MAD and PWA Lab Mini Project

Name : Shubham Nakashe **Class:D15A** **Roll no: 38**

Aim: To include icons, images, fonts in Flutter app

Theory:

- Fonts:
  You can use custom fonts in your Flutter app by adding the font files (e.g., `.ttf` or `.otf`) to your project and then specifying them in your `pubspec.yaml` file. Once added, you can use these fonts in your app's text widgets by specifying the font family.

- Images:
  Flutter supports various image formats (e.g., PNG, JPEG, GIF) and provides widgets like `Image` and `AssetImage` to display images in your app. You can load images from assets or network URLs.

- Videos:
  Flutter provides the `video_player` package for playing videos in your app. You can use this package to display and control video playback, including features like play, pause, seek, and fullscreen.

- Icons:
  Flutter includes a set of built-in icons from popular icon packs like Material Icons and Cupertino Icons. You can use these icons in your app using the `Icon` widget. Additionally, you can use custom icons by importing and referencing them in your app.

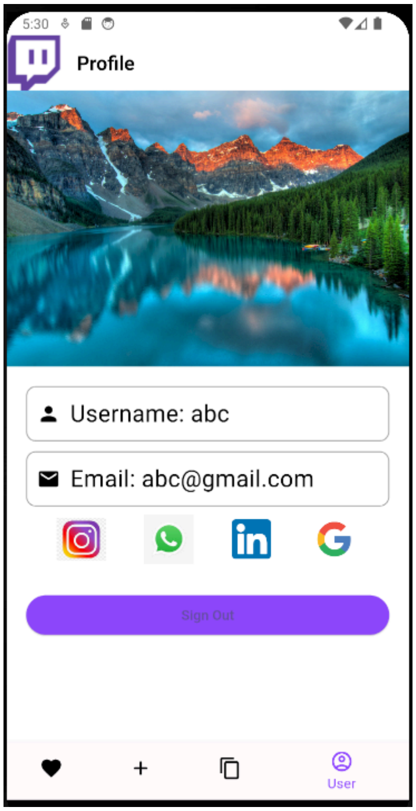
PUBSPEC.yaml

```
assets:
  - assets/twitch-icon.png
  - assets/nature.jpg
  - assets/whatsapp_icon.jpg
  - assets/linkedin_icon.png
  - assets/nature2.jpg
  - assets/google_icon.png
  - assets/instagram_icon.jpg
```

CODE:

```
import 'package:flutter/material.dart';
import 'package:twitch_clone/models/channel_model.dart';
import 'package:twitch_clone/models/video_model.dart';
import 'package:twitch_clone/screens/video_screen.dart';
import 'package:twitch_clone/services/api_service.dart';

class browse extends StatefulWidget {
  @override
  _browseState createState() => _browseState();
}

class _browseState extends State<browse> {
  List<String> channelIds = [
    'UCeVMnSShP_Iviwkknt83cww',
    'UC1XBh-m27kkgwLAwu_SRJBg',
    'UCq-Fj5jknLsUf-MWSy4_brA',
    'UCBwmMxybNva6P_5VmxjzwqA',
    'UCsooa4yRKGN_zEE8iknghZA',
    'UCOQNJjhXwvAScuELTT_i7cQ',
    'UCBqFKDipsnzvJdt6UT0lMIg',
    'UCPXGFu34px86DdXwocV-bYA',
    'UCX8pnu3DYUnx8qy8V_c6oHg',
    'UClfos9f7uDdoun8ZyE9jYFg',
    'UC7eHZXheF8nVOfwB2PEslMw'

    // Add more channel IDs as needed
  ];
  List<Channel> channels = [];
  bool _isLoading = false;

  @override
  void initState() {
    super.initState();
    _initChannels();
  }

  _initChannels() async {
    List<Channel> loadedChannels = [];
    for (String channelId in channelIds) {
      Channel channel =
          await APIService.instance.fetchChannel(channelId: channelId);
      loadedChannels.add(channel);
    }
    setState(() {
      channels = loadedChannels;
    });
  }

  Widget _buildProfileInfo(Channel channel) {
    return Container(
      margin: EdgeInsets.all(20.0),
      padding: EdgeInsets.all(20.0),
      height: 100.0,
      decoration: BoxDecoration(
        color: Colors.white,
        boxShadow: [
          BoxShadow(
            color: Colors.black12,
            offset: Offset(0, 1),
            blurRadius: 6.0,
          ),
        ],
      ),
      child: Row(
        children: <Widget>[
          CircleAvatar(
            backgroundColor: Colors.white,
            radius: 35.0,
            backgroundImage: NetworkImage(channel.profilePictureUrl),
          ),
          SizedBox(width: 12.0),
          Expanded(
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              crossAxisAlignment: CrossAxisAlignment.start,
              children: <Widget>[
                Text(
                  channel.title,
                  style: TextStyle(
                    color: Colors.black,
                    fontSize: 20.0,
                    fontWeight: FontWeight.w600,
                  ),
                  overflow: TextOverflow.ellipsis,
                ),
                Text(
                  '${channel.subscriberCount} subscribers',
                  style: TextStyle(
                    color: Colors.grey[600],
                    fontSize: 16.0,
                    fontWeight: FontWeight.w600,
                  ),
                  overflow: TextOverflow.ellipsis,
                ),
              ],
            ),
          )
        ],
      ),
    );
  }

  _buildVideo(Video video) {
    return GestureDetector(
      onTap: () => Navigator.push(
        context,
        MaterialPageRoute(
          builder: (_) => VideoScreen(id: video.id),
        ),
      ),
      child: Container(
        margin: EdgeInsets.symmetric(horizontal: 20.0, vertical: 5.0),
        padding: EdgeInsets.all(10.0),
        height: 140.0,
        decoration: BoxDecoration(
          color: Colors.white,
          boxShadow: [
            BoxShadow(
              color: Colors.black12,
              offset: Offset(0, 1),
              blurRadius: 6.0,
            ),
          ],
        ),
        child: Row(
          children: <Widget>[
            Image(
              width: 150.0,
              image: NetworkImage(video.thumbnailUrl),
            ),
            SizedBox(width: 10.0),
            Expanded(
              child: Text(
                video.title,
                style: TextStyle(
                  color: Colors.black,
                  fontSize: 18.0,
                ),
              ),
            ),
          ],
        ),
      ),
    );
  }

  _loadMoreVideos(Channel channel) async {
    _isLoading = true;
    List<Video> moreVideos = await APIService.instance
        .fetchVideosFromPlaylist(playlistId: channel.uploadPlaylistId);
    List<Video> allVideos = channel.videos..addAll(moreVideos);
    setState(() {
      channel.videos = allVideos;
    });
    _isLoading = false;
  }

  @override
 Widget build(BuildContext context) {
  // Flatten the list of videos from all channels
  List<Video> allVideos = [];
  for (Channel channel in channels) {
    allVideos.addAll(channel.videos);
  }

  // Shuffle all the videos
  allVideos.shuffle();

  return Scaffold(
    appBar: AppBar(
      title: Text('Browse Channels'),
    ),
    body: ListView.builder(
      itemCount: allVideos.length,
      itemBuilder: (BuildContext context, int index) {
        Video video = allVideos[index];
        return _buildVideo(video);
      },
    ),
  );
}
  Widget _buildChannelWidget(Channel channel) {
  return NotificationListener<ScrollNotification>(
    onNotification: (ScrollNotification scrollDetails) {
      if (!_isLoading &&
          channel.videos.length != int.parse(channel.videoCount) &&
     scrollDetails.metrics.pixels ==
       scrollDetails.metrics.maxScrollExtent) {
    _loadMoreVideos(channel);
   }
   return false;
  },
  child: ListView.builder(
   shrinkWrap: true,
   physics: NeverScrollableScrollPhysics(),
   itemCount: 1 + channel.videos.length,
   itemBuilder: (BuildContext context, int index) {
    if (index == 0) {
     return Container(); // You can add additional information for the channel here
    }
    Video video = channel.videos[index - 1];
    return _buildVideo(video);
   },
  ),
 );
}
}
```

OUTPUT:

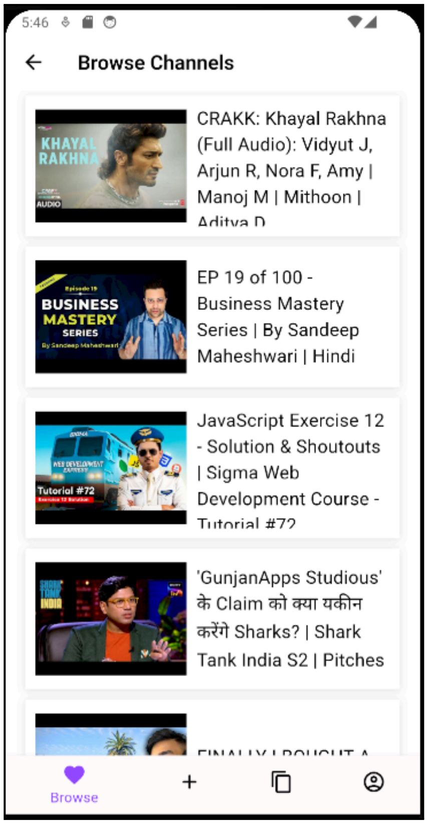


Profile

Username: abc

Email: abc@gmail.com

Images and icons (twitch ,whatsapp,instagram,google etc .)



Browse Channels

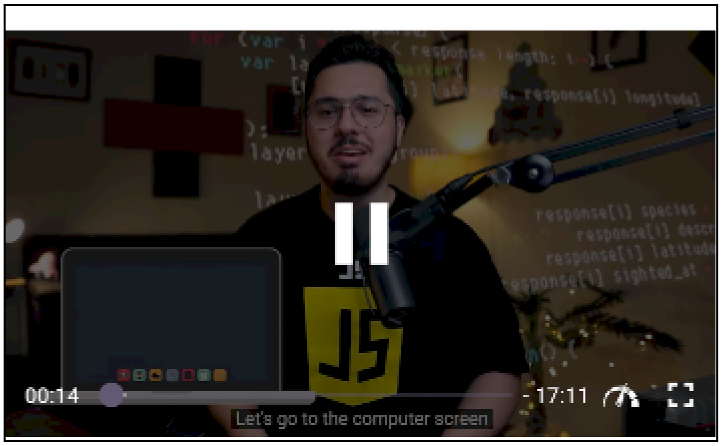
CRAKK: Khayal Rakhna (Full Audio): Vidyut J, Arjun R, Nora F, Amy | Manoj M | Mithoon | Aditya D

EP 19 of 100 - Business Mastery Series | By Sandeep Maheshwari | Hindi

JavaScript Exercise 12 - Solution & Shoutouts | Sigma Web Development Course - Tutorial #72

'GunjanApps Studious' के Claim को क्या यकीन करेंगे Sharks? | Shark Tank India S2 | Pitches

FINALLY I BOUGHT A



Videos