***APP de Figuras Geométricas***

1. **Objetivos:**

* Praticar criação de app juntamente com revisão de conteúdo de vestibular

**2. Projeto**

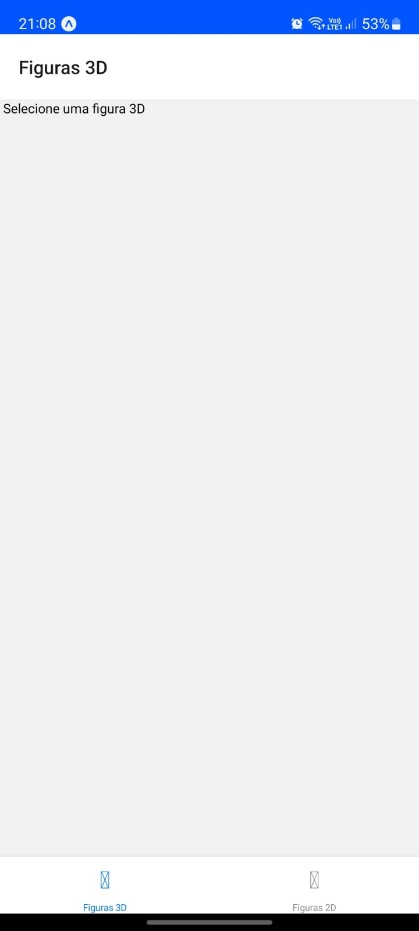
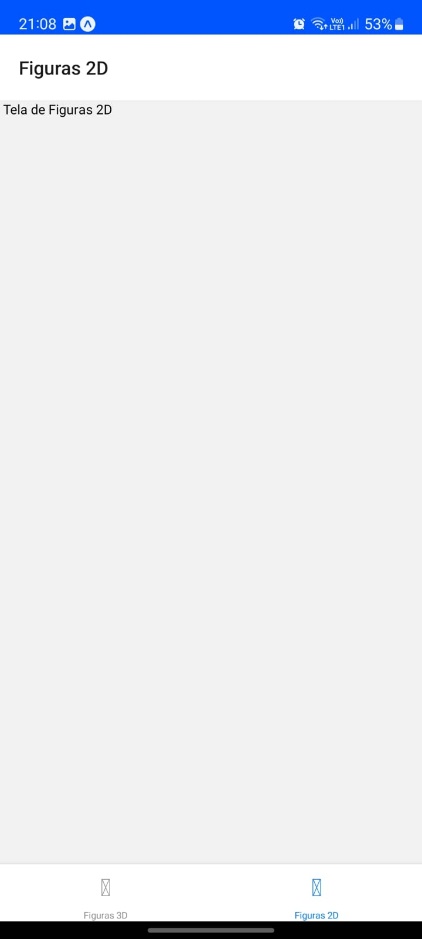
* Para estudarmos esses conceitos criaremos um novo app com o nome SesiGeometricos
  + **npx create-expo-app SesiGeometria -t blank**
* Após finalizar a criação do app verifique se o terminal está apontando para a devida pasta, se não estiver acessar a pasta com o comando:
  + **cd SesiGeometria**

**3. Instalar as bibliotecas**

**npx expo install react-native-web react-dom @expo/metro-runtime  
npx expo install @react-navigation/native  
npx expo install @react-navigation/native-stack  
npx expo install @react-navigation/bottom-tabs  
npx expo install react-native-screens react-native-safe-area-context  
npx expo install react-native-animatable  
npm install react-native-pager-view react-native-tab-view**

**4. Layout inicial do Projeto Base**

* Abaixo o layout de como ficará a estrutura inicial de nosso app

** **

**5. Criando estrutura de pastas e navegação**

* Crie uma pasta chamada **src**
* Dentro da pasta **src**, criar pasta **pages**
* Dentro de **pages**, criar o arquivo **Figuras3D.js**, com o código:

import React from "react";

import { View, Text } from "react-native";

const Figuras3D = ({navigation}) => {

    return (

        <View>

            <Text> Tela de Figuras 3D</Text>

        </View>

    )

}

export default Figuras3D

* Dentro de **pages**, criar o arquivo **Figuras2D.js**, com o código:

import React from "react";

import { View, Text } from "react-native";

const Figuras2D = ({navigation}) => {

    return (

        <View>

            <Text> Tela de Figuras 2D</Text>

        </View>

    )

}

export default Figuras2D

* Dentro de **pages**, criar o arquivo **Principal.js**, com o código:

import React from 'react';

import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';

import Figuras2D from './Figuras2D';

import Figuras3D from './Figuras3D';

const Tab = createBottomTabNavigator();

const Principal = () => {

  return (

    <Tab.Navigator>

      <Tab.Screen name="Figuras 3D" component={Figuras3D} />

      <Tab.Screen name="Figuras 2D" component={Figuras2D} />

    </Tab.Navigator>

  );

};

export default Principal;

* No arquivo **App.js:**

import React from 'react';

import { NavigationContainer } from "@react-navigation/native"

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

import { StatusBar  } from 'react-native';

import Principal from './src/pages/Principal';

const Stack = createNativeStackNavigator();

const App = () => {

  return (

    <NavigationContainer>

      <StatusBar backgroundColor='#0055ff' barStyle='light-content' />

      <Stack.Navigator>

        <Stack.Screen name="Principal" component={Principal} options={{ headerShown: false}} />

      </Stack.Navigator>

    </NavigationContainer>

  );

};

export default App;

**6. Criando arquivo de padronização**

* Dentro da pasta **src**, criar a pasta **styles**, e dentro dela o arquivo **estilos.js**, neste arquivo definiremos algumas estilizações e cores padrões que utilizaremos no app

import { StyleSheet } from "react-native";

const corPrincipal = "#0055ff";

const corSecundaria = "#29292E";

const corBorda = "#CCC";

const corTexto = "#222";

const corTitulo = "#a1a1a1";

const corBranco = "#fff";

const estilos = StyleSheet.create({

  linhaFiguras: {

    flexDirection: "row",

    flexWrap: "wrap",

    justifyContent: "space-around",

    paddingHorizontal: 10,

  },

  figuraGeometrica: {

    marginVertical: 10,

    alignItems: "center",

    backgroundColor: corBranco,

    padding: 10,

    borderRadius: 8,

    // Sombra para iOS

    shadowColor: "#000",

    shadowOffset: {

      width: 0,

      height: 2,

    },

    shadowOpacity: 0.25,

    shadowRadius: 3.84,

    // Sombra para Android

    elevation: 5,

  },

  imgFigura: {

    width: 120,

    height: 120,

    marginTop: 10,

  },

  botao: {

    backgroundColor: corPrincipal,

    borderRadius: 4,

    paddingVertical: 8,

    width: "100%",

    alignItems: "center",

    justifyContent: "center",

  },

  textoBotao: {

    fontSize: 18,

    color: corBranco,

    fontWeight: "bold",

  },

  texto: {

    textAlign: "center",

    fontSize: 20,

    fontWeight: "bold",

    color: corPrincipal,

    marginVertical: 5,

  },

  tituloFigura: {

    fontSize: 18,

    fontWeight: "bold",

    color: corPrincipal,

  },

  input: {

    margin: 3,

    borderWidth: 1,

    borderColor: "#ccc",

    borderRadius: 5,

    padding: 10,

    backgroundColor: "#fff",

  },

  linha: {

    width: '100%',

    height: 1,

    backgroundColor: corPrincipal,

    marginVertical: 10,

  },

});

export {

  corPrincipal,

  corSecundaria,

  corBorda,

  corTexto,

  corTitulo,

  corBranco,

  estilos,

};

**7. Estilizando tela principal**

* Iremos começar a estilizar nossas telas para dar vida a elas, no arquivo **Principal.js**, alterar para:

import React from "react";

import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";

import { MaterialCommunityIcons } from "@expo/vector-icons";

import Figuras2D from "./Figuras2D";

import Figuras3D from "./Figuras3D";

import { corBranco, corPrincipal } from "../styles/estilos";

const Tab = createBottomTabNavigator();

const Principal = () => {

  return (

    <Tab.Navigator

      screenOptions={{

        tabBarLabelStyle: { fontSize: 14 },

      }}

    >

      <Tab.Screen

        name="Figuras 3D"

        component={Figuras3D}

        options={{

          headerStyle: {

            backgroundColor: corPrincipal,

          },

          headerTintColor: corBranco,

          headerTitleStyle: {

            fontWeight: "bold",

          },

          tabBarIcon: ({ color, size }) => (

            <MaterialCommunityIcons name="pyramid" size={size} color={color} />

          ),

        }}

      />

      <Tab.Screen

        name="Figuras 2D"

        component={Figuras2D}

        options={{

          tabBarIcon: ({ color, size }) => (

            <MaterialCommunityIcons

              name="triangle-outline"

              size={size}

              color={color}

            />

          ),

        }}

      />

    </Tab.Navigator>

  );

};

export default Principal;

**8. Criando as figuras 3D**

* Criar a pasta **assets**, dentro da pasta **src** e colocar nela as imagens das figuras disponíveis no Classroom
* No arquivo **Figuras3D.js**, alterar para o conteúdo:

import React from "react";

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

import { estilos } from "../styles/estilos";

const Figuras3D = ({navigation}) => {

    return (

        <ScrollView style={{flex:1}}>

            <Text style={estilos.texto}> Selecione uma figura 3D</Text>

            <View style={estilos.linhaFiguras}>

                <TouchableOpacity style={estilos.figuraGeometrica}

onPress={() => navigation.navigate('Cubo')}>

                    <Text style={estilos.tituloFigura}>Cubo</Text>

                    <Image source={require('../assets/cubo.png') } style={estilos.imgFigura} />

                </TouchableOpacity>

            </View>

        </ScrollView>

    )

}

export default Figuras3D

**9. Criando a tela para cálculo da Figura do Cubo**

* Dentro da pasta **pages**, criar o arquivo Cubo.js com o conteúdo:

import React, { useState } from "react";

import { View, Text, TextInput, Image, TouchableOpacity, ScrollView } from "react-native";

import { estilos } from "../styles/estilos";

const Cubo = () => {

  const [valorA, setValorA] = useState("");

  const [volume, setVolume] = useState("");

  const [area, setArea] = useState("");

  const calcular = () => {

    let resVolume = valorA \*\* 3;

    setVolume(` V = ${resVolume}`);

    let resArea = 6 \* (valorA \*\* 2);

    setArea(` S = ${resArea}`);

  };

  return (

    <ScrollView style={{ flex: 1, padding: 15 }}>

      <View style={estilos.figuraGeometrica}>

        <Image

          source={require("../assets/cubo.png")}

          style={estilos.imgFigura}

        />

        <View style={estilos.linha} />

        <View style={{ flexDirection: "row" }}>

          <View style={{ alignItems: "center" }}>

            <Text style={estilos.texto}>a</Text>

            <TextInput

              placeholder="Valor de a"

              style={estilos.input}

              keyboardType="numeric"

              value={valorA}

              onChangeText={setValorA}

            />

          </View>

        </View>

      </View>

      <TouchableOpacity style={estilos.botao} onPress={calcular}>

        <Text style={estilos.textoBotao}>Calcular</Text>

      </TouchableOpacity>

      <View style={estilos.figuraGeometrica}>

        <Text style={estilos.texto}>Área:</Text>

        <Text> S = 6 x a² </Text>

        { area != '' && <Text> {area} </Text>}

      </View>

      <View style={estilos.figuraGeometrica}>

        <Text style={estilos.texto}>Volume:</Text>

        <Text> V = a³ </Text>

        { volume != '' && <Text> {volume} </Text>}

      </View>

    </ScrollView>

  );

};

export default Cubo;

**10. Declarando o componente para chamar a tela**

* Na tela **App.js** alterar para:

import React from "react";

import { NavigationContainer } from "@react-navigation/native";

import { createNativeStackNavigator } from "@react-navigation/native-stack";

import { StatusBar } from "react-native";

import Principal from './src/pages/Principal'

import { corPrincipal } from './src/styles/estilos'

// Declarar tela das figuras geométricas

import Cubo from "./src/pages/Cubo"

const Stack = createNativeStackNavigator()

const App = () => {

  return(

    <NavigationContainer>

      <StatusBar backgroundColor={corPrincipal} barStyle='light-content' />

      <Stack.Navigator>

        <Stack.Screen name="Principal" component={Principal}  options={{headerShown: false}} />

        {/\* Criando as telas das figuras \*/}

        <Stack.Screen name="Cubo" component={Cubo} />

      </Stack.Navigator>

    </NavigationContainer>

  )

}

export default App

* Na tela **Figuras3D.js** alterar para:

 <View style={estilos.linhaFiguras}>

                {/\* Card da figura Cubo \*/}

                <TouchableOpacity style={estilos.figuraGeometrica}

                    onPress={() => navigation.navigate('Cubo')} >

                    <Text style={estilos.tituloFigura}>Cubo</Text>

                    <Image source={require('../assets/cubo.png')}

                        style={estilos.imgFigura} />

                </TouchableOpacity>

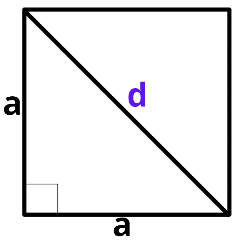
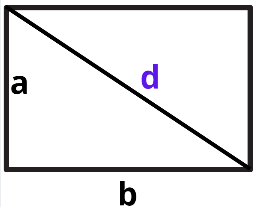
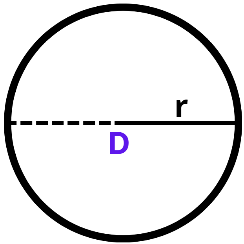
            </View>

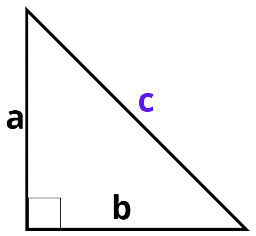
**11. Exercício**

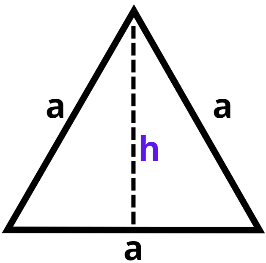
* Criar outras figuras geométricas ajustando a fórmula para calcular a área e o volume dos seguintes sólidos:
* Paralelepipedo
* Esfera
* Cone
* Cilindro

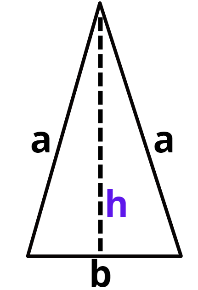
**12. Criando as figuras 2D**

* Agora iremos criar as figuras em 2D, para isso precisamos alterar o arquivo **Figuras2D.js**, adaptando para receber as imagens como a Figuras3D.js
* Criar as seguintes figuras utilizando as fórmulas abaixo:

1. **Quadrado:**
   * Área: S = a²
   * Perímetro: P = 4 \* a
   * Diagonais:
2. **Retângulo:**
   * Área: S = a \* b
   * Perímetro: P = 2 \* (a + b)
   * Diagonais:
3. **Círculo:**
   * Área:
   * Perímetro:
   * Diâmetro:

1. **Triângulo retângulo:**
   * Área:
   * Perímetro: P = a + b + c
   * Hipotenusa:



1. **Triângulo equilátero:**
   * Área:
   * Perímetro:
   * Altura:
2. **Triângulo isósceles:**
   * Área:
   * Perímetro:
   * Altura: