs de auditoria — registrar eventos críticos (cancelamentos, transações).

LGPD — políticas, consentimento e meios de exclusão/exportação de dados.

## 8 ANEXOS

### 8.1 Telas do prestador



### 8.2 Telas do solicitante (cliente)

