

Documentação Geral

Resumo do Sistema

Nome: Aplicativo mobile para gerenciamento de atendimentos e agendamento de serviços prestados KiAgenda.

Tipo: Aplicação mobile **Linguagem:** React Native

Acesso ao banco de dados: Sim (AsyncStorage)

KiAgenda

1) Introdução:

Este documento tem como objetivo descrever um aplicativo mobile para gerenciamento de atendimentos e agendamentos de serviços prestados, sua implementação e armazenamento no banco de dados pelo AsyncStorage.

O aplicativo deve permitir que os usuários realizem o gerenciamento dos agendamentos incluindo a descrição dos serviços que serão realizados.

Os dados devem ser armazenados em um banco de dados AsyncStorage e os usuários devem ser capazes de acessar a qualquer momento, mas somente offline.

O sistema deve permitir que os usuários realizem as seguintes ações:

Os usuários receberam uma mensagem de boas vindas, será exibido também um wizard com as principais funcionalidades do aplicativo sendo permitido que os usuários possam pular essa etapa.

Permitir que os usuários realizem o cadastro com os dados iniciais sendo que esses dados serão informados na primeira inicialização do aplicativo

Os usuários deverão informar o nome do estabelecimento que é obrigatório sendo CNPJ, logotipo e ramo da atividade são opcionais.

O ramo de atividade será obrigatório no caso de atividades pré-cadastradas podendo haver atividades comuns favoritas entre diferentes ramos;

Após o preenchimento dos dados iniciais, o aplicativo irá exibir a tela inicial.

- **Verificando a Primeira Inicialização:**

Ao ser iniciado, o aplicativo verifica se é a primeira vez que está sendo executado no dispositivo. Isso é feito por meio de um registro interno que indica se a inicialização já ocorreu anteriormente.

- **Mensagem de Boas-Vindas**

Se for identificado que esta é a primeira inicialização, o aplicativo apresentará uma mensagem amigável de boas-vindas ao usuário.

- **Apresentação das Principais Funcionalidades**

Após a mensagem de boas-vindas, o aplicativo exibirá uma rápida apresentação das principais funcionalidades por meio de um assistente wizard. Essa etapa tem o objetivo de orientar o usuário sobre como utilizar o aplicativo.

O usuário terá a opção de pular esta etapa se desejar.

- **Cadastro de Dados Iniciais**

Após a apresentação das principais funcionalidades, o aplicativo solicitará ao usuário que forneça alguns dados iniciais para configurar o aplicativo de acordo com as necessidades dele. Os dados a serem informados são os seguintes:

- **Nome do Estabelecimento (Obrigatório)**

O usuário deve informar o nome do estabelecimento ao qual o aplicativo está sendo associado. Este campo é obrigatório.

- **CNPJ (Opcional)**

O CNPJ do estabelecimento pode ser fornecido, mas não é obrigatório. Caso o usuário escolha não fornecer o CNPJ, o aplicativo continuará normalmente com as configurações padrão

- **Logotipo**

O usuário pode carregar o logotipo do estabelecimento, mas isso é opcional. Se o logotipo não for fornecido, o aplicativo utilizará o logotipo padrão.

- **Ramo de Atividade (Opcional)**

O usuário pode especificar o ramo de atividade do estabelecimento, mas isso também é opcional. O campo de ramo de atividade pode ser preenchido com informações como "geral", "escritório contábil", "advocacia", "salão de beleza", "oficina mecânica", etc.

- **Atividades Comuns Favoritas**

O aplicativo permite que o usuário marque atividades comuns como favoritas para facilitar o acesso rápido a elas. O usuário pode adicionar atividades comuns favoritas após preencher os dados iniciais.

- **Tela Inicial**

Após preencher os dados iniciais, o aplicativo exibirá a tela inicial. O conteúdo específico dessa tela será desenvolvido em um próximo incremento.

- **Configurações do Aplicativo**

A partir da tela inicial, o usuário pode acessar as configurações do aplicativo. As opções de configuração incluem:

- **Alteração de Nome do Estabelecimento**

O usuário pode alterar o nome do estabelecimento associado ao aplicativo.

- **Alteração/Remoção de CNPJ**

O usuário pode alterar ou remover o CNPJ do estabelecimento.

- **Alteração/Remoção de Logotipo**

O usuário pode alterar ou remover o logotipo do estabelecimento.

- **Importação de Atividades Comuns de Outros Ramos de Atividade**

O usuário tem a opção de importar atividades comuns de outros ramos de atividade para enriquecer as opções disponíveis no aplicativo.

Isso conclui a documentação da funcionalidade de primeira inicialização do aplicativo, conforme as especificações fornecidas. Certifique-se de testar essa funcionalidade para garantir que tudo funcione conforme o esperado.

2) Arquitetura do sistema:

O sistema desenvolvido utiliza o padrão Model View Controller (MVC) para organizar suas estruturas internas. São utilizadas três camadas:

Aplicação (View):

Camada responsável pela interface com o usuário. Desenvolvida na linguagem React.

Controle (Controller):

Camada responsável pela comunicação entre a camada de aplicação e o banco de dados. Nessa camada são implementadas as regras de negócio do sistema, bem como as validações dos dados.

Modelo (Model):

Camada que possui classes representando as tabelas existentes no banco de dados.

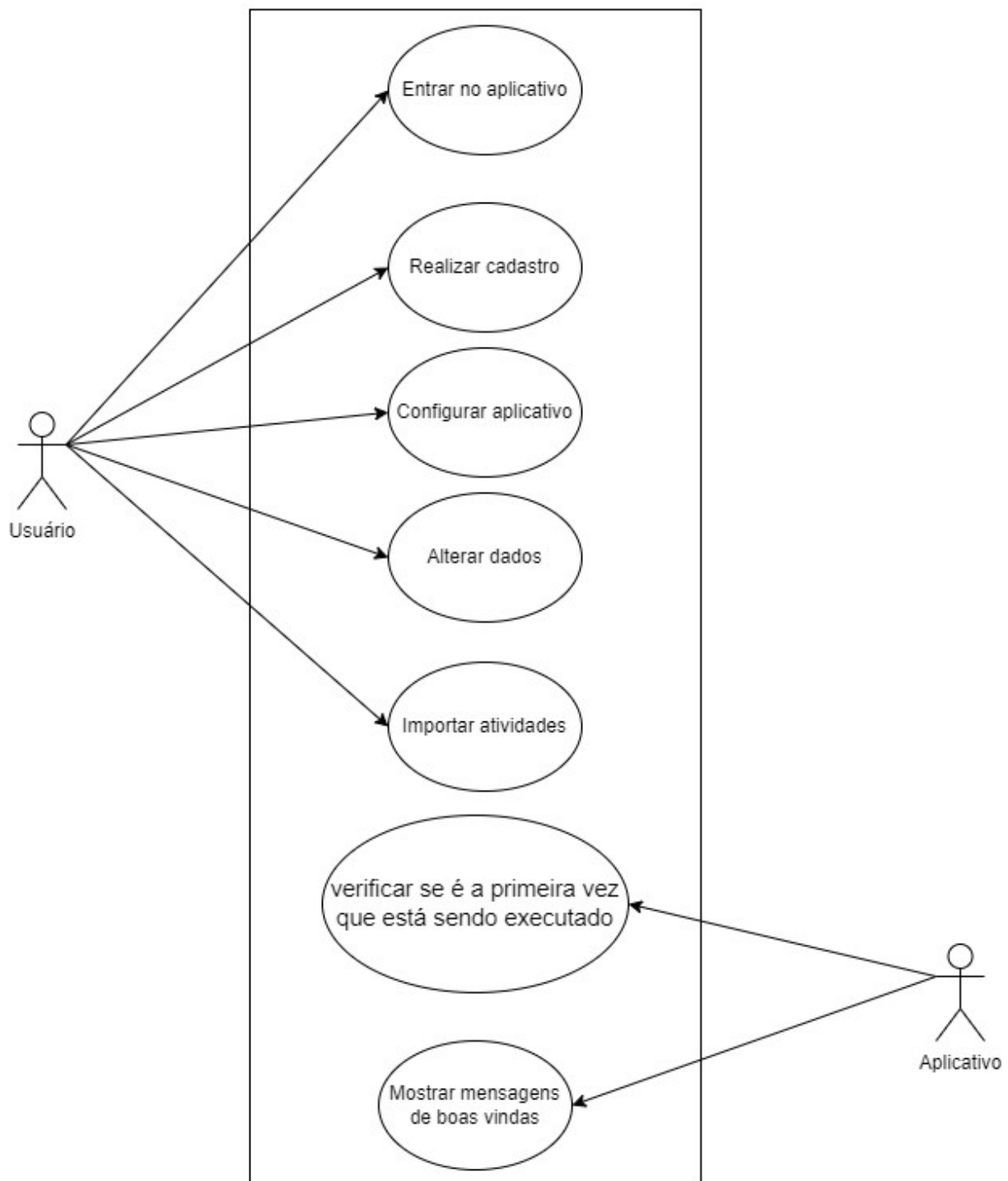
3) Tecnologias utilizadas:

React é um framework JavaScript criado pelo Facebook (atual Meta) que é usado para criar interfaces de usuário (UI) em aplicativos web. Ele é popular por ser fácil de usar, altamente flexível e escalável, e é usado por muitas empresas de tecnologia, incluindo o Facebook, Instagram e Airbnb.

O AsyncStorage é uma API nativa do React Native, utilizada para armazenar dados persistentes no dispositivo. É uma forma de salvar dados no formato chave e valor os dados salvos com a API AsyncStorage, são assíncronos, com isso retornam um Promise, e em caso de erro retorna um Error. A utilidade do AsyncStorage se torna indispensável quando precisamos armazenar dados em nosso dispositivo de forma definitiva e não temporária, um exemplo disso, quando temos uma página de login e gostaríamos de guardar o login ou id do usuário para que o mesmo seja exibido sem a necessidade do usuário informar esse login ao retornar para a app.

4) Diagrama de casos de uso:

Aplicativo para gerenciamento de atendimentos e agendamento KiAgenda



5) Requisito não funcional de usuário

RNF 01 - Utilização do sistema deve ser executado a partir do celular do cliente.

6) Requisito funcional de usuário

RF01 - Na tela de boas vindas clicar em começar e realizar o cadastro do estabelecimento;

RF02 - Realizar o cadastro informando o nome e o CNPJ (opcional) e o ramo de atividade;

RF03 - Agendar o serviço informando o nome do cliente, CPF, data, horário e o tipo de serviço a ser realizado e selecionar agendar serviço;

RF04 - Na tela de agendamentos clique em visualizar agendamentos para listar todos os agendamentos cadastrados;

7) Requisito não funcional do sistema

RNF01 – Utilização como o cliente utiliza o sistema do seu celular seja Android ou IOS, o sistema proposto foi desenvolvido utilizando a linguagem de programação React Native e foi utilizado AsyncStorage dessa forma será feito o armazenamento dos dados do sistema;

8) Requisito funcional do sistema

RFS 01 - Tela de Boas-Vindas e Cadastro do Estabelecimento:

O sistema deve exibir uma tela de boas-vindas.

Deve haver um botão "Começar" na tela de boas-vindas.

Ao clicar em "Começar", o sistema deve permitir o cadastro do estabelecimento.

O cadastro do estabelecimento deve incluir o nome obrigatório e o CNPJ (opcional).

O cadastro do estabelecimento também deve incluir o ramo de atividade do estabelecimento.

RFS 02 - Agendamento de Serviço:

O sistema deve permitir o agendamento de serviços.

Para agendar um serviço, o usuário deve fornecer:

Nome do cliente.

CPF do cliente.

Data do agendamento.

Horário do agendamento.

Tipo de serviço a ser realizado.

Deve haver um botão "Agendar Serviço" para confirmar o agendamento.

RFS 03 - Tela de Agendamentos:

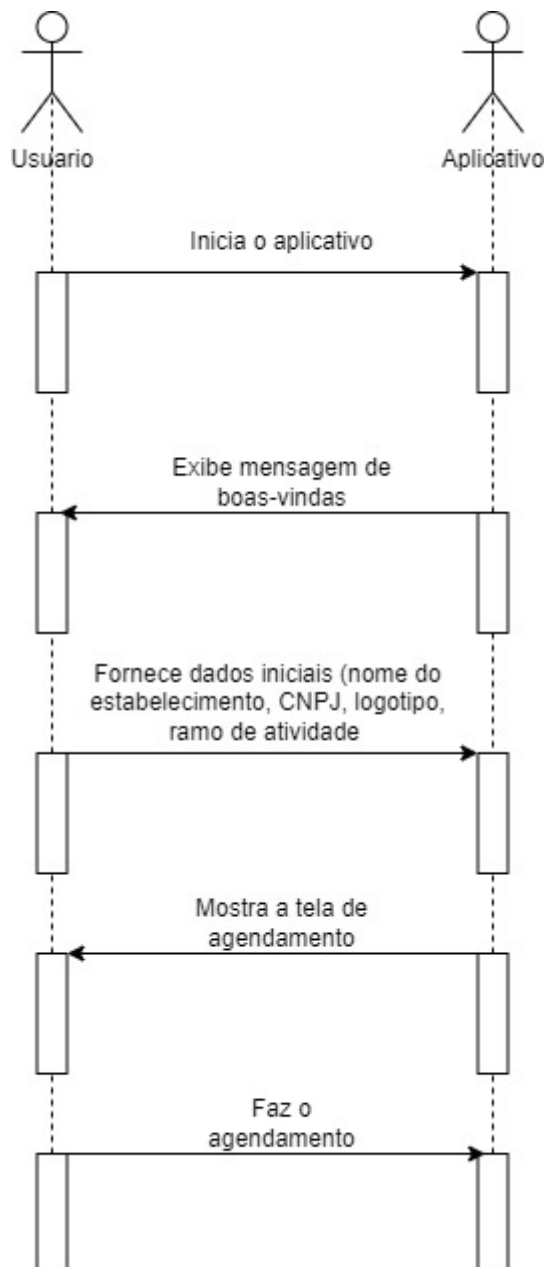
O sistema deve possuir uma tela de agendamentos.

Na tela de agendamentos, deve haver uma opção para "Visualizar Agendamentos".

Ao clicar em "Visualizar Agendamentos", o sistema deve listar todos os agendamentos cadastrados, exibindo informações como nome do cliente, CPF, data, horário e tipo de serviço agendado.

Campos (os mesmos campos da tabela de Agendamento)

9) Diagrama de sequência



10) Diagrama de classes

