



WEEKLY REPORT

In this report students require to mention the work has been done during the week and planning for the next week. (Please fill individual report for each week)

Student ID: 19DCS035	Student Name: Gami Bhavin Harish
From Date: 13/06/2022	To Date: 17/06/2022
Semester: 6th	Internship ID:

Work done last week:

1. Making a firestore database and performing CRUD operation with the user's data.

Code:

firestore.dart

```
import 'package:anonymus_app/helper%20class/constants.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

import '../Screens/SignIn.dart';

class Database {
  Create(String FullName, String address, String DOB, String email,
    context) async {
    await FirebaseFirestore.instance.collection('UserData').doc().set({
      'Full name': FullName,
      'address': address,
      'Email': email,
      'DOB': DOB,
    }).then((value) {
      Navigator.push(
        context, MaterialPageRoute(builder: (context) => SignInScreen()));
    });
    print('Data created');
  }
}
```

```

Update() async {
  await FirebaseFirestore.instance
    .collection('UserData')
    .doc('T0G3GBR5Wp00dgXyalQs')
    .update({'Full name': 'bhavingami'});
}

Delete() async {
  await FirebaseFirestore.instance
    .collection('UserData')
    .doc('5PMdzzZ47MXLg8zVQgkt');
}
}

```

ReadData.dart

```

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

import '../helper class/constants.dart';

class UserData extends StatefulWidget {
  const UserData({Key? key}) : super(key: key);

  @override
  State<UserData> createState() => _UserDataState();
}

class _UserDataState extends State<UserData> {
  @override
  Widget build(BuildContext context) {
    return SafeArea(
      child: Scaffold(
        appBar: AppBar(
          iconTheme: IconThemeData(),
          title: Text('Your Profile'),
          backgroundColor: blk,
        ),
        backgroundColor: Colors.white,
        body: Container(
          margin: EdgeInsets.all(5.0),
          padding: EdgeInsets.all(5.0),
          child: StreamBuilder<QuerySnapshot>(
            stream:

```

```

        FirebaseFirestore.instance.collection('UserData').snapshots()
    ),
    builder: (context, UserSnapshot) {
      if (UserSnapshot.connectionState == ConnectionState.waiting) {
        return Center(
          child: CircularProgressIndicator(),
        );
      } else {
        int index = 0;
        final email =
            FirebaseAuth.instance.currentUser!.email.toString();
        final UserDocs = UserSnapshot.data!.docs;
        for (int i = 0; i < UserDocs.length; i++) {
          if (UserDocs[i]['Email'] == email) {
            index = i;
          } else {
            i++;
          }
        }

        return Column(
          children: [
            Icon(
              Icons.person,
              size: 50.0,
            ),
            SizedBox(
              height: 20.0,
            ),
            ListTile(
              contentPadding: EdgeInsets.all(5.0),
              shape: RoundedRectangleBorder(
                borderRadius:
                  BorderRadius.circular(5.0)),
              tileColor: blk,
              leading: Icon(
                Icons.mail,
                color: ylw,
              ),
              title: Text(
                UserDocs[index]['Email'],
                style: TextStyle(fontSize: 20.0, color: ylw),
              ),
            ),
            SizedBox(
              height: 5.0,
            ),
            ListTile(
              contentPadding: EdgeInsets.all(5.0),

```

```

        shape: RoundedRectangleBorder(
          borderRadius:
            BorderRadius.all(Radius.circular(5.0))),
        tileColor: blk,
        leading: Icon(
          Icons.person,
          color: ylw,
        ),
        title: Text(
          UserDocs[index]['Full name'],
          style: TextStyle(fontSize: 20.0, color: ylw),
        ),
      ),
    ),
    SizedBox(
      height: 5.0,
    ),
    ListTile(
      contentPadding: EdgeInsets.all(5.0),
      shape: RoundedRectangleBorder(
        borderRadius:
          BorderRadius.all(Radius.circular(5.0))),
      tileColor: blk,
      leading: Icon(
        Icons.home,
        color: ylw,
      ),
      title: Text(
        UserDocs[index]['address'],
        style: TextStyle(fontSize: 20.0, color: ylw),
      ),
    ),
  ),
  SizedBox(
    height: 5.0,
  ),
  ListTile(
    contentPadding: EdgeInsets.all(5.0),
    shape: RoundedRectangleBorder(
      borderRadius:
        BorderRadius.all(Radius.circular(5.0))),
    tileColor: blk,
    leading: Icon(
      Icons.calendar_month,
      color: ylw,
    ),
    title: Text(
      UserDocs[index]['DOB'],
      style: TextStyle(fontSize: 20.0, color: ylw),
    ),
  ),

```

```

    ),
    SizedBox(
      height: 10.0,
    ),
    Text(
      'Change password',
      style: TextStyle(
        color: blk,
        fontSize: 18.0,
        fontWeight: FontWeight.w500,
      ),
    ),
  ],
);
}
}),
);
}
}
}

```

2. Made an Investment planner page where users can calculate outcomes of SIP and Lumpsum investment as per invested amount and investment period.

InvestmentPlanner.dart

```

import 'dart:ffi';
import 'dart:math';

import 'package:annonymus_app/helper%20class/constants.dart';
import 'package:flutter/material.dart';
import 'package:flutter/src/foundation/key.dart';
import 'package:flutter/src/widgets/framework.dart';
import 'package:http/http.dart';

class LaxmiChitFund extends StatefulWidget {
  const LaxmiChitFund({Key? key}) : super(key: key);

  @override
  State<LaxmiChitFund> createState() => _LaxmiChitFundState();
}

```

```

}

class _LaxmiChitFundState extends State<LaxmiChitFund> {
  var maxInvestment = 100000.0;
  var finalExpectedResult = '0';
  double years = 5.0;
  double? periodicRate;
  double maxPercentage = 30.0;
  double maxYear = 35.0;
  double returnPercentage = 5.0;
  Color btn1Color = blk;
  double investment = 10000.0;
  double returnAmount = 0.0;
  double InvestedAmount = 0.0;
  String investmentType = 'Investment';
  Color btn2Color = blk;
  Color ctnColor = Color.fromARGB(255, 14, 20, 56);
  Color ctnBorder = Color.fromARGB(255, 14, 20, 56);
  bool SIP = true;
  @override
  Widget build(BuildContext context) {
    void getAmount() {
      if (investmentType == 'Monthly\nInvestment') {
        InvestedAmount = investment * years * 12;
        periodicRate = (returnPercentage / (years * 12)) / 100;
        num temp = pow(1 + periodicRate!, years * 12 - 1);
        returnAmount = investment * temp * (1 + periodicRate!) / periodicRate!;
        finalExpectedResult = returnAmount.toInt().toString();
      } else {
        print(SIP);
        InvestedAmount = investment * years;
        returnAmount = investment * pow(1 + returnPercentage / 100, years);
        finalExpectedResult = returnAmount.toInt().toString();
      }
    }
  }

  print(InvestedAmount);
  return Scaffold(
    backgroundColor: Color.fromARGB(255, 14, 20, 56),
    appBar: AppBar(
      title: Text('Laxmi Chit Fund'),
      backgroundColor: Color(0xFFEB1555),
    ),
    body: Container(
      margin: EdgeInsets.all(10.0),
      padding: EdgeInsets.all(5.0),
      child: Column(
        children: [

```

```

Row(
  children: [
    ElevatedButton(
      style: ElevatedButton.styleFrom(
        primary: btn1Color,
        onPrimary: ylw,
        onSurface: blk, // Disable color
      ),
      onPressed: () {
        setState(() {
          SIP = true;

          if (SIP == true) {
            maxInvestment = 100000;

            btn1Color = Color(0xFFEB1555);
            btn2Color = blk;
            investmentType = 'Monthly\nInvestment';
            ctnBorder = grey2;
            ctnColor = grey;

            //InvestedAmount = investment * years * 12;

          }
        });
      },
      child: Text('SIP'),
    ),
    SizedBox(
      width: 1.0,
    ),
    ElevatedButton(
      style: ElevatedButton.styleFrom(
        primary: btn2Color,
        onPrimary: ylw,
        onSurface: blk, // Disable color
      ),
      onPressed: () {
        setState(() {
          SIP = false;

          if (SIP == false) {
            maxInvestment = 1000000;
            btn2Color = Color(0xFFEB1555);
            ctnBorder = grey2;
            btn1Color = blk;
            ctnColor = grey;
            investmentType = 'Total\nInvestment';
          }
        });
      },
      child: Text('Total'),
    ),
  ],
)

```

```

        //InvestedAmount = investment * years;

    }
    });
  },
  child: Text('Lumpsum'),
),
],
),
Container(
  height: 425.0,
  width: 450.0,
  decoration: BoxDecoration(
    border: Border.all(width: 2, color: ctnBorder),
    borderRadius: BorderRadius.all(Radius.circular(20)),
    color: Color.fromARGB(255, 12, 18, 52),
    boxShadow: [
      BoxShadow(
        color: ctnColor,
        blurRadius: 8,
        spreadRadius: 2,
        offset: Offset(4, 4),
      )
    ],
  ),
  child: Column(
    children: [
      Container(
        margin: EdgeInsets.only(top: 15.0),
        width: 320.0,
        //margin: EdgeInsets.fromLTRB(20, 0.0, 20.0, 0.0),
        height: 120.0,
        decoration: BoxDecoration(
          color: CardColor,
          borderRadius: BorderRadius.circular(10.0),
        ),
        padding: EdgeInsets.only(top: 5.0),
        child: Column(
          children: [
            Row(
              mainAxisAlignment: MainAxisAlignment.spaceEvenly,
              children: [
                Text(
                  investmentType,
                  style: headStyle,
                ),
                Text(
                  investment.toInt().toString(),
                  style: numberText,

```



```

        ),
    ],
),
SliderTheme(
    data: SliderTheme.of(context).copyWith(
        activeTrackColor: Colors.white,
        inactiveTrackColor: Color(0xFF8D8E98),
        thumbColor: Color(0xFFEB1555),
        thumbShape:
15.0),
            RoundSliderThumbShape(enabledThumbRadius:

            overlayShape:
                RoundSliderOverlayShape(overlayRadius: 30.0),
            overlayColor: Color(0x29EB1555),
            trackHeight: 0.5,
        ),
        child: Slider(
            value: investment,
            onChanged: (double newVal) {
                setState(() {
                    investment = newVal.roundToDouble();
                    getAmount();
                });
            },
            min: 500,
            max: SIP
                ? maxInvestment = 100000.0
                : maxInvestment =
                    1000000.0 //maxInvestment.toDouble(),
        ),
    )
],
),
),
SizedBox(
    height: 7.0,
),
Container(
    margin: EdgeInsets.only(top: 8.0),
    width: 320.0,
    //margin: EdgeInsets.fromLTRB(20, 0.0, 20.0, 0.0),
    height: 120.0,
    decoration: BoxDecoration(
        color: CardColor,
        borderRadius: BorderRadius.circular(10.0),
    ),
    padding: EdgeInsets.only(top: 5.0),
    child: Column(

```

```

        children: [
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceAround,
            children: [
              Text(
                'Expected\nreturn (p.a)',
                style: headStyle,
              ),
              Text(
                returnPercentage.toStringAsPrecision(3),
                style: numberText,
              ),
            ],
          ),
          SliderTheme(
            data: SliderTheme.of(context).copyWith(
              activeTrackColor: Colors.white,
              inactiveTrackColor: Color(0xFF8D8E98),
              thumbColor: Color(0xFFEB1555),
              thumbShape:
15.0),
                RoundSliderThumbShape(enabledThumbRadius:

                overlayShape:
                  RoundSliderOverlayShape(overlayRadius: 30.0),
                overlayColor: Color(0x29EB1555),
                trackHeight: 0.5,
            ),
            child: Slider(
              value: returnPercentage,
              onChanged: (double newVal) {
                setState(() {
                  returnPercentage = newVal;
                  getAmount();
                });
              },
              min: 1,
              max: maxPercentage),
          )
        ],
      ),
    ),
    SizedBox(
      height: 7.0,
    ),
    Container(
      width: 320.0,
      margin: EdgeInsets.only(bottom: 15.0, top: 8.0),
      height: 120.0,

```

```

        decoration: BoxDecoration(
          color: CardColor,
          borderRadius: BorderRadius.circular(10.0),
        ),
        padding: EdgeInsets.only(top: 5.0),
        child: Column(
          children: [
            Row(
              mainAxisAlignment: MainAxisAlignment.spaceAround,
              children: [
                Text(
                  'Time\ncperiod',
                  style: headStyle,
                ),
                Text(
                  years.toInt().toString(),
                  style: numberText,
                ),
              ],
            ),
            SliderTheme(
              data: SliderTheme.of(context).copyWith(
                activeTrackColor: Colors.white,
                inactiveTrackColor: Color(0xFF8D8E98),
                thumbColor: Color(0xFFEB1555),
                thumbShape:
15.0),
                  RoundSliderThumbShape(enabledThumbRadius:

                  overlayShape:
                    RoundSliderOverlayShape(overlayRadius: 30.0),
                  overlayColor: Color(0x29EB1555),
                  trackHeight: 0.5,
                ),
              child: Slider(
                value: years,
                onChanged: (double newVal) {
                  setState(() {
                    years = newVal.roundToDouble();
                    getAmount();
                  });
                },
                min: 1,
                max: maxYear,
              ),
            ),
          ],
        ),
      ),
    ),
  ),
),

```

```

        ],
      ),
    ),
    SizedBox(
      height: 25.0,
    ),
    Text(
      'Total Investment',
      style: headStyle,
    ),
    Text(
      '₹$InvestedAmount',
      style: numberText,
    ),
    SizedBox(
      height: 15.0,
    ),
    Text(
      'Expected Return',
      style: headStyle,
    ),
    Text(
      '₹$finalExpectedResult',
      style: numberText,
    ),
  ],
),
);
}
}

```

Screenshots:

12:24 87%

← SignUp

Enter Your Details

usertest

user5@gmail.com

01/01/01

.....

.....

near dena bank bhuj

SIGN UP

12:26 88%

← Your Profile

user5@gmail.com

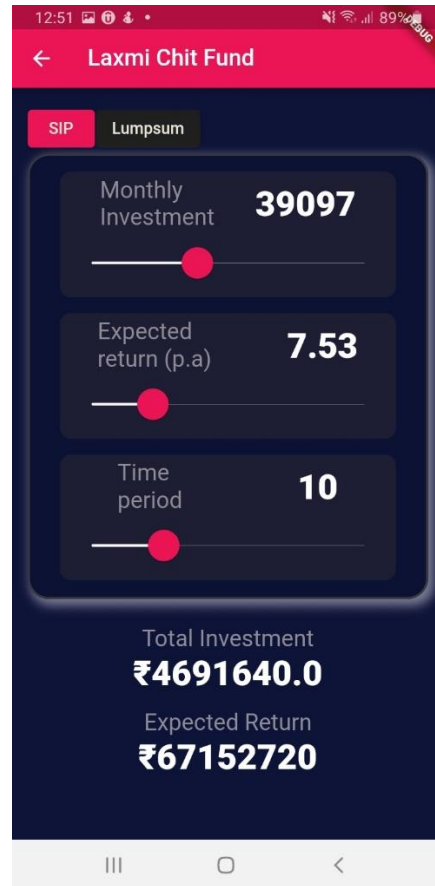
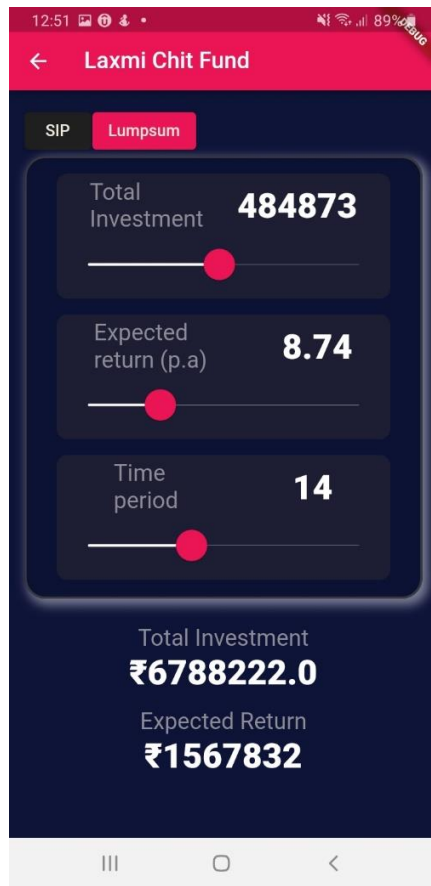
usertest

near dena bank bhuj

01/01/01

Change password

augmentation-cee6d	UserData	SuS1ftHBC5ZeJF4bKqNg
+ Start collection	+ Add document	+ Start collection
User	23IyapIrgk1qL9lQXjWw	+ Add field
UserData >	5PMdzzZ47MXLg8zVQgkt	DOB: "01/01/01"
	EGZnr9UjAGroqF3gQsa0	Email: "user5@gmail.com"
	JhMPdjPPEcFJwts7WzVR	Full name: "usertest"
	SuS1ftHBC5ZeJF4bKqNg >	address: "near dena bank bhuj"
	T0G3GBR5Wp00dgXya1Qs	
	gMnbFNRZ1b97sC0KTdbF	
	gami	



References:

- <https://firebase.flutter.dev/docs/firestore/usage/>
- <https://www.franklintempletonindia.com/investor/lumpsum-calculator>
- <https://blog.codemagic.io/integrate-firebase-cloud-firestore-with-flutter-perform-crud/>

Signature of External Supervisor

Signature of Internal Supervisor