SSM INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Approved by AICTE, New Delhi Affiliated to Anna University, Chennai,

Accredited by NAAC)

Dindigul – Palani Highway, Dindigul – 624 002.

VALUE ADDED COURSE PROJECT REPORT

OPEN SOURCE APP DEVELOPMENT USING FLUTTER

(Department of Computer Science and Engineering)

Submitted By

Umar Farook J

CSE III Year

922121104055

SSM Student Management

Project Overview:

This Flutter mini project, titled "SSMIET," serves as a simple student management application. It allows users to input student details, such as name, register number, department, and college, and stores the data in a SQLite database. The application uses a clean and user-friendly interface for data entry.

Project Structure:

1. main.dart:

- Entry point for the Flutter application.
- Configures the MaterialApp and sets the initial screen to **`PostDataPage`.**

2. postdatapage.dart:

- Defines the **`PostDataPage`** StatefulWidget for data entry.
- Utilizes **`UserInputBar`** custom widget for consistent text input fields.
- Interacts with the **`CollegeDatabase`** class to insert student data into the SQLite database.
- Features a clear button to reset input fields.

3. collegedatabase.dart:

- Manages the SQLite database operations (Create, Read, Update, Delete - CRUD).
- Defines the **`CollegeDatabase`** class as a **GetxController** for state management.
- Handles database initialization, table creation, and CRUD operations.

4. theme.dart:

- Contains color constants used for theming the application.

5. userinputbar.dart:

- Custom widget for text input fields with consistent styling.

Coding:

main.dart:

```
import 'package:flutter/material.dart';
import 'package:ssm/pages/postdatapage.dart';
void main() {
  runApp(const MyApp());
}
class MyApp extends StatelessWidget {
  const MyApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'SSMIET',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ),
      debugShowCheckedModeBanner: false,
      home: const PostDataPage(),
    );
  }
}
```

postdatapage.dart:

```
import 'package:flutter/material.dart';
import 'package:flutter_svg/flutter_svg.dart';
import 'package:gap/gap.dart';
import 'package:get/get.dart';
import 'package:google_fonts/google_fonts.dart';
import 'package:ssm/components/userinputbar.dart';
```

```
import 'package:ssm/constant/theme.dart';
import 'package:ssm/databases/collegedatabase.dart';
class PostDataPage extends StatefulWidget {
  const PostDataPage({super.key});
  @override
  State<PostDataPage> createState() => _PostDataPageState();
}
class _PostDataPageState extends State<PostDataPage> {
  TextEditingController nameController = TextEditingController();
  TextEditingController regController = TextEditingController();
  TextEditingController deptController = TextEditingController();
  TextEditingController collegeController = TextEditingController();
  CollegeDatabase collegeDatabase = Get.put(CollegeDatabase());
  @override
  Widget build(BuildContext context) {
    final screenSize = MediaQuery.sizeOf(context);
    void _showSnackBar(String message) {
      final snackBar = SnackBar(
        content: Text(message),
        duration: Duration(seconds: 2),
      );
      ScaffoldMessenger.of(context).showSnackBar(snackBar);
    }
    void _clearTextFields() {
      nameController.clear();
      regController.clear();
      deptController.clear();
      collegeController.clear();
    }
    return Scaffold(
      appBar: AppBar(
        elevation: 0,
```

```
  backgroundColor: sYellow,
  centerTitle: false,
  leading: Padding(
    padding: const EdgeInsets.all(8.0),
    child: SvgPicture.network(
      'https://www.svgrepo.com/show/196244/list-tick.svg',
      height: 14,
      width: 14,
    ),
  ),
  title: Text(
    "COLLEGE MANAGEMENT",
    style: GoogleFonts.poppins(
      fontSize: 16,
      fontWeight: FontWeight.w600,
      color: sOrange,
    ),
  ),
),
body: Padding(
  padding: const EdgeInsets.only(
    left: 10,
    right: 10,
    top: 20,
  ),
  child: Column(
    children: [
      Row(
        children: [
          Padding(
            padding: const EdgeInsets.only(left: 20),
            child: UserInputBar(
```

```
          txt: 'Enter Name',
          type: TextInputType.name,
          ht: screenSize.height * 0.04,
          wt: screenSize.width * 0.4,
          controller: nameController,
        ),
      ),
      const Gap(30),
      UserInputBar(
        txt: 'Enter Register Number',
        type: TextInputType.number,
        ht: screenSize.height * 0.04,
        wt: screenSize.width * 0.4,
        controller: regController,
      )
    ],
  ),
  const Gap(20),
  Padding(
    padding: const EdgeInsets.only(left: 10),
    child: UserInputBar(
      txt: 'Enter Department',
      type: TextInputType.text,
      ht: screenSize.height * 0.04,
      wt: screenSize.width * 0.8,
      controller: deptController,
    ),
  ),
  const Gap(20),
  Padding(
    padding: const EdgeInsets.only(left: 10),
    child: UserInputBar(
```

```
    txt: 'Enter College',
    type: TextInputType.text,
    ht: screenSize.height * 0.04,
    wt: screenSize.width * 0.8,
    controller: collegeController,
  ),
),
const Gap(20),
InkWell(
  onTap: () async {
    Map<String, dynamic> userData = {
      'name': nameController.text,
      'reg': int.parse(regController.text).toString(),
      'department': deptController.text,
      'college': collegeController.text,
    };
    int dataPushed = await collegeDatabase.InsertData(userData);
    if (dataPushed > 0) {
      _showSnackBar("Data submitted successfully!");
    } else {
      _showSnackBar("Failed to submit data. Please try again.");
    }
  },
  child: Column(
    children: [
      Row(
        children: [
          Padding(
            padding: const EdgeInsets.only(left: 40),
            child: Container(
              height: screenSize.height * 0.04,
              width: screenSize.width * 0.35,
```

```
    decoration: BoxDecoration(
      color: sOrange,
      borderRadius: BorderRadius.circular(20),
    ),
    alignment: Alignment.center,
    child: Text(
      "submit",
      style: GoogleFonts.poppins(
        fontSize: 12,
        fontWeight: FontWeight.w600,
        color: Colors.white,
      ),
    ),
  ),
),
Gap(20),
InkWell(
  onTap: _clearTextFields,
  child: Padding(
    padding: const EdgeInsets.only(left: 10),
    child: Container(
      height: screenSize.height * 0.04,
      width: screenSize.width * 0.35,
      decoration: BoxDecoration(
        color: sOrange,
        borderRadius: BorderRadius.circular(20),
      ),
      alignment: Alignment.center,
      child: Text(
        "Clear",
        style: GoogleFonts.poppins(
          fontSize: 12,
```

```
                    fontWeight: FontWeight.w600,
                    color: Colors.white,
                  ),
                ),
              ),
            ),
          ),
        ],
      ),
    ),
  ),
  Row(
    children: const [],
  ),
 ],
),
),
);
}
}
```

collegedatabase.dart:

```
import 'dart:io';
import 'package:get/get.dart';
import 'package:path/path.dart';
import 'package:path_provider/path_provider.dart';
import 'package:sqflite/sqflite.dart';
class CollegeDatabase extends GetxController {
  static final CollegeDatabase instance = CollegeDatabase();
  static Database?
```

```
    _database;
CollegeDatabase();
Future<Database> get database async {
  if (_database != null) return _database!;
  _database = await _initDatabase();
  return _database!;
}
Future<Database> _initDatabase() async {
  Directory application_directory = await getApplicationDocumentsDirectory();
  print("The Database is Here:" + application_directory.path);
  final String path = join(await getDatabasesPath(), 'collegedata.db');
  return await openDatabase(
    path,
    version: 1,
    onCreate: _CreateTable,
  );
}
Future<void> _CreateTable(Database db, int version) async {
  return await db.execute('''
  CREATE TABLE CollegeTable (
    id INTEGER PRIMARY KEY,
    name TEXT,
    reg TEXT,
    department TEXT,
    college TEXT
  )
  ''');
}
Future<int> InsertData(Map<String, dynamic> data) async {
  Database db = await instance.database;
  return await db.insert('CollegeTable', data);
}
```

```
  Future<List<Map<String, dynamic>>> GetAllData() async {
    Database db = await instance.database;
    return await db.query('CollegeTable');
  }
  Future<int> UpdateData(Map<String, dynamic> data) async {
    Database db = await instance.database;
    int id = data['id'];
    return await db.update('CollegeTable', data,
        where: 'id=?', whereArgs: [id]); // ? for user search
  }
  Future<int> DeleteData(int id) async {
    Database db = await instance.database;
    return await db.delete('CollegeTable',
        where: 'id=?', whereArgs: [id]); // ? for user search
  }
}
```

theme.dart:

```
import 'package:flutter/material.dart';
const sOrange = Color(0xFFFF5A33);
const sYellow = Color(0xFFFFEC5C);
const sLightgreen = Color(0xFFB4CF66);
const sMidgreen = Color(0xFF44803F);
const sThickgreen = Color(0xFF146152);
```

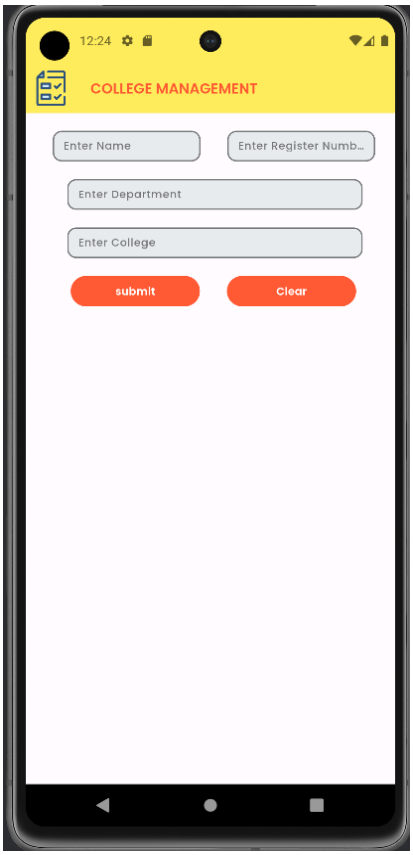
userinputbar.dart:

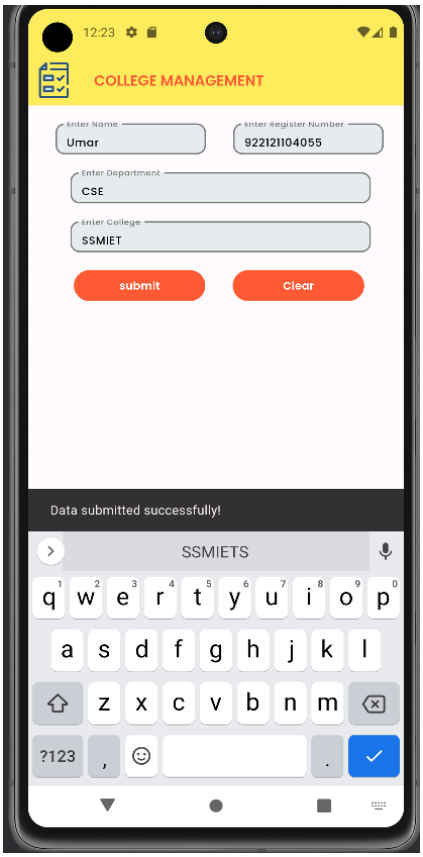
```
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';
import 'package:ssm/constant/theme.dart';
class UserInputBar extends StatelessWidget {
```

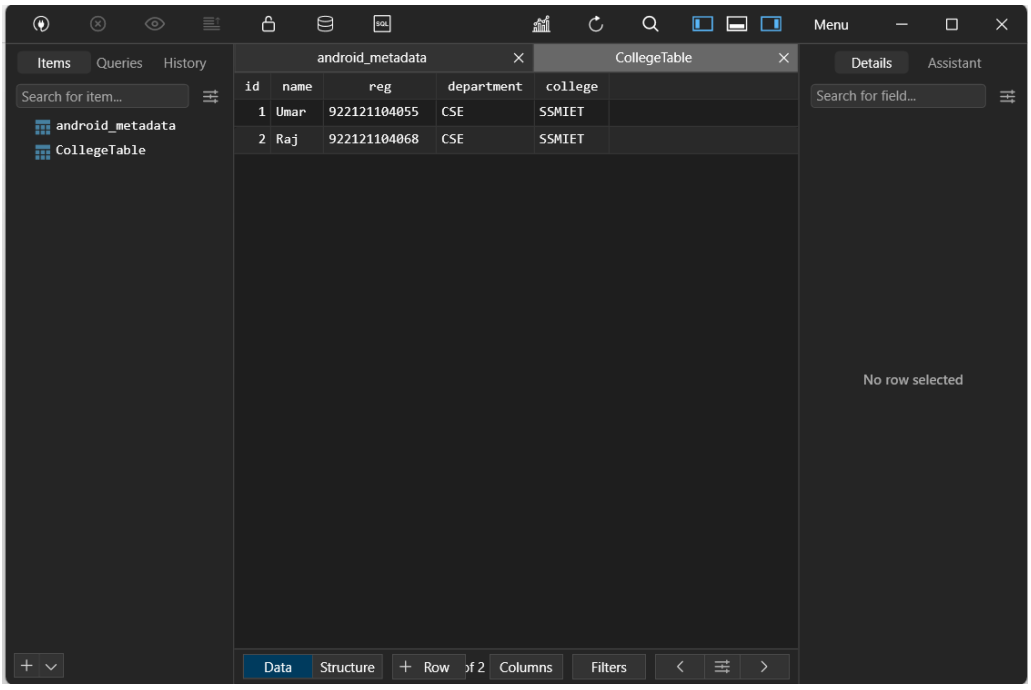
```
const UserInputBar({
  super.key,
  required this.txt,
  required this.type,
  required this.ht,
  required this.wt,
  required this.controller,
});
final String txt;
final TextInputType type;
final double ht;
final double wt;
final TextEditingController controller;
@override
Widget build(BuildContext context) {
  return Container(
    height: ht,
    width: wt,
    decoration: BoxDecoration(
      color: sThickgreen.withOpacity(0.1),
      borderRadius: BorderRadius.circular(10),
    ),
    child: TextField(
      controller: controller,
      keyboardType: type,
      style: GoogleFonts.poppins(
        fontSize: 12,
        fontWeight: FontWeight.w500,
        color: Colors.black,
      ),
      decoration: InputDecoration(
        labelText: txt,
```

```
          labelStyle: GoogleFonts.poppins(
            fontSize: 12,
            fontWeight: FontWeight.w500,
            color: Colors.black.withOpacity(0.5),
          ),
          enabledBorder: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10),
            borderSide: BorderSide(
              color: Colors.black.withOpacity(0.5),
              width: 1.5,
            ),
          ),
          focusedBorder: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10),
            borderSide: BorderSide(
              color: sMidgreen.withOpacity(0.9),
              width: 1.5,
            ),
          ),
        ),
      ),
    );
  }
}
```

Output:


12:24
COLLEGE MANAGEMENT
Enter Name
Enter Register Numb...
Enter Department
Enter College
submit
Clear


12:23
COLLEGE MANAGEMENT
Enter Name
Umar
Enter Register Number
922121104055
Enter Department
CSE
Enter College
SSMIET
submit
Clear
Data submitted successfully!
SSMIETS


Menu
Items
Queries
History
Search for item...
android_metadata
CollegeTable
android_metadata
CollegeTable
id
name
reg
department
college
1
Umar
922121104055
CSE
SSMIET
2
Raj
922121104068
CSE
SSMIET
Details
Assistant
Search for field...
No row selected
Data
Structure
Row
of 2
Columns
Filters

Running the Project

Prerequisites

- Ensure Flutter and Dart SDK are installed on your system.

- A code editor like Visual Studio Code or Android Studio.

- [TablePlus](https://tableplus.com/) app for viewing the SQLite database.

Steps:

1. Install Dependencies:

```
 dependencies:
flutter:
  sdk: flutter
gap: ^4.0.0
get: ^4.3.8
google_fonts: ^3.2.0
path: ^1.16.0
sqflite: ^2.0.0
flutter_svg: ^0.23.0
```

2. Run the Application:

Open Terminal and type the following

```
 flutter run
```

3. Viewing SQLite Database:

- Open **TablePlus** and connect to the SQLite database located in the app's documents directory.

- Path: `<Your-App-Directory>/collegedata.db`

Conclusion:

This mini project provides a foundation for a student management system, demonstrating the integration of Flutter with SQLite for data storage. Users can contribute to expanding features, and the code structure promotes maintainability and scalability.