

---

Faculdade de Tecnologia de Americana “Ministro Ralph Biasi”  
Curso Superior de Tecnologia em Análise e Desenvolvimento de  
Sistemas

Mariana Afonso Abade Couceiro

**Projeto de um Aplicativo de Agenda para Profissionais  
Autônomos: Agenda App**

Mariana Afonso Abade Couceiro

## **Projeto de um Aplicativo de Agenda para Profissionais Autônomos: Agenda App**

Trabalho de Conclusão de Curso desenvolvido em cumprimento à exigência curricular do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, sob a orientação do Prof. Dr. Kleber de Oliveira Andrade

Área de concentração: Engenharia de Software

Americana, SP  
2023

**FICHA CATALOGRÁFICA – Biblioteca Fatec Americana Ministro Ralph Biasi-  
CEETEPS Dados Internacionais de Catalogação-na-fonte**

COUCEIRO, Mariana Afonso Abade

Projeto de um aplicativo de agenda para profissionais autônomos:  
Agenda App. / Mariana Afonso Abade Couceiro –Americana, 2023.

114f.

Monografia (Curso Superior de Tecnologia em Análise e  
Desenvolvimento de Sistemas) - - Faculdade de Tecnologia de Americana  
Ministro Ralph Biasi – Centro Estadual de Educação Tecnológica Paula Souza

Orientador: Prof. Dr. Kleber de Oliveira Andrade

1. Dispositivos móveis - aplicativos 2. Engenharia de software  
3. Modelagem de sistemas. I. COUCEIRO, Mariana Afonso Abade II. ANDRADE,  
Kleber de Oliveira III. Centro Estadual de Educação  
Tecnológica Paula Souza – Faculdade de Tecnologia de Americana Ministro Ralph  
Biasi

CDU: 681519  
681.3.05  
681.5.01

Elaborada pelo autor por meio de sistema automático gerador de ficha  
catalográfica da Fatec de Americana Ministro Ralph Biasi.

Mariana Afonso Abade Couceiro

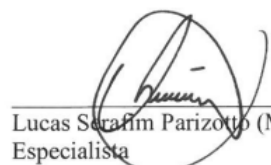
**PROJETO DE UM APLICATIVO DE AGENDA PARA  
PROFISSIONAIS AUTÔNOMOS: AGENDA APP**

Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas pelo Centro Paula Souza – FATEC Faculdade de Tecnologia de Americana – Ralph Biasi.  
Área de concentração: Engenharia de Software

Americana, 27 de novembro de 2023

**Banca Examinadora:**

  
Kleber de Oliveira Andrade (Presidente)  
Doutor  
Faculdade de Tecnologia de Americana

  
Lucas Serafim Parizotto (Membro)  
Especialista  
Faculdade de Tecnologia de Americana

  
Maxwell Vitorino da Silva (Membro)  
Mestre  
Faculdade de Tecnologia de Americana

## **AGRADECIMENTOS**

Ao meu orientador, Professor Dr. Kleber de Oliveira Andrade, pela confiança e ensinamentos, ao qual sou muito grata pela sua atenção durante a realização desse trabalho.

Aos professores e funcionários da Fatec de Americana por toda ajuda na conclusão do curso.

Aos meus colegas de turma pela ótima convivência durante todo o período do curso.

Aos colegas de estágio do Laboratório de Processamento de Dados: Willian, Anderson, Rodrigo, Michael, Pedro, Otávio, Luciana, Laís, Vitor e Lucas, pela amizade e apoio.

Ao meu marido Fernando pelo incentivo, companheirismo e paciência.

Aos meus amigos e irmãos Jéssica e Eduardo, pelos momentos de descontração.

E aos meus familiares pelo apoio e incentivo.

## RESUMO

O trabalho autônomo permite que o profissional tenha flexibilidade e independência em suas carreiras, e essa forma de trabalho está cada vez mais comum em muitos setores. Esses profissionais oferecem seus serviços e produtos diretamente aos clientes e são responsáveis por gerenciar seu próprio negócio. Para auxiliar esses profissionais, este trabalho tem como objetivo desenvolver o protótipo de um aplicativo móvel, o Agenda App, uma ferramenta de agendamento de serviços e venda de produtos. Para o desenvolvimento do aplicativo, várias etapas da engenharia de software foram realizadas, como a pesquisa sobre os aplicativos de agendamentos disponíveis nas lojas de aplicativos, o levantamento de requisitos, o levantamento dos recursos e ferramentas, a modelagem do sistema, os diagramas de arquitetura e a prototipagem. A proposta de desenvolvimento do projeto possui na sua documentação os seguintes componentes principais: detalhes do projeto; requisitos funcionais e não funcionais; diagramas e documentação de casos de uso, diagrama lógico do banco de dados; arquitetura dos códigos de todas as telas; diagrama de fluxo de tela e o protótipo de todas as telas do aplicativo. Os principais recursos e funcionalidades criados para o aplicativo Agenda App foram: o cadastro de clientes, serviços e produtos; agendamento e gestão de compromissos; venda de produtos e gerenciamento financeiro. Com o plano detalhado do aplicativo é possível realizar o desenvolvimento, os testes e a publicação do aplicativo nas lojas App Store e Google Play Store.

Palavras chave: Profissionais autônomos; Aplicativo de agendamento; Agenda App

## **ABSTRACT**

### Project of an Agenda Application for Self-Employed Professionals: Agenda App

Self-employment allows professionals to have flexibility and independence in their careers, and this form of work is increasingly common in many sectors. These professionals offer their services and products directly to customers and are responsible for managing their own business. To assist these professionals, this work aims to develop the prototype of a mobile application, the Agenda App, a tool for scheduling services and selling products. For the development of the application, several software engineering steps were carried out, such as research into scheduling applications available in application stores, requirements gathering, resources and tools survey, system modeling, architecture diagrams and prototyping. The project development proposal has the following main components in its documentation: project details; functional and non-functional requirements; use case diagrams and documentation, database logic diagram; code architecture for all screens; screen flow diagram and the prototype of all application screens. The main features and functionalities created for the Agenda App application were: registration of customers, services and products; scheduling and management of appointments; product sales and financial management. With the detailed application plan, it is possible to develop, test and publish the application in the App Store and Google Play Store.

**Keywords:** Self-employed professionals; Scheduling Application; Agenda App

## LISTA DE FIGURAS

Figura 1 - Diagrama de caso de uso manter conta e <i>login</i> .....	24
Figura 2 - Diagrama de caso de uso do CRUD de clientes.....	25
Figura 3 - Diagrama de caso de uso do CRUD de produtos.....	26
Figura 4 - Diagrama de caso de uso do CRUD de serviços.....	26
Figura 5 - Diagrama de caso de uso de agendamentos.....	27
Figura 6 – Padrão MVC.....	48
Figura 7 – Padrão MVCS.....	49
Figura 8 - MVCS para Usuário.....	51
Figura 9 - MVCS para Cliente.....	52
Figura 10 - MVCS para Serviço.....	53
Figura 11 - MVCS para Produto.....	55
Figura 12 - MVCS para Agenda.....	57
Figura 13 - MVCS para Comanda de Serviço.....	59
Figura 14 - MVCS para Comanda de Venda de Produto.....	61
Figura 15 - MVCS para Confirmação de Pagamento.....	62
Figura 16 - MVCS para os Relatórios.....	64
Figura 17 - MVCS para a Tela Inicial ( <i>Home</i> ).....	65
Figura 18 - Modelo lógico do banco de dados.....	69
Figura 19 - Diagrama de fluxo de tela (mapa das telas).....	71
Figura 20 - Captura da <i>Splash Screen</i> .....	72
Figura 21 - Captura das telas de <i>Login</i> (a) e de Criação de Conta (b).....	73
Figura 22 - Captura da tela Esqueceu sua Senha (a) e de verificação (b).....	74
Figura 23 - Captura da tela <i>Home</i> (a) e continuação da tela <i>Home</i> (b).....	75
Figura 24 - Captura das telas Meu Perfil modo claro (a) e Meu Perfil modo escuro (b).....	77
Figura 25 - Captura da janela Excluir Conta de Usuário.....	78
Figura 26 - Captura das telas Redefinir Senha (a) e Editar Perfil (b).....	79
Figura 27 - Captura da tela Editar Perfil: ao clicar no botão “Alterar foto” (a) e ao clicar no botão “Carregar Imagem” (b).....	80
Figura 28 - Captura da tela Meus Clientes: Todos (a) e Novos (b).....	81
Figura 29 - Captura da tela Cadastrar Cliente.....	83



Figura 30 - Captura da tela Editar Cliente (a) e janela de mensagem Excluir Cliente (b).....	84
Figura 31 - Captura da tela Meus Serviços: Ativos (a) e Desativados (b). ....	86
Figura 32 - Captura da tela Cadastrar Serviço. ....	87
Figura 33 - Captura da tela Editar Serviço (a) e janela de mensagem Excluir Serviço (b).....	88
Figura 34 - Captura da tela Meus Produtos.....	89
Figura 35 - Captura da tela Cadastrar Produto. ....	91
Figura 36 - Captura da tela Editar Produto (a) e mensagem de Excluir Produto (b). ....	92
Figura 37 - Captura da tela Agenda. ....	93
Figura 38 - Captura da tela Agendamento (a), janela selecionar data do agendamento (b) e janela selecionar horário do agendamento (c). ....	95
Figura 39 - Captura da tela Calendário (a) e continuação da tela Calendário (b). ....	97
Figura 40 - Captura da tela Agenda Fechada. ....	98
Figura 41 - Captura das telas Horários Livres (a) e Agendamento (b). ....	99
Figura 42 - Captura da tela Editar Agendamento (a) e mensagem de Excluir agendamento (b). ....	100
Figura 43 - Captura da tela Enviar Mensagem.....	101
Figura 44 - Captura da tela Comanda de Serviço (a) e Comanda de Venda (b).....	102
Figura 45 - Captura da tela Confirmação de pagamento (a) e janela com mensagem de fechamento da comanda (b).....	105
Figura 46 - Captura da janela com confirmação de comanda em aberto.....	106
Figura 47 - Captura da tela Relatório Financeiro: aba Receita (a) e aba Caixa (b). ....	107
Figura 48 - Captura da tela Relatório Financeiro: aba Cobrança (a) e mensagem de confirmação de encerramento da cobrança (b).....	108
Figura 49 - Captura da tela Relatório Clientes. ....	109

## LISTA DE TABELAS

Tabela 1. Comparativo de funcionalidades entre os aplicativos analisados e o aplicativo desenvolvido neste trabalho.....	18
Tabela 2 - Requisitos funcionais do projeto. ....	19
Tabela 3 - Requisitos não funcionais do projeto.....	20
Tabela 4 - Documentação Caso de Uso “Criar Conta”.....	28
Tabela 5 - Documentação Caso de Uso “Editar Conta”.....	29
Tabela 6 - Documentação Caso de Uso “Excluir Conta”.....	30
Tabela 7 - Documentação Caso de Uso “Fazer Login”.....	31
Tabela 8 - Documentação Caso de Uso “Recuperar Senha”.....	31
Tabela 9 - Documentação Casos de Uso “Criar / Editar / Excluir Clientes”.....	32
Tabela 10 - Documentação Casos de Uso “Criar / Editar / Excluir Serviços”.....	34
Tabela 11 - Documentação Casos de Uso “Criar / Editar / Excluir Produtos”.....	35
Tabela 12 - Documentação Caso de Uso “Criar Agendamento”.....	36
Tabela 13 - Documentação Casos de Uso “Editar / Excluir Agendamentos”.....	38
Tabela 14 - Documentação Caso de Uso “Enviar Mensagem”.....	39
Tabela 15 - Documentação Caso de Uso “Gerar Comanda de Serviço”.....	40
Tabela 16 - Documentação Caso de Uso “Vender Produto”.....	42
Tabela 17 - Documentação Caso de Uso “Gerenciar Comandas em Aberto”.....	44
Tabela 18 - Documentação Caso de Uso “Visualizar Relatórios”.....	45
Tabela 19 - Documentação Caso de Uso “Visualizar Horários Livres”.....	46

## LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i> (Interface de Programação de Aplicação)
BaaS	<i>Backend As A Service</i> (Back-end como Serviço)
BLOB	<i>Binary Large Objects</i> (Objetos Binários Grandes)
CRUD	<i>Create Read Update and Delete</i> (Criar Ler Atualizar e Deletar)
IBGE	Instituto Brasileiro de Geografia e Estatística
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
MVC	<i>Model-View-Controller</i> (Modelo-Visão-Controlle)
MVCS	<i>Model-View-Controller-Service</i> (Modelo-Visão-Controlle-Serviço)
NoSQL	<i>Not Only SQL</i> (Não Somente SQL)
PIX	Pagamento Instantâneo Brasileiro
PMV	Produto Mínimo Viável
PNADE	Pesquisa Nacional por Amostra de Domicílios Contínua
SEBRAE	Serviço Brasileiro de Apoio às Micro e Pequenas Empresas
SDK	<i>Software Development Kit</i> (Kit de Desenvolvimento de Software)
SQL	<i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
UML	<i>Unified Modeling Language</i> (Linguagem de Modelagem Unificada)

## SUMÁRIO

1 INTRODUÇÃO .....	13
2 PROJETO DO SISTEMA .....	15
2.1 SOFTWARES SIMILARES .....	16
2.2 LEVANTAMENTO DE REQUISITOS .....	18
2.2.1 Requisitos Funcionais .....	18
2.2.2 Requisitos Não Funcionais .....	20
2.2 RECURSOS E FERRAMENTAS .....	20
3 MODELAGEM .....	23
3.1 DIAGRAMAS DE CASOS DE USO .....	23
3.2 DOCUMENTAÇÃO DOS CASOS DE USO .....	28
3.2 DIAGRAMA DE ARQUITETURA .....	47
3.3 BANCO DE DADOS .....	66
3.3.1 Modelagem Lógica do Banco de Dados .....	66
4 PROTÓTIPO .....	70
5 CONSIDERAÇÕES FINAIS .....	110
REFERÊNCIAS BIBLIOGRÁFICAS .....	112

## 1 INTRODUÇÃO

Segundo a Pesquisa Nacional por Amostra de Domicílios Contínua (PNADC), realizada pelo Instituto Brasileiro de Geografia e Estatística (IBGE) no segundo semestre de 2023, 25,5% da população ocupada, ou, em números absolutos, aproximadamente 25,7 milhões de pessoas, trabalham como profissional autônomo no Brasil (IBGE, 2023).

Mas quem se enquadra nessa categoria? De acordo com a definição do SEBRAE (2022), profissional autônomo é todo trabalhador que possui habilidades técnicas, manuais ou intelectuais e que decide executar sua atividade profissional por conta própria, sem vínculo empregatício, seja em profissões regulamentadas, como o caso de advogados, dentistas, médicos, engenheiros e psicólogos, como também em profissões não regulamentadas, como pintores, pedreiros, barbeiros, manicures, marceneiros, entre outros.

Um dos maiores desafios desse tipo de profissional é justamente ter que executar sozinho, funções que em outros ambientes de trabalho seriam divididas entre setores, às vezes com várias pessoas cada. Nesse caso, eles precisam administrar sozinhos várias áreas do seu negócio como: controlar despesas e receitas, criar estratégias de *marketing*, atender e fidelizar os clientes, alimentar e organizar agenda, entre outras tantas tarefas inerentes à qualquer negócio.

E é para auxiliar esses profissionais, muitas vezes sobrecarregados, que a tecnologia da informação oferece soluções, como: planilhas, e-mail, aplicativos de calendário, software de agendamento online, sistemas de gerenciamento de clientes, redes sociais e aplicativos móveis.

E o objeto desse trabalho é o desenvolvimento de um protótipo de aplicativo móvel, ou seja, de um software desenvolvido para ser executado em dispositivos móveis como *smartphones* e *tablets*, que atenda a necessidade de agendamento de compromissos e da venda de produtos relacionados à atividade. No caso exemplificado um profissional da área de beleza.

Os objetivos específicos são:

- Realizar uma pesquisa sobre as tecnologias que poderão ser implantadas no desenvolvimento deste aplicativo;
- Desenvolver um protótipo do aplicativo móvel na plataforma FlutterFlow;
- Aplicar os conhecimentos adquiridos na disciplina de Engenharia de Software para projetar o aplicativo;
- Aplicar os conhecimentos adquiridos na disciplina de Interação Humano Computador para projetar as interfaces do aplicativo.

Com esse projeto, espera-se que a documentação aqui apresentada seja desenvolvida e publicada nas principais lojas de aplicativos, como a Google Play Store e a Apple Store.

O trabalho está organizado em cinco capítulos. O primeiro capítulo, este, faz uma introdução ao tema do trabalho, explicando o objetivo da pesquisa. O segundo capítulo apresenta o projeto do aplicativo, uma comparação entre aplicativos de agendamentos, levantamento de requisitos e os recursos e ferramentas. O terceiro capítulo descreve a modelagem do sistema com os diagramas e documentação dos casos de uso, os diagramas de arquitetura e o modelo lógico do banco de dados. O quarto capítulo apresenta e descreve o protótipo das telas do aplicativo. O quinto e último capítulo faz uma consideração a respeito do trabalho e uma relação de trabalhos futuros.

## 2 PROJETO DO SISTEMA

O sistema que poderá ser desenvolvido segundo este projeto recebe o nome de “Agenda App”. O aplicativo “Agenda App” tem como objetivo facilitar o agendamento de serviços e/ou venda de produtos. Inicialmente, o “Agenda App” poderá ser baixado de forma gratuita pelos usuários. E no futuro, quando novas funcionalidades e recursos forem desenvolvidos, o aplicativo poderá ser comercializado nas lojas de aplicativos.

O público alvo são os profissionais autônomos que buscam gerenciar seu negócio utilizando um dispositivo móvel. O profissional, ou o usuário, será o único gestor do aplicativo, apenas ele tem acesso à agenda e pode realizar os agendamentos e as vendas de produtos. Dessa forma, o profissional possui total controle de seus compromissos e vendas. Cabe ao usuário manter o cadastro de clientes, serviços e produtos. Os clientes e possíveis colaboradores não terão acesso ao aplicativo para realizarem seus agendamentos ou suas compras. A comunicação entre o profissional usuário do “Agenda App” e seus clientes deverá ser realizada por meio de outras tecnologias como e-mail, WhatsApp, ligação telefônica ou pessoalmente.

Para exemplificar, um cliente pode solicitar um agendamento com o profissional via troca de mensagens pelo aplicativo WhatsApp. O profissional sugere algumas datas e horários e o cliente escolhe uma das opções apresentadas. Então o profissional, por meio do aplicativo “Agenda App” realiza o agendamento. O profissional pode enviar uma mensagem de confirmação do agendamento via WhatsApp na véspera do compromisso. E o cliente pode confirmar também utilizando o aplicativo WhatsApp. Durante o atendimento, o profissional pode utilizar o “Agenda App” para organizar e registrar seus serviços e vendas.

Um diferencial do “Agenda App” está relacionado com a possibilidade do usuário realizar venda de produtos juntamente com a prestação de serviço. Por meio da comanda, o usuário pode controlar as negociações realizadas com seus clientes, sejam elas prestações de serviços, vendas ou prestação de serviço(s) com venda(s). O sistema deve ser capaz de armazenar informações inseridas pelo profissional em um banco de dados na nuvem e localmente, no dispositivo. O sistema também foi

pensado para que o profissional possa compartilhar com seus clientes os horários livres para agendamento. Essa comunicação é feita através do sistema, utilizando um *link* com o aplicativo WhatsApp.

As telas do protótipo do “Agenda App” foram criadas pensando em um profissional do ramo da beleza, como um exemplo de usuário em potencial do aplicativo. Normalmente, profissionais dessa área possuem uma lista de clientes, um portfólio de serviços prestados, uma lista de produtos que podem ser comercializados, agendam seus compromissos e controlam suas finanças. Mas a ideia é que o aplicativo possa ser amplamente utilizado por outros profissionais autônomos inseridos em várias áreas do mercado de trabalho.

## 2.1 SOFTWARES SIMILARES

Nas lojas on-line, Google Play e App Store, é possível encontrar uma lista extensa de aplicativos de agendamento para profissionais. Com o objetivo de entender melhor o que sistema projetado neste trabalho pode oferecer, foram selecionados quatro aplicativos, entre os mais populares e mais bem avaliados pelos usuários nessas lojas, para comparação de suas funcionalidades. Os aplicativos selecionados foram:

- **Agenda Boa:** Um aplicativo de gestão para autônomos, que tem por objetivo profissionalizar pequenos negócios e passar maior credibilidade para os clientes. Por meio do aplicativo, profissionais autônomos de todos os seguimentos podem organizar seus pedidos, compromissos, orçamentos, recibos, documentos, ordens de serviços e controlar as próprias finanças. Essa ferramenta é indicada, prioritariamente, para negócios do ramo da construção, climatização, manicure, assistência técnica, confeitaria, manutenção, pintura, eventos, veículos e elétrica (PESSANHA, 2022). Para cada fase do negócio o aplicativo Agenda Boa oferece um plano diferente, e na modalidade gratuita, estão disponíveis funcionalidades que auxiliam o profissional no suporte inicial para organizar o fluxo diário, como cadastro de clientes, pedidos, agenda e fluxo de entradas e saídas (FOLHA DE S.PAULO, 2023). Quanto aos planos, a Agenda Boa possui uma versão gratuita e duas



versões pagas (PRO e TOP). A versão PRO custa R\$ 62,5 por mês e a TOP R\$ 125 por mês.

- **Minha Agenda:** Um aplicativo de controle de agendamentos e gerenciamento de negócio. O aplicativo auxilia no controle financeiro (receitas, despesas e lucros), relatório de clientes, integração com o WhatsApp, lembretes, evoluções e performance. Feito para quem trabalha com atendimento, como esteticista, manicure, barbeiro, *personaltrainer*, dentista e outros. A integração com o WhatsApp permite que o profissional envie lembretes e mensagens customizadas de forma rápida e simples. É um aplicativo pago, que possui uma versão gratuita de teste com validade de 30 dias, e versões pagas nos planos individuais e empresariais (MINHAAGENDA, 2023).
- **SelfWorks:** Um aplicativo móvel voltado para profissionais autônomos, com gestão de agenda, clientes e serviços. No plano gratuito o SelfWorks oferece as funcionalidades para a gestão de serviços e clientes, confirmação com o cliente por WhatsApp, histórico de clientes, acompanhamento de faturamento e notificações. Na versão paga (Plano Premium), por R\$ 14,90 ao mês, além das funcionalidades da versão gratuita, é disponibilizado o serviço de agenda de atendimentos ilimitado, o perfil on-line com agendamento e a emissão de recibos (SELFWORKS, 2023).
- **Tua Agenda:** Um aplicativo de agendamento para profissionais que funciona *online* e *off-line*. No aplicativo é possível cadastrar produtos e serviços, realizar agendamentos, controlar a agenda de funcionários, enviar lembretes para os clientes, disponibilizar a agenda para que os clientes realizem o agendamento, realizar agendamentos recursivos (diários, semanais, quinzenais, etc.) e definir permissões de acesso para cada funcionário. É possível testar a plataforma de forma gratuita durante o período de 30 dias, e de forma ilimitadas por meio de uma assinatura mensal no valor de R\$ 19,90 (TUAAGENDA, 2023).

A Tabela 1 mostra as principais diferenças entre os aplicativos “Agenda Boa”, “Minha Agenda”, “SelfWorks”, “Tua Agenda” e o “Agenda App”, com relação as funcionalidades.

**Tabela 1. Comparativo de funcionalidades entre os aplicativos analisados e o aplicativo desenvolvido neste trabalho.**

Funcionalidade	Agenda Boa	Minha Agenda	SelfWorks	Tua Agenda	Agenda App
Disponibiliza versão gratuita	X	-	X	-	X
Cadastro de clientes	X	X	X	-	X
Cadastro de serviços	X	X	X	X	X
Cadastro de produtos	X	X	-	X	X
Realiza agendamentos de serviços	X	X	X	X	X
Registrar venda de produtos	-	-	-	-	X
Mensagem de lembrete e confirmação para os clientes via WhatsApp	-	X	X	-	X
Permite inserir desconto	X	X	-	-	X
Gera comanda digital	X	X	-	-	X
Gestão financeira	X	X	X	-	X
Customização do perfil	-	X	-	-	X

**Fonte: Elaborado pela autora (2023).**

## 2.2 LEVANTAMENTO DE REQUISITOS

Requisitos são necessidades a serem atendidas ao sistema em questão. Os requisitos de um sistema são as descrições dos serviços que o sistema deve prestar e as restrições a sua operação (SOMMERVILLE, 2018, p.85). O processo de descoberta, análise, documentação e verificação desses serviços e restrições são chamados de engenharia de requisitos (SOMMERVILLE, 2018, p.85). Frequentemente, os requisitos de sistema de *software* são divididos em dois tipos: requisitos funcionais e requisitos não funcionais.

### 2.2.1 Requisitos Funcionais

Os requisitos funcionais são declarações dos serviços que o sistema deve fornecer, isto é, definem a funcionalidade desejada do *software* (SOMMERVILLE,

2018, p.89). A Tabela 2 apresenta os requisitos funcionais deste projeto. Na descrição da tabela o termo usuário se refere ao profissional que vai utilizar o aplicativo.

**Tabela 2 - Requisitos funcionais do projeto.**

<b>Identificação</b>	<b>Requisito Funcional</b>	<b>Prioridade</b>
RF001	O usuário deve criar uma conta com um e-mail válido e uma senha.	Essencial
RF002	O sistema deve realizar o <i>login</i> do usuário utilizando uma conta de e-mail válida.	Essencial
RF003	O usuário poderá criar, editar e deletar um cliente.	Essencial
RF004	O sistema deve salvar e mostrar a lista de clientes cadastrados pelo usuário.	Essencial
RF005	O usuário poderá criar, editar e deletar um serviço.	Essencial
RF006	O sistema deve salvar e mostrar uma lista dos serviços cadastrados pelo usuário.	Essencial
RF007	O usuário poderá criar, editar e deletar um produto.	Essencial
RF008	O sistema deve salvar e mostrar uma lista dos produtos cadastrados pelo usuário.	Essencial
RF009	O usuário poderá criar agendamentos de serviços.	Essencial
RF010	O sistema deve salvar e mostrar uma lista dos agendamentos marcados pelo usuário.	Essencial
RF011	O sistema deve permitir que o usuário visualize os horários vagos na agenda.	Desejável
RF012	O sistema deve gerar uma comanda do serviço prestado, com a possibilidade de acrescentar no mínimo um serviço e no mínimo um produto da lista de produtos para venda.	Essencial
RF013	O sistema deve gerar uma comanda do(s) produto(s) vendido(s).	Essencial
RF014	O sistema deve permitir que o usuário envie mensagens de confirmação para o cliente via aplicativo WhatsApp.	Importante
RF015	O sistema deve fornecer um histórico dos agendamentos concluídos e cancelados.	Importante
RF016	O sistema deve fornecer um relatório com o histórico dos clientes.	Desejável
RF017	O sistema deve emitir um relatório financeiro (receita, despesas e lucro).	Importante
RF018	O sistema deve mostrar uma lista de cobranças.	Importante

**Fonte: Elaborado pela autora (2023).**

### 2.2.2 Requisitos Não Funcionais

Os requisitos não funcionais são aqueles que não possuem relação direta com os serviços específicos fornecidos pelo sistema aos seus usuários (SOMMERVILLE, 2018, p.91). Normalmente especificam ou restringem as características do sistema como um todo, tais como restrições de tempo, restrições no processo de desenvolvimento, e são mais complexos que os requisitos funcionais, mas requer vários funcionais para ser realizado. A Tabela 3 apresenta os requisitos não funcionais deste projeto.

**Tabela 3 - Requisitos não funcionais do projeto.**

Identificação	Requisito não funcional	Categoria	Prioridade
RNF001	Funcionar sem uma conexão com a internet.	Disponibilidade	Importante
RNF002	Funcionar nos sistemas operacionais Android e iOS.	Compatibilidade	Essencial
RNF003	Interface intuitiva e amigável.	Usabilidade	Importante
RNF004	Banco de Dados Firebase.	Desenvolvimento	Importante
RNF005	Desenvolvido em Flutter.	Desenvolvimento	Essencial
RNF006	Rotinas de <i>Backup</i> .	Confiabilidade	Importante
RNF007	Segurança da informação.	Legais	Importante

**Fonte: Elaborado pela autora (2023).**

## 2.2 RECURSOS E FERRAMENTAS

Esta seção contempla as ferramentas de programação e os conceitos necessários para o desenvolvimento do aplicativo “Agenda App”:

- **Android Studio** – É um Ambiente de Desenvolvimento Integrado (IDE) oficial para o desenvolvimento de aplicativos Android, baseado no IntelliJ IDEA<sup>1</sup>, que oferece um ambiente unificado para o desenvolvimento de aplicativos. No

<sup>1</sup>Intellig IDEA é um JAVA IDE da empresa JetBrains. Disponível em:<<https://www.jetbrains.com/idea>>.

qual é possível desenvolver, fazer *debugs*, testes e *interfaces* para *smartphones* e *tablets Android* e dispositivos *Android Wear*, *Android TV* e *Android Auto* (ANDROID, 2023).

- **Visual Studio Code** – É um Ambiente de Desenvolvimento Integrado (IDE), desenvolvido pela Microsoft. É um editor de código aberto disponível para Windows, macOS e Linux. Ele também contém diversas extensões para auxiliar o desenvolvimento de aplicativos *web*, *desktop* ou móveis (MICROSOFT, 2023).
- **Flutter** – É um *framework* (ferramenta) de desenvolvimento de código aberto, criado pelo Google, que permite criar aplicativos compilados nativamente para dispositivos móveis (Android e iOS), web e desktop (Linux, Windows e macOS), a partir de uma única base de código. O Flutter usa a linguagem de programação Dart. Essa ferramenta possui um recurso, chamado “*Hot Reload*” que permite que os desenvolvedores vejam as alterações instantaneamente durante o processo de desenvolvimento. Além disso, o Flutter possui uma grande comunidade de desenvolvedores e um grande número de *widgets* (componentes visuais, “blocos” de construção) pré-construídos, tornando-o uma escolha robusta para criação de aplicativos bonitos e responsivos (FLUTTER, 2023a).
- **Dart** – É uma linguagem de programação criada pelo Google em 2011, de código aberto, orientada a objetos e multiparadigma, podendo ser executada em ambientes nativos e em ambiente web (ou seja, tanto em aplicações móveis e *desktop*, quanto em aplicações *online* que utilizam *frameworks*). Quando usada em conjunto com o Flutter, permite compilar o código fonte antecipadamente para o código nativo (ROSE, 2022).
- **FlutterFlow** – É uma plataforma visual de desenvolvimento de aplicativos sem código, que fornece uma interface intuitiva de arrastar e soltar (*drag-and-drop*), permitindo aos usuários projetar telas, criar animações e definir fluxos de trabalho. Também oferece modelos de interface pré-construídos como botões, formulários e *layouts*. Além de seu editor visual, o FlutterFlow também inclui suporte de integração com diversos serviços e APIs. Criado por dois engenheiros do Google, o FlutterFlow usa código Dart, que é usada pelo Flutter (FLUTTERFLOW, 2023). Neste projeto, o FlutterFlow foi utilizado para criar o protótipo de alta fidelidade do aplicativo de agendamento “Agenda App”.

- **Firestore** – É uma plataforma de desenvolvimento de aplicativos *backend*, desenvolvida pelo Google, como serviço (BaaS) que fornece serviços de *backend* hospedados, como banco de dados em tempo real, armazenamento em nuvem, autenticação, relatórios de falhas, aprendizagem de máquina, configuração remota e hospedagem para arquivos estáticos (FLUTTER, 2023b). A base do Firestore é construída na infraestrutura do Google e pode ser voltada para as principais plataformas móveis (Android, iOS e Web). Entre os principais serviços do Firestore estão o *Realtime Database*, que é um banco de dados NoSQL (*Not Only SQL*) hospedado na nuvem, embora disponível mesmo quando o aplicativo está *offline*, e o *Firestore Authentication*, que oferece suporte à autenticação de *login* de usuário usando senhas, números de telefone, perfis sociais, entre outros (FIREBASE, 2023).

### 3 MODELAGEM

A modelagem de sistemas computacionais é a representação por uma linguagem visual, notação gráfica, baseada nos tipos de diagrama em UML (*Unified Modeling Language*, em português, Linguagem de Modelagem Unificada). Esses diagramas facilitam a compreensão do projeto de forma padronizada. A documentação deste trabalho utilizará a linguagem UML para modelar os casos de uso.

#### 3.1 DIAGRAMAS DE CASOS DE USO

Os diagramas de casos de uso têm por objetivo apresentar uma visão geral das funcionalidades que o sistema deverá oferecer aos usuários (PRESSMAN e MAXIM, 2021, p.617). Auxilia a comunicação entre os analistas do sistema e o cliente, mostrando as funcionalidades do sistema do ponto de vista do usuário. Esse diagrama se concentra em três itens principais: atores, casos de uso e relacionamentos.

Os atores representam os diversos usuários que poderão utilizar, de alguma maneira, os serviços e funções do sistema, e são representados por símbolos de 'bonecos palito', com uma descrição que identifica seu papel. Os atores podem ser pessoas (usuário, secretária, aluno, professor, etc.), dispositivos (impressoras, máquina ou outro equipamento externo), *hardwares* (placa de modem, etc.) e *softwares* (aplicativos, sistema de banco de dados, etc.).

Um caso de uso representa uma funcionalidade do sistema, um serviço completo que o sistema oferece, e que pode ser utilizado de alguma forma pelos atores que interagem com o sistema. O caso de uso é representado no diagrama por uma elipse contendo o nome da funcionalidade que o caso de uso refere. O retângulo que envolve os casos de uso serve para definir as fronteiras do sistema, por isso os atores ficam fora do sistema.

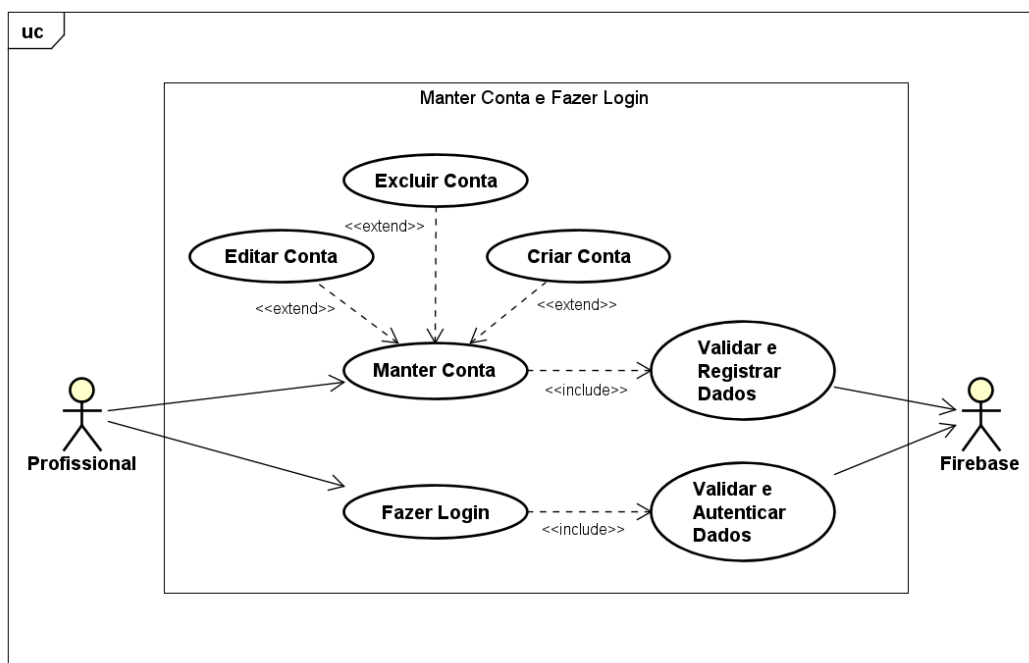
Os relacionamentos no diagrama de casos de uso podem envolver dois atores, dois casos de uso ou um ator e um caso de uso. Essas relações descrevem

parte do modelo de negócio. Os atores que interagem com este projeto de sistema são: o Profissional, o Firebase e o WhatsApp. O sistema é um caso de uso explícito e se trata do sistema em si em que os casos de uso acontecem.

- **Profissional** é o ator que representa a pessoa que vai utilizar o aplicativo como ferramenta de trabalho, o usuário do sistema. Um profissional pode, por exemplo, cadastrar cliente, cadastrar serviços, cadastrar produtos, realizar agendamento e vender produtos.
- **Firestore** representa o banco de dados em tempo real, onde o sistema armazena as informações dos clientes, produtos, serviços e agendamentos.
- **WhatsApp** representa o aplicativo de mensagens que o profissional acessa para enviar mensagem para os clientes cadastrados no sistema.

A Figura 1 apresenta o caso de uso para cadastrar e logar o usuário no sistema.

Figura 1 - Diagrama de caso de uso manter conta e login.

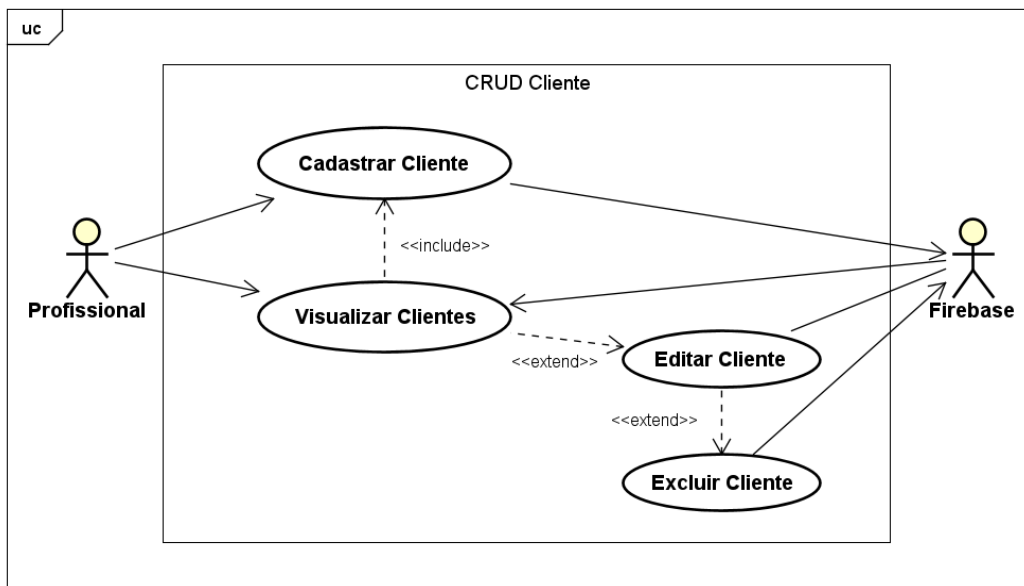


Fonte: Elaborado pela autora (2023).



A Figura 2 apresenta o caso de uso para o CRUD de clientes do sistema.

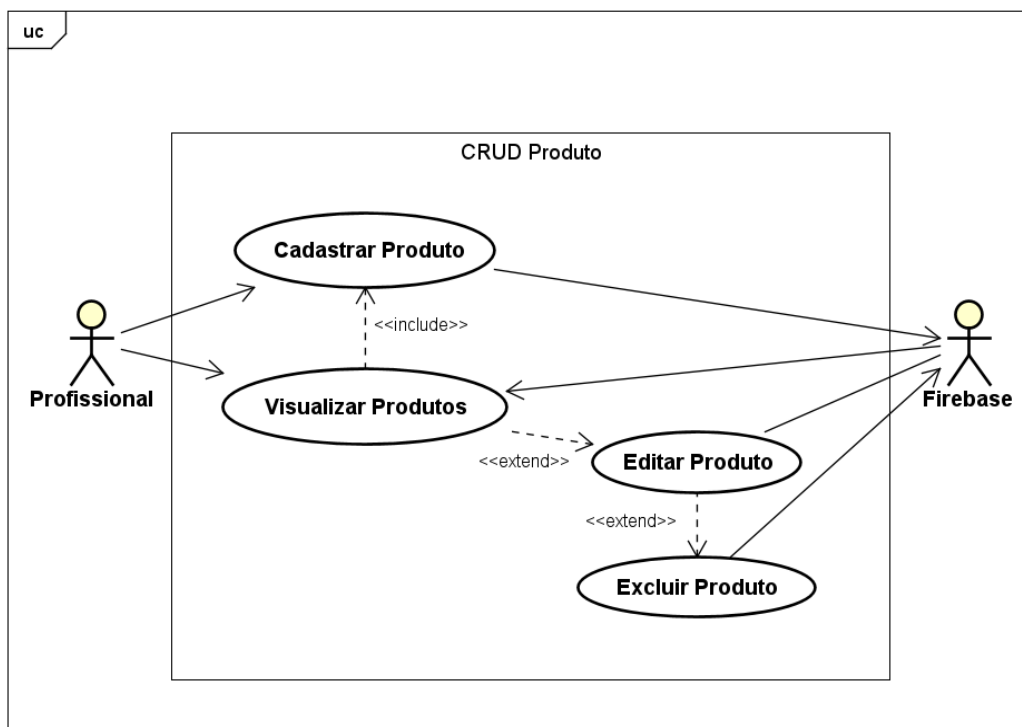
**Figura 2 - Diagrama de caso de uso do CRUD de clientes.**



**Fonte: Elaborado pela autora (2023).**

A Figura 3 apresenta o caso de uso para o CRUD de produtos do sistema.

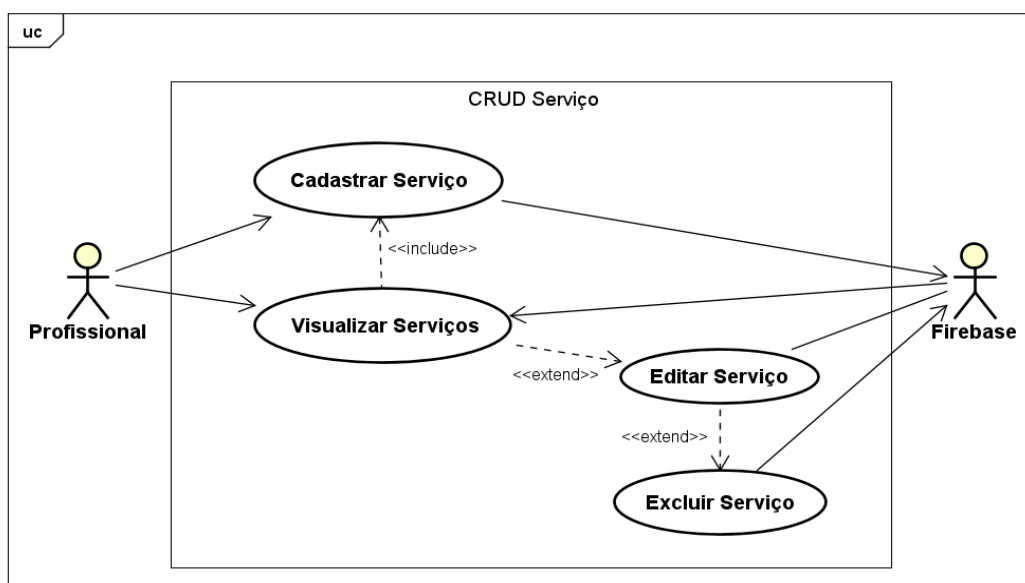
Figura 3 - Diagrama de caso de uso do CRUD de produtos.



Fonte: Elaborado pela autora (2023).

A Figura 4 apresenta o caso de uso para o CRUD de serviços do sistema.

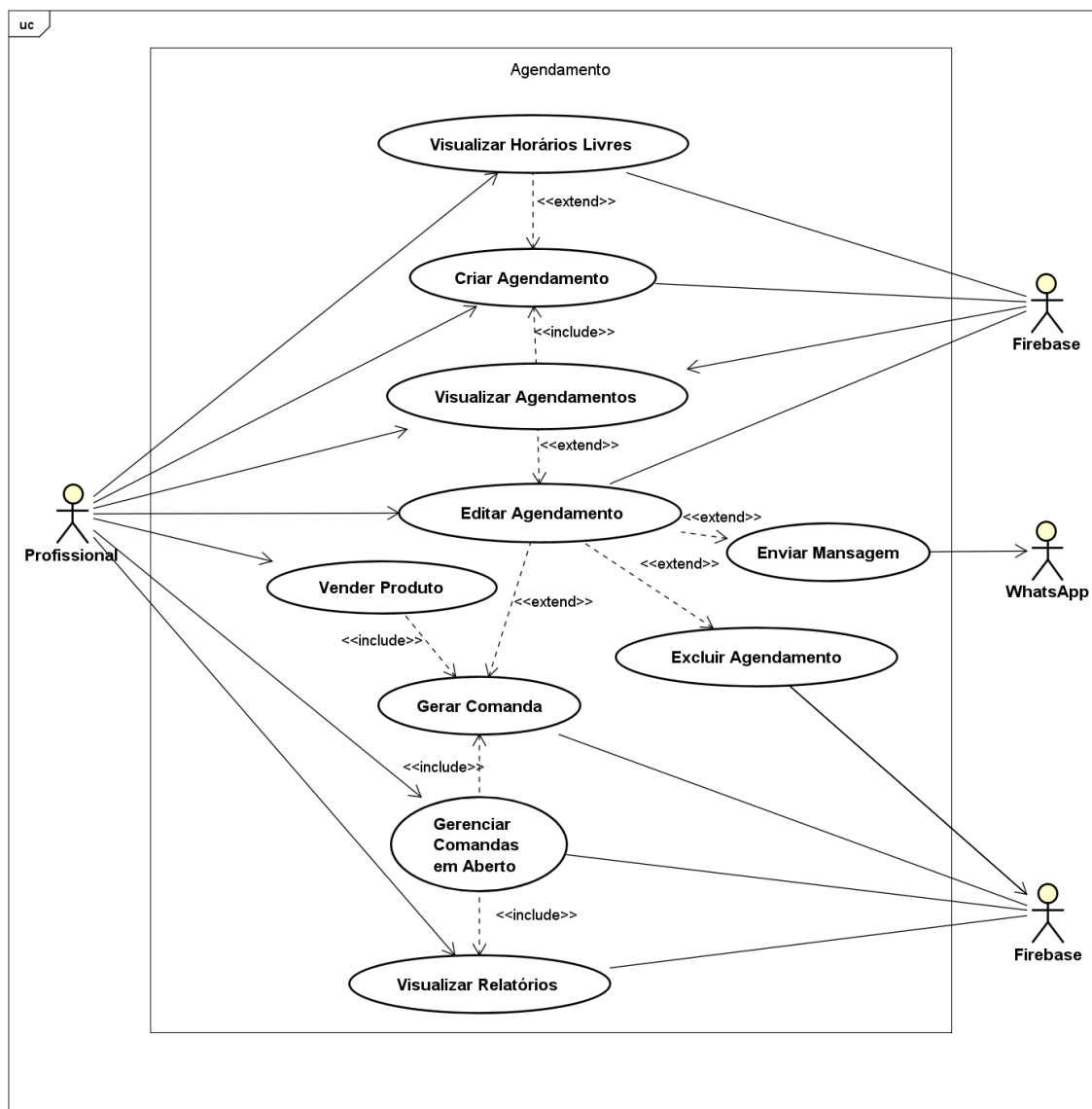
Figura 4 - Diagrama de caso de uso do CRUD de serviços.



Fonte: Elaborado pela autora (2023).

A Figura 5 apresenta o caso de uso para dos agendamentos do sistema.

Figura 5 - Diagrama de caso de uso de agendamentos.



Fonte: Elaborado pela autora (2023).

### 3.2 DOCUMENTAÇÃO DOS CASOS DE USO

As funcionalidades dos diagramas de casos de uso serão descritas da Tabela 4 à Tabela 19.

A Tabela 4 apresenta o caso de uso “Criar Conta”.

**Tabela 4 - Documentação Caso de Uso “Criar Conta”.**

<b>Nome do caso de uso</b>	Criar Conta
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso descreve os passos de abrir uma conta de um usuário no sistema.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em “Criar uma conta”	
	2. O sistema apresenta a tela de registro.
3. O usuário preenche os campos do formulário com seus dados de e-mail e senha e depois clica no botão “Crie sua conta”.	
	4. O firebase cria um usuário com as informações inseridas e um ID único.
	5. O sistema apresenta uma mensagem com a confirmação do registro.
<b>Restrições/Validações</b>	Para que a conta seja criada, o sistema verifica se os dados foram preenchidos e se estão no formato pré-definido. Também é confirmado através do e-mail se o usuário já possui uma conta.

**Fonte: Elaborado pela autora (2023).**

A Tabela 5 apresenta o caso de uso “Editar Conta”.

**Tabela 5 - Documentação Caso de Uso “Editar Conta”.**

<b>Nome do caso de uso</b>	Editar Conta
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso descreve os passos para editar a conta de um usuário no sistema.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em “Meu Perfil” na tela inicial.	
	2. O sistema apresenta a tela “Meu Perfil”.
3. O usuário clica no botão editar perfil.	
	4. O sistema apresenta a tela “Editar Perfil”.
5. O usuário pode alterar, conforme a necessidade, a foto do perfil, clicando no botão alterar foto; mudar o nome do perfil e alterar o e-mail associado à conta. Para salvar as alterações o usuário pode clicar no botão salvar alterações.	
	6. O sistema grava a informação no Firebase e retorna à tela inicial, já atualizada.
<b>Restrições/Validações</b>	Para que a conta seja editada o sistema verifica se campos foram preenchidos e se estão no formato pré-definido.

**Fonte: Elaborado pela autora (2023).**

A Tabela 6 apresenta o caso de uso “Excluir Conta”.

**Tabela 6 - Documentação Caso de Uso “Excluir Conta”.**

<b>Nome do caso de uso</b>	Excluir Conta
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso descreve os passos para excluir a conta de um usuário no sistema.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em “Meu Perfil” na tela inicial.	
	2. O sistema apresenta a tela “Meu Perfil”.
3. O usuário clica no botão deletar conta.	
	4. O sistema apresenta uma janela de mensagem de confirmação da exclusão da conta de usuário do sistema.
5. O usuário clica em excluir.	
	6. O sistema desativa a conta do usuário no Firebase e fecha o aplicativo.
<b>Restrições/Validações</b>	

**Fonte: Elaborado pela autora (2023).**

A Tabela 7 apresenta o caso de uso “Fazer Login”.

**Tabela 7 - Documentação Caso de Uso “Fazer Login”.**

<b>Nome do caso de uso</b>	Fazer Login
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso descreve os passos para o usuário realizar o login.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário insere seu e-mail e senha e clica no botão login.	
	2. O sistema busca informações de usuário e senha no Firebase e faz a autenticação dos dados.
	3. Se os dados de acesso estiverem corretos, o sistema direciona o usuário para a página inicial.
<b>Restrições/Validações</b>	Para que o <i>login</i> seja efetuado, o sistema verifica se o usuário existe e se as credenciais estão corretas.

**Fonte: Elaborado pela autora (2023).**

A Tabela 8 apresenta o caso de uso “Recuperar Senha”.

**Tabela 8 - Documentação Caso de Uso “Recuperar Senha”.**

<b>Nome do caso de uso</b>	Recuperar Senha
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso descreve os passos para o usuário recuperar sua senha de acesso ao sistema.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>

<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em “Esqueceu a senha?”	
	2. O sistema apresenta a tela esqueceu sua senha.
3. O usuário digita o e-mail de registro da conta e clica no botão envie-me um e-mail de recuperação.	
	4. O sistema verifica se o email está registrado no Firebase e cria uma senha provisória associada a este e-mail.
	5. O sistema envia para o usuário um e-mail com uma senha provisória.
	6. O sistema apresenta uma tela informando para o usuário verificar o e-mail.
7. O usuário verifica o e-mail e na tela de login, insere o e-mail e a senha provisória.	
	8. O sistema busca informações de usuário e senha no Firebase e faz a autenticação dos dados.
	9. Se os dados de acesso estiverem corretos, o sistema direciona o usuário para a página inicial.
<b>Restrições/Validações</b>	Para o usuário recuperar a senha é necessário que o e-mail inserido seja válido no sistema. Para realizar o login, a senha provisória de ser a mesma enviada por e-mail pelo sistema.

**Fonte: Elaborado pela autora (2023).**

A Tabela 9 apresenta o caso de uso “Criar / Editar / Excluir Clientes”.

**Tabela 9 - Documentação Casos de Uso “Criar / Editar / Excluir Clientes”.**

<b>Nome do caso de uso</b>	Criar / Editar / Excluir Clientes
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para a criação, edição ou exclusão de clientes.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>



<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em “Clientes” na tela inicial.	
	2. O sistema apresenta a tela “Meus Clientes”.
3. O usuário pode clicar no botão com sinal de mais no canto inferior direito da tela.	
	4. O sistema apresenta a tela “Cadastrar Cliente”.
5. O usuário digita o nome do cliente e clica em salvar.	
	6. O sistema grava a informação no Firebase e retorna à tela “Meus Clientes”, já atualizada.
7. O usuário clica em um cliente da lista que deseja editar.	
	8. O sistema apresenta a janela “Editar Cliente”.
9. O usuário altera o nome do cliente e clica em salvar.	
	10. O sistema grava a informação no Firebase e retorna à tela “Meus Clientes”, já atualizada.
11. O usuário clica em um cliente da lista que deseja excluir.	
	12. O sistema apresenta a janela “Editar Cliente”.
13. O usuário clica no botão deletar.	
	14. O sistema apresenta uma janela com a mensagem de confirmação da ação de excluir o cliente.
15. O usuário clica em deletar.	
	16. O sistema exclui o cliente selecionado da lista, atualiza o Firebase e retorna à tela “Meus Clientes” atualizada.
<b>Restrições/Validações</b>	Tanto ao criar como editar um cliente, o campo nome deve ser obrigatoriamente preenchido. Todos os campos preenchidos devem estar no formato pré-definido.

**Fonte: Elaborado pela autora (2023).**

A Tabela 10 apresenta o caso de uso “Criar / Editar / Excluir Serviços”.

**Tabela 10 - Documentação Casos de Uso “Criar / Editar / Excluir Serviços”.**

<b>Nome do caso de uso</b>	Criar / Editar / Excluir Serviços
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para a criação, edição ou exclusão de serviços.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Serviços na tela inicial.	
	2. O sistema apresenta a tela “Meus Serviços”.
3. O usuário pode clicar no botão com sinal de mais no canto inferior direito da tela.	
	4. O sistema apresenta a tela “Cadastrar Serviço”.
5. O usuário digita o nome do serviço e clica em salvar.	
	6. O sistema grava a informação no Firebase e retorna à tela “Meus Serviços”, já atualizada.
7. O usuário clica em um serviço da lista que deseja editar.	
	8. O sistema apresenta a janela “Editar Serviço”.
9. O usuário altera o nome do serviço e clica em salvar.	
	10. O sistema grava a informação no Firebase e retorna à tela “Meus Serviços”, já atualizada.
11. O usuário clica em um serviço da lista que deseja excluir.	
	12. O sistema apresenta a janela “Editar Serviço”.
13. O usuário clica no botão deletar.	

<b>Ações do ator</b>	<b>Ações do Sistema</b>
	14. O sistema apresenta uma janela com a mensagem de confirmação da ação de excluir o serviço.
15. O usuário clica em deletar.	
	16. O sistema exclui o serviço selecionado da lista, atualiza o Firebase e retorna à tela “Meus Serviços” atualizada.
<b>Restrições/Validações</b>	Tanto ao criar como editar um serviço, o campo nome deve ser obrigatoriamente preenchido. Todos os campos preenchidos devem estar no formato pré-definido.

**Fonte: Elaborado pela autora (2023).**

A Tabela 11 apresenta o caso de uso “Criar / Editar / Excluir Produtos”.

**Tabela 11 - Documentação Casos de Uso “Criar / Editar / Excluir Produtos”.**

<b>Nome do caso de uso</b>	Criar / Editar / Excluir Produtos
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para a criação, edição ou exclusão de produtos.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Produtos na tela inicial.	
	2. O sistema apresenta a tela “Meus Produtos”.
3. O usuário pode clicar no botão com sinal de mais no canto inferior direito da tela.	
	4. O sistema apresenta a tela “Cadastrar Produto”.
5. O usuário digita o nome do produto e clica em salvar.	
	6. O sistema grava a informação no Firebase e retorna à tela “Meus Produtos”, já atualizada.

<b>Ações do ator</b>	<b>Ações do Sistema</b>
7. O usuário clica em um produto da lista que deseja editar.	
	8. O sistema apresenta a janela “Editar Produto”.
9. O usuário altera o nome do produto e clica em salvar.	
	10. O sistema grava a informação no Firebase e retorna à tela “Meus Produtos”, já atualizada.
11. O usuário clica em um produto da lista que deseja excluir.	
	12. O sistema apresenta a janela “Editar Produto”.
13. O usuário clica no botão deletar.	
	14. O sistema apresenta uma janela com a mensagem de confirmação da ação de excluir o produto.
15. O usuário clica em deletar.	
	16. O sistema exclui o produto selecionado da lista, atualiza o Firebase e retorna à tela “Meus Produtos” atualizada.
<b>Restrições/Validações</b>	Tanto ao criar como editar um produto, o campo nome deve ser obrigatoriamente preenchido. Todos os campos preenchidos devem estar no formato pré-definido.

**Fonte: Elaborado pela autora (2023).**

A Tabela 12 apresenta o caso de uso “Criar Agendamento”.

**Tabela 12 - Documentação Caso de Uso “Criar Agendamento”.**

<b>Nome do caso de uso</b>	Criar Agendamento
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para a criação de agendamentos.
<b>Prioridade de desenvolvimento</b>	Essencial

Ações do ator	Ações do Sistema
1. O usuário clica em Agenda na tela inicial.	
	2. O sistema apresenta a tela “Agenda”.
3. O usuário pode clicar no botão com sinal de mais no canto inferior direito da tela.	
	4. O sistema apresenta a tela “Agendamento”.
5. O usuário clica no campo selecionar cliente.	
	6. O sistema apresenta a tela “Meus Clientes”.
7. O usuário clica em um cliente da lista.	
	8. O sistema apresenta a janela “Agenda” com o campo cliente preenchido.
9. O usuário clica no campo selecionar serviço.	
	10. O sistema apresenta a tela “Meus Serviços”.
11. O usuário clica em um serviço da lista.	
	12. O sistema apresenta a janela “Agenda” com o campo serviço preenchido.
13. O usuário clica no campo selecionar o dia.	
	14. O sistema apresenta uma janela com o calendário.
15. O usuário seleciona o mês e o dia e clica em ok.	
	16. O sistema apresenta a tela “Agenda” com o campo selecionar o dia preenchido com uma data.
17. O usuário clica no campo horário.	
	18. O sistema apresenta uma janela com um relógio.
19. O usuário seleciona a hora e os minutos e clica em ok.	

<b>Ações do ator</b>	<b>Ações do Sistema</b>
	20. O sistema apresenta a tela “Agenda” com o campo horário preenchido com um horário.
21. O cliente clica em salvar.	
	22. O sistema grava a informação no Firebase e retorna à tela “Agenda”, já atualizada.
<b>Restrições/Validações</b>	Para criar um agendamento, todos os campos devem ser preenchidos. É necessário haver clientes e serviços cadastrados. O status do agendamento fica salvo automaticamente como marcado ao salvar as informações.

**Fonte: Elaborado pela autora (2023).**

A Tabela 13 apresenta o caso de uso “Editar / Excluir Agendamentos”.

**Tabela 13 - Documentação Casos de Uso “Editar / Excluir Agendamentos”.**

<b>Nome do caso de uso</b>	Editar / Excluir Agendamentos
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para editar ou excluir agendamentos.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Agenda na tela inicial.	
	2. O sistema apresenta a tela “Agenda”.
3. O usuário seleciona um agendamento da lista.	
	4. O sistema apresenta a tela “Editar Agendamento”.
5. O usuário modifica a informação de um ou mais campos, ou muda o status do agendamento e clica em salvar.	

<b>Ações do ator</b>	<b>Ações do Sistema</b>
	6. O sistema grava a informação no Firebase e retorna à tela “Agenda”, já atualizada.
7. O usuário seleciona um agendamento da lista.	
	8. O sistema apresenta a tela “Editar Agendamento”.
9. O usuário clica no botão deletar.	
	10. O sistema apresenta uma janela com a mensagem de confirmação da ação de excluir o agendamento.
11. O usuário clica em excluir.	
	12. O sistema exclui o agendamento selecionado da lista, atualiza o Firebase e retorna à tela “Agenda” atualizada.
<b>Restrições/Validações</b>	Para editar um agendamento, todos os campos devem ser preenchidos. É necessário haver clientes e serviços cadastrados.

**Fonte: Elaborado pela autora (2023).**

A Tabela 14 apresenta o caso de uso “Enviar Mensagem”.

**Tabela 14 - Documentação Caso de Uso “Enviar Mensagem”.**

<b>Nome do caso de uso</b>	Enviar Mensagem
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase, WhatsApp.
<b>Objetivo</b>	Este caso de uso detalha os passos para o profissional enviar uma mensagem de confirmação de agendamento para o cliente.
<b>Prioridade de desenvolvimento</b>	Importante
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Agenda na tela inicial.	
	2. O sistema apresenta a tela “Agenda”.

<b>Ações do ator</b>	<b>Ações do Sistema</b>
3. O usuário seleciona um agendamento da lista.	
	4. O sistema apresenta a tela “Editar Agendamento”.
5. O usuário clica no botão enviar mensagem por WhatsApp.	
	6. O sistema apresenta a tela “Enviar Mensagem”, já preenchida com os campos nome do cliente e telefone.
7. O cliente digita a mensagem para o cliente e clica em enviar.	
	8. O sistema se conecta com a URL do WhatsApp e abre o aplicativo WhatsApp.
9. O usuário seleciona o contato e envia a mensagem.	
	10. O sistema apresenta a tela “Agenda”.
<b>Restrições/Validações</b>	Para enviar uma mensagem o usuário tem que acessar um agendamento cadastrado. O dispositivo do usuário tem que ter o aplicativo do WhatsApp instalado.

**Fonte: Elaborado pela autora (2023).**

A Tabela 15 apresenta o caso de uso “Gerar Comanda de Serviço”.

**Tabela 15 - Documentação Caso de Uso “Gerar Comanda de Serviço”.**

<b>Nome do caso de uso</b>	Gerar Comanda de Serviço
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para o profissional gerar e fechar uma comanda de serviço.
<b>Prioridade de desenvolvimento</b>	Essencial
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Agenda na tela inicial.	



Ações do ator	Ações do Sistema
	2. O sistema apresenta a tela “Agenda”.
3. O usuário seleciona um agendamento da lista.	
	4. O sistema apresenta a tela “Editar Agendamento”.
5. O usuário clica no botão gerar comanda.	
	6. O sistema apresenta a tela “Comanda de Serviço” com as informações do agendamento como nome do cliente, o serviço e o valor total.
7. O cliente pode adicionar um serviço.	
	8. O sistema apresenta a tela “Meus Serviços”.
9. O usuário seleciona um serviço da lista.	
	10. O sistema apresenta a tela “Comanda de Serviço” com o nome do serviço preenchido e o seu valor adicionado ao valor total.
11. O usuário pode adicionar a venda de um produto.	
	12. O sistema apresenta a tela “Meus Produtos”.
13. O usuário seleciona um produto da lista.	
	14. O sistema apresenta a tela “Comanda de Serviço” com o nome do produto preenchido e o seu valor adicionado ao valor total
15. O usuário pode inserir um desconto.	
	16. O sistema calcula e exibe o valor total a receber.
17. O usuário clica no botão pago.	
	18. O sistema apresenta uma janela com a confirmação do pagamento e uma lista com as formas de pagamento.
19. O usuário seleciona entre as opções qual foi realizada pelo cliente no ato de pagamento.	
	20. O sistema apresenta uma janela com uma mensagem de fechamento da comanda.

<b>Ações do ator</b>	<b>Ações do Sistema</b>
	22. O sistema grava as informações no Firebase e retorna à tela “Agenda”, já atualizada.
<b>Restrições/Validações</b>	Para gerar uma comanda é necessário haver agendamentos cadastrados. O status do agendamento não pode ser cancelado. O campo inserir desconto não é obrigatório. O Usuário não pode inserir valores nos campos valor total e total a receber, somente no campo inserir desconto.

**Fonte: Elaborado pela autora (2023).**

A Tabela 16 apresenta o caso de uso “Vender Produto”.

**Tabela 16 - Documentação Caso de Uso “Vender Produto”.**

<b>Nome do caso de uso</b>	Vender Produto
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para o profissional vender um ou mais produtos.
<b>Prioridade de desenvolvimento</b>	Importante
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Produtos na tela inicial.	
	2. O sistema apresenta a tela “Meus Produtos”.
3. O usuário seleciona um produto da lista.	
	4. O sistema apresenta a tela “Editar Produto”.
5. O usuário clica botão vender produto.	
	6. O sistema apresenta a tela “Comanda de Venda” com as informações do produto selecionado (nome do produto, a quantidade de uma unidade e o valor total).
7. O usuário deve selecionar um cliente clicando no ícone de soma ao lado do ícone de cliente.	

<b>Ações do ator</b>	<b>Ações do Sistema</b>
	8. O sistema apresenta a tela “Meus Clientes”.
9. O usuário seleciona um cliente da lista.	
	10. O sistema apresenta a tela “Comanda de Venda” com o nome do cliente preenchido.
11. O usuário pode adicionar a venda de outro produto clicando no ícone de mais no campo adicionar produto.	
	12. O sistema apresenta a tela “Meus Produtos”.
13. O usuário seleciona um produto da lista.	
	14. O sistema apresenta a tela “Comanda de Venda” com o nome do produto preenchido, a quantidade de uma unidade e seu valor adicionado ao valor total
15. O usuário pode aumentar ou diminuir as quantidades dos produtos selecionados.	
	16. O sistema atualiza os valores dos campos valor total e total a receber.
17. O usuário pode inserir um desconto.	
	18. O sistema calcula e exibe o valor total a receber.
19. O usuário clica no botão pago.	
	20. O sistema apresenta uma janela com a confirmação do pagamento e uma lista com as formas de pagamento.
21. O usuário seleciona entre as opções qual foi realizada pelo cliente no ato de pagamento.	
	22. O sistema apresenta uma janela com uma mensagem de fechamento da comanda.
	23. O sistema grava as informações no Firebase e retorna à tela “Editar Produto”, já atualizada com relação à quantidade do produto.
<b>Restrições/Validações</b>	Para gerar uma comanda de venda do produto é necessário haver produtos e clientes cadastrados. A quantidade dos produtos selecionados deve ser menor ou igual ao valor cadastrado na tela Editar Produto no campo quantidade. O campo inserir desconto não é obrigatório. O Usuário não pode inserir valores nos campos valor total e total a receber, somente no campo inserir desconto.

A Tabela 17 apresenta o caso de uso “Gerenciar Comandas em Aberto”.

**Tabela 17 - Documentação Caso de Uso “Gerenciar Comandas em Aberto”.**

<b>Nome do caso de uso</b>	Gerenciar Comandas em Aberto
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para o profissional gerenciar a comanda de serviço e/ou de venda de produto, que ficam em aberto.
<b>Prioridade de desenvolvimento</b>	Importante
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Financeiro na tela inicial.	
	2. O sistema apresenta a tela “Relatório Financeiro”.
3. O usuário seleciona a aba cobranças.	
	4. O sistema apresenta a aba “Cobranças”.
5. O usuário clica no <i>checkbox</i> do serviço ou venda que ficou registrada como aberta.	
	6. O sistema apresenta a tela da Comanda referente ao registro (Serviço ou Venda).
7. O usuário clica no botão pago.	
	8. O sistema apresenta uma janela com a confirmação do pagamento e uma lista com as formas de pagamento.
9. O usuário seleciona entre as opções qual foi realizada pelo cliente no ato de pagamento.	
	10. O sistema apresenta uma janela com uma mensagem de fechamento da comanda.
	11. O sistema grava as informações no Firebase e retorna à tela “Relatório Financeiro”, já atualizado na aba “Cobranças”.
<b>Restrições/Validações</b>	Para efetuar o fechamento da comanda em aberto o usuário não pode alterar nenhum campo da comanda.

**Fonte: Elaborado pela autora (2023).**

A Tabela 18 apresenta o caso de uso “Visualizar Relatórios”.

**Tabela 18 - Documentação Caso de Uso “Visualizar Relatórios”.**

<b>Nome do caso de uso</b>	Visualizar Relatórios
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para o profissional visualizar os relatórios gerados pelo sistema.
<b>Prioridade de desenvolvimento</b>	Importante
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Financeiro na tela inicial.	
	2. O sistema apresenta a tela “Relatório Financeiro”.
3. O usuário seleciona o mês.	
	4. O sistema acessa os dados no Firebase e atualiza os campos das informações na tela.
5. O usuário seleciona a aba receita.	
	6. O sistema apresenta a tela da aba “Receita”.
7. O usuário seleciona a aba caixa	
	8. O sistema apresenta a tela da aba “Caixa”.
9. O usuário seleciona a aba cobranças	
	10. O sistema apresenta a tela da aba “Cobranças”.
<b>Restrições/Validações</b>	Para o sistema gerar informações é necessário haver agendamentos e/ou vendas cadastrados.

**Fonte: Elaborado pela autora (2023).**

A Tabela 19 apresenta o caso de uso “Visualizar Relatórios”.

**Tabela 19 - Documentação Caso de Uso “Visualizar Horários Livres”.**

<b>Nome do caso de uso</b>	Visualizar Horários Livres
<b>Atores envolvidos</b>	Profissional, Sistema, Firebase.
<b>Objetivo</b>	Este caso de uso detalha os passos para o profissional visualizar os horários livres, disponíveis para agendamento.
<b>Prioridade de desenvolvimento</b>	Importante
<b>Ações do ator</b>	<b>Ações do Sistema</b>
1. O usuário clica em Agenda na tela inicial.	
	2. O sistema apresenta a tela “Agenda”.
3. O usuário clica no botão horários livres.	
	4. O sistema apresenta a tela “Horários Livres” com os horários disponíveis para cada dia da semana.
5. O usuário pode selecionar um horário.	
	6. O sistema apresenta a tela “Agendamento” já com o dia e o horário preenchido.
7. O usuário preenche os demais campos e clica em salvar.	
	8. O sistema grava a informação no Firebase e retorna à tela “Agenda”, já atualizada.
<b>Restrições/Validações</b>	Para criar um agendamento, todos os campos devem ser preenchidos. É necessário haver clientes e serviços cadastrados. O status do agendamento fica salvo automaticamente como marcado ao salvar as informações.

**Fonte: Elaborado pela autora (2023).**

## 3.2 DIAGRAMA DE ARQUITETURA

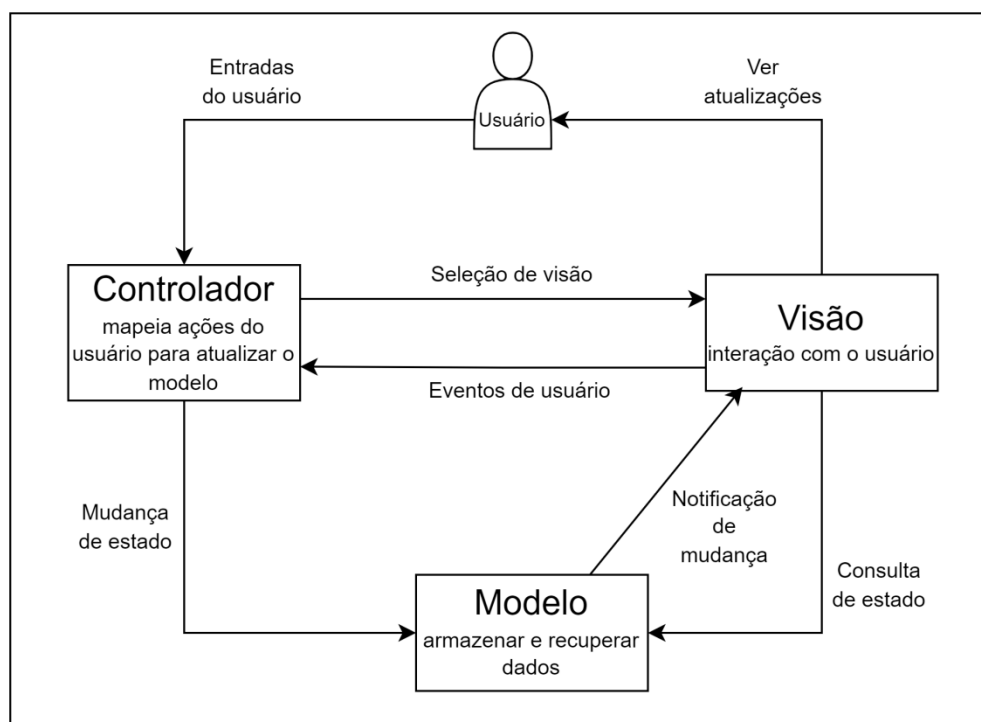
O projeto de arquitetura é um modelo de arquitetura que descreve como o sistema está organizado em um conjunto de componentes de comunicação (SOMMERVILLE, 2011, p. 103). A arquitetura muitas vezes inclui padrões de design que ajudam a organizar e estruturar o código de maneira eficiente, como o padrão Modelo-Visão-Controlador (MVC, do Inglês *Model-View-Controller*). O MVC é um modelo de arquitetura que separa a interface do usuário da funcionalidade e do conteúdo da informação (PRESSMAN e MAXIM, 2021, p279).

No padrão MVC o sistema é estruturado em três componentes lógicos que interagem entre si. O componente “Modelo” gerencia o sistema de dados e as operações associadas a esses dados. O componente “Visão” define e gerencia como os dados são apresentados ao usuário, exibe informações do modelo de forma interativa. O componente “Controlador” gerencia a interação do usuário e passa essa interação para a Visão e o Modelo (SOMMERVILLE, 2011 p. 109).

Esses componentes interagem da seguinte forma, o usuário interage com a camada visão. Então, o controlador pega essas informações e envia para o modelo, que fica responsável por avaliar aqueles dados e transmitir uma resposta. O controlador recebe essas respostas e envia uma notificação de validação daquela informação para a camada visão, fazendo com a mesma apresente o resultado de maneira gráfica e visual (Fig. 6).

Os Controladores são responsáveis por manipular o modelo, lidar com a interação do usuário final e escolher uma visualização para exibir a interface do usuário. A pasta Modelos contém classes que definem objetos e a lógica para interagir com o banco de dados ou itens de dados. Inclui propriedades públicas, que serão usadas pelo aplicativo para armazenar e manipular dados do aplicativo. O padrão MVC permite separar a lógica de negócios da lógica de apresentação, o que torna muito fácil adicionar novos recursos ou alterar os existentes sem afetar todo o aplicativo.

Figura 6 – Padrão MVC



Fonte: Adaptado de SOMMERVILLE, 2011.

O padrão Modelo-Visão-Controle-Serviço (MVCS, do inglês *Model-View-Controller-Service*) é uma extensão do padrão MVC que incorpora um componente de serviço para lidar com a lógica de negócios e regras de negócios (Fig. 7). Isso ajuda a separar ainda mais a lógica de negócios do controlador, tornando o sistema mais modular e mais fácil de manter. Descrição de cada componente do MVCS:

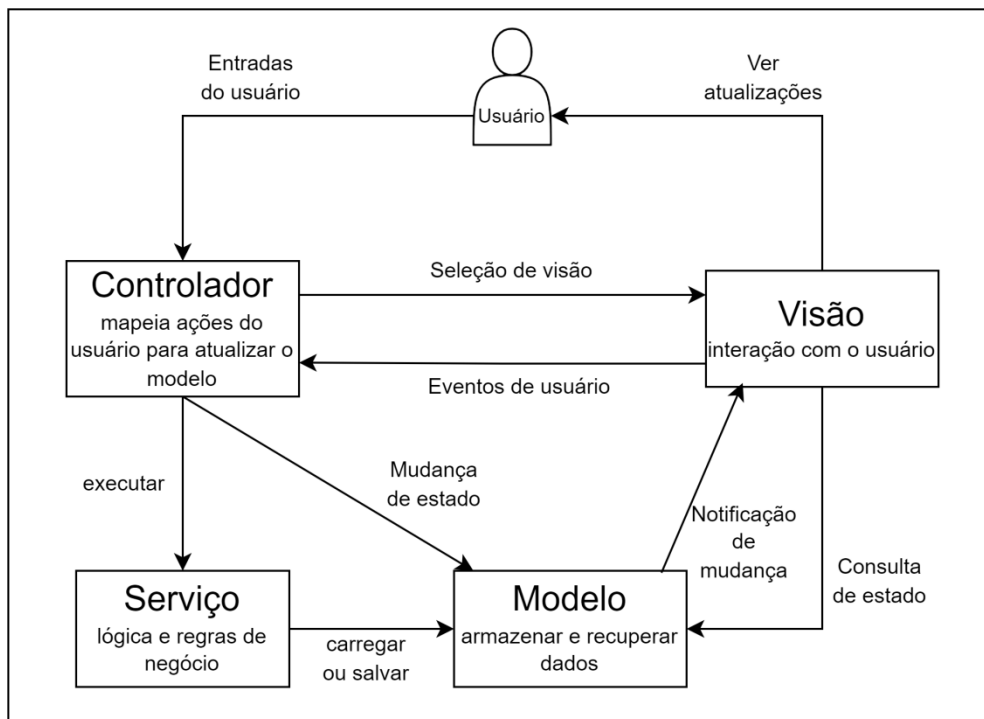
- **Modelo (Model):** Esta camada é responsável por armazenar e recuperar dados da aplicação. Ele não contém lógica de negócios ou regras de negócios, apenas armazena os dados em um formato acessível. Os modelos são geralmente estruturas de dados ou classes que representam entidades, como produtos, clientes, pedidos, etc.
- **Visão (View):** Esta camada é responsável pela interação com o usuário e permitir que eles interajam com o sistema. Este é o *front end* da aplicação que pode ter diferentes campos para entrada de dados, tabelas para representação de informações, animações etc. Elas não contêm lógica de negócios, apenas exibem dados e capturam ações do usuário. As visões



geralmente incluem elementos da interface do usuário, como páginas da web, formulários ou telas de aplicativos.

- **Controlador (*Controller*):** Os Controladores coordenam as ações do usuário, direcionam as solicitações para os serviços apropriados e atualizam as visões correspondentes. Eles não contêm lógica de negócios; em vez disso, orquestram a interação entre a visão, o serviço e o modelo. Os controladores normalmente gerenciam a navegação entre as telas e ações do usuário (como clicar em um botão).
- **Serviço (*Service*):** Esta é outra camada (opcional) que reside entre o modelo e o controlador. O componente de serviço é responsável por conter a lógica de negócios e as regras de negócios. Ele isola a lógica de negócios do controlador, tornando-a reutilizável e fácil de testar. Os serviços fornecem funções ou métodos que executam ações de negócios, como processar um pedido, calcular preços, verificar a disponibilidade de produtos, etc.

Figura 7 – Padrão MVCS



Fonte: Adaptado de SOMMERVILLE, 2011.

No padrão de organização MVCS o usuário interage com a visão, como preenchendo um formulário em uma tela de aplicativo. A visão envia os dados para o controlador relevante. O controlador redireciona a solicitação para o serviço correspondente. O serviço executa a lógica de negócios necessária, como validar os dados, calcular preços ou verificar disponibilidade. O serviço atualiza o modelo conforme necessário (por exemplo, adicionando um novo pedido ao banco de dados). O controlador atualiza a visão para refletir os resultados da ação do usuário.

O MVCS é uma abordagem flexível que ajuda a manter a lógica de negócios separada dos controladores, facilitando a reutilização, a manutenção e os testes. Isso é particularmente útil em sistemas complexos com regras de negócios extensas (TOQEER, 2017).

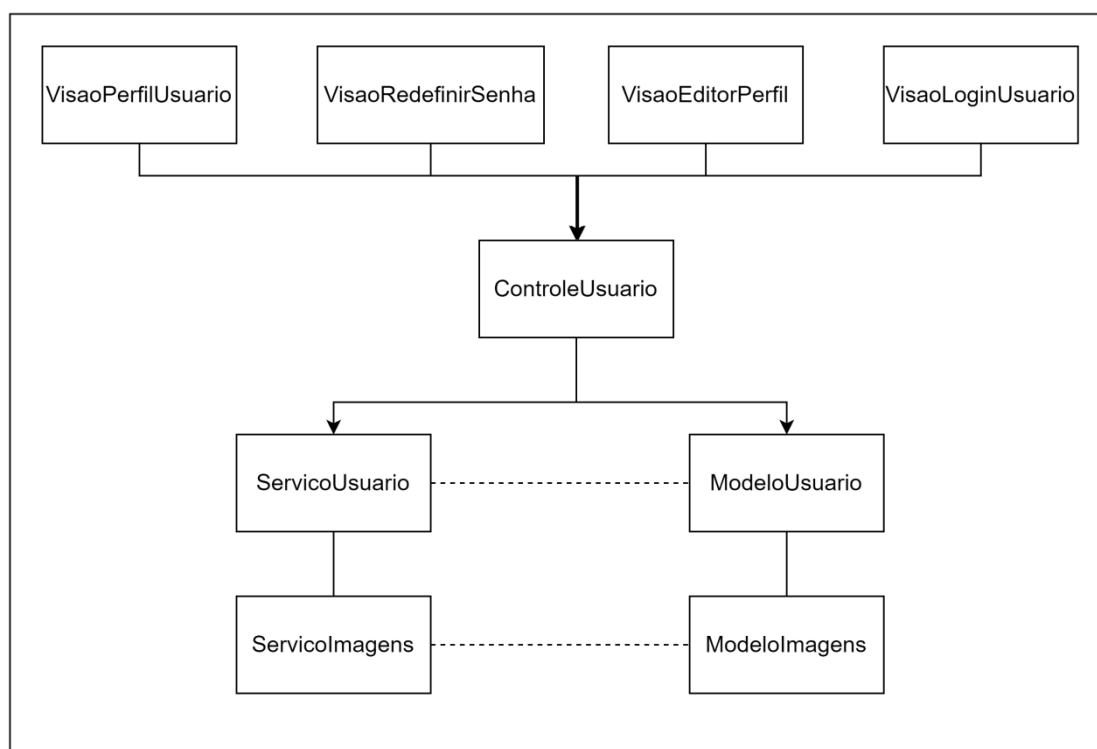
Nesse projeto, a organização dos códigos segue o padrão MVCS. As figuras de 8 a 17 apresentam os diagramas MVCS para as telas, controle, serviços e modelos do aplicativo “Agenda App”.

A Figura 8 apresenta o diagrama MVCS para gerenciar as informações de perfil do usuário do sistema. A camada visão é composta por quatro páginas responsáveis por apresentar informações do perfil do usuário ao próprio usuário e permitir interações com seus dados. O código “VisaoPerfilUsuario” tem a função de apresentar as configurações de conta do usuário, como mudar a senha, editar o perfil e excluir a conta. O código “VisaoRedefinirSenha” tem a função de apresentar o formulário de troca de senha. O código “VisaoEditorPerfil” tem a função de exibir dados do usuário como nome, e-mail associado a conta de usuário e imagem do perfil. O código “VisaoLoginUsuario” tem a função de apresentar o formulário para o acesso ao sistema.

A camada controlador responde as ações do usuário e coordenam as operações de leitura, escrita e exclusão de dados do modelo. O código “ControleUsuario” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. O código “ServicoUsuario” tem a função de executar a autenticação do usuário no sistema. O código “ModeloUsuario” inclui classes ou estruturas que representam informações sobre o usuário, como nome, e-mail e senha. O código “ServicoImagens” tem a função de gerenciar as imagens, fornece métodos para

adicionar um imagem no perfil do usuário e recuperar imagens. O código “ModeloImagens” inclui classes ou estruturas que representam informações sobre imagem, como o registro do tipo BLOB (Objetos Binários Grandes). O “ControleUsuario” utiliza do “ServicoImagens” para buscar e adicionar ou recuperar uma imagem no banco de dados.

**Figura 8 - MVCS para Usuário.**



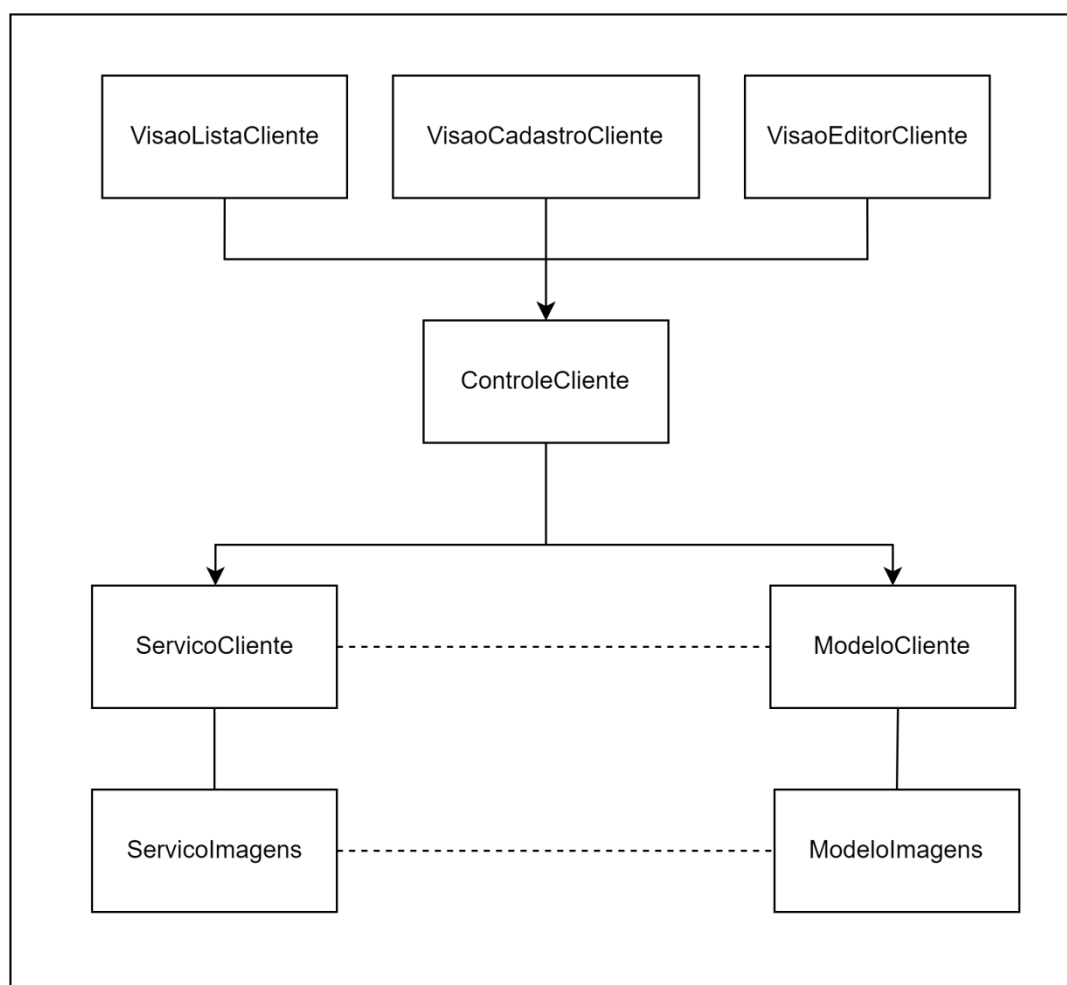
**Fonte: Elaborado pela autora (2023).**

A Figura 9 apresenta o diagrama MVCS para gerenciar as informações de clientes. A camada visão é composta por três páginas responsáveis por apresentar informações dos clientes ao usuário e permitir interações com os dados do cliente. O código “VisaoListaCliente” tem a função de apresentar uma lista dos clientes cadastrados no sistema. O código “VisaoCadastroCliente” tem a função de apresentar o formulário de cadastro do cliente. O código “VisaoEditorCliente” tem a função de exibir a página de edição do cliente.

A camada controlador responde as ações do usuário e coordenam as operações de leitura, escrita e exclusão de dados do modelo. O código

“ControleCliente” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. O código “ServicoCliente” tem a função de gerenciar os dados dos clientes, fornece métodos para adicionar clientes e recuperar informações de clientes. O código “ModeloCliente” inclui classes ou estruturas que representam informações sobre clientes, como nome, endereço, telefone, etc. O código “ServicoImagens” tem a função de gerenciar as imagens, fornece métodos para adicionar um imagem no perfil do cliente e recuperar imagens. O código “ModeloImagens” inclui classes ou estruturas que representam informações sobre imagem, como o registro do tipo BLOB. O “ControleCliente” utiliza do “ServicoImagens” para buscar e adicionar ou recuperar uma imagem no banco de dados.

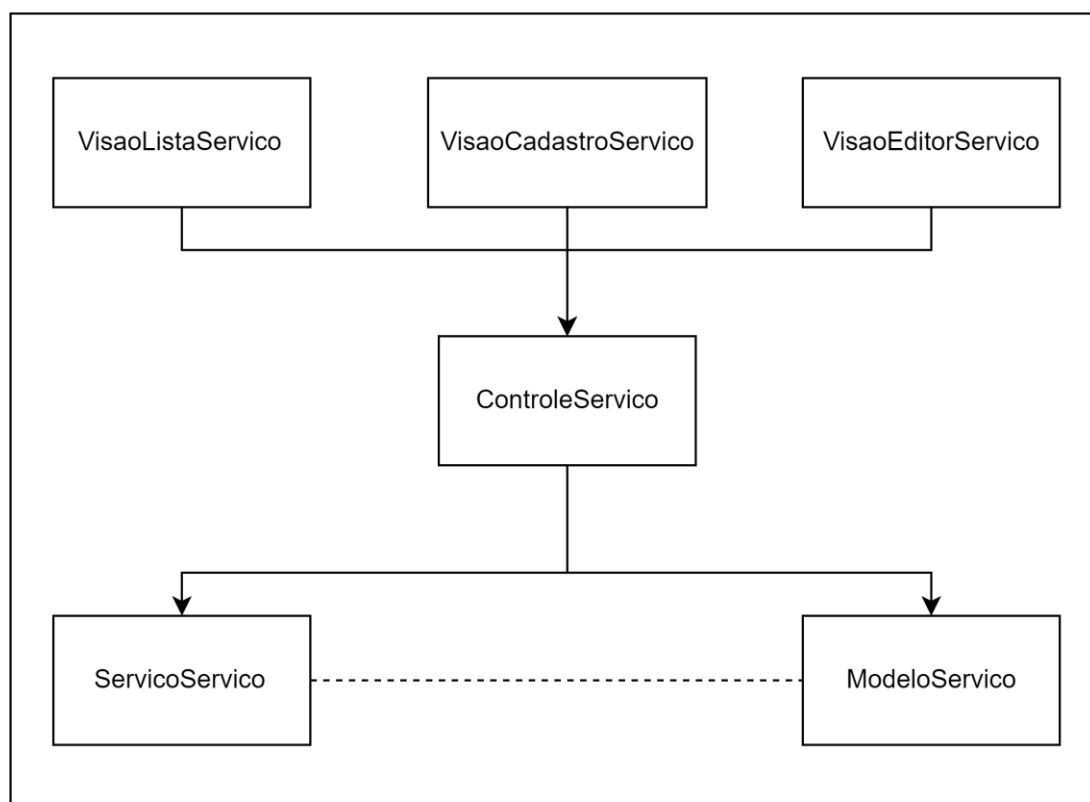
**Figura 9 - MVCS para Cliente**



**Fonte: Elaborado pela autora (2023).**

A Figura 10 apresenta o diagrama MVCS para gerenciar as informações dos serviços prestados pelo usuário. A camada visão é composta por três páginas responsáveis por apresentar informações dos serviços ao usuário e permitir interações com os dados do serviço. O código “VisaoListaServico” tem a função de apresentar uma lista dos serviços cadastrados no sistema. O código “VisaoCadastroServico” tem a função de apresentar o formulário de cadastro do serviço. O código “VisaoEditorServico” tem a função de exibir a página de edição do serviço. O código “ControleServico” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. O código “ServicoServico” tem a função de gerenciar os dados dos serviços, fornece métodos para adicionar serviços e recuperar informações de serviços. O código “ModeloServico” inclui classes ou estruturas que representam informações sobre serviços, como nome, valor, tempo de duração, etc.

**Figura 10 - MVCS para Serviço.**



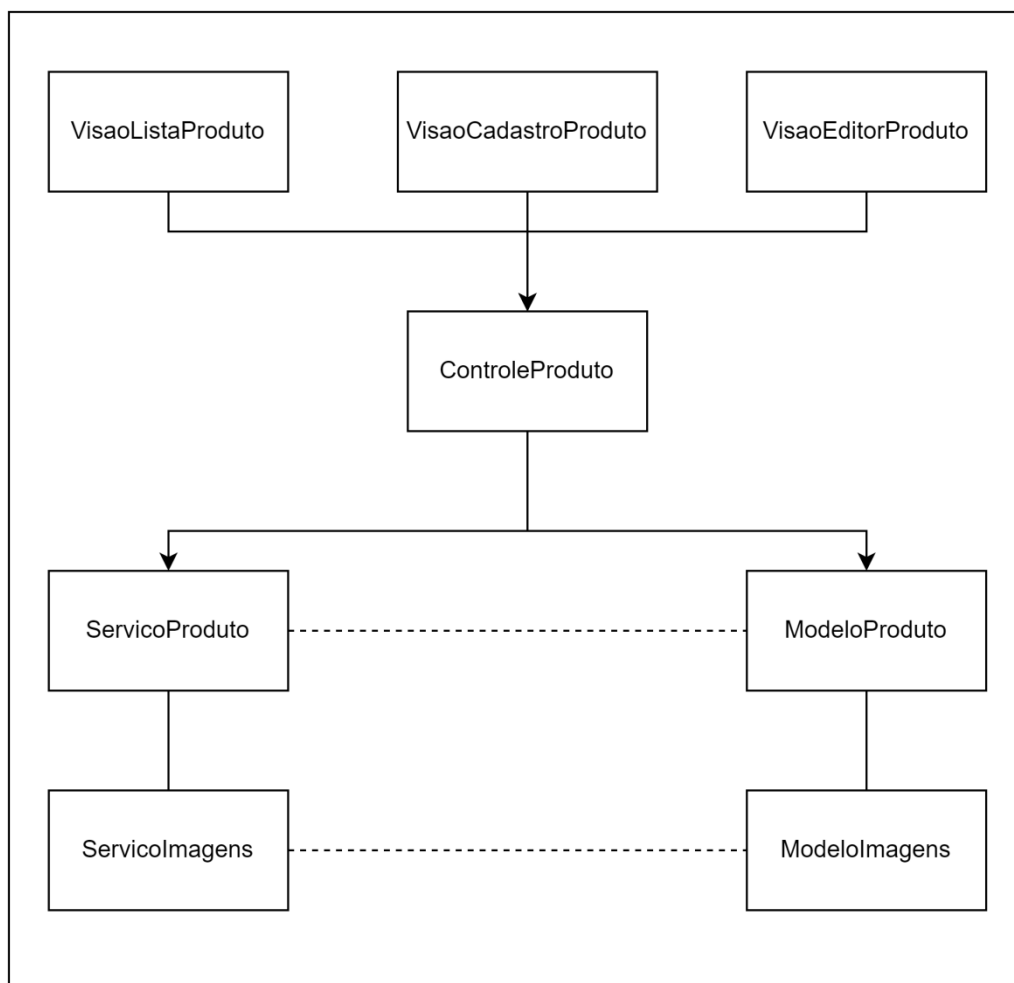
**Fonte: Elaborado pela autora (2023).**

A Figura 11 apresenta o diagrama MVCS para gerenciar as informações dos produtos comercializados pelo usuário. A camada visão é composta por três páginas responsáveis por apresentar informações dos produtos ao usuário e permitir interações com os dados do produto. O código “VisaoListaProduto” tem a função de apresentar uma lista dos produtos cadastrados no sistema. O código “VisaoCadastroProduto” tem a função de apresentar o formulário de cadastro do produto. O código “VisaoEditorProduto” tem a função de exibir a página de edição do produto.

O código “ControleProduto” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. O código “ServicoProduto” tem a função de gerenciar os dados dos produtos, fornece métodos para adicionar produto e recuperar informações de produtos. O código “ModeloProduto” inclui classes ou estruturas que representam informações sobre produtos, como nome, valor, quantidade, etc.

O código “ServicoImagens” tem a função de gerenciar as imagens, fornece métodos para adicionar um imagem no perfil do produto e recuperar imagens. O código “ModeloImagens” inclui classes ou estruturas que representam informações sobre imagem, como o registro do tipo BLOB. O “ControleProduto” utiliza do “ServicoImagens” para buscar e adicionar ou recuperar uma imagem no banco de dados.

**Figura 11 - MVCS para Produto.**



**Fonte: Elaborado pela autora (2023).**

A Figura 12 apresenta o diagrama MVCS para gerenciar as informações dos agendamentos realizados pelo usuário. A camada visão é composta por sete páginas responsáveis por apresentar informações dos agendamentos ao usuário e permitir interações com os dados do agendamento. O código “VisaoListaAgenda” tem a função de apresentar uma lista dos agendamentos diários cadastrados no sistema. O código “VisaoCadastroAgenda” tem a função de apresentar o formulário de cadastro do agendamento. O código “VisaoAgendaFechada” tem a função de apresentar uma lista com dos agendamentos concluídos e cancelados. O código “VisaoEditorAgenda” tem a função de exibir a página de edição do agendamento. O código “VisaoEnviarMensagem” tem como função apresenta o nome do cliente, o telefone e um campo para que o usuário possa digitar a mensagem que será enviada pelo WhatsApp para o cliente selecionado no agendamento. O código

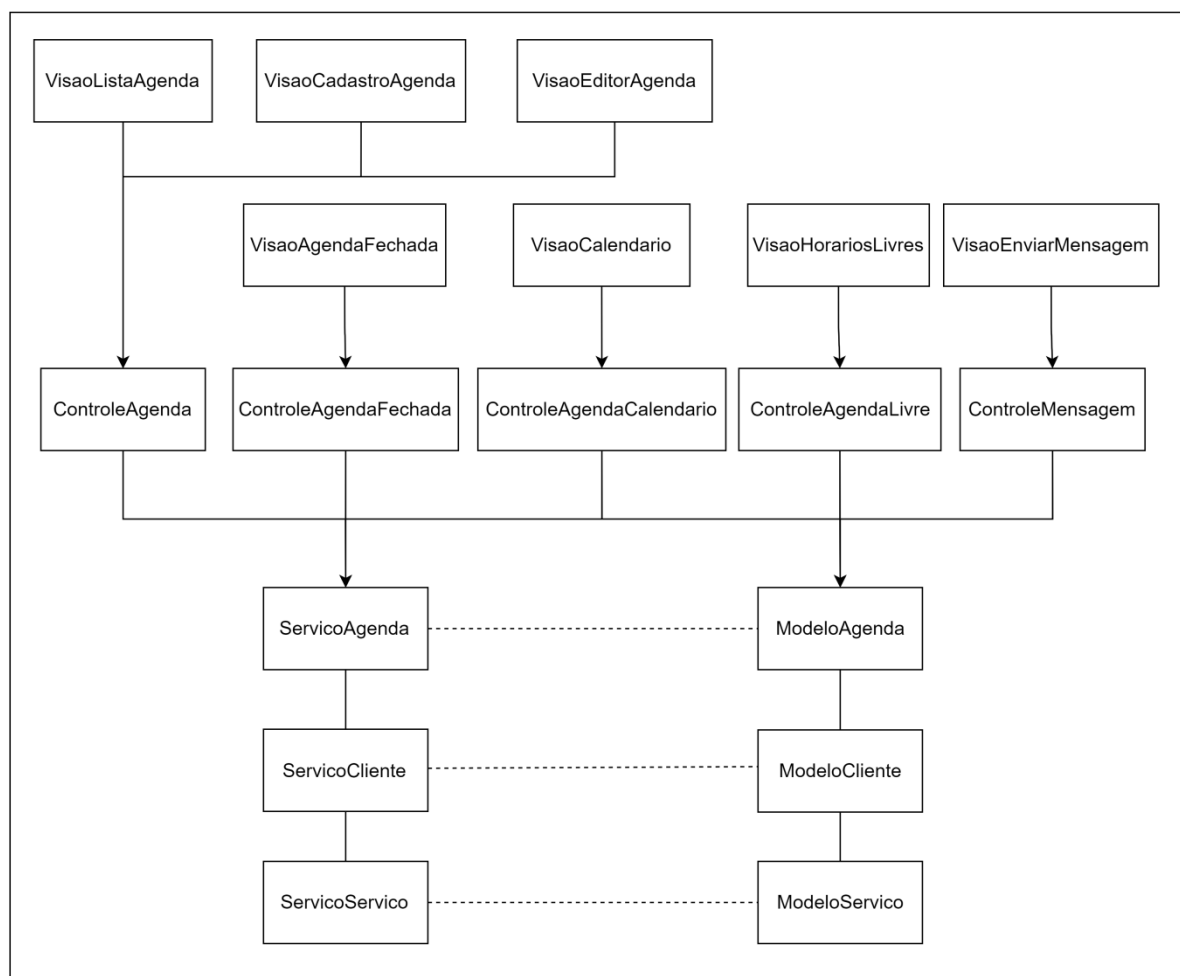
“VisaoHorariosLivres” tem a função de apresentar para o usuário os horários livres para possíveis agendamentos. O código “VisaoCalendario” tem a função de apresentar para o usuário todos os agendamentos cadastrados no sistema.

O código “ControleAgenda” processa as solicitações da camada de apresentação representadas pelos códigos “VisaoListaAgenda”, “VisaoCadastroAgenda” e “VisãoEditorAgenda”, coordena as operações e chama os métodos apropriados da camada de serviço. O código “ControleAgendaFechada” processa as solicitações da camada de apresentação representadas pelo código “VisaoAgendaFechada” e coordena as operações e chama os métodos apropriados da camada de serviço. O código “ControleAgendaCalendario” processa as solicitações da camada de apresentação representadas pelo código “VisaoAgendaCalendario” e coordena as operações e chama os métodos apropriados da camada de serviço. O código “ControleAgendaLivre” processa as solicitações da camada de apresentação representadas pelo código “VisaoHorariosLivres” e coordena as operações e chama os métodos apropriados da camada de serviço. O código “ControleMensagem” processa as solicitações da camada de apresentação representadas pelo código “VisaoEnviarMensagem” e coordena as operações e chama os métodos apropriados da camada de serviço.

O código “ServicoAgenda” tem a função de gerenciar os dados dos agendamentos, fornece métodos para adicionar agendamentos e recuperar informações de agendamentos. O código “ModeloAgenda” inclui classes ou estruturas que representam informações sobre o agendamento, como nome do cliente, nome do serviço, data, horário, etc. O código “ServicoAgenda” utiliza os códigos “ServicoCliente” e “ServicoServico” para buscar as informações do nome do cliente e do nome do serviço nos respectivos modelos.



**Figura 12 - MVCS para Agenda.**



**Fonte: Elaborado pela autora (2023).**

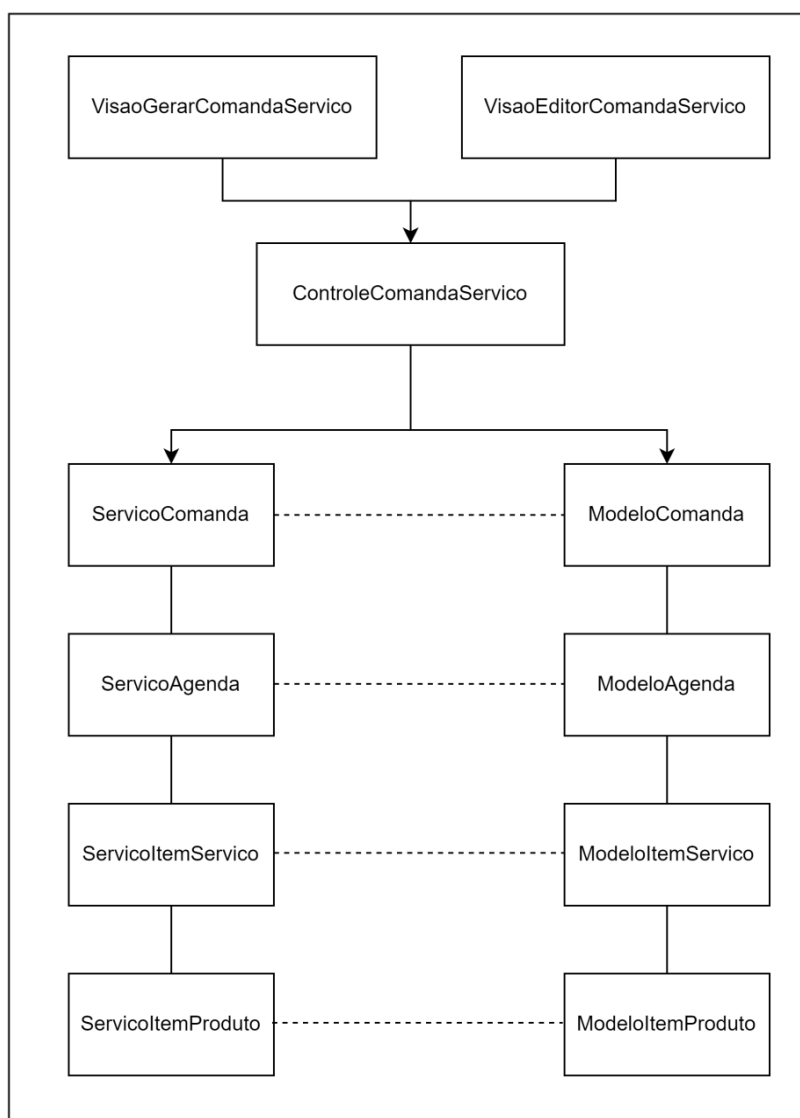
A Figura 13 apresenta o diagrama MVCS para gerenciar as informações das comandas criadas ao final de um agendamento realizado pelo usuário. A camada visão é composta por duas páginas responsáveis por apresentar informações da comanda ao usuário e permitir interações com os dados da comanda. O código “VisaoGerarComandaServico” tem a função de apresentar a comanda de serviço. O código “VisaoEditorComandaServico” tem a função de exibir a página de edição da comanda de serviço.

O código “ControleComandaServico” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. O código “ServicoComanda” tem a função de gerenciar os dados da comanda de serviço, fornece métodos para criar comandas e recuperar informações das comandas. O código “ModeloComanda” inclui classes ou estruturas

que representam informações sobre a comanda, como nome do cliente, nome do serviço, valor total, forma de pagamento, etc.

O “ServicoComanda” utiliza de outros serviços para acessar as informações necessárias para compor a comanda de serviço. Para acessar as informações do cliente e do serviço, o “ServicoComanda” utiliza o “ServicoAgenda” para obter as informações do agendamento que estão no “ModeloAgenda”. Para obter informações sobre os serviços prestados fora do agendamento, o “ServicoComanda” utiliza o “ServicoItemServico”, que acessa o “ModeloItemServico”. Caso o usuário realize uma venda na comanda de serviço, o “ServicoComanda” utiliza o “ServicoItemProduto” para acessar as informações no “ModeloItemProduto”.

**Figura 13 - MVCS para Comanda de Serviço.**



**Fonte: Elaborado pela autora (2023).**

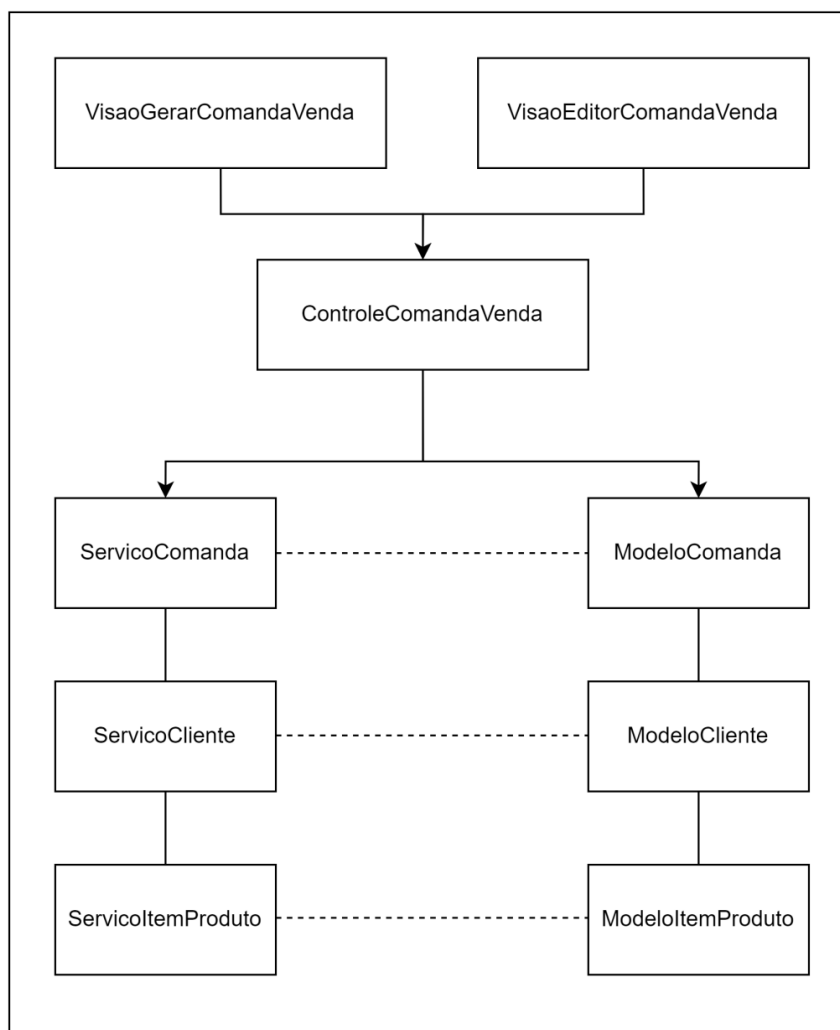
A Figura 14 apresenta o diagrama MVCS para gerenciar as informações das comandas criadas na venda de um ou mais produtos pelo usuário. A camada visão é composta por duas páginas responsáveis por apresentar informações da comanda de venda ao usuário e permitir interações com os dados da comanda. O código “VisaoGerarComandaVenda” tem a função de apresentar a comanda de venda. O código “VisaoEditorComandaVenda” tem a função de exibir a página de edição da comanda de venda.

O código “ControleComandaVenda” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da

camada de serviço. O código “ServicoComanda” tem a função de gerenciar os dados da comanda de venda, fornece métodos para criar comandas e recuperar informações das comandas. O código “ModeloComanda” inclui classes ou estruturas que representam informações sobre a comanda de venda, como nome do cliente, nome do produto, quantidade do produto, valor total, forma de pagamento, etc.

O “ServicoComanda” utiliza de outros serviços para acessar as informações necessárias para compor a comanda de venda. Para acessar as informações do cliente, o “ServicoComanda” utiliza o “ServicoCliente” para obter as informações do cliente que estão no “ModeloCliente”. Para obter informações sobre os produtos vendidos, o “ServicoComanda” utiliza o “ServicoItemProduto”, que acessa o “ModeloItemProduto”.

**Figura 14 - MVCS para Comanda de Venda de Produto.**



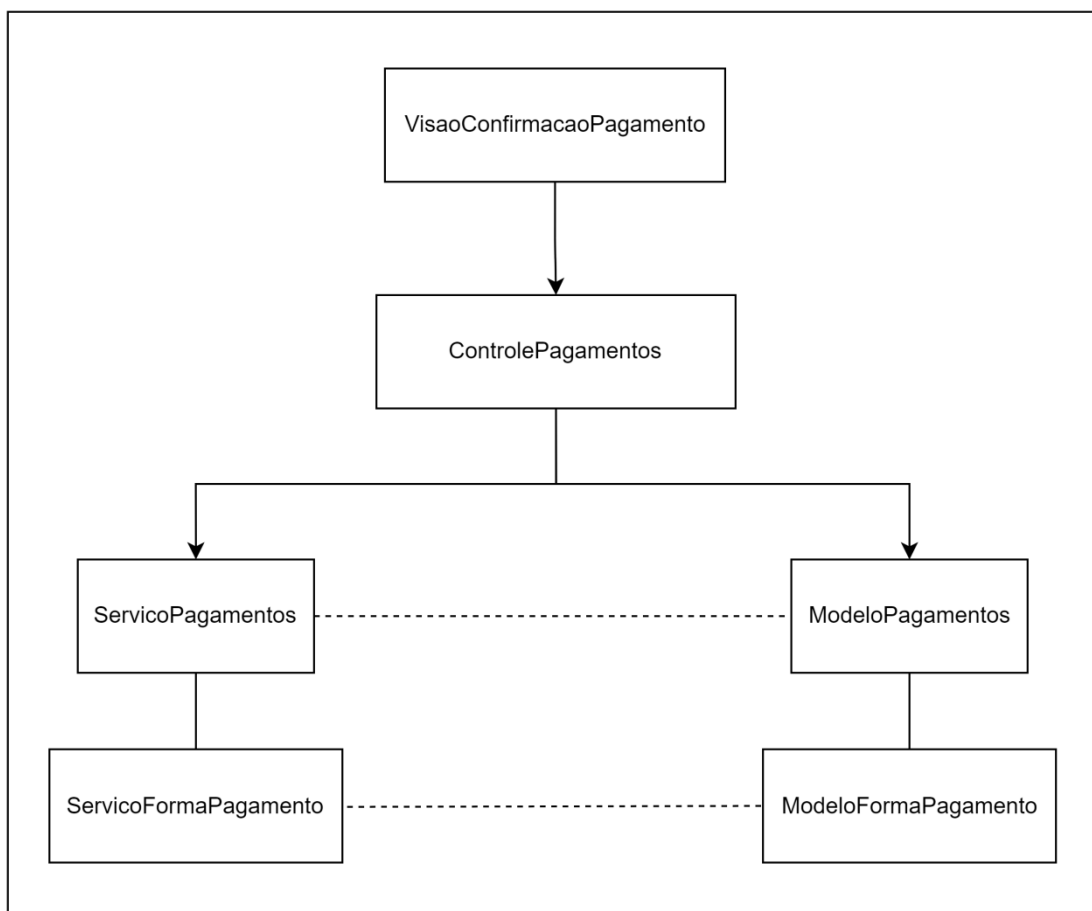
**Fonte: Elaborado pela autora (2023).**

A Figura 15 apresenta o diagrama MVCS para gerenciar as informações sobre a confirmação de pagamento das comandas geradas. A camada visão é composta por uma página responsável por apresentar informações da confirmação do pagamento e permitir interações com os dados do pagamento. O código “VisaoConfirmacaoPagamento” tem a função de apresentar as opções de pagamento aceitas pelo usuário.

O código “ControlePagamentos” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. O código “ServicoPagamentos” tem a função de gerenciar os dados do pagamento, fornece métodos para criar o usuário e recuperar informações

do pagamento. O código “ModeloPagamentos” inclui classes ou estruturas que representam informações sobre os pagamentos recebidos, como a data e o código da comanda relacionada. O código “ServicoPagamentos” utiliza o “ServicoFormaPagamento” para acessar o dado referente a escolha de pagamento no “ModeloFormaPagamento”.

**Figura 15 - MVCS para Confirmação de Pagamento.**



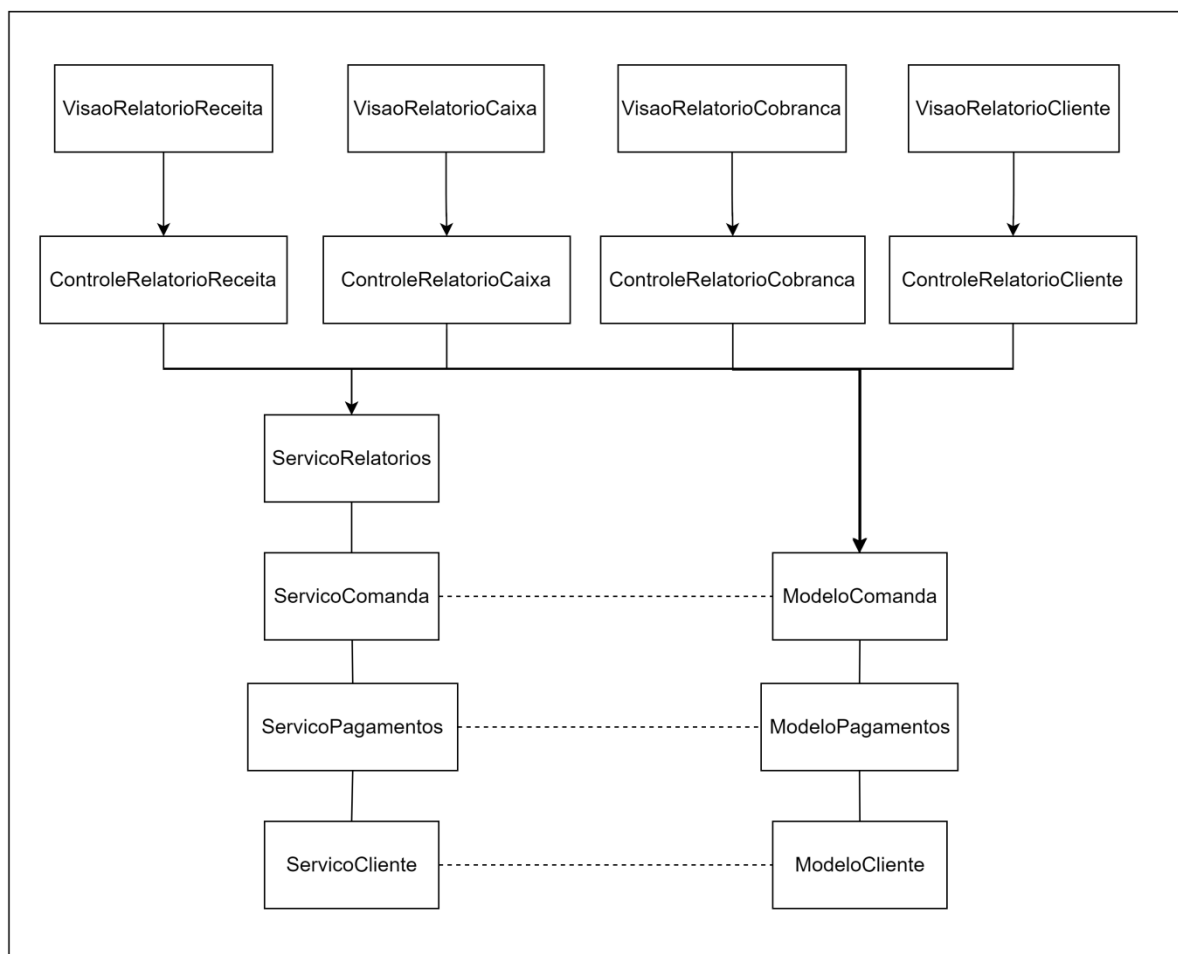
**Fonte: Elaborado pela autora (2023).**

A Figura 16 apresenta o diagrama MVCS para gerenciar as informações dos relatórios gerados pelo sistema. A camada visão é composta por quatro páginas responsáveis por apresentar informações dos relatórios financeiros e de cliente ao usuário e permitir interações com os dados das comanda de serviço e de venda de produto.

O código “VisaoRelatorioReceita” tem a função de apresentar uma série de informações para o usuário sobre a receita, despesas, lucro e relatório de serviços com mais saída. O código “VisaoRelatorioCaixa” tem a função de apresentar relatório diário do fluxo de caixa, entradas e saídas. O código “VisaoRelatorioCobranca” tem a função de apresentar uma lista dos clientes devedores. Neste, o usuário pode interagir com a tela por acessar a comando que está em aberto, ou seja, que ainda não foi paga pelo cliente. O código “VisaoRelatorioCliente” tem a função de exibir informações sobre os clientes como o nome do cliente que contribui mais com a receita do negócio.

Os códigos “ControleRelatorioReceita”, “ControleRelatorioCaixa”, “ControleRelatorioCobranca” e “ControleRelatorioCliente”, processam as solicitações das respectivas camadas de apresentação (visão), coordenam as operações e chamam os métodos apropriados da camada de serviço. Como os relatórios são uma compilação de informações que estão armazenadas em mais de um modelo, a camada de serviço, “ServicoRelatorios”, utiliza outros serviços e seus modelos. No caso, são utilizados o “ServicoComanda”, o “ServicoPagamentos”, o “ServicoCliente” e seus respectivos modelos, para recuperar informações do banco de dados e adicionar informações, como no caso do relatório de cobranças.

**Figura 16 - MVCS para os Relatórios.**



**Fonte: Elaborado pela autora (2023).**

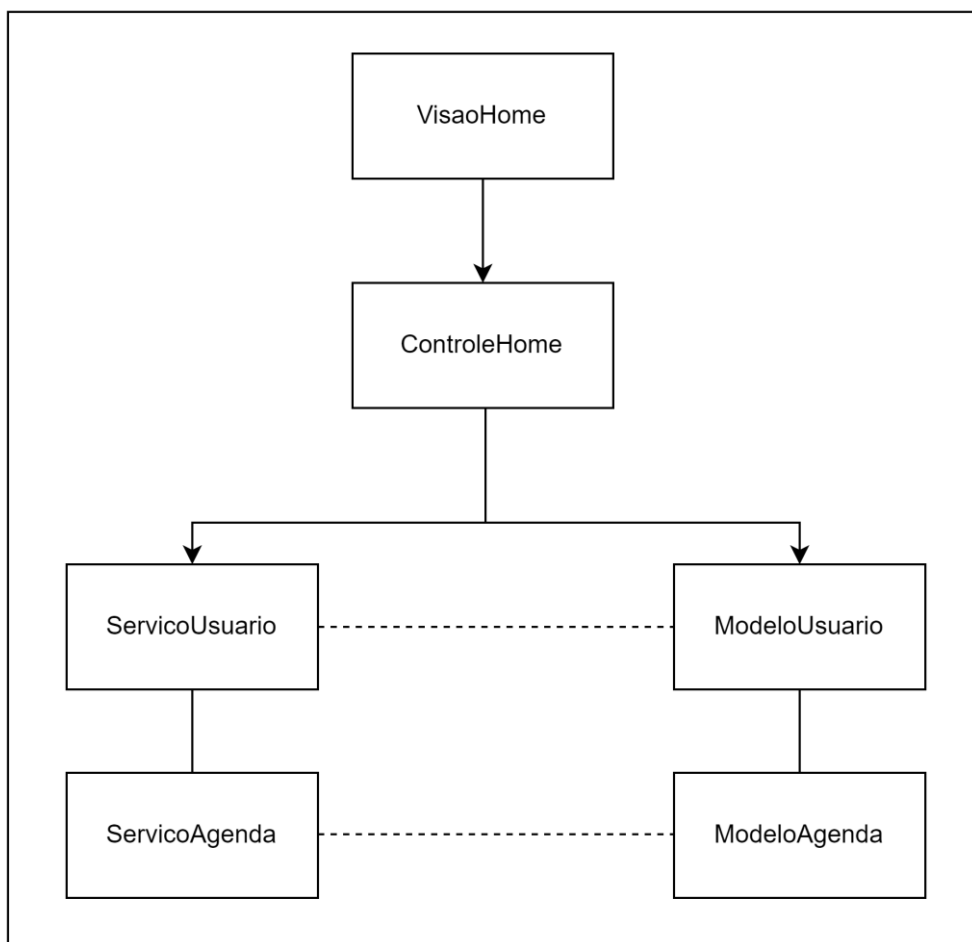
A Figura 17 apresenta o diagrama MVCS para gerenciar os componentes da tela inicial, ou *Home*. A camada visão é composta por uma página responsável por apresentar a tela inicial ao usuário e permitir interações com seus botões. O código “VisaoHome” tem a função de apresentar a tela inicial do aplicativo. A *Home* apresenta algumas informações do usuário, como nome e foto, e botões que permitem o usuário entrar em outras partes do sistema.

O código “ControleHome” processa as solicitações da camada de apresentação (visão), coordena as operações e chama os métodos apropriados da camada de serviço. Recebe as entradas do usuário, como cliques em botões, e decide como responder a essas ações. O “ControleHome” acessa componentes de



outros serviços como “ServicoUsuario” e “ServicoAgenda”, e seus respectivos modelo existentes.

**Figura 17 - MVCS para a Tela Inicial (Home).**



**Fonte: Elaborado pela autora (2023).**

### 3.3 BANCO DE DADOS

Banco de dados é uma coleção de dados relacionados, logicamente coerentes, com algum significado inerente, que representa algum aspecto do mundo real e construído para uma finalidade específica. Esses dados são organizados e distribuídos em tabelas, que estão relacionadas e comunicam-se umas com as outras, armazenadas em um arquivo. Dados são objetos conhecidos que podem ser armazenados e possuem um significado implícito (MACHADO, 2020, p.19). Os dados são “coisas” não processadas, estáticas, muitas vezes imutáveis (ex: nome, cidade, peso, preço, medida, data de nascimento, etc.), que quando processados, com algum propósito, geram informações.

#### 3.3.1 Modelagem Lógica do Banco de Dados

O modelo lógico descreve a estrutura de um banco de dados, as formas de como os dados estarão armazenados e os relacionamentos entre as entidades (tabelas). Determina a estrutura física de armazenamento de dados descrevendo seus campos (colunas) com tipos e tamanhos, chaves primárias e secundárias (NUNES e MOURA, 2018, p. 41). Um modelo lógico tem três componentes principais: entidade, relacionamentos e atributos. Uma entidade é um agrupamento de informações sobre um determinado lugar, objeto ou pessoa, de que se tem interesse guardar informações a seu respeito. Relacionamento indica a forma como as entidades se associam. Os atributos são as informações básicas que qualificam uma entidade e descrevem seus elementos e características.

A Figura 18 apresenta o modelo lógico do banco de dados para o projeto do aplicativo “Agenda App” com as entidades (tabelas), atributos e relacionamentos. Os relacionamentos entre as entidades definem as regras do negócio.

A tabela “Imagens” possui um atributo imagem, do tipo BLOB, e um identificador (chave primária), e se relaciona com as tabelas “Usuario”, “Cliente” e “Produto” de forma que cada imagem corresponde a um único usuário, produto ou

cliente cadastrado. Da mesma forma, um registro de usuário, produto ou cliente possui apenas uma imagem correspondente na tabela “Imagens”.

A tabela “Usuario” possui os atributos pertencentes ao usuário do sistema, como nome, e-mail, senha de acesso e o identificador (chave-primária). Esses dados são acessados pelo sistema quando o usuário cria a conta, realiza o *login* ou quando modifica as informações de perfil e conta.

A tabela “Cliente” possui os atributos pertencentes aos clientes cadastrados pelo usuário, como nome e telefone, e se relaciona com a tabela “Agendamento”, de forma que um cliente pode estar agendado em vários agendamentos, mas um agendamento só pode ter um cliente agendado.

A tabela “Sevico” possui os atributos pertencentes aos serviços prestados pelo usuário, como nome e preço, e se relaciona com a tabela “Agendamento”, de forma que um serviço pode estar em mais de um agendamento, mas um agendamento é realizado para apenas um serviço.

A tabela “Agendamento” possui os atributos relacionados ao agendamento de um serviço prestado pelo usuário, como o nome do cliente e o nome do serviço, e se relaciona com a tabela “Comanda” de forma que a ação de finalização de um agendamento gera uma comanda, mas uma comanda não necessariamente pertence a um agendamento.

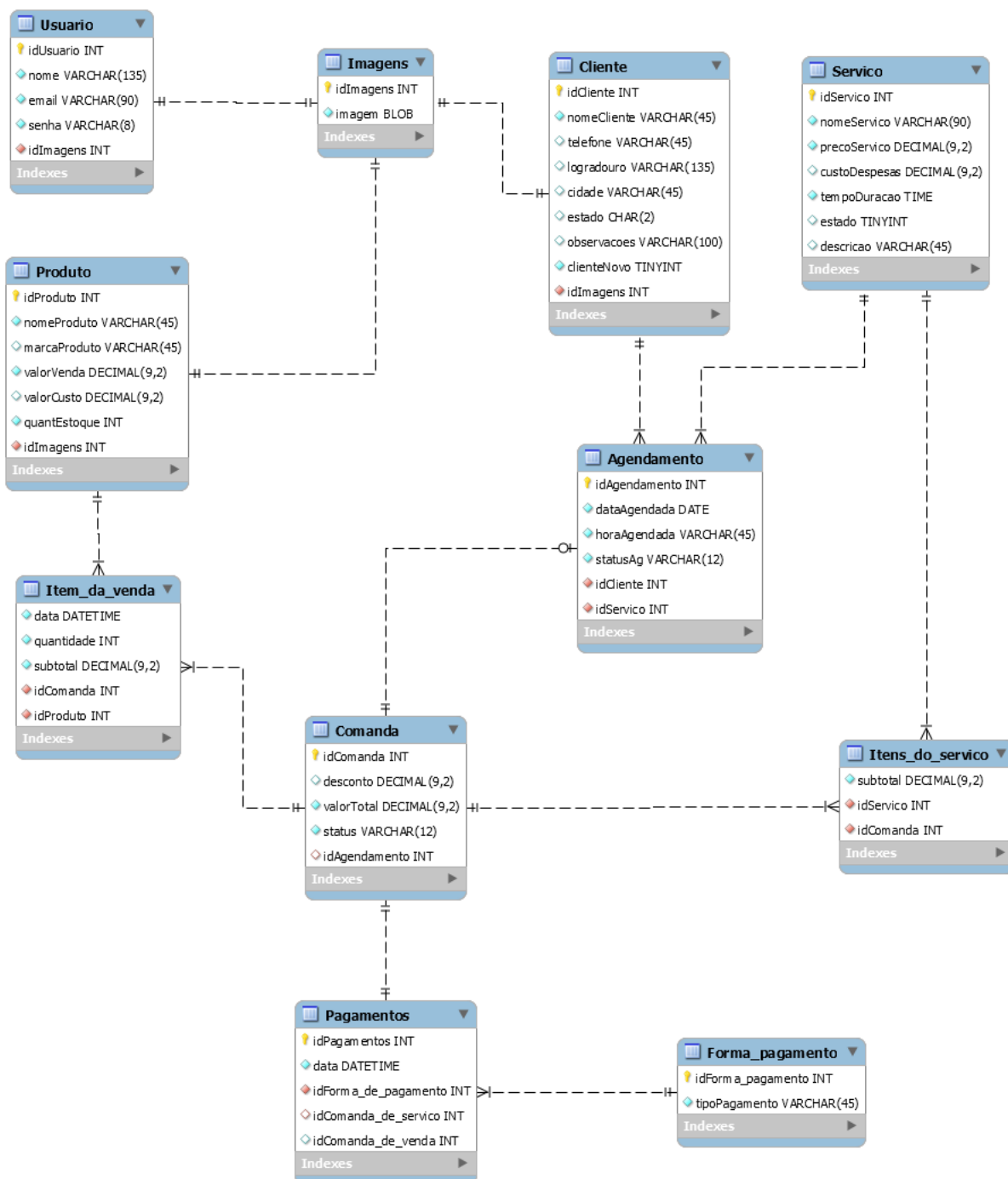
A tabela “Produto” possui atributos relacionados com os produtos comercializados pelo usuário, como nome, marca e valor do produto, e se relaciona com a tabela “Itens\_da\_Venda” de forma que um item da venda pode se referir a exclusivamente um produto e um produto, por sua vez, pode ser adicionado no mínimo nenhum e no máximo muitos itens da venda.

A tabela “Comanda” possui os atributos relacionados com o fechamento de um serviço por meio de um agendamento, ou por meio da venda de um produto. No caso de um fechamento de serviço, a tabela “Comanda” se relaciona com a tabela “Itens\_do\_servico” de forma que ao fechar um agendamento, é possível inserir outro serviço na comanda, como também é possível inserir a venda de um produto. No caso da venda de produtos, a tabela “Comanda” se relaciona com a tabela “Itens\_da\_venda, de forma que uma comanda pode incluir a venda de no mínimo um

e no máximo vários produtos. A tabela “Comanda” também se relaciona com a tabela “Pagamentos” de forma que cada comanda é finalizada por meio da confirmação do pagamento, e cada pagamento se refere a uma única comanda.

A tabela “Pagamentos” possui os atributos relacionados ao pagamento das comandas, como a data de pagamento, a forma de pagamento e o código de identificação da comanda (chave estrangeira). Essa tabela se relaciona com a tabela “Forma\_pagamento” de forma que, um pagamento só pode ser realizado por meio de uma forma de pagamento, mas uma forma de pagamento pode estar presente em vários registros de pagamentos.

Figura 18 - Modelo lógico do banco de dados.



Fonte: Elaborado pela autora (2023).

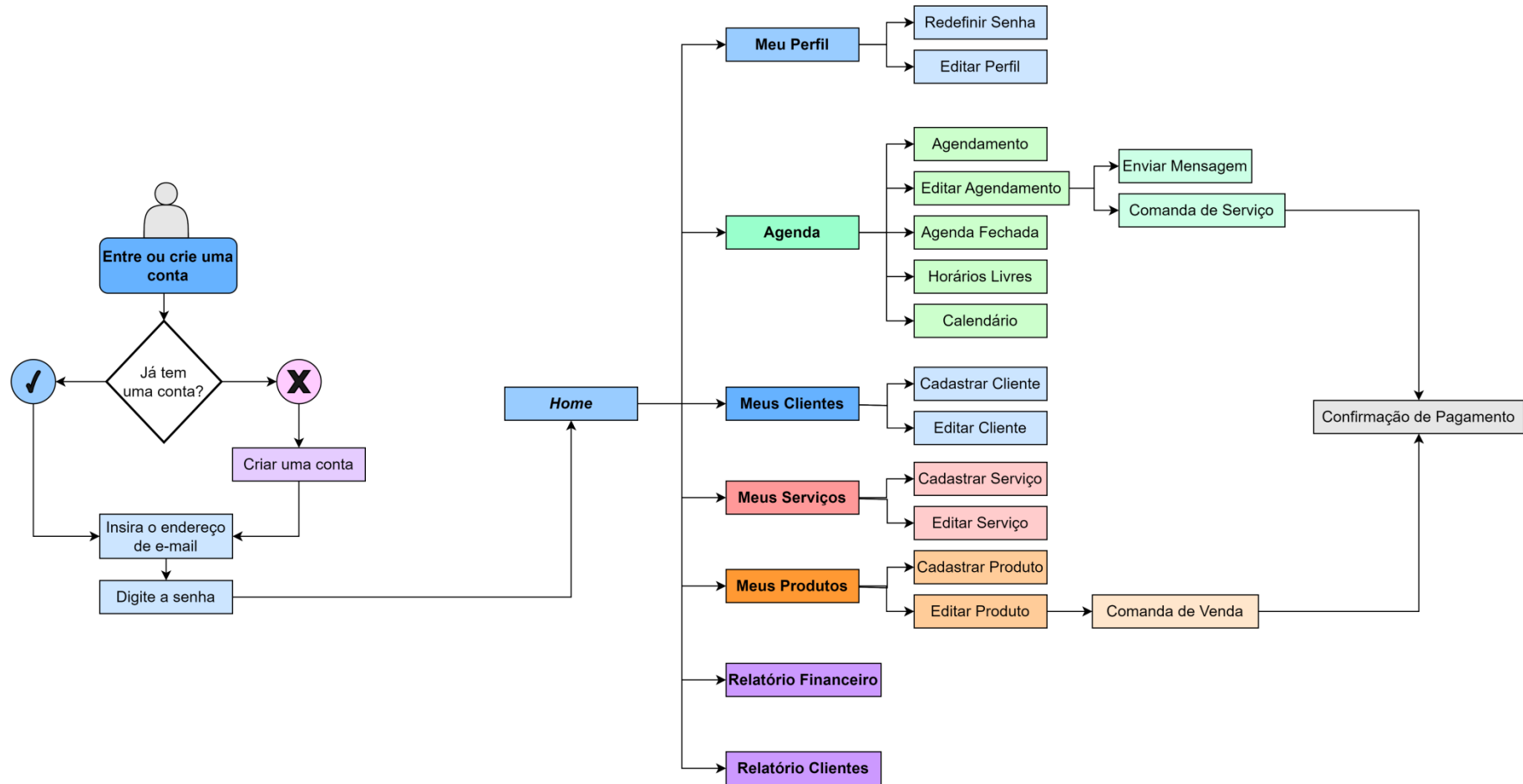
## 4 PROTÓTIPO

Um protótipo é um modelo ou representação inicial de um produto, sistema ou conceito criado para avaliar e testar suas características e funcionalidades antes de iniciar a implementação. A etapa da criação do protótipo é importante no design de aplicativos móveis, pois ajuda a visualizar a experiência do usuário e a estrutura do aplicativo. Nessa etapa as alterações podem ser realizadas sem precisar despende muito tempo de trabalho. É uma etapa que precede o desenvolvimento do código.

Os protótipos de baixa fidelidade são geralmente esboços simples, feitos com papel e caneta, usados nas fases iniciais do projeto. Possuem baixo nível de detalhamento e baixo custo. O protótipo de alta fidelidade é mais parecido com o produto final, possui funcionalidades completas e definição clara da navegação. Pode ser usado para explorar o aplicativo, para testá-lo, ou para vender uma ideia. São desenvolvidos com o auxílio de softwares para criação de interfaces.

A Figura 19 apresenta o diagrama de fluxo de tela para o protótipo do aplicativo “Agenda App”. Esse diagrama é uma representação visual que descreve como as diferentes telas do aplicativo móvel estão conectadas e como o usuário pode navegar entre elas. Dessa forma, o usuário pode compreender qual “caminho” deve seguir para acessar determinada tela ou objeto.

Figura 19 - Diagrama de fluxo de tela (mapa das telas).



Fonte: Elaborado pela autora (2023).

O protótipo de alta fidelidade a seguir apresenta a proposta do aplicativo “Agenda App”. As telas foram criadas no *FlutterFlow*, que é uma plataforma de desenvolvimento visual. Todo o projeto do aplicativo está representado da Figura 20 até a Figura 39.

A Figura 20 apresenta a tela *Splash*. Essa é a primeira tela que é apresentada para o usuário, enquanto o aplicativo carrega. A tela mostra o logotipo do sistema contendo o nome do aplicativo “Agenda App”.

**Figura 20 - Captura da *Splash Screen*.**



**Fonte: Elaborado pela autora (2023).**



A Figura 21 apresenta as telas de *Login* e a de Criação de Conta. Na tela de *Login* (Fig. 11-a), o usuário que já possui uma conta cadastrada, poderá se autenticar preenchendo os campos de “E-mail” e “Senha”, e acessar o sistema clicando no botão “Login”. Caso o usuário ainda não tenha uma conta cadastrada é possível criar uma ao acessar o atalho “Criar uma conta”. No caso do usuário ter esquecido sua senha, poderá recuperá-la pelo atalho “Esqueceu a senha?”. Na tela de Criação de Conta (Fig. 11-b), o usuário cria a sua conta ao preencher, obrigatoriamente, o formulário com os campos de “E-mail”, “Senha” e “Confirmação a senha”, e ao clicar no botão “Crie sua conta”.

Figura 21 - Captura das telas de *Login* (a) e de Criação de Conta (b).

As imagens mostram duas telas de interface de usuário para o sistema 'Agenda'. A tela (a) é a tela de Login, com o título 'Bem-vindo!' e o subtítulo 'Use o formulário abaixo para acessar a sua conta.'. Ela contém campos de entrada para 'E-mail' e 'Senha', um link 'Esqueceu a senha?' e um botão 'Login'. A tela (b) é a tela de Criação de Conta, com o título 'Iniciar' e o subtítulo 'Use o formulário abaixo para começar.'. Ela contém campos de entrada para 'E-mail', 'Senha' e 'Confirme a senha', um botão 'Crie sua conta' e um link 'Já tem uma conta? Log In'. Ambas as telas possuem o logo 'Agenda' no topo.

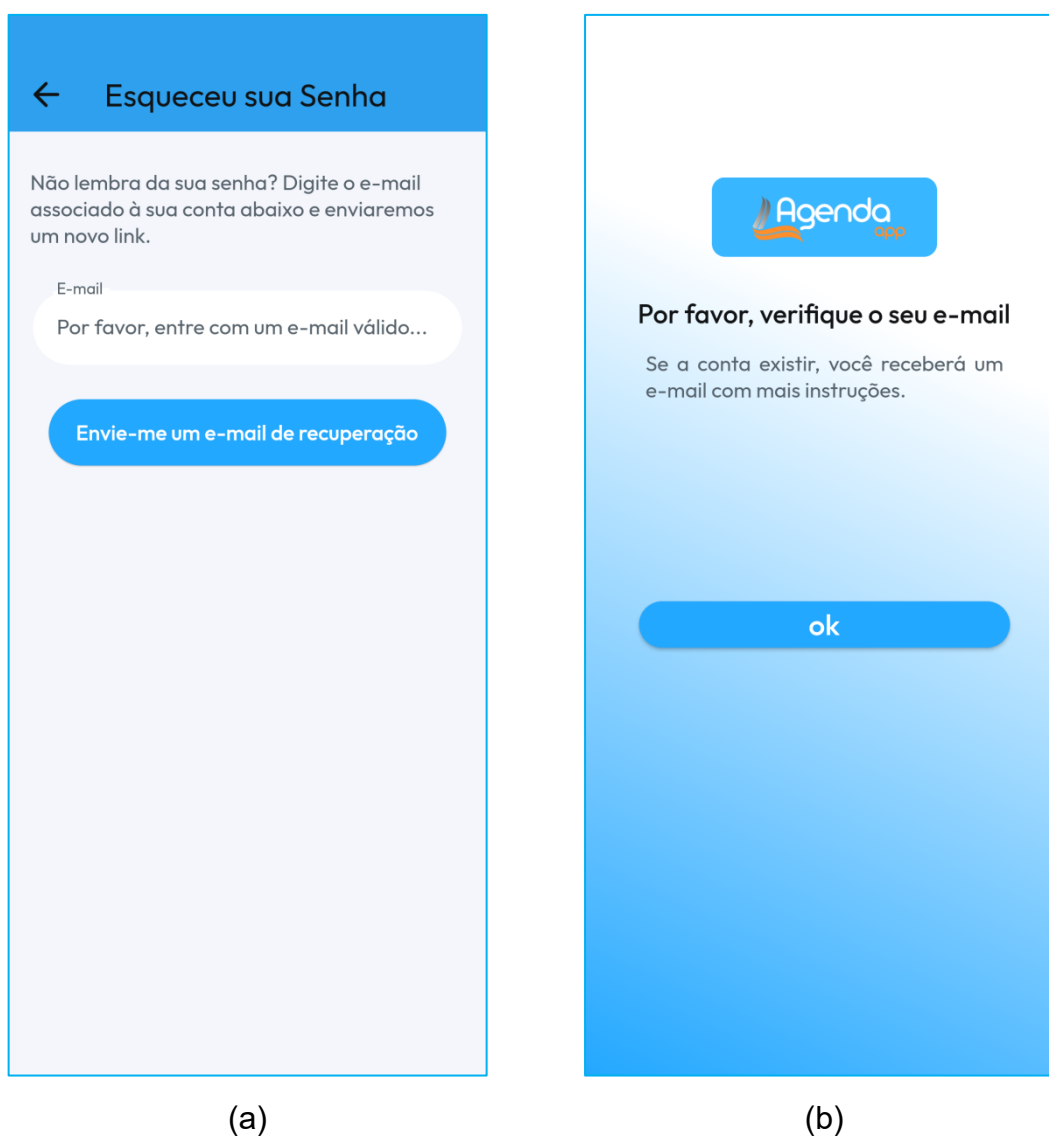
(a)

(b)

Fonte: Elaborado pela autora (2023).

A Figura 22 apresenta as telas relacionadas com a recuperação da senha do usuário. A tela “Esqueceu sua senha” (Fig. 12-a) orienta o usuário, que deseja recuperar sua senha de acesso, a digitar o e-mail cadastrado no momento de criação da conta, para que o sistema lhe envie um e-mail com um link para recuperação da senha. Assim que o usuário clica no botão “Envie-me um e-mail de recuperação”, o sistema apresenta a tela da Figura 12 (b), que solicita que o usuário verifique seu e-mail. Ao clicar no botão “ok”, o usuário é direcionado para a tela de “Login”.

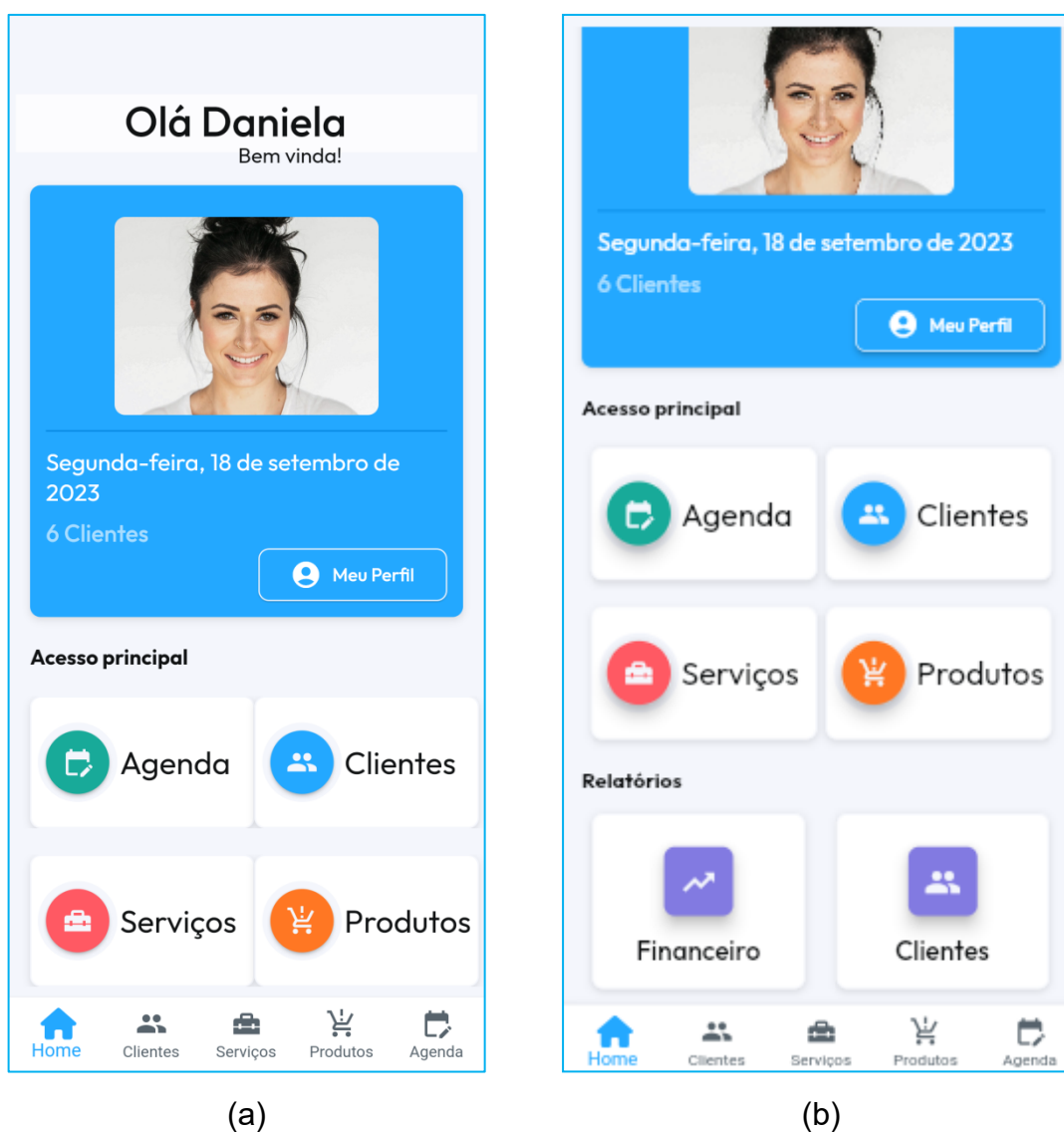
**Figura 22 - Captura da tela Esqueceu sua Senha (a) e de verificação (b).**



Fonte: Elaborado pela autora (2023).

A Figura 23 apresenta a tela “Home”. Essa é a tela inicial do sistema, que será exibida quando o usuário (autenticado) inicia o aplicativo. No topo da tela (Fig. 13-a) é exibida uma mensagem de boas vindas personalizada, com o nome do usuário. Logo abaixo, no container de fundo azul, são exibidas: a foto do usuário, informações da data atual, o número de clientes agendados para o dia e o botão “Meu Perfil”. Este botão direciona o usuário para a tela “Meu Perfil” (Fig. 14), onde é possível realizar algumas configurações da conta, como carregar uma foto de perfil do usuário.

Figura 23 - Captura da tela *Home* (a) e continuação da tela *Home* (b).



Fonte: Elaborado pela autora (2023).

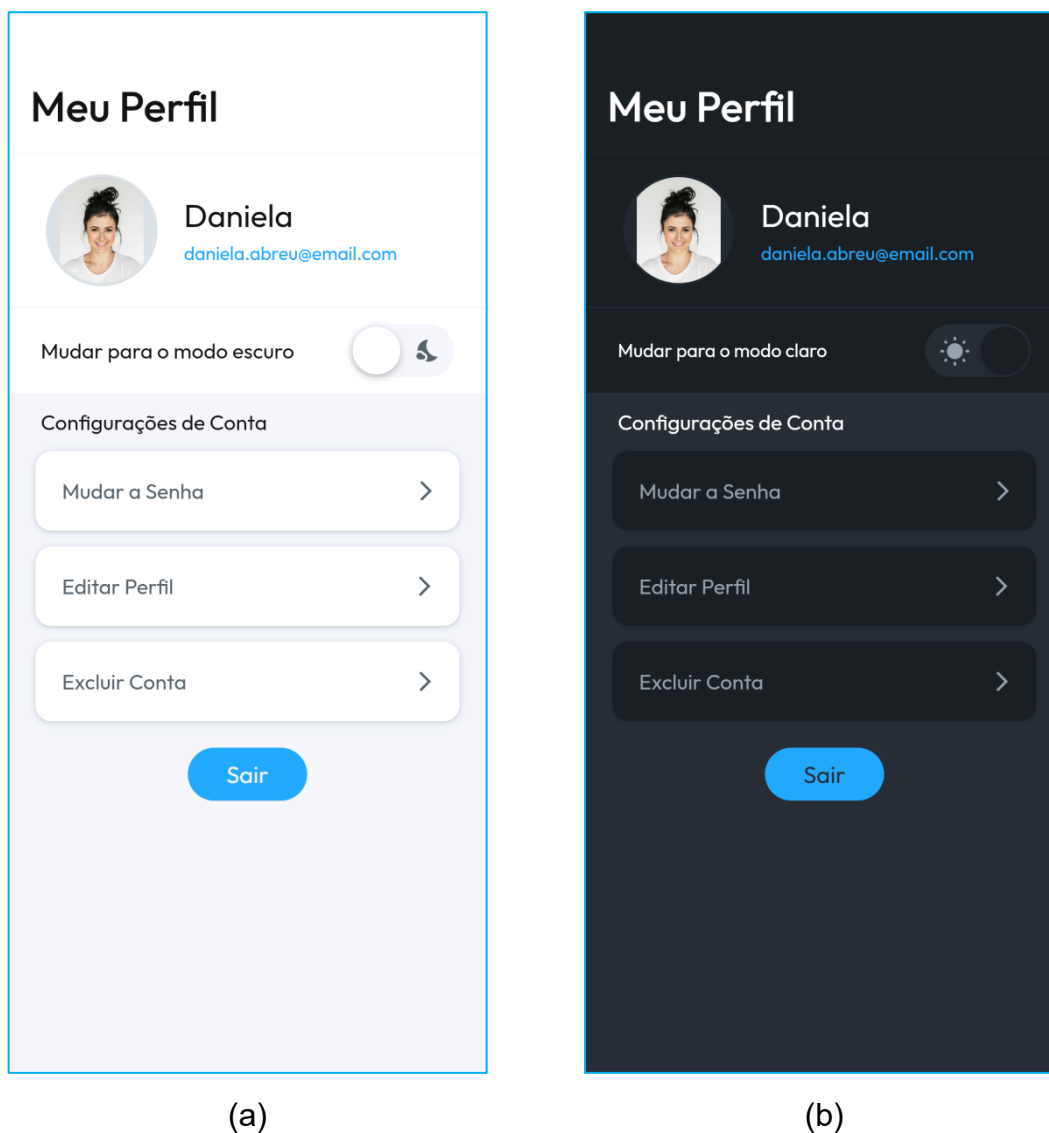
A seção “Acesso principal” é composta por quatro botões que vão para as telas que dão suporte ao negócio do usuário. Ao clicar no botão “Agenda” o usuário é direcionado para a tela “Agenda” (Fig. 37), onde estão listados os agendamentos a partir da data atual. O botão “Clientes” direciona o usuário para a tela “Meus Clientes” (Fig. 28), que contém uma lista de clientes cadastrados pelo usuário. O botão “Serviços” direciona o usuário para a tela “Meus Serviços” (Fig. 31), que contém uma lista dos serviços cadastrados pelo usuário. O botão “Produtos” direciona o usuário para a tela “Meus Produtos” (Fig. 34), onde estão listados os produtos cadastrados pelo usuário e que podem ser comercializados pelo mesmo.

Na seção “Relatórios” estão os botões que permitem o usuário acessar alguns relatórios (Fig. 23-b). Ao clicar no botão “Financeiro”, o usuário é direcionado para a tela “Relatório Financeiro” (Fig. 47), onde o usuário pode visualizar algumas informações sobre a movimentação financeira da sua atividade. O botão “Clientes” direciona o usuário para a tela “Relatório Clientes” (Fig. 49), que apresenta ao usuário algumas informações sobre seus clientes.

A barra de navegação inferior permite que o usuário navegue com facilidade entre as telas principais do aplicativo como “Home”, “Meus Clientes”, “Meus Serviços”, “Meus Produtos” e “Agenda”, por meio dos botões Home, Clientes, Serviços, Produtos e agenda, respectivamente. O botão referente à tela em que o usuário está navegando fica em destaque na cor azul.

A Figura 24 apresenta a tela “Meu Perfil”. Essa tela permite que o usuário realize algumas configurações no seu perfil, como mudar o modo de apresentação das telas (claro e escuro), alterar a senha, editar o perfil e deletar (encerra) a conta. O botão “Mudar a Senha” direciona o usuário para a tela “Redefinir Senha” (Fig. 26-a). O botão “Editar Perfil” direciona o usuário para a tela “Editar Perfil” (Fig. 26-b). O botão “Excluir Conta” desativa a conta do usuário, mas antes de concluir a ação, o sistema apresenta uma janela com uma mensagem de confirmação da ação, perguntando se o usuário deseja realmente excluir sua conta (Fig. 25).

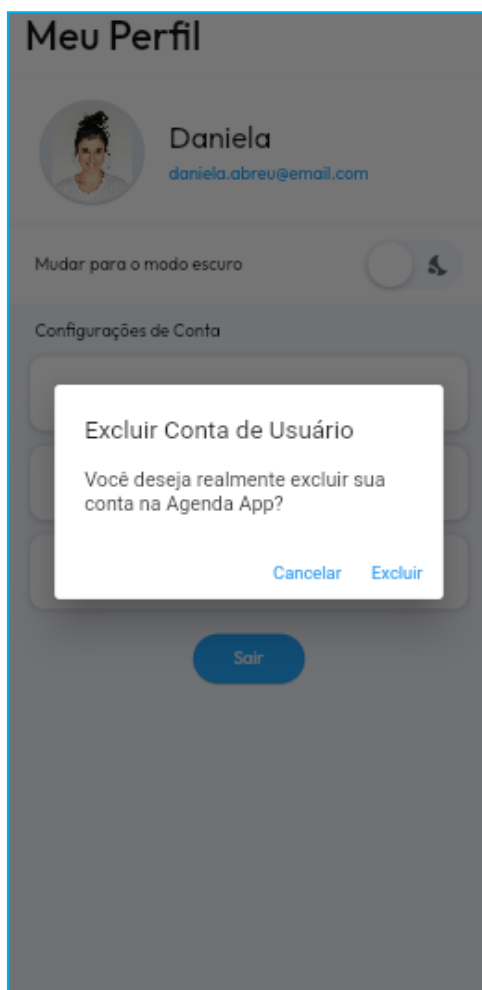
Figura 24 - Captura das telas Meu Perfil modo claro (a) e Meu Perfil modo escuro (b).



Fonte: Elaborado pela autora (2023).

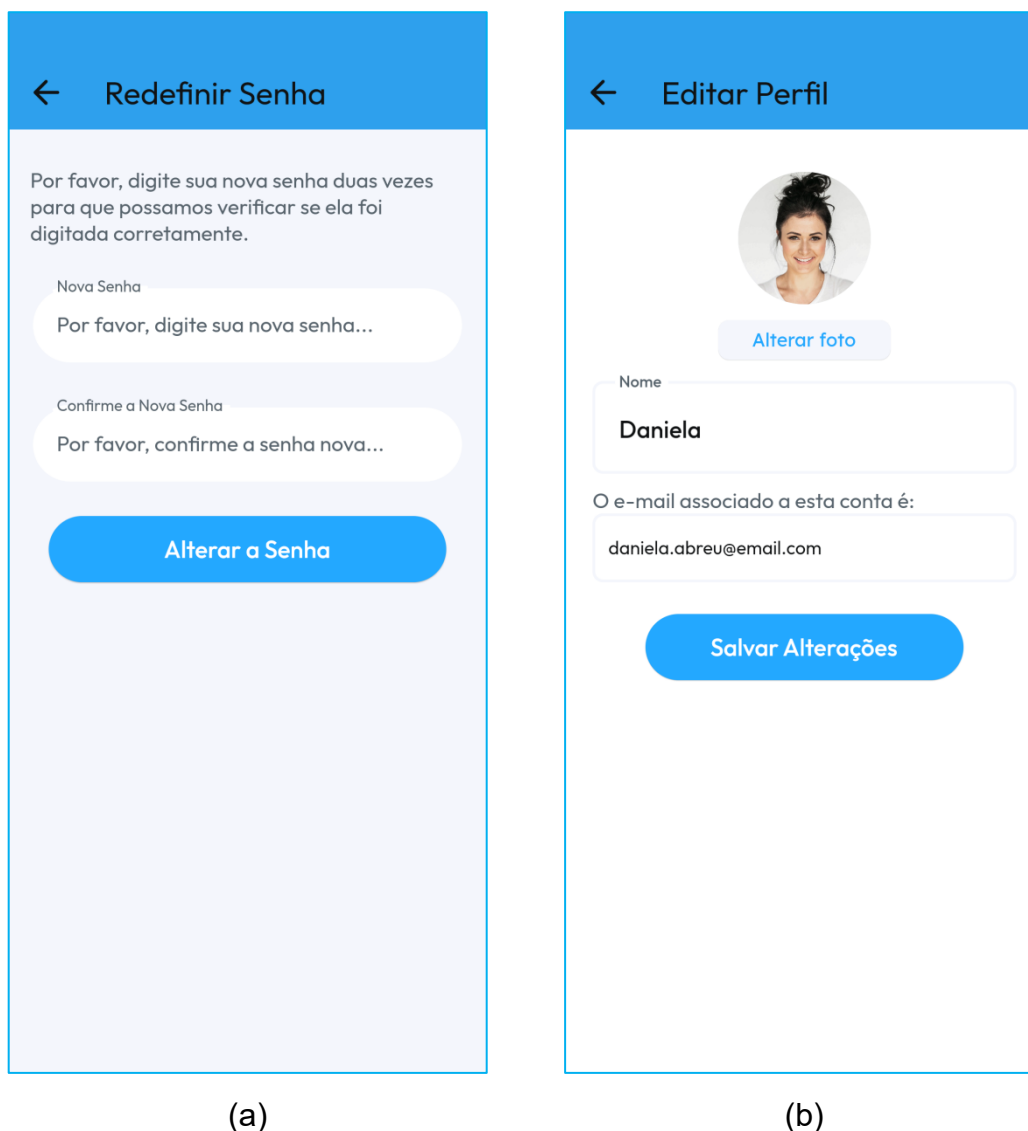
Para alterar o tema de apresentação das telas entre o modo escuro e claro, o usuário pode deslizar o botão *switch* (troca), que está do lado direito das frases “Mudar para o modo escuro” (Fig. 24-a) e “Mudar para o modo claro” (Fig. 24-b). Ao clicar no botão “Sair”, o usuário retorna para a tela “*Home*”.

Figura 25 - Captura da janela Excluir Conta de Usuário.



Fonte: Elaborado pela autora (2023).

A Figura 26 apresenta as telas “Redefinir Senha” e “Editar Perfil”. Na tela “Redefinir Senha” (Fig. 26-a) o usuário deve digitar a senha no campo “Nova Senha” e digitar novamente a mesma senha no campo “Confirme a Nova Senha”. O sistema vai conferir se as duas senhas digitadas pelo usuário são iguais. Caso sejam iguais os campos ficarão preenchidos, caso contrário, uma mensagem vai avisar o usuário que ele deverá digitar novamente as duas senhas. Ao clicar no botão “Alterar a Senha”, o sistema faz a troca de senha e retorna para a tela “Login”. O usuário pode voltar para a tela anterior, no caso a tela “Meu Perfil”, clicando no ícone seta localizado na barra de aplicativos superior, na parte superior da tela “Redefinir Senha”.

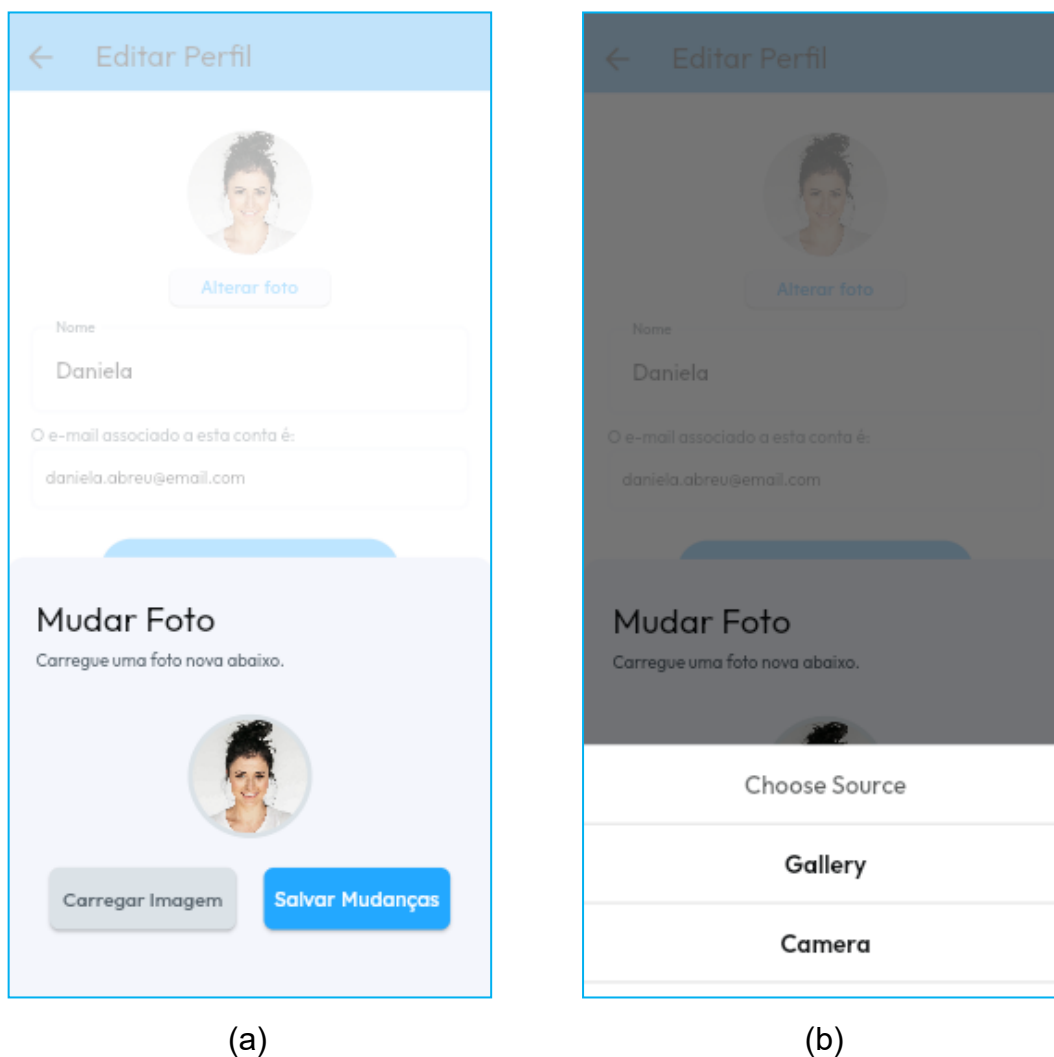
**Figura 26 - Captura das telas Redefinir Senha (a) e Editar Perfil (b).**

Fonte: Elaborado pela autora (2023).

Na tela “Editar Perfil” (Fig. 26-b) o usuário poderá digitar o nome que vai aparecer na tela “Home”, no campo “Nome” e o e-mail associado à conta do usuário no campo abaixo. Para alterar a foto ou carregar uma foto, quando pela primeira vez, o usuário pode clicar no botão “Alterar foto”. Ao clicar neste botão, o sistema apresenta uma janela na parte inferior da tela com as opções de “Carregar Imagem” e “Salvar Alterações”, em forma de botões (Fig. 27-a). Ao clicar no botão “Carregar Imagem” o sistema apresenta outra janela, sobre a janela anterior, com as opções de carregar a imagem da galeria de imagens (*Gallery*) ou pela câmera do próprio dispositivo (*Camera*) (Fig. 27-b). Assim que o usuário seleciona uma imagem por uma dessas opções, volta para a mensagem da tela anterior (Fig. 27-a), onde a

imagem é carregada e o usuário pode salvar a alteração clicando no botão “Salvar Mudanças”. Então a nova imagem carregada é apresentada acima do botão “Alterar foto” na tela “Editar Perfil”.

**Figura 27 - Captura da tela Editar Perfil: ao clicar no botão “Alterar foto” (a) e ao clicar no botão “Carregar Imagem” (b).**



Fonte: Elaborado pela autora (2023).

Ao clicar botão “Salvar as Alterações” na tela “Editar Perfil” (Fig. 26-b), o sistema salva todas as informações inseridas pelo usuário. O usuário pode voltar para a tela anterior, no caso a tela “Meu Perfil”, clicando no ícone seta localizado na barra de aplicativos superior, na parte superior da tela “Editar Perfil”.



A Figura 28 apresenta a tela “Meus Clientes”. Nessa mesma tela, o usuário pode visualizar duas listas de clientes diferentes. Ao clicar na aba “Todos”, o usuário pode visualizar todos os clientes cadastrados (Fig. 28-a) e na aba “Novos”, o usuário pode visualizar os clientes cadastrados recentemente (Fig. 28-b).

**Figura 28 - Captura da tela Meus Clientes: Todos (a) e Novos (b).**



**Fonte: Elaborado pela autora (2023).**

Para cada cliente o sistema carrega a imagem, o nome e a cidade onde o cliente mora. As listas dos clientes seguem um padrão de organização por ordem alfabética, para facilitar a busca pelo cliente. O usuário pode escolher qual lista ele quer visualizar ao clicar na aba correspondente. Para cadastrar um cliente,

primeiramente, o usuário deverá escolher em qual das duas listas o cliente deverá ser visualizado, “Todos” ou “Novos”, e clicar no botão com o sinal de mais, que fica fixo na parte inferior direita da tela (fundo de cor laranja). Ao clicar neste botão, o sistema apresenta ao usuário a tela “Cadastrar Cliente” (Fig. 29).

Para acessar ou editar as informações do cliente, o usuário pode clicar em qualquer região da tela que representa aquele cliente (foto, nome), inclusive no ícone de editar, representado por uma caneta na cor azul, na extremidade oposta do nome do cliente, que o sistema apresenta para o usuário a tela “Editar Cliente” (Fig. 30-a).

O usuário pode voltar para a tela anterior, no caso a tela “Home”, clicando no ícone seta localizado na barra de aplicativos superior da tela “Meus Clientes”, ou navegar pelos botões na barra de navegação inferior.

A Figura 29 apresenta a tela “Cadastrar Cliente”. Nessa tela o usuário pode preencher todos os campos com informações do cliente como “Nome”, “Telefone”, “E-mail”, “Endereço”, “Cidade” e “Observações”. O campo “Estado” é preenchido ao selecionar a opção desejada clicando no ícone de seta para baixo. O campo “Nome” deve ser preenchido obrigatoriamente pelo usuário para cadastrar o cliente no sistema.

**Figura 29 - Captura da tela Cadastrar Cliente.**

×

Cadastrar Cliente

Adicionar foto

Nome

Telefone

E-mail

Endereço

Cidade

Estado ▾

Observações

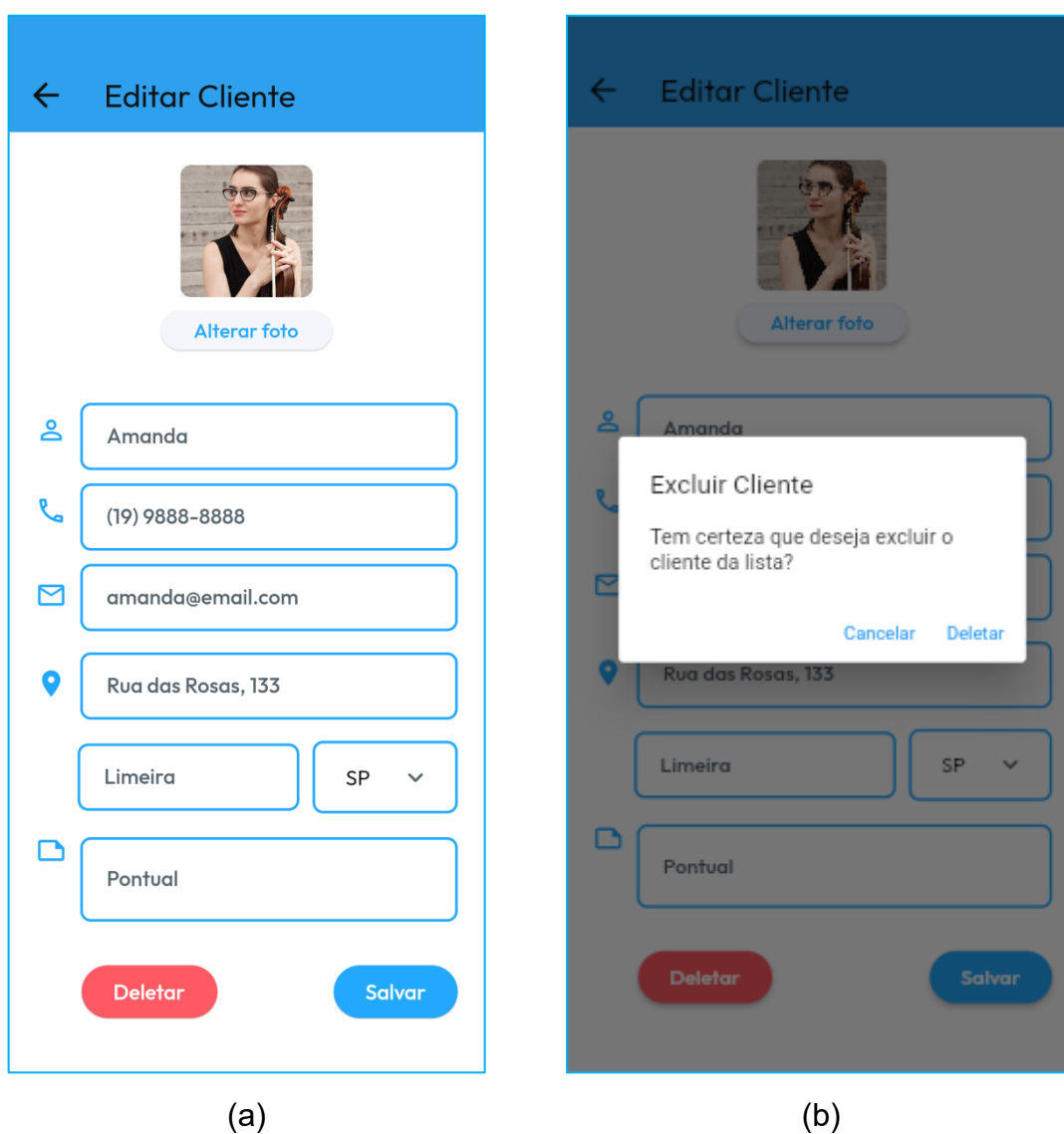
Salvar

Fonte: Elaborado pela autora (2023).

O usuário pode adicionar uma foto do cliente ao clicar no botão “Adicionar foto”. Esse botão abre uma janela com as opções de carregar uma foto da galeria de fotos ou tirar uma foto com a câmera de dispositivo, como o exemplo apresentado na Figura 27. O botão “Salvar” salva as informações do cliente no sistema e retorna para a tela “Meus Clientes”, onde o usuário pode visualizar o cliente cadastrado na lista de clientes. Caso o usuário preencha os campos, mas desista de cadastrar o cliente pode clicar no ícone em forma de “x” na barra superior da tela. Dessa forma, o sistema não guardará nenhuma informação inserida nos campos e retorna para a tela “Meus Clientes”.

A tela “Editar Cliente” (Fig. 30-a) apresenta ao usuário os campos preenchidos com as informações do cliente selecionado na tela “Meus Clientes”. Por meio da tela “Editar Cliente”, o usuário pode alterar as informações do cliente, como também preencher alguns campos que ficaram sem informação no momento de cadastrar o cliente, na tela “Cadastrar Cliente” (Fig. 29). O usuário pode alterar a foto do cliente clicando no botão “Alterar foto”. Para salvar as modificações o usuário deve clicar no botão “Salvar”.

Figura 30 - Captura da tela Editar Cliente (a) e janela de mensagem Excluir Cliente (b).



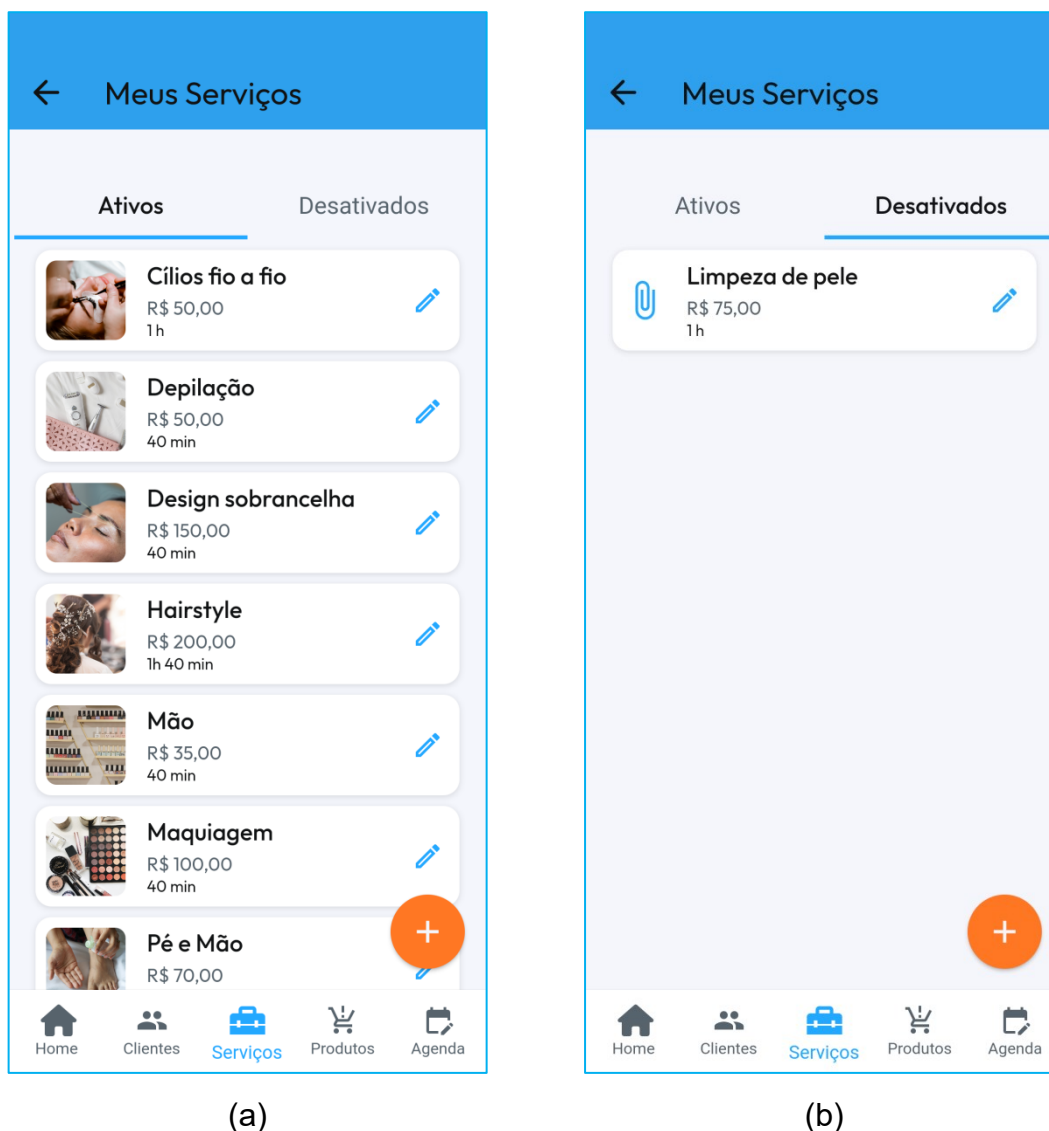
Fonte: Elaborado pela autora (2023).

Para excluir um cliente da lista de clientes o usuário deve clicar no botão “Deletar”. Surgirá na tela uma mensagem de confirmação da ação (Fig. 30-b). Ao clicar na opção “Cancelar”, o sistema retorna para a tela “Editar Cliente” e ao clicar na opção “Deletar”, o cliente é deletado da lista e o sistema apresenta a tela “Meus Clientes”. Para voltar para a tela “Meus Clientes”, o usuário pode clicar no ícone da seta localizado na barra superior do lado esquerdo. Ao clicar no botão “Salvar”, além de salvar as informações no sistema, o usuário também volta para a tela “Meus Clientes” (Fig. 28-a).

A Figura 31 apresenta a tela “Meus Serviços”. Nessa mesma tela, o usuário pode visualizar duas listas de serviços diferentes. Ao clicar na aba “Ativos” (Fig. 31-a), o usuário pode visualizar os serviços cadastrados ativos, que estão sendo realizados pelo usuário. Na aba “Desativados” (Fig. 31-b), o usuário pode visualizar os serviços cadastrados como desativados, ou seja, que não estão sendo realizados pelo usuário, mas que pode voltar a ser ativo. Para cada serviço o sistema carrega a imagem, o nome, o valor e o tempo de duração. As listas dos serviços seguem um padrão de organização por ordem alfabética, para facilitar a busca pelo serviço. O usuário pode escolher qual lista ele quer visualizar ao clicar na aba correspondente.

Na aba “Ativos” o usuário pode cadastrar um serviço clicando no botão com o sinal de mais, que fica fixo na parte inferior direita da tela (fundo de cor laranja). Ao clicar neste botão, o sistema apresenta ao usuário a tela “Cadastrar Serviço” (Fig. 32).

Para acessar ou editar as informações do serviço, o usuário pode clicar em qualquer região da tela que representa aquele serviço (foto, nome), inclusive no ícone de editar, representado por uma caneta na cor azul, na extremidade oposta do nome do cliente, que o sistema apresenta para o usuário a tela “Editar Serviço” (Fig. 33-a). O usuário pode voltar para a tela anterior, no caso a tela “Home”, clicando no ícone seta localizado na barra de aplicativos superior da tela “Meus Serviços”, ou navegar pelos botões na barra de navegação inferior.

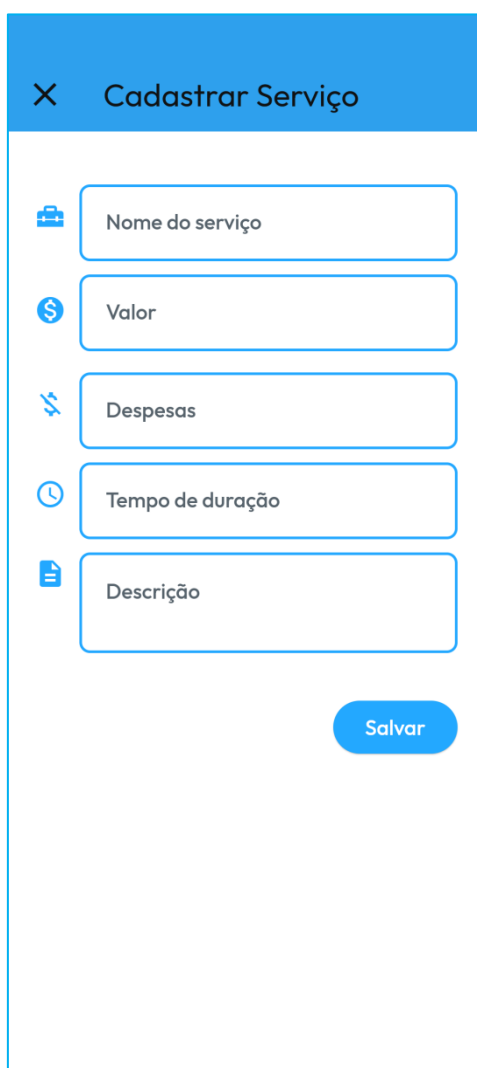
**Figura 31 - Captura da tela Meus Serviços: Ativos (a) e Desativados (b).**

Fonte: Elaborado pela autora (2023).

A Figura 32 apresenta a tela “Cadastrar Serviço”. Nessa tela o usuário pode preencher todos os campos com informações do serviço como “Nome do Serviço”, “Valor”, “Despesas”, “Tempo de duração” e “Descrição”. Os campos “Nome do Serviço”, “Valor” e “Tempo de duração” devem ser preenchido obrigatoriamente pelo usuário para cadastrar o serviço no sistema. O usuário pode adicionar uma foto do serviço ao clicar no botão “Adicionar foto”. Esse botão abre uma janela com as opções de carregar uma foto da galeria de fotos ou tirar uma foto com a câmera de dispositivo, como o exemplo apresentado na Figura 27.

O botão “Salvar” salva as informações do serviço no sistema e retorna para a tela “Meus Serviços”, onde o usuário pode visualizar o serviço cadastrado na lista de serviços. Caso o usuário preencha os campos, mas desista de cadastrar o serviço pode clicar no ícone em forma de “x” na barra superior da tela. Dessa forma, o sistema não guardará nenhuma informação inserida nos campos e retorna para a tela “Meus Serviços”.

**Figura 32 - Captura da tela Cadastrar Serviço.**

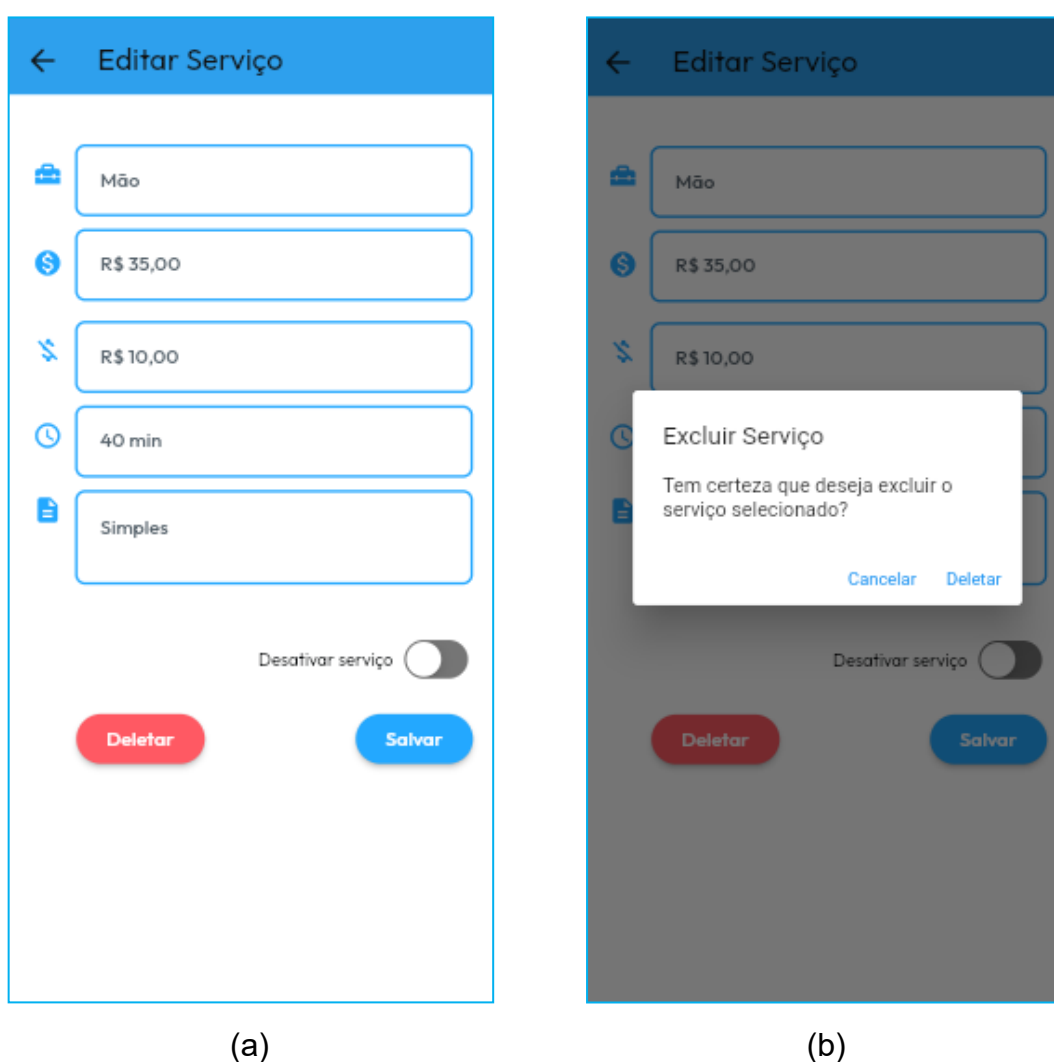
A imagem mostra a interface de usuário para cadastrar um novo serviço. No topo, há uma barra azul com um ícone de 'X' para fechar e o título 'Cadastrar Serviço'. Abaixo, há cinco campos de entrada, cada um com um ícone representativo: 'Nome do serviço' (ícone de briefcase), 'Valor' (ícone de dólar), 'Despesas' (ícone de setas cruzadas), 'Tempo de duração' (ícone de relógio) e 'Descrição' (ícone de documento). Um botão azul 'Salvar' está localizado na parte inferior direita da tela.

**Fonte: Elaborado pela autora (2023).**

A tela “Editar Serviço” (Fig. 33-a) apresenta ao usuário os campos preenchidos com as informações do serviço selecionado na tela “Meus Serviços”. Por meio da tela “Editar Serviços”, o usuário pode alterar as informações do serviço,

como também preencher alguns campos que ficaram sem informação no momento de cadastrar o cliente, na tela “Cadastrar Serviço”. Para desativar ou ativar um serviço, o usuário pode deslizar o botão para a direita (ligado) ou para a esquerda (desligado). Esta opção indica ao sistema em qual lista o serviços poderá ser visualizado, se na lista de serviços “Desativados”, quando o botão está “ligado”, ou na lista de “Ativos”, quando o botão está “desligado”.

**Figura 33 - Captura da tela Editar Serviço (a) e janela de mensagem Excluir Serviço (b).**



Fonte: Elaborado pela autora (2023).

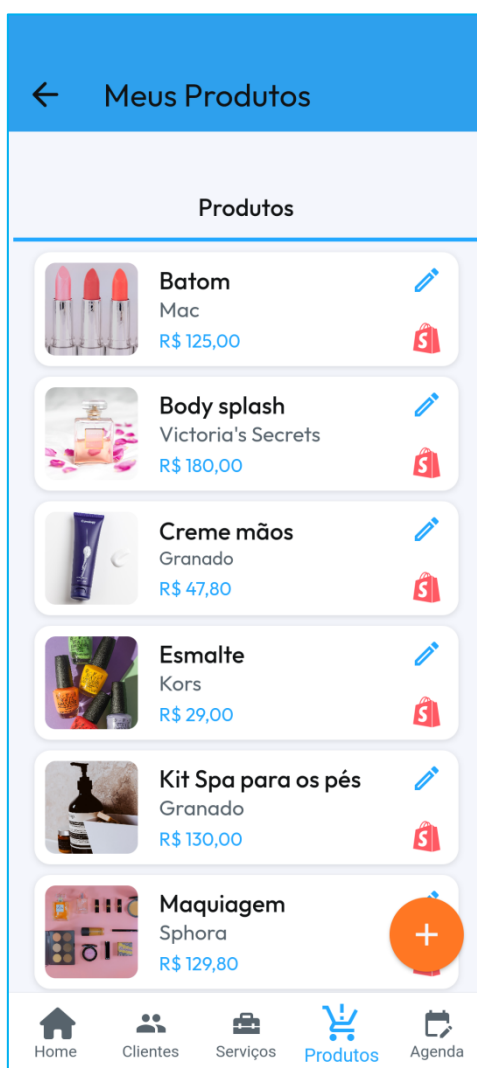
Para excluir um serviço da lista de serviços o usuário deve clicar no botão “Deletar”. Surgirá na tela uma mensagem de confirmação da ação (Fig. 33-b). Ao clicar na opção “Cancelar”, o sistema retorna para a tela “Editar Serviço”, e ao clicar



na opção “Deletar”, o serviço é deletado da lista e o sistema apresenta a tela “Meus Serviços”. Para voltar para a tela “Meus Serviços” o usuário pode clicar no ícone da seta, localizado na barra superior do lado esquerdo. Ao clicar no botão “Salvar”, além de salvar as informações no sistema o usuário também volta para a tela “Meus Serviços”.

A Figura 34 apresenta a tela “Meus Produtos”. Nessa tela o usuário pode visualizar a lista de produtos cadastrados. Para cada produto o sistema carrega a imagem, o nome, a marca e o preço de venda. A lista é apresentada por ordem alfabética, para facilitar a busca pelo produto.

**Figura 34 - Captura da tela Meus Produtos.**



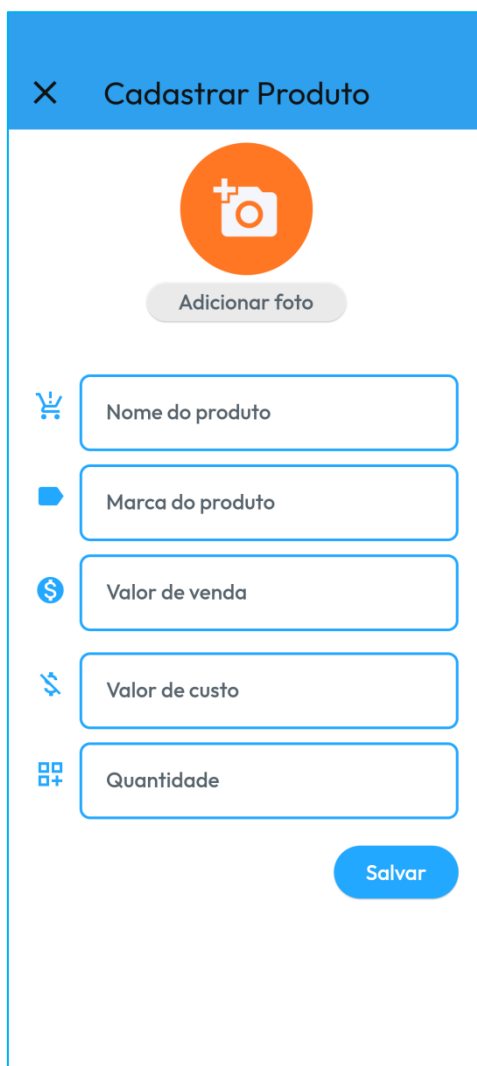
**Fonte: Elaborado pela autora (2023).**

Para cadastrar um produto o usuário deve clicar no botão com o sinal de mais, que fica fixo na parte inferior direita da tela (fundo de cor laranja). Ao clicar neste botão, o sistema apresenta ao usuário a tela “Cadastrar Produto” (Fig. 35).

Para acessar e editar as informações do produto, o usuário pode clicar em qualquer região da tela que representa aquele produto (foto, nome), inclusive no ícone de editar, representado por uma caneta na cor azul, na extremidade oposta do nome do produto, que o sistema apresenta para o usuário a tela “Editar Produto” (Fig. 35). Para realizar a venda do produto, sem ter que abrir a tela “Editar Produto”, o usuário pode clicar no ícone de venda (sacola vermelha) localizado no canto inferior direito da região da tela que representa determinado produto. Dessa forma o sistema apresenta para o usuário a tela “Comanda de Venda” (Fig. 44-b). O usuário pode voltar para a tela anterior, no caso a tela “Home”, clicando no ícone seta localizado na barra de aplicativos superior da tela “Meus Produtos”, ou navegar pelos botões na barra de navegação inferior.

A Figura 35 apresenta a tela “Cadastrar Produto”. Nessa tela o usuário pode preencher todos os campos com informações do produto como “Nome do produto”, “Marca do produto”, “Valor de venda”, “Valor de custo”, “Quantidade” e “Descrição”. Os campos “Nome do produto” e “Valor de venda” devem ser preenchidos obrigatoriamente pelo usuário para cadastrar o produto no sistema. O usuário pode adicionar uma foto do produto ao clicar no botão “Adicionar foto”. Esse botão abre uma janela com as opções de carregar uma foto da galeria de fotos ou tirar uma foto com a câmera de dispositivo, como o exemplo apresentado na Figura 27.

O botão “Salvar” salva as informações do produto no sistema e retorna para a tela “Meus Produtos”, onde o usuário pode visualizar o produto cadastrado na lista de produtos. Caso o usuário preencha os campos, mas desista de cadastrar o produto pode clicar no ícone em forma de “x” na barra superior da tela. Dessa forma, o sistema não guardará nenhuma informação inserida nos campos e retorna para a tela “Meus Produtos”.

**Figura 35 - Captura da tela Cadastrar Produto.**

A captura de tela mostra a interface de usuário para o registro de um novo produto. No topo, há uma barra azul com o ícone de fechar (X) e o título "Cadastrar Produto". Abaixo, um ícone de câmera laranja indica a opção de "Adicionar foto". O formulário principal contém cinco campos de entrada, cada um com um ícone representativo à esquerda: "Nome do produto" (carrinho de compras), "Marca do produto" (etiqueta), "Valor de venda" (símbolo de dólar), "Valor de custo" (setas azuis) e "Quantidade" (ícone de grade). Um botão azul "Salvar" está posicionado no canto inferior direito do formulário.

**Fonte: Elaborado pela autora (2023).**

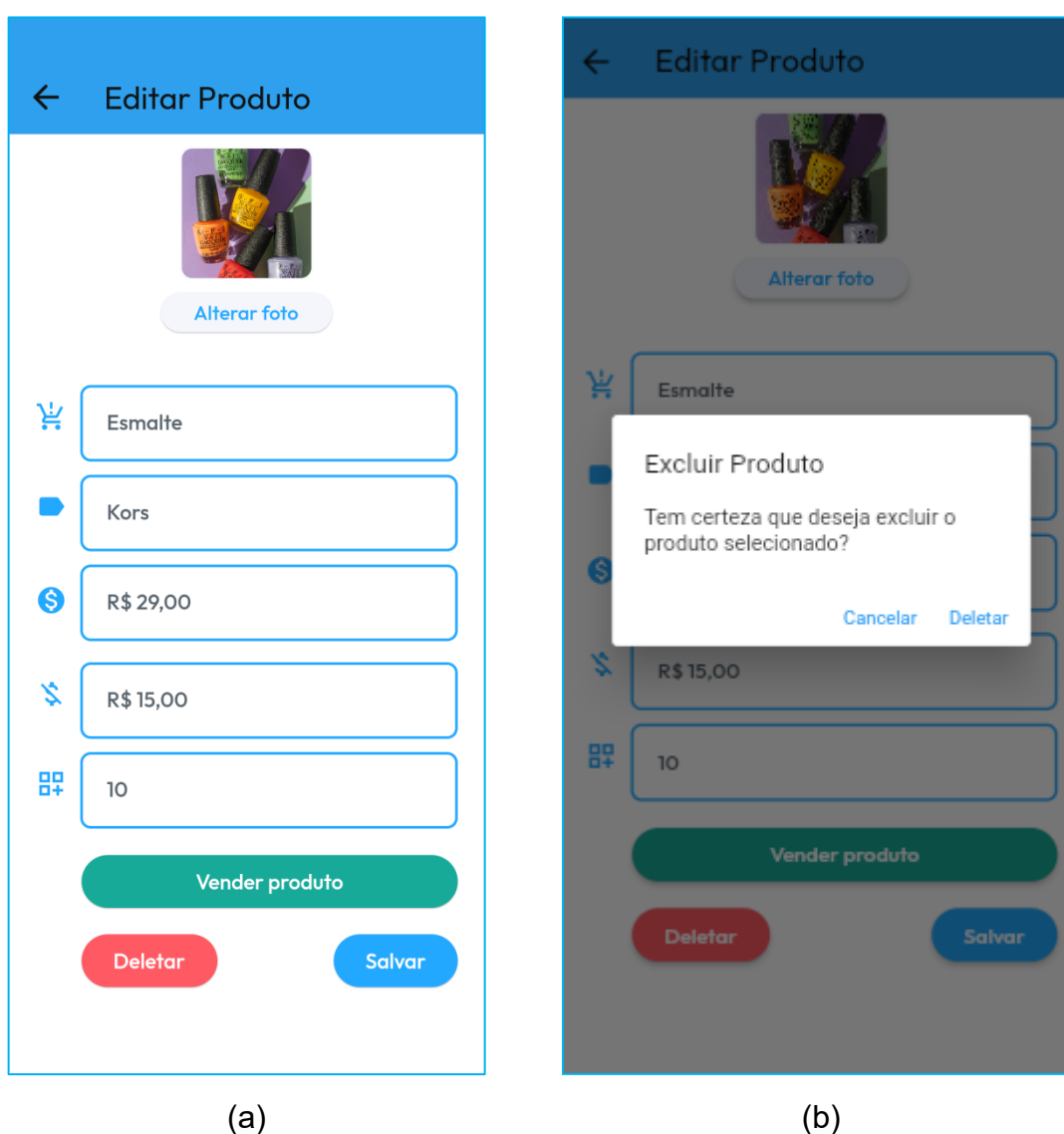
A tela “Editar Produto” (Fig. 36-a) apresenta ao usuário os campos preenchidos com as informações do produto selecionado na tela “Meus Produtos”. Por meio dessa tela o usuário pode alterar as informações do produto e preencher os campos que ficaram sem informação no momento de cadastrar o produto, na tela “Cadastrar Produto”.

Para realizar a venda do produto, o usuário deve clicar no botão “Vender Produto”. O sistema vai direcionar o usuário para a tela “Comanda de Venda” (Fig. 44-b), onde outras informações podem ser incluídas para a realização da venda.

Para excluir um produto da lista de produtos o usuário deve clicar no botão “Deletar”. Surgirá na tela uma mensagem de confirmação da ação (Fig. 36-b). Ao

clicar na opção “Cancelar”, o sistema retorna para a tela “Editar Produto” e ao clicar na opção “Deletar”, o produto é deletado da lista e o sistema apresenta a tela “Meus Produtos”. Para voltar para a tela “Meus Produtos” o usuário pode clicar no ícone da seta, localizado na barra superior do lado esquerdo. Ao clicar no botão “Salvar”, além de salvar as informações no sistema o usuário também volta para a tela “Meus Produtos”.

Figura 36 - Captura da tela Editar Produto (a) e mensagem de Excluir Produto (b).



Fonte: Elaborado pela autora (2023).

A Figura 37 apresenta a tela “Agenda”. Nessa tela o usuário pode visualizar seus agendamentos diários, criar um agendamento e navegar para outras telas relacionadas com a agenda. No início da tela observa-se um texto informando a data atual e um calendário semanal (segunda a domingo). Ao clicar em um dia da semana desse calendário, o usuário pode visualizar seus agendamentos para o dia em formato de lista, logo abaixo. O botão com o ícone de calendário, na parte superior direita da tela, direciona o usuário para a tela “Calendário” (Fig. 39), onde é possível visualizar uma lista de agendamentos para vários dias. O botão “Horários livres” direciona o usuário para a tela “Horários Livres” (Fig. 41), que mostra os horários livres para a semana.

**Figura 37 - Captura da tela Agenda.**



**Fonte: Elaborado pela autora (2023).**

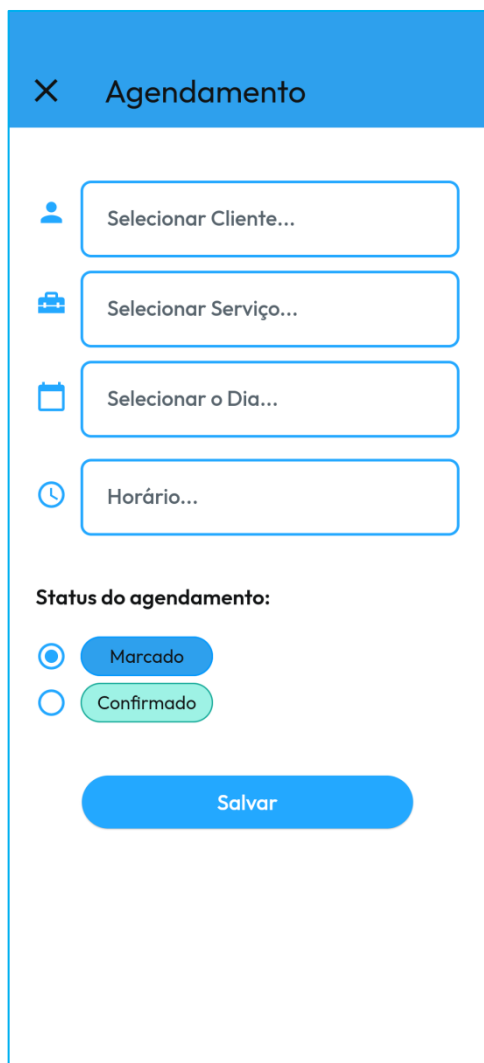
O botão “Finalizados” direciona o usuário para a tela “Agenda Fechada” (Fig. 40), onde o usuário pode visualizar os agendamentos concluídos e cancelados. Os agendamentos são organizados em ordem cronológica, para cada dia da semana selecionado. Cada retângulo ou container dessa lista corresponde a um agendamento, e nele são carregados pelo sistema as informações de tempo do serviço (início e fim), o nome do cliente e o nome do serviço. A cor do container mostra para o usuário qual é o estado do agendamento (marcado, confirmado ou cancelado), conforme é definido na tela “Agendamento” (Fig. 38-a) ou na tela “Editar Agendamento” (Fig. 40-a). Para fazer alterações em um agendamento, ou encerra o serviço, o usuário pode clicar em qualquer parte do container correspondente que o sistema apresenta a tela “Editar Agendamento” (Fig. 40-a).

Para criar um novo agendamento, o usuário pode clicar no botão com o sinal de mais (cor laranja), localizado no canto inferior direito da tela. O sistema vai direcioná-lo para a tela “Agendamento” (Fig. 38-a). Para voltar para a tela “Home” o usuário pode clicar no ícone volta localizado do lado esquerdo da barra superior ou clicar na aba “Home” da barra inferior.

A Figura 38 apresenta a tela “Agendamento”. Essa tela permite que o usuário crie um agendamento. O usuário deve preencher obrigatoriamente os campos “Selecionar Cliente”, “Selecionar Serviço”, “Selecionar o Dia” e “Horário”.

Ao clicar no campo “Selecionar Cliente”, o sistema apresenta a tela “Meus Clientes” (Fig. 28), onde o usuário deve selecionar um cliente. Ao selecionar o cliente, o sistema retorna para a tela “Agendamento” com o campo “Selecionar Cliente” preenchido automaticamente com o nome do cliente selecionado pelo usuário. O mesmo processo ocorre ao selecionar o serviço no campo “Selecionar Serviço”, o sistema apresenta a tela “Meus Serviços” (Fig. 31), onde o usuário deve selecionar um serviço. Ao selecionar o serviço, o sistema retorna para a tela “Agendamento” com o campo “Selecionar Serviço” preenchido automaticamente com o nome do serviço selecionado pelo usuário.

**Figura 38 - Captura da tela Agendamento (a), janela selecionar data do agendamento (b) e janela selecionar horário do agendamento (c).**



Agendamento

Selecione o Cliente...

Selecione o Serviço...

Selecione o Dia...

Horário...

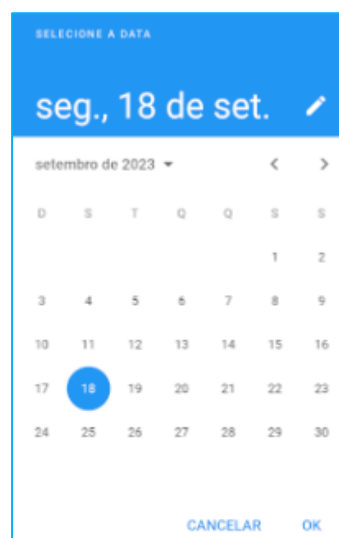
Status do agendamento:

Marcado

Confirmado

Salvar

(a)



SELECIONE A DATA

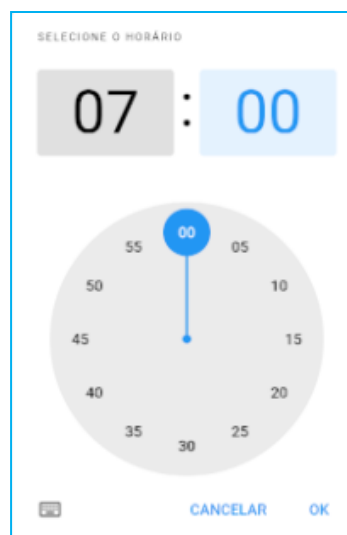
seg., 18 de set.

setembro de 2023

D	S	T	Q	Q	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

CANCELAR OK

(b)



SELECIONE O HORÁRIO

07 : 00

00

05 10 15 20 25 30 35 40 45 50 55

CANCELAR OK

(c)

Fonte: Elaborado pela autora (2023).

Ao clicar no campo “Selecione o Dia” o sistema apresenta ao usuário uma janela com o calendário do mês atual e a opção de avançar os meses, onde é possível selecionar a data para a realização do agendamento (Fig. 38-b). Ao clicar no campo “Horário” o sistema apresenta ao usuário uma janela com a representação de um relógio, onde é possível selecionar um horário para definir o início do agendamento (Fig. 38-c).

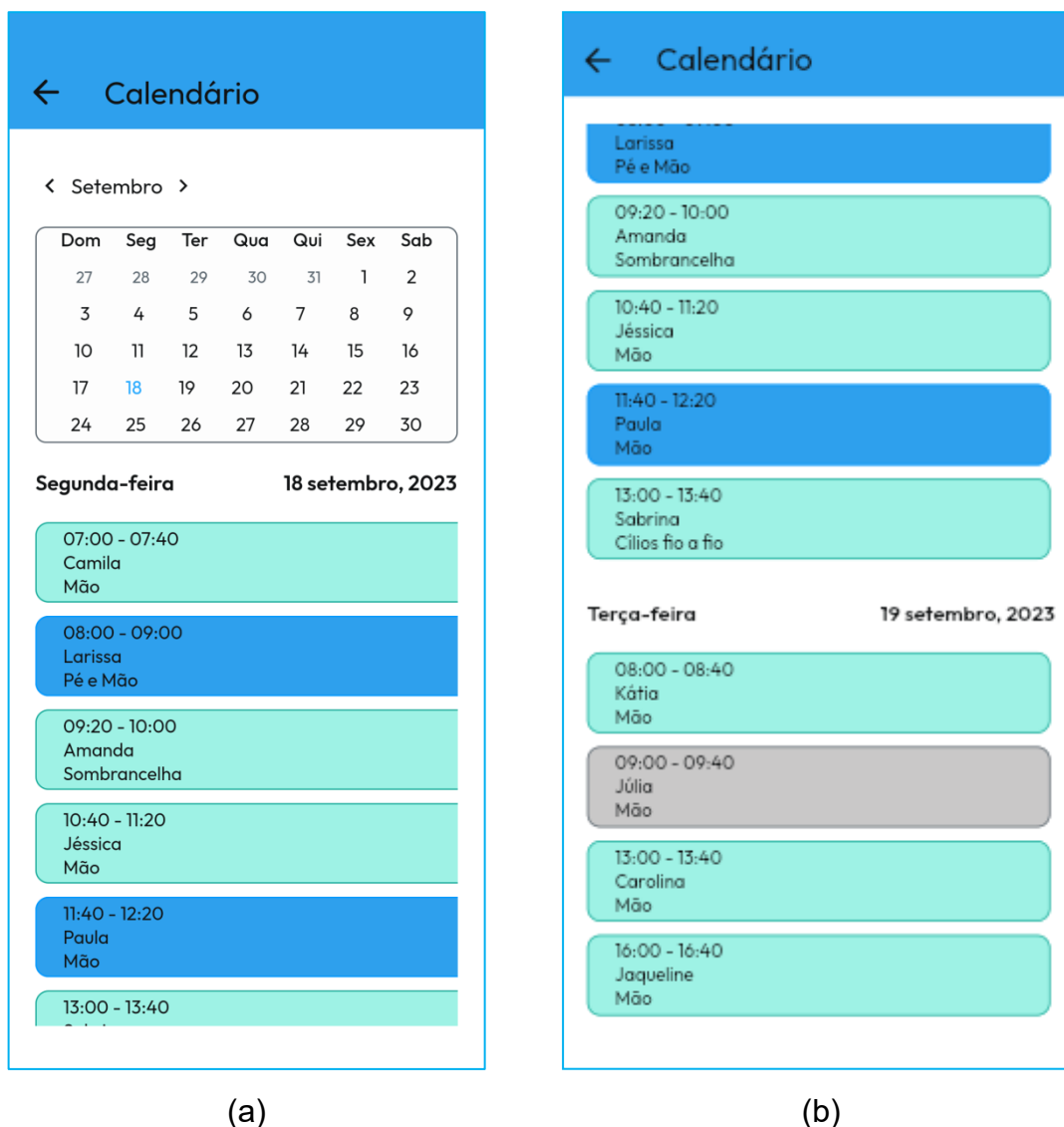
Na seção “Status do agendamento”, o usuário pode selecionar o estado do agendamento entre “Marcado” ou “Confirmado”. De forma automática, o sistema já deixa selecionado o status como “Marcado”. O botão “Salvar” salva as informações do agendamento no sistema e retorna para a tela “Agenda”. Caso o usuário preencha os campos, mas desista do agendamento pode clicar no ícone em forma de “x” na barra superior da tela. Dessa forma, o sistema não guardará nenhuma informação inserida nos campos e retorna para a tela “Agenda”.

A Figura 39 apresenta a tela “Calendário”. Nessa tela o usuário pode visualizar uma lista de agendamentos a partir de um dia selecionado no calendário. O usuário pode selecionar o mês, por meio dos ícones de seta na parte superior do calendário. Ao selecionar um dia, clicando no calendário, o sistema retorna, nessa mesma tela, a lista de agendamento a partir do primeiro horário marcado para o dia selecionando e os agendamentos seguintes.

Nessa tela, cada agendamento é representado por um container com as informações do horário, do cliente e do serviço, e são organizados nas seções que representam a data do agendamento. A cor de fundo do container indica o estado do agendamento, sendo azul para “Marcado”, verde para “Confirmado” e cinza para “Cancelado”. Para editar um agendamento o usuário pode clicar em qualquer parte do container que representa o agendamento selecionado. Nesse caso o sistema direciona o usuário para a tela “Editar Agendamento” (Fig. 42). Para voltar para a tela “Agenda”, o usuário deve clicar na seta que simboliza a ação de voltar na barra superior da tela.



**Figura 39 - Captura da tela Calendário (a) e continuação da tela Calendário (b).**



Fonte: Elaborado pela autora (2023).

A Figura 40 apresenta a tela “Agenda Fechada”. Essa tela mostra para o usuário algumas informações sobre os agendamentos concluídos e cancelados. O usuário pode selecionar o mês clicando nas setas localizadas ao lado do texto que indica o mês atual, no topo do corpo da tela. O sistema apresenta ao usuário o total de agendamentos concluídos, ou seja, agendamentos que foram finalizados por meio da comanda, o total de agendamentos cancelados e uma lista desses agendamentos. Cada linha da lista corresponde a um agendamento. O fundo branco da linha representa os agendamentos concluídos e o fundo cinza os agendamentos cancelados, conforme a legenda localizada na parte superior da lista. As informações que o sistema apresenta são a data, o nome do cliente e o serviço que

foi marcado no momento do agendamento. Para voltar para a tela “Agenda”, o usuário pode clicar no ícone voltar (seta para esquerda) localizado na barra superior.

**Figura 40 - Captura da tela Agenda Fechada.**

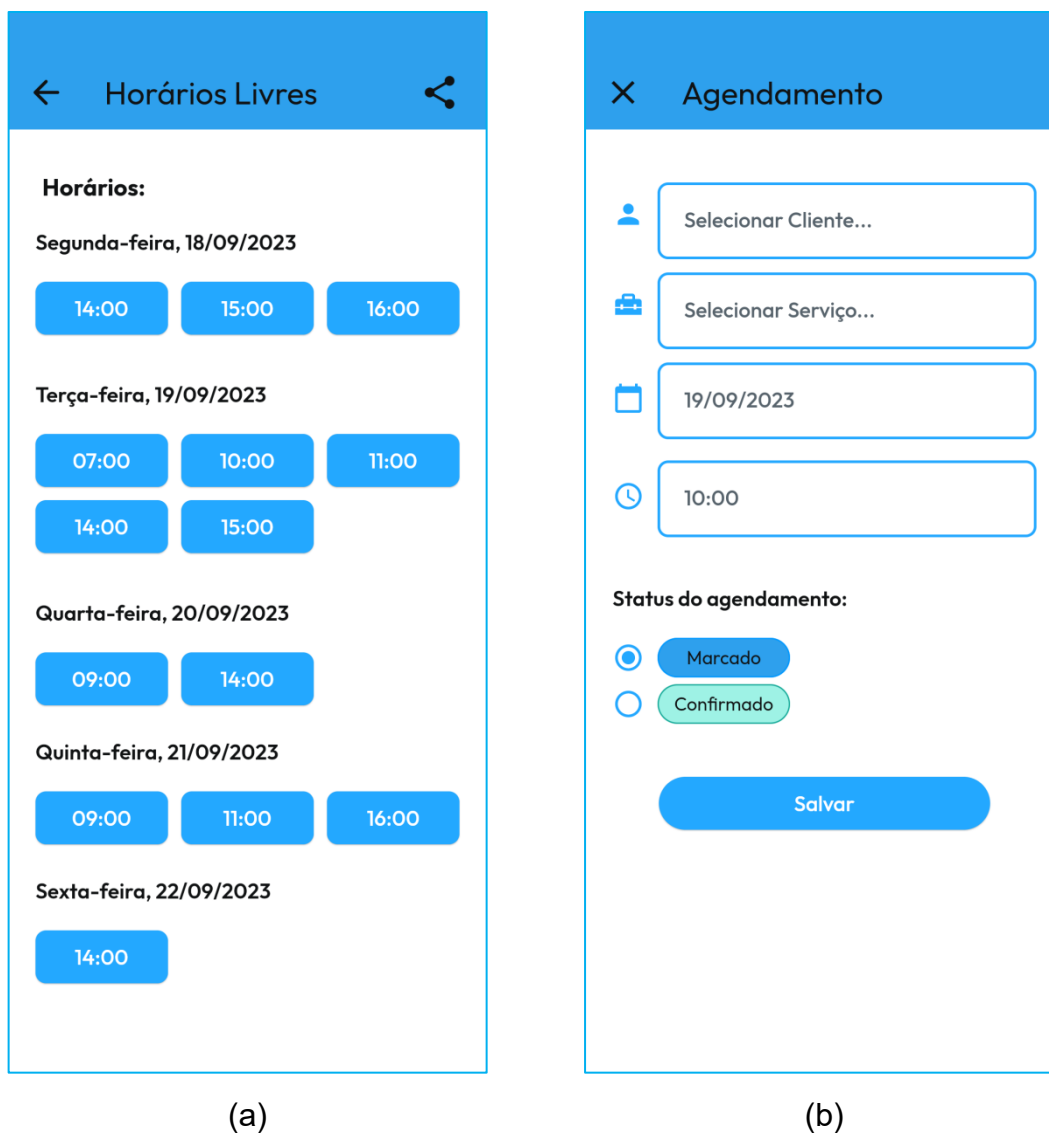
Setembro de 2023		
Agendamentos Concluídos	Agendamentos Cancelados	
42	6	
📅		
<input type="checkbox"/> Concluída	<input type="checkbox"/> Cancelada	
Data	Cliente	Serviço
15/09	Júlia	Mão
15/09	Paula	Pé e Mão
15/09	Maria	Maquiagem
15/09	Carla	Hidratação
14/09	Andressa	Escova
14/09	Tháís	Depilação
14/09	Lara	Mão
14/09	Cássia	Pé e Mão
13/09	Paloma	Mão
13/09	Juliana	Mão
13/09	Gabriela	Escova
13/09	Kátia	Hairstyle

**Fonte: Elaborado pela autora (2023).**

A Figura 41 (a) apresenta a tela “Horários Livres”. A tela de horários livres apresenta para o usuário os horários disponíveis para a semana atual, separados por seções de acordo com o dia da semana. O usuário pode compartilhar uma imagem dessa tela com seus clientes clicando no ícone de compartilhar localizado do lado direito da barra superior. O sistema abre o aplicativo WhatsApp de mensagens e o usuário pode selecionar um contato e compartilhar a imagem. Para realizar um agendamento a partir dessa tela, o usuário pode clicar em um horário (p.

ex., 10h do dia 19 de setembro) que o sistema direciona o usuário para a tela “Agendamento” e preenche automaticamente os campos de data e horário (Fig. 41-b). Os demais campos o usuário deve preencher e clicar no botão “Salvar” (Fig. 41-b) para salvar o agendamento.

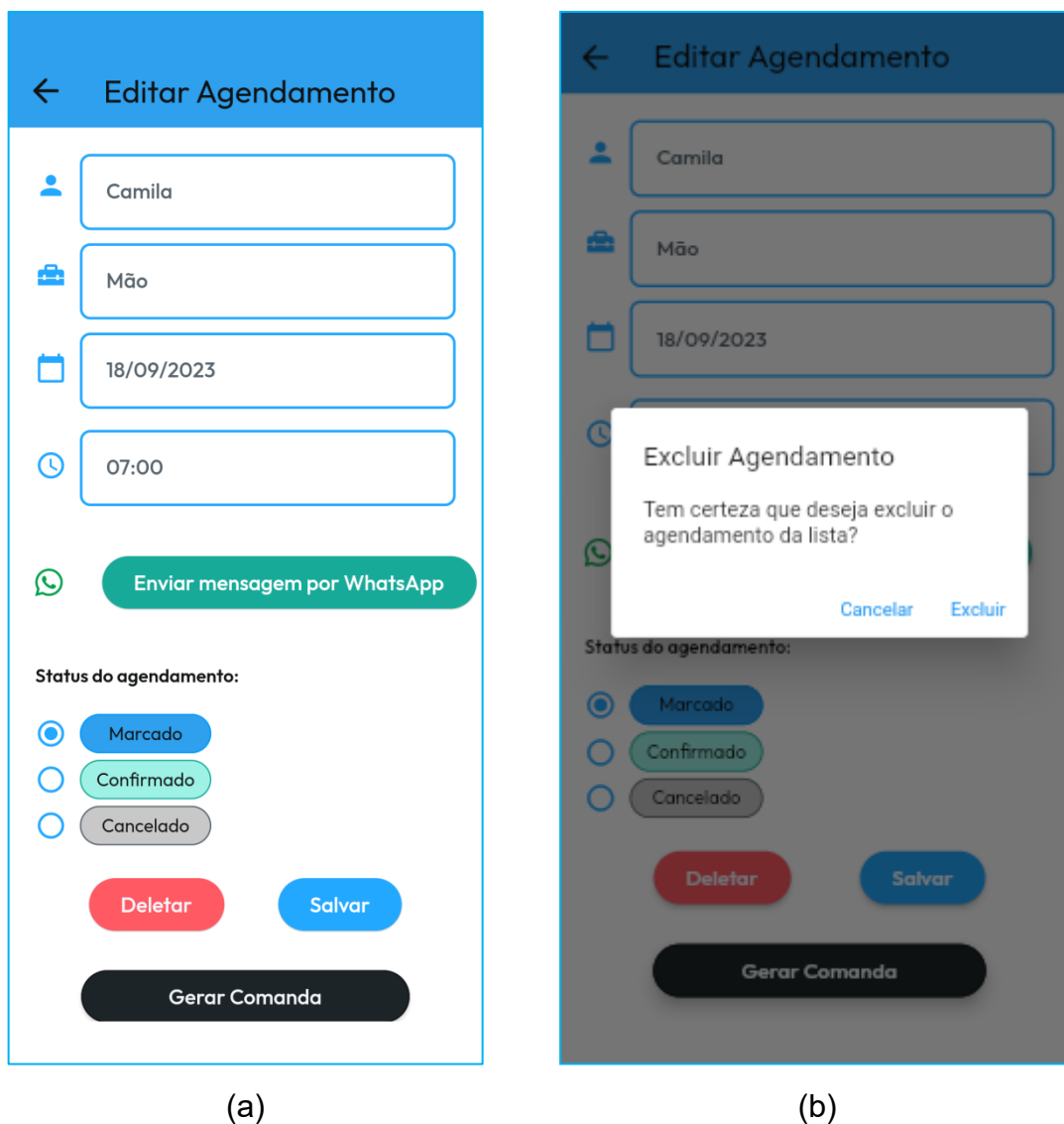
**Figura 41 - Captura das telas Horários Livres (a) e Agendamento (b).**



Fonte: Elaborado pela autora (2023).

A Figura 42 apresenta a tela “Editar Agendamento”. Nessa tela o usuário pode fazer alterações nos campos já preenchidos, enviar mensagem pelo aplicativo WhatsApp para o cliente agendado, redefinir o status do agendamento, deletar o agendamento, salvar as modificações e gerar a comanda do serviço.

Figura 42 - Captura da tela Editar Agendamento (a) e mensagem de Excluir agendamento (b).



Fonte: Elaborado pela autora (2023).

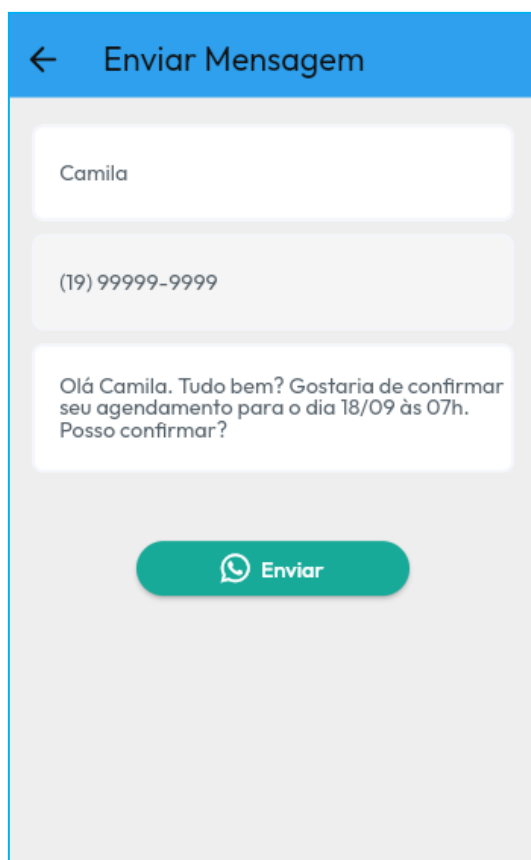
O botão “Enviar mensagem por WhatsApp” direciona o usuário para a tela “Enviar Mensagem”, onde o sistema preenche as informações do nome e telefone do cliente e o usuário digita uma mensagem, por exemplo, de confirmação do agendamento.

Caso o usuário mude a opção do estado (*status*) do agendamento, o sistema faz as alterações nas telas “Agenda” e “Calendário”. Ao clicar no botão “Deletar”, o sistema apresenta ao usuário uma janela com uma mensagem de confirmação de duas opções de escolha: “Cancelar” e “Excluir”. Se o usuário clicar em “Cancelar”, o sistema retorna a tela de “Editar Agendamento”, com todas as informações

previamente editadas. Caso o usuário opte por clicar no botão “Deletar”, o sistema deleta o agendamento e retorna para a tela “Agenda”. Ao clicar no botão “Salvar”, além de salvar as informações no sistema o usuário também volta para a tela “Agenda”. O botão “Gerar Comanda” direciona o usuário para a tela “Comanda de Serviço” (Fig. 44-a). Nesta tela, o usuário pode inserir mais informações e encerrar o agendamento.

A Figura 43 apresenta a tela “Enviar Mensagem”. Essa tela permite que o usuário envie uma mensagem personalizada para seu cliente via aplicativo WhatsApp. Os campos “Nome do cliente” e “Telefone” são preenchidos automaticamente pelo sistema na tela “Editar Agendamento”. O campo “Mensagem” deve ser preenchido pelo usuário obrigatoriamente antes de clicar no botão “Enviar”.

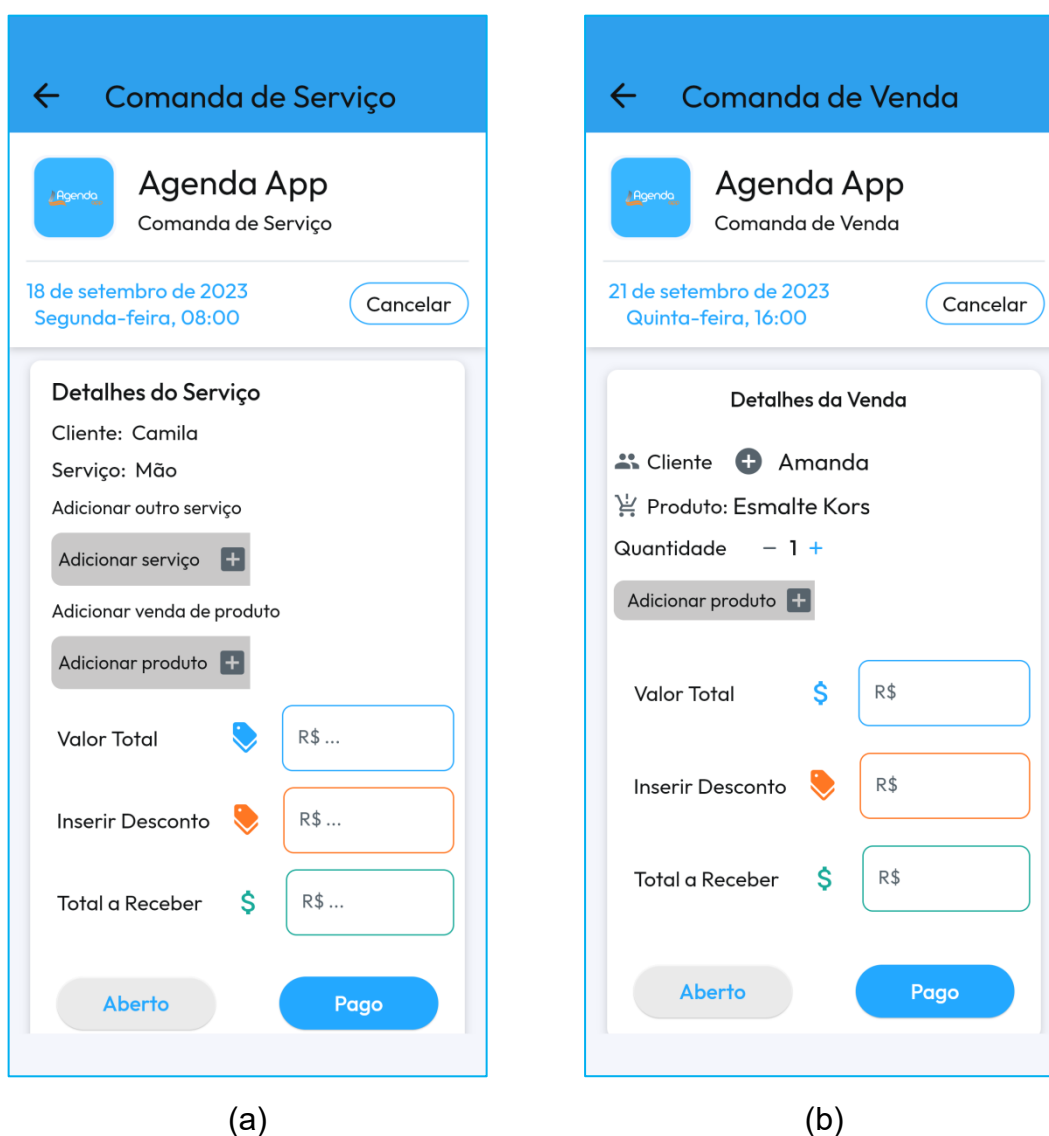
**Figura 43 - Captura da tela Enviar Mensagem.**



**Fonte: Elaborado pela autora (2023).**

A Figura 44 apresenta as telas das comandas digitais que o sistema gera ao finalizar um atendimento de serviço e de venda de produtos. A tela “Comanda de Serviço” (Fig. 44-a) mostra para o usuário algumas informações sobre o serviço que foi concluído, como a data, o nome do cliente, o serviço prestado e o valor total cobrado. Ao clicar no botão “Cancelar”, o usuário desiste de concluir o atendimento e volta para a tela “Editar Agendamento”. Dessa forma o sistema não salva nenhuma informação sobre o agendamento.

**Figura 44 - Captura da tela Comanda de Serviço (a) e Comanda de Venda (b).**



Fonte: Elaborado pela autora (2023).

Na comanda de serviço, na seção “Detalhes do Serviço”, as informações sobre o nome do cliente e o serviço prestado são preenchidas automaticamente pelo sistema por meio do agendamento. Nessa comanda o usuário pode adicionar outro serviço como também a venda de um produto. Para selecionar outro serviço o usuário pode clicar no sinal de mais ao lado do atalho “Adicionar serviço”. O sistema apresentará a tela “Meus Serviços” (Fig. 31) para o usuário selecionar um serviço. Selecionado o serviço o sistema preenche o campo ao lado com o nome do serviço. Para selecionar a venda de um produto, o usuário pode clicar no sinal de mais ao lado do atalho “Adicionar produto”. O sistema apresenta a tela “Meus Produtos” (Fig.34) para o usuário selecionar um produto. Selecionado o produto o sistema preenche o campo ao lado com o nome do produto.

No campo correspondente ao “Valor Total”, o sistema carrega o valor do primeiro serviço. Caso o usuário selecione outro serviço e um produto para venda, o sistema soma o novo valor ao valor inicial. No campo “Inserir Desconto” o usuário pode inserir um valor a ser descontado do valor total. Esse campo não é obrigatório, caso o usuário não insira um valor, o valor do campo “Total a Receber” será igual ao do campo “Valor Total”.

Caso o usuário opte por deixar a conta em aberto, ou seja, para o cliente pagar em um momento futuro, o usuário deve clicar no botão “Aberto”. Ao clicar em “Aberto” o sistema apresenta uma janela onde o usuário confirma a opção ao clicar no botão “Continuar”. Dessa forma, o sistema armazena as informações da comanda em aberto e fica disponível para visualização do usuário no “Relatório de Cobranças” (Fig. 48-a). O botão “Voltar” cancela a ação e retorna a tela “Comanda de Serviço”.

Para confirmar o pagamento, o usuário deve clicar no botão “Pago”. Ao clicar no botão “Pago”, o sistema apresenta a tela “Confirmação de pagamento” (Fig. 45-a), onde o usuário deve escolher qual a forma de pagamento realizada pelo cliente: dinheiro, crédito, débito ou PIX (Pagamento Instantâneo Brasileiro). Quando o usuário clica em uma destas opções, o sistema mostra uma janela com uma mensagem de confirmação do fechamento da comanda (Fig. 45-b). Ao clicar no botão “Continue” desta janela, o sistema retorna para a tela “Agenda”. Caso o usuário opte por clicar no botão “Cancela” da tela “Confirmação de pagamento”, o sistema retorna para a tela “Comanda de Serviço”.

A tela “Comanda de Venda” (Fig. 44-b) apresenta para o usuário algumas informações sobre a venda de um ou mais produtos. Ao clicar no botão “Cancelar”, o usuário desiste de concluir a venda e volta para a tela “Meus Produtos” ou “Editar Produto”, dependendo da forma como o produto foi acessado pelo usuário.

Na seção “Detalhes da Venda”, a informação sobre o nome do produto e é preenchida automaticamente pelo sistema. Para selecionar o cliente vinculado a compra, o usuário pode clicar no sinal de mais ao lado do campo “Cliente”. Esta ação não é obrigatória para a realização da venda de um produto, para o sistema é necessário ter ao menos um produto selecionado. Ao carregar o nome do produto o sistema insere automaticamente o valor de 1 (um) no campo “Quantidade”. Caso o usuário venda mais de um mesmo produto, ele pode clicar no sinal de mais no campo “Quantidade”. A venda de outro produto pode ser realizada na mesma comanda. Para selecionar outro produto, o usuário pode clicar no sinal de mais ao lado do atalho “Adicionar produto”. O sistema apresentará a tela “Meus Produtos” para o usuário selecionar um produto. Selecionado o produto o sistema preenche automaticamente o campo ao lado com o nome do produto.

No campo correspondente ao “Valor Total”, o sistema carrega o valor do primeiro produto. Caso o usuário selecione outro produto para venda, o sistema atualiza o valor total, somando o novo valor ao valor inicial. No campo “Inserir Desconto” o usuário pode inserir um valor a ser descontado do valor total. Esse campo não é obrigatório, caso o usuário não insira um valor, o valor do campo “Total a Receber” será igual ao do campo “Valor Total”.

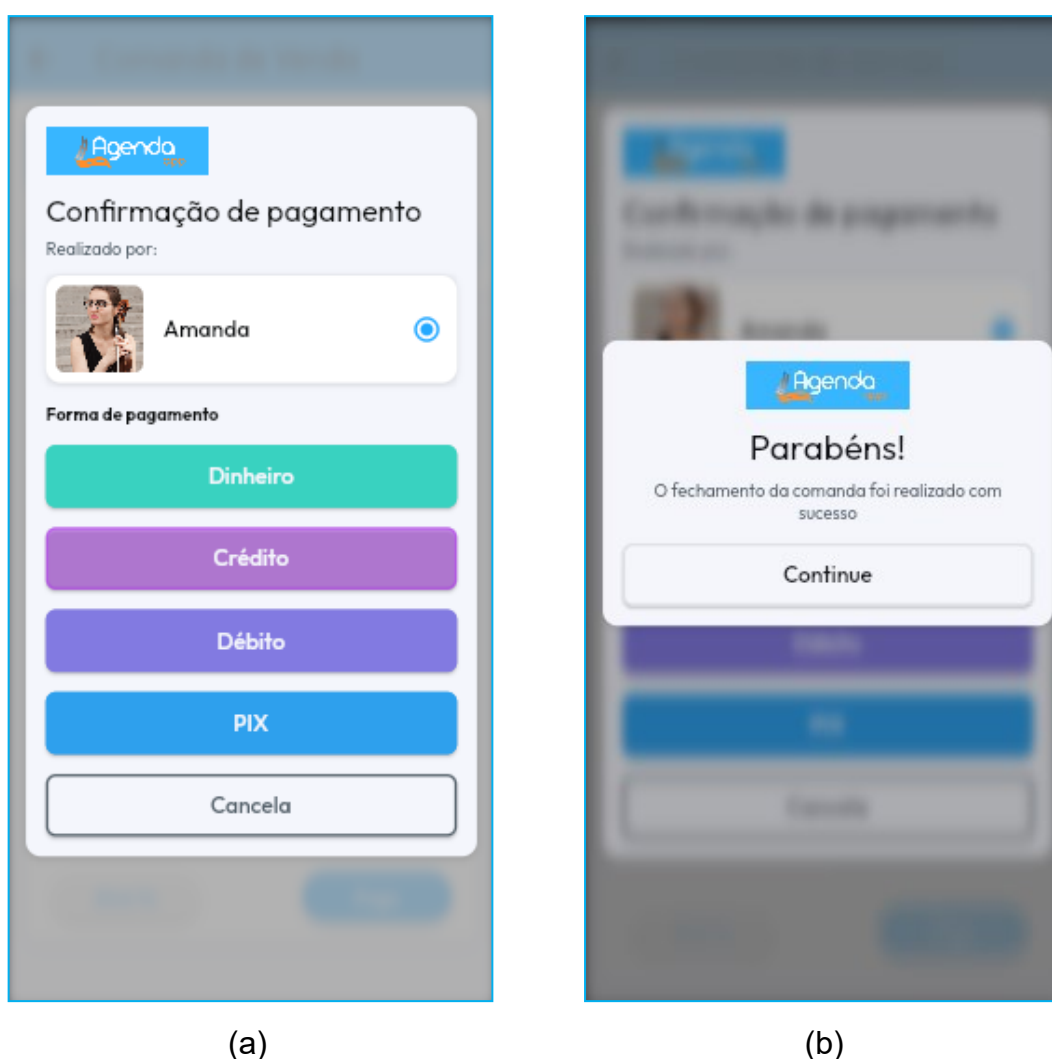
Caso o usuário opte por deixar a conta em aberto, ou seja, para o cliente pagar em um momento futuro, o usuário deve clicar no botão “Aberto”. Ao clicar em “Aberto” o sistema apresenta uma janela onde o usuário confirma a opção ao clicar no botão “Continuar” (Fig. 46). Dessa forma, o sistema armazena as informações da comanda e fica disponível para visualização do usuário no “Relatório de Cobranças” (Fig. 48-a). O botão “Voltar” cancela a ação e retorna a tela da “Comanda de Serviço”.

Para confirmar o pagamento, o usuário deve clicar no botão “Pago”. Ao clicar no botão “Pago”, o sistema apresenta a tela “Confirmação de pagamento” (Fig. 45-a), onde o usuário deve escolher qual foi a forma de pagamento (Dinheiro, Crédito,



Débito ou PIX) realizada pelo cliente. Quando o usuário clica em uma destas opções, o sistema mostra uma janela com uma mensagem de confirmação do fechamento da comanda (Fig. 44-b). Ao clicar no botão “Continue” desta janela, o sistema retorna para a tela “Meus Produtos”. Caso o usuário opte por clicar no botão “Cancela” da tela “Confirmação de pagamento”, o sistema retorna para a tela “Comanda de Venda”.

**Figura 45 - Captura da tela Confirmação de pagamento (a) e janela com mensagem de fechamento da comanda (b).**



Fonte: Elaborado pela autora (2023).

**Figura 46 - Captura da janela com confirmação de comanda em aberto.**

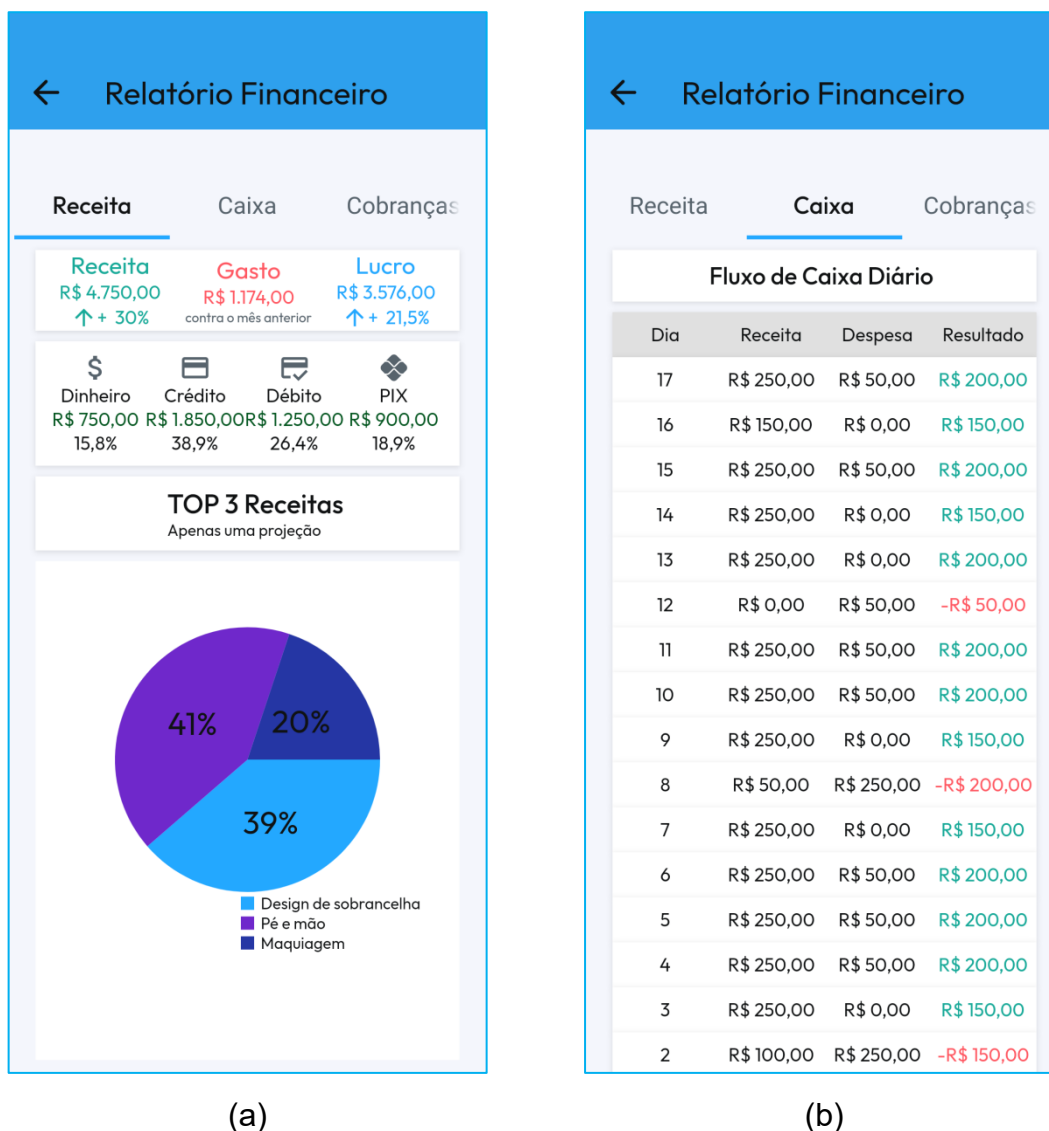


**Fonte: Elaborado pela autora (2023).**

A tela “Relatório Financeiro” apresenta ao usuário três tipos de relatórios distribuídos nas abas “Receita” (Fig. 47-a), “Caixa” (Fig. 47-b) e “Cobranças” (Fig. 48-a). O sistema gera essas informações através dos dados armazenados no próprio sistema, alimentado pelo usuário. Tanto na aba “Receita” quanto na aba “Caixa” a única ação disponível para o usuário é a seleção do mês no topo das telas.

O relatório “Receita” mostra para o usuário algumas informações importantes sobre a situação financeira do seu negócio. Ao selecionar o mês no topo da tela, o sistema traz as informações sobre: a receita, o total recebido; a despesas, ou gastos; o lucro; o valor total dos recebimentos realizados com dinheiro, cartão de crédito, cartão de débito e PIX, e na forma de um gráfico de pizza, os três serviços prestados que geram maior renda para o usuário.

Figura 47 - Captura da tela Relatório Financeiro: aba Receita (a) e aba Caixa (b).



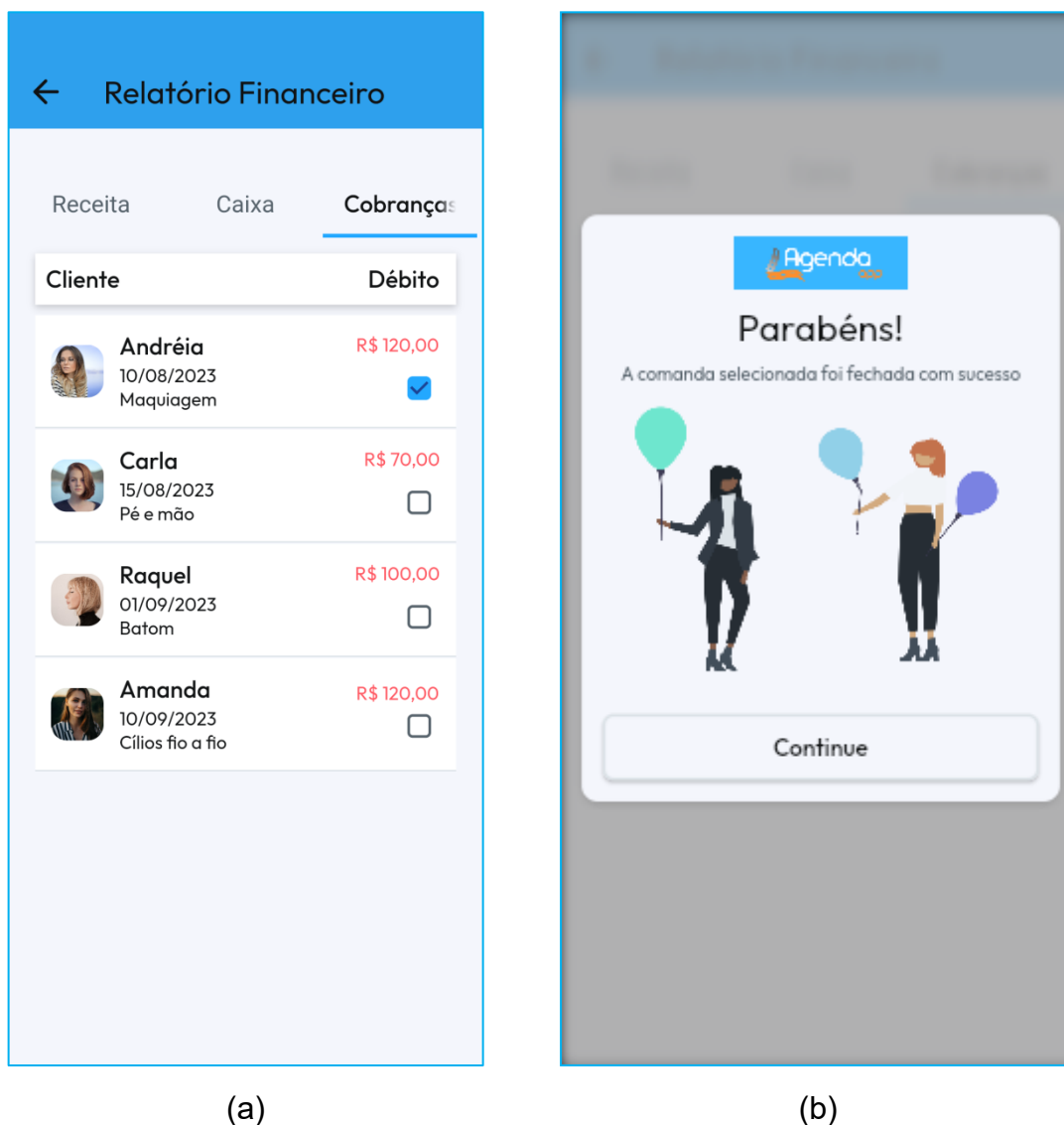
Fonte: Elaborado pela autora (2023).

O relatório “Caixa” mostra para o usuário, em forma de tabela, o fluxo de caixa diário do mês selecionado. Para cada dia do mês o sistema calcula a receita, as despesas e o resultado (receita – despesas).

A aba “Cobranças” (Fig. 48-a), da tela “Relatório Financeiro”, apresenta para o usuário uma lista das comandas que estão em aberto. Cada container da lista mostra as informações do nome do cliente em débito, a data que a comanda foi criada, o serviço prestado e o valor a ser recebido. Quando o usuário recebe o pagamento do cliente em débito, ele deve clicar no *Checkbox* correspondente no canto inferior direito. Então o sistema apresenta a tela da comanda correspondente,

de serviço (Fig. 44-a) ou de venda (Fig. 44-b). O usuário deve clicar no botão “Pago” para confirmar o recebimento. O sistema então apresenta a tela referente à forma de pagamento (Fig. 45-a) e ao selecionar o tipo de pagamento, o sistema apresenta a janela com a mensagem de confirmação do encerramento da comanda (Fig. 48-b). Ao clicar no botão “Continue”, dessa janela, o sistema retorna para a aba “Cobrança” da tela “Relatório Financeiro”. É por meio do relatório de cobranças que o usuário tem acesso as comandas em aberto.


**Figura 48 - Captura da tela Relatório Financeiro: aba Cobrança (a) e mensagem de confirmação de encerramento da cobrança (b).**



Fonte: Elaborado pela autora (2023).

A tela “Relatório Clientes”, apresentada na Figura 49, mostra para o usuário algumas informações relevantes sobre seus clientes como: o total de clientes cadastrados no sistema, o total de novos clientes cadastrados no sistema e uma lista de classificação dos clientes que mais contribuíram com a receita nos últimos doze meses. Os clientes cadastrados como sendo “Novos”, permanecem nessa lista por um período de seis meses. Após esse período, o sistema transfere o cliente para a lista “Todos”.

**Figura 49 - Captura da tela Relatório Clientes.**

A imagem mostra a interface de usuário da tela "Relatório Clientes". No topo, há um cabeçalho azul com um ícone de seta para trás e o título "Relatório Clientes". Abaixo, há duas colunas de estatísticas: "Total de Clientes" com o valor 245 e "Novos Clientes" com o valor 12. Segue-se o subtítulo "Classificação dos melhores clientes por rec nos últimos 12 meses". Abaixo disso, há uma tabela com duas colunas: "Nome" e "Receita". A tabela contém 10 linhas de dados, cada uma com um número de ranking, um nome e um valor monetário em reais.

	Total de Clientes	Novos Clientes
	245	12
Classificação dos melhores clientes por rec nos últimos 12 meses		
Nome	Receita	
1 Paula	R\$ 2.580,00	
2 Camila	R\$ 1.960,00	
3 Sabrina	R\$ 1.900,00	
4 Jéssica	R\$ 1.850,00	
5 Helena	R\$ 1.760,00	
6 Amanda	R\$ 1.660,00	
7 Larissa	R\$ 1.500,00	
8 Simone	R\$ 1.450,00	
9 Júlia	R\$ 1.400,00	
10 Andressa	R\$ 1.250,00	

**Fonte: Elaborado pela autora (2023).**

## 5 CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo desenvolver o protótipo de um aplicativo de agendamento de compromissos e venda de produtos para profissionais autônomos. A documentação do projeto, o que inclui o levantamento de requisitos, os estudos de casos de uso, a arquitetura dos códigos, a estrutura do banco de dados e a prototipagem, foi desenvolvida conforme as etapas da engenharia de software.

Com o aplicativo, o usuário pode cadastrar seus clientes, serviços e produtos, caso realize as duas modalidades de trabalho ou apenas uma das duas. As funcionalidades relacionadas com o agendamento permitem que o usuário possa gerenciar seu compromisso de forma simples e eficiente. O calendário ajuda na organização e no planejamento eficaz. A criação da comanda auxilia o usuário no controle e organização de seus recebimentos. Os relatórios gerados pelo sistema ajudam o usuário a gerir as finanças do seu negócio.

O diagrama de organização dos códigos, com o padrão MVCS, facilitará o desenvolvimento, a manutenção e a evolução do aplicativo, além de permitir a reutilização de componentes, uma vez que os códigos são organizados em camadas.

O design foi pensado para manter a consistência e a usabilidade, com o caso do *Material Design* para a plataforma Android. E o layout intuitivo, lógico e de fácil navegação.

A escolha pelo FlutterFlow como ferramenta para a criação das telas possibilitou o desenvolvimento de um protótipo interativo, o qual pode ser usado como um Produto Mínimo Viável (MVP) para testar a ideia de negócio do aplicativo “Agenda App”. Além de ser uma ferramenta de fácil utilização, essa plataforma de desenvolvimento de aplicativos possibilita a transferência dos códigos para o Flutter. Neste, os códigos podem ser aperfeiçoados e desenvolvidos para garantir que todas as funcionalidades do aplicativo sejam operantes.

Além dos testes de usuário, do desenvolvimento do sistema e da publicação do aplicativo nas principais lojas (Google Play Store e Apple Store), trabalhos futuros podem incluir no sistema algumas funcionalidades como: (1) opções de marketing

para divulgar promoções, eventos, novidades, mudanças, período de férias, etc.; (2) mensagens de lembrete dos compromissos; (3) gerenciamento de estoque; (4) opção de crédito para o cliente; (5) interfaces personalizadas de acordo com a área de atuação do profissional e (6) disponibilizar versão Web. Também pode ser implementado futuramente, um sistema dividido em módulos, sendo um módulo para o administrador do estabelecimento e outro para os colaboradores.

## REFERÊNCIAS BIBLIOGRÁFICAS

AGENDA BOA. Disponível em: <<https://www.agendaboa.com/>>. Acesso em 31 de ago. 2023.

ANDROID STUDIO. Disponível em: <<https://developer.android.com/studio>>. Acesso em: 07 de set. 2023.

APLICATIVO apoia profissionalização de pequenos negócios. **Folha de S.Paulo**, São Paulo, 23 de dez. de 2022, às 12h00. Disponível em: <<https://www1.folha.uol.com.br/folha-social-mais/2022/12/aplicativo-apoia-profissionalizacao-de-pequenos-negocios.shtml>>. Acesso em: 03 de set. de 2023.

FIREBASE. Disponível em: <<https://firebase.google.com/>>. Acesso em: 07 de set. 2023.

FLUTTER(a). Disponível em: <<https://flutter.dev/>>. Acesso em: 29 de maio de 2023.

FLUTTER(b). Disponível em: <<https://docs.flutter.dev/data-and-backend/firebase>>. Acesso em: 08 de set. 2023.

FLUTTERFLOW. Disponível em: <<https://flutterflow.io/>>. Acesso em: 30 de ago. 2023.

IBGE. Indicadores IBGE. **Pesquisa Nacional por Amostra de Domicílios Contínua Segundo Trimestre, 2023**. Disponível em: <<https://biblioteca.ibge.gov.br/index.php/biblioteca-catalogo?view=detalhes&id=72421>>. Acesso em: 02 nov. 2023.

MICROSOFT. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 05 de jun. 2023.

MINHA AGENDA. Disponível em: <<https://minhaagendaapp.com.br/>>. Acesso em: 31 de ago. 2023.

NUNES, S. E.; MOURA, R. A. P. **Programação em banco de dados**. Editora e Distribuidora Educacional S.A., Londrina, 2018.



PESSANHA, Maure. Negócio de impacto apoia a profissionalização de pequenos empreendedores. **Estadão**, São Paulo, 29 de jun. de 2022, às 11h00. Disponível em: <<https://www.estadao.com.br/pme/blog-do-empendedor/negocio-impacto-profissionalizacao-pequenos-empresarios/>>. Acesso em: 03 de set. 2023.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de software**: uma abordagem profissional. Editora: AMGH Editora Ltda. Porto Alegre, 2021.

ROSE, R. **Flutter and Dart Cookbook**. Developing Full-Stack Applications for the Cloud. Editora: O'Reilly Media, Inc. United States, 2022.

SEBRAE. **Qual a diferença entre MEI e profissional autônomo?** 2022. Disponível em:< <https://sebrae.com.br/sites/PortalSebrae/artigos/qual-a-diferenca-entre-mei-e-profissional-autonomo,71560d01760f1710VgnVCM1000004c00210aRCRD>>. Acesso em: 19 de out. 2023.

SELFWORK. Disponível em:<<https://www.selfworks.com.br/>>. Acesso em: 31 de ago. 2023.

SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Addison-Wesley, 2011.

SOMMERVILLE, I. **Engenharia de Software**. 10. ed. São Paulo: Pearson Education do Brasil., 2018.

TUA AGENDA. Disponível em: <<https://tuaagenda.com/>>. Acesso em: 31 de ago. 2023.

TOQEER, A. M. **MVC and MVCS**: software engineering. LinkedIn. 26 de fevereiro de 2017. Disponível e:< <https://www.linkedin.com/pulse/mvc-mvcs-software-engineering-asher-toqeer/>>. Acesso em: 18 de out. 2023.

VISUAL STUDIO CODE. Disponível em:<<https://visualstudio.microsoft.com/pt-br/#vscode-section>>. Acesso em: 29 de maio de 2023.

## APÊNDICE A – EXEMPLO DAS TELAS COMANDA DE SERVIÇO E DE VENDA PREENCHIDAS

The image displays two screenshots of a mobile application interface. The left screenshot shows a 'Comanda de Serviço' (Service Order) screen for 'Agenda App' on September 18, 2023, at 08:00. It lists a client named Camila and a service of 'Mão' (Hand). There are options to add other services (like 'Maquiagem') and products (like 'Creme mãos'). The total value is R\$ 182,80, with a discount of R\$ 2,80, resulting in a total to receive of R\$ 180,00. The status is 'Aberto' (Open). The right screenshot shows a 'Comanda de Venda' (Sale Order) screen for 'Agenda App' on September 21, 2023, at 16:00. It lists a client named Amanda and two products: 'Esmalte Kors' (quantity 1) and 'Batom Mac' (quantity 2). There is an option to add more products. The total value is R\$ 279,00, with a discount of R\$ 9,00, resulting in a total to receive of R\$ 270,00. The status is 'Aberto' (Open).

**Comanda de Serviço**

Agenda App  
Comanda de Serviço

18 de setembro de 2023  
Segunda-feira, 08:00

Cancelar

**Detalhes do Serviço**

Cliente: Camila  
Serviço: Mão

Adicionar outro serviço

Adicionar serviço + Maquiagem

Adicionar venda de produto

Adicionar produto + Creme mãos

Valor Total R\$ 182,80

Inserir Desconto R\$ 2,80

Total a Receber R\$ 180,00

Aberto Pago

**Comanda de Venda**

Agenda App  
Comanda de Venda

21 de setembro de 2023  
Quinta-feira, 16:00

Cancelar

**Detalhes da Venda**

Cliente + Amanda

Produto: Esmalte Kors  
Quantidade - 1 +

Produto: Batom Mac  
Quantidade - 2 +

Adicionar produto +

Valor Total R\$ 279,00

Inserir Desconto R\$ 9,00

Total a Receber R\$ 270,00

Aberto Pago

Fonte: Elaborada pela autora (2023)