loginPage.dart

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'dart:convert'; // Importa il pacchetto per la gestione del JSON

import 'package:progetto/screens/profilePage.dart';

class LoginPage extends StatefulWidget {

@override

\_LoginPageState createState() => \_LoginPageState();

}

class \_LoginPageState extends State<LoginPage> {

final \_formKey = GlobalKey<FormState>();

String? \_accountName;

String? \_password;

Future<void> \_login() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

String? usersData = prefs.getString('users');

Map<String, dynamic> users = usersData != null ? json.decode(usersData) : {};

if (users.containsKey(\_accountName) && users[\_accountName]['Password'] == \_password) {

// Salva il nome dell'account nelle SharedPreferences

await prefs.setString('loggedInAccount', \_accountName!);

Navigator.pushReplacement(

context,

MaterialPageRoute(builder: (context) => ProfilePage(accountName: \_accountName!)),

);

} else {

ScaffoldMessenger.of(context).showSnackBar(SnackBar(

content: Text('Wrong Account name or password'),

));

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Login'),

),

body: Padding(

padding: EdgeInsets.all(16.0),

child: Form(

key: \_formKey,

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: <Widget>[

TextFormField(

decoration: InputDecoration(labelText: 'Account name'),

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter your account name';

}

return null;

},

onSaved: (value) {

\_accountName = value;

},

),

TextFormField(

decoration: InputDecoration(labelText: 'Password'),

obscureText: true,

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter your password';

}

return null;

},

onSaved: (value) {

\_password = value;

},

),

SizedBox(height: 20),

ElevatedButton(

onPressed: () {

if (\_formKey.currentState!.validate()) {

\_formKey.currentState!.save();

\_login();

}

},

child: Text('Login'),

),

],

),

),

),

);

}

}

plots.dart

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'dart:convert';

import 'package:provider/provider.dart';

import 'cigarette\_counter.dart';

import 'package:progetto/charts/plot\_creation.dart';

class Plots extends StatefulWidget {

final String accountName;

Plots({required this.accountName});

@override

\_PlotsState createState() => \_PlotsState();

}

class \_PlotsState extends State<Plots> {

DateTime? registrationDate;

List<NicotineLevel> data = [];

int \_cigarettesPerDay = 0;

bool isLoading = true;

@override

void initState() {

super.initState();

\_loadRegistrationData();

}

Future<void> \_loadRegistrationData() async {

try {

SharedPreferences prefs = await SharedPreferences.getInstance();

String? usersData = prefs.getString('users');

print("usersData: $usersData");

Map<String, dynamic> users = usersData != null ? json.decode(usersData) : {};

if (users.containsKey(widget.accountName)) {

var userProfile = users[widget.accountName];

String? dateStr = userProfile['registrationDate'];

\_cigarettesPerDay = userProfile['CigarettesPerDay'] ?? 0;

if (dateStr != null) {

registrationDate = DateTime.parse(dateStr);

} else {

registrationDate = DateTime.now();

}

\_generateChartData(users);

} else {

print("User not found: ${widget.accountName}");

}

} catch (e) {

print("Error loading registration data: $e");

} finally {

setState(() {

isLoading = false;

});

}

}

void \_generateChartData(Map<String, dynamic> users) {

if (registrationDate != null) {

DateTime startDate = registrationDate!;

data = [];

for (int i = 0; i < \_cigarettesPerDay \* 7; i++) {

DateTime currentDate = startDate.add(Duration(days: i));

String dateKey = "cigarettes\_${currentDate.year}${currentDate.month}${currentDate.day}";

double cigarettes = users[widget.accountName][dateKey]?.toDouble() ?? 0.0;

data.add(NicotineLevel(date: currentDate, level: cigarettes));

}

}

}

@override

Widget build(BuildContext context) {

final cigaretteCounter = Provider.of<CigaretteCounter>(context);

return Scaffold(

appBar: AppBar(title: Text('Plots')),

body: isLoading

? Center(child: CircularProgressIndicator())

: Column(

children: <Widget>[

// Grafico ridotto in altezza

Container(

height: MediaQuery.of(context).size.height \* 0.4, // Altezza del grafico ridotta

child: SingleChildScrollView(

scrollDirection: Axis.horizontal,

child: Container(

width: (data.length \* 60.0),

child: NicotineChart(

NicotineChart.createSampleData(data),

animate: true,

),

),

),

),

// Testo "Slide horizontally"

Padding(

padding: const EdgeInsets.symmetric(horizontal: 16.0, vertical: 8.0),

child: Align(

alignment: Alignment.centerLeft,

child: Text(

'Slide horizontally to view more data',

style: TextStyle(fontSize: 16),

),

),

),

// Sezione "Cigarettes smoked today"

Padding(

padding: const EdgeInsets.all(16.0),

child: Row(

mainAxisAlignment: MainAxisAlignment.start,

children: [

Text(

'Cigarettes smoked today: ${cigaretteCounter.cigarettesSmokedToday}/$\_cigarettesPerDay',

style: TextStyle(fontSize: 20),

),

],

),

),

],

),

);

}

}

profilePage.dart

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'dart:convert';

import 'package:provider/provider.dart';

import 'cigarette\_counter.dart';

import 'modify.dart';

import 'plots.dart';

class ProfilePage extends StatefulWidget {

final String accountName;

ProfilePage({required this.accountName});

@override

\_ProfilePageState createState() => \_ProfilePageState();

}

class \_ProfilePageState extends State<ProfilePage> {

String? \_cigaretteType;

double? \_nicotine;

@override

void initState() {

super.initState();

\_loadUserData();

\_loadCigarettesSmokedToday();

}

Future<void> \_loadUserData() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

String? accountName = prefs.getString('loggedInAccount');

if (accountName != null) {

String? usersData = prefs.getString('users');

Map<String, dynamic> users = usersData != null ? json.decode(usersData) : {};

if (users.containsKey(accountName)) {

setState(() {

\_cigaretteType = users[accountName]['CigaretteType'];

\_nicotine = users[accountName]['Nicotine'];

});

}

}

}

Future<void> \_loadCigarettesSmokedToday() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

String todayKey = \_getTodayKey();

int cigarettes = prefs.getInt(todayKey) ?? 0;

Provider.of<CigaretteCounter>(context, listen: false).setCigarettes(cigarettes);

}

String \_getTodayKey() {

DateTime now = DateTime.now();

return "cigarettes\_${now.year}${now.month}${now.day}";

}

Future<void> \_incrementCigaretteCount() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

String todayKey = \_getTodayKey();

int newCount = Provider.of<CigaretteCounter>(context, listen: false).cigarettesSmokedToday + 1;

setState(() {

prefs.setInt(todayKey, newCount);

});

Provider.of<CigaretteCounter>(context, listen: false).incrementCigarettes();

}

@override

Widget build(BuildContext context) {

final cigaretteProvider = Provider.of<CigaretteCounter>(context);

return Scaffold(

appBar: AppBar(title: Text('Profile')),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: <Widget>[

Text('Account Name: ${widget.accountName}'),

Text('Cigarette Type: $\_cigaretteType'),

Text('Nicotine: $\_nicotine'),

SizedBox(height: 20),

ElevatedButton(

onPressed: () {

Navigator.push(

context,

MaterialPageRoute(

builder: (context) => ModifyPage(

accountName: widget.accountName,

cigaretteType: \_cigaretteType!,

nicotine: \_nicotine!,

),

),

).then((\_) {

\_loadUserData();

});

},

child: Text('Modify Profile'),

),

Row(

mainAxisAlignment: MainAxisAlignment.center,

children: <Widget>[

ElevatedButton(

onPressed: () {

\_incrementCigaretteCount();

},

child: Text('Add a Cigarette'),

),

SizedBox(width: 20),

Text(

'${cigaretteProvider.cigarettesSmokedToday}',

style: TextStyle(fontSize: 24),

),

],

),

SizedBox(height: 20),

ElevatedButton(

onPressed: () {

Navigator.push(

context,

MaterialPageRoute(

builder: (context) => Plots(accountName: widget.accountName),

),

);

},

child: Text('View Progress'),

),

],

),

),

);

}

}

registration.dart

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'dart:convert';

class RegistrationPage extends StatefulWidget {

@override

\_RegistrationPageState createState() => \_RegistrationPageState();

}

class \_RegistrationPageState extends State<RegistrationPage> {

final \_formKey = GlobalKey<FormState>();

String? \_firstName;

String? \_lastName;

String? \_accountName;

String? \_password;

int? \_cigPerDay;

String? \_cigType;

double? \_nicotine;

Future<void> \_saveData() async {

if (\_formKey.currentState!.validate()) {

\_formKey.currentState!.save();

SharedPreferences prefs = await SharedPreferences.getInstance();

// Recupera i dati degli utenti esistenti

String? usersData = prefs.getString('users');

Map<String, dynamic> users = usersData != null ? json.decode(usersData) : {};

// Aggiungi i dati del nuovo utente

users[\_accountName!] = {

'FirstName': \_firstName,

'LastName': \_lastName,

'Password': \_password,

'CigarettesPerDay': \_cigPerDay,

'CigaretteType': \_cigType,

'Nicotine': \_nicotine,

};

// Salva i dati aggiornati

await prefs.setString('users', json.encode(users));

print('User data saved: ${users[\_accountName!]}'); // Debug

Navigator.pop(context);

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Registration'),

),

body: Padding(

padding: EdgeInsets.all(16.0),

child: Form(

key: \_formKey,

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: <Widget>[

TextFormField(

decoration: InputDecoration(labelText: 'First name'),

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter your name';

}

return null;

},

onSaved: (value) {

\_firstName = value;

},

),

TextFormField(

decoration: InputDecoration(labelText: 'Last name'),

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter your last name';

}

return null;

},

onSaved: (value) {

\_lastName = value;

},

),

TextFormField(

decoration: InputDecoration(labelText: 'Account name'),

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter your account name';

}

return null;

},

onSaved: (value) {

\_accountName = value;

},

),

TextFormField(

decoration: InputDecoration(labelText: 'Password'),

obscureText: true,

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter your password';

}

return null;

},

onSaved: (value) {

\_password = value;

},

),

TextFormField(

decoration: InputDecoration(labelText: 'Number of cigarettes per day'),

keyboardType: TextInputType.number,

validator: (value) {

if (value == null || value.isEmpty) {

return 'Enter the number of cigarettes per day';

}

return null;

},

onSaved: (value) {

\_cigPerDay = int.parse(value!);

},

),

DropdownButtonFormField<String>(

decoration: InputDecoration(labelText: 'Cigarette type'),

items: <String>['Light', 'Regular', 'Heavy'].map((String value) {

return DropdownMenuItem<String>(

value: value,

child: Text(value),

);

}).toList(),

onChanged: (newValue) {

setState(() {

\_cigType = newValue;

if (newValue == 'Light') {

\_nicotine = 0.5;

} else if (newValue == 'Regular') {

\_nicotine = 1.0;

} else if (newValue == 'Heavy') {

\_nicotine = 1.5;

}

});

},

),

SizedBox(height: 20),

ElevatedButton(

onPressed: \_saveData,

child: Text('Sign up'),

),

],

),

),

),

);

}

}

add\_cigarette.dart

import 'package:flutter/material.dart';

class AddCigarettePage extends StatelessWidget {

final String accountName;

AddCigarettePage({required this.accountName});

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Add a Cigarette'),

),

body: Padding(

padding: const EdgeInsets.all(16.0),

child: Column(

children: [

Align(

alignment: Alignment.centerLeft,

child: Text(

'Cigarette smoked today...',

style: TextStyle(fontSize: 18),

),

),

SizedBox(height: 20),

// Altri widget per l'aggiunta della sigaretta

],

),

),

);

}

}

modify.dart

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'dart:convert';

class ModifyPage extends StatefulWidget {

final String accountName;

final String cigaretteType;

final double nicotine;

ModifyPage({

required this.accountName,

required this.cigaretteType,

required this.nicotine,

});

@override

\_ModifyPageState createState() => \_ModifyPageState();

}

class \_ModifyPageState extends State<ModifyPage> {

String? \_selectedCigaretteType;

double? \_nicotine;

@override

void initState() {

super.initState();

\_selectedCigaretteType = widget.cigaretteType;

\_nicotine = widget.nicotine;

}

Future<void> \_saveChanges() async {

SharedPreferences prefs = await SharedPreferences.getInstance();

String? usersData = prefs.getString('users');

Map<String, dynamic> users = usersData != null ? json.decode(usersData) : {};

if (users.containsKey(widget.accountName)) {

users[widget.accountName]['CigaretteType'] = \_selectedCigaretteType;

users[widget.accountName]['Nicotine'] = \_nicotine;

await prefs.setString('users', json.encode(users));

}

Navigator.pop(context);

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Modify Cigarette'),

),

body: Padding(

padding: EdgeInsets.all(16.0),

child: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

crossAxisAlignment: CrossAxisAlignment.center,

children: <Widget>[

DropdownButton<String>(

value: \_selectedCigaretteType,

onChanged: (newValue) {

setState(() {

\_selectedCigaretteType = newValue;

if (newValue == 'Light') {

\_nicotine = 0.5;

} else if (newValue == 'Regular') {

\_nicotine = 1.0;

} else if (newValue == 'Heavy') {

\_nicotine = 1.5;

}

});

},

items: <String>['Light', 'Regular', 'Heavy'].map((String value) {

return DropdownMenuItem<String>(

value: value,

child: Text(value),

);

}).toList(),

),

SizedBox(height: 20),

Text('Selected Nicotine: $\_nicotine mg'),

SizedBox(height: 20),

ElevatedButton(

onPressed: \_saveChanges,

child: Text('Save Changes'),

),

],

),

),

),

);

}

}

homePage.dart

import 'package:flutter/material.dart';

import 'loginPage.dart';

import 'registration.dart';

class HomePage extends StatelessWidget {

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Home Page'),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: <Widget>[

ElevatedButton(

child: Text('Login'),

onPressed: () {

Navigator.push(

context,

MaterialPageRoute(builder: (context) => LoginPage()),

);

},

),

ElevatedButton(

child: Text('Register'),

onPressed: () {

Navigator.push(

context,

MaterialPageRoute(builder: (context) => RegistrationPage()),

);

},

),

],

),

),

);

}

}

cigarette\_counter.dart

import 'package:flutter/foundation.dart';

class CigaretteCounter with ChangeNotifier {

int \_cigarettesSmokedToday = 0;

int get cigarettesSmokedToday => \_cigarettesSmokedToday;

void incrementCigarettes() {

\_cigarettesSmokedToday++;

notifyListeners();

}

void setCigarettes(int count) {

\_cigarettesSmokedToday = count;

notifyListeners();

}

}

main.dart

import 'package:flutter/material.dart';

import 'package:provider/provider.dart';

import 'package:progetto/screens/cigarette\_counter.dart';

import 'package:progetto/screens/homepage.dart'; // Cambiato da ProfilePage a HomePage

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MultiProvider(

providers: [

ChangeNotifierProvider(create: (\_) => CigaretteCounter()),

],

child: MaterialApp(

title: 'Flutter Demo',

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: HomePage(), // Imposta HomePage come pagina iniziale

),

);

}

}