

MAD & PWA Lab Journal

Experiment No.	05
Experiment Title.	To apply navigation, routing and gestures in Flutter App
Roll No.	50
Name	Siddhant Sathe
Class	D15A
Subject	MAD & PWA Lab
Lab Outcome	LO1: Understand cross platform mobile application development using Flutter framework
Grade:	

Code: home_page.dart

```
import 'package:flutter/material.dart';
import 'ask_question.dart';
import 'profile_page.dart';
import 'answer_page.dart';
import
'package:firebase_database/firebase_databas
e.dart';
import
'package:cloud_firestore/cloud_firestore.dart';
```

```
class HomePage extends StatefulWidget {
  @override
  State<HomePage> createState() =>
  _HomePageState();
}
```

```
class _HomePageState extends
State<HomePage> {
  int _selectedIndex = 0;
  final List<String> _appBarTitles =
["Questions", "My Profile", "Settings"];

  // Firebase reference
  final DatabaseReference _database =
FirebaseDatabase.instance.ref('question_ans
wers');
```

```
void _onItemTapped(int index) {
  setState(() {
    _selectedIndex = index;
  });
}
```

```
void addQuestion(String title, String details) {
  _database.push().set({
    'title': title,
    'details': details,
    'timestamp':
DateTime.now().toIso8601String(),
  });
}
```

```
List<Widget> get _pages => [
  StreamBuilder<QuerySnapshot>(
```

```
    stream:
FirebaseFirestore.instance.collection('questio
n_answers').snapshots(),
    builder: (context, snapshot) {
      if (snapshot.connectionState ==
ConnectionState.waiting) {
        return const Center(child:
CircularProgressIndicator());
      }
    }
```

```
    if (!snapshot.hasData ||
snapshot.data!.docs.isEmpty) {
      return const Center(child: Text("No
questions asked yet"));
    }
```

```
    final questions = snapshot.data!.docs;

    return ListView.builder(
      itemCount: questions.length,
      itemBuilder: (context, index) {
        final question =
questions[index].data() as Map<String,
dynamic>;
        return Padding(
          padding: const
EdgeInsets.symmetric(vertical: 8.0, horizontal:
16.0),
```

```
          child: ElevatedButton(
            onPressed: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (context) =>
QuestionDetailPage(
                    title: question['title'] ?? 'No
title',
                    details: question['details'] ??
'No details',
                    timestamp:
(question['timestamp'] as
Timestamp).toDate().toString(),
                  ),
                ),
```

```

);
},
child: Column(
  crossAxisAlignment:
CrossAxisAlignment.start,
  children: [
    Align(
      alignment:
Alignment.centerLeft,
      child: Text(
        question['title'] ?? 'No title',
        style: const TextStyle(
          fontSize: 18,
          fontWeight:
FontWeight.bold,
        ),
      ),
      child: Text(
        question['details'] ?? 'No
details',
        style: const TextStyle(fontSize:
14),
      ),
    ),
  ],
),
style:
ElevatedButton.styleFrom(
  backgroundColor:
Colors.white,
  foregroundColor: Colors.black,
  elevation: 4,
  padding: const
EdgeInsets.all(16),
  shape:
RoundedRectangleBorder(
    borderRadius:
BorderRadius.circular(8),
  ),
),
),

```

```

);
},
);
},
),
ProfilePage(),
const Center(
  child: Text("Settings Page", style:
TextStyle(fontSize: 20, fontWeight:
FontWeight.bold)),
),
];

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor:
Colors.purple.shade50,
    appBar: AppBar(
      backgroundColor: Colors.white,
      elevation: 1,
      title: Text(
        _appBarTitles[_selectedIndex],
        style: const TextStyle(color:
Colors.black, fontSize: 18, fontWeight:
FontWeight.bold),
      ),
      actions: _selectedIndex == 0
? [
        TextButton(
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(
                builder: (context) =>
AskQuestionPage(onSubmit: addQuestion),
              ),
            );
          },
          child: const Text(
            "Ask Question",
            style: TextStyle(color:
Colors.purple, fontWeight: FontWeight.bold),
          ),
        ),
      ],
      const SizedBox(width: 10),
    ),
  ),

```



```

        .doc(title)
        .collection('answers')
        .orderBy('timestamp',
descending: true)
        .snapshots(),
        builder: (context, snapshot) {
            if (snapshot.connectionState
== ConnectionState.waiting) {
                return const Center(child:
CircularProgressIndicator());
            }

            if (!snapshot.hasData ||
snapshot.data!.docs.isEmpty) {
                return const Center(child:
Text("No answers yet"));
            }

            final answers =
snapshot.data!.docs;

            return ListView.builder(
                itemCount: answers.length,
                itemBuilder: (context, index) {
                    final answer =
answers[index].data() as Map<String,
dynamic>;

                    return Padding(
                        padding: const
EdgeInsets.symmetric(vertical: 8.0, horizontal:
16.0),
                        child: Container(
                            padding: const
EdgeInsets.all(16),
                            decoration: BoxDecoration(
                                color: Colors.white,
                                borderRadius:
BorderRadius.circular(8),
                                boxShadow: [
                                    BoxShadow(
                                        color:
Colors.grey.withOpacity(0.2),

```

```

                                spreadRadius: 1,
                                blurRadius: 4,
                                offset: const Offset(0,
2),
                                ),
                            ],
                        ),
                        child: Column(
                            crossAxisAlignment:
CrossAxisAlignment.start,
                            children: [
                                Text(
                                    answer['answer'] ??
'No answer',
                                    style: const
TextStyle(fontSize: 16),
                                ),
                                const SizedBox(height:
8),
                                Text(
                                    "Answered on:
${DateTime.parse(answer['timestamp']).toStri
ng()}",
                                    style: const
TextStyle(fontSize: 12, color: Colors.grey),
                                ),
                            ],
                        ),
                    ],
                ),
            ),
        ],
    );
}
}

```

Code: ask_question.dart

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

class AskQuestionPage extends
StatefulWidget {
  final Function(String, String) onSubmit;

  AskQuestionPage({required this.onSubmit});

  @override
  _AskQuestionPageState createState() =>
  _AskQuestionPageState();
}

class _AskQuestionPageState extends
State<AskQuestionPage> {
  final _formKey = GlobalKey<FormState>();
  final _titleController = TextEditingController();
  final _detailsController =
TextEditingController();

  Future<void> _submitQuestion() async {
    if (_formKey.currentState!.validate()) {
      try {
        await
FirebaseFirestore.instance.collection('question_answers').add({
          'title': _titleController.text,
          'details': _detailsController.text,
          'timestamp':
FieldValue.serverTimestamp(),
        });

        ScaffoldMessenger.of(context).showSnackBar
(
          const SnackBar(content:
Text('Question submitted successfully!')),

```

```
);
        widget.onSubmit(_titleController.text,
_detailsController.text);
        Navigator.pop(context);
      } catch (e) {

        ScaffoldMessenger.of(context).showSnackBar
(
          SnackBar(content: Text('Failed to
submit question: $e')),
        );
      }
    }
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: const
Color(0xFFFF7EBFB),
      appBar: AppBar(
        title: const Text("Ask Question"),
        backgroundColor: Colors.white,
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Form(
          key: _formKey,
          child: Column(
            children: [
              TextFormField(
                controller: _titleController,
                decoration: const
InputDecoration(labelText: 'Title',
                  validator: (value) => value!.isEmpty ?
'Enter a title' : null,
                ),
              const SizedBox(height: 16),
              TextFormField(

```

```

        controller: _detailsController,
        maxLines: 5,
        decoration: const
InputDecoration(labelText: 'Details'),
        validator: (value) => value!.isEmpty ?
'Enter details' : null,
    ),
    const SizedBox(height: 24),
    ElevatedButton(
        onPressed: _submitQuestion,

```

```

        child: const Text("Review Your
Question"),
    ),
    ],
    ),
    ),
    );
    }
}

```

Code:profile_page.dart


```

import 'package:flutter/material.dart';
import 'package:flutter_svg/flutter_svg.dart';

class ProfilePage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor:
Colors.purple.shade50,
      body: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.all(20.0),
          child: Column(
            crossAxisAlignment:
CrossAxisAlignment.start,
            children: [
              Row(
                children: [
                  CircleAvatar(
                    radius: 40,
                    child:
SvgPicture.asset('project/assets/images/icon.
svg'),
                ),
                const SizedBox(width: 16),
                Column(
                  crossAxisAlignment:
CrossAxisAlignment.start,
                  children: const [
                    Text(
                      'Siddhant Sathé',
                      style: TextStyle(

```

```

fontSize: 20,
fontWeight: FontWeight.bold,
        ),
      ),
      SizedBox(height: 4),
      Text(
        '
 Member since 1st Feb
2025',
        style: TextStyle(
          color: Colors.grey,
          fontSize: 14,
        ),
      ),
    ],
  ),
  const SizedBox(height: 20),
  Row(
    mainAxisAlignment:
MainAxisAlignment.spaceBetween,
    children: const [
      Text(
        'About',
        style: TextStyle(
          fontSize: 18,
          fontWeight: FontWeight.bold,
        ),
      ),
      Text(
        'Log out',
        style: TextStyle(

```

```

        color: Colors.blue,
        fontWeight: FontWeight.bold,
      ),
    ],
  ),
  const SizedBox(height: 10),
  Container(
    width: double.infinity,
    padding: const EdgeInsets.all(16),
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius:
BorderRadius.circular(12),
    ),
    child: const Text(
      'No about me has been yet',
      style: TextStyle(
        color: Colors.grey,
      ),
    ),
  ),
  const SizedBox(height: 20),
  Row(
    mainAxisAlignment:
MainAxisAlignment.spaceBetween,
    children: const [
      Text(
        'Answers',
        style: TextStyle(
          fontSize: 18,
          fontWeight: FontWeight.bold,
        ),
      ),
      Text(
        'See All',
        style: TextStyle(
          color: Colors.blue,
          fontWeight: FontWeight.bold,
        ),
      ),
    ],
  ),
  const SizedBox(height: 10),
  Container(
    width: double.infinity,

```

```

padding: const EdgeInsets.all(16),
decoration: BoxDecoration(
  color: Colors.white,
  borderRadius:
BorderRadius.circular(12),
),
    child: const Text(
      'You have not answered any
questions.',
      style: TextStyle(
        color: Colors.grey,
      ),
    ),
  ),
  const SizedBox(height: 20),
  Row(
    mainAxisAlignment:
MainAxisAlignment.spaceBetween,
    children: const [
      Text(
        'Questions',
        style: TextStyle(
          fontSize: 18,
          fontWeight: FontWeight.bold,
        ),
      ),
      Text(
        'See All',
        style: TextStyle(
          color: Colors.blue,
          fontWeight: FontWeight.bold,
        ),
      ),
    ],
  ),
  const SizedBox(height: 10),
  Container(
    width: double.infinity,
    padding: const EdgeInsets.all(16),
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius:
BorderRadius.circular(12),
    ),
    child: const Text(

```



```

questions.',
    style: TextStyle(
      color: Colors.grey,
    ),
  ),
),
),

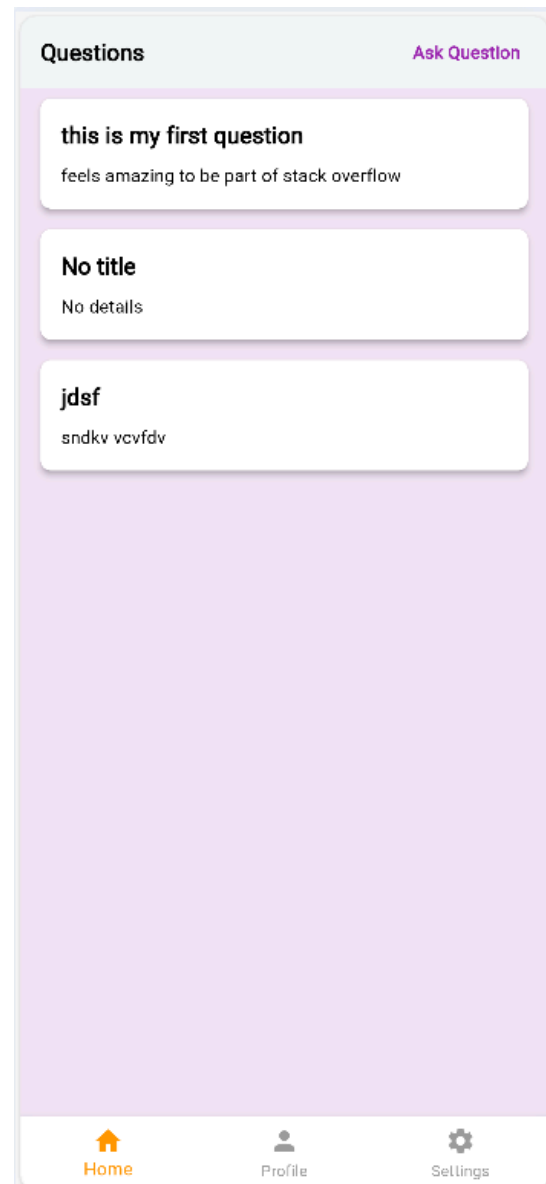
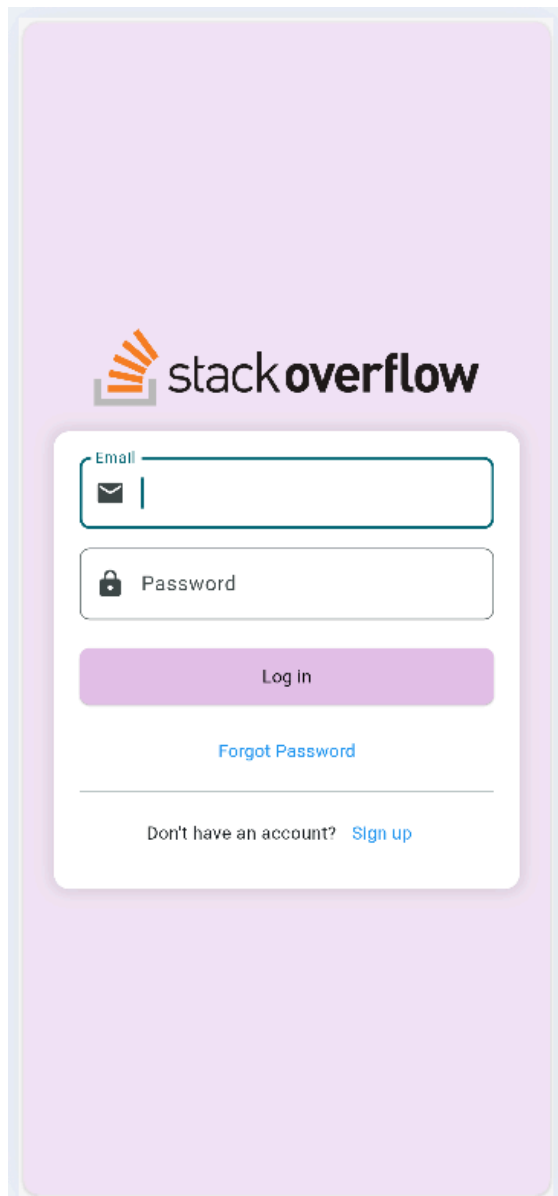
```

```

],
),
),
),
);
}
}

```

Output:



←

Ask Question

Ask a public question

Title

Be specific and imagine you're asking a question to another person.

Title

What are the details of your problem?

Introduce the problem and expand on what you put in the title. Minimum 20 characters.

Content

Review your question

My Profile

Siddhant Sathe

MEMBER

Member since 1st Feb 2025

About

Log out

No about me has been yet

Answers

See All

You have not answered any questions.

Questions

See All

You have not asked any questions.

Home

Profile

Settings