Software Tools 4

Assignment 6

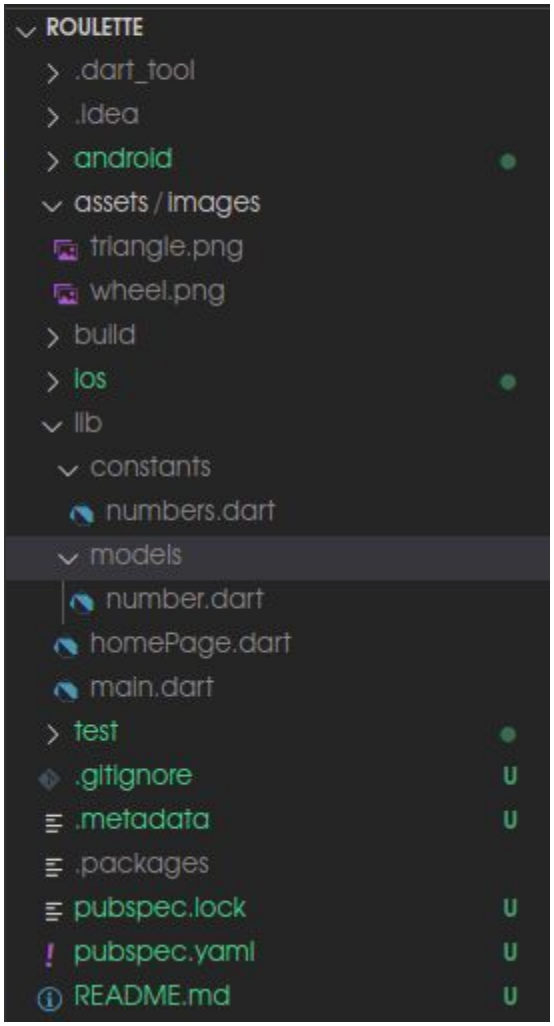
Krunal Rank
U18CO081

Develop a Roulette Game for Android.

Answer:

Tech Stack used :
Dart
Flutter SDK

Project Directory Structure:



Code:

./lib/main.dart:

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

import 'homePage.dart';

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

class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
   return MaterialApp(
     title: 'Casino Roulette',
     theme: ThemeData(
       primaryColor: Color(0xff3f51b5),
       accentColor: Color(0xff3f51b5),
       visualDensity: VisualDensity.adaptivePlatformDensity,
     ),
     home: HomePage(title: 'Casino Roulette'),
   );
 }
}
```

./lib/homePage.dart:

```dart
import 'dart:math';

import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:roulette/constants/numbers.dart';
import 'package:roulette/models/number.dart';

class HomePage extends StatefulWidget {
 HomePage({Key key, this.title}) : super(key: key);

 final String title;

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

class _MyHomePageState extends State<HomePage>
   with SingleTickerProviderStateMixin {
 AnimationController _controller;
 double rotatedDegree = 0.0;
 bool wheelRotated = false;
 bool wheelRotating = false;
 double startValue = 0;
 double endValue = Random().nextDouble() + Random().nextInt(100);
 Number num = numbers[0];

 @override
 void initState() {
   _controller = AnimationController(
     duration: const Duration(milliseconds: 5000),
     vsync: this,
   );
   super.initState();
 }

 @override
 void dispose() {
   _controller.dispose();
   super.dispose();
 }

 void _playRoulette() async {
   setState(() {
```

```
    startValue = endValue;
    endValue = endValue + Random().nextDouble() + Random().nextInt(10);
    wheelRotating = true;
  });
  _controller.reset();
  await _controller.forward();
  final eachSector = 1.0 / 37.0;
  final deg = endValue - endValue.floor() + eachSector / 2.0;
  dynamic idx = deg / eachSector;
  idx = idx.floor() % 37;
  setState(() {
    wheelRotated = true;
    wheelRotating = false;
    num = numbers[idx];
  });
}

@override
Widget build(BuildContext context) {
  final screenHeight = MediaQuery.of(context).size.height;
  final screenWidth = MediaQuery.of(context).size.width;
  return Scaffold(
    appBar: AppBar(
      title: Text(widget.title),
    ),
    body: Center(
      child: ListView(
        scrollDirection: Axis.vertical,
        shrinkWrap: true,
        children: [
          Padding(
            padding: EdgeInsets.all(screenHeight * 0.01),
            child: Image(
              image: AssetImage('assets/images/triangle.png'),
              height: screenHeight * 0.05,
            ),
          ),
          Padding(
            padding: EdgeInsets.all(screenHeight * 0.01),
            child: RotationTransition(
              turns: Tween(begin: startValue, end: endValue)
                  .animate(_controller),
              child: Image(
                image: AssetImage('assets/images/wheel.png'),
```

```
                height: screenHeight * 0.4,
              ),
            ),
          ),
          Padding(
              padding: EdgeInsets.all(screenHeight * 0.01),
              child: Center(
                  child: Text(
                wheelRotated
                    ? 'Number is ${num.value} !!'
                    : 'Rotate the Wheel',
                style: TextStyle(
                    color: wheelRotated
                        ? num.color == 'Red'
                            ? Colors.red
                            : num.color == 'Green'
                                ? Colors.green
                                : Colors.black
                        : Theme.of(context).primaryColor,
                    fontWeight: FontWeight.bold,
                    fontSize: 28),
              ))),
        ],
      ),
    ),
    floatingActionButton: wheelRotating
        ? null
        : FloatingActionButton(
            onPressed: _playRoulette,
            tooltip: 'Play',
            child: Icon(Icons.play_arrow),
          ),
  );
 }
}
```

./lib/models/number.dart:

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

class Number {
 int value;
 String color;
 Number({@required this.value, @required this.color});
}
```

./lib/constants/numbers.dart:

```dart
import 'package:roulette/models/number.dart';

List<Number> numbers = [
 Number(color: 'Green', value: 0),
 Number(color: 'Black', value: 26),
 Number(color: 'Red', value: 3),
 Number(color: 'Black', value: 35),
 Number(color: 'Red', value: 12),
 Number(color: 'Black', value: 28),
 Number(color: 'Red', value: 7),
 Number(color: 'Black', value: 29),
 Number(color: 'Red', value: 18),
 Number(color: 'Black', value: 22),
 Number(color: 'Red', value: 9),
 Number(color: 'Black', value: 31),
 Number(color: 'Red', value: 14),
 Number(color: 'Black', value: 20),
 Number(color: 'Red', value: 1),
 Number(color: 'Black', value: 33),
 Number(color: 'Red', value: 16),
 Number(color: 'Black', value: 24),
 Number(color: 'Red', value: 5),
 Number(color: 'Black', value: 10),
 Number(color: 'Red', value: 23),
 Number(color: 'Black', value: 8),
 Number(color: 'Red', value: 30),
 Number(color: 'Black', value: 11),
 Number(color: 'Red', value: 36),
 Number(color: 'Black', value: 13),
 Number(color: 'Red', value: 27),
 Number(color: 'Black', value: 6),
 Number(color: 'Red', value: 34),
 Number(color: 'Black', value: 17),
 Number(color: 'Red', value: 25),
 Number(color: 'Black', value: 2),
 Number(color: 'Red', value: 21),
 Number(color: 'Black', value: 4),
 Number(color: 'Red', value: 19),
 Number(color: 'Black', value: 15),
 Number(color: 'Red', value: 32),
];
```

Screenshots: