

A close-up, high-contrast photograph of a microphone grille, showing the intricate mesh pattern. The lighting creates strong highlights and shadows, emphasizing the texture. The background is dark and out of focus.

RADIO APPLICATION

PROBLEM STATEMENT

THIS APPLICATION IS BASED ON USER EXPERIENCE. BASICALLY WE PROVIDE A USER A PLATFORM WHERE THEY CAN LISTEN LIVE RADIO FOR SOME PARTICULAR CHANNELS ADDED BY US MANUALLY , HERE THEY CAN LISTEN SONGS, LIVE DISCUSSIONS ETC OR ANYTHING GOING ON RADIO CHANNEL

MOTIVATION AND OBJECTIVE

THIS APPLICATION HELPS USER TO PROVIDE RADIO CHANNEL AT PARTICULAR FREQUENCY.

IT PROVIDES USER TO HAVE SOME RELAXATION TIME AND CHILL TIME WHICH HELPS ANYONE TO RELEASE THEIR STRESS

THIS APPLICATION ALSO PROVIDES SOME RADIO CHANNEL ACCORDING THE TRENDING CATEGORY OF SONGS LIKE HIP HOP ETC.

MOTIVATION AND OBJECTIVE

IT IS VERY EASY TO USE AND DOES NOT REQUIRE OUR PERSONAL INFO.

IT MAY ALSO WORK ON POWERED VOICE ON WHICH WE ARE CURRENTLY WORKING ON.

ABSTRACT IDEA

I AM USING FLUTTER AND DART LANGUAGE TO
DESIGN THIS APPLICATION

IF WE NEED TO ADD THE POWERED VOICE FEATURE THEN
WE NEED TO USE SOME ARTIFICIAL INTELLIGENCE VOICE
ASSITANT WHICH WILL HELP US TO GIVE COMMANDS .

IT IS A PLATFORM WITH NUMEROUS RADIO CHANNELS AT
A PARTICULAR FREQUENCY WHICH HELP US TO LISTEN
ANYTHING LIVE HAPPENING ON CHANNEL

WE CAN SELECT ANY CHANNEL ACCORDING TO OUR
INTEREST

FEATURES

UI/UX ANDROID AND IOS PLATFORM

ARTIFICIAL INTELLIGENCE VOICE ASSISTANT WHICH IS
UNIQUE FEATURE IN RADIO APPLICATION

IT HAS NUMEROUS RADIO CHANNELS, WHICH IS SELECTED
ON PARTICULAR CATEGORY.

WE CAN ADD ANY CHANNELS MANUALLY ALSO

FEATURES

WORKS ON BOTH ANDROID AND IOS PLATFORM

WE CAN GIVE BASIC VOICE COMMANDS WHICH WORK
ON RADIO APPLICATIONS

EXCEPTIONS

WELL EVERY APPLICATION HAS SOME FEATURES AND HAS SOME EXCEPTIONS.

THIS RADIO APPLICATION BASICALLY A BETA VERSION SO IT WILL REACT DIFFERENTLY ON DIFFERENT DEVICES

SAMSUNG DEVICES HAVE SOME ISSUES REGARDING THIS APPLICATION WHICH IS MAIN THREAD AND BSETTING FALSE ERROR BUT IT WILL BE RESOLVED IN FUTURE

WE CAN ONLY ADD SOME RADIO CHANNELS BECOZ OF SOME LIMITATIONS

THE ALL RADIOS LINK HAS BEEN UPDATED AND TURNED TO PRIVATE SO WE CANT ACCESS IT.

SO ONLY 1 LINK WAS AVAILABLE WHICH WAS WORKING

INTRODUCTION TO THE ELEMENTS OF AI RADIO APPLICATION

THE RADIO APPLICATION IS CREATED USING FLUTTER WITH DART LANGUAGE ON VISUAL CODE ENVIRONMENT

THE KEY SOFTWARES USED -

- FLUTTER
- DART LANGUAGE
- VISUAL STUDIO CODE
- AI ALAN VOICE

LITERATURE REVIEW

Research methodology: The research papers I study to work on my ai radio application is related to the flutter .

Flutter works on basically a web based, object oriented programming language called DART, and I will tell you about the dart language in further slides.

FLUTTER

Native mobile applications :

The meaning of native in the field of mobile applications refers to applications that are built to run on a specific platform or OS. There are numerous languages that can be used for building native mobile applications and a few of examples of these are: Kotlin, Java and Swift.

Cross-platform mobile application development:

Cross-platforming refers to a product or software that can be used on another platform than what it was developed for. In app development the principle of cross-platforming is to build and maintain only one code base, which is the alluring part of using it.

Native implementation of UI :

One thing that is associated with native mobile applications, is that there are more fluid looking animations and easier integration with the mobile technology. Native applications inherit the targeted platforms looks and apply it to their appearance, this is called Native UI.

FLUTTER

WHAT IS FLUTTER??

Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase.

WHAT FLUTTER IS USED FOR?

1 Fast development

Flutter engineered for high development velocity. Stateful hot reload allows you to change your code and see it come to life is less than a second without losing the state of the app.

2. Expressive + Flexible UI

Flutter moves to a widget, rendering, animation and gestures into this framework to give you to complete control over every pixel on the screen. It means you have the flexibility to build a custom design.

FLUTTER

3 Native apps for Android and IOS

Flutter apps follow platform conventions and interface details such as scrolling, navigation, icons, fonts, etc. That why apps built with Flutter features on both of the **APPSTORE** and **GOOGLE PLAY STORE**.

4 Hot Reload

In flutter, very save on the app and just as you do on the web just hit a refresh and your codes also refresh. Imaging that Facebook SDK it would be so humongous if it would have been designing android and you hit a recompile. So many things to have recompile and it would take probably days.

6. Using Dart as a Programming Language

Dart is an object-oriented programming language that which used for writing mobile application code for Flutter and which contributes to the efficiency and effectiveness of app development flow.

WE ARE GOING TO EXPLORE ABOUT DART LANGUAGE IN FURTHER SLIDES.

FLUTTER

WHY FLUTTER IS BETTER THAN REACT NATIVE AND OTHER TOOLS??

Now we don't need to go further into this topic . As it is the whole different concept but one of the main key benefit of using flutter over react native is, flutter is faster and common platform for ios ,web and android applications.

For further research I will recommend you to read -

--A Comparison of Performance and Looks Between Flutter and Native Applications When to prefer Flutter over native in mobile application development

--By Matilda Olsson June 13th, 2020

Link-<https://www.diva-portal.org/smash/get/diva2:1442804/FULLTEXT01.pdf>

DART LANGUAGE

WHAT IS DART LANGUAGE?

Dart is a programming language designed for client development, such as for the web and mobile apps. It is developed by Google and can also be used to build server and desktop applications. Dart is an object-oriented, class-based, web programming based language.

Why do we use dart language?

Dart is a **client-optimized language for developing fast apps on any platform**. Its goal is to offer the most productive programming language for multi-platform development, paired with a flexible execution runtime platform for app frameworks. Its major advantages are **its stability and ease of learning**

DART LANGUAGE

Amazing Apps Built with Flutter Framework

Google Ads

KlasterMe

Reflectly

Xianyu by Alibaba

Etc.

Research paper:

DART Evolved for Web - A Comparative Study with
JavaScript by--Sabyasachi Mohanty, Smriti Rekha Dey

Department of Computer Sc & Engg Centurion University of
Technology & Management, Bhubaneswar, India

Link-

https://www.researchgate.net/publication/278412445_DART_Evolved_for_Web_-_A_Comparative_Study_with_JavaScript

The COMPARISON BETWEEN THE MOST POPULAR UI'S

FLUTTER VS REACT NATIVE

The difference between the performances of React Native and Flutter is quite debatable. The respective community of both RN and Flutter are a bit divided over the topic 'performance' as everything about them sounds good in terms of agility and speed. However, RN has also received some criticism for its performance due to the involvement of native modules and third-party libraries. That said, let's dive in a bit more in detail to learn about the React Native vs. Flutter performance difference. For this, we will consider a simple "hello world" app with one simple image built with both RN and Flutter.

DART LANGUAGE VS JAVASCRIPT

Dart is **approximately two times faster than JavaScript**. Dart is type-safe and compiled with both AOT and JIT compilers. Dart is very scalable across projects. Dart is very similar to Javascript and easy to learn if you already know Javascript.

JavaScript is an interpreted language, so it might feel lighter and faster. It's actually faster than other compiled languages like Java. However, Dart proved to be much faster when benchmarked against JavaScript.

VISUAL STUDIO

WHAT IS VISUAL STUDIO CODE?

Visual Studio Code is a code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring

NOW WE CAN ALSO USE ANDROID STUDIO TO WORK WITH FLUTTER BUT ITS PERSONAL PREFERNCES ACCORDING TO THE COMFORT OF INDIVIDUAL.

AI ALAN VOICE

WHAT IS AI ALAN VOICE?

Alan is an **advanced Voice AI Platform** that allows you to add a voice interface to your app without overhead. Alan provides a complete serverless environment to build robust and reliable in-app voice assistants and chatbots.

WHY WE ARE USING ALAN VOICE IN OUR RADIO APPLICATION?

1. **Enable a highly engaging user experience**
2. **Make your application frustration free**
5. **Understand what users really want and improve your product offering**

AI ALAN VOICE

WE ARE NOT USING ANY PARTICULARLY VOICE ASSISTANT
CREATED BY US USING NLP AS IT WOULD SAVE TIME AND
EFFORT , AND ALAN VOICE PLATFORM PROVIDE US
DIFFERENT INTERACTIONS AND WE JUST NEED TO SET UP
ITS DEPENDENCIES IN OUR PROJECT.

NOW I USED THE WORD DEPENDENCIES , LET ME TELL YOU ABOUT DEPENDENCIES

TO SET UP ALL THE FUNCTIONS AND LIBRAIES WE NEED TO SET UP SOME FLUTTER DEPENDENCIES/PACKAGES IN OUR PROJECT-

THE DEPENDENCIES I USED ARE-

audioplayers: ^0.17.4

cupertino_icons: ^1.0.2

google_fonts: ^1.1.1

velocity_x: ^1.0.1

alan_voice: ^2.4.0

AI ALAN SCRIPTS

The screenshot displays the AI Alan Studio web interface. The browser address bar shows the URL `studio.alan.app/projects/AIRADIO/AIRADIO`. The top navigation bar includes the AI Alan logo, a project selector for "AIRADIO", a "Development" dropdown, and buttons for "Integrations", "Edit", and "Test".

On the left sidebar, under the "Scripts" section, there is a list of scripts: "AIRADIO" (15 hours ago), "AIDATA" (15 hours ago), and "SmallTalk" (4 days ago). Below this list are buttons for "Sync with GitHub", "Export", "Import" (with a note "Drag&drop files here, or click to select files"), and a "Saved" status.

The main workspace contains a DartPad editor with the following code:

```
1 intent('hello world', p => {
2   p.play('hello|hi there');
3 });
4
5 question(
6   'what this app can do?', 'what does this app do?',
7   reply('This is a radio app where you can ask me to play some music.'));
8 );
9
10
11
12 intent('play ${CHANNEL* (.*)} fm', p => {
13   let channel = project.radios.filter(x => x.name.toLowerCase() === p.CHANNEL.value.toLowerCase())[0];
14   try {
15     p.play({ "command": "play_channel", "id": channel.id });
16     p.play('Playing Now|on it|Ok boss|Doing it');
17   } catch (err) {
18     console.log("Can't play");
19     p.play("I cannot play this");
20   }
21 });
22
23 intent('play (some|) ${CATEGORY* (.*)} music', p => {
24   let channel = project.radios.filter(x => x.category.toLowerCase() === p.CATEGORY.value.toLowerCase())[0];
25   try {
26     p.play({ 'command': 'play_channel', 'id': channel.id });
27     p.play('playing now|On it|Ok boss');
28   } catch (error) {
29     console.log("Can't play");
30     p.play('I could not find this genre');
31   }
32 });
33
34
35
36 intent('play (some|) music', p => {
```

On the right side, there is a "Debugging Chat" window with a blue background and a speech bubble icon. It contains the text "Type here to chat or press the mic button to speak..." and a microphone icon.

The bottom status bar shows "Syntax OK", "LOGS", "Errors", "Info 27", "PHRASES", "Unrecognized", "Input 21", "Output 21", and "FILTERS". The system tray at the very bottom shows the date and time as "14:33 30-10-2021" and the temperature as "30°C".

SOME INTERACTIONS COMMAND-

- PLAY 107 FM
- PLAY SOME MUSIC
- PLAY CLASSIC
- PLAY NEXT CHANNEL
- PLAY PREVIOUS CHANNEL

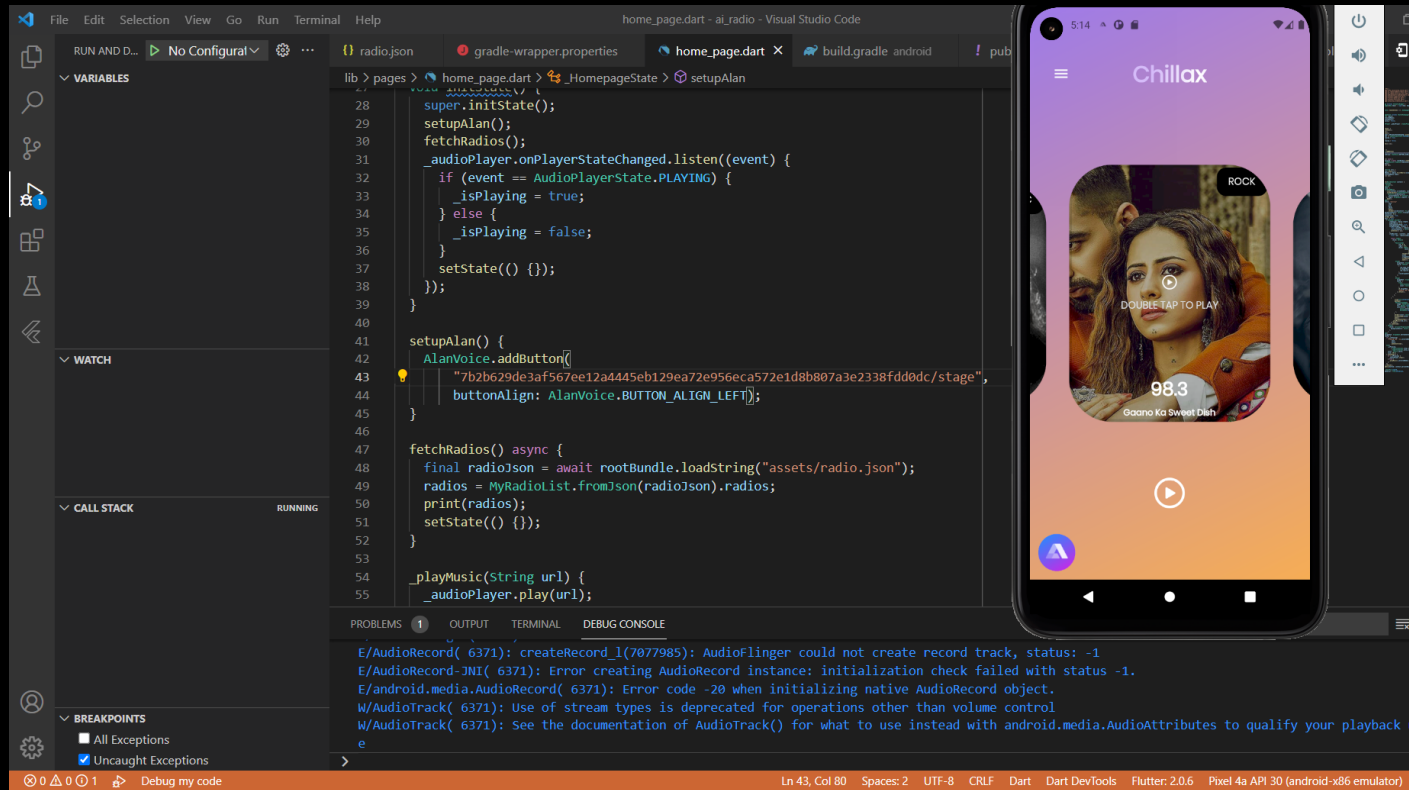
ETC....

SOME INTERACTIONS COMMAND-

- HEY ALAN
- ARE U GOOD OR BAD?
- HOW ARE U ?
- I LOVE U
- WHO ARE U?

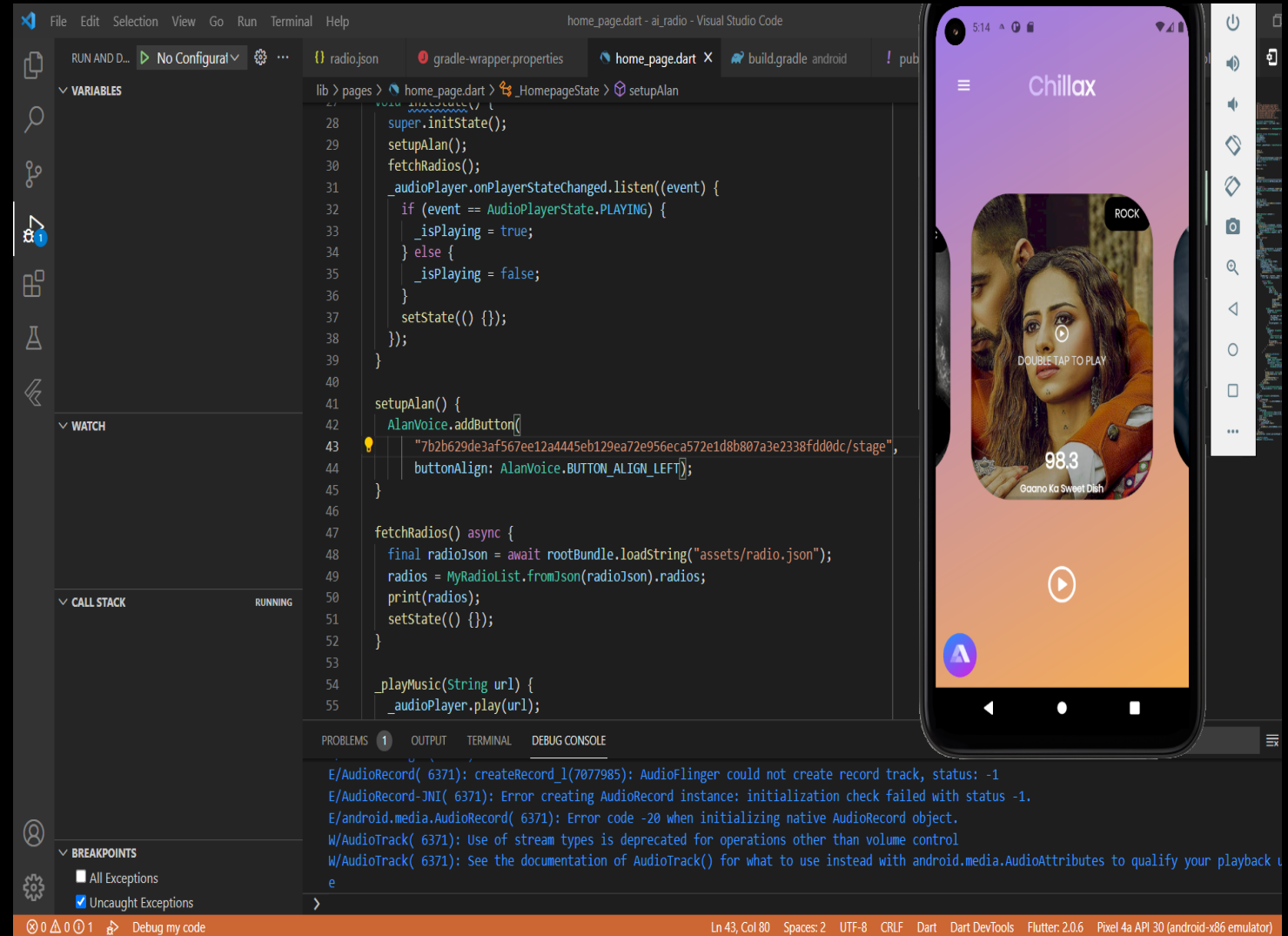
ETC.....

DESIGN OF MY PROJECT-



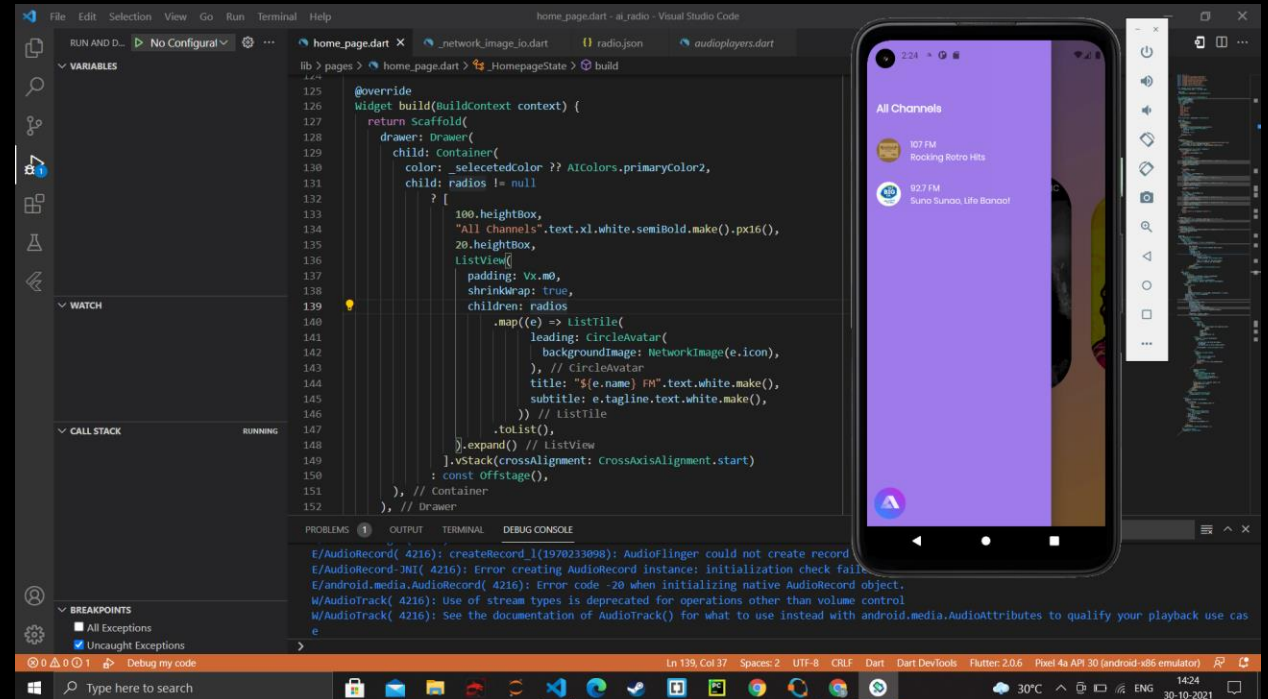
The center button
is play button.
The bottom left
corner is voice
assitant button.
The top right is
navbar.

DESIGN



NAVBAR WITH LIST OF 2 RADIOS

DESIGN



PLAY 107 FM

DESIGN

The image shows a screenshot of a Flutter application in Visual Studio Code, displaying the design of a radio player interface. The application is running on an Android emulator, showing a mobile app with a purple header, a white notification bar, and a main content area with a large image of a woman and a play button. The interface includes a list of radio stations, a play button, and a progress bar.

The code in the editor shows the widget build function for the radio player:

```
lib > pages > home_page.dart > _HomepageState > build
124
125 @override
126 Widget build(BuildContext context) {
127   return Scaffold(
128     drawer: Drawer(
129       child: Container(
130         color: _selectedColor ?? AIColors.primaryColor2,
131         child: radios != null
132           ? [
133             100.heightBox,
134             "All Channels".text.xl.white.semiBold.make().px16(),
135             20.heightBox,
136             ListView(
137               padding: Vx.m0,
138               shrinkWrap: true,
139               children: radios
140                 .map(e => ListTile(
141                   leading: CircleAvatar(
142                     backgroundImage: NetworkImage(e.icon),
143                   ), // CircleAvatar
144                   title: "${e.name} FM".text.white.make(),
145                   subtitle: e.tagline.text.white.make(),
146                 )) // ListTile
147                 .toList(),
148             ).expand() // ListView
149           ].vstack(crossAxisAlignment: CrossAxisAlignment.start)
150           : const Offstage(),
151     ), // Container
152   ), // Drawer
```

The bottom of the screen shows the status bar with the following information: Ln 139, Col 37, Spaces: 2, UTF-8, CRLF, Dart, Dart DevTools, Flutter: 2.0.6, Pixel 4a API 30 (android-x86 emulator), 14:27, 30-10-2021.

QUES ASKED "
HOW ARE U
ALAN?"

DESIGN

The image shows a development environment with Visual Studio Code. The editor displays the `home_page.dart` file, which contains the `HomePageState` widget. The code defines a `build` method that returns a `Scaffold` with a `Drawer` and a `Container`. The `Container` has a `color` of `_selecetedColor ?? AIColors.primaryColor2` and a `child` of `radios != null`. The `radios` are a `ListView` of `radios` with a `padding` of `Vx.m0` and `shrinkWrap` of `true`. The `children` of the `radios` are a `map` of `ListTile` with a `leading` of `CircleAvatar` and a `backgroundImage` of `NetworkImage(e.icon)`. The `title` is `"${e.name} FM".text.white.make()` and the `subtitle` is `e.tagline.text.white.make()`. The `toList` method is called on the `radios` and the `expand` method is called on the `ListView`. The `vsStack` method is called with `crossAxisAlignment: CrossAxisAlignment.start` and `const Offstage()`. The `Container` is then returned. The `Drawer` is also returned.

On the right, a smartphone mockup displays the app's interface. It features a purple header with the text "Alan Service" and "I'm doing well.". Below this is a large image of a woman's face with the text "CLASSIC" and "DOUBLE TAP TO PLAY". At the bottom, there is a play button and a text input field with the text "how are you Alan".

The bottom of the image shows a Windows taskbar with various icons, including the Start button, search bar, and several application icons. The system tray shows the date and time as 14:26 on 30-10-2021.

Pubspec.yaml file

The screenshot shows the VS Code editor with the `pubspec.yaml` file open. The file is for an AI radio application and includes the following content:

```
environment:
  sdk: ">=2.7.0 <3.0.0"

dependencies:
  flutter:
    sdk: flutter
  audioplayers: ^0.17.4

# The following adds the Cupertino Icons font to your application.
# Use with the CupertinoIcons class for iOS style icons.
cupertino_icons: ^1.0.2
google_fonts: ^1.1.1
velocity_x: ^1.0.1
alan_voice: ^2.4.0

dev_dependencies:
  flutter_test:
    sdk: flutter

# For information on the generic Dart part of this file, see the
# following page: https://dart.dev/tools/pub/pubspec

# The following section is specific to Flutter.
flutter:

# The following line ensures that the Material Icons font is
# included with your application, so that you can use the icons in
```

The Explorer sidebar on the left shows the project structure, including `home_page.dart`, `pubspec.yaml`, `_network_image_io.dart`, `radio.json`, `assets`, `lib`, `models`, `radio.dart`, `pages`, `home_page.dart`, `utils`, `ai_util.dart`, `widgets`, `generated_plugin_registrant.dart`, `main.dart`, `test`, `web`, `flutter-plugins`, `flutter-plugins-dependencies`, `.gitignore`, `.metadata`, `.packages`, `ai_radio.iml`, `flutter_01.log`, `pubspec.lock`, `pubspec.yaml`, `README.md`, `OUTLINE`, and `DEPENDENCIES`.

The bottom status bar shows the file is at line 30, column 46, with 2 spaces, UTF-8 encoding, CRLF line endings, and is a YAML file.

Radio.json

The screenshot shows the VS Code editor with the `radio.json` file open. The file is for an AI radio application and includes the following content:

```
{
  "radios": [
    {
      "id": 1,
      "name": "107",
      "tagline": "Rocking Retro Hits",
      "color": "0xff090909",
      "desc": "Your heartbeat changes to mimics the music you listen to.",
      "url": "https://puranijeanshdli-1h.akamaihd.net/i/PuranijeansHDLive_1_1@334555/index_1_a-p.m3u8?sd=10&rebase=on",
      "icon": "https://mytuner.global.ssl.fastly.net/media/tvos_radios/uvclz26a6d7x.jpeg",
      "image": "https://i.pinimg.com/originals/7f/f6/13/7ff613ed815f1eb56a90794ec64eecfe.jpg",
      "lang": "Hindi",
      "category": "classic",
      "disliked": false,
      "order": 1
    },
    {
      "id": 2,
      "name": "92.7",
      "tagline": "Suno Sunao, Life Banao!",
      "color": "0xffa11431",
      "desc": "The chills you get when you listen to music, is mostly caused by the brain releasing dopamine while anticipating",
      "url": "http://sc-bb.1.fm:8017/;",
      "icon": "https://mytuner.global.ssl.fastly.net/media/tvos_radios/m8afyszryagt.png",
      "image": "https://mir-s3-cdn-cf.behance.net/project_modules/max_1200/b5df4c18876369.562d0d4bd94cf.jpg",
      "lang": "Hindi",
      "category": "non"
    }
  ]
}
```

The Explorer sidebar on the left shows the project structure, including `home_page.dart`, `main.dart`, `_network_image_io.dart`, `radio.json`, `assets`, `lib`, `models`, `radio.dart`, `pages`, `home_page.dart`, `utils`, `ai_util.dart`, `widgets`, `generated_plugin_registrant.dart`, `main.dart`, `test`, `web`, `flutter-plugins`, `flutter-plugins-dependencies`, `.gitignore`, `.metadata`, `.packages`, `ai_radio.iml`, `flutter_01.log`, `pubspec.lock`, `pubspec.yaml`, `README.md`, `OUTLINE`, and `DEPENDENCIES`.

The bottom status bar shows the file is at line 30, column 34, with 4 spaces, UTF-8 encoding, CRLF line endings, and is a JSON file.

Main.dart

main.dart - ai_radio - Visual Studio Code

EXPLORER

- home_page.dart
- main.dart
- _network_image_io.dart
- radio.json

lib > main.dart

```
1 import 'package:ai_radio/pages/home_page.dart';
2 import 'package:flutter/material.dart';
3 import 'package:flutter/services.dart';
4 import 'package:google_fonts/google_fonts.dart';
5
6 void main() {
7   SystemChrome.setSystemUIOverlayStyle(
8     SystemUiOverlayStyle(statusBarColor: Colors.transparent));
9   runApp(MyApp());
10 }
11
12 class MyApp extends StatelessWidget {
13   // This widget is the root of your application.
14   @override
15   Widget build(BuildContext context) {
16     return MaterialApp(
17       title: 'Flutter Demo',
18       theme: ThemeData(
19         fontFamily: GoogleFonts.poppins().fontFamily,
20         primarySwatch: Colors.blue,
21       ),
22       debugShowCheckedModeBanner: false,
23       home: HomePage(),
24     );
25   }
26 }
27
```

PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS C:\Users\sanka\OneDrive\Desktop\radio app\ai_radio>

Ln 19, Col 1 Spaces: 2 UTF-8 CRLF Dart Dart DevTools Flutter: 2.0.6 Chrome (web-javascript)

Radios.dart

radio.dart - ai_radio - Visual Studio Code

EXPLORER

- home_page.dart
- _network_image_io.dart
- radio.json
- radio.dart

lib > models > radio.dart > MyRadioList > copyWith

```
1 import 'dart:convert';
2
3 import 'package:flutter/foundation.dart';
4
5 class MyRadioList {
6   final List<MyRadio> radios;
7   MyRadioList({
8     this.radios,
9   });
10
11   MyRadioList copyWith({
12     List<MyRadio> radios,
13   }) {
14     return MyRadioList(
15       radios: radios ?? this.radios,
16     );
17   }
18
19   Map<String, dynamic> toMap() {
20     return {
21       'radios': radios?.map((x) => x?.toMap()).toList(),
22     };
23   }
24
25   factory MyRadioList.fromMap(Map<String, dynamic> map) {
26     if (map == null) return null;
27
28     return MyRadioList(
29       radios: List<MyRadio>.from(map['radios']?.map((x) => MyRadio.fromMap(x))
30
```

PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS C:\Users\sanka\OneDrive\Desktop\radio app\ai_radio>

Ln 16, Col 7 Spaces: 2 UTF-8 CRLF Dart Dart DevTools Flutter: 2.0.6 Chrome (web-javascript)

VARIOUS FILES WERE USED

RADIOS.DART

HOMEPAGE.DART

AI_UTILS.DART

MAIN.DART

RADIOS.JSON

PUBSPEC.YAML

AI ALAN SCRIPTS

AND ETC

CONCLUSION

WE CREATE AND SUCESSFULLY IMPLEMENTED
RADIO APPLICATION WITH AI VOICE ASSITANT.
THE DEMO AND MODULE IMPLEMENTATION WAS
SUCCESSFULLY DONE.

Thank you!