

DICE ROLLER

SMART BRIDGE- ANDROID APP DEVELOPMENT



21BCE8951

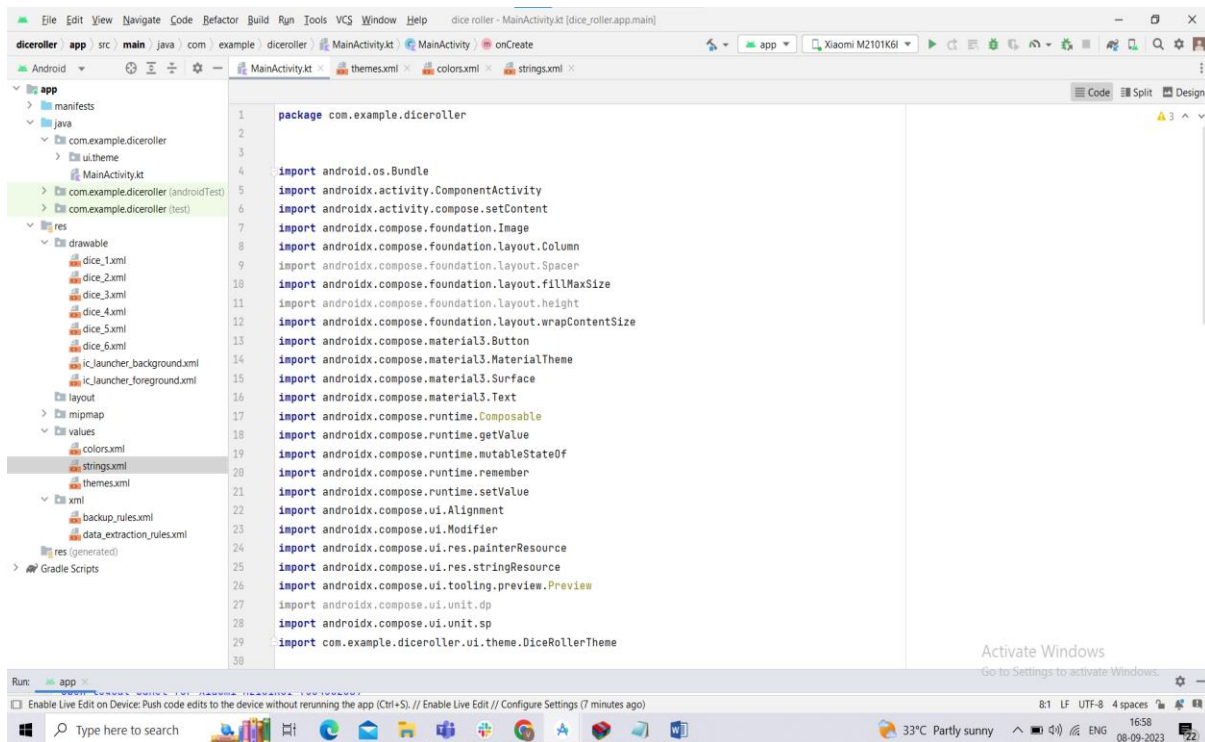
VANAPARTHI BULLI SIVA

VIT-AP, AMARAVATHI

SMART INTERNZ

ASSIGNMENT-2

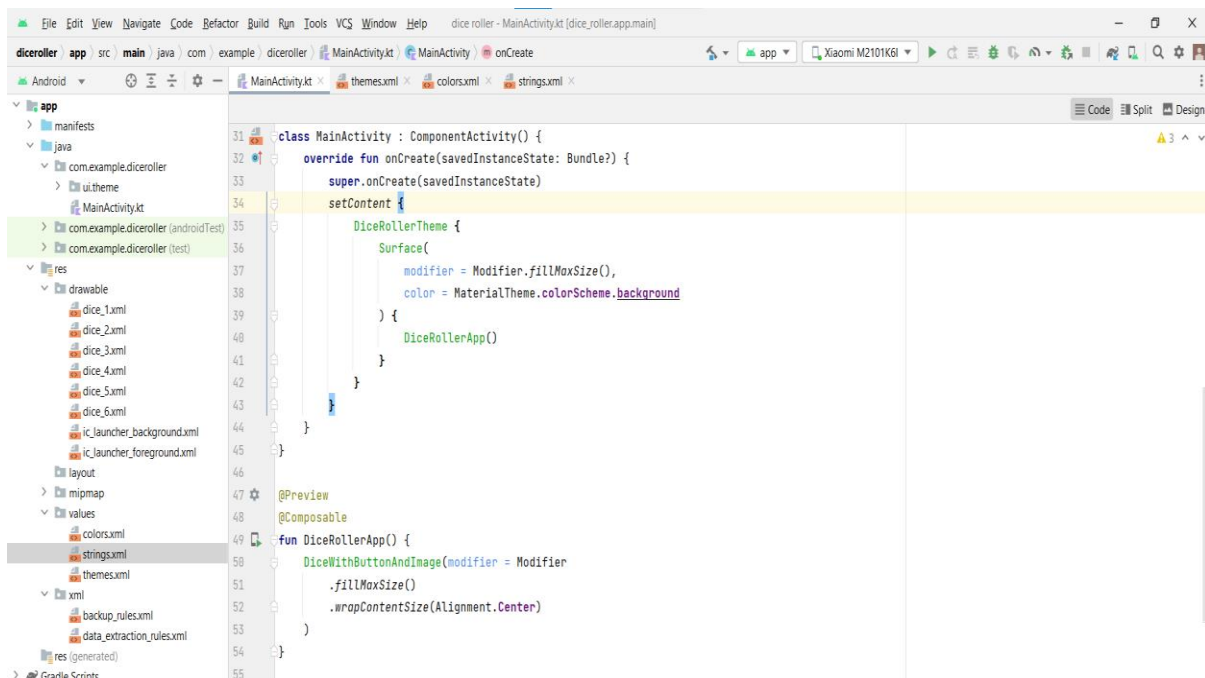
Snippets of the code



```

1 package com.example.diceroller
2
3
4 import android.os.Bundle
5 import androidx.activity.ComponentActivity
6 import androidx.activity.compose.setContent
7 import androidx.compose.foundation.Image
8 import androidx.compose.foundation.layout.Column
9 import androidx.compose.foundation.layout.Spacer
10 import androidx.compose.foundation.layout.fillMaxSize
11 import androidx.compose.foundation.layout.height
12 import androidx.compose.foundation.layout.wrapContentSize
13 import androidx.compose.material3.Button
14 import androidx.compose.material3.MaterialTheme
15 import androidx.compose.material3.Surface
16 import androidx.compose.material3.Text
17 import androidx.compose.runtime.Composable
18 import androidx.compose.runtime.getValue
19 import androidx.compose.runtime.mutableStateOf
20 import androidx.compose.runtime.remember
21 import androidx.compose.runtime.setValue
22 import androidx.compose.ui.Alignment
23 import androidx.compose.ui.Modifier
24 import androidx.compose.ui.res.painterResource
25 import androidx.compose.ui.res.stringResource
26 import androidx.compose.ui.tooling.preview.Preview
27 import androidx.compose.ui.unit.dp
28 import androidx.compose.ui.unit.sp
29 import com.example.diceroller.ui.theme.DiceRollerTheme
30

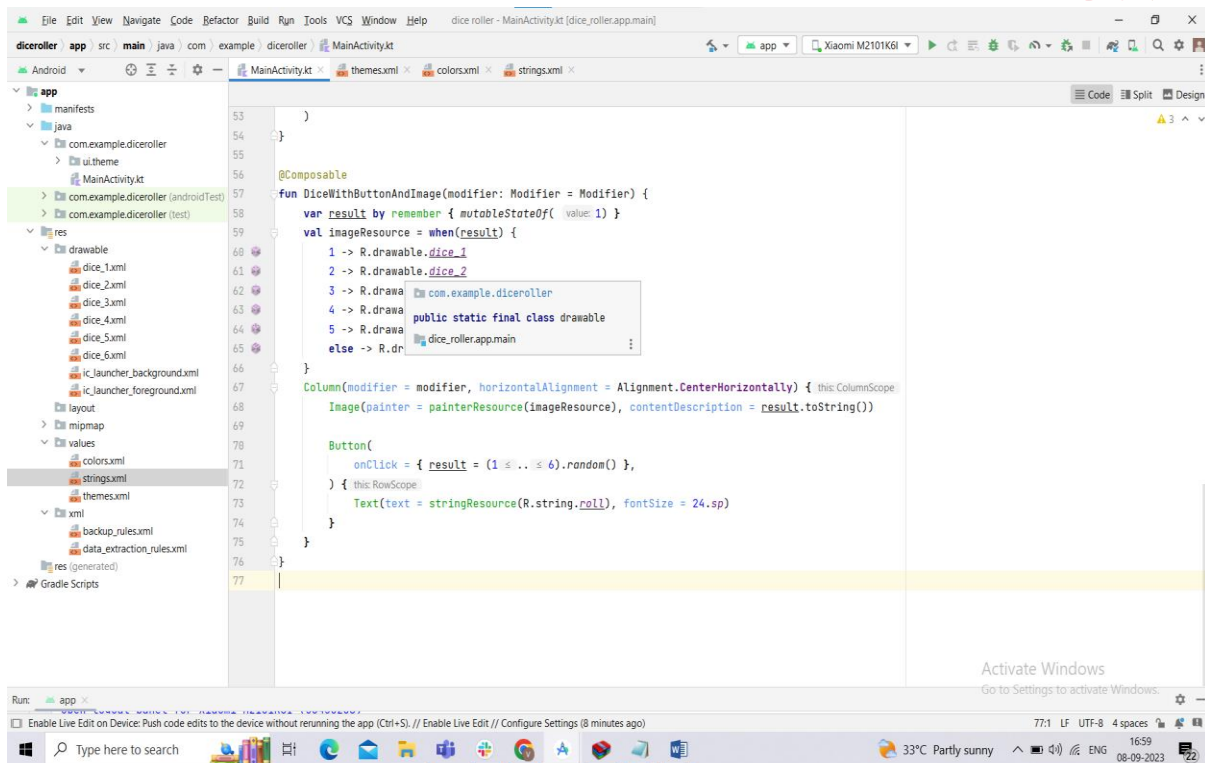
```



```

31 class MainActivity : ComponentActivity() {
32     override fun onCreate(savedInstanceState: Bundle?) {
33         super.onCreate(savedInstanceState)
34         setContent {
35             DiceRollerTheme {
36                 Surface(
37                     modifier = Modifier.fillMaxSize(),
38                     color = MaterialTheme.colorScheme.background
39                 ) {
40                     DiceRollerApp()
41                 }
42             }
43         }
44     }
45 }
46
47 @Preview
48 @Composable
49 fun DiceRollerApp() {
50     DiceWithButtonAndImage(modifier = Modifier
51         .fillMaxSize()
52         .wrapContentSize(Alignment.Center)
53     )
54 }
55

```



CODE:

```
package com.example.diceroller
```

```
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.Spacer
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.height
import androidx.compose.foundation.layout.wrapContentSize
import androidx.compose.material3.Button
import androidx.compose.material3.MaterialTheme
import androidx.compose.material3.Surface
import androidx.compose.material3.Text
import androidx.compose.runtime.Composable
import androidx.compose.runtime.getValue
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.remember
import androidx.compose.runtime.setValue
import androidx.compose.ui.Alignment
```

```

import androidx.compose.ui.Modifier
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.res.stringResource
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import com.example.diceroller.ui.theme.DiceRollerTheme

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            DiceRollerTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    DiceRollerApp()
                }
            }
        }
    }
}

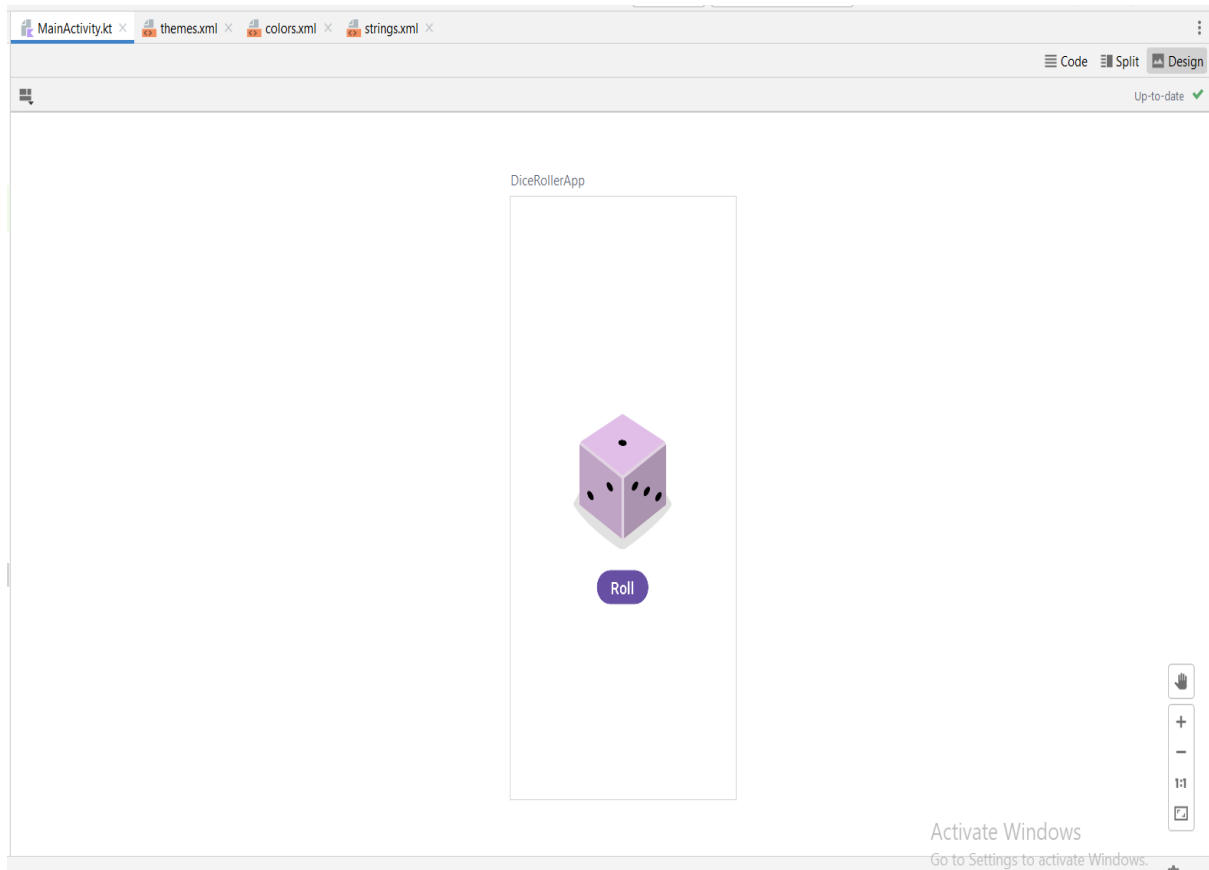
@Preview
@Composable
fun DiceRollerApp() {
    DiceWithButtonAndImage(modifier = Modifier
        .fillMaxSize()
        .wrapContentSize(Alignment.Center)
    )
}

@Composable
fun DiceWithButtonAndImage(modifier: Modifier = Modifier) {
    var result by remember { mutableStateOf( 1) }
    val imageResource = when(result) {
        1 -> R.drawable.dice_1
        2 -> R.drawable.dice_2
        3 -> R.drawable.dice_3
        4 -> R.drawable.dice_4
        5 -> R.drawable.dice_5
        else -> R.drawable.dice_6
    }
    Column(modifier = modifier, horizontalAlignment =
    Alignment.CenterHorizontally) {
        Image(painter = painterResource(imageResource), contentDescription
        = result.toString())

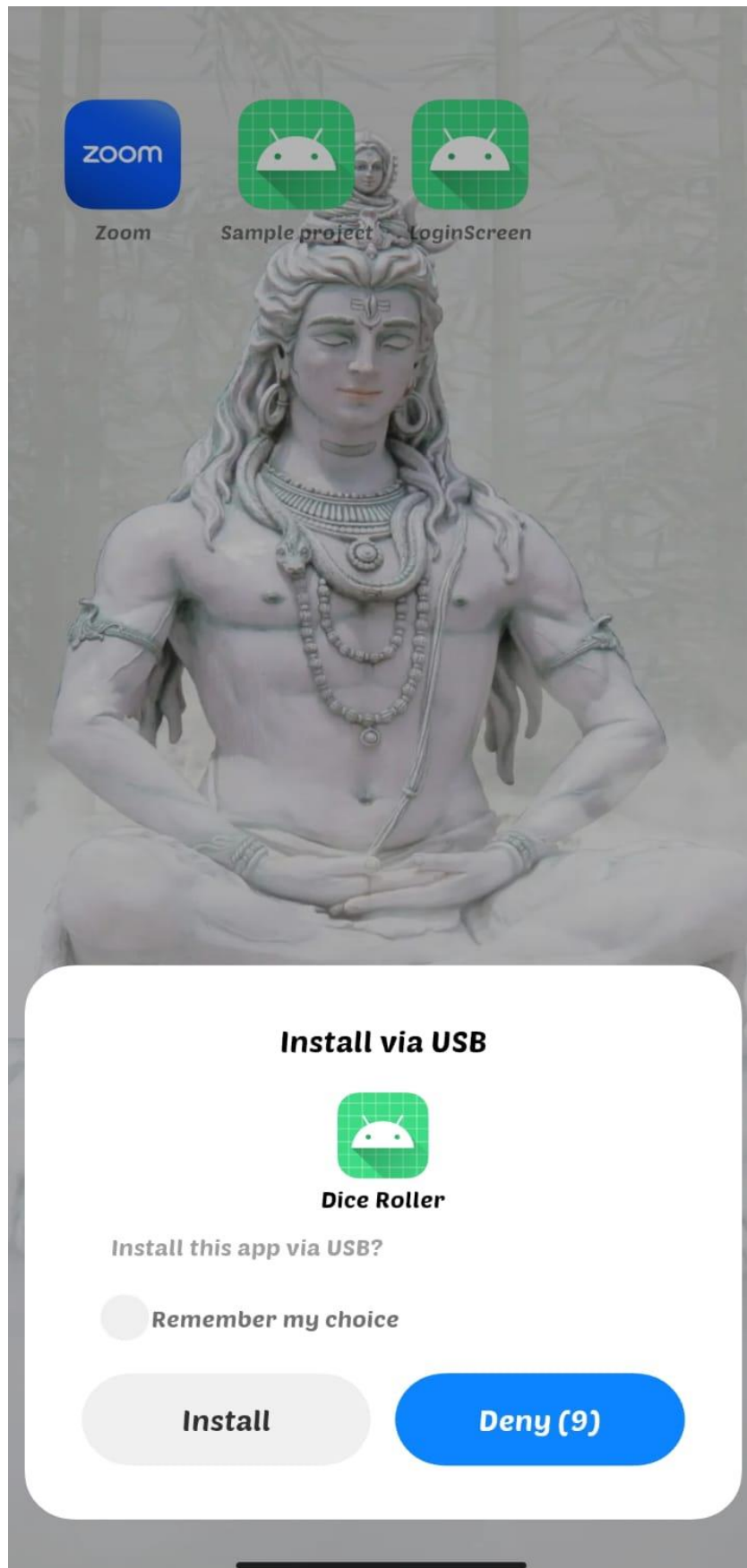
        Button(
            onClick = { result = (1..6).random() },
        ) {
            Text(text = stringResource(R.string.roll), fontSize = 24.sp)
        }
    }
}

```

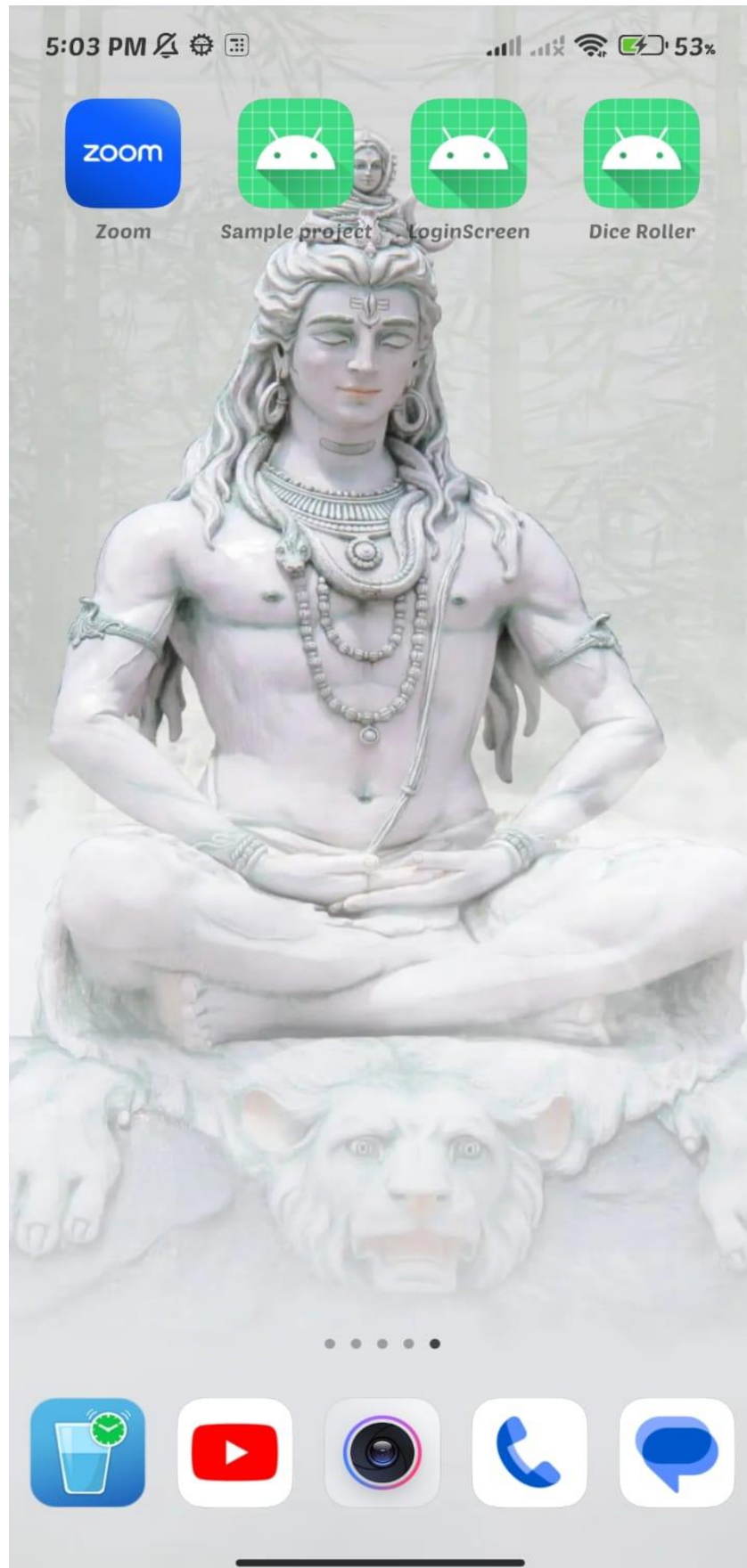
DESIGN



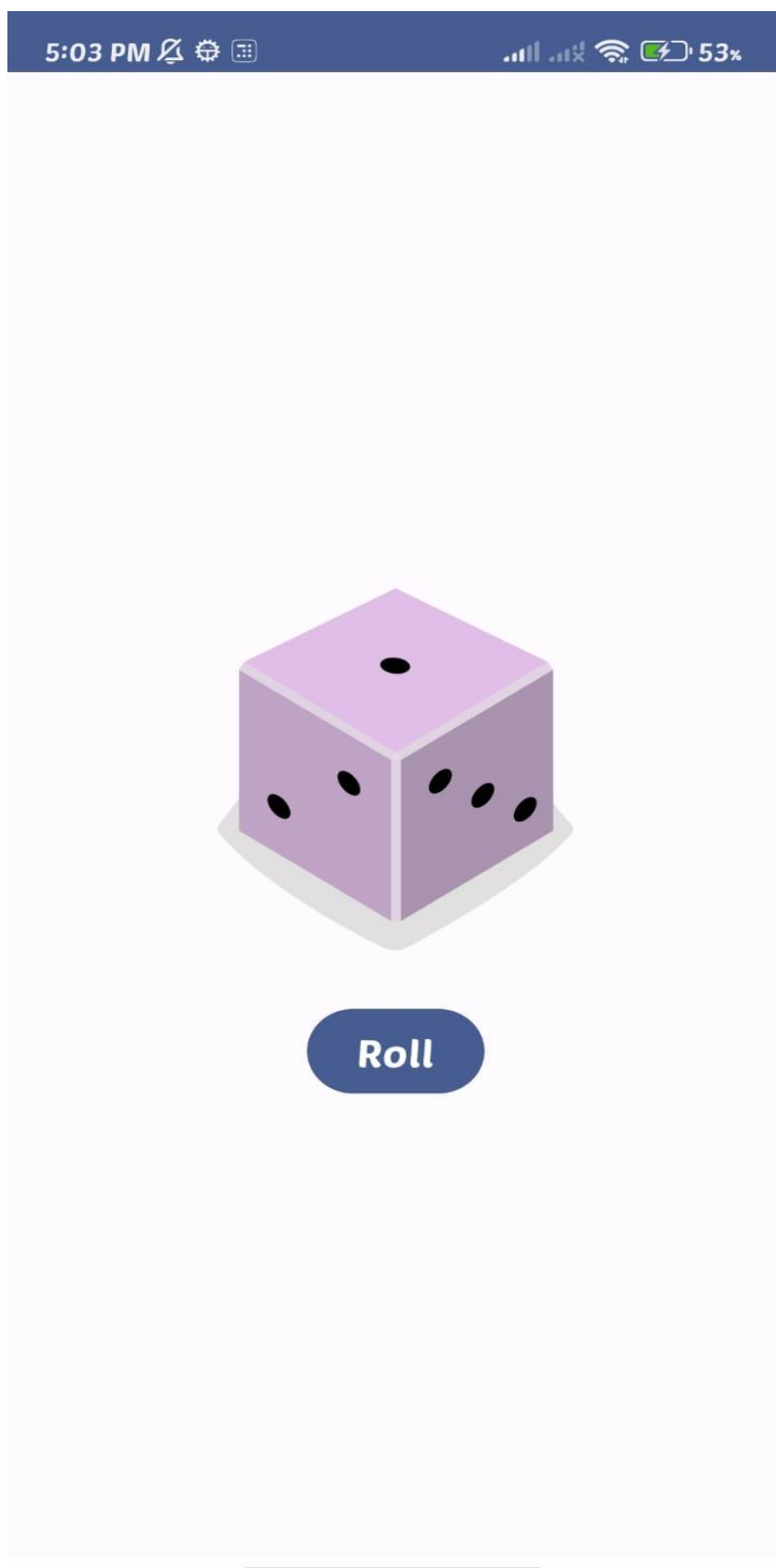
Installing the app into device via USB



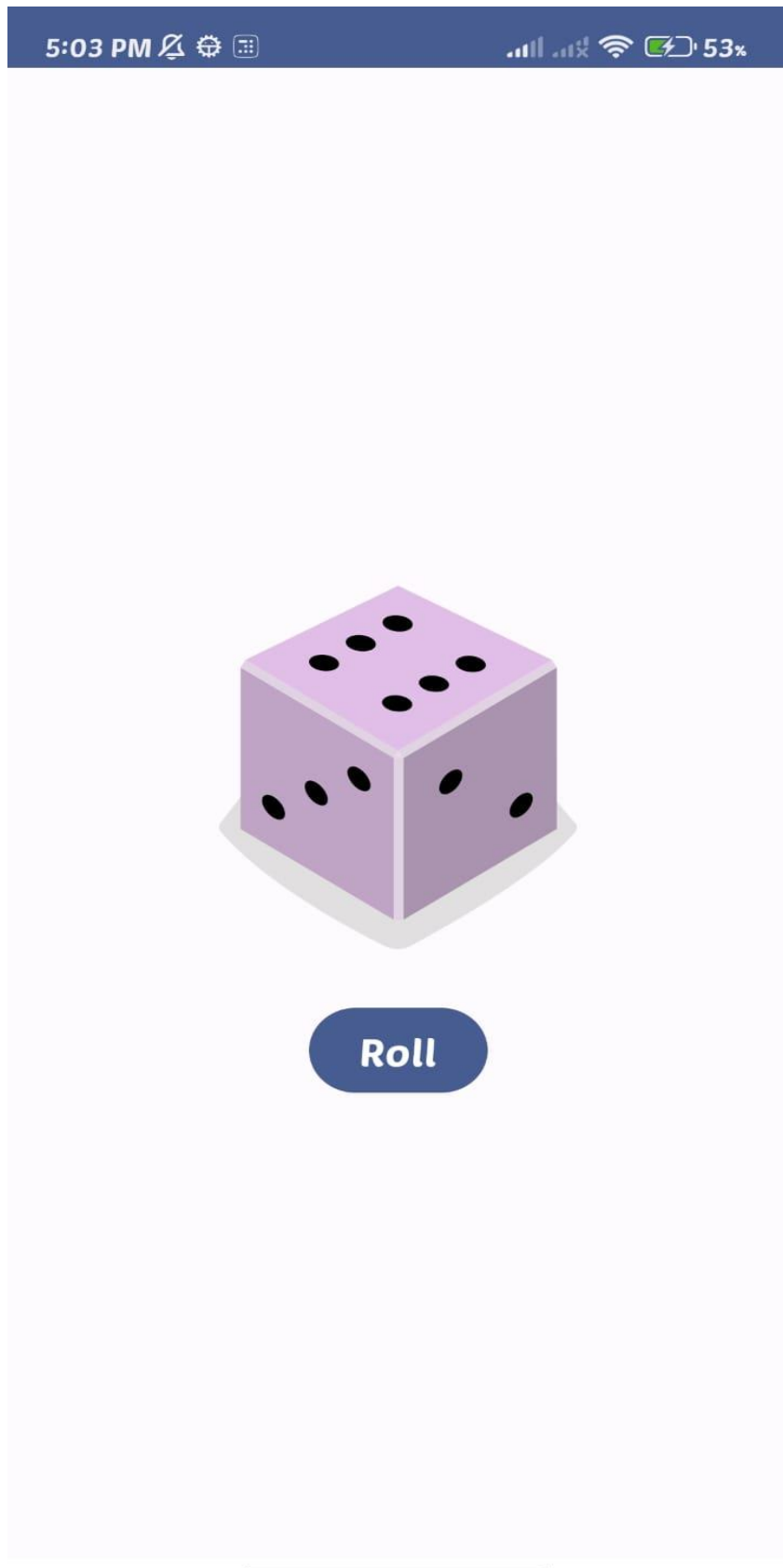
After Installation

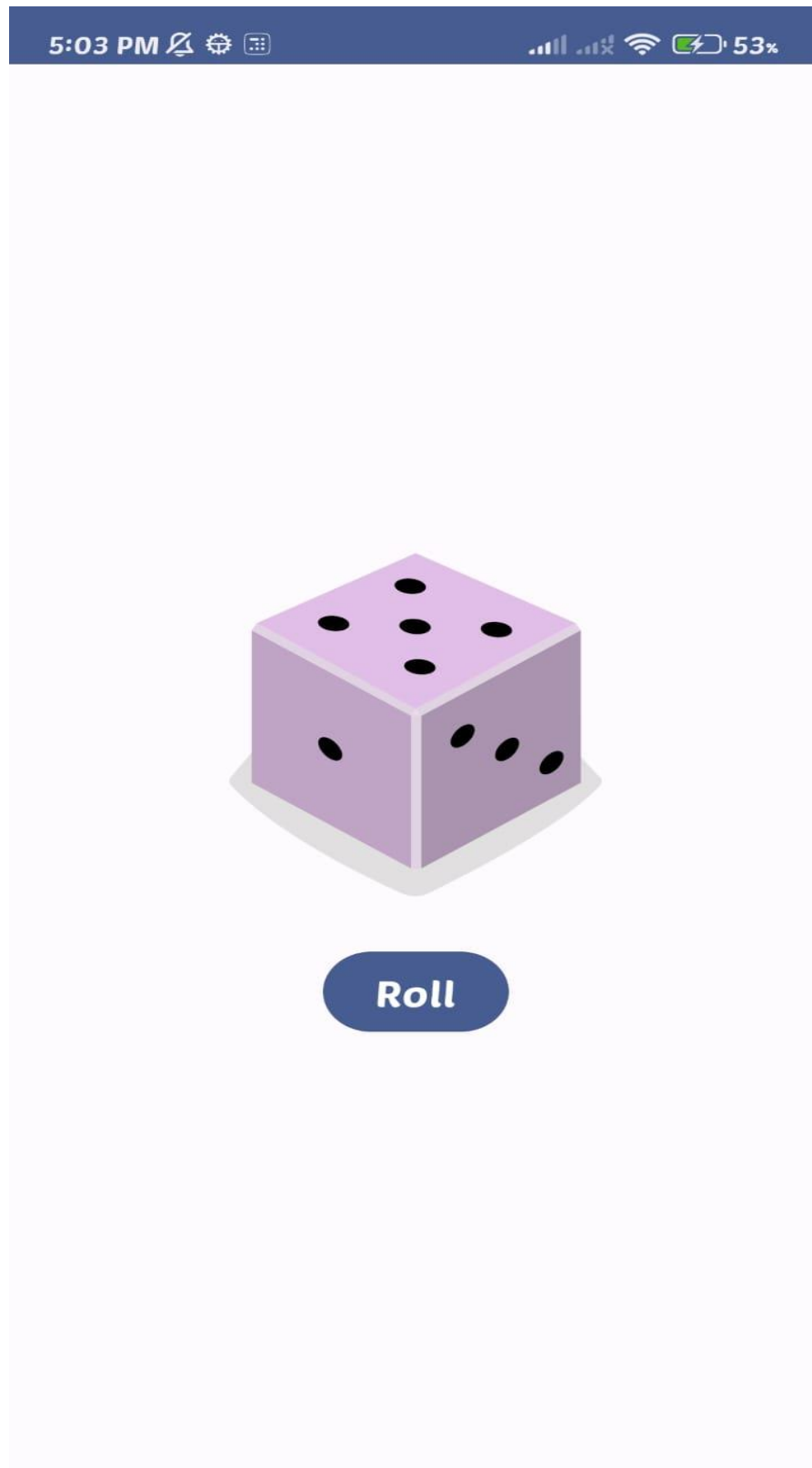


Intial Screen of the application



Rolling on clicking roll button. Random outputs are.





THE END