



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE SISTEMAS E COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

YAGO DA COSTA TAVARES

**SISTEMA DE GESTÃO ONLINE PARA EMPRESA
NO RAMO DE BELEZA**

CAMPINA GRANDE - PB

2019

YAGO DA COSTA TAVARES

**SISTEMA DE GESTÃO ONLINE PARA EMPRESA
NO RAMO DE BELEZA**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em Ciência
da Computação.**

Orientador: Professor Dr. Kyller Costa Gorgônio.

CAMPINA GRANDE - PB

2019



T231s Tavares, Yago da Costa.
 Sistema de gestão online para empresa no ramo de
 beleza. / Yago da Costa Tavares. - 2019.

 10 f.

 Orientador: Prof. Dr. Kyller Costa Gorgônio.
 Trabalho de Conclusão de Curso - Artigo (Curso de
 Bacharelado em Ciência da Computação) - Universidade
 Federal de Campina Grande; Centro de Engenharia Elétrica
 e Informática.

 1. Gestão empresarial. 2. Serviços de beleza. 3.
 React Native. 4. Aplicativo mobile. 5. Desenvolvimento
 de aplicação androide. I. Gorgônio, Kyller Costa. II.
 Título.

CDU:004(045)

Elaboração da Ficha Catalográfica:

Johnny Rodrigues Barbosa
Bibliotecário-Documentalista
CRB-15/626

YAGO DA COSTA TAVARES

SISTEMA DE GESTÃO ONLINE PARA EMPRESA

NO RAMO DE BELEZA

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

**Professor Dr. Kyller Costa Gorgônio
Orientador – UASC/CEEI/UFCG**

**Professor Dr. João Arthur Brunet Monteiro
Examinador – UASC/CEEI/UFCG**

**Professor Dr. Tiago Lima Massoni
Examinador – UASC/CEEI/UFCG**

Trabalho aprovado em: 25 de novembro 2019.

CAMPINA GRANDE - PB

Sistema de gestão online para empresa no ramo de beleza

Trabalho de Conclusão de Curso

Yago da Costa Tavares
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
yago.tavares@ccc.ufcg.edu.br

RESUMO

É essencial que atualmente estabelecimentos tenham um sistema de gestão online. Porém nem todos oferecem esses serviços e poucos usam na plataforma mobile. Assim sendo, este trabalho tem como objetivo desenvolver uma aplicação Android, com intuito de dar suporte ao proprietário e funcionários de estabelecimentos com foco no ramo da beleza. Com essa ferramenta, será mais fácil gerir empresas desse ramo, assim como facilitará o trabalho diário da equipe profissional.

PALAVRAS-CHAVE

Gestão empresarial, serviços de beleza, React Native, aplicativo mobile

1. INTRODUÇÃO

Para que uma micro e pequena empresa seja exitosa, é imprescindível que ela seja bem organizada, com uma eficiente gestão financeira, como também que seu gestor tenha total controle sobre ela. Acompanhando de perto todo o fluxo de entrada e saída de recursos, para que só assim possa tomar decisões melhores para seu estabelecimento. Porém em empresas deste porte, nem sempre é possível ter departamentos específicos responsáveis por todas as áreas administrativas, tanto por causa da estrutura quanto da falta de recursos para mantê-los.

Por esse motivo, empresários buscam cada vez mais por sistemas de gestão empresarial que permitem administrar os setores da empresa de forma unificada e simples. Segundo a 30ª edição Pesquisa do Uso de Tecnologia nas Empresas da FGV [1], feita em 2019, o gasto e investimento em TI cresceu para 7,9% da receita das empresas. Além disso, a mesma pesquisa aponta que o aumento em 1% no investimento em tecnologia tem aumentado o lucro da empresa em 7% após 2 anos.

Um setor que possui grande quantidade de pequenas empresas é o da beleza. Este que tem um grande número de vendas e que vem em constante crescimento no Brasil tem como agravante a falta de gestão. Pesquisas do Sebrae-SP [2] mostram que 82% dos

proprietários têm experiência na área, porém apenas 22% utilizam algum sistema de gestão financeira. Ou seja, se preocupam muito com as vendas, porém, por falta de estudo não sabem como administrar seu dinheiro.

Diante dessas evidências, o objetivo deste trabalho é criar um sistema que torne esse processo mais fácil, de forma a centralizar um serviço que hoje demanda muito tempo para ser feito. Todas as entradas e saídas do empreendimento serão cadastradas e poderão ser vistas através de gráficos. Será possível também gerenciar as vendas com um sistema de agendamento dos clientes, através do qual os funcionários poderão marcar horários para os serviços que os clientes pretendem realizar, para inovar e otimizar todo o gerenciamento do estabelecimento.

2. SOLUÇÃO

Já existem no mercado sistemas de gestão online que disponibilizam contas para que empresas façam uso e paguem um valor mensal pelo serviço utilizado. No entanto, a maioria delas dispõe apenas de plataforma web, com responsividade para mobile, ou tem aplicativos integrados, mas com funcionalidades limitadas. Segundo a pesquisa The State of Mobile 2019 realizada pelo App Annie [3], cerca de 96% da Geração Z (16 a 24 anos) já possui smartphone, se envolvem 30% mais com seus aplicativos e gastam 20% a mais do que o restante da população. Além disso, foi constatado um aumento em 50%, de 2016 para 2018, onde é afirmado que os usuários hoje gastam em média 3 horas do seu dia em aplicativos móveis. O Brasil é um dos países que estão na fase de experimentação, onde o número de downloads é impulsionado pelo aumento de novos dispositivos móveis (geralmente mais modernos e com mais espaço de armazenamento). Constatando o crescimento que os aplicativos móveis vêm tendo no Brasil e no mundo, além da sua importância dele no mercado, podemos dizer que aplicativos são essenciais para as empresas que almejam uma maior visibilidade e crescimento.

Para levantamento de requisitos, foi feita uma pesquisa em dois estabelecimentos no ramo de beleza em Campina Grande, LaBelle Spa e RR Clothing and Barber Shop, como também uma análise em um sistema de gestão online já existente, Trinks. Com base nas funcionalidades vistas, foram escolhidas as essenciais e de maior serventia para os gestores dos estabelecimentos. Diante

Os autores retêm os direitos, ao abrigo de uma licença Creative Commons Atribuição CC BY, sobre todo o conteúdo deste artigo (incluindo todos os elementos que possam conter, tais como figuras, desenhos, tabelas), bem como sobre todos os materiais produzidos pelos autores que estejam relacionados ao trabalho relatado e que estejam referenciados no artigo (tais como códigos fonte e bases de dados). Essa licença permite que outros distribuam, adaptem e evoluam seu trabalho, mesmo comercialmente, desde que os autores sejam creditados pela criação original.

dessa imprescindibilidade, criamos o aplicativo com as funcionalidades essenciais para esse tipo de trabalho:

- **Contas de administrador e funcionários:** algumas funcionalidades na empresa só podem ser feitas por pessoas específicas. Dessa forma, deixamos o aplicativo com dois tipos de contas, a de administrador e a de funcionário, sendo que a segunda terá suas funcionalidades limitadas dentro do aplicativo.
- **Cadastro de clientes:** temos o cadastro no aplicativo de todos os clientes do estabelecimento. Com isso fica mais fácil um possível contato com o cliente que não retornam há a muito tempo, por exemplo.
- **Cadastro dos serviços:** todo administrador pode visualizar, cadastrar, editar e remover os serviços que o estabelecimento oferece. Cada serviço tem seu nome, valor, descrição e duração em minutos.
- **Sistema de agendamento:** muitas empresas de beleza trabalham com hora marcada. Dessa forma, colocamos o sistema de agendamento para que o estabelecimento possa gerenciá-lo no próprio aplicativo, conectando-o diretamente com o setor financeiro (fechamento de conta e fluxo de caixa).
- **Setor financeiro:** com essa funcionalidade, será possível gerenciar todo o fluxo de caixa da empresa, podendo-se, assim, tomar decisões financeiras mais facilmente, baseando-se nos dados do aplicativo.
- **Gráficos demonstrativos:** baseados nos dados do setor financeiro e no fechamento de contas, podemos demonstrar, em gráfico de linha, o lucro da empresa em determinado período. Como também podemos ver, em um gráfico de colunas, quais são os serviços mais realizados naquele estabelecimento. Com isso, as decisões poderão ser tomadas de forma mais segura, sendo baseadas em dados reais.

Com essas funções, a aplicação se torna útil para grande parte dos estabelecimentos de pequeno e médio porte.

2.1 ARQUITETURA

Para o desenvolvimento da aplicação, separamos nosso sistema em duas partes, *front-end* e *back-end*, sendo o *front-end* a parte que o usuário final utiliza (mobile) e que se comunica através de requisições GET, PUT, DELETE e POST com o *back-end*.

Já o *back-end* é uma API REST que é responsável por gerenciar e persistir os dados da aplicação no banco de dados.

Por motivos dos dados serem totalmente relacionados, escolhemos usar o banco de dados relacional SQLite.

2.1.1 BACK-END

2.1.1.1 TECNOLOGIAS DO BACK-END

No *back-end*, foi escolhida a linguagem *Node.js* junto com o framework *Express* para a criação da API REST, por ser mais fácil de usar e ter uma comunidade mais ativa no ramo da computação. É baseada em JavaScript, uma linguagem fácil e com boa documentação na web, além de que o *front-end* da aplicação foi feito com *React Native*, que também é baseado em JavaScript, facilitando assim a programação para o desenvolvedor.

Foi escolhido também um banco de dados relacional, SQLite, junto com o ORM *Sequelize* para uma manipulação e persistência dos dados. Além disso, utilizamos o *JWT* como *token* de autenticação nas requisições ao *back-end*.

2.1.1.2 ESTRUTURA DO BACK-END

O *back-end* do sistema é uma API REST que fornece os dados à aplicação mobile, ele está conectado a um banco de dados relacional SQLite, a quem faz requisições para criar, ver, editar e deletar os dados do sistema. Seguindo padrões já utilizados para projetos em *node.js* [4], a estrutura foi dividida em quatro partes, são elas: *controllers*, *middlewares*, *models* e *routes*. [Figura 1]

Os *controllers* são a parte principal do servidor, onde é feito o gerenciamento e manipulação dos dados. Recebe os dados vindo das rotas, manipula-os e faz as requisições ao banco de dados através dos *models*.

O *middleware* é onde ficam os interceptadores, que neste caso temos apenas o *security*, responsável por interceptar as rotas para verificar a autenticidade dos *tokens* que vêm nas requisições feitas pelos usuários no *front-end*.

Models são por onde são feitas as requisições ao banco de dados. São responsáveis por definir e especificar todos os modelos do banco. Recebem os dados vindos do controller e permitem o acesso para os dados serem coletados, gravados, editados e deletados.

Por último, na pasta *routes* é onde ficam localizadas todas as rotas do servidor nas quais o *front-end* faz as requisições. É a porta de entrada do *back-end*. Recebe as requisições, quando necessário, faz a autenticação da chamada no *middleware* e envia os dados para o controller.

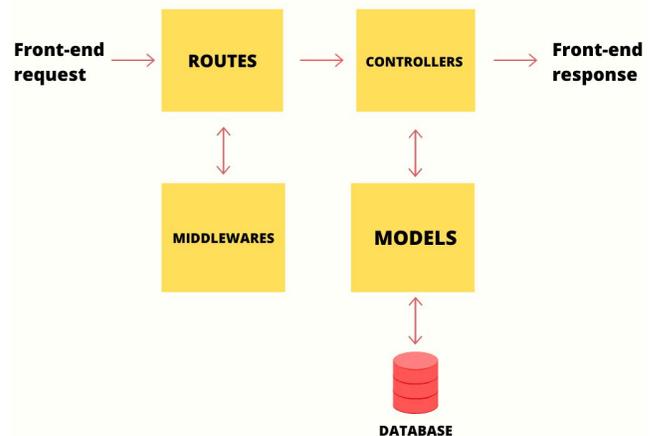


Figura 1 - Estrutura do *back-end* da aplicação

2.1.2 FRONT-END

2.1.2.1 TECNOLOGIAS DO FRONT-END

Utilizamos a tecnologia *React Native* para o *front-end* pelo motivo de ser uma linguagem nativa, podendo criar a aplicação tanto para dispositivos Android como para IOS, com o reaproveitamento do mesmo código. A aplicação do sistema de gestão foi desenvolvida apenas para Android, porém pode ser estendida futuramente para

IOS com mais facilidade. É uma linguagem que assim como o *Node* no *back-end*, também é baseada em JavaScript. Tem uma alta performance e está em alta na comunidade da programação. Grandes aplicativos como Facebook, Instagram, Nubank, entre outros, também foram desenvolvidos com *React Native*. Toda a parte de design do aplicativo foi baseada nos components do Native Base [5] e da própria API do React Native [6]

2.1.2.2 ESTRUTURA DO FRONT-END

A parte mobile tem uma estrutura já indicada pela documentação e tutoriais sobre *React Native* na web. Como mostrado na Figura 2, podemos ver que a aplicação é dividida em quatro partes. Que são: *Components*, *Pages*, *Routes* e *Services*.

Pages são a principal parte do *front-end*, onde ficam localizadas todas as telas do aplicativo e as configurações de navegação entre elas. Em nossa solução utilizamos o *StackNavigator* e o *DrawerNavigator*. Especificamos nas classes de navegação as restrições de acesso para usuários do tipo administrador e do tipo funcionários.

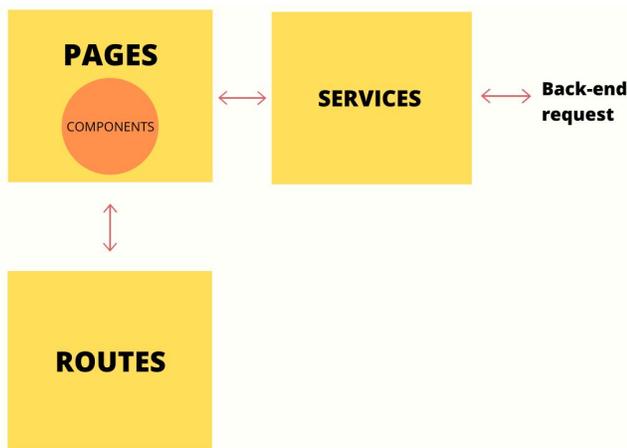


Figura 2 - Estrutura da aplicação mobile

Os *Components*, como o próprio nome diz, são onde se localizam todos os componentes da aplicação. O *React Native* permite essa componentização [7], facilitando assim o reuso destes nas páginas.

Nos *Routes* se localizam todas as rotas da aplicação. Através deles é que é feita a navegação entre telas do aplicativo.

E por fim os *Services*, onde é feita a conexão com o *back-end* e por onde são feitas as requisições ao mesmo, através do cliente *http Axios*.

2.1.3 LÓGICA DA APLICAÇÃO

Login: a aplicação tem dois tipos de login, e cada um tem suas restrições e acessos, Administrador e Funcionário. A conta de administrador é criada assim que os *migrations* do banco de dados são executados, ou seja, a aplicação sempre terá pelo menos um administrador. O administrador tem acesso a todas as

funcionalidades da página, podendo criar, ver, editar e excluir todos os dados do sistema. [Figura 3]

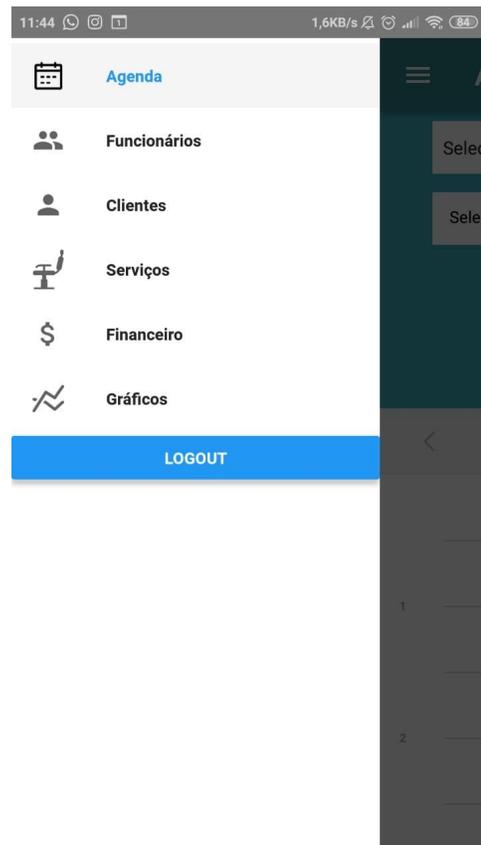


Figura 3 - Tela do menu da aplicação para conta de administrador

Já o funcionário tem algumas restrições. Ele não pode criar, editar e nem excluir um outro funcionário. Além disso, também não tem acesso às páginas do sistema financeiro, dos serviços e dos gráficos demonstrativos da empresa. [Figura 4]

Ao fazer login na aplicação, o *back-end* envia um *token* de autenticação para o *front-end* o qual irá guardar e usar em todas as requisições ao servidor, enviando sempre o *token* no cabeçalho.

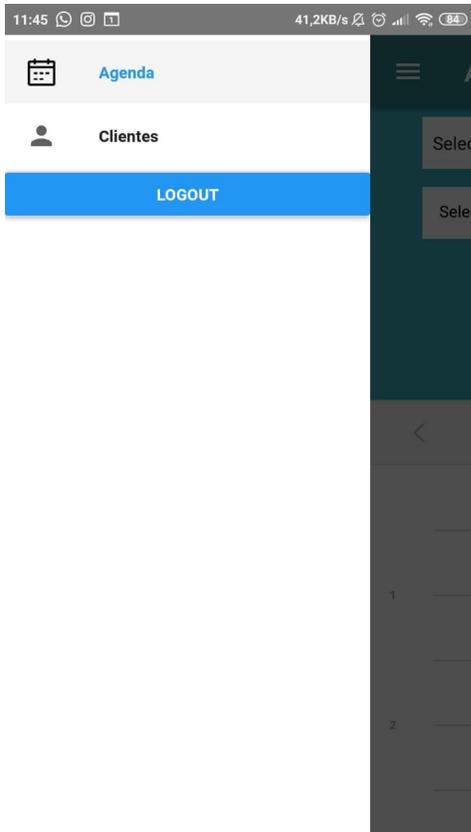


Figura 4 - Tela de menu da aplicação para conta de funcionário

Agendamento: a principal funcionalidade do sistema gira em torno do agendamento de serviços. O administrador ou funcionário pode criar um novo agendamento informando o nome do cliente, o serviço que irá fazer, o dia e hora que o cliente deseja. Com isso, o agendamento será criado e inserido no calendário de eventos. Nesse calendário, podemos ver todos os agendamentos cadastrados de acordo com o dia e hora, facilitando a checagem de serviços a serem realizados por dia. Ao clicar no agendamento, podemos editá-lo ou finalizá-lo. Ao finalizar, é preciso indicar a forma de pagamento. O valor será automaticamente inserido como entrada no fluxo de caixa. Na figura 5 podemos ver que no dia 13 de novembro temos 3 horários marcados, dois com serviço de unha e outro para cabelo.

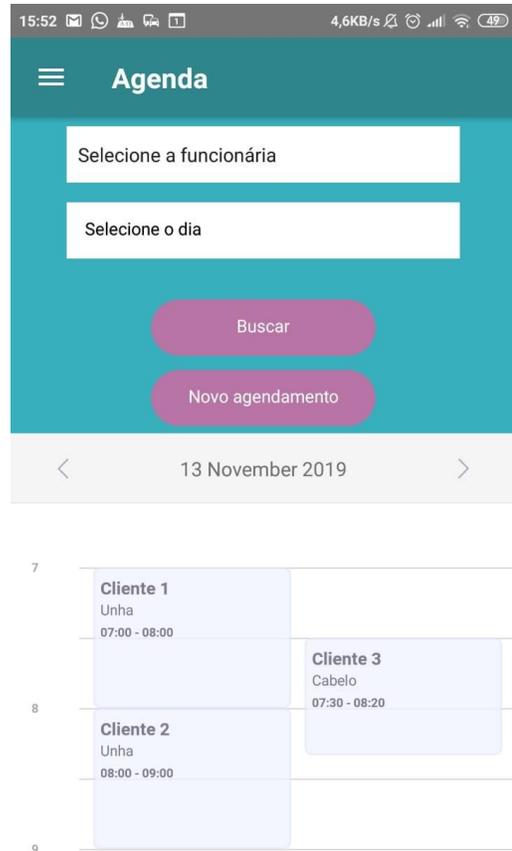


Figura 5 - Tela de agendamentos do dia

Setor financeiro: na tela do setor financeiro, podemos visualizar todas as contas de clientes que foram finalizadas em um determinado período de tempo. Também temos o fluxo de caixa de todos os dias, com o saldo do dia anterior, o dinheiro arrecadado em espécie e cartão, o saldo do caixa atual, a retirada de dinheiro que foi feita e o saldo final do caixa. Esses dados são lidos e colocados em gráficos demonstrativos, onde é possível analisar o comportamento financeiro da empresa. [Figura 6]

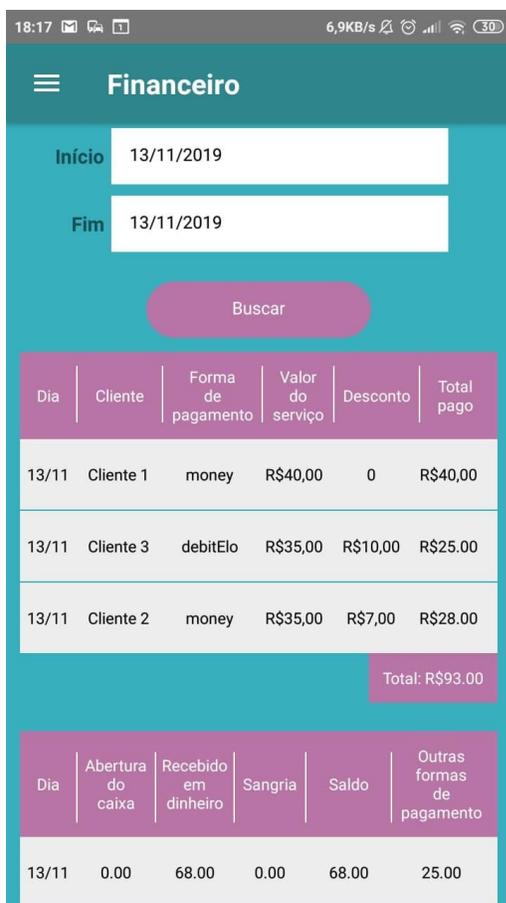


Figura 6 - Setor financeiro do aplicativo com todas as contas finalizadas do dia 13/11/2019 e o fluxo de caixa desse mesmo dia

Gráficos: Na página de gráficos, podemos selecionar um período de tempo (mínimo 2 dias) para acompanhar o histórico de vendas da empresa, por dia, passado em um gráfico de linha, como também em um gráfico de coluna, os serviços mais procurados pelos clientes. Dessa forma, será possível analisar os gráficos e verificar mais facilmente quais os dias do mês em que mais houve faturamento e quais os que precisam de um foco maior em divulgação e marketing, por exemplo. Já no outro gráfico, pode ser visto quais os serviços que estão sendo menos vendidos e, assim, criar uma promoção em cima deles. Na figura 7, podemos ver que foi feita a busca dos dias 12 à 14 de novembro. Na figura 8 podemos ver os dois gráficos, demonstrando o faturamento e o ranking de serviços dos dias que foram selecionados



Figura 7 - Tela de gráficos com período de início e fim da busca dos dados



Figura 8 - Gráficos demonstrativos sobre o faturamento e o ranking de serviços realizados no período

2.1.4 AMBIENTE DE PRODUÇÃO

O servidor e o banco de dados foram hospedados em uma instância na AWS, escolhemos essa tecnologia pela facilidade de hospedagem, pelo serviço gratuito nos primeiros 12 meses e pensando no crescimento do aplicativo, pois será mais fácil o escalonamento. O aplicativo não foi postado na *PlayStore*, pois será vendido para empresas privadas do ramo de beleza. Dessa forma, só poderemos postar na plataforma quando atualizarmos cores e logotipos destas empresas parceiras.

3. EXPERIÊNCIA

3.1 PROCESSO DE DESENVOLVIMENTO

Primeiramente, foi feita uma avaliação de quais tecnologias seriam usadas no sistema, escolhendo aquelas que tornassem o desenvolvimento mais rápido e fácil.

Posteriormente, montamos o escopo do projeto, decidindo quais as funcionalidades mais necessárias para o aplicativo, de forma que ficasse o mais completo possível dentro do tempo estimado.

Após isso, foi feito cronograma de desenvolvimento, colocando os dias e horários a serem trabalhados, para que gradativamente o sistema fosse sendo implementado.

Dessa forma todo o sistema foi sendo criado, até que, por fim, foi feito o *deploy* do *back-end* em uma instância da AWS. Assim como, foi gerado o apk do aplicativo [8] para que pudéssemos colocar em produção, para que fosse avaliado.

3.2 DESAFIOS

O maior desafio encontrado no desenvolvimento deste trabalho foi o de criar as funcionalidades que englobassem a maioria dos estabelecimentos de beleza, de forma que não fossem específicas de apenas um tipo único de empresa (salão de beleza, clínica de estética, etc).

Encontrar uma ferramenta de agendamento, com boa documentação e com interface usual para *React Native*, também foi uma tarefa difícil, pois é um componente muito específico e não muito utilizado em aplicativos.

4. AVALIAÇÃO

A aplicação depois de finalizada foi avaliada por 9 pessoas ligadas a este tipo de estabelecimento, onde duas dessas pessoas são donos de salões de beleza e sete são funcionários.

Foi feito um formulário de satisfação, onde foram feitos alguns questionamentos. Foram eles:

1. Você concorda que a aplicação é útil para estabelecimentos de beleza?
2. Você concorda que o aplicativo tem as funcionalidades essenciais para esse tipo de estabelecimento?
3. Como você avalia a usabilidade do aplicativo?
4. Como você avalia o design do aplicativo?
5. Qual a nota você daria para o aplicativo no geral?
6. Deseja fazer alguma observação a mais sobre a aplicação?

Nas perguntas de 1 a 5 foram colocadas escalas de 1 a 5 como opções de resposta, onde 1 significa “Discordo totalmente” e 5 “Concordo totalmente” para as questões 1 e 2. E para as questões 3, 4 e 5 as respostas equivalem a nota que deram para cada pergunta. Por fim, na pergunta 6 foi colocada uma caixa de texto para que opcionalmente eles escrevessem melhorias e observações em geral para o sistema.

Foi entregue um celular com o aplicativo instalado para que cada um pudesse utilizar como quisesse e por tempo indeterminado. Após isso, eles preencheram o formulário de satisfação.

Com base nos resultados da pesquisa, vimos que o design da aplicação é o principal item a ser melhorado, visto que seis das nove pessoas que responderam o formulário avaliaram com nota menor igual a 3 a usabilidade da aplicação e mais do que 50% colocaram nota 3 para o design do aplicativo. Ou seja, a maior necessidade no momento é deixar o aplicativo mais fácil de ser usado e com uma interface mais agradável.

Por outro lado, vimos que a aplicação é bastante útil para os estabelecimentos de beleza, pois 100% dos entrevistados concordaram totalmente com essa afirmação. Além disso, na pergunta de número 5, sete dos entrevistados avaliaram o aplicativo com nota máxima.

Outro ponto importante a ser comentado é que nas observações da questão 6 foram citadas funcionalidades que são necessárias na aplicação, como:

- Colocar mais gráficos demonstrativos;
- Funcionalidade para estabelecimentos que não trabalham com hora marcada e sim com ordem de chegada;
- Desenvolver área para acesso de clientes onde eles mesmos possam marcar seu horários.

A avaliação foi bastante proveitosa, pois foi discutido com os entrevistados alguns pontos que não tinham sido observados ainda, como este segundo ponto citado anteriormente.

Os dois empresários ficaram interessados na aplicação, contanto que fossem feitas as alterações que pontuaram.

5. TRABALHOS FUTUROS

Há abertura para implementar novas funcionalidades no aplicativo, deixando-o mais completo e atraente para empresários que visam comprar a aplicação. São elas:

- Acesso dos clientes ao sistema para que eles mesmos possam marcar seus horários.
- Com o acesso de clientes, colocar um sistema de *push notification* que envia cupons de desconto ou promoções exclusivas para determinados clientes.
- Sistema de *push notifications* para alertar os funcionários quando seus agendamentos estiverem próximos.
- Função de envio de mensagens por e-mail e/ou mensagens de texto para clientes, lembrando-lhes dos seus agendamentos.

- Cadastro de despesas da empresa, de forma separada, ligando diretamente com o fluxo de caixa e gerando gráfico demonstrativo.
- Melhorias no design da aplicação.

6. REFERÊNCIAS

- [1] FGV EASP, 2019. Pesquisa Anual do Uso de TI. Disponível em: <<https://eaesp.fgv.br/ensinoeconhecimento/centros/cia/pesquisa/>>. Acessado em: 15 de nov. de 2019.
- [2] Setor de beleza exige inovação constante e gestão. Sebrae, 2017. Disponível em: <<http://www.sp.agenciasebrae.com.br/sites/asn/uf/SP/setor-de-beleza-exige-inovacao-constante-e-gestao,30c0c6e73b6ef510VgnVCM1000004c00210aRCRD/>>. Acesso em: 20 de ago. de 2019.
- [3] App Annie, 2019. The State of Mobile 2019. Disponível em: <<https://www.appannie.com/en/go/state-of-mobile-2019/>>. Acessado em: 06 de set. de 2019.
- [4] Estrutura de diretórios e arquivos em projetos Node.js. Disponível em: <<https://walde.co/2016/10/24/estrutura-de-diretorios-e-arquivos-em-projetos-node-js/>>. Acesso em: 10 de set. de 2019
- [5] Nativebase, 2019. Components. Disponível em: <<https://docs.nativebase.io/Components.html#picker-def-headref/>>. Acesso em: 14 de nov. de 2019.
- [6] React Native, 2019. API. Disponível em: <<https://facebook.github.io/react-native/docs/activityindicator/>>. Acesso em: 14 de nov. de 2019.
- [7] FERNANDES, Diego. React do zero: componentização, propriedades e estado. Rocketseat, 2017. Disponível em: <<https://blog.rocketseat.com.br/react-do-zero-componentizacao-propriedades-e-estado/>>. Acesso em: 17 de out. de 2019.
- [8] LUZ, Kássio. React Native build release android. Tableless, 2018. Disponível em: <<https://tableless.com.br/react-native-build-release-android/>>. Acesso em: 14 de nov. de 2019