Código do App Flutter - Análise da Qualidade da Água

Este documento contém os dois principais arquivos para o app Flutter:  
1. pubspec.yaml  
2. lib/main.dart  
  
Você pode criar um projeto Flutter e substituir os arquivos correspondentes por estes para executar o app.

# pubspec.yaml

name: analise\_agua\_flutter  
description: App Flutter para análise da qualidade da água  
publish\_to: 'none'  
  
version: 1.0.0+1  
  
environment:  
 sdk: '>=2.19.0 <3.0.0'  
  
dependencies:  
 flutter:  
 sdk: flutter  
  
 cupertino\_icons: ^1.0.2  
  
dev\_dependencies:  
 flutter\_test:  
 sdk: flutter  
  
flutter:  
 uses-material-design: true

# lib/main.dart

import 'package:flutter/material.dart';  
  
void main() {  
 runApp(const QualidadeAguaApp());  
}  
  
class QualidadeAguaApp extends StatelessWidget {  
 const QualidadeAguaApp({super.key});  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 title: 'Análise da Qualidade da Água',  
 theme: ThemeData(primarySwatch: Colors.teal),  
 home: const AnalisePage(),  
 );  
 }  
}  
  
class AnalisePage extends StatefulWidget {  
 const AnalisePage({super.key});  
  
 @override  
 State<AnalisePage> createState() => \_AnalisePageState();  
}  
  
class \_AnalisePageState extends State<AnalisePage> {  
 final \_formKey = GlobalKey<FormState>();  
 final TextEditingController \_phController = TextEditingController();  
 final TextEditingController \_temperaturaController = TextEditingController();  
 final TextEditingController \_oxigenioController = TextEditingController();  
 final TextEditingController \_amoniaController = TextEditingController();  
 String \_especieSelecionada = 'Tilápia';  
 List<String> \_resultado = [];  
  
 final Map<String, dynamic> parametrosEspecie = {  
 'Tilápia': {  
 'ph': [6.5, 8.5],  
 'temperatura': [26, 30],  
 'oxigenio': 5.0,  
 'amonia': 0.02  
 },  
 'Tambaqui': {  
 'ph': [6.5, 8.5],  
 'temperatura': [28, 32],  
 'oxigenio': 5.0,  
 'amonia': 0.05  
 },  
 'Camarão Vannamei': {  
 'ph': [7.5, 8.5],  
 'temperatura': [28, 32],  
 'oxigenio': 4.0,  
 'amonia': 0.01  
 },  
 };  
  
 void analisarParametros() {  
 final double ph = double.tryParse(\_phController.text) ?? 0.0;  
 final double temperatura = double.tryParse(\_temperaturaController.text) ?? 0.0;  
 final double oxigenio = double.tryParse(\_oxigenioController.text) ?? 0.0;  
 final double amonia = double.tryParse(\_amoniaController.text) ?? 0.0;  
 final especie = parametrosEspecie[\_especieSelecionada];  
  
 List<String> resultado = [];  
  
 if (ph < especie['ph'][0] || ph > especie['ph'][1]) {  
 resultado.add("pH fora do ideal (${especie['ph'][0]} - ${especie['ph'][1]})");  
 }  
 if (temperatura < especie['temperatura'][0] || temperatura > especie['temperatura'][1]) {  
 resultado.add("Temperatura fora do ideal (${especie['temperatura'][0]} - ${especie['temperatura'][1]}°C)");  
 }  
 if (oxigenio < especie['oxigenio']) {  
 resultado.add("Oxigênio baixo (mínimo ${especie['oxigenio']} mg/L)");  
 }  
 if (amonia > especie['amonia']) {  
 resultado.add("Amônia alta (máximo ${especie['amonia']} mg/L)");  
 }  
  
 setState(() {  
 \_resultado = resultado.isEmpty ? ["Parâmetros ideais para o cultivo."] : resultado;  
 });  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(title: const Text('Análise da Água para Cultivo')),  
 body: Padding(  
 padding: const EdgeInsets.all(16.0),  
 child: Form(  
 key: \_formKey,  
 child: ListView(  
 children: [  
 TextFormField(  
 controller: \_phController,  
 keyboardType: TextInputType.number,  
 decoration: const InputDecoration(labelText: 'pH da água'),  
 ),  
 TextFormField(  
 controller: \_temperaturaController,  
 keyboardType: TextInputType.number,  
 decoration: const InputDecoration(labelText: 'Temperatura (°C)'),  
 ),  
 TextFormField(  
 controller: \_oxigenioController,  
 keyboardType: TextInputType.number,  
 decoration: const InputDecoration(labelText: 'Oxigênio dissolvido (mg/L)'),  
 ),  
 TextFormField(  
 controller: \_amoniaController,  
 keyboardType: TextInputType.number,  
 decoration: const InputDecoration(labelText: 'Amônia (mg/L)'),  
 ),  
 const SizedBox(height: 12),  
 DropdownButtonFormField<String>(  
 value: \_especieSelecionada,  
 items: parametrosEspecie.keys.map((especie) {  
 return DropdownMenuItem(  
 value: especie,  
 child: Text(especie),  
 );  
 }).toList(),  
 onChanged: (valor) {  
 setState(() {  
 \_especieSelecionada = valor!;  
 });  
 },  
 decoration: const InputDecoration(labelText: 'Espécie cultivada'),  
 ),  
 const SizedBox(height: 20),  
 ElevatedButton(  
 onPressed: analisarParametros,  
 child: const Text('Analisar'),  
 ),  
 const SizedBox(height: 20),  
 ...\_resultado.map((res) => Text(res, style: const TextStyle(fontSize: 16))).toList(),  
 ],  
 ),  
 ),  
 ),  
 );  
 }  
}