MOHD USMAAN MOGAL
36
D15A

EXP 2

Aim: To design Flutter UI by including common widgets.

Theory:

Image:

The Image widget supports various image formats such as JPEG, PNG, GIF, and more.
It can load images from the network using NetworkImage, from local assets using AssetImage, or from the file system using FileImage.
You can apply transformations, like scaling and cropping, to the displayed image.
Text:

The Text widget is used to display text and can be styled using the style property.
It supports rich formatting, such as different fonts, sizes, colors, and text alignments.
Text can be styled using the TextStyle class.

Icon:

The Icon widget displays a graphical icon. Flutter includes a set of built-in icons (from the Material Icons and Cupertino Icons libraries), and you can also use custom icons.
Icons can be styled using the color, size, and semanticLabel properties.

FlatButton:

The FlatButton widget is a simple button without any elevation.
It responds to touches by filling with color and can trigger actions when pressed.

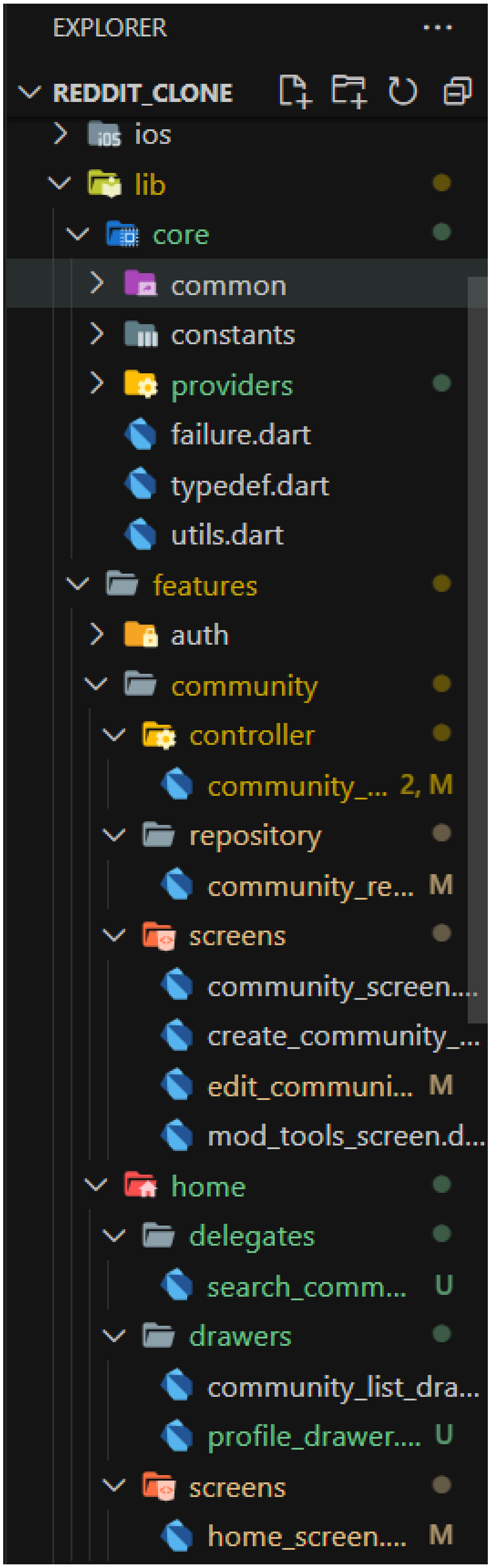
Container:

The Container widget is a box model that can contain other widgets.
It can be used for layout purposes, such as setting width, height, padding, margin, and decoration.
The decoration property allows you to apply background color, borders, and more.

Row and Column:

Row and Column widgets are used for arranging child widgets horizontally and vertically, respectively.
They automatically adjust the size of their children based on the available space.

File structure:

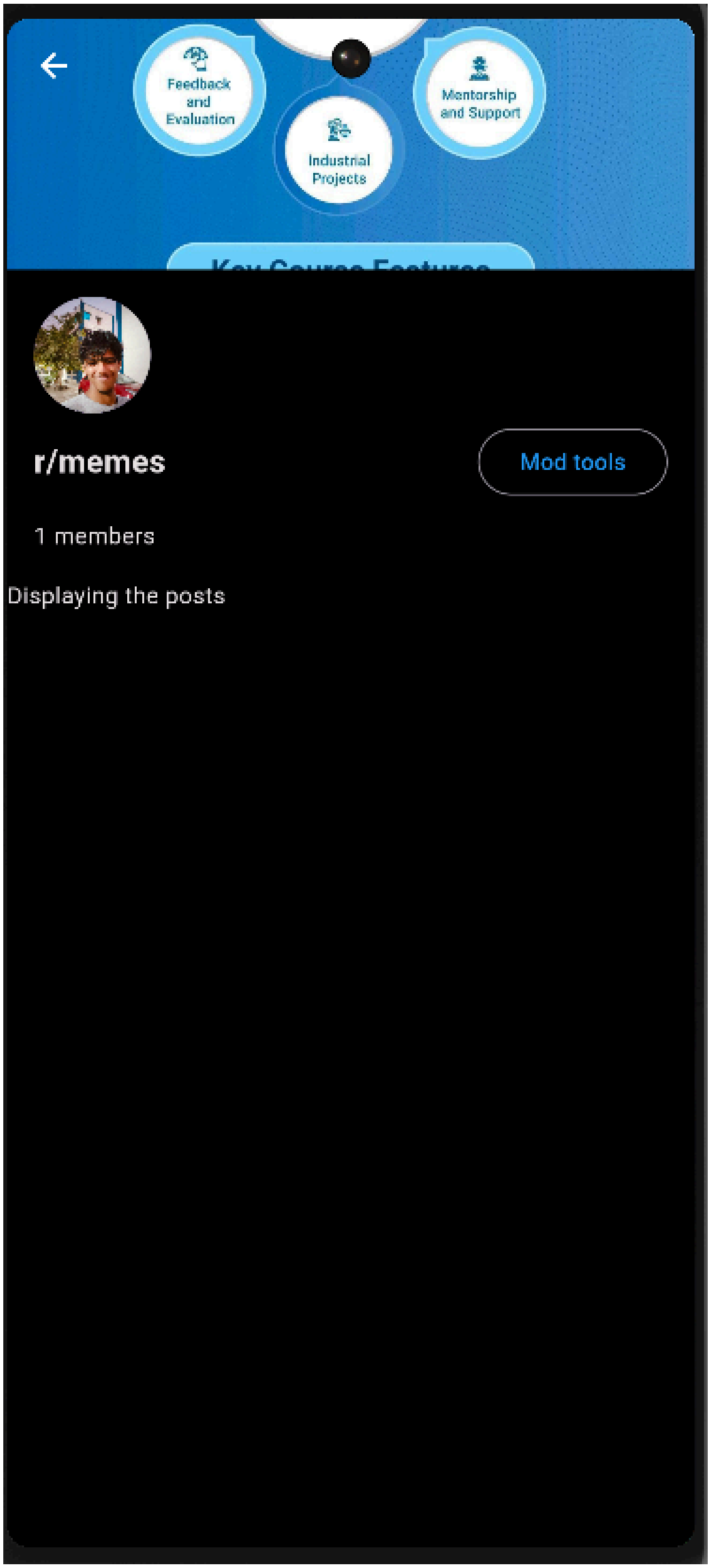


```
EXPLORER                                   ···
∨ REDDIT_CLONE
  > ios
  ∨ lib                                    ●
    ∨ core                                 ●
      > common
      > constants
      > providers                          ●
        failure.dart
        typedef.dart
        utils.dart
    ∨ features                             ●
      > auth
      ∨ community                          ●
        ∨ controller                       ●
            community_...              2, M
        ∨ repository                       ●
            community_re...               M
        ∨ screens                          ●
            community_screen....
            create_community_...
            edit_communi...               M
            mod_tools_screen.d...
      ∨ home                               ●
        ∨ delegates                        ●
            search_comm...                U
        ∨ drawers                          ●
            community_list_dra...
            profile_drawer....            U
        ∨ screens                          ●
            home_screen....               M
```

Code:

```
// ignore_for_file: public_member_api_docs, sort_constructors_first
import 'package:flutter/material.dart';
import 'package:flutter_riverpod/flutter_riverpod.dart';
import 'package:reddit_clone/core/common/error_text.dart';
import 'package:reddit_clone/core/common/loader.dart';
import 'package:reddit_clone/features/auth/controller/auth_controller.dart';
import 'package:reddit_clone/features/community/controller/community_controller.dart';
import 'package:routemaster/routemaster.dart';

class CommunityScreen extends ConsumerWidget {
  final String name;

  const CommunityScreen({super.key,
    required this.name,
  });

  void navigateToModTools(BuildContext context) {
    Routemaster.of(context).push('/mod-tools/$name');
  }

  @override
  Widget build(BuildContext context, WidgetRef ref) {
    final user = ref.watch(userProvider)!;
    return Scaffold(
      body: ref.watch(getCommunityNameProvider(name)).when(
          data: (community) => NestedScrollView(
              headerSliverBuilder: (context, innerBoxIsScrolled) {
                return [
                  SliverAppBar(
                    expandedHeight: 150,
                    floating: true,
                    snap: true,
                    flexibleSpace: Stack(
                      children: [
                        Positioned.fill(
                          child: Image.network(
                            community.banner,
                            fit: BoxFit.cover,
                          ),
                        ),
                      ],
                    ),
),
SliverPadding(
  padding: const EdgeInsets.all(16),
  sliver: SliverList(
      delegate: SliverChildListDelegate([
    Align(
      alignment: Alignment.topLeft,
      child: CircleAvatar(
        backgroundImage: NetworkImage(community.avatar),
        radius: 35,
      ),
    ),
    const SizedBox(
      height: 5,
    ),
    Row(
      mainAxisAlignment: MainAxisAlignment.spaceBetween,
      children: [
        Text(
          'r/${community.name}',
          style: const TextStyle(
              fontSize: 19, fontWeight: FontWeight.bold),
        ),
        community.mods.contains(user.uid)
            ? OutlinedButton(
                onPressed: () {
                  navigateToModTools(context);
                },
                style: ElevatedButton.styleFrom(
                    shape: RoundedRectangleBorder(
                      borderRadius: BorderRadius.circular(20),
                    ),
                    padding: const EdgeInsets.symmetric(
                        horizontal: 25),
                    shadowColor: Colors.blue),
                child: const Text(
                  'Mod tools',
                  style: TextStyle(color: Colors.blue),
                ))
            : OutlinedButton(
                onPressed: () {},
                style: ElevatedButton.styleFrom(
                    shape: RoundedRectangleBorder(
                      borderRadius: BorderRadius.circular(20),
                                      ),
                                      padding: const EdgeInsets.symmetric(
                                          horizontal: 25),
                                      shadowColor: Colors.blue),
                                  child: Text(
                                    community.members.contains(user.uid)
                                        ? 'Joined'
                                        : 'Join',
                                    style: const TextStyle(color: Colors.blue),
                                  ))
                        ],
                      ),
                      Padding(
                        padding: const EdgeInsets.only(top: 10),
                        child: Text('${community.members.length} members'),
                      )
                    ])),
                  )
                ];
              },
              body: const Text('Displaying the posts')),
          error: (error, stackTrace) => ErrorText(error: error.toString()),
          loading: () => const Loader()),
    );
  }
}
```

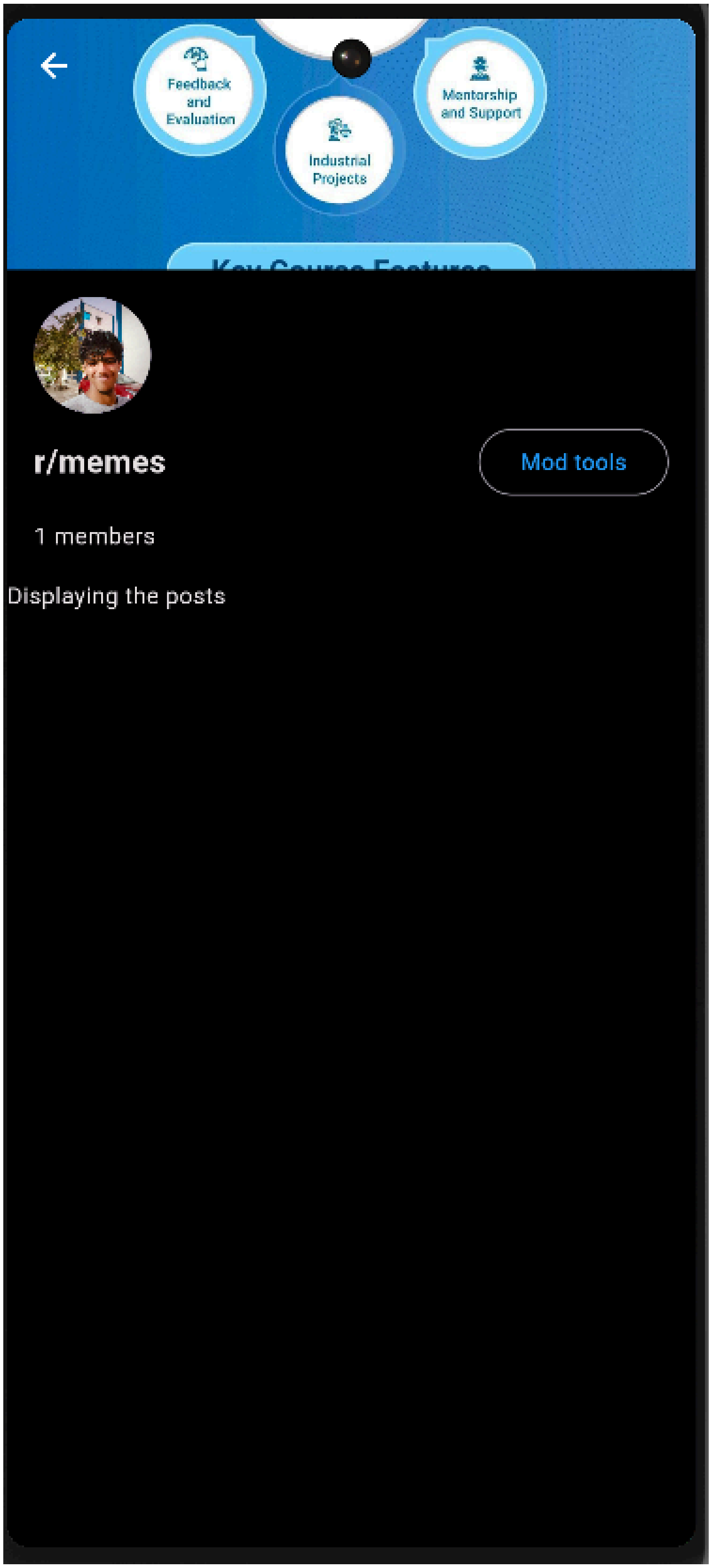
Output:



r/memes

1 members

Displaying the posts



r/memes

1 members

Displaying the posts