# Mohamed Mostafa ElMaghraby

# 20p7732

# Mobile programming

# Milestone 1

# Project components overview:

* **MaterialApp:** Provides the main application structure with defined routes.
* **Scaffold:** Basic layout structure with AppBar, body, and FloatingActionButton.
* Screens and Pages
* **LoadingScreen:** Simple page with a loading animation.
* **LoginPage:** User login interface with text fields and a login button.
* **SignUpPage:** Registration page with fields for user details.
* **HomePage:** Main navigation hub for the app.
* **EventListPage**: Lists events with sorting options and provides access to the Gift List.
* **AppBar:** Provides a title, a profile icon, and navigation options.
* **CircleAvatar:** Displays a profile picture icon in the AppBar.
* **DropdownButton:** Sorting options (by name, category, or status).
* **ListView.builder:** Dynamically generates a list of events.
* **Card:** Displays each event item in a visually distinct container.
* **ListTile:** Holds the event details (title, subtitle, and trailing actions).
* **IconButton:** Edit and delete options for each event item.
* **FloatingActionButton.extended:** Button to add a new event.
* Styling and Theming
* **BoxDecoration with LinearGradient:** Background styling for gradients.
* **TextStyle:** Customizes font color and style across various text elements.
* **IconThemeData:** Configures icon color in the AppBar.
* **Colors and Color.fromARGB:** Provides color consistency for backgrounds, icons, and other UI elements.

# Screenshots from the UI:

## 1-Loading Screen:

A screenshot of a phone

Description automatically generated

## A screenshot of a login screen Description automatically generated2-Login Page

## 3-Sign up page:

A screenshot of a phone

Description automatically generated

## 4-HomePage

A screenshot of a phone

Description automatically generated

## A screenshot of a phone Description automatically generated

## 5-Event List:

## 6- Gifts List:

A screenshot of a phone

Description automatically generated

# Dart Code:

## 1-Loading\_screen.dart:

import 'package:flutter/material.dart';

import 'package:flutter\_spinkit/flutter\_spinkit.dart';

import 'package:hedieaty/title\_widget.dart';

class LoadingScreen extends StatefulWidget {

  const LoadingScreen({super.key});

  @override

  State<LoadingScreen> createState() => \_LoadingScreenState();

}

class \_LoadingScreenState extends State<LoadingScreen> {

  @override

  void initState() {

    super.initState();

    // Navigate to HomeScreen after 3 seconds

    Future.delayed(const Duration(seconds: 3), () {

      Navigator.pushReplacementNamed(context, '/login');

    });

  }

  @override

  Widget build(BuildContext context) {

    return const Scaffold(

      backgroundColor: Color.fromARGB(255, 58, 2, 80),

      body: Center(

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          children: [

            TitleWidget(),

            SizedBox(height: 20),

            Text(

              "Gifting made easy!",

              style: TextStyle(

                fontFamily: 'arima',

                fontSize: 18.0,

                color: Colors.white,

                fontStyle: FontStyle.italic,

              ),

            ),

            SizedBox(height: 50),

            SpinKitSpinningLines(

              color: Colors.amber,

              size: 50.0,

            ),

            SizedBox(height: 20),

            Text(

              'Loading...',

              style: TextStyle(

                fontSize: 18.0,

                color: Colors.white,

              ),

            ),

          ],

        ),

      ),

    );

  }

}

## 2-Main.dart:

import 'package:flutter/material.dart';

import 'package:hedieaty/loading\_screen.dart';

import 'package:hedieaty/sign\_up.dart';

import 'package:hedieaty/login.dart';

import 'package:hedieaty/home\_page.dart';

import 'package:hedieaty/event\_list.dart';

import 'package:hedieaty/gift\_list.dart';

void main() {

  runApp(MaterialApp(

    initialRoute: '/loading',

    routes: {

      '/loading': (context) => const LoadingScreen(),

      '/login': (context) => const LoginPage(),

      '/signup': (context) => const SignUpPage(),

      '/home': (context) => const HomePage(),

      '/eventListPage': (context) => const EventListPage(),

      '/giftListPage': (context) => const GiftListPage(),

    },

  ));

}

3-Login.dart:  
import 'package:flutter/material.dart';

import 'package:hedieaty/title\_widget.dart';

class LoginPage extends StatefulWidget {

  const LoginPage({super.key});

  @override

  State<LoginPage> createState() => \_LoginPageState();

}

class \_LoginPageState extends State<LoginPage> {

  final GlobalKey<FormState> loginKey = GlobalKey();

  final TextEditingController emailController = TextEditingController();

  final TextEditingController passwordController = TextEditingController();

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: const Color.fromARGB(255, 58, 2, 80),

      body: Center(

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          children: [

            const TitleWidget(),

            const SizedBox(height: 60),

            Container(

              width: 280,

              padding: const EdgeInsets.all(20.0),

              decoration: BoxDecoration(

                color: const Color.fromARGB(255, 58, 2, 80),

                border: Border.all(color: Colors.white),

                borderRadius: BorderRadius.circular(10.0),

              ),

              child: Form(

                key: loginKey,

                child: Column(

                  mainAxisSize: MainAxisSize.min,

                  children: [

                    const Text(

                      "Login",

                      style: TextStyle(

                        fontSize: 28,

                        fontWeight: FontWeight.bold,

                        color: Colors.white,

                      ),

                    ),

                    const SizedBox(height: 16.0),

                    TextFormField(

                      controller: emailController,

                      decoration: const InputDecoration(

                        labelText: 'Enter your Email...',

                        border: OutlineInputBorder(

                          borderRadius: BorderRadius.all(Radius.circular(12.0)),

                        ),

                        filled: true,

                        fillColor: Colors.white,

                      ),

                      keyboardType: TextInputType.emailAddress,

                      validator: (value) {

                        if (value == null || value.isEmpty) {

                          return 'Please enter your email';

                        }

                        return null;

                      },

                    ),

                    const SizedBox(height: 16.0),

                    TextFormField(

                      controller: passwordController,

                      decoration: const InputDecoration(

                        labelText: 'Enter your password...',

                        border: OutlineInputBorder(

                          borderRadius: BorderRadius.all(Radius.circular(12.0)),

                        ),

                        hintText: 'Enter your password',

                        hintStyle: TextStyle(color: Color.fromARGB(255, 200, 200, 200)),

                        filled: true,

                        fillColor: Colors.white,

                      ),

                      obscureText: true,

                      validator: (value) {

                        if (value == null || value.isEmpty) {

                          return 'Please enter your password';

                        }

                        return null;

                      },

                    ),

                    const SizedBox(height: 20.0),

                    ElevatedButton(

                      style: ElevatedButton.styleFrom(

                        backgroundColor: Colors.amber,

                        foregroundColor: Colors.black,

                      ),

                      onPressed: () {

                        if (loginKey.currentState?.validate() ?? false) {

                          Navigator.pushReplacementNamed(context, '/home');

                        }

                      },

                      child: const Text('Login', style: TextStyle(fontSize: 16),),

                    ),

                    const SizedBox(height: 10),

                    GestureDetector(

                      onTap: () {

                        Navigator.pushReplacementNamed(context, '/signup');

                      },

                      child: const Text(

                        "Don't have an account? Sign up!",

                        style: TextStyle(

                          color: Colors.amber,

                          decoration: TextDecoration.underline,

                        ),

                      ),

                    ),

                  ],

                ),

              ),

            ),

          ],

        ),

      ),

    );

  }

}

## 4-Sign\_up.dart:

import 'package:flutter/material.dart';

import 'package:hedieaty/title\_widget.dart';

class SignUpPage extends StatefulWidget {

  const SignUpPage({super.key});

  @override

  State<SignUpPage> createState() => \_SignUpPageState();

}

class \_SignUpPageState extends State<SignUpPage> {

  final GlobalKey<FormState> signUpKey = GlobalKey();

  final TextEditingController nameController = TextEditingController();

  final TextEditingController phoneController = TextEditingController();

  final TextEditingController emailController = TextEditingController();

  final TextEditingController passwordController = TextEditingController();

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: const Color.fromARGB(255, 58, 2, 80),

      body: Center(

        child: Column(

          mainAxisAlignment: MainAxisAlignment.center,

          children: [

            const TitleWidget(),

            const SizedBox(height: 60),

            Container(

              width: 280,

              padding: const EdgeInsets.all(20.0),

              decoration: BoxDecoration(

                color: const Color.fromARGB(255, 58, 2, 80),

                border: Border.all(color: Colors.white),

                borderRadius: BorderRadius.circular(10.0),

              ),

              child: Form(

                key: signUpKey,

                child: Column(

                  mainAxisSize: MainAxisSize.min,

                  children: [

                    const Text(

                      "Register",

                      style: TextStyle(

                        fontSize: 24,

                        fontWeight: FontWeight.bold,

                        color: Colors.white,

                      ),

                    ),

                    const SizedBox(height: 16.0),

                    TextFormField(

                      controller: nameController,

                      decoration: const InputDecoration(

                        labelText: 'Enter your name...',

                        border: OutlineInputBorder(

                          borderRadius: BorderRadius.all(Radius.circular(12.0)),

                        ),

                        hintText: 'Enter your name',

                        hintStyle: TextStyle(color: Colors.grey),

                        filled: true,

                        fillColor: Colors.white,

                      ),

                      validator: (value) {

                        if (value == null || value.isEmpty) {

                          return 'Please enter your name';

                        }

                        return null;

                      },

                    ),

                    const SizedBox(height: 16.0),

                    TextFormField(

                      controller: phoneController,

                      decoration: const InputDecoration(

                        labelText: 'Enter your phone number...',

                        border: OutlineInputBorder(

                          borderRadius: BorderRadius.all(Radius.circular(12.0)),

                        ),

                        hintText: 'Enter your phone number',

                        hintStyle: TextStyle(color: Colors.grey),

                        filled: true,

                        fillColor: Colors.white,

                      ),

                      keyboardType: TextInputType.phone,

                      validator: (value) {

                        if (value == null || value.isEmpty) {

                          return 'Please enter your phone number';

                        }

                        return null;

                      },

                    ),

                    const SizedBox(height: 16.0),

                    TextFormField(

                      controller: emailController,

                      decoration: const InputDecoration(

                        labelText: 'Enter your Email...',

                        border: OutlineInputBorder(

                          borderRadius: BorderRadius.all(Radius.circular(12.0)),

                        ),

                        hintText: 'Enter your email',

                        hintStyle: TextStyle(color: Colors.grey),

                        filled: true,

                        fillColor: Colors.white,

                      ),

                      keyboardType: TextInputType.emailAddress,

                      validator: (value) {

                        if (value == null || value.isEmpty) {

                          return 'Please enter your email';

                        }

                        return null;

                      },

                    ),

                    const SizedBox(height: 16.0),

                    TextFormField(

                      controller: passwordController,

                      decoration: const InputDecoration(

                        labelText: 'Enter your password...',

                        border: OutlineInputBorder(

                          borderRadius: BorderRadius.all(Radius.circular(12.0)),

                        ),

                        hintText: 'Enter your password',

                        hintStyle: TextStyle(color: Colors.grey),

                        filled: true,

                        fillColor: Colors.white,

                      ),

                      obscureText: true,

                      validator: (value) {

                        if (value == null || value.isEmpty) {

                          return 'Please enter your password';

                        }

                        return null;

                      },

                    ),

                    const SizedBox(height: 20.0),

                    ElevatedButton(

                      style: ElevatedButton.styleFrom(

                        backgroundColor: Colors.amber,

                        foregroundColor: Colors.black,

                      ),

                      onPressed: () {

                        if (signUpKey.currentState?.validate() ?? false) {

                          Navigator.pushReplacementNamed(context, '/home');

                        }

                      },

                      child: const Text('Sign Up', style: TextStyle(fontSize: 16),),

                    ),

                    const SizedBox(height: 10),

                    GestureDetector(

                      onTap: () {

                        Navigator.pushReplacementNamed(context, '/login');

                      },

                      child: const Text(

                        "Already have an account? Login!",

                        style: TextStyle(

                          color: Colors.amber,

                          decoration: TextDecoration.underline,

                        ),

                      ),

                    ),

                  ],

                ),

              ),

            ),

          ],

        ),

      ),

    );

  }

}

## 5-Title\_widget.dart:

import 'package:flutter/material.dart';

class TitleWidget extends StatelessWidget {

  const TitleWidget({super.key});

  @override

  Widget build(Object context) {

    return const Row(

      mainAxisAlignment: MainAxisAlignment.center,

      children: [

        Text(

          'Hedieaty',

          style: TextStyle(

            fontSize: 45.0,

            color: Color(0xFFF8F8FF),

            fontWeight: FontWeight.bold,

            fontFamily: "gvibes",

          ),

        ),

        SizedBox(width: 10),

        Icon(

          Icons.card\_giftcard,

          color: Color.fromARGB(255, 248, 220, 9),

          size: 35.0,

        ),

      ],

    );

  }

}

## 6-Homepage.dart:

import 'package:flutter/material.dart';

import 'title\_widget.dart';

import 'package:hedieaty/event\_list.dart';

class HomePage extends StatefulWidget {

  const HomePage({super.key});

  @override

  State<HomePage> createState() => \_HomePageState();

}

class \_HomePageState extends State<HomePage> {

  final List<Map<String, dynamic>> friends = [

    {

      'name': 'Alice',

      'profilePic': 'assets/images/profile\_icon.png',

      'phone': '+123456789',

      'events': 2,

    },

    {

      'name': 'Bob',

      'profilePic': 'assets/images/profile\_icon.png',

      'phone': '+987654321',

      'events': 0,

    },

    {

      'name': 'Charlie',

      'profilePic': 'assets/images/profile\_icon.png',

      'phone': '+456123789',

      'events': 1,

    }

  ];

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        toolbarHeight: 80,

        backgroundColor: const Color.fromARGB(255, 58, 2, 80),

        title: const TitleWidget(),

        actions: [

          GestureDetector(

            onTap: () {

              Navigator.pushNamed(context, '/profile');

            },

            child: const Padding(

              padding: EdgeInsets.only(right: 16.0),

              child: CircleAvatar(

                backgroundImage: AssetImage('assets/images/profile\_icon.png'),

                radius: 18.0,

              ),

            ),

          ),

        ],

      ),

      body: Container(

        decoration: const BoxDecoration(

          gradient: LinearGradient(

            colors: [

              Color.fromARGB(255, 58, 2, 80),

              Color.fromARGB(255, 219, 144, 5),

            ],

            begin: Alignment.topLeft,

            end: Alignment.bottomRight,

          ),

        ),

        padding: const EdgeInsets.all(16.0),

        child: Column(

          crossAxisAlignment: CrossAxisAlignment.center,

          children: [

            const Text(

              'Welcome, User!',

              style: TextStyle(

                fontSize: 24,

                fontWeight: FontWeight.bold,

                color: Colors.white,

              ),

            ),

            const SizedBox(height: 20),

            Row(

              children: [

                Expanded(

                  child: ElevatedButton.icon(

                    style: ElevatedButton.styleFrom(

                      backgroundColor: Colors.amber,

                      foregroundColor: Colors.black,

                      minimumSize: const Size.fromHeight(48),

                    ),

                    icon: const Icon(Icons.person\_add),

                    label: const Text("Add Friend"),

                    onPressed: () {

                    },

                  ),

                ),

                const SizedBox(width: 10),

                Expanded(

                  child: ElevatedButton.icon(

                    style: ElevatedButton.styleFrom(

                      backgroundColor: Colors.amber,

                      foregroundColor: Colors.black,

                      minimumSize: const Size.fromHeight(48),

                    ),

                    icon: const Icon(Icons.add\_circle\_outline),

                    label: const Text(

                      'Create Your Own Event/List',

                      style: TextStyle(fontSize: 16),

                    ),

                    onPressed: () {

                      Navigator.pushNamed(context, '/createEvent');

                    },

                  ),

                ),

              ],

            ),

            const SizedBox(height: 20),

            TextField(

              decoration: InputDecoration(

                hintText: 'Search for friends’ gift lists...',

                prefixIcon: const Icon(Icons.search),

                filled: true,

                fillColor: Colors.white,

                border: OutlineInputBorder(

                  borderRadius: BorderRadius.circular(15.0),

                  borderSide: BorderSide.none,

                ),

              ),

              onChanged: (value) {},

            ),

            const SizedBox(height: 20),

            Expanded(

              child: ListView.builder(

                itemCount: friends.length,

                itemBuilder: (context, index) {

                  final friend = friends[index];

                  return Card(

                    color: Colors.grey[200],

                    shape: RoundedRectangleBorder(

                      borderRadius: BorderRadius.circular(15.0),

                    ),

                    margin: const EdgeInsets.symmetric(vertical: 8.0),

                    child: ListTile(

                      leading: CircleAvatar(

                        backgroundColor: Colors.white,

                        backgroundImage: AssetImage(friend['profilePic']),

                      ),

                      title: Text(

                        friend['name'],

                        style: const TextStyle(

                          color: Color.fromARGB(255, 58, 2, 80),

                          fontWeight: FontWeight.bold,

                        ),

                      ),

                      subtitle: Text(

                        friend['phone'],

                        style: const TextStyle(color: Colors.black54),

                      ),

                      trailing: friend['events'] > 0

                          ? CircleAvatar(

                              radius: 12,

                              backgroundColor: Colors.amber,

                              child: Text(

                                friend['events'].toString(),

                                style: const TextStyle(

                                  color: Colors.black,

                                  fontWeight: FontWeight.bold,

                                ),

                              ),

                            )

                          : null,

                      onTap: () {

                        Navigator.push(

                          context,

                          MaterialPageRoute(

                            builder: (context) =>

                                EventListPage(friendName: friend['name']),

                          ),

                        );

                      },

                    ),

                  );

                },

              ),

            ),

          ],

        ),

      ),

    );

  }

}

7-event\_list.dart:  
import 'package:flutter/material.dart';

import 'package:intl/intl.dart';

class EventListPage extends StatefulWidget {

  final String? friendName;

  const EventListPage({super.key, this.friendName});

  @override

  State<EventListPage> createState() => \_EventListPageState();

}

class \_EventListPageState extends State<EventListPage> {

  List<Map<String, dynamic>> events = [

    {

      'name': 'Birthday Party',

      'category': 'Social',

      'status': 'Upcoming',

      'date': DateTime(2024, 1, 15),

    },

    {

      'name': 'Wedding',

      'category': 'Family',

      'status': 'Past',

      'date': DateTime(2023, 12, 22),

    },

    {

      'name': 'Office Meeting',

      'category': 'Work',

      'status': 'Current',

      'date': DateTime(2024, 2, 10),

    }

  ];

  String sortBy = 'name';

  void addEvent() {

  }

  void editEvent(int index) {

  }

  void deleteEvent(int index) {

    setState(() {

      events.removeAt(index);

    });

  }

  @override

  Widget build(BuildContext context) {

    events.sort((a, b) => a[sortBy].compareTo(b[sortBy]));

    return Scaffold(

      appBar: AppBar(

        title: Text(

          "${widget.friendName}'s Event List",

          style: const TextStyle(color: Colors.white),

        ),

        backgroundColor: const Color.fromARGB(255, 58, 2, 80),

        iconTheme: const IconThemeData(color: Colors.white),

        actions: [

          GestureDetector(

            onTap: () {

              Navigator.pushNamed(

                  context, '/profile');

            },

            child: const Padding(

              padding: EdgeInsets.only(right: 16.0),

              child: CircleAvatar(

                backgroundImage: AssetImage('assets/images/profile\_icon.png'),

                radius: 18.0,

              ),

            ),

          ),

        ],

      ),

      body: Container(

        decoration: const BoxDecoration(

          gradient: LinearGradient(

            colors: [

              Color.fromARGB(255, 58, 2, 80),

              Color.fromARGB(255, 219, 144, 5),

            ],

            begin: Alignment.topLeft,

            end: Alignment.bottomRight,

          ),

        ),

        padding: const EdgeInsets.all(16.0),

        child: Column(

          children: [

            DropdownButton<String>(

              value: sortBy,

              dropdownColor: const Color.fromARGB(255, 58, 2, 80),

              onChanged: (value) {

                setState(() {

                  sortBy = value!;

                });

              },

              items: const [

                DropdownMenuItem(

                  value: 'name',

                  child: Text(

                    'Sort by Name',

                    style: TextStyle(color: Colors.white),

                  ),

                ),

                DropdownMenuItem(

                  value: 'category',

                  child: Text(

                    'Sort by Category',

                    style: TextStyle(color: Colors.white),

                  ),

                ),

                DropdownMenuItem(

                  value: 'status',

                  child: Text(

                    'Sort by Status',

                    style: TextStyle(color: Colors.white),

                  ),

                ),

              ],

            ),

            const SizedBox(height: 16.0),

            Expanded(

              child: ListView.builder(

                itemCount: events.length,

                itemBuilder: (context, index) {

                  final event = events[index];

                  String formattedDate =

                      DateFormat('MMMM dd, yyyy').format(event['date']);

                  return Card(

                    color: Colors.grey[200],

                    shape: RoundedRectangleBorder(

                      borderRadius: BorderRadius.circular(15.0),

                    ),

                    child: ListTile(

                      title: Text(event['name']),

                      subtitle: Text(

                        '${event['category']} • ${event['status']} • $formattedDate',

                      ),

                      trailing: Row(

                        mainAxisSize: MainAxisSize.min,

                        children: [

                          IconButton(

                            icon: const Icon(Icons.edit, color: Colors.amber),

                            onPressed: () => editEvent(index),

                          ),

                          IconButton(

                            icon: const Icon(Icons.delete, color: Colors.red),

                            onPressed: () => deleteEvent(index),

                          ),

                        ],

                      ),

                    ),

                  );

                },

              ),

            ),

          ],

        ),

      ),

      floatingActionButton: FloatingActionButton.extended(

        backgroundColor: Colors.amber,

        foregroundColor: Colors.black,

        icon: const Icon(Icons.add),

        label: const Text('Add New Event'),

        onPressed: addEvent,

      ),

    );

  }

}

## 8-Gift\_list.dart:

import 'package:flutter/material.dart';

class GiftListPage extends StatefulWidget {

  final String? eventName;

  const GiftListPage({super.key, this.eventName});

  @override

  State<GiftListPage> createState() => \_GiftListPageState();

}

class \_GiftListPageState extends State<GiftListPage> {

  List<Map<String, dynamic>> gifts = [

    {

      'name': 'Smartphone',

      'category': 'Electronics',

      'status': 'Available',

      'price': 999.99,

      'pledged': false,

    },

    {

      'name': 'Book',

      'category': 'Books',

      'status': 'Pledged',

      'price': 19.99,

      'pledged': true,

    },

    {

      'name': 'Watch',

      'category': 'Accessories',

      'status': 'Available',

      'price': 149.99,

      'pledged': false,

    },

  ];

  String sortBy = 'name';

  void addGift() {

  }

  void editGift(int index) {

  }

  void deleteGift(int index) {

    setState(() {

      gifts.removeAt(index);

    });

  }

  @override

  Widget build(BuildContext context) {

    gifts.sort((a, b) => a[sortBy].compareTo(b[sortBy]));

    return Scaffold(

      appBar: AppBar(

        title: Text(

          "${widget.eventName}'s Gift List",

          style: const TextStyle(color: Colors.white),

        ),

        backgroundColor: const Color.fromARGB(255, 58, 2, 80),

        iconTheme: const IconThemeData(color: Colors.white),

      ),

      body: Container(

        decoration: const BoxDecoration(

          gradient: LinearGradient(

            colors: [

              Color.fromARGB(255, 58, 2, 80),

              Color.fromARGB(255, 219, 144, 5),

            ],

            begin: Alignment.topLeft,

            end: Alignment.bottomRight,

          ),

        ),

        padding: const EdgeInsets.all(16.0),

        child: Column(

          children: [

            DropdownButton<String>(

              value: sortBy,

              dropdownColor: const Color.fromARGB(255, 58, 2, 80),

              onChanged: (value) {

                setState(() {

                  sortBy = value!;

                });

              },

              items: const [

                DropdownMenuItem(

                  value: 'name',

                  child: Text(

                    'Sort by Name',

                    style: TextStyle(color: Colors.white),

                  ),

                ),

                DropdownMenuItem(

                  value: 'category',

                  child: Text(

                    'Sort by Category',

                    style: TextStyle(color: Colors.white),

                  ),

                ),

                DropdownMenuItem(

                  value: 'status',

                  child: Text(

                    'Sort by Status',

                    style: TextStyle(color: Colors.white),

                  ),

                ),

              ],

            ),

            const SizedBox(height: 16.0),

            Expanded(

              child: ListView.builder(

                itemCount: gifts.length,

                itemBuilder: (context, index) {

                  final gift = gifts[index];

                  return Card(

                    color: gift['pledged'] ? Colors.red[100] : Colors.grey[200],

                    shape: RoundedRectangleBorder(

                      borderRadius: BorderRadius.circular(15.0),

                    ),

                    child: ListTile(

                      title: Text(gift['name']),

                      subtitle: Text(

                        '${gift['category']} • ${gift['status']} • \$${gift['price'].toStringAsFixed(2)}',

                      ),

                      trailing: Row(

                        mainAxisSize: MainAxisSize.min,

                        children: [

                          IconButton(

                            icon: const Icon(Icons.edit, color: Colors.amber),

                            onPressed: gift['pledged'] ? null : () => editGift(index),

                          ),

                          IconButton(

                            icon: const Icon(Icons.delete, color: Colors.red),

                            onPressed: gift['pledged'] ? null : () => deleteGift(index),

                          ),

                        ],

                      ),

                    ),

                  );

                },

              ),

            ),

          ],

        ),

      ),

      floatingActionButton: FloatingActionButton.extended(

        backgroundColor: Colors.amber,

        foregroundColor: Colors.black,

        icon: const Icon(Icons.add),

        label: const Text('Add New Gift'),

        onPressed: addGift,

      ),

    );

  }

}