



RPG0023 - Vamos criar um App

Paulo Wuéliton Horacio Fernandes (202302025919)

507 Polo Tucuruvi – São Paulo/SP

Nível 1: Vamos criar um App – 2024.2 FLEX - 4º Semestre Letivo

Meeting Fornecedores



Meeting Fornecedores

Aplicativo de cadastro de fornecedores para a empresa Meeting.

O projeto Meeting tem como objetivo desenvolver um aplicativo móvel eficiente para o cadastro de fornecedores. Focado em proporcionar uma excelente experiência ao usuário e economizar recursos, o app utiliza React Native para estabelecer uma forte presença no mercado móvel.

Dependências do Projeto

react-native: possibilita desenvolver uma mesma aplicação multiplataformas, podendo gerar um app para iOS, Android e Web.

expo: Ambiente de Desenvolvimento da aplicação.

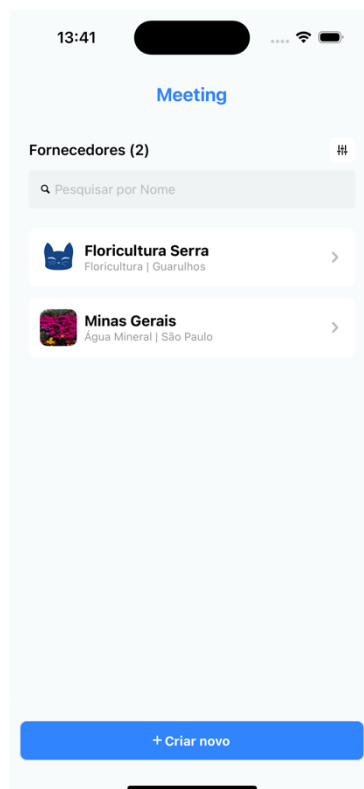
expo-router: Responsável pelas rotas da aplicação.

react-native-reanimated: Animações para deixar a experiência de uso mais fluída no app.

react-native-mask-text: Máscaras de inserção de texto, informa a formatação e formata o texto inserido pelo usuário nos padrões necessários dos dados.

Interfaces da aplicação

Tela Inicial:

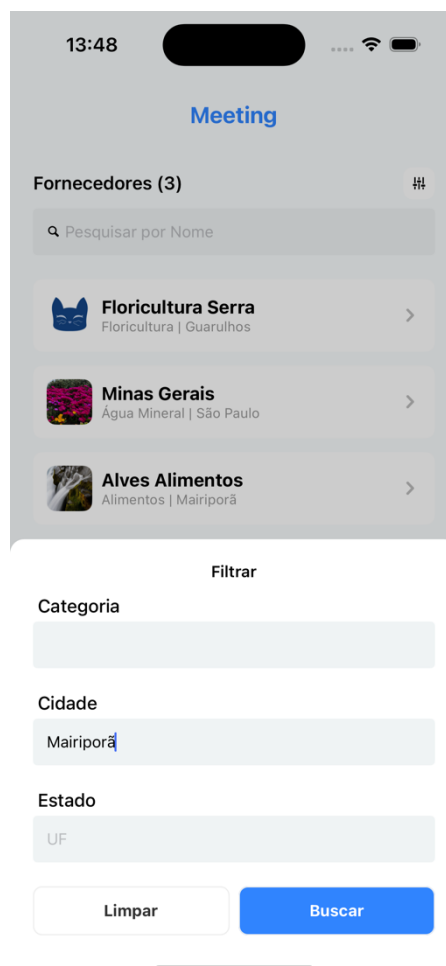


Listagem dos fornecedores cadastrados, dispostos como uma lista de contatos (agenda), utilizando uma FlatList, ordenando os itens por data de cadastro, do mais antigo para o mais recente.

Contém um campo rápido de busca que filtra a listagem abaixo, mas, permite também a busca por filtros avançados clicando no ícone de filtro ao lado do título.

Botão de cadastro rápido de um novo fornecedor.

Busca Avançada:



Ao clicar na opção de filtro, as opções de filtros avançados são exibidos, permitindo a busca por Categoria do produto do Fornecedor, Cidade do Fornecedor e Estado do Fornecedor.

Adicionar Fornecedor:

A mobile application screen titled "Adicionar Fornecedor" (Add Supplier). At the top, there is a blue square icon with a white person silhouette and a camera icon in the bottom right corner. Below the icon are several input fields with labels: "Nome do Fornecedor" (Supplier Name) with a placeholder "Digite o nome do Fornecedor"; "Categoria" (Category) with a placeholder "Digite a categoria do Produtos fornecidos"; "Telefone" (Phone) with a placeholder "(00) 00000-0000"; "CEP" (Postal Code) with a placeholder "00000-000"; "Endereço" (Address); "Número" (Number); and "Complemento" (Complement). The screen has a white background and a black top bar.

Formulário para adicionar um novo fornecedor, permite a seleção de uma imagem do dispositivo do usuário como imagem do fornecedor.

Os campos de telefone e CEP são formatados com uma máscara de inserção do conteúdo com o objetivo de melhorar a experiência do usuário ao entender o formato dos dados que devem ser inseridos.

Visualizar Fornecedor:



Exibe os detalhes de um fornecedor, como a categoria dos produtos fornecidos, telefone de contato e endereço.

É possível entrar em contato com o fornecedor ao clicar sobre o telefone.

Permite a Edição do fornecedor selecionado.

Editar Fornecedor:

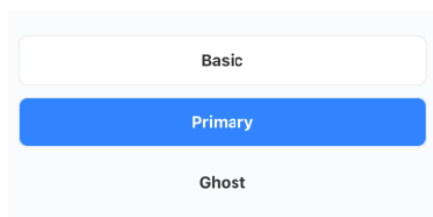
A screenshot of a mobile application interface for editing a contact. At the top, the status bar shows the time 13:41 and battery level. Below the status bar is a back arrow. The contact's profile picture is a blue cat face icon. The form contains the following fields: 'Nome do Fornecedor' with the value 'Floricultura Serra', 'Categoria' with 'Floricultura', 'Telefone' with '(11) 90000-0000', 'CEP' with '00000-000', 'Endereço' with 'Rua sem saída', 'Número' with '1', and 'Complemento' which is empty. The bottom of the screen shows a home indicator bar.

Exibe o formulário de edição do contato selecionado, permitindo editar todos os dados inseridos ou excluir o contato.

Componentes

Componentes que compõe a identidade visual do aplicativo.

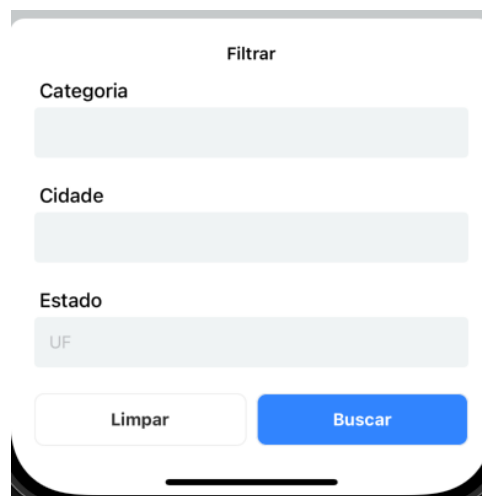
Button



```
<Button>Basic</Button>  
<Button variant="primary">Primary</Button>  
<Button variant="ghost">Ghost</Button>
```

Possui 3 variantes, sendo a Basic padrão, e podendo ser configurada pela propriedade variant do Botão.

Filter

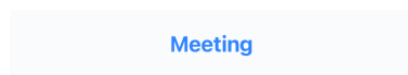


Componente que desliza de baixo para cima quando aberto, com sobreposição na tela em que é utilizado, permite visualizar um modal de filtros que podem ser aplicados a listagem.

```
<Filter />
```

Não possui propriedades e seu funcionamento é exclusivo para os filtros da listagem.

Header



Cabeçalho da aplicação que contém a logo da empresa.

```
<SafeAreaView>
  <Header></Header>
</SafeAreaView>
```

Não possui propriedades e foi desenvolvido para uso exclusivo da home

Input

Nome do Fornecedor

Digite o nome do Fornecedor

Campos de inserção de texto.

```
<Input
  label="Nome do Fornecedor"
  placeholder="Digite o nome do Fornecedor"
  value={value}
  onChange={() => {}}
/>
```

label: Texto que vai ser exibido acima do campo.

placeholder: Texto alternativo que vai ser exibido quando não possuir texto inserido no campo.

value: Valor do campo.

onChange: função que deve ser executada quando o valor do campo mudar.

MaskedInput

Telefone

(11) 90000-0000

Inserção de textos com um padrão de formato, como CEP e Telefone.


```
<MaskedInput
  label="Telefone"
  placeholder="(00) 00000-0000"
  mask="(99) 99999-9999"
  keyboardType="phone-pad"
  value=""
  onChange={() => {}}
/>
```

label: Texto que vai ser exibido acima do campo.

placeholder: Texto alternativo que vai ser exibido quando não possuir texto inserido no campo.

mask: Máscara para inserir os dados no campo, consultar documentação [react-native-mask-text](#).

keyboardType: Tipo do teclado que o dispositivo deve exibir para inserir os dados.

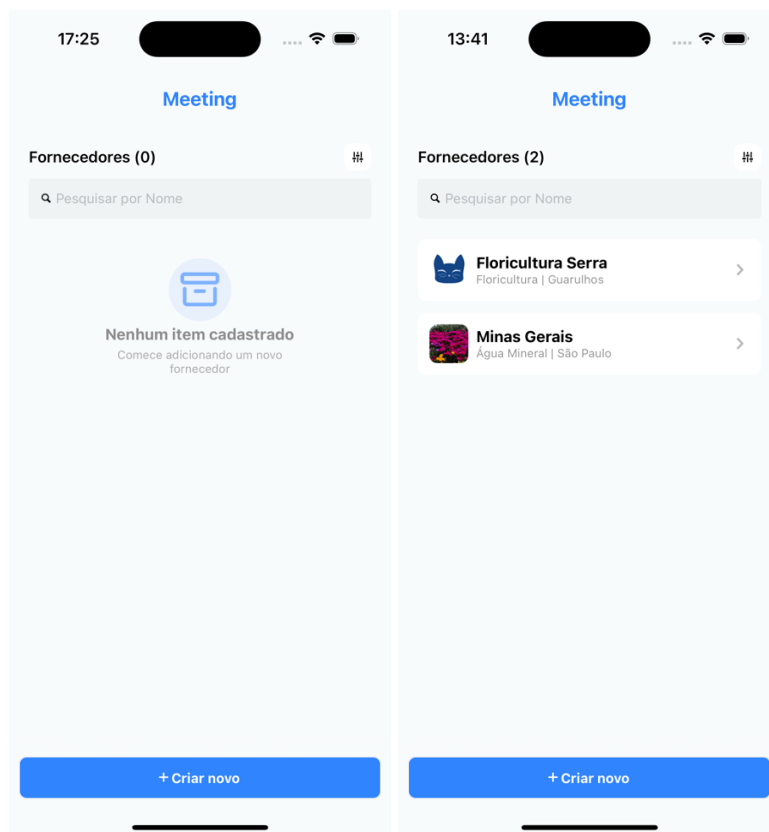
value: Valor do campo.

onChange: função que deve ser executada quando o valor do campo mudar.

SupplierForm

Formulário de cadastro/edição do Fornecedor

SupplierList



Listagem dos Fornecedores cadastrados, possui mensagem para informar quando a lista estiver vazia.

Hooks

Funções que centralizam as principais funcionalidades da aplicação, separando a lógica das telas da interface.

`useEditSupplier`

Possui todas as propriedades necessárias para tela de edição de um Fornecedor.

`useFilterForm`

Possui todas as propriedades e funções necessárias para o formulário de filtro da listagem de fornecedores da tela inicial.

`useForm`

Hook de lógica para administrar o estado dos formulários da aplicação, responsável por registrar os campos do formulário e controlar o estado de erro dos campos. É utilizado em todos os formulários da aplicação.

`useSupplierForm`

Hook responsável pela lógica do formulário de criação dos Fornecedores, possui todas as propriedades e funções necessárias para a tela de Criar Fornecedor.

Contextos

Gerencia o compartilhamento dos estados (dados) entre as telas da aplicação.

`useFilterModalContext`

Gerencia a exibição do modal de filtros avançados da aplicação.

`useSupplierContext`

Gerencia a listagem, filtraem, exibição, edição e exclusão dos dados dos fornecedores na aplicação.