Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

Лабораторная работа №11

по дисциплине "Технологии разработки мобильных приложений"

ТЕМА РАБОТЫ:

Знакомство с Flutter

Выполнил:

студент гр. ПРИ-120

Парахин К.В.

Принял:

преподаватель кафедры ИСПИ

Рощина А.И.

Владимир 2023 г.

Цель работы:

Реализовать приложение с использованием фреймворка Flutter.

Выполнение работы

Реализация NavigationBar:

class Home extends StatefulWidget {  
 @override  
 \_HomeState createState() => \_HomeState();  
}  
  
class \_HomeState extends State<Home> {  
 int \_currentIndex = 0;  
   
 final List<Widget> \_children = [  
 PlaceholderWidget(color: Colors.*red*, text: 'Главная страница', index: 0),  
 PlaceholderWidget(color: Colors.*green*, text: 'Страница создания объявления', index: 1),  
 ];  
  
 void onTabTapped(int index) {  
 setState(() {  
 \_currentIndex = index;  
 });  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 title: Text('Поиск вакансий'),  
 ),  
 body: \_children[\_currentIndex],  
 bottomNavigationBar: BottomNavigationBar(  
 currentIndex: \_currentIndex,  
 onTap: onTabTapped,  
 items: [  
 BottomNavigationBarItem(  
 icon: Icon(Icons.*toc*),  
 label: 'Главная',  
 ),  
 BottomNavigationBarItem(  
 icon: Icon(Icons.*add*),  
 label: 'Создать объявление',  
 ),  
 ],  
 ),  
 );  
 }  
}

Реализация отображения списка и добавления новых элементов в него:

class AnnouncementPageScreenState extends State<AnnouncementPageScreen> {  
  
 final url = Uri.*parse*('http://172.20.10.3:8092/api\_announcements/announcements\_status\_ok');  
 final headers = {'Content-Type': 'application/json'};  
 bool isColor = false;  
  
 List<ModelAnnouncement> dataList = [];  
  
 Future<void> addAnnouncement() async {  
 final response = await http.get(Uri.*parse*('http://172.20.10.3:8092/api\_announcements/announcements\_status\_ok'));

if (response.statusCode == 200) {  
 final jsonData = json.decode(response.body);  
 setState(() {  
 dataList = List<ModelAnnouncement>.from(jsonData.map((data) => ModelAnnouncement.fromJson(data)));  
 });  
 }  
 }  
  
 @override  
 void initState() {  
 super.initState();  
 addAnnouncement();  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 home: Scaffold(  
 appBar: AppBar(  
 title: Text('Список объявлений'),  
 leading: IconButton(  
 icon: Icon(Icons.*arrow\_back*),  
 onPressed: () {  
 Navigator.*pushReplacement*(context, MaterialPageRoute(builder: (context) => UserPageScreen()),);  
 },  
 ),  
 ),  
 body: ListView.builder(  
 itemCount: dataList.length,  
 itemBuilder: (context, index) {  
 final data = dataList[index];  
 return Card(  
 color: isColor ? Colors.*red* : Colors.*teal*,  
 elevation: 15,  
 child: InkWell(  
 onTap: () {  
 setState(() {  
 isColor = !isColor;  
 });  
 },  
 child: Padding(  
 padding: EdgeInsets.all(25),  
 child: Column(  
 crossAxisAlignment: CrossAxisAlignment.start,  
 children: [  
 Text(  
 utf8.decode(data.name.codeUnits),  
 style: TextStyle(  
 color: Colors.*white*,  
 fontWeight: FontWeight.*bold*,  
 ),  
 ),  
 SizedBox(height: 8),  
 Text(  
 utf8.decode(data.description.codeUnits),  
 style: TextStyle(  
 color: Colors.*white*,  
 ),  
 ),  
 Text(  
 'Условия и требования:',  
 style: TextStyle(  
 color: Colors.*white*,  
 ),  
 ),  
 Text(  
 utf8.decode(data.conditions\_and\_requirements.codeUnits),  
 style: TextStyle(  
 color: Colors.*white*,  
 ),  
 ),  
 SizedBox(height: 8),  
 ElevatedButton(  
 child: Text('Отклик'),  
 onPressed: () {},  
 ),  
 ],  
 ),  
 ),  
 ),  
 );  
  
 },  
 ),  
 ),  
 );  
 }

Реализация смены цвета фона у элементов списка:

return Card(  
 color: isColor ? Colors.*red* : Colors.*teal*,  
 elevation: 15,  
 child: InkWell(  
 onTap: () {  
 setState(() {  
 isColor = !isColor;  
 });  
 },

Реализация смены изображения по клику на него:

import 'package:flutter/material.dart';  
  
void main() {  
 runApp(Home());  
}  
class Home extends StatefulWidget {  
 @override  
 \_HomeState createState() => \_HomeState();  
}  
  
class \_HomeState extends State<Home> {  
 String url = "https://source.unsplash.com/random/800x600";  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 home: Scaffold(  
 body: Center(  
 child: GestureDetector(  
 onTap: changeURL,  
 child: Image.network(url),  
 ),  
 ),  
 )  
 );  
 }  
 void changeURL() {  
 setState(() {  
 url = "https://source.unsplash.com/random/800x600/?" +  
 "q=${new DateTime.now().millisecondsSinceEpoch}";  
 });  
 }  
  
}

Реализация смены изображения автоматически раз в 3 секунды:

import 'package:flutter/material.dart';  
import 'dart:async';  
  
void main() {  
 runApp(Home());  
}  
class Home extends StatefulWidget {  
 @override  
 \_HomeState createState() => \_HomeState();  
}  
  
class \_HomeState extends State<Home> {  
 String url = "https://source.unsplash.com/random/800x600";  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 home: Scaffold(  
 body: Center(  
 child: GestureDetector(  
 onTap: changeURL,  
 child: Image.network(url),  
 ),  
 ),  
 )  
 );  
 }  
  
 void changeURL() {  
 Timer.periodic(Duration(seconds: 3), (timer) {  
 setState(() {  
 url = "https://source.unsplash.com/random/800x600/?" +  
 "q=${new DateTime.now().millisecondsSinceEpoch}";  
 });  
 });  
 }  
  
}

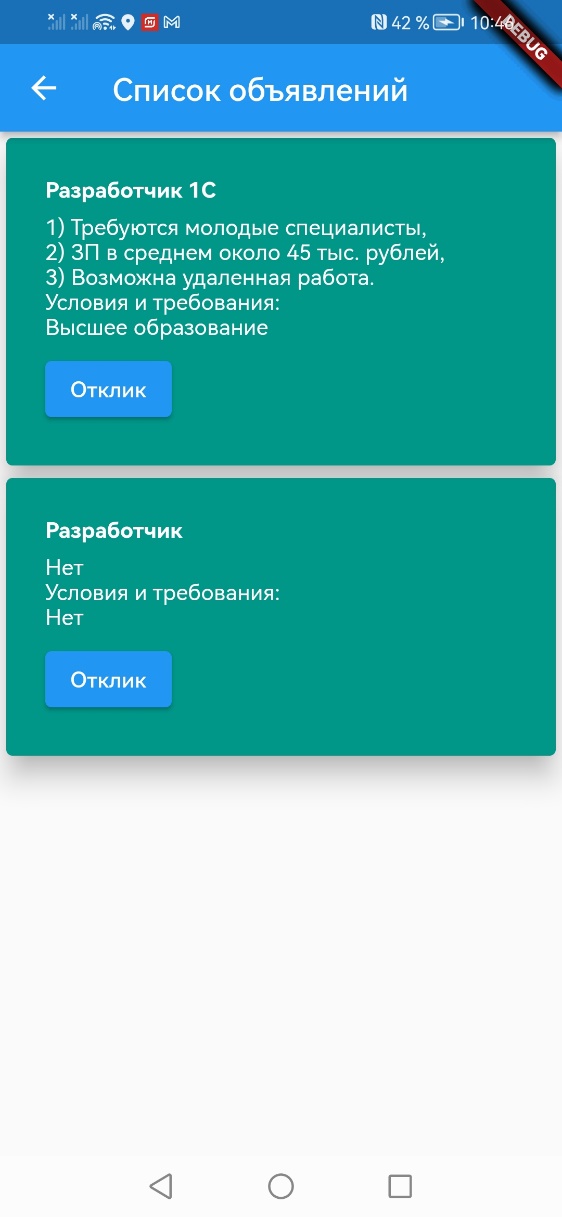
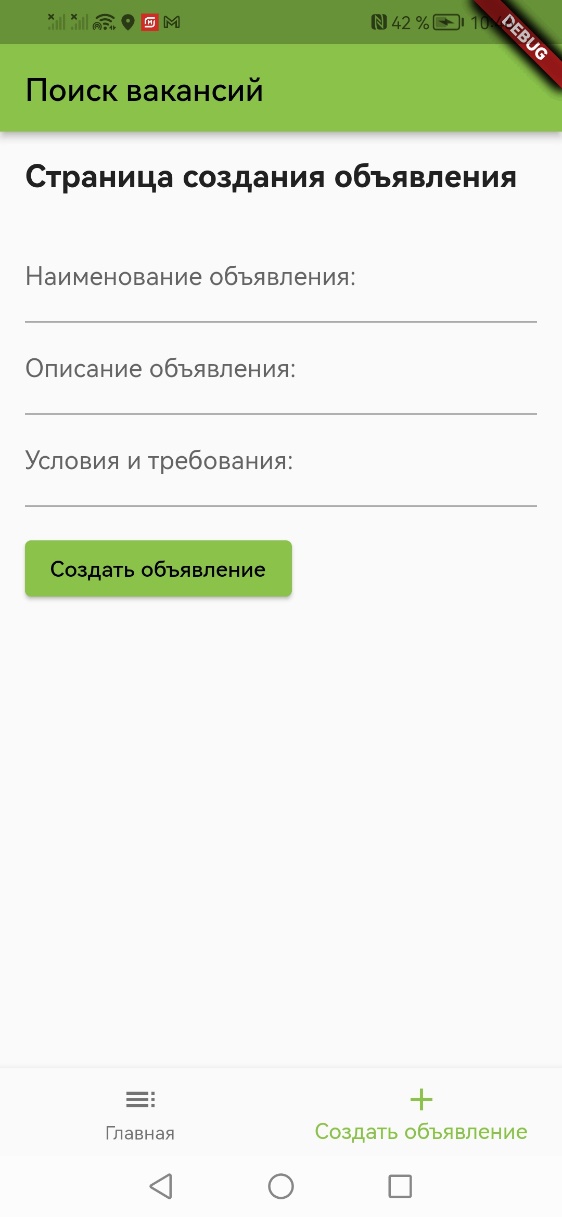


Рисунок 1. Скриншот демонстрации приложения



Рисунок 2. Скриншот демонстрации приложения

Вывод:

Реализовал приложение с использованием фреймворка flutter.