Estácio de Sa

Campus Niterói

Telas e Código-fonte das telas do sistema.

Sistema De Gerenciamento Da Clínica Odontológica Dente Forte

Integrantes:

DIEGO OLIVEIRA DA FONTE

Matricula: 202202764485

DANIEL LINDGREN GUIMARÃES

Matricula: 202202191451

BRENNO DE PAIVA GUIMARÃES

Matricula: 202002029961

Kaio César Lúcio Loureiro Pegado

Matricula: 201908399821

Professor e orientador:

Edberto Mariano

2023

Niterói / Rio De Janeiro

Tela Inicial

Código-Fonte

import React from "react";

import { View, Text, StyleSheet, Image, TouchableOpacity } from 'react-native';

import \* as Animatable from 'react-native-animatable'

import { useNavigation } from '@react-navigation/native'

export default function Welcome(){

    const navigation = useNavigation();

    return(

        <View style={styles.container}>

            <View style={styles.containerLogo}>

                <Animatable.Image

                animation="flipInY"

                source={require('../../assets/Logo03.png')}

                style= {{ width: '100%' }}

                resizeMode="contain"

                />

        </View>

        <Animatable.View delay={600} animation="fadeInUp" style={styles.containerForm}>

            <Text style={styles.title}>Seu APP de gerenciamento preferido</Text>

            <Text style={styles.text}>Faça o login para começar</Text>

            <TouchableOpacity style={styles.button} onPress={ () => navigation.navigate('SignIn')}>

            <Text style={styles.buttonText}>Acessar</Text>

            </TouchableOpacity>

        </Animatable.View>

        </View>

    );

}

const styles = StyleSheet.create({

    container:{

        flex:1,

        backgroundColor: '#38a69d'

    },

    containerLogo:{

        flex:2,

        backgroundColor: '#38a69d',

        justifyContent: 'center',

        alignItems:'center'

    },

    containerForm:{

        flex:1,

        backgroundColor: '#fff',

        borderTopLeftRadius: 25,

        borderTopRightRadius: 25,

        paddingStart: '5%',

        paddingEnd: '5%'

    },

    title:{

        fontSize: 24,

        fontWeight: 'bold',

        marginTop: 28,

        marginBottom: 12,

    },

    text:{

        color: '#a1a1a1'

    },

    button:{

        position: 'absolute',

        backgroundColor: '#38a69d',

        borderRadius: 50,

        paddingVertical: 8,

        width: '60%',

        alignSelf: 'center',

        bottom: '15%',

        alignItems: 'center',

        justifyContent: 'center',

    },

    buttonText:{

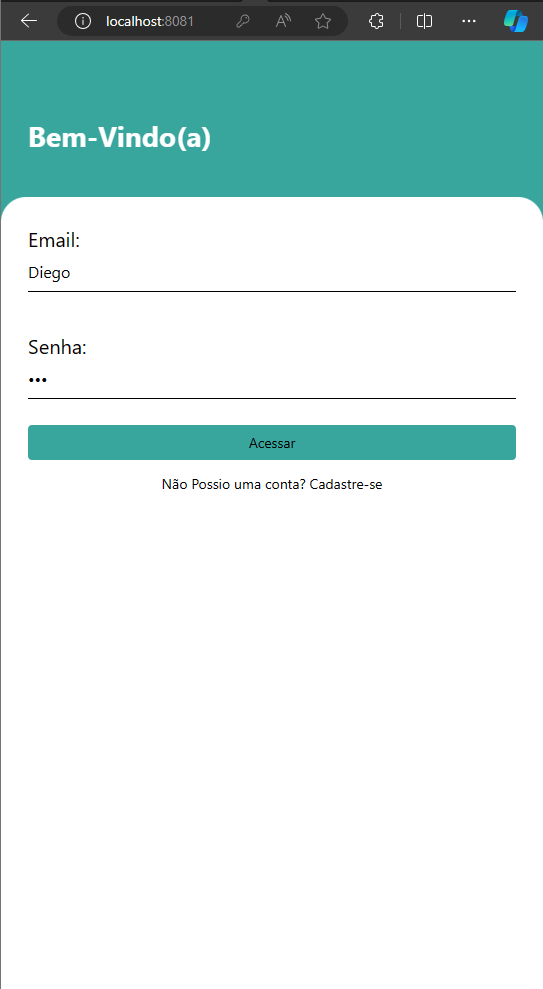
        fontSize: 18,

        color: '#fff',

        fontWeight: 'bold',

    },

})

Tela de Login

Código-Fonte

import React, { useState } from "react";

import { View, Text, StyleSheet, TextInput, TouchableOpacity } from 'react-native';

import \* as Animatable from 'react-native-animatable'

import { useNavigation } from '@react-navigation/native'

export default function Signin(){

    const [email, setEmail] = useState('');

    const [password, setPassword] = useState('');

    const navigation = useNavigation();

    const handleLogin = () => {

        if(email === 'Diego' && password === '123'){

            navigation.navigate('Menu');

        } else {

            alert('Email ou senha incorretos!');

        }

    }

    return(

        <View style={styles.container}>

            <Animatable.View animation="fadeInLeft" delay={500} style={styles.containerHeader}>

                <Text style={styles.message}>Bem-Vindo(a)</Text>

            </Animatable.View>

            <Animatable.View animation="fadeInUp" style={styles.containerForm}>

            <Text style={styles.title}>Email:</Text>

            <TextInput

            placeholder="Digite um Email..."

            style={styles.input}

            onChangeText={text => setEmail(text)}

            value={email}>

            </TextInput>

            <Text style={styles.title}>Senha:</Text>

            <TextInput

            placeholder="Digite sua senha..."

            style={styles.input}

            onChangeText={text => setPassword(text)}

            value={password}

            secureTextEntry>

            </TextInput>

            <TouchableOpacity style={styles.button} onPress={handleLogin}>

                <Text styles={styles.buttonText}>Acessar</Text>

            </TouchableOpacity>

            <TouchableOpacity style={styles.buttonRegister}>

                <Text styles={styles.registerText}>Não Possio uma conta? Cadastre-se</Text>

            </TouchableOpacity>

            </Animatable.View>

        </View>

    );

}

const styles = StyleSheet.create({

    container:{

        flex: 1,

        backgroundColor: "#38a69d",

    },

    containerHeader:{

        marginTop: '14%',

        marginBottom: '8%',

        paddingStart: '5%',

    },

    message:{

        fontSize: 28,

        fontWeight: 'bold',

        color: '#fff'

    },

    containerForm:{

        backgroundColor: '#fff',

        flex: 1,

        borderTopLeftRadius: 25,

        borderTopRightRadius: 25,

        paddingStart: '5%',

        paddingEnd: '5%',

    },

    title:{

        fontSize: 20,

        marginTop: 28,

    },

    input:{

        borderBottomWidth: 1,

        height: 40,

        marginBottom: 12,

        fontSize: 16,

    },

    button:{

        backgroundColor: '#38a69d',

        width: '100%',

        borderRadius: 4,

        paddingVertical: 8,

        marginTop: 14,

        justifyContent: 'center',

        alignItems: 'center',

    },

    buttonText:{

        color: '#fff',

        fontSize: 18,

        fontWeight: 'bold',

    },

    buttonRegister:{

        marginTop: 14,

        alignSelf: 'center',

    },

    registerText:{

        color: '#a1a1a1',

    },

})

Tela Menu

Código-Fonte

import React from "react";

import { View, Text, StyleSheet, Image, TouchableOpacity } from 'react-native';

import \* as Animatable from 'react-native-animatable'

import { useNavigation } from '@react-navigation/native'

export default function Menu(){

    const navigation = useNavigation();

    return(

        <View style={styles.container}>

            <Animatable.View animation="fadeInUp" delay={500} style={styles.containerHeader}>

                <Text style={styles.message}>Escolha Uma Opção</Text>

            </Animatable.View>

            <Animatable.View animation="fadeInUp" style={styles.containerForm}>

            <Text style={styles.buttonTextAbove}>Área Do Cliente</Text>

            <TouchableOpacity style={styles.button} onPress={ () => navigation.navigate('Cclient')}>

                <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/adicionar-usuario.png')}

                    resizeMode="stretch"

                />

                <Text styles={styles.buttonText}>Cadastro De Cliente</Text>

            </TouchableOpacity>

            <TouchableOpacity style={styles.button} onPress={ () => navigation.navigate('Gclient')}>

                <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/do-utilizador.png')}

                    resizeMode="stretch"

                />

                <Text styles={styles.buttonText}>Gerenciamento De Cliente</Text>

            </TouchableOpacity>

            <Text style={styles.buttonTextAbove}>Àrea Do Funcionário</Text>

            <TouchableOpacity style={styles.button}>

                <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/adicionar-usuario.png')}

                    resizeMode="stretch"

                />

                <Text styles={styles.buttonText}>Cadastre De Funcionário</Text>

            </TouchableOpacity>

            <TouchableOpacity style={styles.button}>

                <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/do-utilizador.png')}

                    resizeMode="stretch"

                />

                <Text styles={styles.buttonText}>Gerenciamento De Funcionário</Text>

            </TouchableOpacity>

            <TouchableOpacity style={styles.button\_deslogar} onPress={ () => navigation.navigate('Welcome')}>

                <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/deslogar.png')}

                    resizeMode="stretch"

                />

                <Text styles={styles.buttonText}>Deslogar</Text>

            </TouchableOpacity>

            </Animatable.View>

        </View>

    );

}

const styles = StyleSheet.create({

    container:{

        flex: 1,

        backgroundColor: "#38a69d",

    },

    containerHeader:{

        marginTop: '14%',

        marginBottom: '8%',

    },

    message:{

        fontSize: 28,

        fontWeight: 'bold',

        color: '#fff',

        alignSelf: 'center',

    },

    containerForm:{

        backgroundColor: '#fff',

        flex: 1,

        borderTopLeftRadius: 25,

        borderTopRightRadius: 25,

        paddingStart: '5%',

        paddingEnd: '5%',

    },

    buttonTextAbove:{

        textAlign: "center",

        fontSize: 16,

        fontWeight: 'bold',

        color:'#000',

        marginTop: 20,

    },

    iconimage:{

        width: 35,

        height: 35,

    },

    button:{

        backgroundColor: '#38a69d',

        width: '100%',

        borderRadius: 4,

        paddingVertical: 8,

        marginTop: 15,

        justifyContent: 'center',

        alignItems: 'center',

    },

    button\_deslogar:{

        backgroundColor: '#CD4242',

        width: '100%',

        borderRadius: 4,

        paddingVertical: 8,

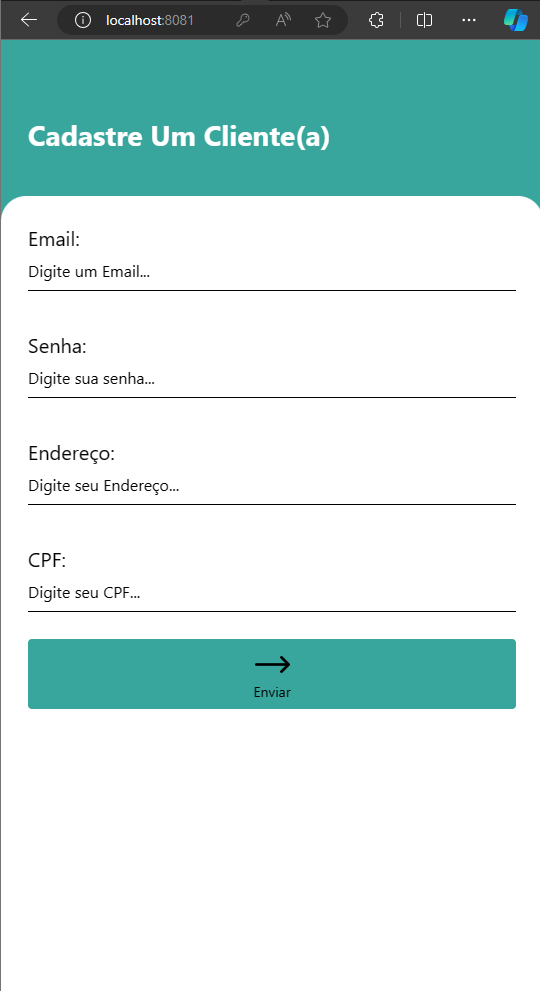
        marginTop: 54,

        justifyContent: 'center',

        alignItems: 'center',

    },

})

Tela de Cadastro de Cliente

Código-fonte

import React, { useState } from "react";

import { View, Text, TextInput, StyleSheet, Image, TouchableOpacity, Alert, Button } from 'react-native';

import axios from 'axios';

import \* as Animatable from 'react-native-animatable'

import { useNavigation } from '@react-navigation/native'

const CadastroUsuarioScreen = () => {

  const [email, setEmail] = useState('');

  const [senha, setSenha] = useState('');

  const [endereco, setEndereco] = useState('');

  const [cpf, setCpf] = useState('');

  const [emailValid, setEmailValid] = useState(false);

  const [senhaValid, setSenhaValid] = useState(false);

  const cadastrarUsuario = () => {

    if (emailValid && senhaValid) {

      axios

       .post('http://localhost/cadastro\_cliente.php', {

          email: email,

          senha: senha,

          endereco: endereco,

          cpf: cpf,

        }, {

          headers: {

            'Content-Type': 'application/json'

          }

        })

       .then(response => {

          console.log(response.data);

          // Faça algo após o cadastro bem-sucedido

        })

       .catch(error => {

          console.error(error);

        });

    } else {

      Alert.alert(

        'Atenção',

        'Por favor, preencha corretamente o email e a senha.',

        [{ text: 'OK', onPress: () => console.log('OK Pressed') }]

      );

    }

  }

  return (

    <View style={styles.container}>

      <Animatable.View animation="fadeInLeft" delay={500} style={styles.containerHeader}>

        <Text style={styles.message}>Cadastre Um Cliente(a)</Text>

      </Animatable.View>

      <Animatable.View animation="fadeInUp" style={styles.containerForm}>

        <Text style={styles.title}>Email:</Text>

        <TextInput

          placeholder="Digite um Email..."

          value={email}

          onChangeText={text => {

            setEmail(text);

            if (text.length > 0 && text.includes('@')) {

              setEmailValid(true);

            } else {

              setEmailValid(false);

            }

          }}

          style={styles.input}

        />

        <Text style={styles.title}>Senha:</Text>

        <TextInput

          placeholder="Digite sua senha..."

          value={senha}

          onChangeText={text => {

            setSenha(text);

            if (text.length > 5) {

              setSenhaValid(true);

            } else {

              setSenhaValid(false);

            }

          }}

          style={styles.input}

        />

        <Text style={styles.title}>Endereço:</Text>

        <TextInput

          placeholder="Digite seu Endereço..."

          value={endereco}

          onChangeText={text => setEndereco(text)}

          style={styles.input}

        />

        <Text style={styles.title}>CPF:</Text>

        <TextInput

          placeholder="Digite seu CPF..."

          value={cpf}

          onChangeText={text => setCpf(text)}

          style={styles.input}

        />

        <TouchableOpacity title="Cadastrar" onPress={cadastrarUsuario} style={styles.button}>

          <Image style={styles.iconimage}

            animation="flipInY"

            source={require('../../assets/seta-direita.png')}

            resizeMode="stretch"

          />

          <Text styles={styles.buttonText}>Enviar</Text>

        </TouchableOpacity>

      </Animatable.View>

    </View>

  )

}

const styles = StyleSheet.create({

  container:{

    flex: 1,

    backgroundColor: "#38a69d",

  },

  containerHeader:{

    marginTop: '14%',

    marginBottom: '8%',

    paddingStart: '5%',

  },

  message:{

    fontSize: 28,

    fontWeight: 'bold',

    color: '#fff'

  },

  containerForm:{

    backgroundColor: '#fff',

    flex: 1,

    borderTopLeftRadius: 25,

    borderTopRightRadius: 25,

    paddingStart: '5%',

    paddingEnd: '5%',

  },

  title:{

    fontSize: 20,

    marginTop: 28,

  },

  input:{

    borderBottomWidth: 1,

    height: 40,

    marginBottom: 12,

    fontSize: 16,

  },

  button:{

    backgroundColor: '#38a69d',

    width: '100%',

    borderRadius: 4,

    paddingVertical: 8,

    marginTop: 15,

    justifyContent: 'center',

    alignItems: 'center',

  },

  iconimage:{

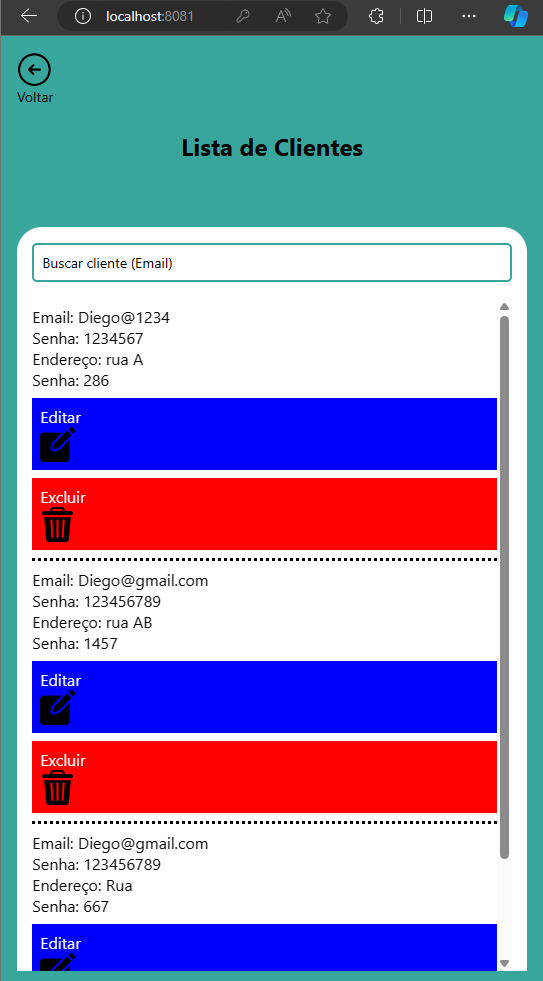
    width: 35,

    height: 35,

  },

})

export default CadastroUsuarioScreen;

Tela de Gerenciamento de Cliente

Código-Fonte

import React, { useState, useEffect } from "react";

import { View, Text, StyleSheet, FlatList, TouchableOpacity, TextInput, Image } from 'react-native';

import \* as Animatable from 'react-native-animatable'

import axios from 'axios';

import { useNavigation } from '@react-navigation/native';

const ListaClientesScreen = () => {

  const [clientes, setClientes] = useState([]);

  const [searchTerm, setSearchTerm] = useState('');

  const navigation = useNavigation();

  useEffect(() => {

    const fetchData = async () => {

      try {

        const response = await axios.get('http://localhost/listar\_cliente.php');

        setClientes(response.data);

      } catch (error) {

        console.error(error);

      }

    };

    fetchData();

  }, []);

  const handleEdit = (idcliente) => {

    navigation.navigate('Aclient', { idcliente });

  };

  const deleteClient = async (id) => {

    try {

      await axios.delete(`http://localhost/deletar\_cliente.php?idcliente=${id}`);

      setClientes(clientes.filter(cliente => cliente.idcliente!== id));

    } catch (error) {

      console.error(error);

    }

  };

  const handleSearch = (text) => {

    setSearchTerm(text);

  };

  const filteredClientes = clientes.filter(cliente => {

    const nome = cliente.email.toLowerCase();

    const searchTermLower = searchTerm.toLowerCase();

    return nome.includes(searchTermLower);

  });

  return (

    <View style={styles.container}>

      <TouchableOpacity style={styles.button} onPress={ () => navigation.navigate('Menu')}>

                <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/seta-esquerda.png')}

                    resizeMode="stretch"

                />

                <Text styles={styles.buttonText}>Voltar</Text>

      </TouchableOpacity>

      <Text style={styles.title}>Lista de Clientes</Text>

      <Animatable.View animation="fadeInUp" style={styles.containerForm}>

      <TextInput

        style={styles.searchInput}

        placeholder="Buscar cliente (Email)"

        value={searchTerm}

        onChangeText={handleSearch}

      />

      <FlatList

        data={filteredClientes}

        keyExtractor={item => item.idcliente.toString()}

        renderItem={({ item }) => (

          <View style={styles.cliente}>

            <Text style={styles.email}>Email: {item.email}</Text>

            <Text style={styles.email}>Senha: {item.senha}</Text>

            <Text style={styles.email}>Endereço: {item.endereco}</Text>

            <Text style={styles.email}>Senha: {item.CPF}</Text>

            <TouchableOpacity style={styles.editButton} onPress={() => handleEdit(item.idcliente)}>

              <Text style={styles.editButtonText}>Editar</Text>

              <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/editing.png')}

                    resizeMode="stretch"

                />

            </TouchableOpacity>

            <TouchableOpacity style={styles.deleteButton} onPress={() => deleteClient(item.idcliente)}>

              <Text style={styles.deleteButtonText}>Excluir</Text>

              <Image style={styles.iconimage}

                    animation="flipInY"

                    source={require('../../assets/bin.png')}

                    resizeMode="stretch"

                />

            </TouchableOpacity>

          </View>

        )}

      />

      </Animatable.View>

    </View>

  );

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    backgroundColor: "#38a69d",

    padding: 16,

  },

  containerForm:{

    backgroundColor: '#fff',

    flex: 1,

    marginTop: 40,

    borderTopLeftRadius: 25,

    borderTopRightRadius: 25,

    paddingStart: '3%',

    paddingEnd: '3%',

},

  button:{

    marginBottom: 24,

    marginRight: "92%",

  },

  iconimage:{

    width: 35,

    height: 35,

  },

  title: {

    fontSize: 24,

    fontWeight: 'bold',

    marginBottom: 25,

    alignSelf: "center",

  },

  cliente: {

    borderBottomWidth: 3,

    borderBottomColor: 'black',

    borderStyle: "dotted",

    paddingVertical: 8,

  },

  email: {

    fontSize: 16,

  },

  editButton: {

    backgroundColor: 'blue',

    padding: 8,

    marginTop: 8,

  },

  editButtonText: {

    color: '#fff',

    fontSize: 16,

  },

  deleteButton: {

    backgroundColor: 'red',

    padding: 8,

    marginTop: 8,

  },

  deleteButtonText: {

    color: '#fff',

    fontSize: 16,

  },

  searchInput: {

    height: 40,

    borderColor: '#38a69d',

    borderWidth: 2,

    borderRadius: 5,

    padding: 8,

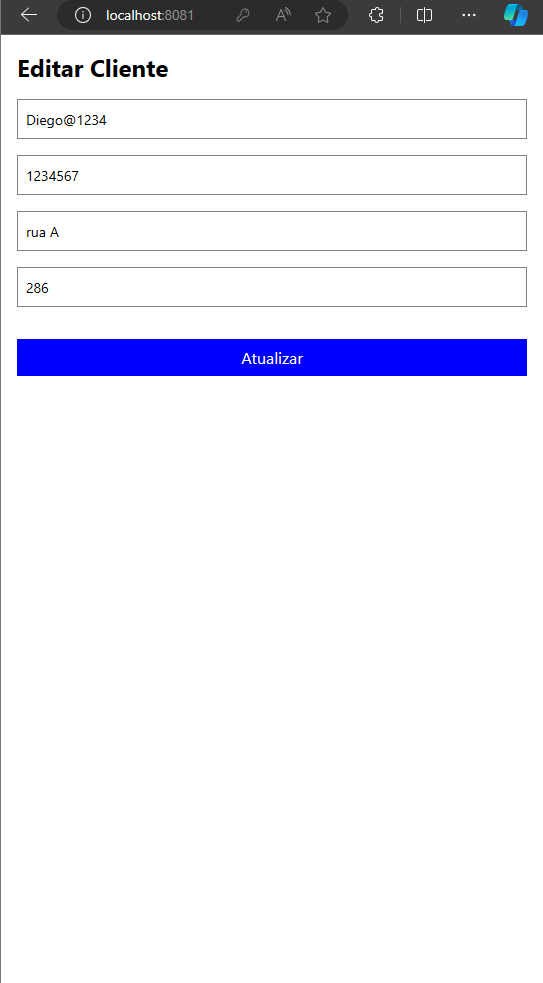
    marginBottom: 16,

    marginTop: 16,

  },

});

export default ListaClientesScreen;

Tela de Atualizar Cliente

Código-Fonte

import React, { useState, useEffect } from "react";

import { View, Text, StyleSheet, TextInput, TouchableOpacity, ScrollView } from 'react-native';

import axios from 'axios';

import { useRoute } from '@react-navigation/native';

const EditarClienteScreen = () => {

  const [email, setEmail] = useState('');

  const [senha, setSenha] = useState('');

  const [endereco, setEndereco] = useState('');

  const [cpf, setCpf] = useState('');

  const route = useRoute();

  useEffect(() => {

    const fetchData = async () => {

      try {

        const response = await axios.get(`http://localhost/editar\_cliente.php?idcliente=${route.params.idcliente}`);

        setEmail(response.data.email);

        setSenha(response.data.senha);

        setEndereco(response.data.endereco);

        setCpf(response.data.CPF);

      } catch (error) {

        console.error(error);

      }

    };

    fetchData();

  }, []);

  const updateClient = async () => {

    try {

      await axios.put(`http://localhost/atualizar\_cliente.php?idcliente=${route.params.idcliente}`, {

        email,

        senha,

        endereco,

        cpf,

      });

      // Redirecione para a tela de lista de clientes após atualizar

    } catch (error) {

      console.error(error);

    }

  };

  return (

    <ScrollView style={styles.container}>

      <Text style={styles.title}>Editar Cliente</Text>

      <TextInput

        style={styles.input}

        value={email}

        onChangeText={text => setEmail(text)}

        placeholder="Email"

      />

      <TextInput

        style={styles.input}

        value={senha}

        onChangeText={text => setSenha(text)}

        placeholder="Senha"

      />

      <TextInput

        style={styles.input}

        value={endereco}

        onChangeText={text => setEndereco(text)}

        placeholder="Endereço"

      />

      <TextInput

        style={styles.input}

        value={cpf}

        onChangeText={text => setCpf(text)}

        placeholder="CPF"

      />

      <TouchableOpacity style={styles.updateButton} onPress={updateClient}>

        <Text style={styles.updateButtonText}>Atualizar</Text>

      </TouchableOpacity>

    </ScrollView>

  );

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    backgroundColor: "#fff",

    padding: 16,

  },

  title: {

    fontSize: 24,

    fontWeight: 'bold',

    marginBottom: 16,

  },

  input: {

    height: 40,

    borderColor: 'gray',

    borderWidth: 1,

    marginBottom: 16,

    paddingHorizontal: 8,

  },

  updateButton: {

    backgroundColor: 'blue',

    padding: 8,

    marginTop: 16,

    alignItems: 'center',

  },

  updateButtonText: {

    color: '#fff',

    fontSize: 16,

  },

});

export default EditarClienteScreen;

Codigos PHP  
  
atualizar\_cliente.php

<?php

header("Access-Control-Allow-Origin: \*");

header("Content-Type: application/json; charset=UTF-8");

header("Access-Control-Allow-Methods: PUT");

header("Access-Control-Max-Age: 3600");

header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");

header("Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS");

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "react\_app";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

    die("Connection failed: ". $conn->connect\_error);

}

$idcliente = isset($\_GET['idcliente']) ? $\_GET['idcliente'] : die();

$data = json\_decode(file\_get\_contents('php://input'), true);

$email = $data['email'] ?? '';

$senha = $data['senha'] ?? '';

$endereco = $data['endereco'] ?? '';

$cpf = $data['cpf'] ?? '';

$sql = "UPDATE cliente SET email = ?, senha = ?, endereco = ?, CPF = ? WHERE idcliente = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("ssssi", $email, $senha, $endereco, $cpf, $idcliente);

$stmt->execute();

$conn->close();

echo json\_encode(array('message' => 'Cliente atualizado com sucesso'));

cadastro\_cliente.php

<?php

// Permite solicitações de qualquer origem

/\* header("Access-Control-Allow-Origin: \*");

header("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept");

$conexao = mysqli\_connect("127.0.0.1", "root", "", "react\_app");

$data = json\_decode(file\_get\_contents('php://input'));

$email = $data->email;

$senha = $data->senha;

$query = "INSERT INTO cliente (email, senha) VALUES ('$email', '$senha')";

if(mysqli\_query($conexao, $query)) {

    echo "Usuario cadastrado com sucesso";

} else {

    echo "Erro ao cadastrar usuario: " . mysqli\_error($conexao);

}

mysqli\_close($conexao);

\*/

header("Access-Control-Allow-Origin: \*");

header("Content-Type: application/json; charset=UTF-8");

header("Access-Control-Allow-Methods: POST");

header("Access-Control-Max-Age: 3600");

header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");

    $servername = "localhost";

    $username = "root";

    $password = "";

    $dbname = "react\_app";

    // Create connection

    $conn = new mysqli($servername, $username, $password, $dbname);

    // Check connection

    if ($conn->connect\_error) {

        die("Connection failed: ". $conn->connect\_error);

    }

    $data = json\_decode(file\_get\_contents('php://input'));

    $email = $data->email;

    $senha = $data->senha;

    $endereco = $data->endereco;

    $cpf = $data->cpf;

    echo "Email: $email\n";

    echo "Senha: $senha\n";

    $sql = "INSERT INTO cliente (email, senha, endereco, CPF) VALUES ('$email', '$senha', '$endereco', '$cpf')";

    echo $sql;

    if ($conn->query($sql) === TRUE) {

        echo "New record created successfully";

    } else {

        echo "Error: ". $sql. "<br>". $conn->error;

    }

    $conn->close();

?>

Deletar\_cliente.php

<?php

header("Access-Control-Allow-Origin: \*");

header("Content-Type: application/json; charset=UTF-8");

header("Access-Control-Allow-Methods: DELETE");

header("Access-Control-Max-Age: 3600");

header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "react\_app";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

    die("Connection failed: ". $conn->connect\_error);

}

$id = $\_GET['idcliente'];

$sql = "DELETE FROM cliente WHERE idcliente=$id";

if ($conn->query($sql) === TRUE) {

    echo "Cliente excluído com sucesso";

} else {

    echo "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

?>

Editar\_cliente.php

<?php

header("Access-Control-Allow-Origin: \*");

header("Content-Type: application/json; charset=UTF-8");

header("Access-Control-Allow-Methods: POST");

header("Access-Control-Max-Age: 3600");

header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");

header("Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS");

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "react\_app";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

    die("Connection failed: ". $conn->connect\_error);

}

$idcliente = isset($\_GET['idcliente']) ? $\_GET['idcliente'] : die();

$sql = "SELECT idcliente, email, senha, endereco, CPF FROM cliente WHERE idcliente = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("i", $idcliente);

$stmt->execute();

$result = $stmt->get\_result();

$cliente = $result->fetch\_assoc();

$conn->close();

echo json\_encode($cliente);

listar\_cliente.php

<?php

header("Access-Control-Allow-Origin: \*");

header("Content-Type: application/json; charset=UTF-8");

header("Access-Control-Allow-Methods: GET");

header("Access-Control-Max-Age: 3600");

header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "react\_app";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

    die("Connection failed: ". $conn->connect\_error);

}

$sql = "SELECT idcliente, email, senha, endereco, CPF FROM cliente";

$result = $conn->query($sql);

$clientes = [];

if ($result->num\_rows > 0) {

    while($row = $result->fetch\_assoc()) {

        $clientes[] = $row;

    }

} else {

    echo "0 results";

}

echo json\_encode($clientes);

$conn->close();

?>