

MINI-PROJECT
(2020-2021)

BLOGGING APP

FINAL REPORT



SUBMITTED BY:
AYURISH CHANDNA
(181500166)
AMAN KUMAR
(181500072)

SUBMITTED TO:
Mr. AKASH KUMAR CHOUDHARY
(Asst. Professor)
Department of Computer Engineering and Applications
Institute of Engineering and Technology

Acknowledgment

I wish to express my profound gratitude and indebtedness to Mr. Akash Kumar Choudhary, Department of Computer Engineering and Applications, GLA University for introducing the present topic and for their inspiring guidance and valuable suggestion throughout the project work.

Last but not least, my sincere thanks to all our friends who have patiently extended all sorts of help for accomplishing this undertaking.

Ayurish Chandna(181500166)

Aman Kumar(181500072)

Dept. of Computer Engineering and Applications

GLA University



**Department of computer Engineering and Applications
GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406**

CERTIFICATE

This is to certify that the Final Report entitled, “Blogging App” submitted by Mr./Ms. “Ayurish Chandna” and “Aman Kumar” to GLA University, Mathura, is a record of bonafide Project work carried out by them under the supervision of “Mr. Akash Kumar Choudhary” and guidance and is worthy of consideration for the award of the degree of Bachelor of Technology in Computer Science & Engineering of the Institute.

Signature of Candidate: Ayurish Chandna

Aman Kumar

Name of Candidate: Ayurish Chandna (181500166)

Aman Kumar (181500072)

Course: B.Tech (CS)

Year: 2020-2021

Semester: 5th

Training Certificates

Ayurish Chandna –



Certificate of Completion

*This is to certify that Ayurish chandna
successfully completed 3.5 total hours of Flutter -
Intermediate online course on Oct. 10, 2020*

Bryan Cairns

Bryan Cairns, Instructor

&



Certificate no: UC-5ebfcf65-aa11-4e82-8edc-f5b47ff6aca7
Certificate url: ude.my/UC-5ebfcf65-aa11-4e82-8edc-f5b47ff6aca7

#BeAble



CERTIFICATE of Participation

THIS IS TO CERTIFY THAT

AYURISH CHANDNA

GLA UNIVERSITY

HAS PARTICIPATED IN FLUTTER BOOTCAMP

DURING graVITas'20 - A TECHNO-MANAGEMENT CARNIVAL DURING OCTOBER 1-4, 2020
AT VELLORE INSTITUTE OF TECHNOLOGY, VELLORE.

A handwritten signature in blue ink, appearing to read 'Ayurish Chandna'.

CONVENOR

A handwritten signature in blue ink, appearing to read 'Dr. S. Venkateswaran'.

PRO VICE CHANCELLOR



Aman Kumar:





Index

Acknowledgment

Certificate

Project Description

Working code

Working Screenshots

Conclusion

Project Description

1. The Problem Statement

In college students finds difficulty in getting connected to a right senior or mentor who can help them in a particular field in which they are interested. Being away from home students lose focus and get's demotivated. It is seen that most of the students lack clarity in knowing what they want to in their career and they end up doing a job that doesn't match their interest.

2. Objectives of the project

We are developing a Blogging App. There is at most need for this nowadays. Many students who are not fortunate enough to study in schools or colleges. If we have got this opportunity to study then why not do something good and contribute to our Nation. Even its seen that how students are just following the set lanes and getting a degree but they all lack skills. The main cause is a lack of mentorship. The age between 20-25 is very crucial.

This project will give a very crucial contribution to the field of education among college students who are unclear about how to build a successful career. It will boost the student's confidence and motivation. It will also help students in developing various skills like trending technical skills, decision making, storytelling, communication skills, marketing skills, etc. It will help students getting clear what are their interests and how to convert that interest into a profession. It will help our country to increase the employability level.

Flutter helps in building apps very efficiently. Flutter allows developers to build apps for both major platforms, including Android and iOS, with a single code base, therefore allowing start-ups to save a high cost on development services. Flutter has an efficient yet portable GPU rendering UI that allows it to work on multiple interfaces.

Why Flutter ?

Open-Source and Free to Use

Both Flutter and the underlying programming language, Dart are open-source and free to use. This empowers developers to view the documentation, post their

queries in the open developer forum, and eventually, learn and implement the code.

Hot Reload Functionality

One of the biggest reasons why Google Flutter is preferred by mobile application development agencies is its hot reloading functionality.

The functionality makes it easier for anyone to make any change at the backend and view the difference at the front-end simultaneously. And this way, easily decide whether the change looks perfect with other elements of the application or not.

Lower Development Timeline and Cost

One of the prime benefits of focusing on Flutter development is that it renders opportunities to create native-like apps for both iOS and Android using a single codebase. Something that makes the cost to create a Flutter mobile app considerably low.

Support for Google Firebase

With the facility of Firebase integration, Flutter enables developers to add backend facility into the development environment.

Flutter Testing Support

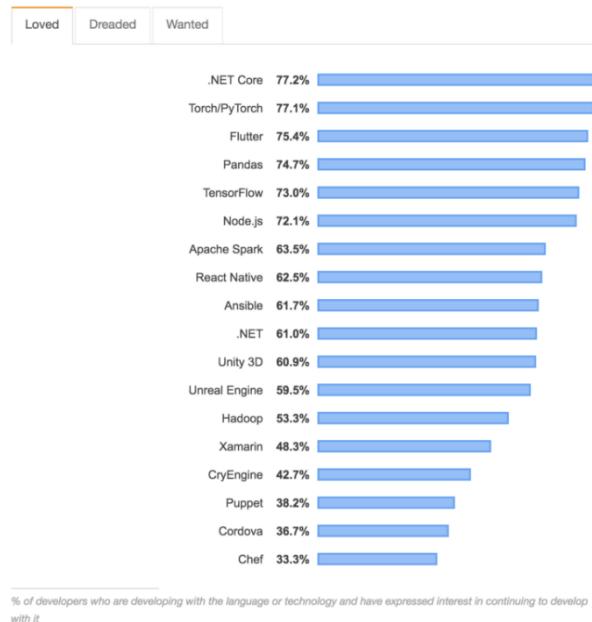
It is commonly known among the developers that Native Android tools support testing. Testing for native apps can be done using the command line of Gradle. One can add elements, integration and UI tests for Android apps. Numerous frameworks in Java and Kotlin are available for lower-level testing and also for UI testing. There are various levels of testing Android apps.

Flutter provides solid testing that allows quality analysts to write tests at various levels of such as unit, UI. One of the best features Flutter offers is Widget testing which runs UI test fast, equivalent to unit tests. If you are a beginner Flutter developer, you can refer to several levels of testing Android apps in this shell. It runs in a separate process that can be run on real devices or emulators. Flutter has a separate package called Flutter Driver to drive the test in an easy manner.

According to Stack overflow :

Flutter ranks higher with 75.4% and React Native also made the cut with 62.5% among most loved frameworks.

Most Loved, Dreaded, and Wanted Other Frameworks, Libraries, and Tools



% of developers who are developing with the language or technology and have expressed interest in continuing to develop with it

3. Future Scope

Blogging comes in many forms, it addresses countless topics, and it can, without a doubt, be divided by its quality. But when it comes to education, blogging can be an excellent tool for improving a student's involvement, excitement, and most importantly, writing skills.

So the future scope of blogging is very vast and it has an infinite number of benefits, especially for young people. So here are some of the benefits regarding blogging-

- Blogging is effective and is still a very popular marketing tool in 2020. The majority of internet users in 2020 regularly read blogs.
- They can develop much-improved writing skills which will stick with them forever, their thinking muscles will be in continuous development, and they will also develop a better understanding of the world that surrounds them.
- According to the World Economic Forum Creativity comes at number 3rd in the most important skills for the future industry.

- One of the best benefits that come as a result of blogging is that it allows people to express and hone their creativity. This can turn a boring paper into a fun creative process.
- Blogging is a great source to earn an income as well!

4. Hardware & Software to be used

Hardware requirements –

1. System Software

- Operating System (Windows 10, Linux)
- Core i3 Processor (minimum)
- Minimum 8 GB Ram

Software Requirements

1. Application Software

- Android Studio
- VS Code

2. Programming Language

- Flutter
- Dart

3. Back-end

- FireBase
- FireStore

5. What contribution would the project make and where?

This project will give a very crucial contribution to the field of education among college students who are unclear about how to build a successful career. It will boost the student's confidence and motivation. It will also help students

in developing various skills like trending technical skills, decision making, storytelling, communication skills, marketing skills, etc. It will help students getting clear what are their interests and how to convert that interest into a profession. It will help our country to decrease the unemployability level.

6. Literature Survey

Students use various technologies for purposes of communicating from texting, to chat rooms, to e-mail. Children are now more able than ever to effectively communicate through technology. In today's world and more and more as we progress into the future, it is imperative that individuals possess skills that allow them to "access, gain, transform, and transmit information". It would seem that by using blogs, students can develop skills in various fields including technology. The use of blogs "can result in increased motivation among students to read, write, create, and produce results for meaningful purposes". Through social interactions mediated by language, blogs offer students the opportunity to surface their ideas on a social plane. The ability to comment on these ideas enables individuals to participate in the social construction of knowledge and meaning-making.

There are some negative aspects of blogging –

- "Don't confuse blogging with social networking". The academic focus must be clear for students to be engaged in academic tasks and social construction of knowledge.
- "Don't leap at the freebies". There are many free blog sites available; however, these sites often do not fit educational purposes. Advertisements and lack of educator control are two negatives of free blog sites.

Implementation Details

Part1: Designing UI

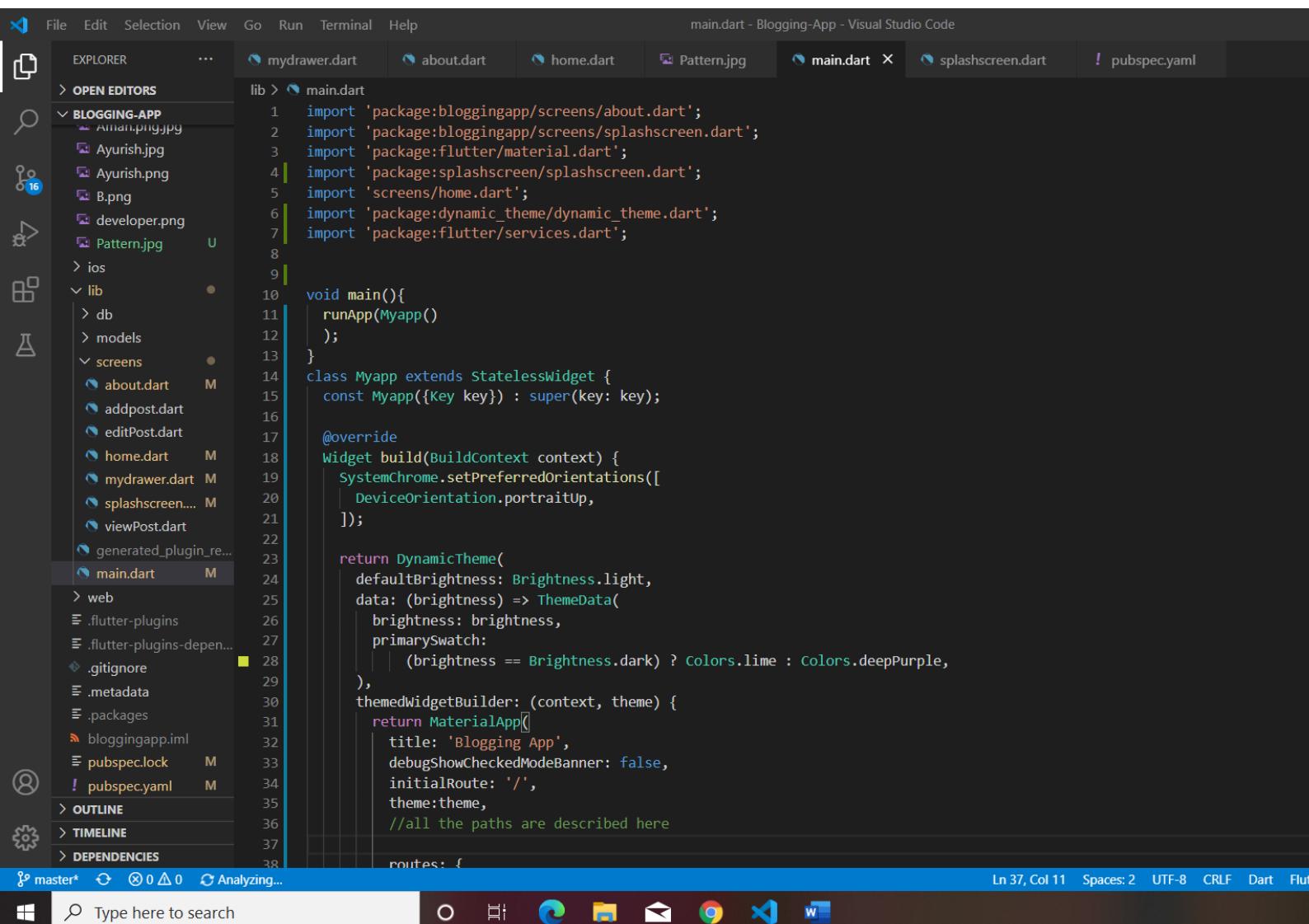
- Home Page
- Navigation Drawer
- About us
- Change Theme
- Welcome Screen
- Edit Post
- Delete Post

Part2: Backend work

- Configuring Firebase for android apps
- FireStore Configuration
- Upload Realtime Data to Firebase

Working Code and Working App Screenshots

File name: main.dart



The screenshot shows the Visual Studio Code interface with the main.dart file open in the editor. The code implements a dynamic theme for a Flutter application.

```
File Edit Selection View Go Run Terminal Help
main.dart - Blogging-App - Visual Studio Code

EXPLORER ... mydrawer.dart about.dart home.dart Pattern.jpg main.dart splashscreen.dart ! pubspec.yaml

lib > main.dart
1 import 'package:bloggingapp/screens/about.dart';
2 import 'package:bloggingapp/screens/splashscreen.dart';
3 import 'package:flutter/material.dart';
4 import 'package:splashscreen/splashscreen.dart';
5 import 'screens/home.dart';
6 import 'package:dynamic_theme/dynamic_theme.dart';
7 import 'package:flutter/services.dart';

8
9
10 void main(){
11   runApp(Myapp()
12 );
13 }
14 class Myapp extends StatelessWidget {
15   const Myapp({Key key}) : super(key: key);

16   @override
17   Widget build(BuildContext context) {
18     SystemChrome.setPreferredOrientations([
19       DeviceOrientation.portraitUp,
20     ]);

21   }

22   return DynamicTheme(
23     defaultBrightness: Brightness.light,
24     data: (brightness) => ThemeData(
25       brightness: brightness,
26       primarySwatch:
27         (brightness == Brightness.dark) ? colors.lime : colors.deepPurple,
28     ),
29     themedWidgetBuilder: (context, theme) {
30       return MaterialApp(
31         title: 'Blogging App',
32         debugShowCheckedModeBanner: false,
33         initialRoute: '/',
34         theme: theme,
35         //all the paths are described here
36         routes: f
37     );
38   }
39 }

Ln 37, Col 11 Spaces: 2 UTF-8 CRLF Dart Flt
master* 0 0 Analyzing...
Type here to search O H+ 🌐 📁 📧 🚗 🌐 🌐 🌐 🌐
```

File Edit Selection View Go Run Terminal Help

main.dart - Blogging-App - Visual Studio Code

EXPLORER OPEN EDITORS

BLOGGING-APP

- Amarni.png.jpg
- Ayurish.jpg
- Ayurish.png
- B.png
- developer.png
- pattern.jpg

ios

lib

- db
- models
- screens
- about.dart M
- addpost.dart
- editPost.dart
- home.dart M
- mydrawer.dart M
- splashscreen... M
- viewPost.dart

generated_plugin_re...

main.dart M

web

- .flutter-plugins
- .flutter-plugins-depend...
- .gitignore
- .metadata
- .packages
- bloggingapp.iml
- pubspec.lock M
- ! pubspec.yaml M

OUTLINE TIMELINE DEPENDENCIES

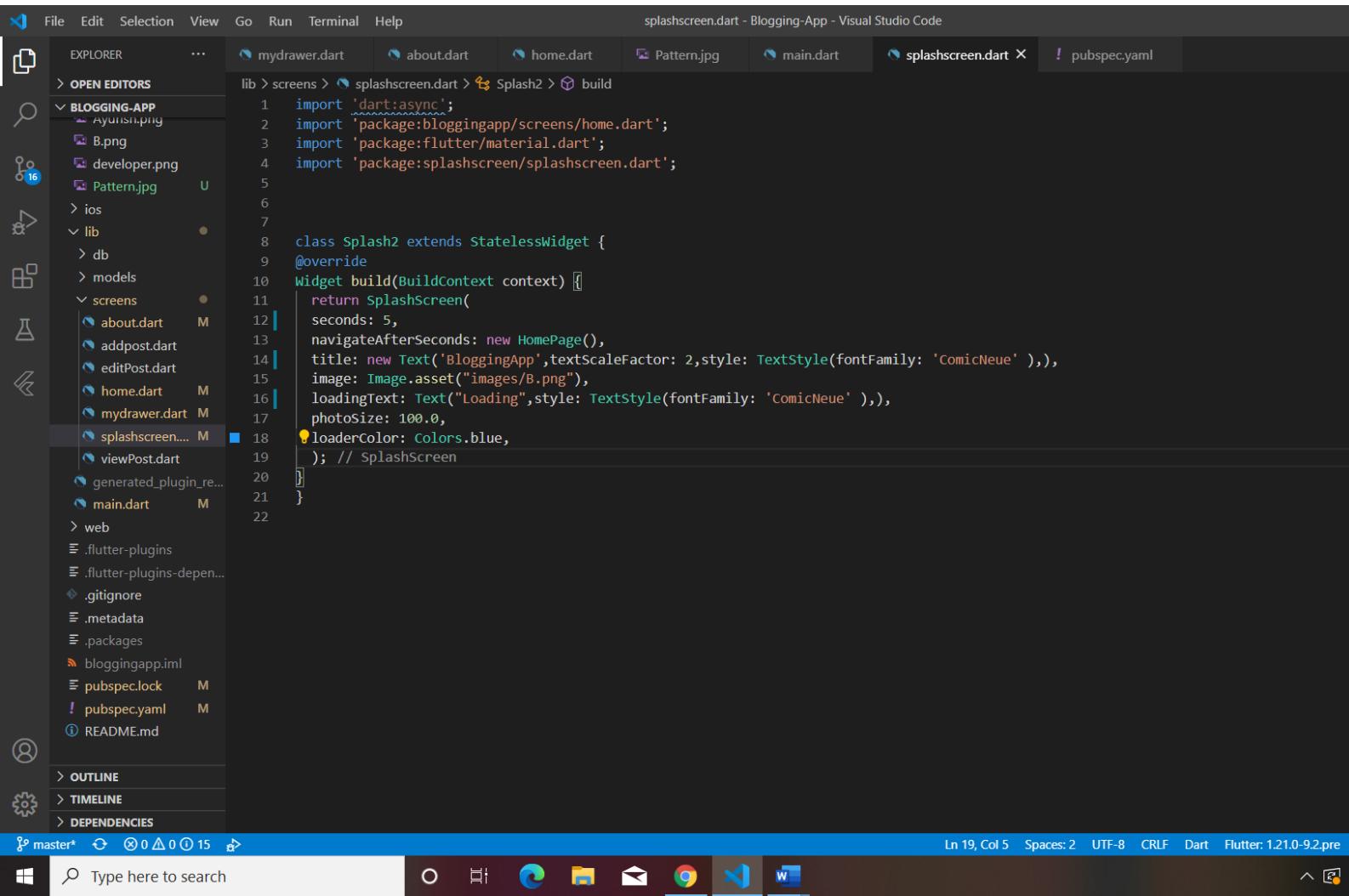
main.dart

```
14 class Myapp extends StatelessWidget {  
15   const Myapp({Key key}) : super(key: key);  
16  
17   @override  
18   Widget build(BuildContext context) {  
19     SystemChrome.setPreferredOrientations([  
20       DeviceOrientation.portraitUp,  
21     ]);  
22  
23     return DynamicTheme(  
24       defaultBrightness: Brightness.light,  
25       data: (brightness) => ThemeData(  
26         brightness: brightness,  
27         primarySwatch:  
28           (brightness == Brightness.dark) ? colors.lime : colors.deepPurple,  
29       ),  
30       themedWidgetBuilder: (context, theme) {  
31         return MaterialApp(  
32           title: 'Blogging App',  
33           debugShowCheckedModeBanner: false,  
34           initialRoute: '/',  
35           theme: theme,  
36           //all the paths are described here  
37           routes: {  
38             '/': (context) => Splash2(),  
39             '/about': (context) => About(),  
40           },  
41         );  
42       },  
43     );  
44   );  
45 }  
46 }  
47 }  
48 }  
49 }
```

master* 0 △ 0 Analyzing... Type here to search

Ln 37, Col 11 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0-9.2.pre Chrome

File name: splashscreen.dart



The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** splashscreen.dart - Blogging-App - Visual Studio Code
- Explorer:** Shows the project structure:
 - BLOGGING-APP
 - AYURISHPRG
 - B.png
 - developer.png
 - Pattern.jpg
 - ios
 - lib
 - db
 - models
 - screens
 - about.dart
 - addpost.dart
 - editPost.dart
 - home.dart
 - mydrawer.dart
 - splashscreen.dart
 - viewPost.dart
 - generated_plugin_re...
 - main.dart
 - web
 - .flutter-plugins
 - .flutter-plugins-depend...
 - .gitignore
 - .metadata
 - .packages
 - bloggingapp.iml
 - pubspec.lock
 - pubspec.yaml
 - README.md
- Editor:** The file `splashscreen.dart` is open, displaying the following code:

```
lib > screens > splashscreen.dart > Splash2 > build
1 import 'dart:async';
2 import 'package:bloggingapp/screens/home.dart';
3 import 'package:flutter/material.dart';
4 import 'package:splashscreen/splashscreen.dart';
5
6
7
8 class Splash2 extends StatelessWidget {
9   @override
10  Widget build(BuildContext context) {
11    return SplashScreen(
12      seconds: 5,
13      navigateAfterSeconds: new HomePage(),
14      title: new Text('BloggingApp',textScaleFactor: 2,style: TextStyle(fontFamily: 'ComicNeue' ),),
15      image: Image.asset("images/B.png"),
16      loadingText: Text("Loading",style: TextStyle(fontFamily: 'ComicNeue' ),),
17      photoSize: 100.0,
18      loaderColor: Colors.blue,
19    ); // SplashScreen
20  }
21 }
```
- Bottom Status Bar:** master*, 0 0 0 0 15, Ln 19, Col 5, Spaces: 2, UTF-8, CRLF, Dart, Flutter: 1.21.0-9.2.pre

Output:

12:41 AM

50.9KB/s   4G    



BloggingApp



Loading

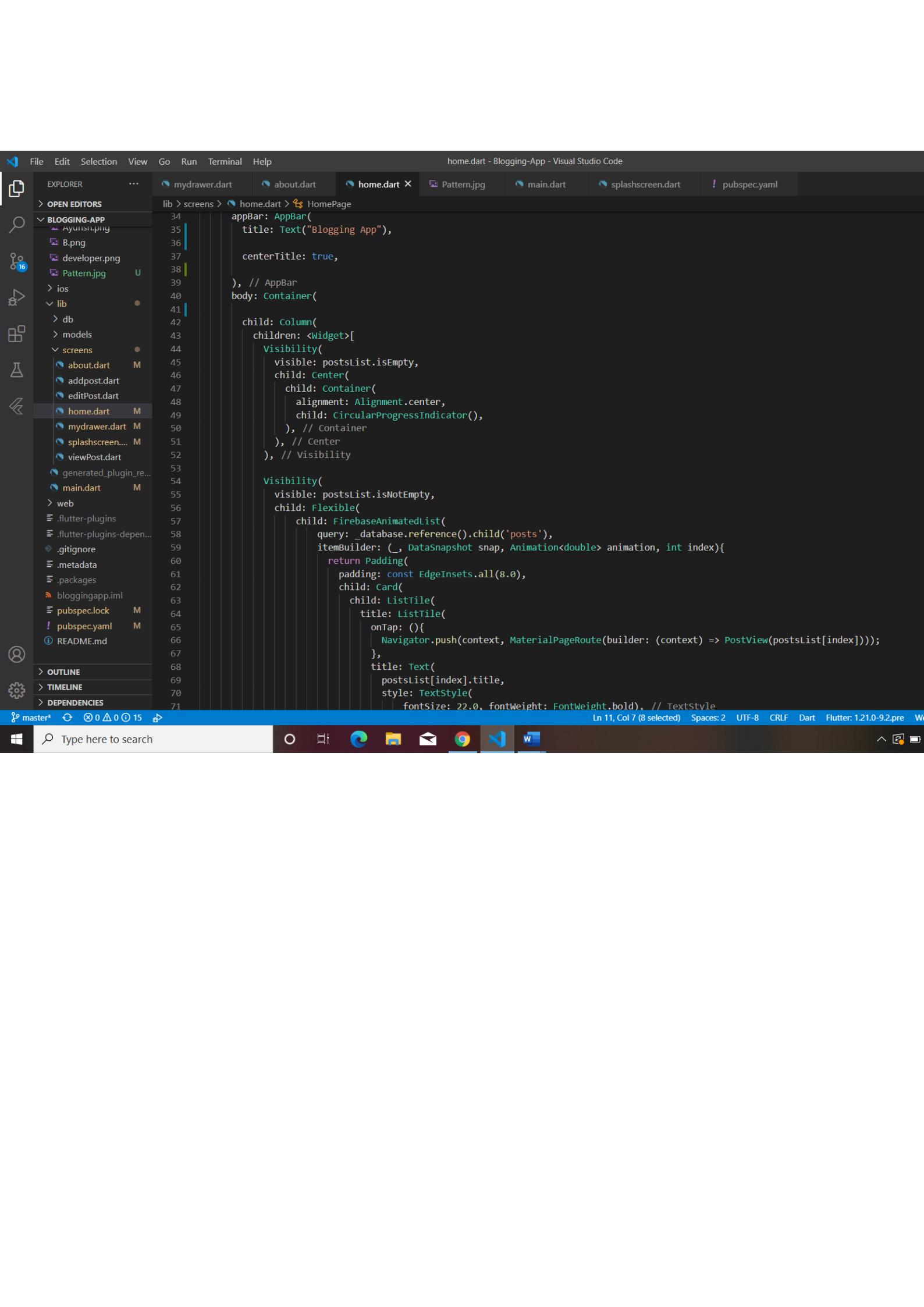
File name: home.dart

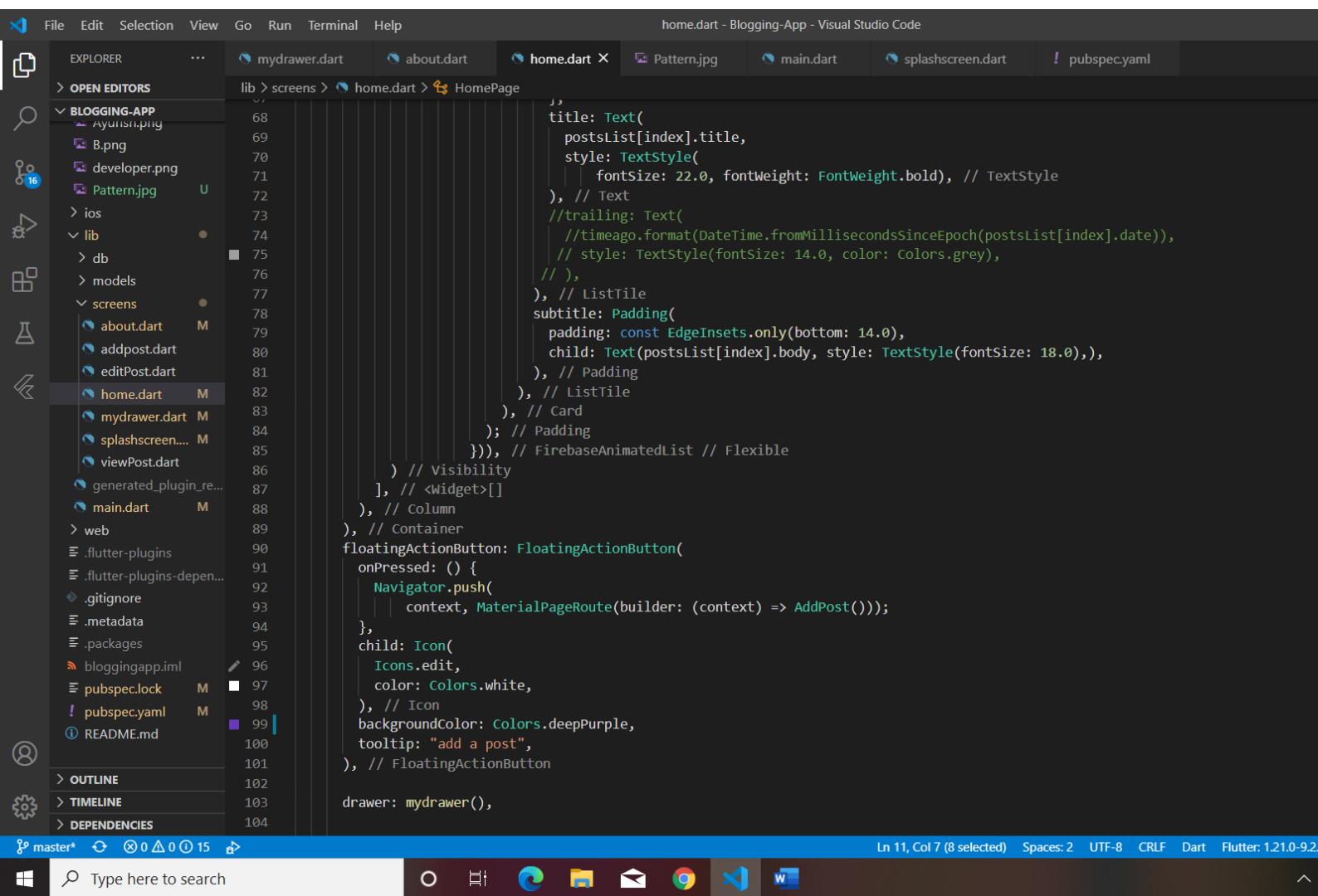
The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** home.dart - Blogging-App - Visual Studio Code
- Explorer View:** Shows the project structure:
 - BLOGGING-APP
 - Ayushin.png
 - B.png
 - developer.png
 - Pattern.jpg
 - ios
 - lib
 - db
 - models
 - screens
 - about.dart
 - addpost.dart
 - editPost.dart
 - home.dart (selected)
 - mydrawer.dart
 - splashscreen....
 - viewPost.dart
 - generated_plugin_re...
 - main.dart
 - web
 - .flutter-plugins
 - .flutter-plugins-depend...
 - .gitignore
 - .metadata
 - .packages
 - bloggingapp.iml
 - pubspec.lock
 - pubspec.yaml
 - README.md
- Editor View:** Displays the content of the 'home.dart' file:

```
1 import 'package:bloggingapp/models/post.dart';
2 import 'package:bloggingapp/screens/viewpost.dart';
3 import 'package:flutter/material.dart';
4 import 'addpost.dart';
5 import 'package:firebase_database/ui/firebase_animated_list.dart';
6 import 'package:firebase_database.firebaseio_database.dart';
7 import 'package:firebase_core/firebase_core.dart';
8 import 'package:bloggingapp/screens/mydrawer.dart';
//import 'package:timeago/timeago.dart';

9
10 class HomePage extends StatefulWidget {
11   @override
12     _HomePageState createState() => _HomePageState();
13 }
14
15
16 class _HomePageState extends State<HomePage> {
17   FirebaseDatabase _database = FirebaseDatabase.instance;
18   String nodeName = "posts";
19   List<Post> postsList = <Post>[];
20
21
22   @override
23   void initState() {
24     _database.reference().child(nodeName).onChildAdded.listen(_childAdded);
25     _database.reference().child(nodeName).onChildRemoved.listen(_childRemoves);
26     _database.reference().child(nodeName).onChildChanged.listen(_childChanged);
27   }
28
29
30   @override
31   Widget build(BuildContext context) {
32     return Scaffold(
33       appBar: AppBar(
34         title: Text("Blogging App"),
35       ),
36       centerTitle: true,
37     );
38 }
```
- Bottom Status Bar:** master*, 0 ▲ 0 ⏪ 15 ⏩ Ln 11, Col 7 (8 selected) Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0-9.2.pre
- Taskbar:** Shows icons for File Explorer, Search, Task List, Recent Files, and others.

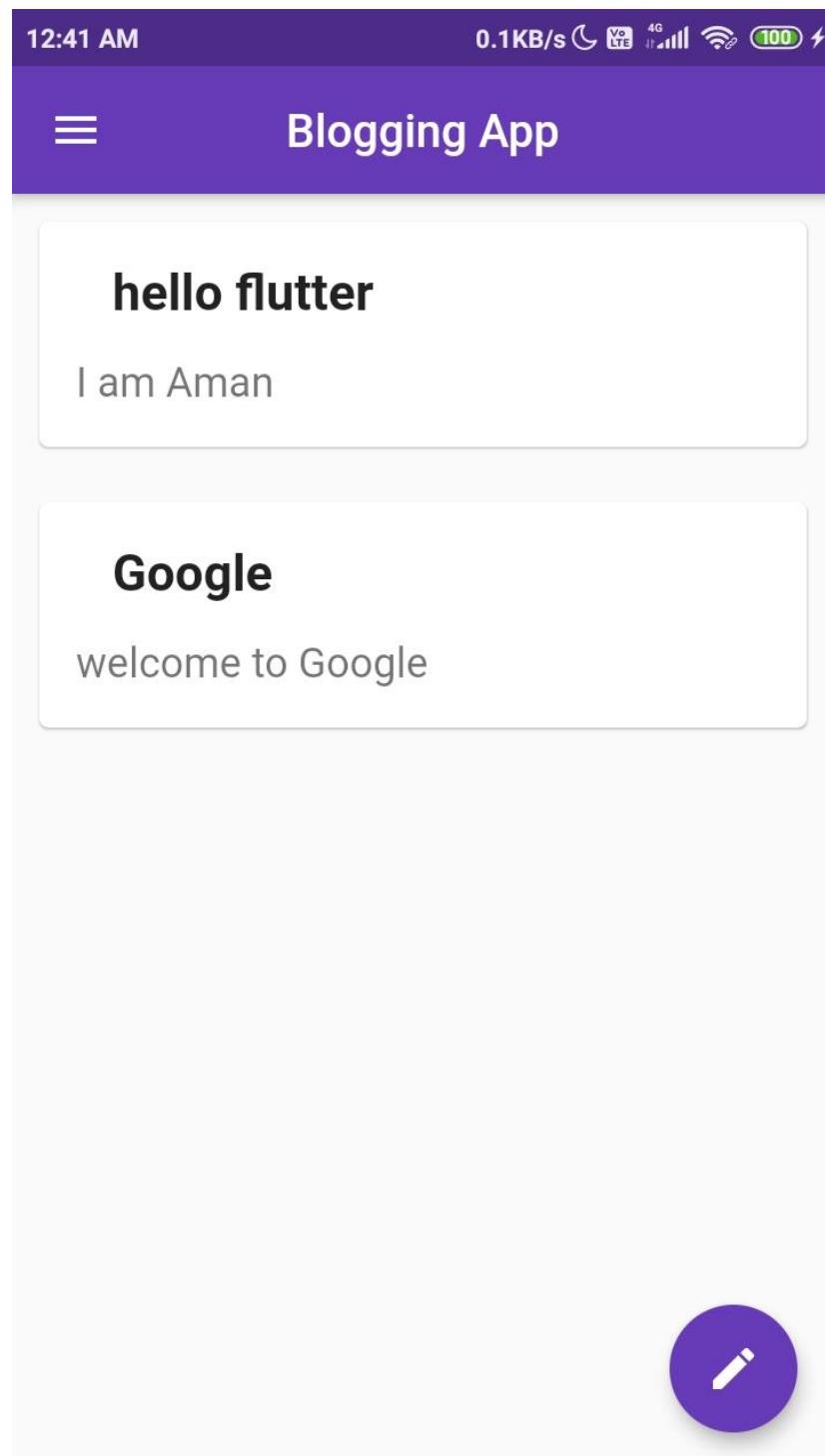




The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** home.dart - Blogging-App - Visual Studio Code
- Explorer View:** Shows the project structure:
 - BLOGGING-APP
 - Ayushn.png
 - B.png
 - developer.png
 - Pattern.jpg
 - ios
 - lib
 - db
 - models
 - screens
 - about.dart
 - addpost.dart
 - editPost.dart
 - home.dart
 - mydrawer.dart
 - splashscreen....
 - viewPost.dart
 - generated_plugin_re...
 - main.dart
 - web
 - .flutter-plugins
 - .flutter-plugins-depend...
 - .gitignore
 - .metadata
 - .packages
 - bloggingapp.iml
 - pubspec.lock
 - pubspec.yaml
 - README.md
- Code Editor:** The current file is home.dart, showing code related to a HomePage. The code includes methods for handling child additions, removals, and changes in a list of posts.
- Status Bar:** master*, 0 △ 0 ① 15, Ln 11, Col 7 (8 selected), Spaces: 2, UTF-8, CRLF.
- Bottom Bar:** Includes icons for Windows Taskbar, Search, Task View, Start, Taskbar View, Mail, Google Chrome, and Visual Studio Code.

Output:



Filename: addpost.dart

File Edit Selection View Go Run Terminal Help addpost.dart - Blogging-App - Visual Studio Code

EXPLORER OPEN EDITORS

BLOGGING-APP Ayutism.png B.png developer.png Pattern.jpg

ios lib db models screens about.dart addpost.dart editPost.dart home.dart mydrawer.dart splashscreen.... viewPost.dart generated_plugin_re... main.dart

web .flutter-plugins .flutter-plugins-depen... .gitignore .metadata .packages bloggingapp.iml pubspec.lock pubspec.yaml README.md

OUTLINE TIMELINE DEPENDENCIES

addpost.dart

```
36     if(val.isEmpty ){
37         return "title field cant be empty";
38     }else if(val.length > 16){
39         return "title cannot have more than 16 characters ";
40     }
41 },
42 ),
43 ),
44 ),
45 ),
46 ),
47 ),
48 ),
49 ),
50 ),
51 ),
52 ),
53 ),
54 ),
55 ),
56 ),
57 ),
58 ),
59 ),
60 ),
61 ],
62 ],
63 ],
64 ],
65 ],
66 //,
67 },
68 },
69 ),
70 ),
71 ),
72 },
73 ); // Scaffold
```

Ln 1, Col 1 Spaces: 2 UTF-8

File Edit Selection View Go Run Terminal Help addpost.dart - Blogging-App - Visual Studio Code

EXPLORER OPEN EDITORS BLOGGING-APP Ayurishi.png B.png developer.png Pattern.jpg U ios lib db models screens about.dart M addpost.dart editPost.dart home.dart M mydrawer.dart M splashscreen... M viewPost.dart generated_plugin_re... main.dart M web .flutter-plugins .flutter-plugins-depen... .gitignore .metadata .packages bloggingapp.iml pubspec.lock M pubspec.yaml M README.md

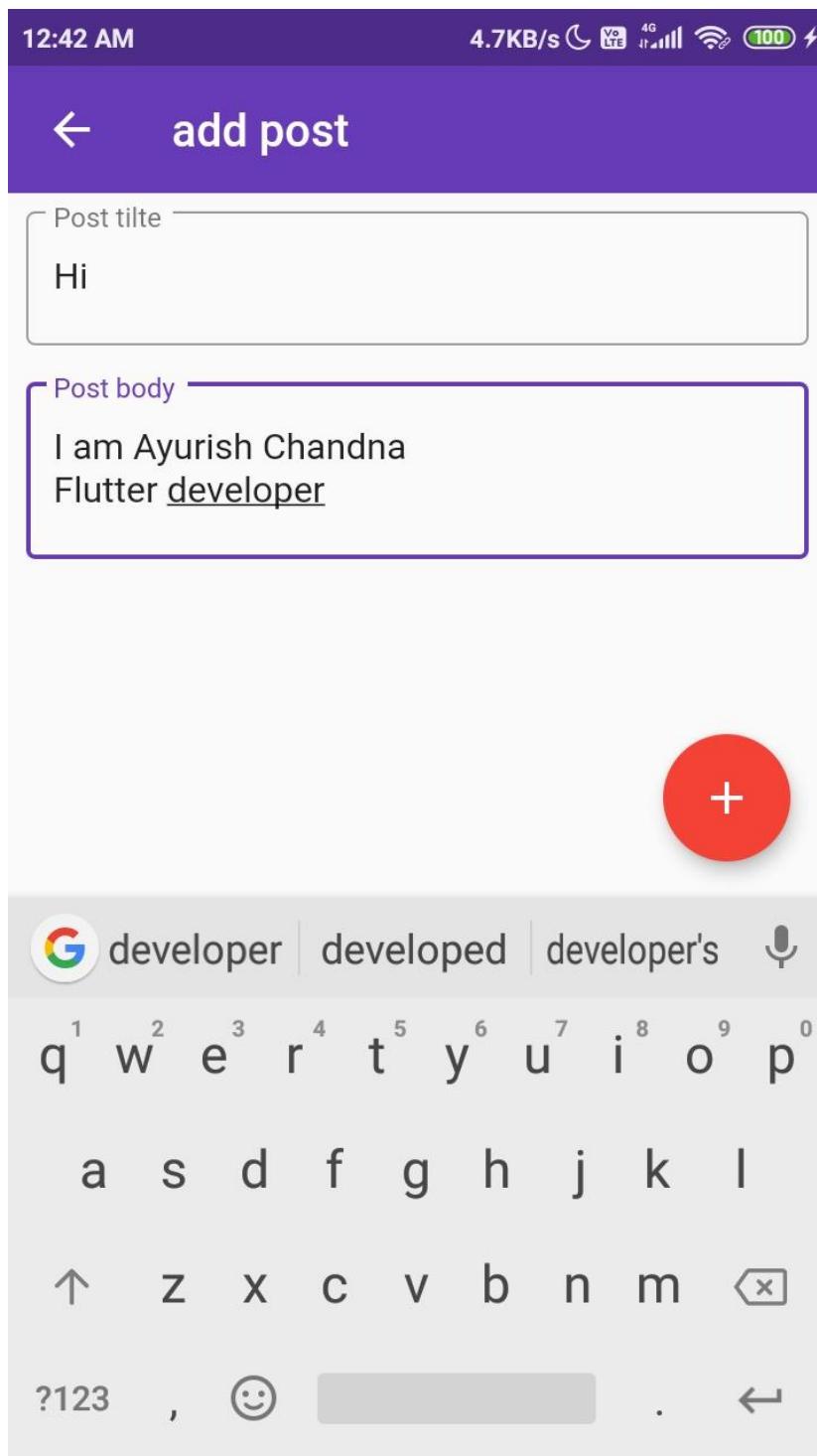
lib > screens > addpost.dart > ...

```
57     }
58     },
59     ), // TextFormField
60     ), // Padding
61     ], // <Widget>[]
62     )), // ListView // Form
63     floatingActionButton: FloatingActionButton(onPressed: (){
64       insertPost();
65       Navigator.pop(context);
66       // Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) => HomePage()));
67     },
68     child: Icon(Icons.add, color: Colors.white,),
69     backgroundColor: Colors.red,
70     tooltip: "add a post",), // FloatingActionButton
71   ); // Scaffold
72 }

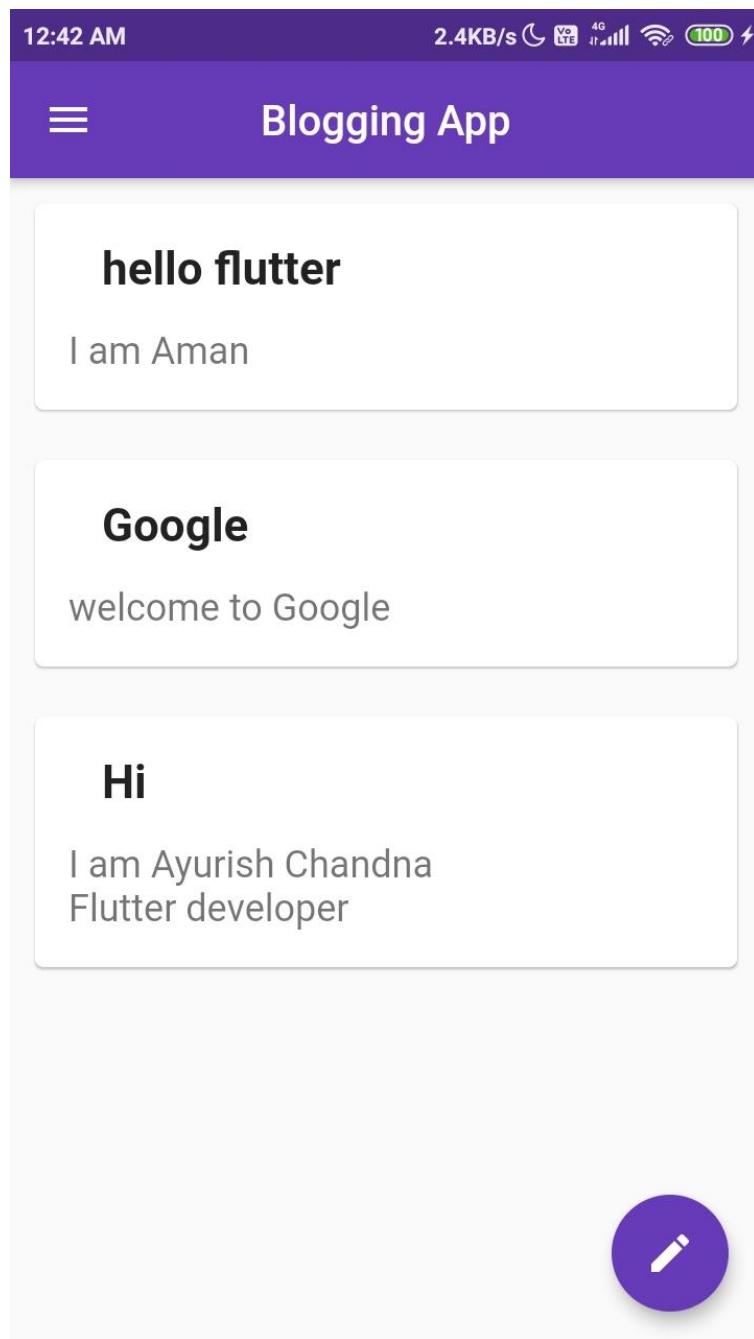
void insertPost() {
  final FormState form = formkey.currentState;
  if(form.validate()){
    form.save();
    form.reset();
    post.date = DateTime.now().millisecondsSinceEpoch;
    PostService postService = PostService(post.toMap());
    postService.addPost();
  }
}
```

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0

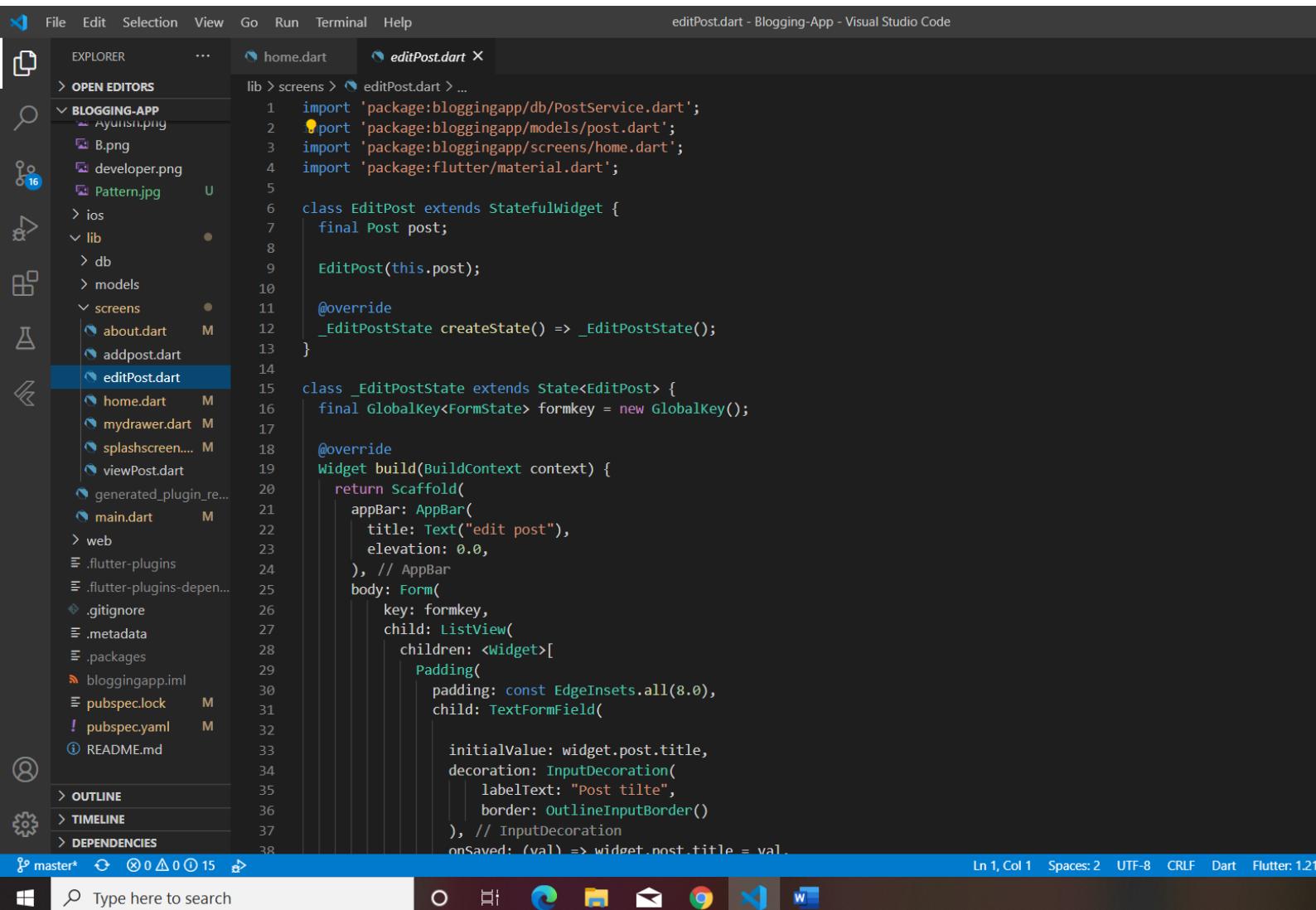
Output:



The new added post:



File name: edit.dart



```
File Edit Selection View Go Run Terminal Help
editPost.dart - Blogging-App - Visual Studio Code

EXPLORER ... home.dart editPost.dart ...
OPEN EDITORS
BLOGGING-APP Ayurish.png B.png developer.png Pattern.jpg
ios lib db models screens about.dart addpost.dart editPost.dart
home.dart mydrawer.dart splashscreen.... viewPost.dart
generated_plugin_re...
main.dart
web .flutter-plugins .flutter-plugins-depen...
.gitignore .metadata .packages bloggingapp.iml pubspec.lock pubspec.yaml README.md
OUTLINE
TIMELINE
DEPENDENCIES

lib > screens > editPost.dart > ...
1 import 'package:bloggingapp/db/PostService.dart';
2 import 'package:bloggingapp/models/post.dart';
3 import 'package:bloggingapp/screens/home.dart';
4 import 'package:flutter/material.dart';
5
6 class EditPost extends StatefulWidget {
7   final Post post;
8
9   EditPost(this.post);
10
11   @override
12   _EditPostState createState() => _EditPostState();
13 }
14
15 class _EditPostState extends State<EditPost> {
16   final GlobalKey<FormState> formKey = new GlobalKey();
17
18   @override
19   Widget build(BuildContext context) {
20     return Scaffold(
21       appBar: AppBar(
22         title: Text("edit post"),
23         elevation: 0.0,
24     ), // AppBar
25     body: Form(
26       key: formKey,
27       child: ListView(
28         children: <Widget>[
29           Padding(
30             padding: const EdgeInsets.all(8.0),
31             child: TextFormField(
32               initialValue: widget.post.title,
33               decoration: InputDecoration(
34                 labelText: "Post tilte",
35                 border: OutlineInputBorder()
36               ), // InputDecoration
37               onSaved: (val) => widget.post.title = val,
38             ),
39           ],
40         ),
41       ),
42     );
43   }
44 }

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF Dart Flutter: 121
```

File Edit Selection View Go Run Terminal Help editPost.dart - Blogging-App - Visual Studio Code

EXPLORER ... home.dart editPost.dart X

> OPEN EDITORS

> BLOGGING-APP

- Ayush.png
- B.png
- developer.png
- Pattern.jpg

> ios

> lib

- db
- models
- > screens
- about.dart M
- addpost.dart
- editPost.dart
- home.dart M
- mydrawer.dart M
- splashscreen.... M
- viewPost.dart

generated_plugin_re...

main.dart M

> web

.flutter-plugins

.flutter-plugins-depen...

.gitignore

.metadata

.packages

bloggingapp.iml

pubspec.lock M

pubspec.yaml M

README.md

> OUTLINE

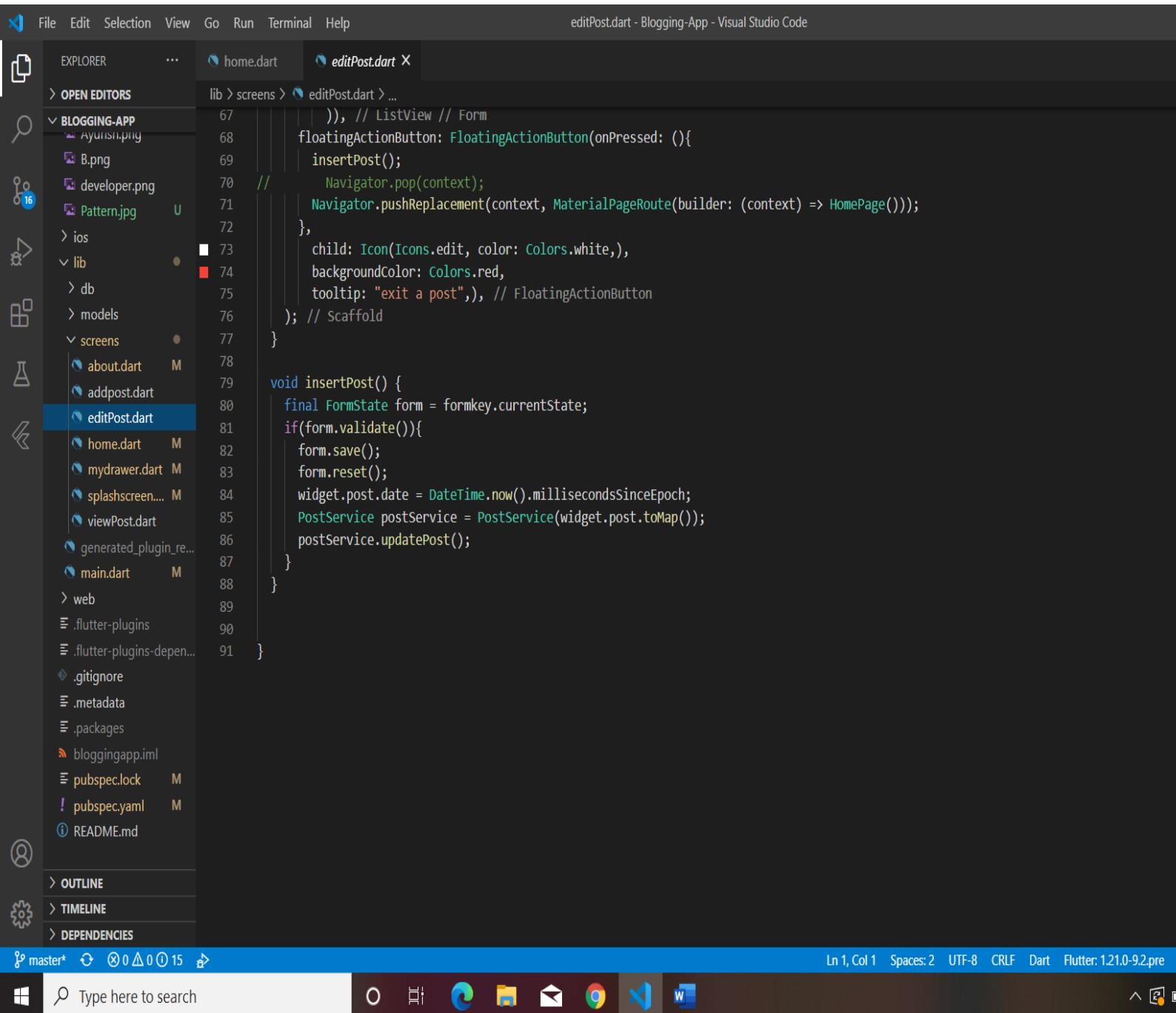
> TIMELINE

> DEPENDENCIES

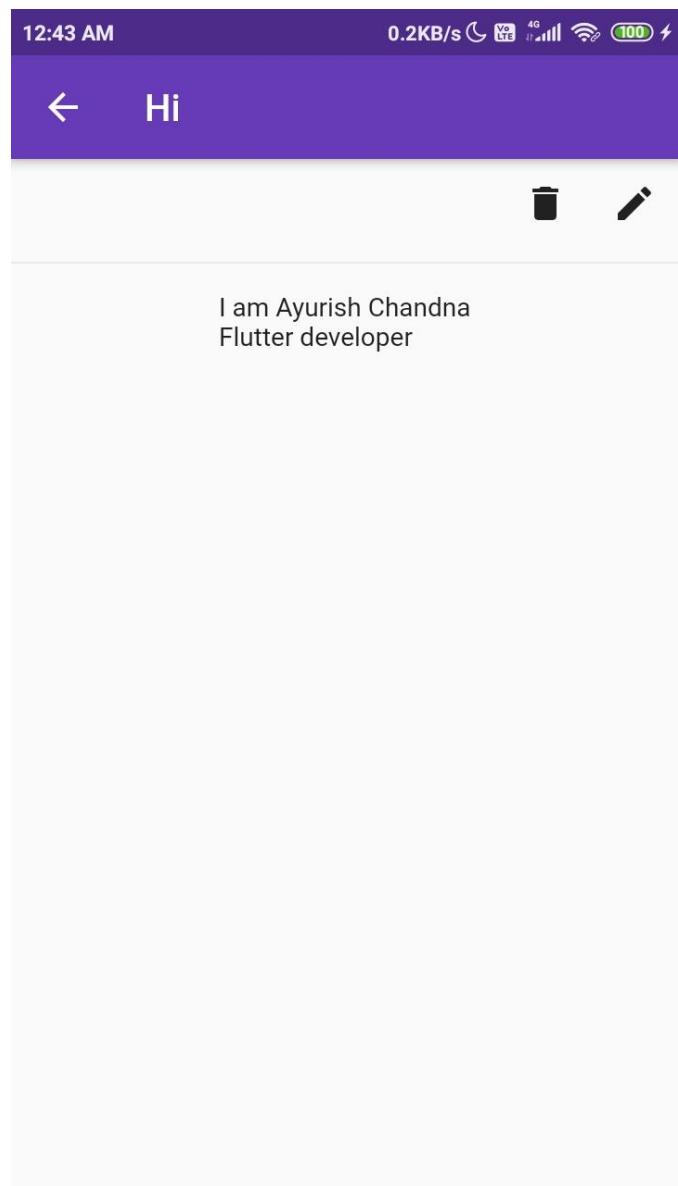
36 border: OutlineInputBorder()
37), // InputDecoration
38 onSaved: (val) => widget.post.title = val,
39 validator: (val){
40 if(val.isEmpty){
41 return "title field cant be empty";
42 }else if(val.length > 16){
43 return "title cannot have more than 16 characters ";
44 }
45 },
46), // TextFormField
47), // Padding
48 Padding(
49 padding: const EdgeInsets.all(8.0),
50 child: TextFormField(
51 keyboardType: TextInputType.multiline,
52 maxLines: null,
53 initialValue: widget.post.body,
54 decoration: InputDecoration(
55 labelText: "Post body",
56 border: OutlineInputBorder()
57), // InputDecoration
58 onSaved: (val) => widget.post.body = val,
59 validator: (val){
60 if(val.isEmpty){
61 return "body field cant be empty";
62 }
63 },
64), // TextFormField
65), // Padding
66], // <Widget>[]
67)), // ListView // Form
68 floatingActionButton: FloatingActionButton(onPressed: (){
69 insertPost();
70 // Navigator.pop(context);
71 Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) => HomePage()));
72 },
73 child: Icon(Icons.edit, color: Colors.white,),

Ln 1, Col 1 Spaces: 2 CRLF Dart Flutter: 1.21.0-9.2.pre

Type here to search



Output:



12:43 AM

14.1KB/s LTE 4G WiFi 100%



edit post

Post title

Hi

Post body

I am Ayurish Chandna
Flutter developer



Hi

His

High



q¹ w² e³ r⁴ t⁵ y⁶ u⁷ i⁸ o⁹ p⁰

a s d f g h j k l

↑ z x c v b n m ✕

?123

,



.



File name: viewpost.dart

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** viewPost.dart - Blogging-App - Visual Studio Code
- Left Sidebar (Explorer):**
 - OPEN EDITORS
 - BLOGGING-APP
 - Amani.png.jpg
 - Ayurish.jpg
 - Ayurish.png
 - B.png
 - developer.png
 - Pattern.jpg
 - ios
 - lib
 - db
 - models
 - screens
 - about.dart
 - addpost.dart
 - editPost.dart
 - home.dart
 - mydrawer.dart
 - splashscreen....
 - viewPost.dart
 - generated_plugin_re...
 - main.dart
 - web
 - .flutter-plugins
 - .flutter-plugins-depen...
 - .gitignore
 - .metadata
 - .packages
 - bloggingapp.iml
 - pubspec.lock
 - pubspec.yaml
- OUTLINE
- TIMELINE
- DEPENDENCIES

- Editor Area:** Displays the code for `viewPost.dart`. The code is a Flutter StatelessWidget named `PostView` that takes a `Post` object as a parameter. It overrides the `createState` method to return a `_PostViewState` object. The `_PostViewState` class extends `State<PostView>` and builds a `Scaffold` with an `AppBar` containing the post title and a `Column` body. The `Column` contains a `Row` with an `Expanded` child that has padding and a `Text` child showing the post date in a specific format. It also contains a `IconButton` for deleting the post.
- Bottom Status Bar:** master*, 0△0 ① 15, Ln 1, Col 1, Spaces: 2, UTF-8, CRLF, Dart, Flutter.
- Bottom Icons:** Windows Start, Task View, Edge, File Explorer, Mail, Google Chrome, Visual Studio Code, Word.

File Edit Selection View Go Run Terminal Help viewPost.dart - Blogging-App - Visual Studio Code

EXPLORER ... home.dart viewPost.dart X

BLOGGING-APP

- Alian.png.jpg
- Ayurish.jpg
- Ayurish.png
- B.png
- developer.png
- Pattern.jpg

ios

lib

- db
- models
- screens

 - about.dart M
 - addpost.dart
 - editPost.dart
 - home.dart M
 - mydrawer.dart M
 - splashscreen.... M

viewPost.dart

generated_plugin_re... M

main.dart M

web

- .flutter-plugins
- .flutter-plugins-depen...
- .gitignore
- .metadata
- .packages
- bloggingapp.iml
- pubspec.lock M
- pubspec.yaml M

OUTLINE

TIMELINE

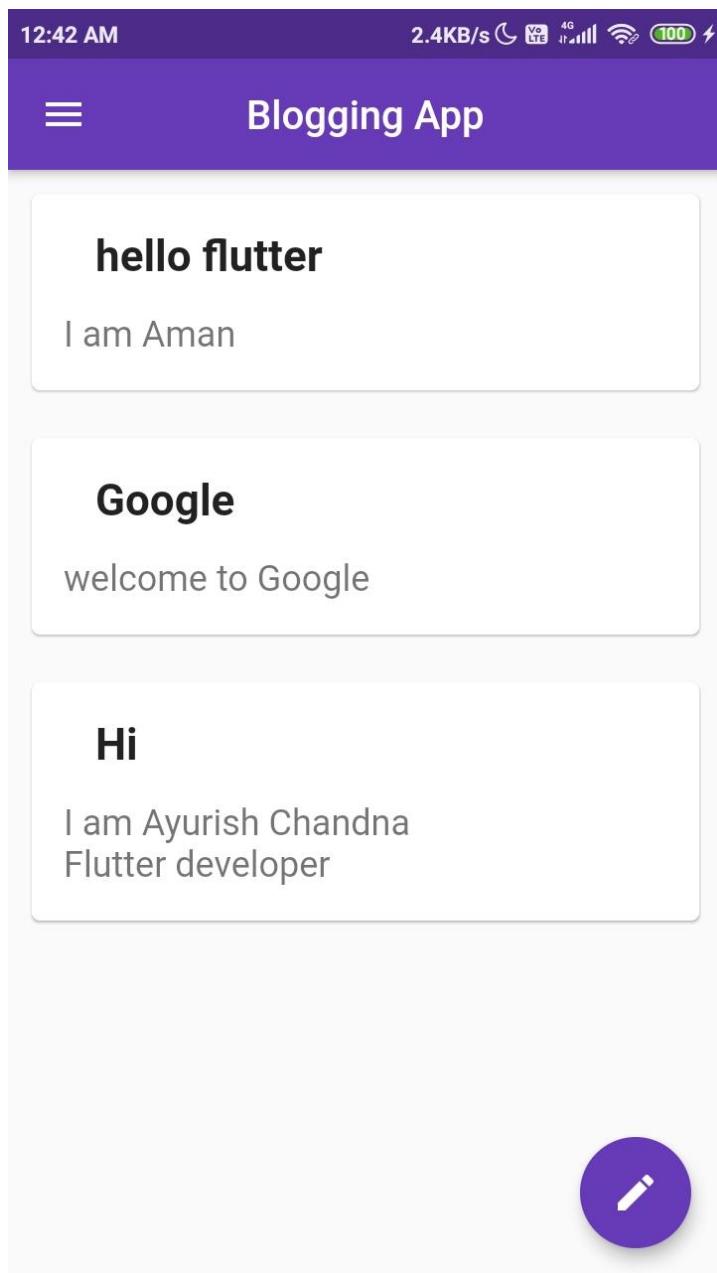
DEPENDENCIES

```
31 //child: Text(
32 //timeago.format(DateTime.fromMillisecondsSinceEpoch(widget.post.date)),
33 // style: TextStyle(fontSize: 14.0, color: Colors.grey),
34 //),
35 ), // Padding // Expanded
36 IconButton(icon: Icon(Icons.delete),
37 onPressed: (){
38 PostService postService = PostService(widget.post.toMap());
39 postService.deletePost();
40 Navigator.pop(context);
41 },
42 ), // IconButton
43 IconButton(icon: Icon(Icons.edit),
44 onPressed: (){
45 Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) => EditPost(widget.post)));
46 },
47 ), // IconButton
48 ],
49 ),
50 Padding(
51 padding: const EdgeInsets.all(8.0),
52 child: Text(widget.post.body),
53 ),
54 ],
55 ),
56 );
57 }
58 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0-9.2

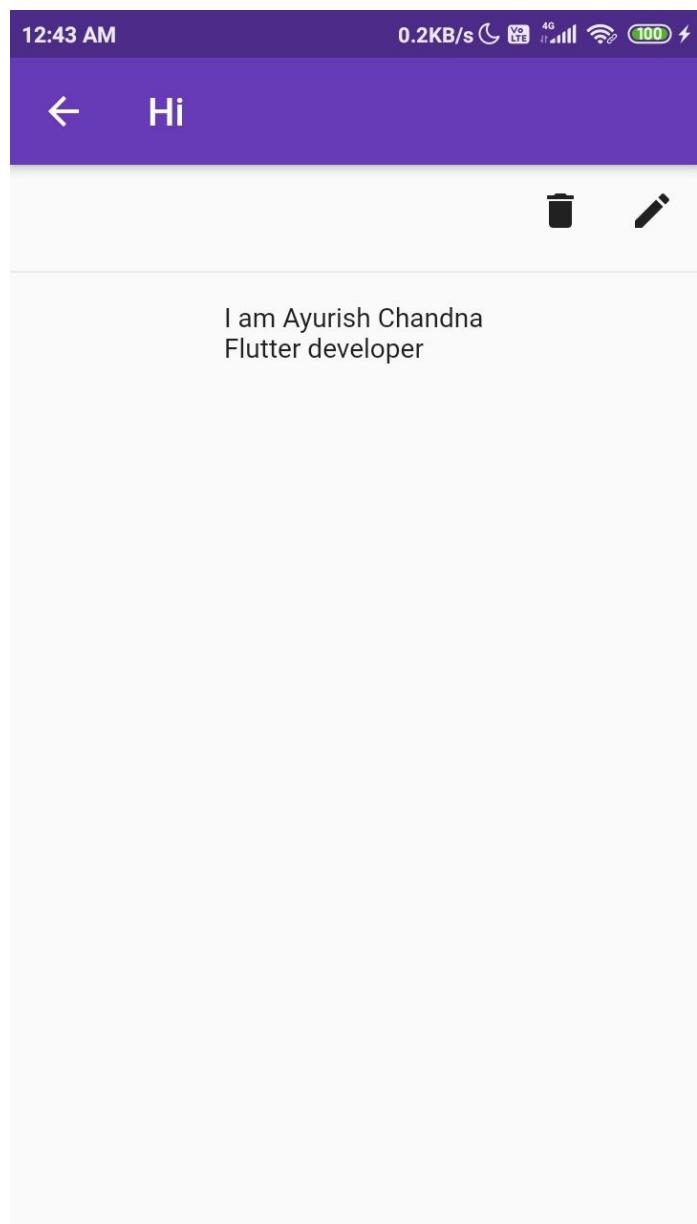
Type here to search

Output:

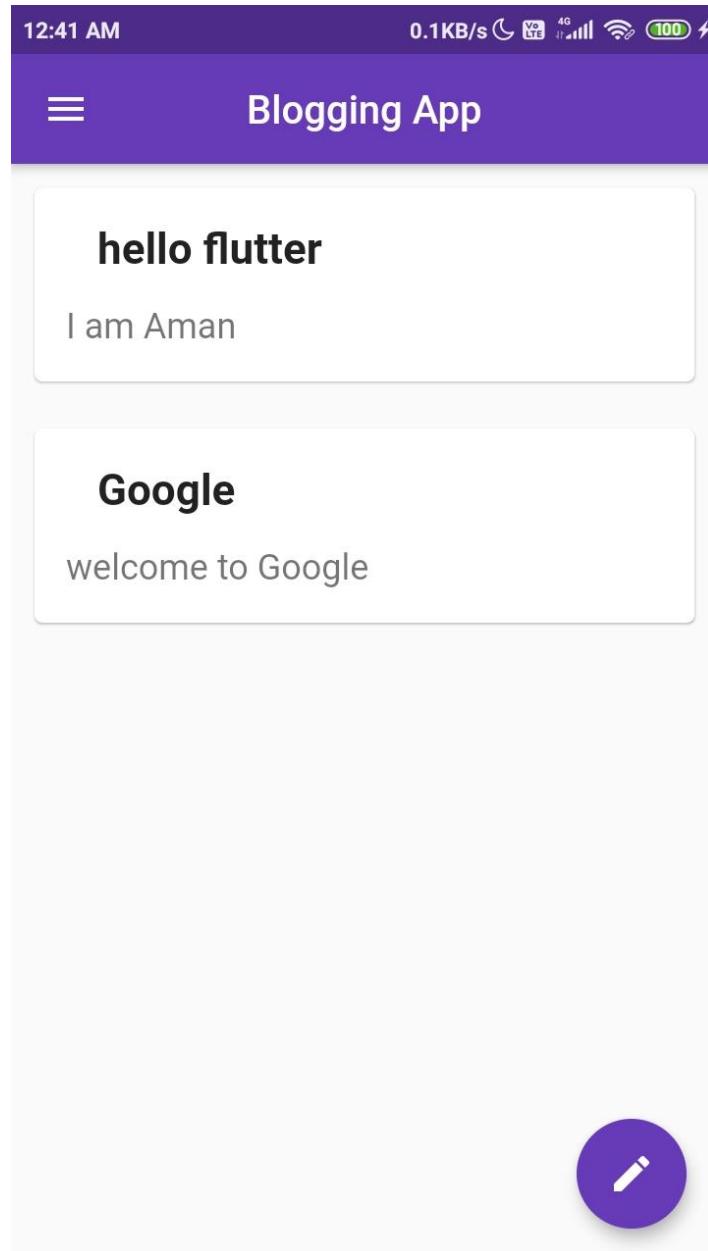


Delete post Output:

Step 1: Click on delete button



Step 2: Post deleted



File name: mydrawer.dart

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** mydrawer.dart - Blogging-App - Visual Studio Code
- Explorer:** Shows the project structure under 'BLOGGING-APP'. Files listed include: .AndroidManifest.xml, Ayurish.jpg, Ayurish.png, B.png, developer.png, Pattern.jpg, about.dart, addpost.dart, editPost.dart, home.dart, mydrawer.dart (selected), splashscreen...., viewPost.dart, generated_plugin_re..., main.dart, and web/.flutter-plugins.
- Editor:** The 'mydrawer.dart' file is open. The code defines a StatelessWidget named 'mydrawer' that returns a Drawer. The Drawer contains a ListView with a DrawerHeader and a Container. The DrawerHeader has a padding of EdgeInsets.symmetric(). The Container has a padding of EdgeInsets.symmetric(vertical: 0.2). The Container's child is a UserAccountsDrawerHeader, which requires accountEmail, accountName, and currentAccountPicture parameters. It also includes a backgroundImage set to AssetImage('images/developer.png'). The Container's decoration is a BoxDecoration with a color condition based on Theme.brightness. The Container's child is a BoxDecoration, which is a DrawerHeader.
- Bottom Status Bar:** master*, 0△0①15, Ln 17, Col 19, Spaces: 2, UTF-8, CRLF, Dart, Flutter: 1.21.0-92.pre, Chrome.
- Taskbar:** Shows icons for File Explorer, Find, Home, Task View, Mail, Google Chrome, Visual Studio Code, and Windows Start.

```
lib > screens > mydrawer.dart > build
1 import 'package:flutter/material.dart';
2 import 'package:font_awesome_flutter/font_awesome_flutter.dart';
3 import 'package:bloggingapp/screens/about.dart';
4 import 'package:dynamic_theme/dynamic_theme.dart';
5 class mydrawer extends StatelessWidget {
6   @override
7   Widget build(BuildContext context) {
8     return Drawer(
9       child: ListView(
10         padding: EdgeInsets.zero,
11         children: [
12           DrawerHeader(
13             padding: EdgeInsets.symmetric(),
14             child: Container(
15               padding: EdgeInsets.symmetric(vertical: 0.2),
16               child: UserAccountsDrawerHeader(
17                 //use the actual details and fill here
18                 accountEmail: Text(""),
19                 accountName: Text("Blogging App", style: TextStyle(fontsize: 20),),
20                 currentAccountPicture: CircleAvatar(
21                   backgroundImage: AssetImage('images/developer.png'),
22                 ),
23                 // CircleAvatar
24               ),
25               decoration: BoxDecoration(
26                 color: (Theme.of(context).brightness==Brightness.dark)?Colors.black26:Colors.deepPurple,
27               ),
28             ),
29           ), // UserAccountsDrawerHeader
30           Container(
31             decoration: BoxDecoration(
32               color: (Theme.of(context).brightness==Brightness.dark)?Colors.black26:Colors.deepPurple,
33             ),
34           ),
35           BoxDecoration(
36             color: (Theme.of(context).brightness==Brightness.dark)?Colors.black26:Colors.deepPurple,
37           ),
38         ],
39       ),
40     );
41   }
42 }
```

File Edit Selection View Go Run Terminal Help mydrawer.dart - Blogging-App - Visual Studio Code

EXPLORER OPEN EDITORS

BLOGGING-APP

- Aman.png.jpg
- Ayurish.jpg
- Ayurish.png
- B.png
- developer.png
- Pattern.jpg

ios

lib

- db
- models
- screens

 - about.dart M
 - addpost.dart
 - editPost.dart
 - home.dart M
 - mydrawer.dart M
 - splashscreen.... M
 - viewPost.dart

generated_plugin_re...

main.dart M

web

.flutter-plugins

.flutter-plugins-depen...

.gitignore

.metadata

.packages

bloggingapp.iml

pubspec.lock M

! pubspec.yaml M

OUTLINE

TIMELINE

DEPENDENCIES

home.dart X mydrawer.dart X

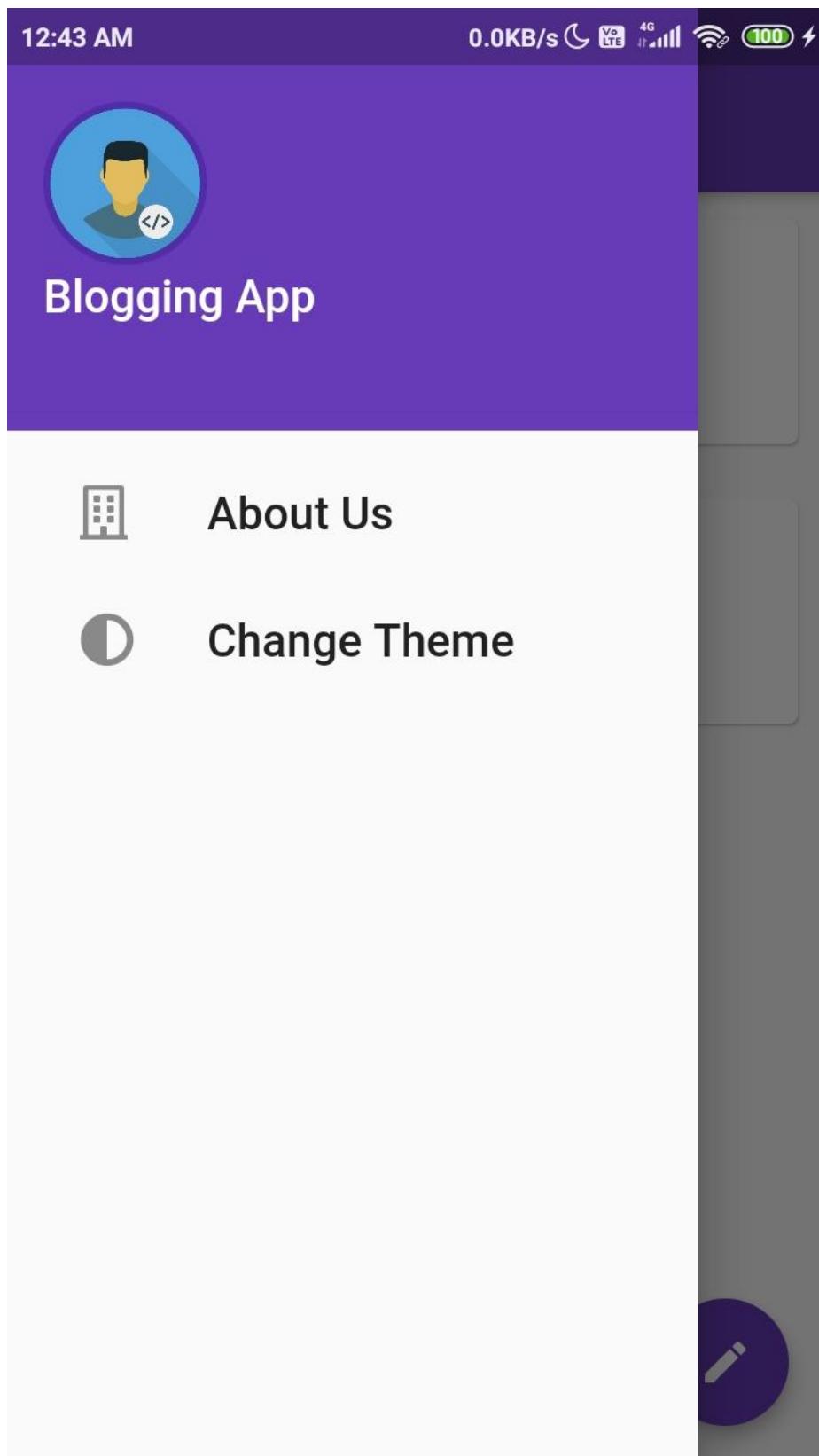
lib > screens > mydrawer.dart > mydrawer > build

```
35      ), // BoxDecoration
36      ), // DrawerHeader
37
38
39
40      FlatButton(
41        onPressed: () {
42          Navigator.pushNamed(context, '/about');
43        },
44        child: ListTile(
45          leading: FaIcon(FontAwesomeIcons.building),
46          title: Text('About Us', style: TextStyle(fontSize: 20),),
47        ), // ListTile
48      ), // FlatButton
49
50      FlatButton(
51        onPressed: () {
52          DynamicTheme.of(context).setBrightness(
53            Theme.of(context).brightness == Brightness.dark
54              ? Brightness.light
55              : Brightness.dark);
56        },
57        child: ListTile(
58          leading: FaIcon(FontAwesomeIcons.adjust),
59          title: Text('Change Theme', style: TextStyle(fontSize: 20),),
60        ), // ListTile
61      ), // FlatButton
62
63    ],
64  ), // ListView
65 ); // Drawer
66
67 }
68 }
69 }
70
71
72
```

Ln 17, Col 19 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.2

Type here to search

Output:



Filename: about.dart

The screenshot shows the Visual Studio Code interface with the 'about.dart' file open in the editor. The code defines a StatelessWidget named 'About' that displays a centered text and a circular image.

```
lib > screens > about.dart > About > build
1 import 'package:flutter/material.dart';
2 import 'package:bloggingapp/screens/mydrawer.dart';
3 class About extends StatelessWidget {
4     @override
5         Widget build(BuildContext context) {
6             return SafeArea(
7                 child: Scaffold(
8                     body: Center(
9                         child: Column(
10                            children: [
11                                SizedBox(
12                                    height: 25,
13                                ), // SizedBox
14                                Text(
15                                    'This app is developed by : - ',
16                                    style: TextStyle(
17                                        fontSize: 20,
18                                        color: Theme.of(context).brightness == Brightness.dark ? Colors.limeAccent[400] : Colors.deepOrange
19                                    ), // TextStyle
20                                    textAlign: TextAlign.center,
21                                ), // Text
22                                SizedBox(height: 10),
23                                CircleAvatar(
24                                    backgroundImage: AssetImage('images/Aman.png'),
25                                    radius: 87,
26                                ), // CircleAvatar
27                                SizedBox(height: 10),
28                                Text(
29                                    'Aman Kumar',
30                                    style: TextStyle(
31                                        fontSize: 20,
32                                        fontFamily: 'Comic',
33                                    ), // TextStyle
34                                ), // Text
35                                Text(
36                                    'Phone: +91943595XXXX',
37                                    style: TextStyle(
38                                        fontSize: 13,
```

File Edit Selection View Go Run Terminal Help about.dart - Blogging-App - Visual Studio Code

EXPLORER OPEN EDITORS

BLOGGING-APP

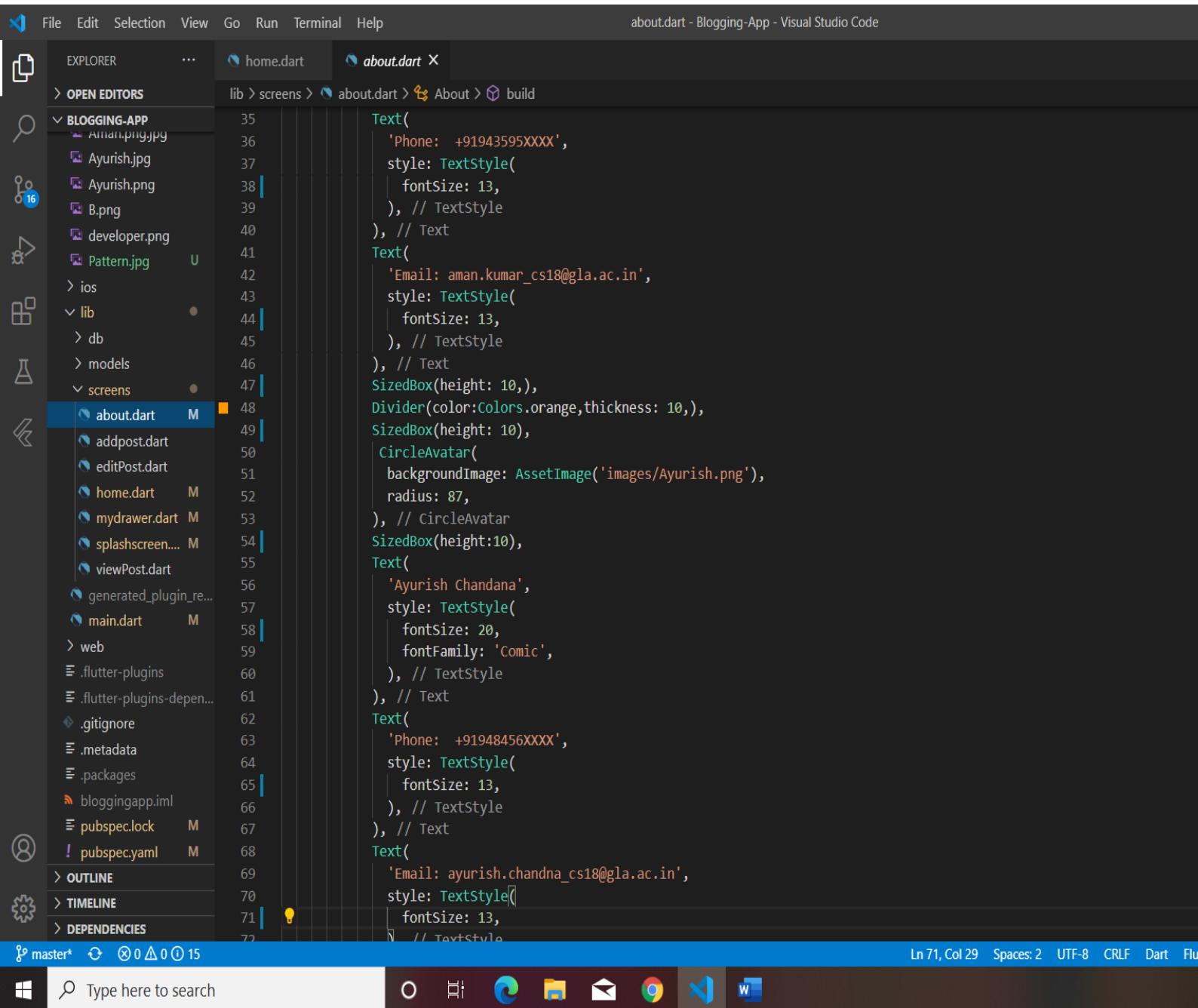
- Aman.jpg
- Ayurish.jpg
- Ayurish.png
- B.png
- developer.png
- Pattern.jpg

ios lib db models screens

about.dart M addpost.dart editPost.dart home.dart M mydrawer.dart M splashscreen.... M viewPost.dart generated_plugin_re... main.dart M web flutter-plugins flutter-plugins-depen... .gitignore .metadata .packages bloggingapp.iml pubspec.lock M pubspec.yaml M OUTLINE TIMELINE DEPENDENCIES

Ln 71, Col 29 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0-9.2.pre Chrome

Type here to search



File Edit Selection View Go Run Terminal Help about.dart - Blogging-App - Visual Studio Code

EXPLORER OPEN EDITORS

BLOGGING-APP

- Aman.png.jpg
- Ayurish.jpg
- Ayurish.png
- B.png
- developer.png
- Pattern.jpg

ios

lib

- db
- models
- screens

about.dart M

addpost.dart

editPost.dart

home.dart M

mydrawer.dart M

splashscreen.... M

viewPost.dart

generated_plugin_re...

main.dart M

web

.flutter-plugins

.flutter-plugins-depen...

.gitignore

.metadata

.packages

bloggingapp.iml

pubspec.lock M

pubspec.yaml M

OUTLINE

TIMELINE

DEPENDENCIES

about.dart X

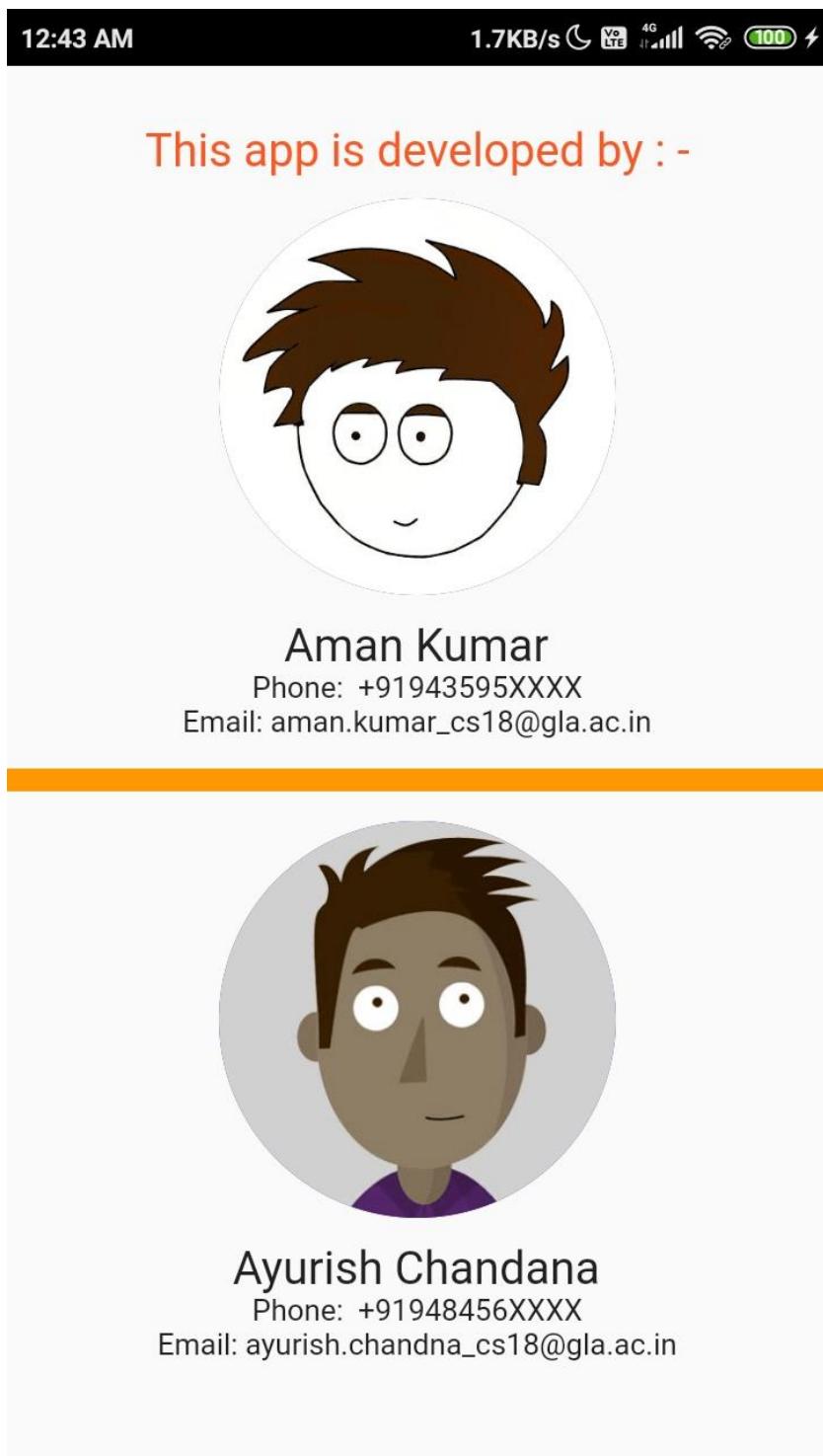
lib > screens > about.dart > About > build

```
63     'Phone: +91948456XXXX',
64     style: TextStyle(
65       fontSize: 13,
66     ), // TextStyle
67   ), // Text
68   Text(
69     'Email: ayurish.chandna_cs18@gl.a.ac.in',
70     style: TextStyle(
71       fontSize: 13,
72     ), // TextStyle
73   ), // Text
74   ],
75   ), // Column
76   ), // Center // Scaffold
77 ); // SafeArea
78 }
79 }
```

Ln 71, Col 29 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0-92.pre Chrome (web-javascript) 0132 24-11-2020

Type here to search

Output:



Filename: main.dart

The screenshot shows the Visual Studio Code interface with the main.dart file open in the editor. The code implements a DynamicTheme widget that provides a light theme by default, with a dark theme if the brightness is dark. It also defines routes for the application's pages.

```
File Edit Selection View Go Run Terminal Help main.dart - Blogging-App - Visual Studio Code

EXPLORER ... home.dart main.dart X
OPEN EDITORS > BLOGGING-APP lib > main.dart > Myapp > build
Ayurish.jpg Ayurish.png B.png developer.png Pattern.jpg
ios lib db models screens
about.dart addpost.dart editPost.dart home.dart mydrawer.dart splashscreen... viewPost.dart
generated_plugin_re...
main.dart M
web .flutter-plugins .flutter-plugins-dependencies .gitignore .metadata .packages bloggingapp.iml pubspec.lock pubspec.yaml
OUTLINE TIMELINE DEPENDENCIES

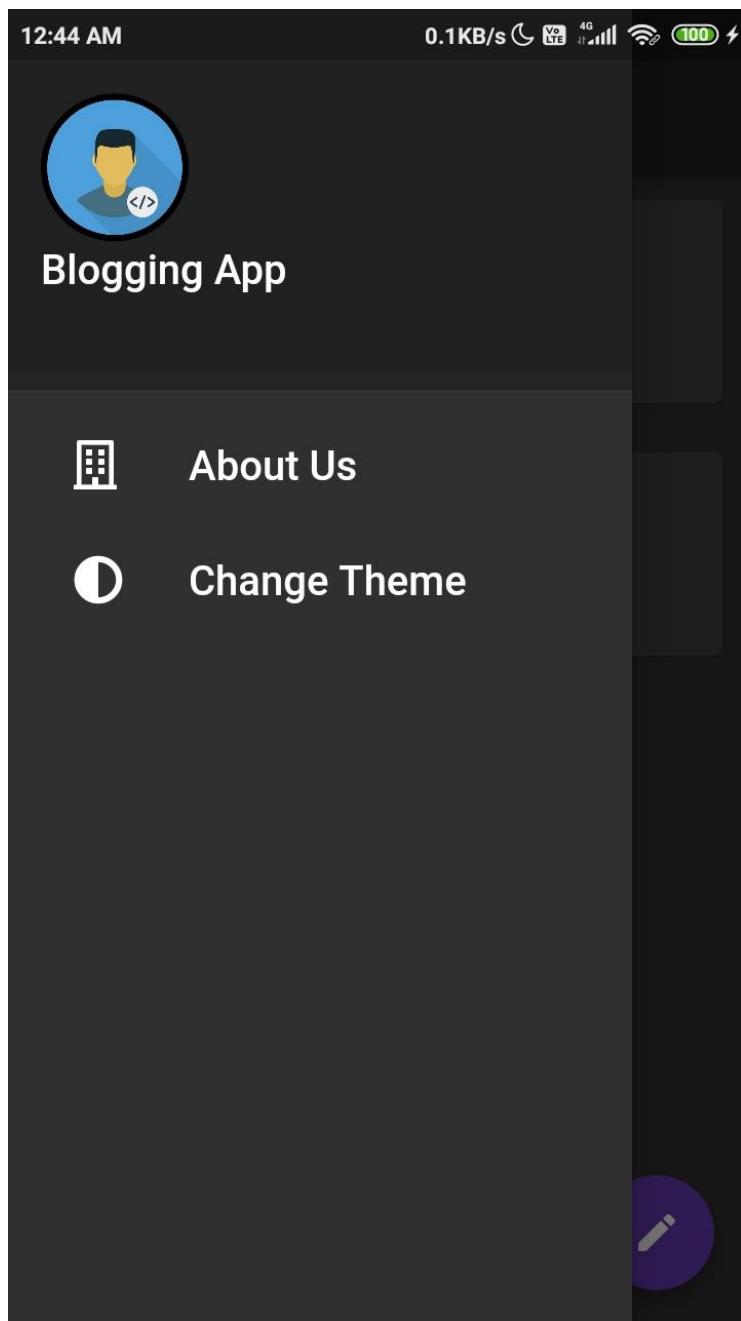
main.dart - Blogging-App - Visual Studio Code
15 const Myapp({Key key}) : super(key: key);
16
17 @override
18 Widget build(BuildContext context) {
19   SystemChrome.setPreferredOrientations([
20     DeviceOrientation.portraitUp,
21   ]);
22
23   return DynamicTheme(
24     defaultBrightness: Brightness.light,
25     data: (brightness) => ThemeData(
26       brightness: brightness,
27       primarySwatch: (brightness == Brightness.dark) ? Colors.lime : Colors.deepPurple,
28     ), // ThemeData
29     themedWidgetBuilder: (context, theme) {
30       return MaterialApp(
31         title: 'Blogging App',
32         debugShowCheckedModeBanner: false,
33         initialRoute: '/',
34         theme: theme,
35         //all the paths are described here
36
37         routes: {
38           '/': (context) => Splash2(),
39           '/about': (context) => About(),
40         },
41       ); // MaterialApp
42     }, // DynamicTheme
43   );
44 }
45
46
47 }
48
49 
```

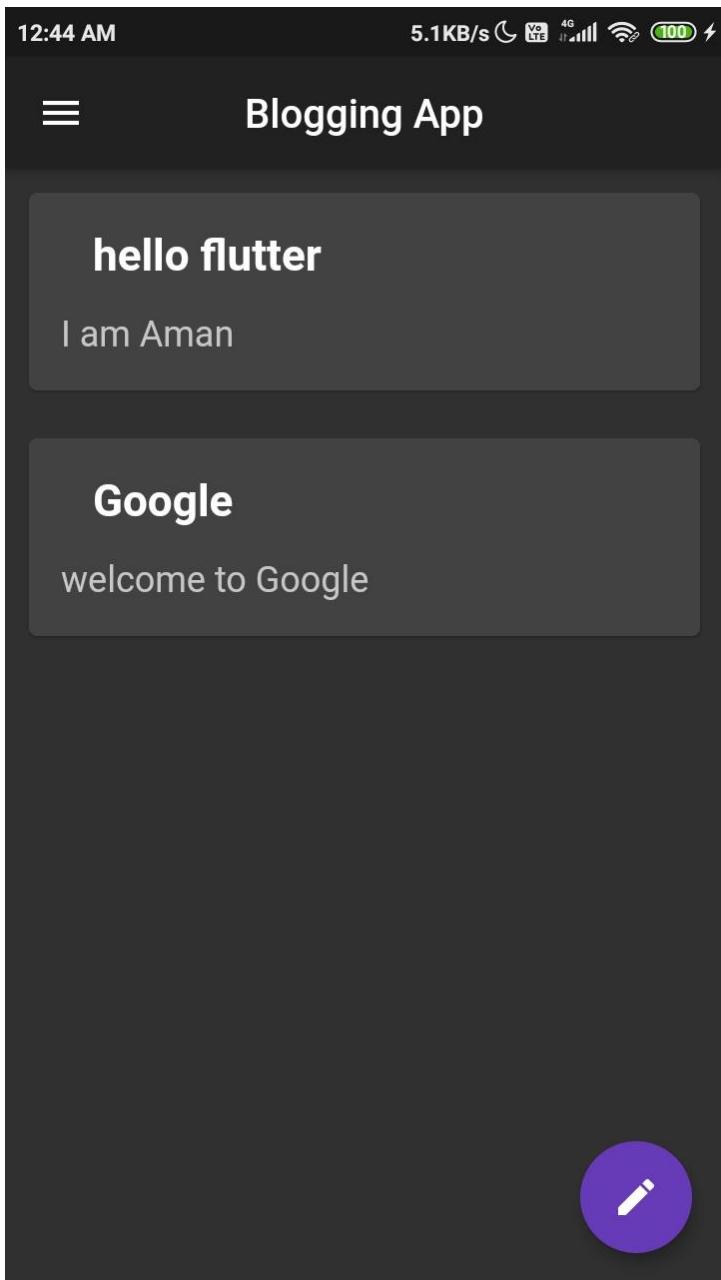
Ln 37, Col 11 Spaces: 2 UTF-8 CRLF Dart Flutter: 1.21.0-9.2.pre Chrome (web-javascript)

Type here to search

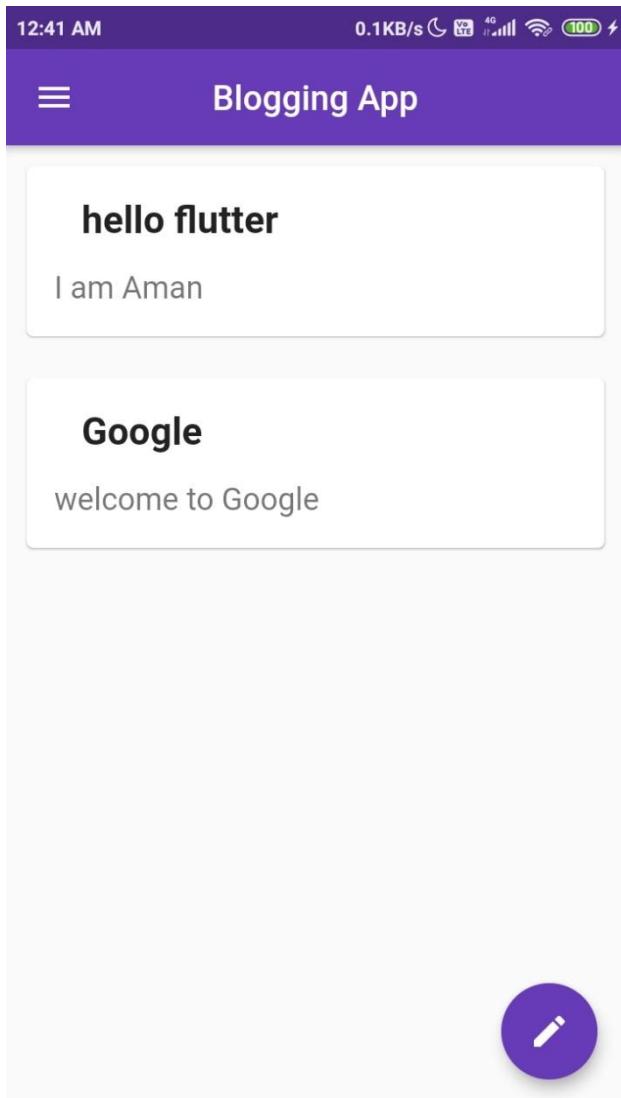
Output: Change Theme

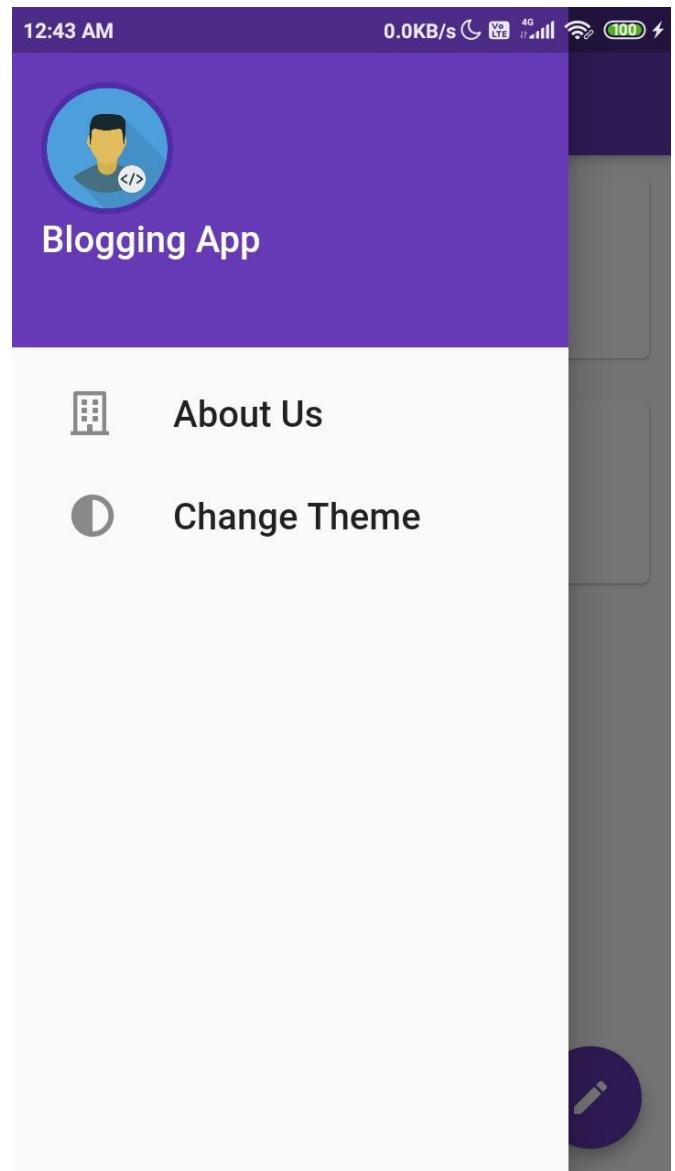
Dark mode





Light Mode





Firebase

Filename: post.dart

File Edit Selection View Go Run Terminal Help

post.dart - Blogging-App - Visual Studio Code

EXPLORER ...

> OPEN EDITORS

> BLOGGING-APP

/ android

/ build

/ fonts

/ images

Aman.png

Aman.png.jpg

Ayurish.jpg

Ayurish.png

B.png

developer.png

Pattern.jpg

/ ios

/ lib

/ db

PostService.dart

/ models

post.dart

/ screens

generated_plugin_re...

main.dart M

/ web

.flutter-plugins

.flutter-plugins-depen...

.gitignore

.metadata

.packages

bloggingapp.iml

pubspec.lock M

pubspec.yaml M

/ OUTLINE

/ TIMELINE

/ DEPENDENCIES

post.dart X

lib > models > post.dart > ...

```
1 import 'package:firebase_database.firebaseio_database.dart';
2 class Post{
3
4     static const KEY = "key";
5     static const DATE = "date";
6     static const TITLE = "title";
7     static const BODY = "body";
8
9     int date;
10    String key;
11    String title;
12    String body;
13
14    Post(this.date, this.title, this.body);
15
16    // String get key => _key;
17    // String get date => _date;
18    // String get title => _title;
19    // String get body => _body;
20
21
22    Post.fromSnapshot(DataSnapshot snap):
23        this.key = snap.key,
24        this.body = snap.value[BODY],
25        this.date = snap.value[DATE],
26        this.title = snap.value[TITLE];
27
28    Map toMap(){
29        return {BODY: body, TITLE: title, DATE: date, KEY: key};
30    }
31 }
```

Ln 1, Col 1 Spaces: 2 UTF-8

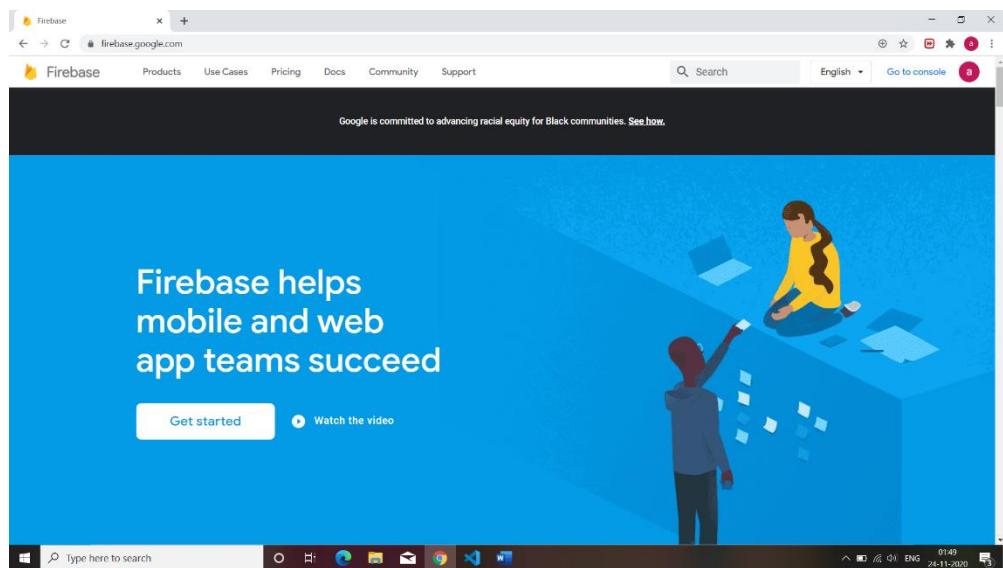
master* Type here to search

Filename: postservice.dart

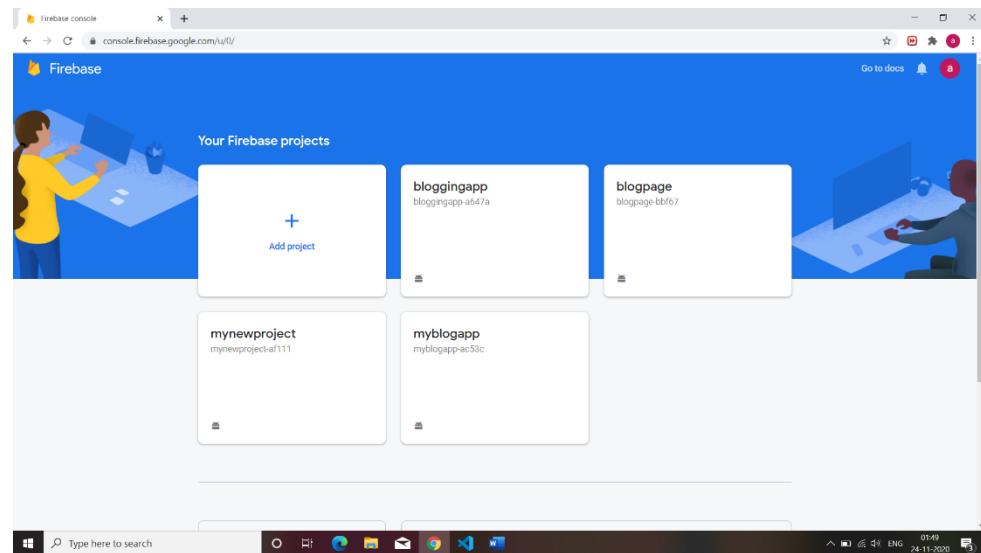
```
lib > db > PostService.dart > ...
1 import 'package:firebase_database.firebaseio_database.dart';
2
3
4 class PostService{
5   String nodeName = "posts";
6   FirebaseDatabase database = FirebaseDatabase.instance;
7   DatabaseReference _databaseReference;
8   Map post;
9
10 PostService(this.post);
11
12 addPost(){
13   // this is going to give a reference to the posts node
14   database.reference().child(nodeName).push().set(post);
15 }
16
17 deletePost(){
18   database.reference().child('${nodeName}/${post['key']}').remove();
19 }
20
21 updatePost(){
22   database.reference().child('${nodeName}/${post['key']}').update(
23     {"title": post['title'], "body": post['body'], "date":post['date']}
24   );
25 }
26 }
```

Output:

Firebase Screen



Add Project:



Realtime Database:

The screenshot shows the Firebase Project Overview page for a project named 'bloggingapp'. The left sidebar contains navigation links for Project Overview, Develop (Authentication, Cloud Firestore, Realtime Database, Storage, Hosting, Functions, Machine Learning), Quality (Crashlytics, Performance, Test Lab), Analytics (Dashboard, Events, Conversions, App Distribution), and Extensions. The main area displays the 'Develop' section for the Realtime Database. It shows two charts: 'Downloads (7d total)' at 263KB and 'Storage (current)' at 190B. Below the charts is a line graph showing storage usage from Nov 16 to Nov 22, with a significant spike on Nov 21. A modal window titled 'Store and sync app data in milliseconds' is open in the foreground.

The screenshot shows the Realtime Database Data tab for the 'bloggingapp' project. The left sidebar is identical to the previous screenshot. The main area shows the database structure under the 'Realtime Database' tab. A message at the top encourages using the Local Emulator Suite. The database tree shows a single 'posts' node with two child nodes: '-MMjzytcJnaC5eKEmpJk' and '-MMIPCDKyvR3CrgmsNIN'. Each node has three fields: 'body', 'date', and 'title'. The first node's body is 'I am Aman', date is '160604953937', and title is 'hello flutter'. The second node's body is 'welboard to Google', date is '1606072980366', and title is 'Google'. A note at the bottom indicates the database location is United States (us-central1).

Conclusion

While blogs can be useful, it is important to note that they are just tools. However, the use of blogging apps as a learning tool seems to be low-cost with high-returns. It is a given that technology will continue to influence learning. Blogging apps help students tend to learn best when they need information that they can put to use immediately. They are an effective and efficient method of allowing students to access information as it is needed and to make connections. This also helps students connect to online learning communities so they can access and evaluate information, and construct new learning paradigms for themselves. Finally, effectively using technology to develop blogging apps and to use them as a teaching and learning tool is a useful skill for students to have as they embark on the journey of life-long learning.