#### VIVACIDADE: ESTÍMULO E CRIATIVIDADE PARA IDOSOS

PROJETO DE PROGRAMAÇÃO PARA DISPOSITIVOS MÓVEIS

### INTEGRANTES DO GRUPO

Cauê Edi de Souza e Braga –202202154768

Professor:

Paulo Ingrácio

### INFORMAÇÕES SOBRE O CLIENTE

Minha cliente é Nilza Araújo, artesã especializada na criação de vasos e laços artesanais. Ao longo do tempo, ela construiu uma sólida base de fãs, tanto em plataformas digitais como YouTube e TikTok, quanto na comunidade do condomínio onde reside.

Observando o crescimento desse público e a necessidade de se conectar de forma mais próxima e personalizada com seus seguidores, surgiu a ideia de desenvolver um **aplicativo mobile** exclusivo para compartilhar seus conteúdos e fortalecer ainda mais essa relação.

### PROPOSTA PARCERIA COM O **PUBLICO ALVO**

- Colaboração na criação de conteúdo: O público-alvo pode contribuir com suas próprias "receitas" e tutoriais, criando um espaço para a troca de experiências. Isso gera um senso de comunidade e participação, tornando o aplicativo mais dinâmico e relevante.
- Feedback e sugestões: Convidar os usuários a darem feedbacks sobre o app, sugerindo melhorias ou novas funcionalidades. Isso pode ajudá-lo a ajustar o app de acordo com as necessidades reais dos usuários.
- Eventos ou comunidades: Criar um canal de comunicação (como fóruns ou redes sociais) para formar uma comunidade onde os usuários possam interagir, compartilhar suas criações e tirar dúvidas. Também pode incluir eventos online, como workshops de como usar o app ou tutoriais ao vivo.

### OBJETIVO DO APLICATIVO

O projeto consiste em um aplicativo móvel para registrar, atualizar, visualizar e excluir passo a passo de "receitas", como a criação de vasos, laços, brindes, entre outros. A proposta é permitir que a comunidade acesse, por meio do celular, tutoriais detalhados para reproduzir diversos itens de forma precisa. Além disso, o app permitirá o registro de outros tipos de "receitas", como preparos de comidas, técnicas de costura, cuidados com roupas e outros "life-hacks". A plataforma será construída utilizando o React Native, que garante compatibilidade tanto para iOS quanto Android, Express.js para desenvolvimento da api e o banco de dados MySQL, que armazenará as informações de forma segura e eficiente. Com isso, o aplicativo visa proporcionar uma experiência prática e interativa, facilitando o aprendizado e a troca de conhecimento entre os usuários.

## DESENVOLVI MENTO DO APLICATIVO

O aplicativo foi desenvolvido utilizando React Native para o front-end, garantindo agilidade e compatibilidade com dispositivos Android e iOS.

O **back-end** foi implementado com **Express.js**, escolhido pela sua leveza e praticidade na criação de APIs.

O banco de dados foi estruturado com MySQL e está rodando em um ambiente isolado utilizando Docker, garantindo maior estabilidade e facilidade na gestão do ambiente de desenvolvimento.







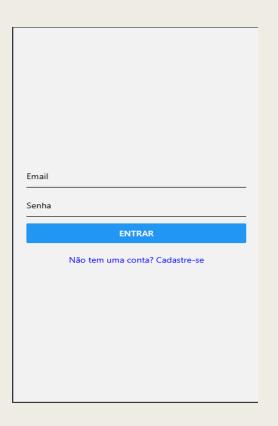
=xpress

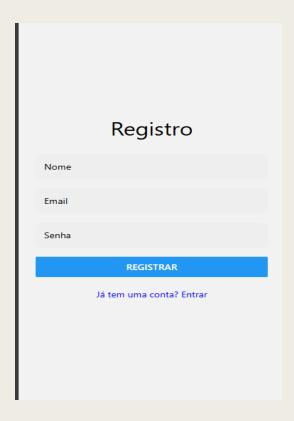
### DESENVOLVIMENTO DO APLICATIVO

### Até o momento, o aplicativo conta com as seguintes telas desenvolvidas:

- **Login:** Permite o acesso seguro dos usuários cadastrados.
- Registro: Tela para novos usuários criarem suas contas.
- Home: Tela principal de boas-vindas e navegação.
- Formulário de Posts: Área onde os usuários podem criar e enviar novos conteúdos.
- Essas funcionalidades representam a base inicial do projeto, focadas em garantir a autenticação e a interação dos usuários com a plataforma.

# TELAS DO APLICATIVO: LOGIN E CADASTRO





### TELAS DO APLICATIVO: HOME E FORMULÁRIO DE POSTS

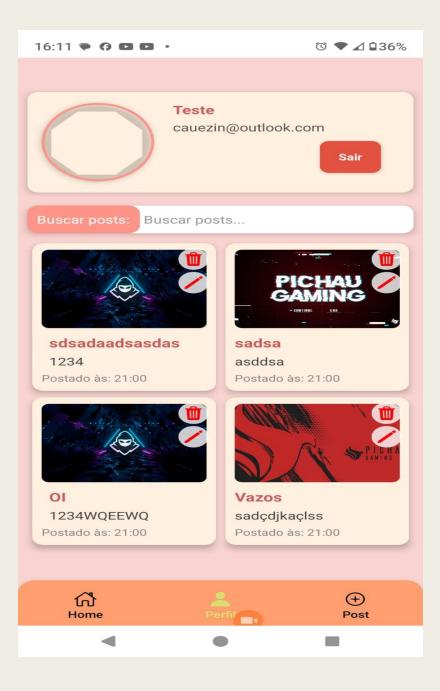




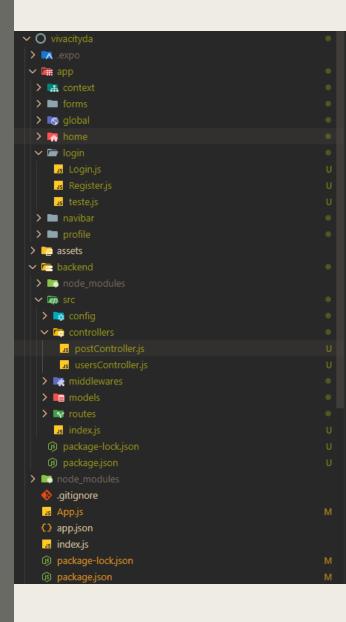
## TELAS DO APLICATIVO: DETALHES DO POST

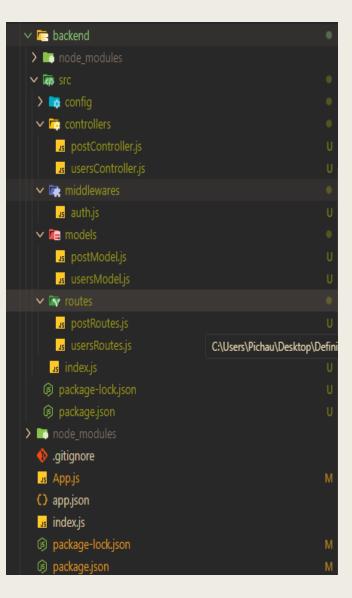


### TELAS DO APLICATIVO: PERFIL



### CÓDIGO: ESTRUTURA DE PASTAS:





### CÓDIGO

```
v import React, { useState } from 'react';
                                                                               const { createContext, useContext, useEffect, useState } = React;
                                                                               import api from '../global/services/api';
                                                                               import { jwtDecode } from "jwt-decode";
import AsyncStorage from '@react-native-async-storage/async-storage';
                                                                               export const AuthContext = createContext();
   StvleSheet.
                                                                                const [isAuthenticated, setIsAuthenticated] = useState(false);
                                                                                const [user, setUser] = useState(null);
v export default function Login({ navigation }) {
                                                                                const [loading, setLoading] = useState(true);
   const [email, setEmail] = useState('');
   const [senha, setSenha] = useState('');
                                                                                useEffect(() => {
   const { login } = useAuth();
                                                                                  const loadToken = async () => {
                                                                                    const token = await AsyncStorage.getItem('token');
const handleLogin = async () => {
                                                                                    if (token) {
    if (!email || !senha) {
                                                                                       api.defaults.headers.common['Authorization'] = `Bearer ${token}`;
       const response = await fetch('http://192.168.0.249:3000/api/users/login', {
                                                                                         console.log('Token recebido:', token);
                                                                                         const decodedUser = jwtDecode(token);
                                                                                         console.log('Usuário decodificado:', decodedUser);
           'Content-Type': 'application/json',
        body: JSON.stringify({ email, senha }),
                                                                                         setUser(decodedUser);
                                                                                         setIsAuthenticated(true);
      if (!response.ok) {
        throw new Error('Falha no login');
                                                                                         await AsyncStorage.setItem('user', JSON.stringify(decodedUser));
                                                                                       } catch (error)
                                                                                         console.error('Erro ao decodificar token:', error);
      const data = await response.json();
      const { token } = data;
       await login(token);
      console.log('Resposta do servidor:', data);
       navigation.navigate('HomeTabs');
                                                                                  loadToken();
     l catch (error)
```

#### DESAFIOS

O desenvolvimento do aplicativo tem representado um grande desafio, especialmente em dois aspectos principais:

- Design do Projeto: Criar uma interface intuitiva, agradável e acessível exigiu bastante pesquisa e atenção aos detalhes.
- Consumo de APIs: Integrar o aplicativo ao back-end e garantir uma comunicação eficiente entre as partes foi um desafio técnico importante durante o processo.

Esses obstáculos, no entanto, têm proporcionado um grande aprendizado e evolução no projeto.

### CONCLUSÃO

- Obrigado a todos pela atenção.
- Link dos canais da cliente: <a href="https://www.youtube.com/@nilzacriativ">https://www.youtube.com/@nilzacriativ</a> a

Segue foto de comprovação com a cliente.

