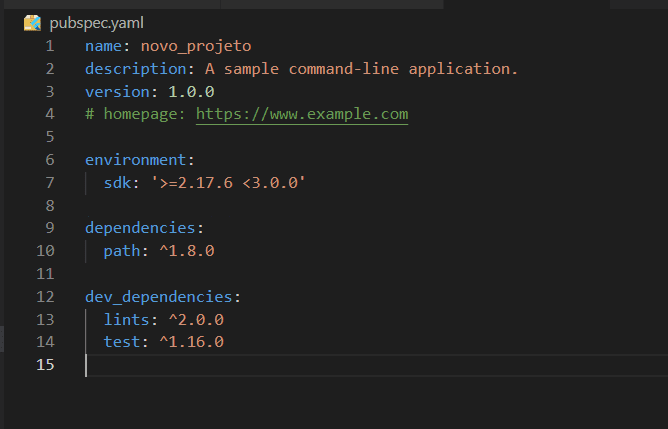
O que é Arquivo pubspec.yaml

Ao instalar pacotes em nossas aplicações, eles devem ficar listados de alguma forma para que outros desenvolvedores consigam efetuar novas instalações e/ou possam fazer novas alterações/atualizações. Para isso existe o arquivo pubspec.yaml

Nele também estão contidas informações como a versão Dart, nome, descrição, versão do projeto, etc.



É importante ressaltar que arquivos, como o pubspec, tem a extensão .yaml ele é um tipo de arquivo que considera a sua indentação.

Arquivos com a extensão yaml é um tipo de arquivo de serialização de dados legíveis por humanos, inspirado em linguagens como XML e Python.

Adicionando o plugin image\_picker

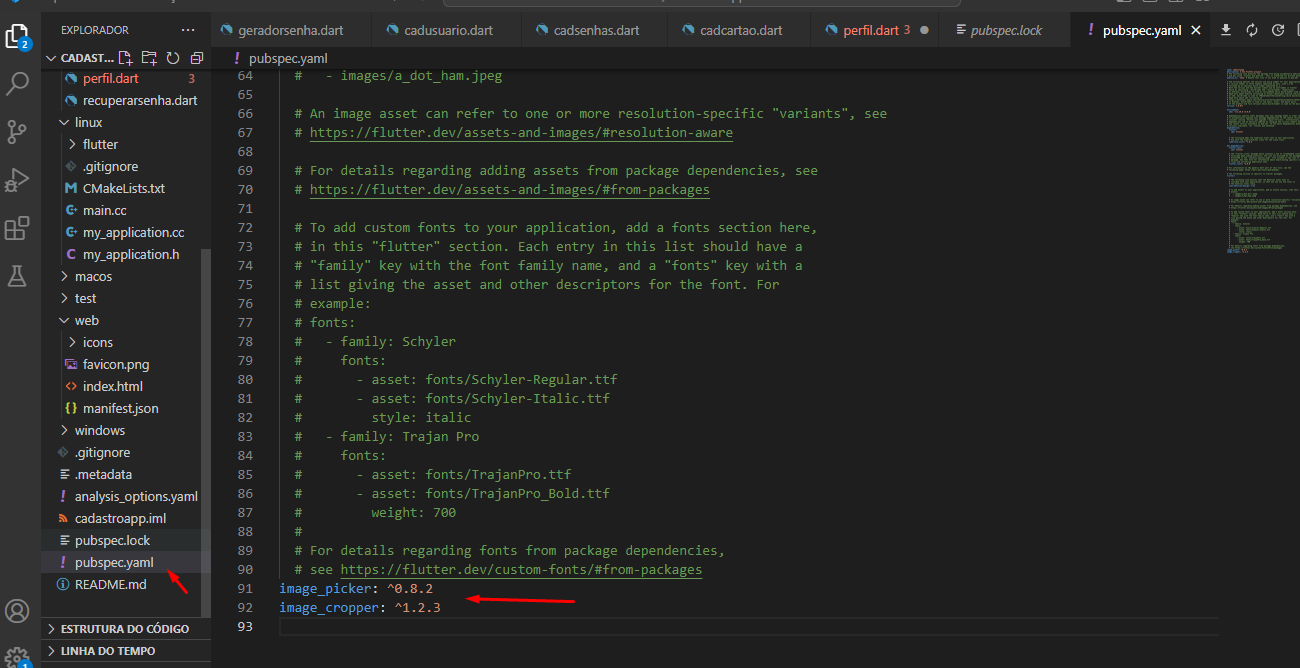
A próxima etapa é adicionar o plugin image\_picker ao nosso Projeto Flutter.

Abra o arquivo pubspec.yaml e adicione o image\_picker à seção de dependências:

dependencies: flutter: sdk: flutter image\_picker: ^ 0.8.2

Se você estiver usando VS Code, ele puxará automaticamente o image\_picker assim que você salvar o arquivo pubspec.yaml. Se você não estiver usando o VS Code, execute o seguinte comando para obter a dependência adicionada mais recente:

flutter pub get



Image\_picker configurações

No arquivo ios/Runner/Info.plist adicionar as permissões para acessar a câmera e galeria no iphone<key>NSCameraUsageDescription</key>

**<string>App precisa de acesso para tirar fotos</string>**

**<key>NSPhotoLibraryUsageDescription</key>**

**<string>App precisa de acesso a galeria</string>.**

Image\_crop configurações

No arquivo android/app/src/main/AndroidManifest.xml

**<activity**

**android:name="com.yalantis.ucrop.UCropActivity"**

**android:screenOrientation="portrait"**

**android:theme="@style/Theme.AppCompat.Light.NoActionBar"**

**/>**

// main.dart

import 'package:flutter/material.dart';

import 'dart:io';

import 'package:image\_picker/image\_picker.dart';

class perfilUsuario extends StatelessWidget {

  const perfilUsuario({Key? key}) : super(key: key);

  @override

  Widget build(BuildContext context) {

    return const MaterialApp(

      title: 'Perfil do usuário',

      home: HomePage(),

    );

  }

}

class HomePage extends StatefulWidget {

  const HomePage({Key? key}) : super(key: key);

  @override

  State<HomePage> createState() => \_HomePageState();

}

class \_HomePageState extends State<HomePage> {

  File? \_image;

  final \_picker = ImagePicker();

  Future<void> \_openImagePicker() async {

    final XFile? pickedImage =

        await \_picker.pickImage(source: ImageSource.gallery);

    if (pickedImage != null) {

      setState(() {

        \_image = File(pickedImage.path);

      });

    }

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

        appBar: AppBar(

          title: const Text('Perfil do usuário'),

        ),

        body: SafeArea(

          child: Padding(

            padding: const EdgeInsets.all(35),

            child: Column(children: [

              Center(

                child: ElevatedButton(

                  onPressed: \_openImagePicker,

                  child: const Text('Selecione a imagem'),

                ),

              ),

              const SizedBox(height: 35),

              Container(

                alignment: Alignment.center,

                width: double.infinity,

                height: 300,

                color: Colors.grey[300],

                child: \_image != null

                    ? Image.file(\_image!, fit: BoxFit.cover)

                    : const Text('Por favor selecione a imagem'),

              ),

              const SizedBox(height: 35),

              fieldNome(),

              const SizedBox(height: 35),

              fieldEmail(),

              const SizedBox(height: 35),

              fieldSenha(),

            ]),

          ),

        ));

  }

}

Widget fieldEmail() {

  return TextFormField(

      keyboardType: TextInputType.emailAddress,

      decoration:

          InputDecoration(border: OutlineInputBorder(), labelText: 'E-mail'));

}

Widget fieldSenha() {

  return TextFormField(

      keyboardType: TextInputType.emailAddress,

      decoration:

          InputDecoration(border: OutlineInputBorder(), labelText: 'Senha'));

}

Widget fieldNome() {

  return TextFormField(

      keyboardType: TextInputType.emailAddress,

      decoration:

          InputDecoration(border: OutlineInputBorder(), labelText: 'Nome'));

}