Source :- api_service.dart

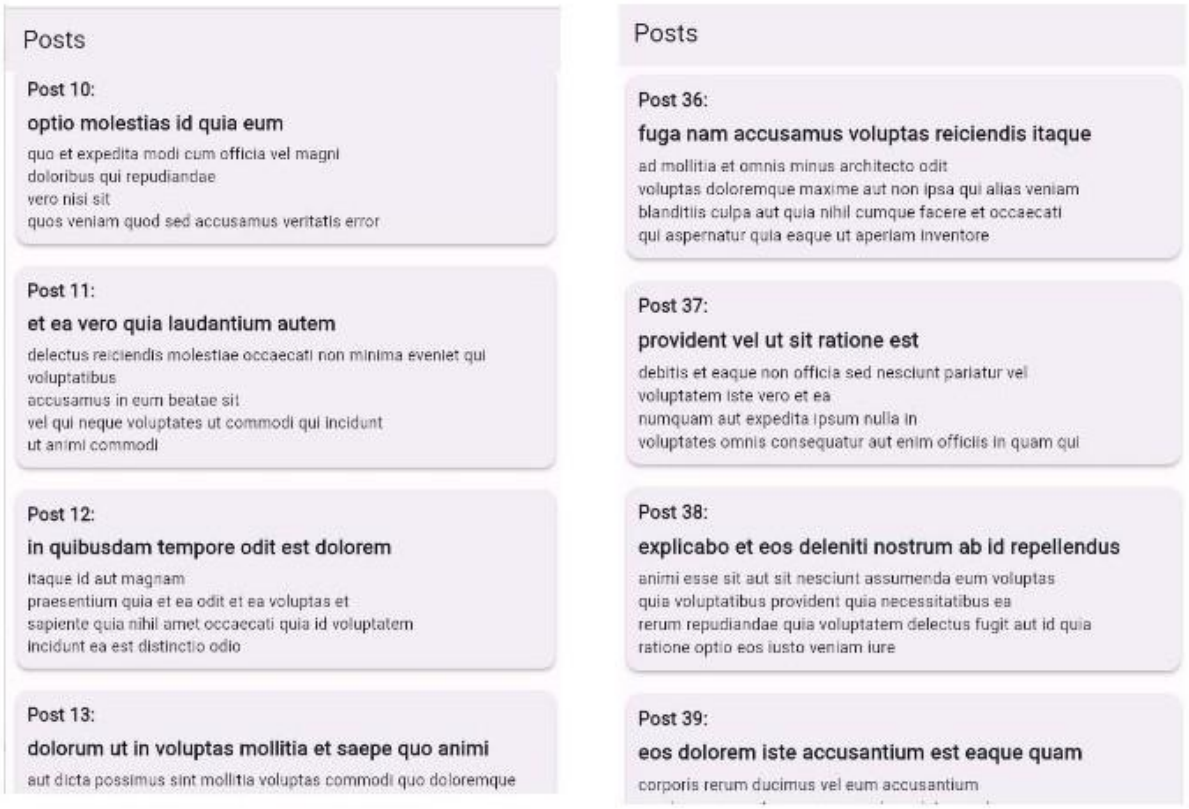
```
import 'dart:convert';
import 'package:http/http.dart' as http;
class Post { final int userId; final int id;
final String title; final String body;
Post({
required this.userId, required this.id, required this.title, required this.body,
});
factory Post.fromJson(Map<String, dynamic> json) { return Post(
userId: json['userId'], id: json['id'],
title: json['title'], body: json['body'],
);
}
}
class ApiService {
static const String baseUrl = 'https://jsonplaceholder.typicode.com/todos/1';
static Future<List<Post>> fetchPosts() async {
final response = await http.get(Uri.parse('$baseUrl/posts'));
if (response.statusCode == 200) {
List<dynamic> jsonResponse = json.decode(response.body); return jsonResponse.map((post) =>
Post.fromJson(post)).toList();
} else {
throw Exception('Failed to load posts');
}
}
}
```

Source :- data_screen.dart

```
import 'package:flutter/material.dart';
import 'package:resetapi/api_service.dart';
class DataScreen extends StatefulWidget {
@override
_DataScreenState createState() => _DataScreenState();
}
class _DataScreenState extends State<DataScreen> {
late Future<List<Post>> posts;
@override
void initState() { super.initState();
posts = ApiService.fetchPosts();
}
@override
Widget build(BuildContext context) { return Scaffold(
appBar: AppBar( title: Text('Posts'),
),
body: Center(
child: FutureBuilder<List<Post>>(
```

```
future: posts,
builder: (context, snapshot) {
if (snapshot.hasData) {
return ListView.builder(
itemCount: snapshot.data!.length,
itemBuilder: (context, index) {
return Card(
elevation: 3,
margin: EdgeInsets.all(10),
child: Padding(
padding: EdgeInsets.all(10),
child: Column(
crossAxisAlignment: CrossAxisAlignment.start, children: [
Text(
'Post ${index + 1}:', // Add label here style: TextStyle(
fontWeight: FontWeight.bold, fontSize: 16,
),
),
SizedBox(height: 5), Text(
snapshot.data![index].title, style: TextStyle(
fontWeight: FontWeight.bold, fontSize: 18,
),
),
SizedBox(height: 5), Text(snapshot.data![index].body),
],
),
),
);
},
);
} else if (snapshot.hasError) { return Text("${snapshot.error}");
}
// By default, show a loading spinner. return CircularProgressIndicator();
},
),
),
);
}
}
```

Output :-


Posts
Post 10:
optio molestias id quia eum
quo et expedita modi cum officia vel magni
doloribus qui repudiandae
vero nisi sit
quos veniam quod sed accusamus veritatis error
Post 11:
et ea vero quia laudantium autem
delectus reiciendis molestiae occaecati non minima eveniet qui
voluptatibus
accusamus in eum beatae sit
vel qui neque voluptates ut commodi qui incidunt
ut animi commodi
Post 12:
in quibusdam tempore odit est dolorem
itaque id aut magnam
praesentium quia et ea odit et ea voluptas et
sapiente quia nihil amet occaecati quia id voluptatem
incidunt ea est distinctio odio
Post 13:
dolorum ut in voluptas mollitia et saepe quo animi
aut dicta possimus sint mollitia voluptas commodi quo doloremque
Posts
Post 36:
fuga nam accusamus voluptas reiciendis itaque
ad mollitia et omnis minus architecto odit
voluptas doloremque maxime aut non ipsa qui alias veniam
blanditiis culpa aut quia nihil cumque facere et occaecati
qui aspernatur quia eaque ut aperiam inventore
Post 37:
provident vel ut sit ratione est
debitis et eaque non officia sed nesciunt pariatur vel
voluptatem iste vero et ea
numquam aut expedita ipsum nulla in
voluptates omnis consequatur aut enim officiis in quam qui
Post 38:
explicabo et eos deleniti nostrum ab id repellendus
animi esse sit aut sit nesciunt assumenda eum voluptas
quia voluptatibus provident quia necessitatibus ea
rerum repudiandae quia voluptatem delectus fugit aut id quia
ratione optio eos iusto veniam iure
Post 39:
eos dolorem iste accusantium est eaque quam
corporis rerum ducimus vel eum accusantium

Experiment 12

AIM:- Create and application Parsing JSON data from REST API in Flutter.

Source :- Main.dart

Code :-

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

import 'package:resetapi/data_screen.dart';

void main() { runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp( debugShowCheckedModeBanner: false,

title: 'Flutter REST API Demo',

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: DataScreen(),

);

}

}
```

Source :- api_service.dart

Code :-

```
import 'dart:convert';
```

```
import 'package:http/http.dart' as http;
class Post { final int userId; final int id;
final String title; final String body;
Post({
required this.userId, required this.id, required this.title, required this.body,
});
factory Post.fromJson(Map<String, dynamic> json) { return Post(
userId: json['userId'], id: json['id'],
title: json['title'], body: json['body'],
);
}
}
class ApiService {
static const String baseUrl = 'https://jsonplaceholder.typicode.com/todos/1';
static Future<List<Post>> fetchPosts() async {
final response = await http.get(Uri.parse('$baseUrl/posts'));
if (response.statusCode == 200) {
List<dynamic> jsonResponse = json.decode(response.body); return jsonResponse.map((post) =>
Post.fromJson(post)).toList();
} else {
throw Exception('Failed to load posts');
}
}
}
```

Source :- data_screen.dart

```
import 'package:flutter/material.dart';
import 'package:resetapi/api_service.dart';
class DataScreen extends StatefulWidget { @override
_DataScreenState createState() => _DataScreenState();
}
class _DataScreenState extends State<DataScreen> {
late Future<List<Post>> posts;
@override
void initState() { super.initState();
posts = ApiService.fetchPosts();
}
@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar( title: Text('Posts'),
),
body: Center(
child: FutureBuilder<List<Post>>( future: posts,
builder: (context, snapshot) { if (snapshot.hasData) {
return ListView.builder(
itemCount: snapshot.data!.length, itemBuilder: (context, index) {
```

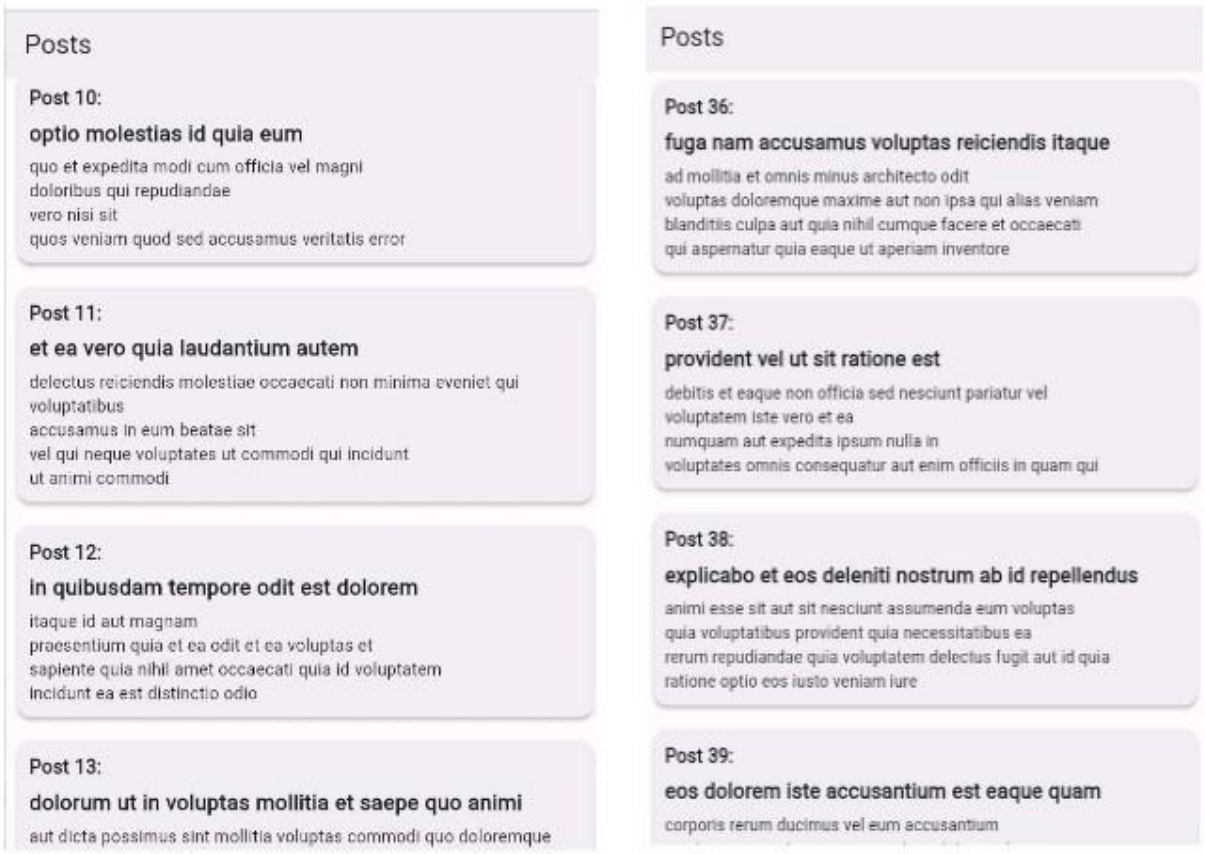
```
return Card( elevation: 3,
margin: EdgeInsets.all(10),
child: Padding(
padding: EdgeInsets.all(10),
child: Column(
crossAxisAlignment: CrossAxisAlignment.start,
children: [
Text(
'Post ${index + 1}:', // Add label here style: TextStyle(
fontWeight: FontWeight.bold, fontSize: 16,
),
),
SizedBox(height: 5), Text(
snapshot.data![index].title, style: TextStyle(
fontWeight: FontWeight.bold, fontSize: 18,
),
),
SizedBox(height: 5), Text(snapshot.data![index].body),
],
),
),
);
},
);
} else if (snapshot.hasError) { return Text("${snapshot.error}");
}
// By default, show a loading spinner. return CircularProgressIndicator();
},
),
),
);
}
}
```

post_model.dart:

```
class Post {
final int userId;
final int id;
final String title;
final String body;
Post({
required this.userId, required this.id, required this.title, required this.body,
});
factory Post.fromJson(Map<String, dynamic> json) { return Post(
userId: json['userId'], id: json['id'],
title: json['title'], body: json['body'],
);
}
```

```
}
dev_dependencies: flutter_test:
sdk: flutter http: ^0.13.3
```

Output :-



Experiment 13

AIM:- Create and application using Hardware Interaction in Flutter.

Source :- Main.dart

Code :-

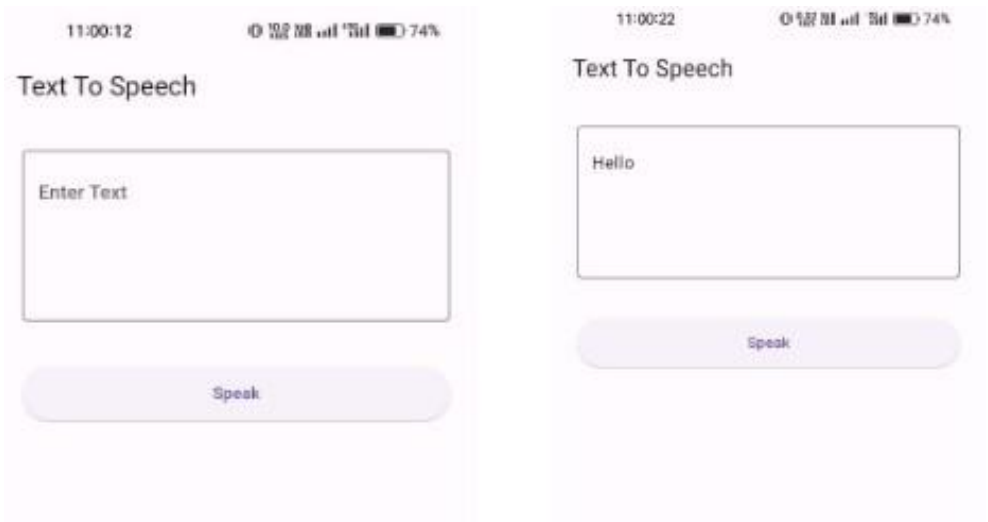
```
import 'package:flutter/material.dart';
import 'home_screen.dart';
void main(){
runApp(MyApp());
}
class MyApp extends StatelessWidget {
const MyApp({super.key});
@override
Widget build(BuildContext context) {
return MaterialApp(
debugShowCheckedModeBanner: false,
title: "Text To Speech",
theme: ThemeData(
primarySwatch: Colors.indigo,
),
home: HomeScreen(),
);
}
}
```

homescreen.dart:

```
import 'dart:async';
import 'package:flutter/material.dart';
import 'package:flutter_tts/flutter_tts.dart';
class HomeScreen extends StatefulWidget {
const HomeScreen({super.key});
@override
State<HomeScreen> createState() => _HomeScreenState();
}
class _HomeScreenState extends State<HomeScreen> {
final FlutterTts flutterTts = FlutterTts();
final TextEditingController textController = TextEditingController();
@override
void dispose() {
textController.dispose();
super.dispose();
} Future<void> speak(String text) async{
await flutterTts.setLanguage('en-US');
await flutterTts.setPitch(1.0);
await flutterTts.setSpeechRate(0.5);
await flutterTts.speak(text);
}
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(
title: Text("Text To Speech"),
),
body: Padding(
padding: EdgeInsets.all(20),
child: Column(
crossAxisAlignment: CrossAxisAlignment.stretch,
children: [
TextField(
controller: textController,
decoration: InputDecoration(
hintText: 'Enter Text',
border: OutlineInputBorder(),
),
maxLines: 4,
),
SizedBox(height: 30,),
ElevatedButton(onPressed: () {
speak(textController.text);
},
child: Text('Speak'),
),
],
),
),
```

```
);
}
}
```

Output :-


11:00:12
74%
Text To Speech
Enter Text
Speak
11:00:22
74%
Text To Speech
Hello
Speak