**FINAL PRACTICAL EXAM**

**Name-Hrishikesh Tiwari Subject Code:** 20CAB755

**UID: 20MCA1296 Section/Group: 2/A**

**Semester: 4th Date of Performance: 19/05/22**

##### Aim/Overview of the practical: Create a flutter application to create the NOTICE BOARD APP. show all necessary functionality require to notify users about the important events.

##### OUTPUT /SCREENSHOT OF THE APPLICATION:

##### 

##### 

##### GITHUBLINK: -

##### 

**Task to be done:**

1. **Code for experiment/practical:**

**main.dart**

import 'package:myprogram4/login/SplashScreen.dart';

import 'package:firebase\_core/firebase\_core.dart';

import 'package:flutter/material.dart';

import 'firebase\_options.dart';

Future<void> main() async {

  WidgetsFlutterBinding.ensureInitialized();

  await Firebase.initializeApp(

    options: DefaultFirebaseOptions.currentPlatform,

  );

  runApp(const MyApp());

}

class MyApp extends StatelessWidget {

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

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      debugShowCheckedModeBanner: false,

      title: 'Flutter Demo',

      theme: ThemeData(

        primarySwatch: Colors.blue,

      ),

      home: const SplashScreen(),

    );

  }

}

**SplashScreen.dart**

lass \_SplashScreenState extends State<SplashScreen> {

  @override

  void initState() {

    super.initState();

    Timer(

      const Duration(seconds: 3),

          () => Navigator.pushReplacement(context, MaterialPageRoute(

        builder: (context) => const MyHomePage(),

      ),

      ),

    );

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      body: Center(

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          crossAxisAlignment: CrossAxisAlignment.center,

          children: [

            logo(160, 160),

            const SizedBox(

              height: 25,

            ),

            richText(30),

          ],

        ),

      ),

    );

  }

  Widget logo(double height\_, double width\_) {

    return SvgPicture.asset(

      'assets/logo.svg',

      height: height\_,

      width: width\_,

    );

  }

  Widget richText(double fontSize) {

    return Text.rich(

      TextSpan(

        style: GoogleFonts.inter(

          fontSize: fontSize,

          color: const Color(0xFF21899C),

          letterSpacing: 3,

          height: 1.03,

        ),

        children: const [

          TextSpan(

            text: 'UIC',

            style: TextStyle(

              fontWeight: FontWeight.w800,

            ),

          ),

          TextSpan(

            text: 'NOTIFIER \n HELLO ',

            style: TextStyle(

              color: Color(0xFFFE9879),

              fontWeight: FontWeight.w800,

            ),

          ),

          TextSpan(

            text: 'KIT',

            style: TextStyle(

              fontWeight: FontWeight.w800,

            ),

          ),

        ],

      ),

      textAlign: TextAlign.center,

    );

  }

}

**MyHomePage.dart**

import 'package:flutter/material.dart';

import 'package:cloud\_firestore/cloud\_firestore.dart';

import 'package:intl/intl.dart';

class MyHomePage extends StatefulWidget {

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

  @override

  State<MyHomePage> createState() => \_MyHomePageState();

}

class \_MyHomePageState extends State<MyHomePage> {

  var firestoredb =

      FirebaseFirestore.instance.collection("MyData").snapshots(); // STEP 1

  late TextEditingController nameinput = new TextEditingController();

  late TextEditingController titleinput = new TextEditingController();

  late TextEditingController descinput = new TextEditingController();

  @override

  void initState() {

    super.initState();

    nameinput = new TextEditingController();

    titleinput = new TextEditingController();

    descinput = new TextEditingController();

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

        appBar: AppBar(

          title: Text("BORAD DATA"),

        ),

        floatingActionButton: FloatingActionButton(

          onPressed: () {

            showDialogox(context);

          },

          child: Icon(Icons.edit),

        ),

        body: StreamBuilder(

          stream: firestoredb,

          builder: (context, AsyncSnapshot snapshot) {

            if (!(snapshot.hasData)) {

              return CircularProgressIndicator();

            } else {

              return ListView.builder(

                  itemCount: snapshot.data?.docs.length,

                  itemBuilder: (context, int index) {

                    return CustomCard(snapshot: snapshot.data, index: index);

                  });

            }

          },

        ));

  }

showDialogox(BuildContext context) async {

    await showDialog(

      context: context,

      builder: (context) => AlertDialog(

        contentPadding: EdgeInsets.all(12),

        content: Column(

          children: [

            Text("ENTER THE REQUIRED FEILDS"),

            Expanded(

                child: TextField(

              autofocus: true,

              autocorrect: true,

              decoration: InputDecoration(labelText: "Your Name"),

              controller: nameinput, // variable

            )),

            Expanded(

                child: TextField(

              autofocus: true,

              autocorrect: true,

              decoration: InputDecoration(labelText: "Title"),

              controller: titleinput,

            )),

            Expanded(

                child: TextField(

              autofocus: true,

              autocorrect: true,

              decoration: InputDecoration(labelText: "Description: "),

              controller: descinput,

            ))

          ],

        ),

        actions: [

          ElevatedButton(

              onPressed: () {

                nameinput.clear();

                titleinput.clear();

                descinput.clear();

                Navigator.pop(context);

              },

              child: Text("Cancel")),

          ElevatedButton(

              onPressed: () {

                if (nameinput.text.isNotEmpty &&

                    titleinput.text.isNotEmpty &&

                    descinput.text.isNotEmpty) {

                  FirebaseFirestore.instance.collection("MyData").add({

                    "name": nameinput.text,

                    "Description": descinput.text,

                    "Title": titleinput.text,

                    "timestamp": new DateTime.now()

                  }).then((response) {

                    print(response.id);

                    var docid = response.id;

                    Navigator.pop(context);

                    nameinput.clear();

                    titleinput.clear();

                    descinput.clear();

                  }).catchError((error) => print(error));

                }

              },

              child: Text("Save"))

        ],

      ),

    );

  }

}

class CustomCard extends StatelessWidget {

  final int index;

  final QuerySnapshot snapshot;

  const CustomCard({Key? key, required this.snapshot, required this.index})

      : super(key: key);

  @override

  Widget build(BuildContext context) {

    var docId = snapshot.docs[index].id;

    var td = DateTime.fromMillisecondsSinceEpoch(

        snapshot.docs[index]["timestamp"].seconds \* 1000);

    var dateformat = new DateFormat("EEEE,MMM,d,y").format(td);

    late TextEditingController nameinput =

        new TextEditingController(text: snapshot.docs[index]["name"]);

    late TextEditingController titleinput =

        new TextEditingController(text: snapshot.docs[index]["Title"]);

    late TextEditingController descinput =

        new TextEditingController(text: snapshot.docs[index]["Description"]);

    return Container(

      height: 170,

      child: Card(

        elevation: 10,

        child: Column(

          children: [

            ListTile(

              title: Text(snapshot.docs[index]["Title"]),

              subtitle: Text(snapshot.docs[index]["Description"]),

              leading: CircleAvatar(

                radius: 35,

                child: Text(

                    snapshot.docs[index]["Title"].toString()[0].toUpperCase()),

              ),

            ),

            Padding(

              padding: const EdgeInsets.all(8.0),

              child: Row(

                mainAxisAlignment: MainAxisAlignment.end,

                children: [

                  Text("BY: ${snapshot.docs[index]["name"]}  "),

                  Text(dateformat),

                ],

              ),

            ),

            Row(

              mainAxisAlignment: MainAxisAlignment.start,

              children: [

                IconButton(

                    onPressed: () async {

                      await showDialog(

                        context: context,

                        builder: (context) => AlertDialog(

                          contentPadding: EdgeInsets.all(10),

                          content: Column(

                            children: [

                              Text("UPADTE THE CURRENT RECORD"),

                              Expanded(

                                  child: TextField(

                                autofocus: true,

                                autocorrect: true,

                                decoration:

                                    InputDecoration(labelText: "Your Name"),

                                controller: nameinput, // variable

                              )),

                              Expanded(

                                  child: TextField(

                                autofocus: true,

                                autocorrect: true,

                                decoration: InputDecoration(labelText: "Title"),

                                controller: titleinput,

                              )),

                              Expanded(

                                  child: TextField(

                                autofocus: true,

                                autocorrect: true,

                                decoration:

                                    InputDecoration(labelText: "Description: "),

                                controller: descinput,

                              ))

                            ],

                          ),

                          actions: [

                            ElevatedButton(

                                onPressed: () {

                                  nameinput.clear();

                                  titleinput.clear();

                                  descinput.clear();

                                  Navigator.pop(context);

                                },

                                child: Text("Cancel")),

                            ElevatedButton(

                                onPressed: () {

                                  if (nameinput.text.isNotEmpty &&

                                      titleinput.text.isNotEmpty &&

                                      descinput.text.isNotEmpty) {

                                    FirebaseFirestore.instance

                                        .collection("MyData")

                                        .doc(docId)

                                        .update({

                                      "name": nameinput.text,

                                      "Description": descinput.text,

                                      "Title": titleinput.text,

                                      "timestamp": new DateTime.now()

                                    }).then((response) {

                                      Navigator.pop(context);

                                      nameinput.clear();

                                      titleinput.clear();

                                      descinput.clear();

                                    }).catchError((error) => print(error));

                                  }

                                },

                                child: Text("UPDATE"))

                          ],

                        ),

                      );

                    },

                    icon: Icon(

                      Icons.edit\_location\_outlined,

                      size: 20,

                    )),

                IconButton(

                    onPressed: () async {

                      var fbOb =

                          FirebaseFirestore.instance.collection("MyData");

                      await fbOb.doc(docId).delete();

                      print("ID : $docId");

                    },

                    icon: Icon(

                      Icons.delete\_outline\_outlined,

                      size: 20,

                    ))

              ],

            )

          ],

        ),

      ),

    );

  }

}

**Evaluation Grid:**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. | Demonstration and Performance |  | 5 |
| 2. | Worksheet |  | 10 |
| 3. | Post Lab Quiz |  | 5 |