# **MOBILE APPLICATION DEVELOPNMENT LAB 2:**

**3x3 grid layout Code:**

This project is **Samavia's first Flutter demo**, where I explores **Flutter's UI design** and **state management** concepts. The app features:

✅ A **3x3 grid layout**, displaying different sections.  
✅ A **counter functionality** with **increment and decrement** buttons.  
✅ **Preventing negative values** by restricting the counter from going below zero.  
✅ **Visually appealing UI**, using different background colors for each grid element.  
✅ **Floating Action Buttons (FABs)** for user-friendly interactions.

This project serves as a **great introduction to Flutter**, covering **widgets, stateful widgets, event handling, and UI structuring**. 🚀

**Code:-**

import 'package:flutter/material.dart';  
  
void main() {  
 runApp(const MyApp());  
}  
  
class MyApp extends StatelessWidget {  
 const MyApp({super.key});  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 title: 'Flutter Demo',  
 theme: ThemeData(  
 colorScheme: ColorScheme.fromSeed(seedColor: Colors.*deepPurple*),  
 ),  
 home: const MyHomePage(title: "Samavia's First Flutter Demo Home Page"),  
 );  
 }  
}  
  
class MyHomePage extends StatefulWidget {  
 const MyHomePage({super.key, required this.title});  
 final String title;  
  
 @override  
 State<MyHomePage> createState() => \_MyHomePageState();  
}  
  
class \_MyHomePageState extends State<MyHomePage> {  
 int \_counter = 0;  
  
 void \_incrementCounter() {  
 setState(() {  
 \_counter++;  
 });  
 }  
  
 void \_decrementCounter() {  
 setState(() {  
 if (\_counter > 0) \_counter--; // Prevents negative counter values  
 });  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 backgroundColor: Theme.*of*(context).colorScheme.inversePrimary,  
 title: Text(widget.title),  
 ),  
 body: Center(  
 child: Column(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: List.generate(3, (rowIndex) {  
 return Row(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: List.generate(3, (colIndex) {  
 return Container(  
 margin: const EdgeInsets.all(8.0),  
 padding: const EdgeInsets.all(16.0),  
 decoration: BoxDecoration(  
 color: Colors.*primaries*[(rowIndex \* 3 + colIndex) % Colors.*primaries*.length],  
 borderRadius: BorderRadius.circular(10),  
 ),  
 child: Column(  
 children: <Widget>[  
 Text(  
 'Row ${rowIndex + 1}, Col ${colIndex + 1}',  
 style: const TextStyle(color: Colors.*white*),  
 ),  
 if (rowIndex == 2 && colIndex == 2)  
 Row(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 Text(  
 '$\_counter',  
 style: Theme.*of*(context)  
 .textTheme  
 .headlineMedium  
 ?.copyWith(color: Colors.*white*),  
 ),  
 const SizedBox(width: 10),  
 const Icon(Icons.*numbers*, size: 30, color: Colors.*white*),  
 ],  
 ),  
 ],  
 ),  
 );  
 }),  
 );  
 }),  
 ),  
 ),  
 floatingActionButton: Row(  
 mainAxisAlignment: MainAxisAlignment.end,  
 children: [  
 FloatingActionButton(  
 onPressed: \_decrementCounter,  
 tooltip: 'Decrement',  
 backgroundColor: Colors.*red*,  
 child: const Icon(Icons.*remove*),  
 ),  
 const SizedBox(width: 10),  
 FloatingActionButton(  
 onPressed: \_incrementCounter,  
 tooltip: 'Increment',  
 backgroundColor: Colors.*green*,  
 child: const Icon(Icons.*add*),  
 ),  
 ],  
 ),  
 );  
 }  
}

A screenshot of a computer

AI-generated content may be incorrect.

**Reusable codes:**

**With counter feature**

**"Samavia's First Mobile App Journey 🚀"** is an interactive Flutter app that visually represents key **Mobile App Development concepts**—UI/UX, Backend, and Database—using a **reusable widget structure**. Each category is displayed with relevant icons in a **dynamic layout**. Additionally, the app includes a **counter feature** with **increment and decrement buttons**, allowing users to interact with the app while maintaining **clean, efficient, and reusable code**. This project showcases core Flutter concepts like **Stateless and Stateful widgets, UI design, event handling, and code reusability**—all essential skills for scalable app development!

**Code:-**

import 'package:flutter/material.dart';  
  
void main() {  
 runApp(MyApp());  
}  
  
class MyApp extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 debugShowCheckedModeBanner: false,  
 home: MyHomePage(),  
 );  
 }  
}  
  
class MyHomePage extends StatefulWidget {  
 @override  
 \_MyHomePageState createState() => \_MyHomePageState();  
}  
  
class \_MyHomePageState extends State<MyHomePage> {  
 int \_counter = 0;  
  
 void \_incrementCounter() {  
 setState(() {  
 \_counter++;  
 });  
 }  
  
 void \_decrementCounter() {  
 setState(() {  
 if (\_counter > 0) \_counter--;  
 });  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(title: Text("Samavia's First Mobile App Journey 🚀")),  
 body: Column(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 ReusableRow('UI/UX', Icons.*design\_services*, Icons.*palette*, Icons.*touch\_app*),  
 ReusableRow('Backend', Icons.*code*, Icons.*storage*, Icons.*sync*),  
 ReusableRow('Database', Icons.*dataset*, Icons.*security*, Icons.*cloud*),  
 SizedBox(height: 20),  
 Text(  
 'Counter: $\_counter',  
 style: TextStyle(fontSize: 24, fontWeight: FontWeight.*bold*),  
 ),  
 ],  
 ),  
 floatingActionButton: Row(  
 mainAxisAlignment: MainAxisAlignment.end,  
 children: [  
 FloatingActionButton(  
 onPressed: \_decrementCounter,  
 tooltip: 'Decrement',  
 child: Icon(Icons.*remove*),  
 ),  
 SizedBox(width: 10),  
 FloatingActionButton(  
 onPressed: \_incrementCounter,  
 tooltip: 'Increment',  
 child: Icon(Icons.*add*),  
 ),  
 ],  
 ),  
 );  
 }  
}  
  
class ReusableRow extends StatelessWidget {  
 final String text;  
 final IconData icon1, icon2, icon3;  
  
 ReusableRow(this.text, this.icon1, this.icon2, this.icon3);  
  
 @override  
 Widget build(BuildContext context) {  
 return Padding(  
 padding: const EdgeInsets.all(10.0),  
 child: Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
 buildColumn(icon1, Colors.*blue*),  
 buildColumn(icon2, Colors.*green*),  
 buildColumn(icon3, Colors.*orange*),  
 ],  
 ),  
 );  
 }  
  
 Widget buildColumn(IconData icon, Color iconColor) {  
 return Column(  
 children: [  
 Icon(icon, size: 40, color: iconColor),  
 SizedBox(height: 10),  
 Text(  
 text,  
 style: TextStyle(fontSize: 18, fontWeight: FontWeight.*bold*, color: Colors.*black*),  
 ),  
 ],  
 );  
 }  
}

**Output:-**

A screenshot of a computer

AI-generated content may be incorrect.

**Without counter app**

This Flutter project represents **Samavia's first step into mobile app development**, showcasing **core app architecture concepts** through a **structured UI layout**. The app highlights:

✅ A **clean and interactive UI**, categorizing key development areas: **UI/UX, Backend, and Database**.  
✅ **Reusable components**, ensuring efficiency and maintainability in code structure.  
✅ **Icon-based representation**, making concepts visually intuitive and engaging.  
✅ **Flutter's powerful widget system**, including **Stateless Widgets** for a structured and scalable app.

This project serves as a **solid foundation for future mobile app development**, reinforcing best practices in **UI design, reusable coding, and efficient layouts**. ✨📱

**Code:-**

import 'package:flutter/material.dart';  
  
void main() {  
 runApp(MyApp());  
}  
  
class MyApp extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 debugShowCheckedModeBanner: false,  
 home: Scaffold(  
 appBar: AppBar(title: Text("Samavia's First Mobile App Journey 🚀")),  
 body: Column(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 ReusableRow('UI/UX', Icons.*design\_services*, Icons.*palette*, Icons.*touch\_app*),  
 ReusableRow('Backend', Icons.*code*, Icons.*storage*, Icons.*sync*),  
 ReusableRow('Database', Icons.*dataset*, Icons.*security*, Icons.*cloud*),  
 ],  
 ),  
 ),  
 );  
 }  
}  
  
class ReusableRow extends StatelessWidget {  
 final String text;  
 final IconData icon1, icon2, icon3;  
  
 ReusableRow(this.text, this.icon1, this.icon2, this.icon3);  
  
 @override  
 Widget build(BuildContext context) {  
 return Padding(  
 padding: const EdgeInsets.all(10.0),  
 child: Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
 buildColumn(icon1, Colors.*blue*),  
 buildColumn(icon2, Colors.*green*),  
 buildColumn(icon3, Colors.*orange*),  
 ],  
 ),  
 );  
 }  
  
 Widget buildColumn(IconData icon, Color iconColor) {  
 return Column(  
 children: [  
 Icon(icon, size: 40, color: iconColor),  
 SizedBox(height: 10),  
 Text(  
 text,  
 style: TextStyle(fontSize: 18, fontWeight: FontWeight.*bold*, color: Colors.*black*),  
 ),  
 ],  
 );  
 }  
}

**Output:-**

