

SoftHair - Aplicativo de Gestão de Salões de Beleza e Clínicas de Estética

Cesar E. Silva¹, Luiz Guilherme S. Oliveira²

¹Instituto Federal do Paraná
Caixa Postal 15.064 – 91.501-970 – Foz do Iguaçu – PR – Brazil

Abstract. *The beauty area is growing every day, with the increase in the number of customers and also the quantity of products sold, moving the economy of this sector. Brazil is currently the fourth in the world ranking in consumption of personal hygiene products, perfumes and cosmetics (FORBES). Only by this data is it possible to understand the great potential that the segment has. Even in the face of the severe crisis that Brazil faced, the annual average growth of this sector in the last 10 years was 4.1%. In 2019 a survey commissioned by Beauty Fair (International Professional Beauty Fair) in which around 500 thousand beauty salons were registered all over Brazil. A major factor that has contributed to such an expressive number is the significant increase in franchises in the sector, be it beauty salons, enamels or eyebrow design, for example. In fact, the number of beauty salons in the country is much higher, since almost 50% of the salons are still informal. And as we are in a digital age, where processes take place in an agile way, tools appear to make people's lives easier, including in the area of aesthetics and related services in the area of beauty. In AppSoftHair the proposed theme is that the provider that provides beauty services, will have the functionality to control their customer schedule, the services provided and scheduling and attendance management, which will ultimately add agility and security to the user's routines to the system.*

Resumo. *A área da beleza vem crescendo a cada dia, com o aumento no número de clientes e também quantidade de produtos vendidos movimentando a economia desse setor. O Brasil atualmente é o quarto no ranking mundial em consumo de produtos de higiene pessoal, perfumaria e cosméticos (FORBES). Só por este dado é possível entender o grande potencial que o segmento tem. Mesmo diante da forte crise que o Brasil enfrentou, a média anual de crescimento desse setor nos últimos 10 anos foi 4,1%. Em 2019 uma pesquisa encomendada pela Beauty Fair (Feira Internacional de Beleza Profissional) em que foram registrados cerca de 500 mil salões de beleza em todo o Brasil. Um grande fator que tem contribuído para um número tão expressivo é o aumento significativo de franquias no setor, sejam de salões de beleza, esmalterias ou design de sobrancelhas, por exemplo. Na verdade, o número de salões de beleza no país é muito maior, já que quase 50% dos salões ainda estão na informalidade. E como estamos numa era digital, onde os processos acontecem de forma ágil, ferramentas surgem para facilitar a*

vida das pessoas, inclusive na área da estética e serviços relacionados na área da beleza. No AppSoftHair o tema proposto é que o prestador que proverá os serviços de beleza, terá as funcionalidades de controlar a sua agenda de clientes, dos serviços prestados e gestão de agendamento e atendimentos, que por fim agregará agilidade e segurança nas rotinas do usuário ao sistema.

1. Introdução

O setor de beleza no Brasil é um dos mais promissores. São vários aspectos que colaboram para esse fato: inclusão das classes “D” e “E” decorrente do aumento de renda, inserção da mulher no mercado de trabalho, lançamento constante de novos produtos, elevação da expectativa de vida, entre outros.

A população apresenta alta demanda por produtos e serviços de qualidade e procedimentos específicos que contribuam para a elevação da autoestima e do bem estar e atendam necessidades de higiene pessoal.

Mesmo diante de cenários econômicos de crise nos anos recentes, o Brasil está em 4º lugar no ranking mundial com relação ao mercado de beleza e cuidados pessoais de acordo com a revista Forbes. Segundo o IBGE, o brasileiro gasta mais com beleza do que com comida. Então, este é um dos melhores ramos para se investir atualmente. (SALOMÃO, 2020).

Este trabalho tem como objetivo o desenvolvimento de um aplicativo mobile para gestão de salões de beleza e clínicas de estética, destinado a controlar e organizar os serviços de rotina em tal ambiente. Este público foi escolhido visto o crescente número de micro empreendedores individuais e também de micro empresas.

Os benefícios de informatizar as atividades relacionadas a esse nicho de mercado é fundamental na gestão desse tipo de empreendimento. As facilidades são altamente positivas, automatizando os processos, se reúnem as informações e rotinas do estabelecimento, como fluxo de agendamentos aos clientes, cadastro de funcionários, controle e gestão de serviços, levantamento de vendas e calculo de comissão.

Este aplicativo em fase de elaboração se deve a uma continuidade de um projeto em versão web que foi desenvolvido com o proposta de resolver as dificuldades em controlar as rotinas de um salão de beleza, com as duas modalidades dos sistemas implantados, isto agregara segurança e praticidade na gestão dos serviços prestados.

2. Cenário Atual

O salão de beleza utilizado como modelo no desenvolvimento do sistema web teve como objetivo liquidar com as antigas agendas de papel, nas quais haviam muitas rasuras e letras mal escritas. Este projeto de software de um aplicativo mobile tem como interesse integrar ao sistema web, funcionalidades e rotinas específicas deste mercado, afim de resolver problemas relacionados a falta de um sistema informatizado, com o intuito de trazer robustez e segurança nos serviços prestados.

3. Metas

Para entregar todas as funcionalidades do aplicativo, seguiu-se o planejamento do projeto, atentando-se ao cronograma. Dividindo-se as tarefas entre os membros da equipe e efetuando-se o versionamento de código para garantir agilidade e consistência na construção do app. Abaixo segue a lista de funções principais do software.

- Cadastro de Usuários para acesso ao Sistema;
- Cadastro de atividades;

- Controle de agendamento do Cliente;

4. Requisitos e Regras de Negócio

4.1. Requisitos Funcionais

Tabela 1. Requisitos Funcionais

Código	Requisito	Descrição do requisito funcional
RF01	Cadastrar prestador	O cadastro dos prestadores no sistema deverá ser efetuado com nome, e-mail e senha. Esse cadastro será realizado pelo administrador.
RF02	Definir acesso e permissões	Administrador concede acesso a determinados recursos do sistema para uma entidade cadastrada.
RF03	Cadastrar serviços	Cadastra os tipos de serviços realizados pelo <i>App</i> . O sistema permitira a manutenção de inclusão, alteração, consulta e exclusão de dados, essas funcionalidades serão a cargo do administrador.
RF04	Cadastrar clientes	Esse cadastro será atrelado ao tipo de serviço, ao prestador que ira executar a atividade. A manutenção dos dados pelo sistema permitira a inclusão, exclusão, consulta e alteração, a cargo do administrador e do prestador de serviço.
RF05	Cadastrar agendamento	Uma atividade pode ser agendada para ser realizada por um prestador. Apenas usuários podem registrar um agendamento.
RF06	Cadastrar atendimento	O atendimento é um agendamento com alguns campos adicionais como valor dos procedimentos, valor dos produtos, descrição detalhada dos procedimentos para consulta futura.

4.2. Requisitos Não Funcionais

Tabela 2. Requisitos Não-Funcionais

Código	Descrição do requisito não-funcional
RNF01	O <i>framework Expo</i> será utilizado no desenvolvimento do aplicativo
RNF02	O sistema utilizara banco de dados <i>MySQL</i>
RNF03	O sistema utilizara <i>API-REST</i> para a comunicação entre servidor web e o aplicativo
RNF04	O sistema será desenvolvido em linguagem <i>JavaScript</i>
RNF05	O aplicativo será implementado na plataforma <i>Android</i> na versão 10
RNF06	<i>Validação de token JWT (JSON WEB TOKEN) será implementado no backend Laravel, para autenticação de requisições entre as duas partes servidor web e o aplicativo</i>
RNF07	Através de validação de <i>token</i> , uma vez que o prestador estiver conectado, cada solicitação irá incluir o <i>JWT</i> , permitindo que o prestador continue acessando o sistema que foram liberados com tal <i>token</i> .

4.3. Regras de Negócio

Tabela 3. Requisitos Não-Funcionais

Código	Descrição regra de negócio
RN01	O acesso ao servidor web, para evitar manipulação de dados no banco de dados terá como protocolo de segurança a implementação de geração de <i>token</i> , afim de barrar esta eventualidade.
RN02	Somente o administrador poderá manipular dados de serviços e prestadores
RN03	A alteração de senha de acesso ao aplicativo somente será realizada pelo prestador
RN04	Somente será permitido iniciar um serviço com agendamento criado
RN05	O agendamento se tornará um atendimento mediante a alteração do status.
RN06	O valor dos serviços prestados serão vinculados ao atendimento caso ele se concretize.
RN07	O agendamento que não se torna um atendimento tem seu status alterado para inativo.
RN08	O prestador de serviço terá um único cliente por atendimento.
RN09	O prestador de serviço terá um atendimento único por horário.

5. Metodologia

Inicialmente, o desenvolvimento do aplicativo App SoftHair foi idealizado com foco em um projeto advindo de uma versão web, pois envolve o atendimento de demandas as quais o segmento necessitam, com o intuito de contemplar agilidade e segurança nas rotinas e atividades de serviços relacionados a salões de beleza a implantação de uma software na área mobile.

Assim, a metodologia do projeto envolveu em basicamente três etapas:

(1) Desenvolvimento e testes da API Rest, que faz a comunicação por meio de requisições HTTP das operações lógicas do aplicativo, onde o aplicativo que é parte de front end(interface), a qual o usuário interage com este servidor backend; (2) levantamento das rotinas e atividades a qual o aplicativo envolveriam, para suprir as dificuldades no segmento da beleza que poderão agregar benefícios para os usuários deste software. (3) Implementação do banco de dados e as suas relações entre as entidades levantadas no escopo inicial do projeto.

As tecnologias computacionais adotadas foram: linguagem de programação PHP com uso do framework Laravel para implementação da API Rest, no lado backend da aplicação, agregando a API foi usado o método JWT(JSON Web Token)que permite a autenticação entre as requisições entre as duas aplicações, para os testes e agilidade na integração destes foi utilizada a ferramenta Postman. Para a implementação do aplicativo no lado frontend a linguagem de programação JavaScript com o framework Expo na plataforma Android, o MySql utilizado para armazenamento dos dados, servidor local Apache2 webserver, e os testes das chamadas da requisi e o ambiente de desenvolvimento o VSCode. As telas do aplicativo foram implementados no software de prototipagem Figma. Para controle de versão de código no GitHub.

6. Resultados e especificações da API Rest

6.1 API Rest



Figura 1. Tela do endpoint de API.

A tela da API está imprimindo uma mensagem “APP Soft Hair API, “status Connected”, quando se inicializa o servidor, na rota *localhost:8000/api/*, onde está o primeiro endpoint do aplicativo.

6.2 Testes de comunicação da API pela ferramenta Postman

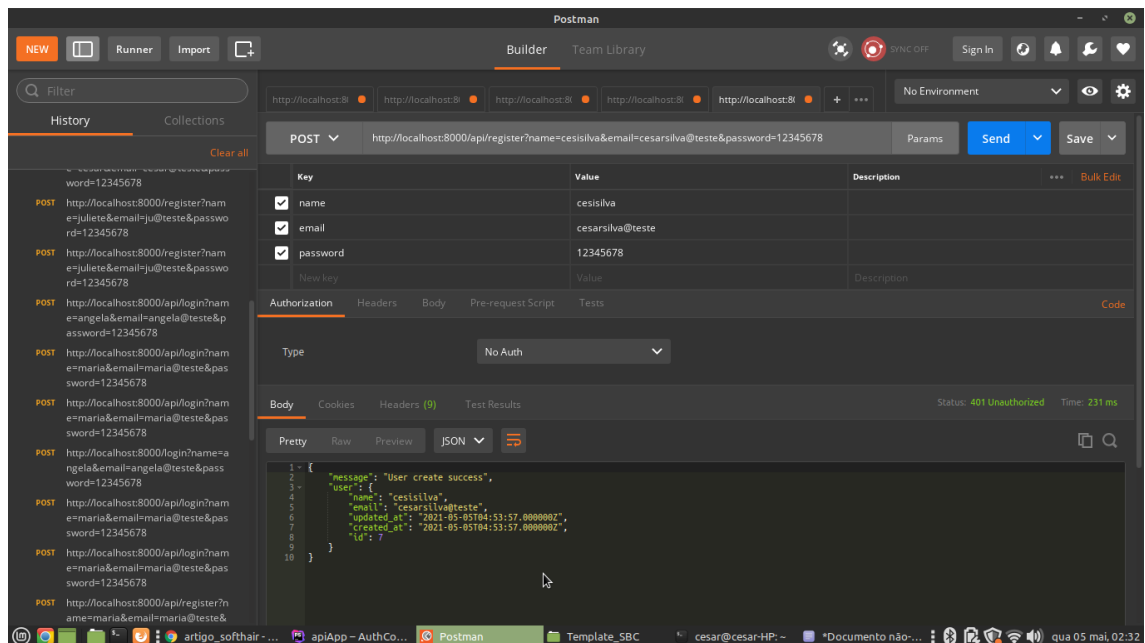


Figura 2. Tela do Postman com os resultados das chamadas HTTP.

Com a utilização da ferramenta *Postman*, pode ser realizada os testes das requisições da *API*, mesmo antes de implementar a interação do aplicativo *frontend* com o lado *backend*, passando parâmetros nos campos solicitados e no seu retorno imprime os dados que foram definidos na modelagem, assim as chamadas aos métodos da API podem ser corrigidas e testadas sem necessitar passar pelo aplicativo.

6.3 JWT (JSON Web Token)

É um método RCT 7519 padrão da indústria para realizar autenticação entre duas partes por meio de um token assinado que autentica uma requisição HTTP. Esse token é um código que armazena objetos JSON com os dados que permitem a autenticação das requisições.

Através de uma requisição HTTP ao endpoint de autenticação da API. Nela o usuário envia, no corpo da requisição dados como e-mail e senha.

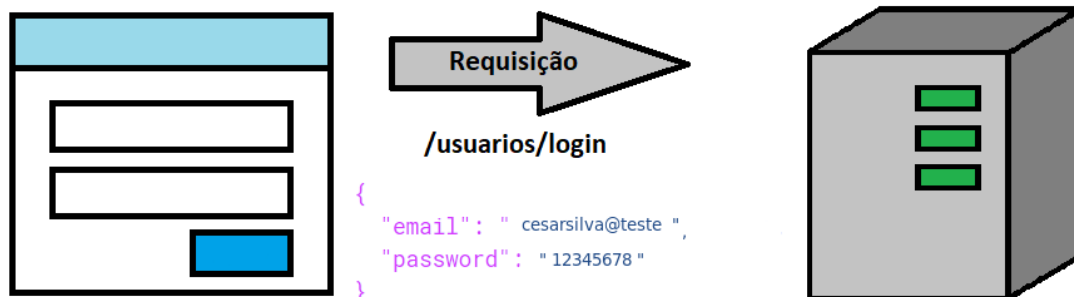


Figura 3. Usuário enviando requisição com dados de autenticação .

Uma vez que os dados enviados pelo usuário tenham sido autenticados no servidor, este criará um token JWT assinado com um segredo interno da API e enviará este token de volta ao usuário.

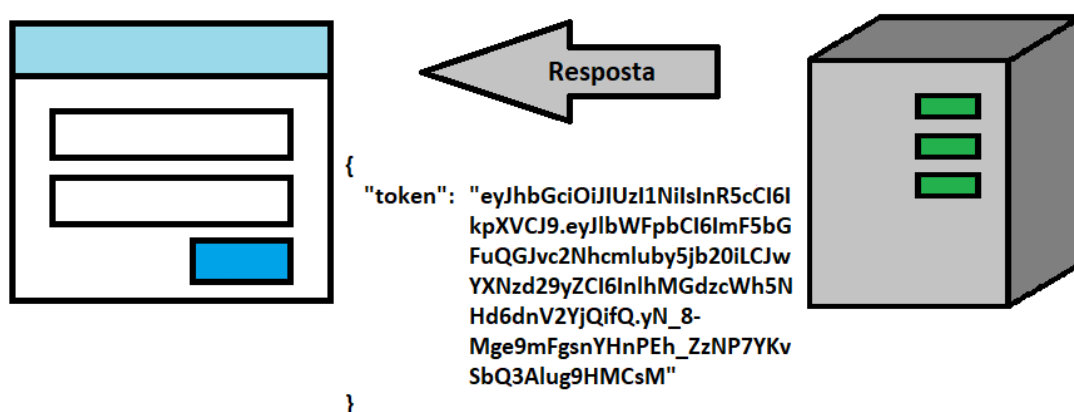


Figura 4. Servidor envia o token assinado para o usuário.

Provido com o token autenticado, o usuário possui acesso aos endpoints da aplicação que antes eram restritos.

7. Prototipação

Para a etapa de construção das telas utilizou-se da ferramenta BuilderX que possibilita criar o design das interfaces e também conta com o recurso de gerar componentes que podem ser integrados facilmente ao app. Um exemplo de componente pode ser visualizado na FIGURA 5.

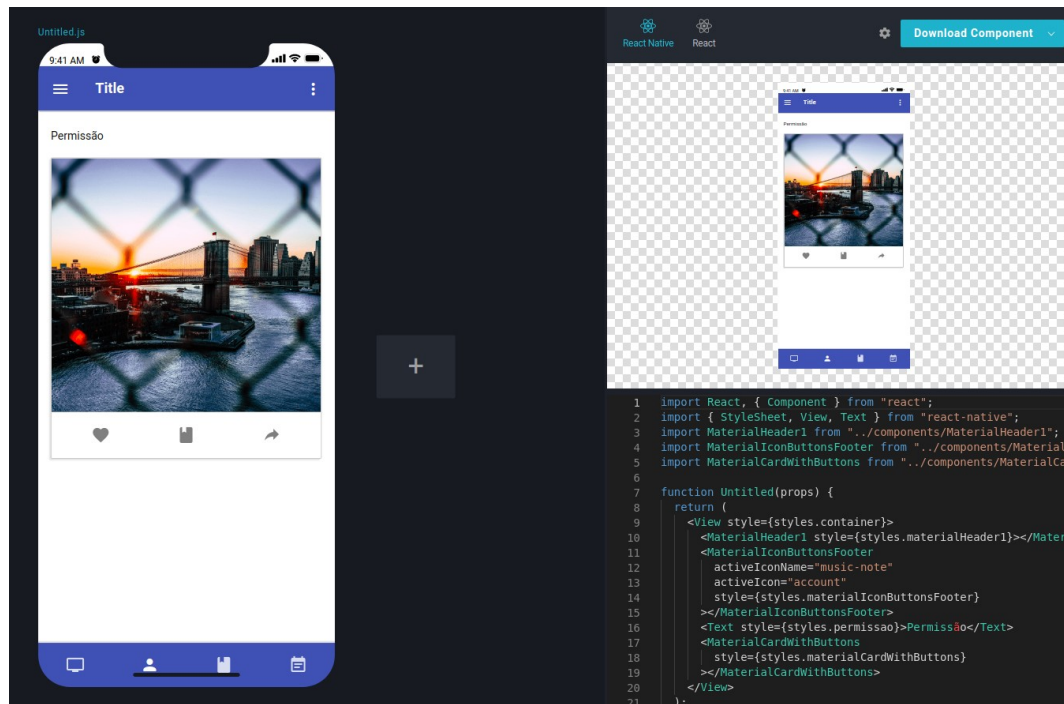


Figura 5. Tela de criação de interfaces do BuilderX

8. Resultados Obtidos

Demonstra-se neste capítulo as telas de interação do usuário com o sistema. Na FIGURA 6 observa-se a tela de login com os campos obrigatórios e-mail e senha e também com a opção de criar uma nova conta.

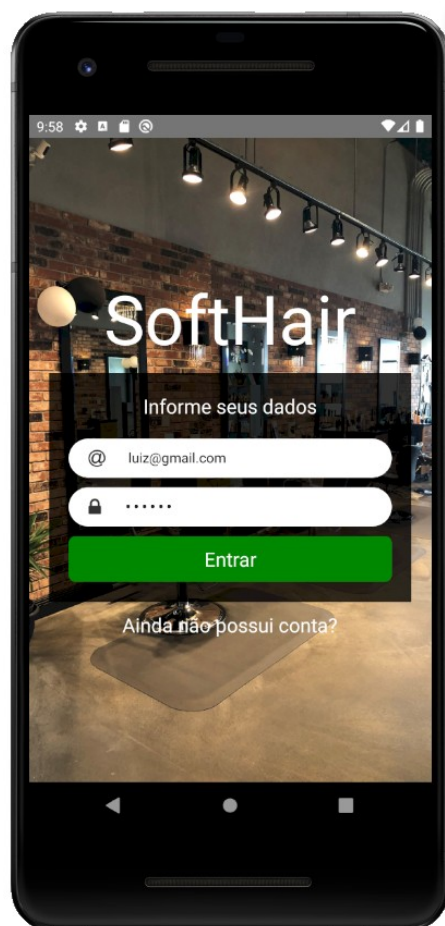


Figura 6. Tela de Login

Conforme mencionado anteriormente no capítulo de api, o login visto na FIGURA 6 utiliza-se do jwt para autenticar o usuário e permitir que ele realize as funções do sistema.

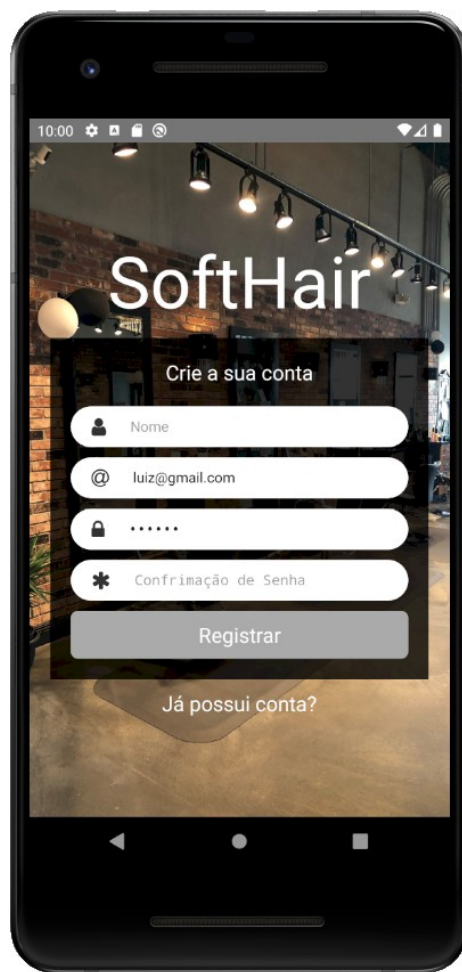


Figura 7. Tela de criação de conta

Na FIGURA 7 observa-se os campos necessários para criação de uma nova conta.

A FIGURA 8 demonstra a tela de lista de agendamentos que concede as seguintes funcionalidades de gerenciamento, inserção, visualização, atualização e remoção.

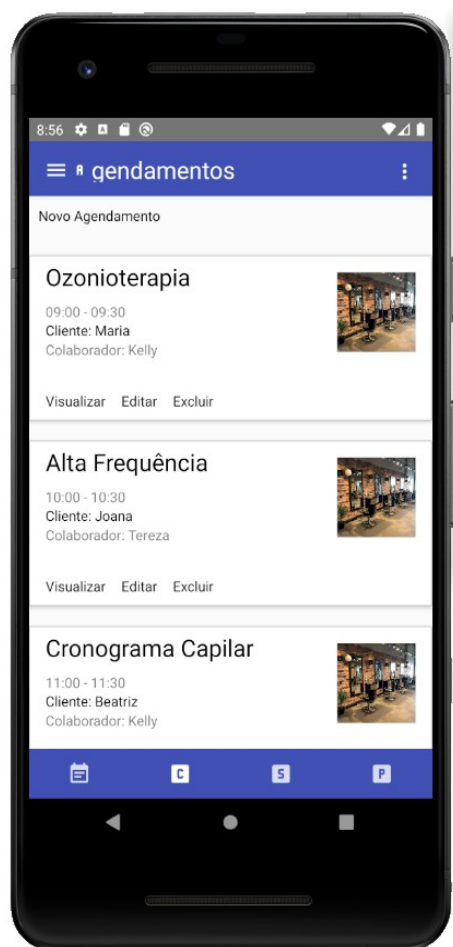


Figura 8. Tela de Lista de Agendamentos

Na FIGURA 9 observa-se a tela de manipulação de atendimento, o agendamento se torna um atendimento e permite que nele sejam acrescentados mais informações como produtos vendidos, serviços que foram realizados além do previsto e uma descrição técnica de como foi realizado o atendimento.

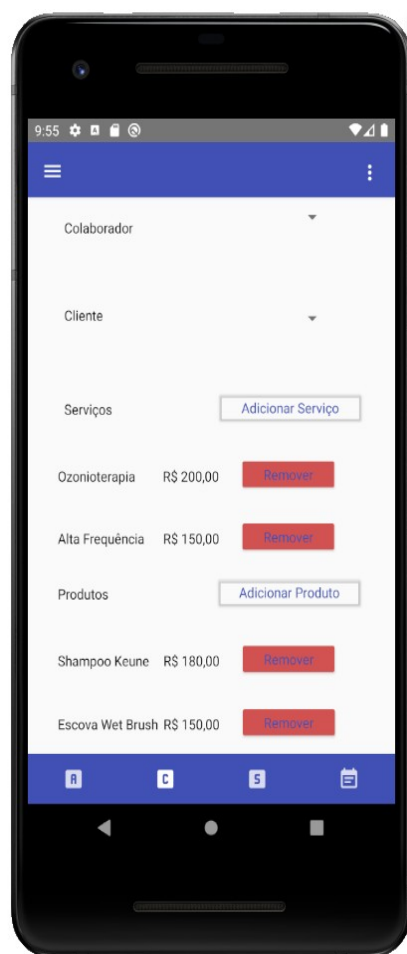


Figura 9. Tela de Cadastro de Atendimento

9. Referências

BuilderX Introduction (2021), <https://builderx.io/learn>.

Dall'Oglio, Pablo. (2007), PHP Programando com Orientação a Objetos, Novatec Editora, São Paulo.

Expo SDK 40.v Framework documentation (2020), <https://docs.expo.io/>.

JWT-Auth 1.0.5v Json Web Token (2020), <https://jwt-auth.readthedocs.io/en/develop/>.

Laravel 7.x^ Framework documentation (2021), <https://laravel.com/docs/7.x/releases>, March.

Postman API Platform version 5.5.5 (2020), <https://learning.postman.com/docs/>.

SALOMÃO, R. M. (2020) "O promissor Mercado de Salões de Beleza do Brasil". <https://buyco.com.br/blog/mercado/mercado-de-saloes-de-beleza>.

WEBER, M. (2020) “Brasil é o quarto maior mercado de beleza e cuidados pessoais do mundo”. <https://www.forbes.com.br/principal/2020/07/brasil-e-o-quarto-maior-mercado-de-beleza-e-cuidados-pessoais-do-mundo>.