**CENTRO UNIVERSITÁRIO DE VOTUPORANGA – UNIFEV**

**BRUNO APARECIDO ZUPP NOVELI R.A: 95110**

**CURSO: ENGENHARIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO 8**

**2° BIMESTRE**

**VOTUPORANGA**

**2020**

**LISTA DE FIGURAS**

[Figura 1 - Sobre Mim - 1 4](#_Toc58085623)

[Figura 2 - Sobre Mim - 2 5](#_Toc58085624)

[Figura 3 - Mensagem de aviso sobre o whatsapp 6](#_Toc58085625)

[Figura 4 - Navegação para o facebook 7](#_Toc58085626)

[Figura 5 - Navegação para o instagram 8](#_Toc58085627)

[Figura 6 - Navegação para o linkedin 9](#_Toc58085628)

[Figura 7 - Navegação para o whatsapp 10](#_Toc58085629)

[Figura 8 - Formações 11](#_Toc58085630)

[Figura 9 - Experiências 12](#_Toc58085631)

[Figura 10 - Projetos 13](#_Toc58085632)

[Figura 11 - Habilidades 14](#_Toc58085633)

[Figura 12 - Página de erro caso a requisição falhe 15](#_Toc58085634)

[Figura 13 - Página de carregamento 16](#_Toc58085635)

**SUMÁRIO**

[1 ESCOPO 3](#_Toc58085636)

[2 TELA - SOBRE MIM 3](#_Toc58085637)

[3 TELA - FORMAÇÕES 11](#_Toc58085638)

[4 TELA – EXPERIÊNCIAS 12](#_Toc58085639)

[5 TELA – PROJETOS 13](#_Toc58085640)

[6 TELA – HABILIDADES 14](#_Toc58085641)

[7 TELAS – ESTADOS 15](#_Toc58085642)

[8 CÓDIGOS 17](#_Toc58085643)

[8.1 **Controllers** 17](#_Toc58085644)

[8.2 **Core** 18](#_Toc58085645)

[8.3 **Interfaces** 18](#_Toc58085646)

[8.4 **Models** 18](#_Toc58085647)

[8.5 **Pages** 24](#_Toc58085648)

[8.6 **Pubspec.yaml** 43](#_Toc58085649)

# ESCOPO

Projeto desenvolvido usando o framework flutter para a apresentação do portfólio utilizando a *API* desenvolvida na matéria de TLP II. Foi utilizado o esquema de *PageView* para a navegação entra as telas, o gerenciador de estado *GetX* que a partir da nova versão lançada também disponibiliza métodos para trabalhar com requisições *HTTP* e o plugin expandable para a listagem de grupos de muita informação.

# TELA - SOBRE MIM

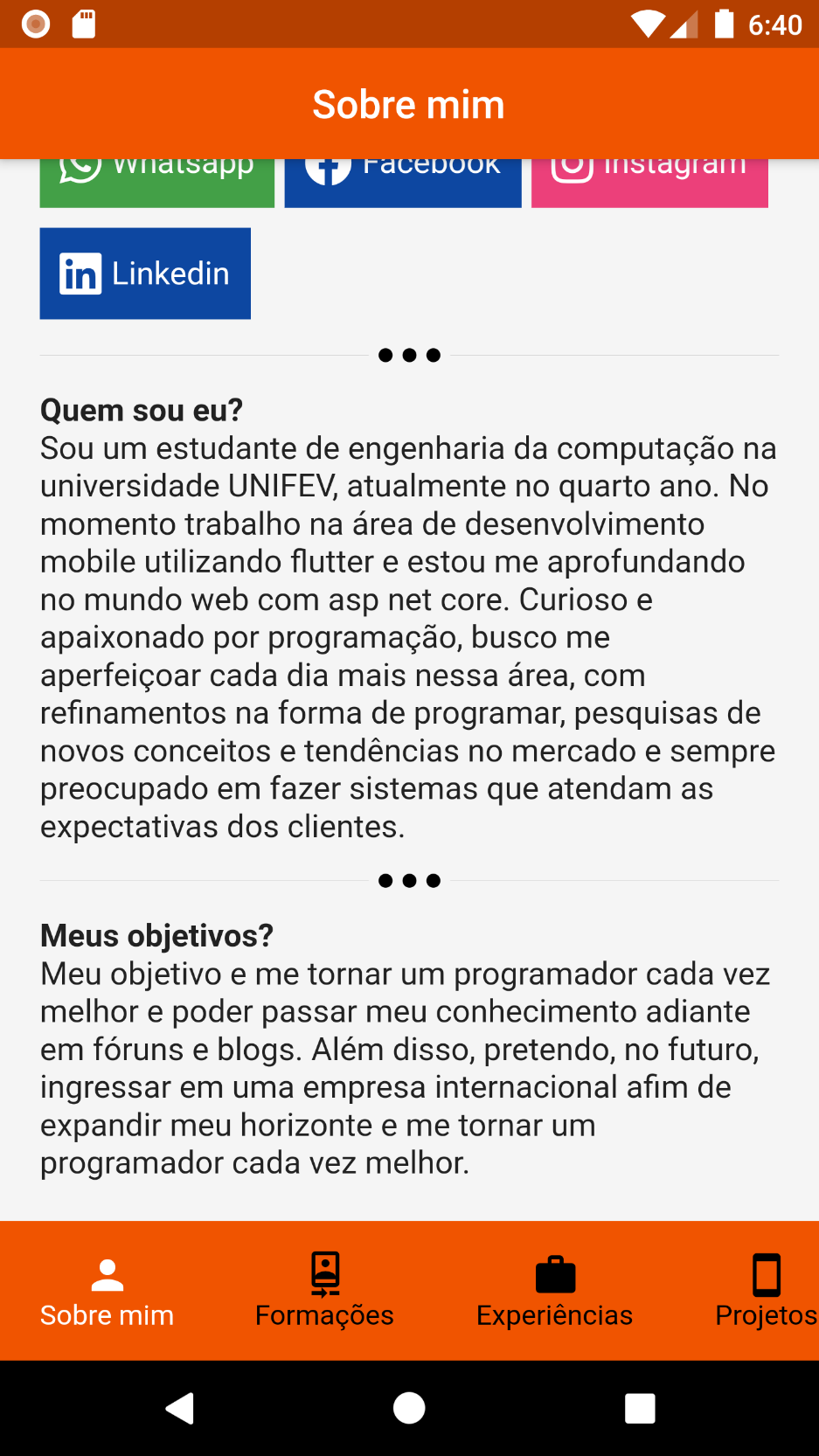
A seguir vai ser mostrada fotos relacionadas ao grupo “Sobre mim” composto por informações básicas para a identificação, além de contar com sistema de navegação para as redes sociais Whatsapp, Facebook, Instagram e Linkedin. No caso do whatsapp, caso a pessoa não possua o aplicativo instalado ou por algum motivo interno vai mostrar uma notificação conhecida como *Snackbar*. Esse sistema de links foi trabalhado utilizando o plugin *url\_launcher*.

Figura 1 - Sobre Mim - 1



Fonte: próprio autor

Figura 2 - Sobre Mim - 2



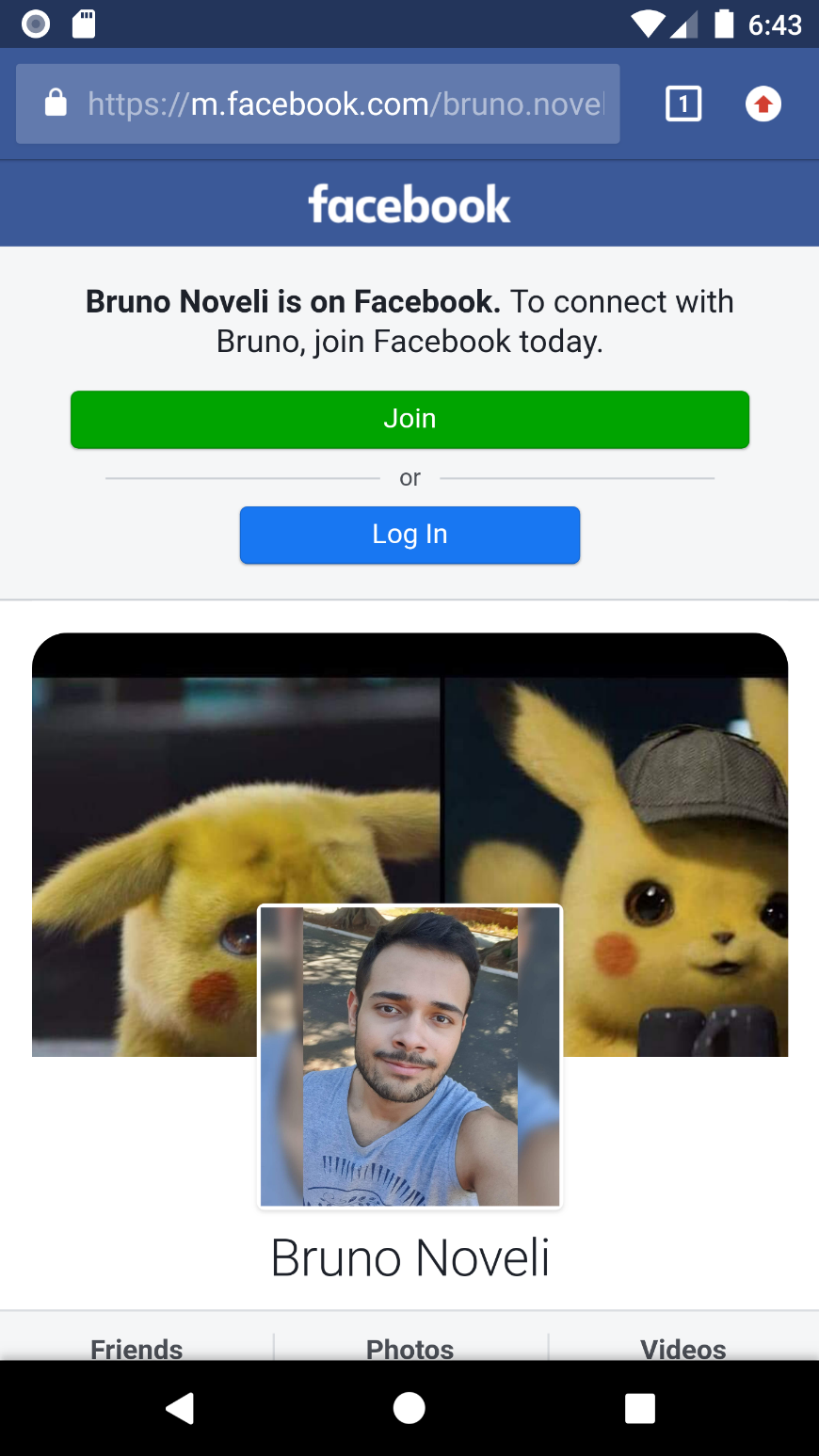
Fonte: próprio autor

Figura 3 - Mensagem de aviso sobre o whatsapp



Fonte: próprio autor

Figura 4 - Navegação para o facebook



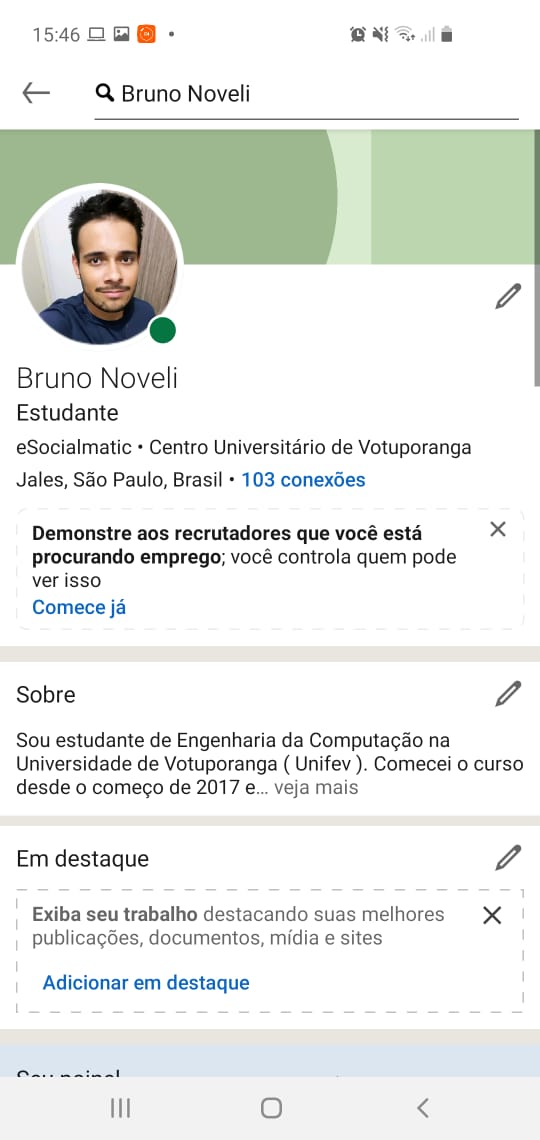
Fonte: próprio autor

Figura 5 - Navegação para o instagram



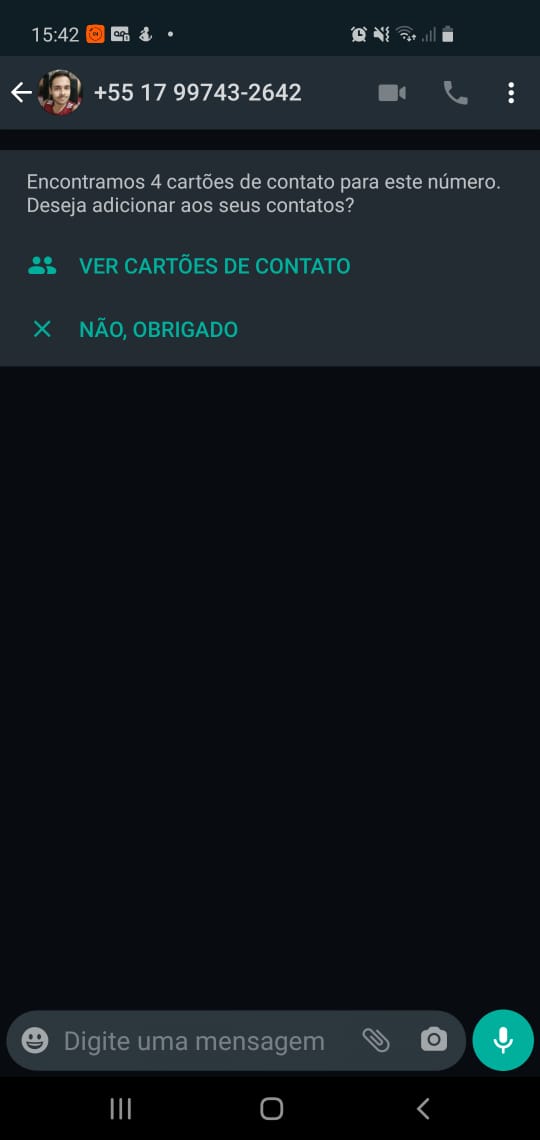
Fonte: próprio autor

Figura 6 - Navegação para o linkedin



Fonte: próprio autor

Figura 7 - Navegação para o whatsapp



Fonte: próprio autor

# TELA - FORMAÇÕES

Página que vai mostrar todas as formações de cursos já feitos. Foi utilizado um ListView para esse processo.

Figura 8 - Formações

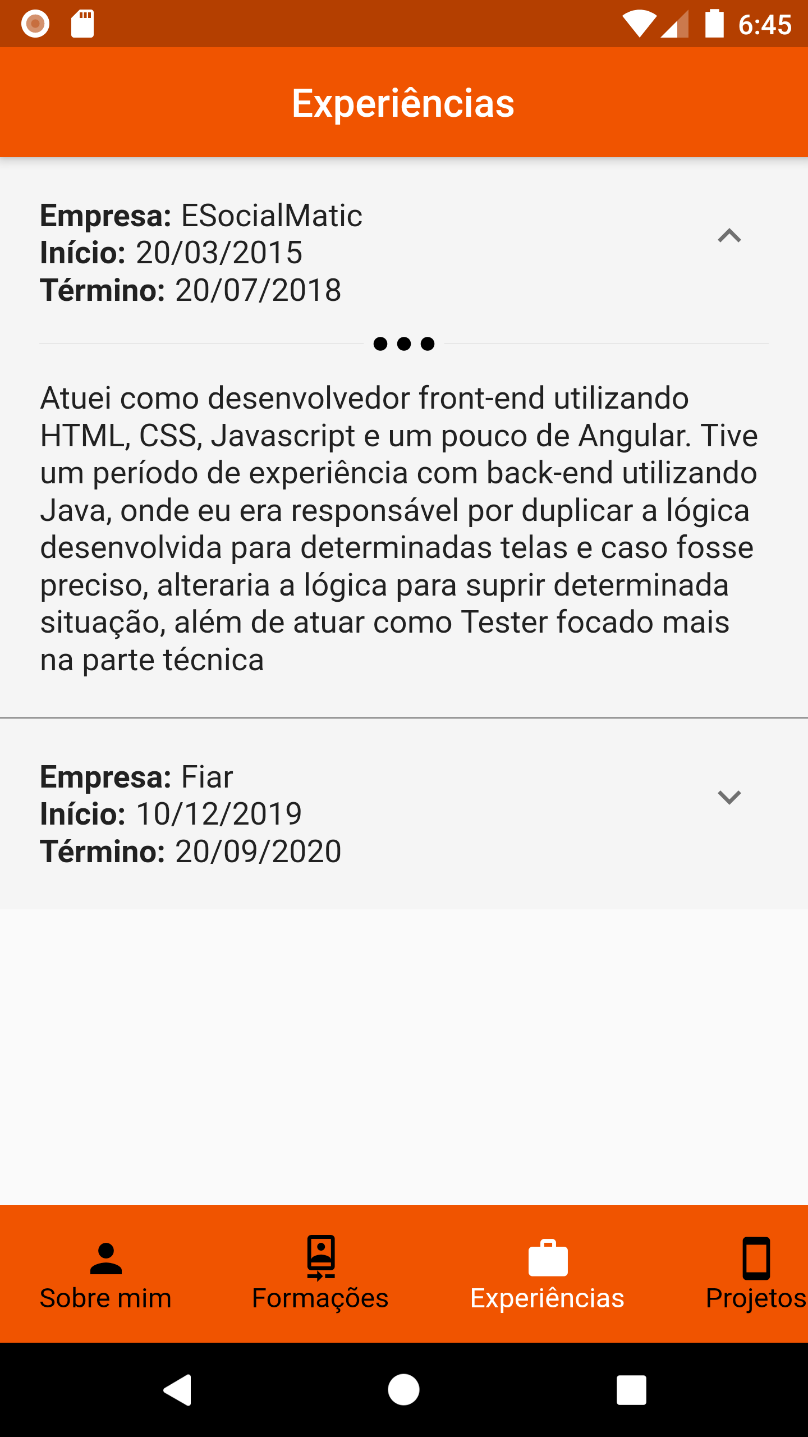


Fonte: próprio autor

# TELA – EXPERIÊNCIAS

Página que vai mostrar todas as experiências profissionais já obtidas em empresas. Nessa tela foi utilizado o plugin expandable pois o montante de informação mostrada ficaria inadequada se fosse colocado para a visualização direta. Para isso, a pessoa vai clicar em um dos itens da lista e com isso vai expandir para mostrar os detalhes da experiência.

Figura 9 - Experiências

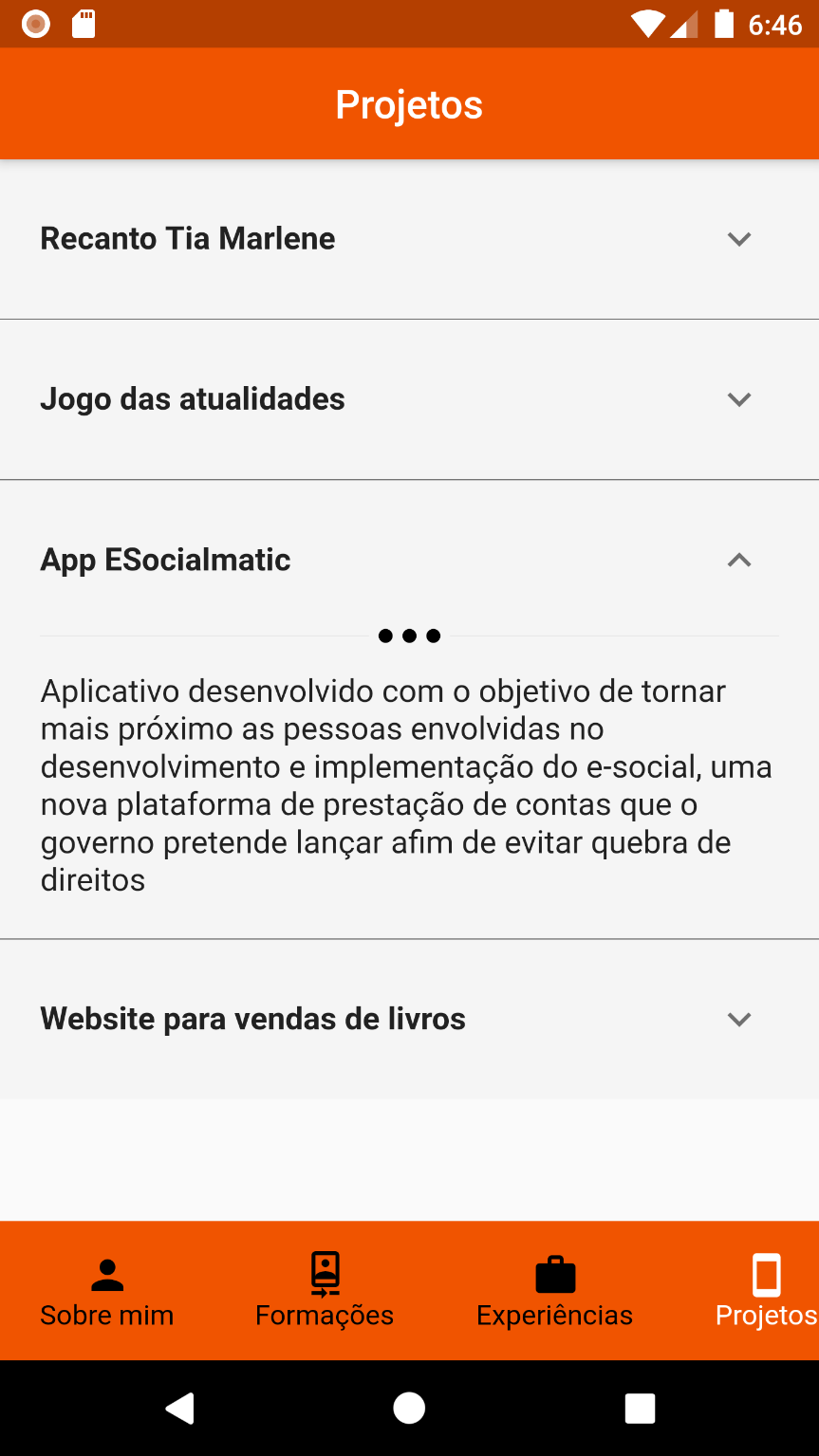


Fonte: próprio autor

# TELA – PROJETOS

Página que vai mostrar todas os projetos já feitos. Nessa tela foi utilizado o plugin expandable pois o montante de informação mostrada ficaria inadequada se fosse colocado para a visualização direta. Para isso, a pessoa vai clicar em um dos itens da lista e com isso vai expandir para mostrar os detalhes do projeto.

Figura 10 - Projetos



Fonte: próprio autor

# TELA – HABILIDADES

Nessa tela vai ser mostrado todas as habilidades com uma pontuação de 0 á 5. Foi utilizado uma listagem simples com o ListView

Figura 11 - Habilidades



Fonte: próprio autor

# TELAS – ESTADOS

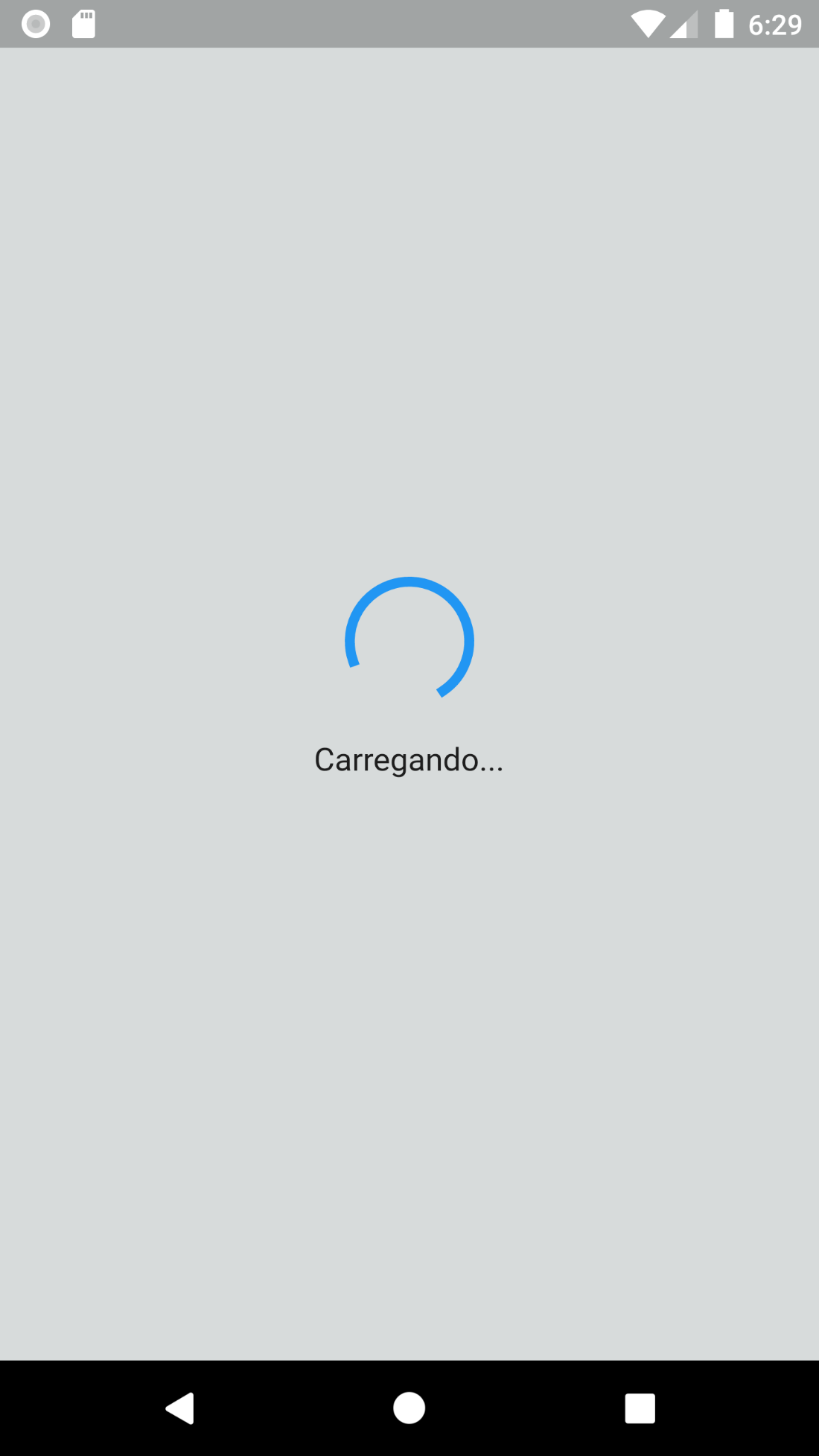
Afim de agregar um feedback do que está acontecendo, foi criado duas telas com o intuito de dizer o estado atual do aplicativo em relação a chamada á API.

Figura 12 - Página de erro caso a requisição falhe



Fonte: próprio autor

Figura 13 - Página de carregamento



Fonte: próprio autor

# CÓDIGOS

## **Controllers**

portfolio\_controller.dart

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/interfaces/idetail\_service.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/portfolio.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/services/detail\_service.dart';

class *PortfolioController* extends *GetxController* {

  static *PortfolioController* get to => *Get*.find();

*IDetailService* \_detailService;

  var \_numberPage = 0.obs;

*int* get numberPage => \_numberPage.value;

  set numberPage(*int* value) => \_numberPage(value);

  @override

  void onInit() {

*super*.onInit();

    \_detailService = *DetailService*();

  }

*Future*<*Portfolio*> getPortfolio() async {

    var result = await \_detailService.getPortfolio();

    return result;

  }

}

## **Core**

utils.dart

abstract class *Utils* {

  static *String* shortDate(*DateTime* dateTime) {

    if(dateTime == null) return "Data não informada";

*String* date = dateTime.toString().replaceAll("-", "/").substring(0, 10);

    return date.split("/").reversed.join("/");

  }

  static *String* age(*DateTime* birthDate) {

    if(birthDate == null) return "Data não informada";

    var days = *DateTime*.now().difference(birthDate).inDays / 365;

    return days.toInt().toString();

  }

}

## **Interfaces**

idetail\_service.dart

import 'package:portfolio\_prova\_segundo\_bimestre/models/portfolio.dart';

abstract class *IDetailService* {

*Future*<*Portfolio*> getPortfolio();

}

## **Models**

academic\_training.dart

import 'dart:convert';

class *AcademicTraining* {

*AcademicTraining*({

*this*.id,

*this*.name,

*this*.institution,

*this*.begin,

*this*.end,

  });

*int* id;

*String* name;

*String* institution;

*DateTime* begin;

*DateTime* end;

  factory *AcademicTraining*.fromJson(*String* str) => *AcademicTraining*.fromMap(json.decode(str));

*String* toJson() => json.encode(toMap());

  factory *AcademicTraining*.fromMap(*Map*<*String*, *dynamic*> json) => *AcademicTraining*(

    id: json["id"],

    name: json["name"],

    institution: json["institution"],

    begin: *DateTime*.parse(json["begin"]),

    end: *DateTime*.parse(json["end"]),

  );

*Map*<*String*, *dynamic*> toMap() => {

    "id": id,

    "name": name,

    "institution": institution,

    "begin": begin.toIso8601String(),

    "end": end.toIso8601String(),

  };

}

detail.dart

import 'dart:convert';

class *Detail* {

*Detail*({

*this*.id,

*this*.name,

*this*.email,

*this*.phone,

*this*.birthDate,

*this*.linkedin,

*this*.instagram,

*this*.facebook,

*this*.nationality,

*this*.about,

*this*.goals,

*this*.photo,

  });

*int* id;

*String* name;

*String* email;

*String* phone;

*DateTime* birthDate;

*String* linkedin;

*String* instagram;

*String* facebook;

*String* nationality;

*String* about;

*String* goals;

*String* photo;

  factory *Detail*.fromJson(*String* str) => *Detail*.fromMap(json.decode(str));

*String* toJson() => json.encode(toMap());

  factory *Detail*.fromMap(*Map*<*String*, *dynamic*> json) => *Detail*(

    id: json["id"],

    name: json["name"],

    email: json["email"],

    phone: json["phone"],

    birthDate: *DateTime*.parse(json["birthDate"]),

    linkedin: json["linkedin"],

    instagram: json["instagram"],

    facebook: json["facebook"],

    nationality: json["nationality"],

    about: json["about"],

    goals: json["goals"],

    photo: json["photo"],

  );

*Map*<*String*, *dynamic*> toMap() => {

    "id": id,

    "name": name,

    "email": email,

    "phone": phone,

    "birthDate": birthDate.toIso8601String(),

    "linkedin": linkedin,

    "instagram": instagram,

    "facebook": facebook,

    "nationality": nationality,

    "about": about,

    "goals": goals,

    "photo": photo,

  };

}

experience.dart

import 'dart:convert';

class *Experience* {

*Experience*({

*this*.id,

*this*.company,

*this*.description,

*this*.begin,

*this*.end,

  });

*int* id;

*String* company;

*String* description;

*DateTime* begin;

*DateTime* end;

  factory *Experience*.fromJson(*String* str) => *Experience*.fromMap(json.decode(str));

*String* toJson() => json.encode(toMap());

  factory *Experience*.fromMap(*Map*<*String*, *dynamic*> json) => *Experience*(

    id: json["id"],

    company: json["company"],

    description: json["description"],

    begin: *DateTime*.parse(json["begin"]),

    end: *DateTime*.parse(json["end"]),

  );

*Map*<*String*, *dynamic*> toMap() => {

    "id": id,

    "company": company,

    "description": description,

    "begin": begin.toIso8601String(),

    "end": end.toIso8601String(),

  };

}

portfolio.dart

import 'dart:convert';

import 'package:portfolio\_prova\_segundo\_bimestre/models/academic\_training.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/detail.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/experience.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/project.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/skill.dart';

class *Portfolio* {

*Portfolio*({

*this*.detail,

*this*.academicTrainings,

*this*.experiences,

*this*.projects,

*this*.skills,

  });

*Detail* detail;

*List*<*AcademicTraining*> academicTrainings;

*List*<*Experience*> experiences;

*List*<*Project*> projects;

*List*<*Skill*> skills;

  factory *Portfolio*.fromJson(*String* str) => *Portfolio*.fromMap(json.decode(str));

*String* toJson() => json.encode(toMap());

  factory *Portfolio*.fromMap(*Map*<*String*, *dynamic*> json) => *Portfolio*(

    detail: *Detail*.fromMap(json["detail"]),

    academicTrainings: *List*<*AcademicTraining*>.from(json["academicTrainings"].map((x) => *AcademicTraining*.fromMap(x))),

    experiences: *List*<*Experience*>.from(json["experiences"].map((x) => *Experience*.fromMap(x))),

    projects: *List*<*Project*>.from(json["projects"].map((x) => *Project*.fromMap(x))),

    skills: *List*<*Skill*>.from(json["skills"].map((x) => *Skill*.fromMap(x))),

  );

*Map*<*String*, *dynamic*> toMap() => {

    "detail": detail.toMap(),

    "academicTrainings": *List*<*dynamic*>.from(academicTrainings.map((x) => x.toMap())),

    "experiences": *List*<*dynamic*>.from(experiences.map((x) => x.toMap())),

    "projects": *List*<*dynamic*>.from(projects.map((x) => x.toMap())),

    "skills": *List*<*dynamic*>.from(skills.map((x) => x.toMap())),

  };

}

project.dart

import 'dart:convert';

class *Project* {

*Project*({

*this*.id,

*this*.name,

*this*.description,

  });

*int* id;

*String* name;

*String* description;

  factory *Project*.fromJson(*String* str) => *Project*.fromMap(json.decode(str));

*String* toJson() => json.encode(toMap());

  factory *Project*.fromMap(*Map*<*String*, *dynamic*> json) => *Project*(

    id: json["id"],

    name: json["name"],

    description: json["description"],

  );

*Map*<*String*, *dynamic*> toMap() => {

    "id": id,

    "name": name,

    "description": description,

  };

}

skill.dart

import 'dart:convert';

class *Skill* {

*Skill*({

*this*.id,

*this*.name,

*this*.aptitude,

  });

*int* id;

*String* name;

*double* aptitude;

  factory *Skill*.fromJson(*String* str) => *Skill*.fromMap(json.decode(str));

*String* toJson() => json.encode(toMap());

  factory *Skill*.fromMap(*Map*<*String*, *dynamic*> json) => *Skill*(

    id: json["id"],

    name: json["name"],

    aptitude: json["aptitude"].toDouble(),

  );

*Map*<*String*, *dynamic*> toMap() => {

    "id": id,

    "name": name,

    "aptitude": aptitude,

  };

}

## **Pages**

about\_me\_page.dart

import 'package:flutter/material.dart';

import 'package:font\_awesome\_flutter/font\_awesome\_flutter.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/core/utils.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/detail.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/widgets/inline\_information.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/widgets/separator.dart';

import 'package:url\_launcher/url\_launcher.dart';

class *AboutMePage* extends *StatelessWidget* {

  final *Detail* detail;

  const *AboutMePage*({*Key* key, @required *this*.detail}) : *super*(key: key);

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      appBar: *AppBar*(

        backgroundColor: *Color*.fromARGB(255,240,84,0),

        title: *Text*("Sobre mim"),

        centerTitle: true,

      ),

      backgroundColor: *Colors*.grey[100],

      body: *SingleChildScrollView*(

        child: *Column*(

          crossAxisAlignment: *CrossAxisAlignment*.start,

          children: [

*Stack*(

              children: [

*Container*(

                  height: 250,

                  decoration: *BoxDecoration*(

                    image: *DecorationImage*(

                      colorFilter: *ColorFilter*.mode(

*Colors*.black.withOpacity(0.6),

*BlendMode*.darken

                      ),

                      fit: *BoxFit*.fitWidth,

                      image: *AssetImage*("assets/fundo\_programacao.png"),

                    )

                  ),

                ),

*Align*(

                  alignment: *Alignment*.center,

                  child: *Container*(

                    margin: *EdgeInsets*.only(top: 20),

                    height: 250,

                    width: 200,

                    decoration: *BoxDecoration*(

                      image: *DecorationImage*(

                        fit: *BoxFit*.contain,

                        image: *NetworkImage*(detail.photo)

                      )

                    ),

                  ),

                ),

              ],

            ),

*Container*(

              padding: *EdgeInsets*.all(20),

              child: *Column*(

                crossAxisAlignment: *CrossAxisAlignment*.start,

                children: [

*InlineInformation*(title: "Nome: ", information: detail.name),

*InlineInformation*(title: "Nacionalidade: ", information: detail.nationality),

*InlineInformation*(title: "Idade: ", information: *Utils*.age(detail.birthDate) + " anos"),

*Separator*(),

*Text*(

                    "Informações de contato",

                    style: *TextStyle*(

                      fontWeight: *FontWeight*.bold,

                      fontSize: 16,

                    ),

                  ),

*InlineInformation*(title: "Email: ", information: detail.email),

*InlineInformation*(title: "Telefone: ", information: detail.phone),

*SizedBox*(height: 10),

*Wrap*(

                    runSpacing: 10,

                    spacing: 5,

                    children: [

                      \_cardContact(

                        iconData: *FontAwesomeIcons*.whatsapp,

                        description: "Whatsapp",

                        colorContainer: *Colors*.green[600],

                        function: () async {

                          var whatsappUrl = "whatsapp://send?phone=+" + detail.phone;

                          if(await canLaunch(whatsappUrl)) {

                            await launch(whatsappUrl);

                          } else {

*Get*.snackbar(

                              "Ação não aceita",

                              "Por algum problema interno, não foi possível conectar ao whatsapp",

                              duration: *Duration*(seconds: 2),

                              backgroundColor: *Colors*.grey[350]

                            );

                          }

                        }

                      ),

                      \_cardContact(

                        iconData: *FontAwesomeIcons*.facebook,

                        description: "Facebook",

                        colorContainer: *Colors*.blue[900],

                        function: () async {

                          await launch(detail.facebook);

                        }

                      ),

                      \_cardContact(

                        iconData: *FontAwesomeIcons*.instagram,

                        description: "Instagram",

                        colorContainer: *Colors*.pink[400],

                        function: () async {

                          await launch(detail.instagram);

                        }

                      ),

                      \_cardContact(

                        iconData: *FontAwesomeIcons*.linkedin,

                        description: "Linkedin",

                        colorContainer: *Colors*.blue[900],

                        function: () async {

                          await launch(detail.linkedin);

                        }

                      ),

                    ],

                  ),

*Separator*(),

*Text*(

                    "Quem sou eu?",

                    style: *TextStyle*(

                      fontWeight: *FontWeight*.bold,

                      fontSize: 16

                    ),

                  ),

*Text*(

                    detail.about,

                    style: *TextStyle*(

                      fontSize: 16

                    ),

                  ),

*Separator*(),

*Text*(

                    "Meus objetivos?",

                    style: *TextStyle*(

                      fontWeight: *FontWeight*.bold,

                      fontSize: 16

                    ),

                  ),

*Text*(

                    detail.goals,

                    style: *TextStyle*(

                      fontSize: 16

                    ),

                  ),

                ],

              ),

            ),

          ],

        ),

      ),

    );

  }

*Widget* \_cardContact({

    @required *IconData* iconData,

    @required *Color* colorContainer,

*Color* colorIcon = *Colors*.white,

    @required *String* description,

*Function* function

  }) {

    return *GestureDetector*(

      onTap: function ?? () {},

      child: *Container*(

        padding: *EdgeInsets*.all(10),

        decoration: *BoxDecoration*(

          color: colorContainer

        ),

        child: *Row*(

          mainAxisSize: *MainAxisSize*.min,

          children: [

*FaIcon*(iconData, color: colorIcon),

*SizedBox*(width: 5),

*Text*(

              description,

              style: *TextStyle*(

                fontSize: 16,

                color: *Colors*.white

              ),

            ),

          ],

        ),

      ),

    );

  }

}

academic\_trainings\_page.dart

import 'package:flutter/material.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/core/utils.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/academic\_training.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/widgets/inline\_information.dart';

class *AcademicTrainingPage* extends *StatelessWidget* {

  final *List*<*AcademicTraining*> academicTrainings;

  const *AcademicTrainingPage*({

*Key* key, @required *this*.academicTrainings

  }) : *super*(key: key);

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      appBar: *AppBar*(

        backgroundColor: *Color*.fromARGB(255,240,84,0),

        title: *Text*("Formações"),

        centerTitle: true,

      ),

      body: *ListView*.separated(

        separatorBuilder: (context, index) => *Container*(

          width: *Get*.width,

          height: 0.7,

          color: *Colors*.black54

        ),

        itemCount: academicTrainings.length,

        itemBuilder: (context, index) {

          var training = academicTrainings[index];

          return *Container*(

            decoration: *BoxDecoration*(

              color: *Colors*.grey[100]

            ),

            child: *Padding*(

              padding: const *EdgeInsets*.only(top: 10, bottom: 10),

              child: *ListTile*(

                title: *Column*(

                  crossAxisAlignment: *CrossAxisAlignment*.start,

                  children: [

*InlineInformation*(

                      title: "Formação: ",

                      information: training.name

                    ),

*InlineInformation*(

                      title: "Instituição: ",

                      information: training.institution

                    ),

                  ],

                ),

                subtitle: *Column*(

                  crossAxisAlignment: *CrossAxisAlignment*.start,

                  children: [

*InlineInformation*(

                      title: "Início: ",

                      information: *Utils*.shortDate(training.begin)

                    ),

*InlineInformation*(

                      title: "Término: ",

                      information: *Utils*.shortDate(training.end)

                    ),

                  ],

                ),

              ),

            ),

          );

        },

      ),

    );

  }

}

experiences\_page.dart

import 'package:expandable/expandable.dart';

import 'package:flutter/material.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/core/utils.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/experience.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/widgets/inline\_information.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/widgets/separator.dart';

class *ExperiencePage* extends *StatelessWidget* {

  final *List*<*Experience*> experiences;

  const *ExperiencePage*({

*Key* key, @required *this*.experiences

  }) : *super*(key: key);

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      appBar: *AppBar*(

        backgroundColor: *Color*.fromARGB(255,240,84,0),

        title: *Text*("Experiências"),

        centerTitle: true,

      ),

      body: *Container*(

        child: *ListView*.separated(

          separatorBuilder: (context, index) => *Container*(

            width: *Get*.width,

            height: 0.7,

            color: *Colors*.black54

          ),

          itemCount: experiences.length,

          itemBuilder: (context, index) {

            var experience = experiences[index];

            return *Container*(

              decoration: *BoxDecoration*(

                color: *Colors*.grey[100]

              ),

              child: *Container*(

                padding: *EdgeInsets*.all(20),

                child: *ExpandablePanel*(

                  header: *Column*(

                    crossAxisAlignment: *CrossAxisAlignment*.start,

                    children: [

*InlineInformation*(

                        title: "Empresa: ",

                        information: experience.company,

                      ),

*InlineInformation*(

                        title: "Início: ",

                        information: *Utils*.shortDate(experience.begin),

                      ),

*InlineInformation*(

                        title: "Término: ",

                        information: *Utils*.shortDate(experience.end),

                      ),

                    ],

                  ),

                  expanded: *Column*(

                    children: [

*Separator*(),

*Text*(

                        experience.description,

                        style: *TextStyle*(

                          fontSize: 16

                        ),

                      ),

                    ],

                  ),

                ),

              ),

            );

          },

        ),

      ),

    );

  }

}

portfolio\_page.dart

import 'package:flutter/material.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/controllers/portfolio\_controller.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/portfolio.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/presentation/about\_me\_page.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/presentation/academic\_training\_page.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/presentation/experience\_page.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/presentation/projects\_page.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/presentation/skill\_page.dart';

class *PortfolioPage* extends *GetView*<*PortfolioController*> {

  final *Portfolio* portfolio;

  final pageViewController = *PageController*(

    initialPage: 0,

  );

*PortfolioPage*({*this*.portfolio});

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      bottomNavigationBar: *Obx*(() => \_bottomNavigator(numberPage: controller.numberPage)),

      body: *Stack*(

        children: [

*PageView*(

            controller: pageViewController,

            physics: *NeverScrollableScrollPhysics*(),

            children: [

*AboutMePage*(detail: portfolio.detail),

*AcademicTrainingPage*(academicTrainings: portfolio.academicTrainings),

*ExperiencePage*(experiences: portfolio.experiences),

*ProjectsPage*(projects: portfolio.projects),

*SkillPage*(skills: portfolio.skills)

            ],

          ),

        ],

      ),

    );

  }

*Widget* \_bottomNavigator({*int* numberPage = 0}) {

    return *Container*(

      height: 70,

      color: *Color*.fromARGB(255,240,84,0),

      child: *ListView*(

        scrollDirection: *Axis*.horizontal,

          children: [

            \_itemOptionBottomNavigator(

              icon: *Icons*.person,

              description: "Sobre mim",

              numberPage: 0,

              isActivated: numberPage == 0,

            ),

            \_itemOptionBottomNavigator(

              icon: *Icons*.camera\_front\_outlined,

              description: "Formações",

              numberPage: 1,

              isActivated: numberPage == 1,

            ),

            \_itemOptionBottomNavigator(

              icon: *Icons*.work,

              description: "Experiências",

              numberPage: 2,

              isActivated: numberPage == 2,

            ),

            \_itemOptionBottomNavigator(

              icon: *Icons*.smartphone,

              description: "Projetos",

              numberPage: 3,

              isActivated: numberPage == 3,

            ),

            \_itemOptionBottomNavigator(

              icon: *Icons*.addchart\_outlined,

              description: "Habilidades",

              numberPage: 4,

              isActivated: numberPage == 4,

            ),

          ],

      ),

    );

  }

*Widget* \_itemOptionBottomNavigator({

    @required *IconData* icon,

    @required *String* description,

    @required *int* numberPage,

*bool* isActivated = false

  }) {

    return *InkWell*(

      onTap: () {

        \_updatePage(numberPage: numberPage);

      },

      child: *Container*(

        margin: *EdgeInsets*.symmetric(horizontal: 20),

        child: *Column*(

          crossAxisAlignment: *CrossAxisAlignment*.center,

          mainAxisAlignment: *MainAxisAlignment*.center,

          children: [

*Icon*(

              icon,

              color: isActivated ? *Colors*.white : *Colors*.black

            ),

*Text*(

              description,

              style: *TextStyle*(

                color: isActivated ? *Colors*.white : *Colors*.black

              ),

            ),

          ],

        ),

      ),

    );

  }

  void \_updatePage({@required *int* numberPage}) {

    controller.numberPage = numberPage;

    pageViewController.jumpToPage(numberPage);

  }

}

projects\_page.dart

import 'package:expandable/expandable.dart';

import 'package:flutter/material.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/project.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/widgets/separator.dart';

class *ProjectsPage* extends *StatelessWidget* {

  final *List*<*Project*> projects;

  const *ProjectsPage*({*Key* key, @required *this*.projects}) : *super*(key: key);

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      appBar: *AppBar*(

        backgroundColor: *Color*.fromARGB(255,240,84,0),

        title: *Text*("Projetos"),

        centerTitle: true,

      ),

      body: *Container*(

        child: *ListView*.separated(

          separatorBuilder: (context, index) => *Container*(

            width: *Get*.width,

            height: 0.7,

            color: *Colors*.black54

          ),

          itemCount: projects.length,

          itemBuilder: (context, index) {

            var project = projects[index];

            return *Container*(

              decoration: *BoxDecoration*(

                color: *Colors*.grey[100]

              ),

              child: *Container*(

                padding: *EdgeInsets*.all(20),

                child: *ExpandablePanel*(

                  header: *Padding*(

                    padding: const *EdgeInsets*.only(top: 10),

                    child: *Text*(

                      project.name,

                      style: *TextStyle*(

                        fontSize: 16,

                        fontWeight: *FontWeight*.bold

                      ),

                    ),

                  ),

                  expanded: *Column*(

                    children: [

*Separator*(),

*Text*(

                        project.description,

                        style: *TextStyle*(

                          fontSize: 16

                        ),

                      ),

                    ],

                  ),

                ),

              ),

            );

          },

        ),

      ),

    );

  }

}

skills\_page.dart

import 'package:flutter/material.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/skill.dart';

class *SkillPage* extends *StatelessWidget* {

  final *List*<*Skill*> skills;

  const *SkillPage*({*Key* key, @required *this*.skills}) : *super*(key: key);

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      appBar: *AppBar*(

        backgroundColor: *Color*.fromARGB(255,240,84,0),

        title: *Text*("Habilidades (0 á 5)"),

        centerTitle: true,

      ),

      body: *Container*(

        child: *ListView*.separated(

          separatorBuilder: (context, index) => *Container*(

            width: *Get*.width,

            height: 0.7,

            color: *Colors*.black54

          ),

          itemCount: skills.length,

          itemBuilder: (context, index) {

            var skill = skills[index];

            return *Container*(

              decoration: *BoxDecoration*(

                color: *Colors*.grey[100]

              ),

              child: *Container*(

                padding: *EdgeInsets*.fromLTRB(20,10,20,10),

                child: *Row*(

                  children: [

*Container*(

                      alignment: *Alignment*.center,

                      height: 60,

                      width: 60,

                      decoration: *BoxDecoration*(

                        color: \_colorByAptitude(skill.aptitude),

                        shape: *BoxShape*.circle

                      ),

                      child: *Text*(

                        skill.aptitude.toString(),

                        textAlign: *TextAlign*.center,

                        style: *TextStyle*(

                          fontSize: 16,

                          color: *Colors*.white

                        ),

                      ),

                    ),

*SizedBox*(width: 10),

*Text*(

                      skill.name,

                      style: *TextStyle*(

                        fontSize: 16,

                      ),

                    ),

                  ],

                ),

              ),

            );

          },

        ),

      ),

    );

  }

*Color* \_colorByAptitude(*double* value) {

    if(value >= 0.0 && value <= 1.0) return *Colors*.red[900];

    if(value > 1.0 && value <= 2.0) return *Colors*.yellow[900];

    if(value > 1.0 && value <= 3.0) return *Colors*.purple[900];

    if(value > 1.0 && value <= 4.0) return *Colors*.blue[900];

    if(value > 1.0 && value <= 5.0) return *Colors*.green[900];

    return *Colors*.black;

  }

}

error\_page.dart

import 'package:flutter/material.dart';

class *ErrorPage* extends *StatelessWidget* {

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      backgroundColor: *Colors*.red,

      body: *Center*(

        child: *Column*(

          mainAxisAlignment: *MainAxisAlignment*.center,

          children: [

*Container*(

              decoration: *BoxDecoration*(

                //color: Colors.white,

                shape: *BoxShape*.circle

              ),

              child: *Icon*(

*Icons*.error\_rounded,

                color: *Colors*.white,

                size: 250,

              ),

            ),

*SizedBox*(height: 10),

*Text*(

              "Algo inesperado ocorreu, sentimos muito.",

              style: *TextStyle*(

                fontSize: 16,

                color: *Colors*.white

              ),

            ),

          ],

        ),

      ),

    );

  }

}

loading\_page.dart

import 'package:flutter/material.dart';

class *LoadingPage* extends *StatelessWidget* {

  @override

*Widget* build(*BuildContext* context) {

    return *Scaffold*(

      backgroundColor: *Color*.fromARGB(255,215,219,219),

      body: *Center*(

        child: *Column*(

          mainAxisAlignment: *MainAxisAlignment*.center,

          children: [

*SizedBox*(

              height: 60,

              width: 60,

              child: *CircularProgressIndicator*(

                strokeWidth: 5,

              ),

            ),

*SizedBox*(height: 20),

*Text*(

              "Carregando...",

              style: *TextStyle*(

                fontSize: 16

              ),

            ),

          ],

        ),

      ),

    );

  }

}

inline\_information.dart

import 'package:flutter/material.dart';

class *InlineInformation* extends *StatelessWidget* {

  final *String* title;

  final *String* information;

  const *InlineInformation*({

*Key* key,

*this*.title = "",

    @required *this*.information

    }) : *super*(key: key);

  @override

*Widget* build(*BuildContext* context) {

    return *Wrap*(

      children: [

        title.isNotEmpty

        ? *Text*(

          title,

          style: *TextStyle*(

            fontWeight: *FontWeight*.bold,

            fontSize: 16,

          ),

        )

        : *Container*(),

*Text*(

          information,

          style: *TextStyle*(

            fontSize: 16,

          ),

        ),

      ],

    );

  }

}

separator.dart

import 'package:flutter/material.dart';

class *Separator* extends *StatelessWidget* {

  @override

*Widget* build(*BuildContext* context) {

    return *Container*(

      margin: *EdgeInsets*.symmetric(vertical: 10),

      child: *Row*(

        children: [

*Expanded*(child: *Divider*()),

*Container*(

            margin: *EdgeInsets*.only(left: 5, right: 5),

            width: 7,

            height: 7,

            decoration: *BoxDecoration*(

              shape: *BoxShape*.circle,

              color: *Colors*.black

            ),

          ),

*Container*(

            width: 7,

            height: 7,

            decoration: *BoxDecoration*(

              shape: *BoxShape*.circle,

              color: *Colors*.black

            ),

          ),

*Container*(

            margin: *EdgeInsets*.only(left: 5, right: 5),

            width: 7,

            height: 7,

            decoration: *BoxDecoration*(

              shape: *BoxShape*.circle,

              color: *Colors*.black

            ),

          ),

*Expanded*(child: *Divider*()),

        ],

      ),

    );

  }

}

main.dart

import 'package:flutter/material.dart';

import 'package:get/get.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/controllers/portfolio\_controller.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/home\_binding.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/models/portfolio.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/presentation/portfolio\_page.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/status\_operation/error\_page.dart';

import 'package:portfolio\_prova\_segundo\_bimestre/pages/status\_operation/loading\_page.dart';

void main() {

  runApp(*MyApp*());

}

class *MyApp* extends *StatelessWidget* {

  final *PortfolioController* \_portfolioController = *Get*.put(*PortfolioController*());

  @override

*Widget* build(*BuildContext* context) {

    return *GetMaterialApp*(

      debugShowCheckedModeBanner: false,

      initialBinding: *HomeBinding*(),

      title: 'Portfólio',

      theme: *ThemeData*(

        primarySwatch: *Colors*.blue,

        visualDensity: *VisualDensity*.adaptivePlatformDensity,

      ),

      home: \_initialNavigation(),

    );

  }

*FutureBuilder*<*Portfolio*> \_initialNavigation() {

    return *FutureBuilder*<*Portfolio*>(

    future: \_portfolioController.getPortfolio(),

      builder: (context, snapshot) {

        if(snapshot.hasError) return *ErrorPage*();

        if(snapshot.connectionState == *ConnectionState*.done && snapshot.data != null)

          return *PortfolioPage*(

            portfolio: snapshot.data,

          );

        return *LoadingPage*();

      },

    );

  }

}

## **Pubspec.yaml**

name: portfolio\_prova\_segundo\_bimestre

description: A new Flutter project.

# The following line prevents the package from being accidentally published to

# pub.dev using `pub publish`. This is preferred for private packages.

publish\_to: 'none' # Remove this line if you wish to publish to pub.dev

# The following defines the version and build number for your application.

# A version number is three numbers separated by dots, like 1.2.43

# followed by an optional build number separated by a +.

# Both the version and the builder number may be overridden in flutter

# build by specifying --build-name and --build-number, respectively.

# In Android, build-name is used as versionName while build-number used as versionCode.

# Read more about Android versioning at https://developer.android.com/studio/publish/versioning

# In iOS, build-name is used as CFBundleShortVersionString while build-number used as CFBundleVersion.

# Read more about iOS versioning at

# https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html

version: 1.0.0+1

environment:

  sdk: ">=2.7.0 <3.0.0"

dependencies:

  flutter:

    sdk: flutter

  # The following adds the Cupertino Icons font to your application.

  # Use with the CupertinoIcons class for iOS style icons.

  cupertino\_icons: ^1.0.0

  get: ^3.21.3

  font\_awesome\_flutter: ^8.10.0

  url\_launcher: ^5.7.10

  expandable: ^4.1.4

dev\_dependencies:

  flutter\_test:

    sdk: flutter

# For information on the generic Dart part of this file, see the

# following page: https://dart.dev/tools/pub/pubspec

# The following section is specific to Flutter.

flutter:

  # The following line ensures that the Material Icons font is

  # included with your application, so that you can use the icons in

  # the material Icons class.

  uses-material-design: true

  # To add assets to your application, add an assets section, like this:

  assets:

    - assets/fundo\_programacao.png

  # An image asset can refer to one or more resolution-specific "variants", see

  # https://flutter.dev/assets-and-images/#resolution-aware.

  # For details regarding adding assets from package dependencies, see

  # https://flutter.dev/assets-and-images/#from-packages

  # To add custom fonts to your application, add a fonts section here,

  # in this "flutter" section. Each entry in this list should have a

  # "family" key with the font family name, and a "fonts" key with a

  # list giving the asset and other descriptors for the font. For

  # example:

  # fonts:

  #   - family: Schyler

  #     fonts:

  #       - asset: fonts/Schyler-Regular.ttf

  #       - asset: fonts/Schyler-Italic.ttf

  #         style: italic

  #   - family: Trajan Pro

  #     fonts:

  #       - asset: fonts/TrajanPro.ttf

  #       - asset: fonts/TrajanPro\_Bold.ttf

  #         weight: 700

  #

  # For details regarding fonts from package dependencies,

  # see https://flutter.dev/custom-fonts/#from-packages