Capstone Project: Develop a YouTube Clone Using the MERN Stack (400 marks)

Objective: Create a full-stack YouTube clone where users can view and interact with videos. This project will help you understand how to build a real-world application using MongoDB, Express, React, and Node.js (MERN stack).

Requirements:

Front-End (React)

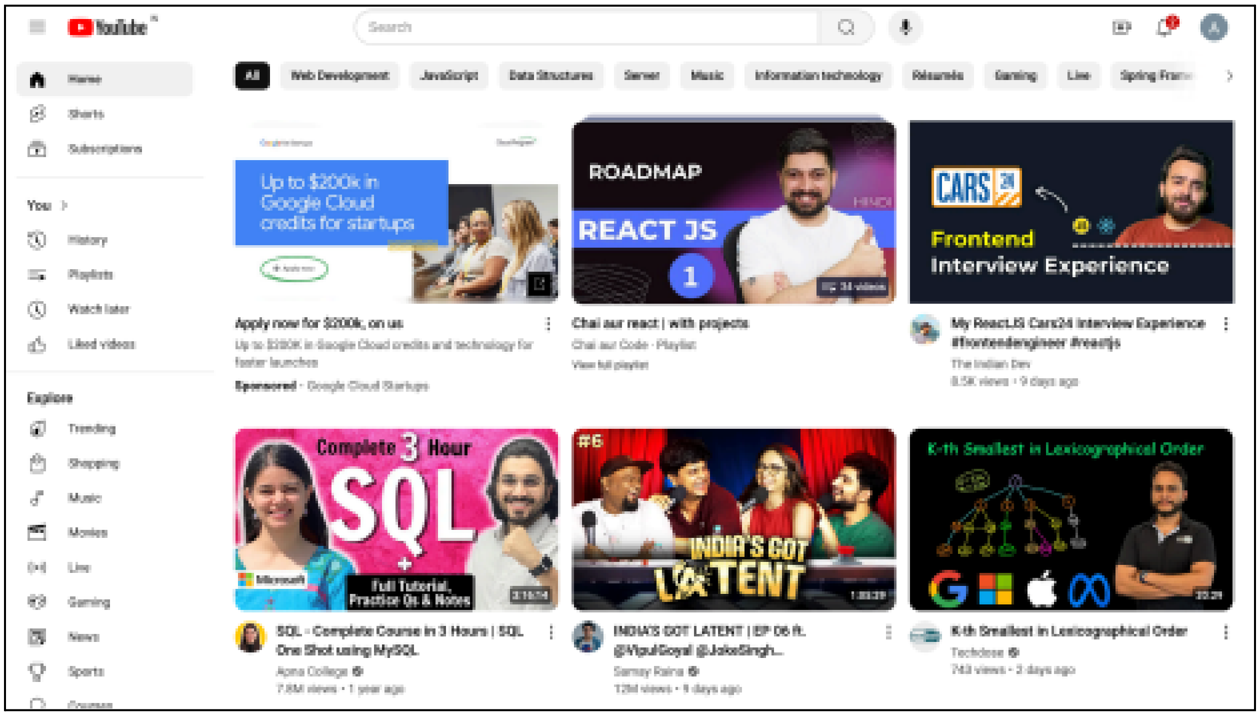
1. **Home Page:**
    - ○ Display a YouTube Header.
        - ○ Display a static sidebar that you can toggle from the top hamburger menu.
    - ○ Display filter buttons at the top.
    - ○ Display a grid of video thumbnails.
    - ○ Each video card should show:
        - ■ Title
        - ■ Thumbnail
        - ■ Channel Name
        - ■ Views

Sample data for videos:

Sample Data:

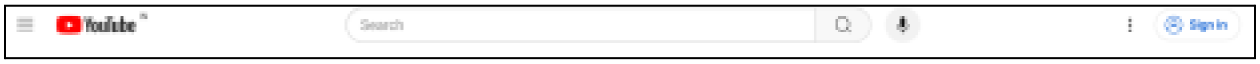
[{ "videoId": "video01", "title": "Learn React in 30 Minutes", "thumbnailUrl": "https://example.com/thumbnails/react30min.png", "description": "A quick tutorial to get started with React.", "channelId": "channel01", "uploader": "user01", "views": 15200, "likes": 1023, "dislikes": 45, "uploadDate": "2024-09-20", "comments": [{ "commentId": "comment01", "userId": "user02", "text": "Great video! Very helpful.", "timestamp": "2024-09-21T08:30:00Z" }] }]

Home Page will look like this:


YouTube
Search
Home
Shorts
Subscriptions
You
History
Playlists
Watch later
Liked videos
Explore
Trending
Shopping
Music
Movies
Live
Gaming
News
Sports
All
Web Development
JavaScript
Data Structures
Server
Music
Information technology
Gaming
Live
Up to $200k in Google Cloud credits for startups
Apply now for $200k, on us
Sponsored · Google Cloud Startups
ROADMAP
REACT JS
Chai aur react | with projects
Chai aur Code · Playlist
View full playlist
CARS 24
Frontend Interview Experience
My ReactJS Cars24 Interview Experience #frontendengineer #reactjs
The Indian Dev
Complete 3 Hour SQL
Full Tutorial, Practice Qs & Notes
SQL - Complete Course in 3 Hours | SQL One Shot using MySQL
Apna College
INDIA'S GOT LATENT
INDIA'S GOT LATENT | EP 06 ft. @VipulGoyal @JaloSingh...
Samay Raina
K-th Smallest in Lexicographical Order
Techdose

2. **User Authentication:**
    - Allow users to register and log in with:
        - Username
        - Email
        - Password
    - Use JWT for authentication.

Before signing in, the header should have a sign-in button



When the user clicks on sign in, it should take them to a new URL where a Google form to log in and register should be present.

The user should sign in, and after signing in, his/her name should be present at the top, and the home page will be displayed.

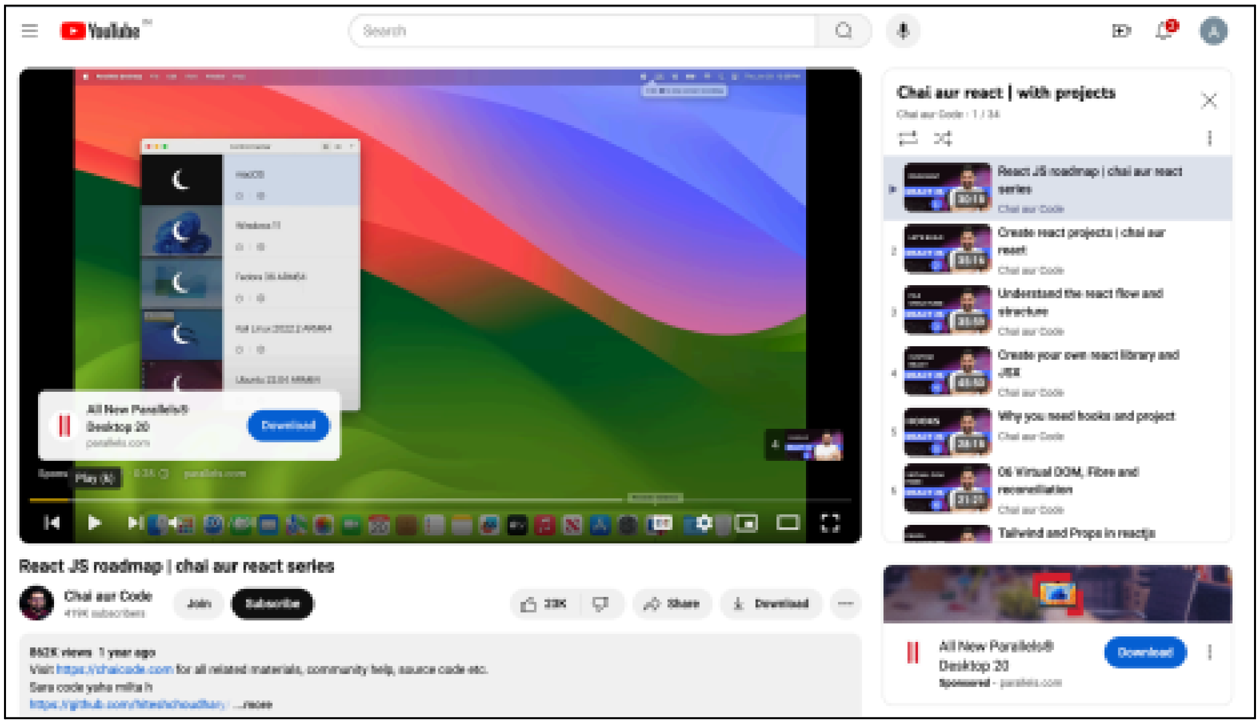
Sample data for users:

{ userId: "user01", username: "JohnDoe", email: "john@example.com", password: "hashedPassword123", avatar: "https://example.com/avatar/johndoe.png", channels: ["channel01"], }

3. **Search and Filter Functionality:**
   - Implement a search bar on the homepage.
     This search bar is present in the header. Filter videos based on title.
   - Filter buttons should be implemented, and filters should work based on category.

4. **Video Player Page:**
   - Display the selected video with:
     - Video player
     - Title and description
     - Channel name
     - Like and dislike buttons
     - Comment section (**Ignore** nested comments for now)

Users should be able to add, edit, and delete comments. When a new comment is added, it should be saved in the database along with that video.

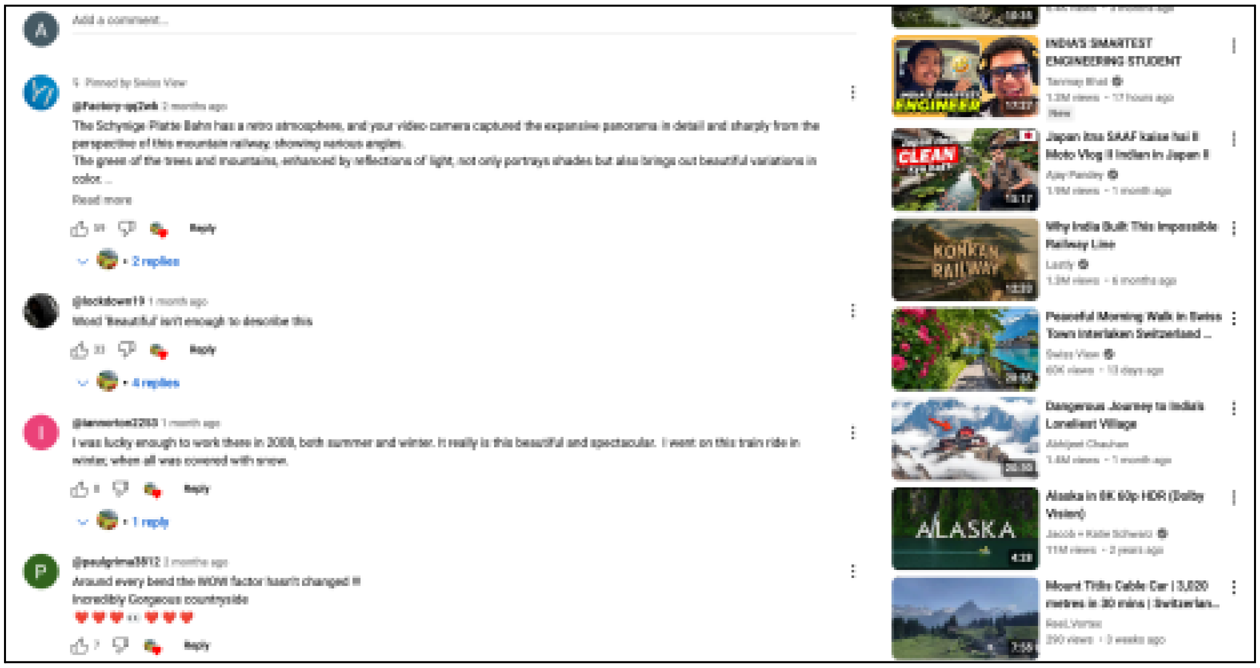


YouTube
Search
Chai aur react | with projects
Chai aur Code · 1 / 34
React JS roadmap | chai aur react series
Create react projects | chai aur react
Understand the react flow and structure
Create your own react library and JSX
Why you need hooks and project
08 Virtual DOM, Fibre and reconciliation
Tailwind and Props in reactjs
All New Parallels® Desktop 20
Download
Join
Subscribe
23K
Share
Download

React JS roadmap | chai aur react series

Chai aur Code
419K subscribers

862K views 1 year ago
Visit https://chaicode.com for all related materials, community help, source code etc.
Sara code yaha milta h
https://github.com/hiteshchoudhary ...more



INDIA'S SMARTEST ENGINEERING STUDENT
Japan itna SAAF kaise hai Il Moto Vlog Il Indian in Japan Il
Why India Built This Impossible Railway Line
KONKAN RAILWAY
Peaceful Morning Walk in Swiss Town Interlaken Switzerland ...
Dangerous Journey to India's Loneliest Village
Alaska in 8K 60p HDR (Dolby Vision)
ALASKA
Mount Titlis Cable Car | 3,020 metres in 30 mins | Switzerlan...

Add a comment...

Pinned by Swiss View

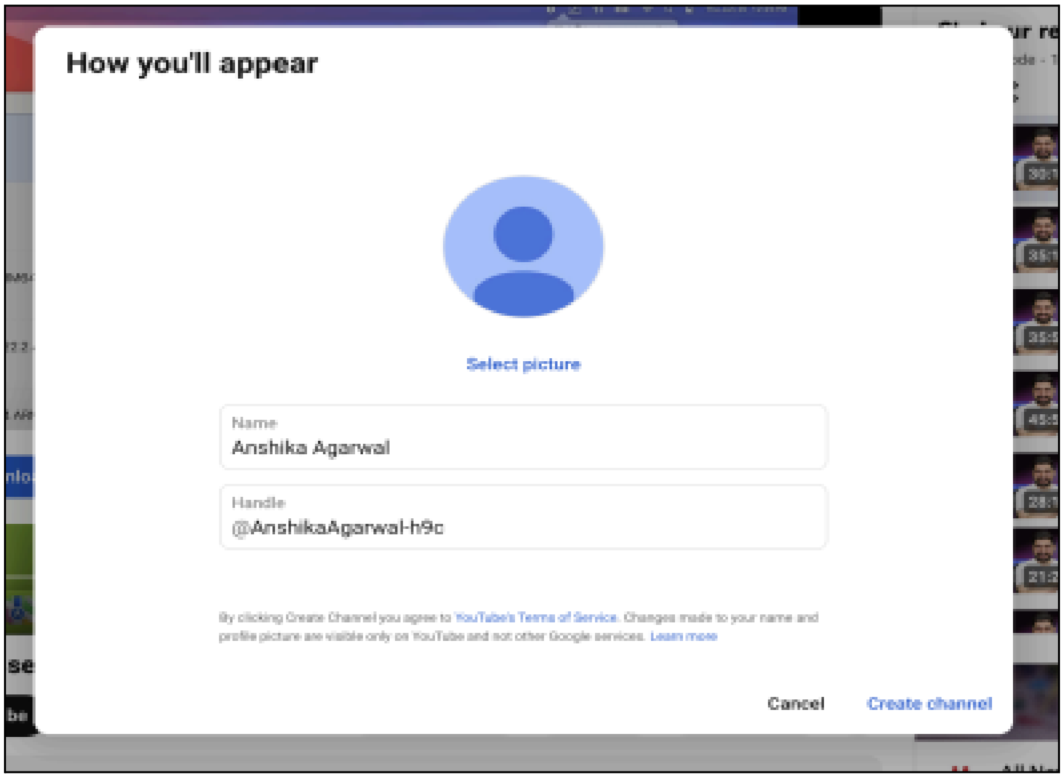
@Factory-sg2wk 2 months ago
The Schynige Platte Bahn has a retro atmosphere, and your video camera captured the expansive panorama in detail and sharply from the perspective of this mountain railway, showing various angles.
The green of the trees and mountains, enhanced by reflections of light, not only portrays shades but also brings out beautiful variations in color. ...
Read more
Reply
2 replies

@lockdown19 1 month ago
Word 'Beautiful' isn't enough to describe this
Reply
4 replies

@iannorton2253 1 month ago
I was lucky enough to work there in 2008, both summer and winter. It really is this beautiful and spectacular. I went on this train ride in winter, when all was covered with snow.
Reply
1 reply

@paulgrima3812 3 months ago
Around every bend the WOW factor hasn't changed !!
Incredibly Gorgeous countryside
❤️❤️❤️ ❤️❤️❤️
Reply

5. **Channel Page:**
   - ○ Option to create a channel only after the user is signed in.
   - ○ Display a list of videos which belong to that particular channel.
   - ○ Allow the user to edit or delete their videos.



How you'll appear

Select picture

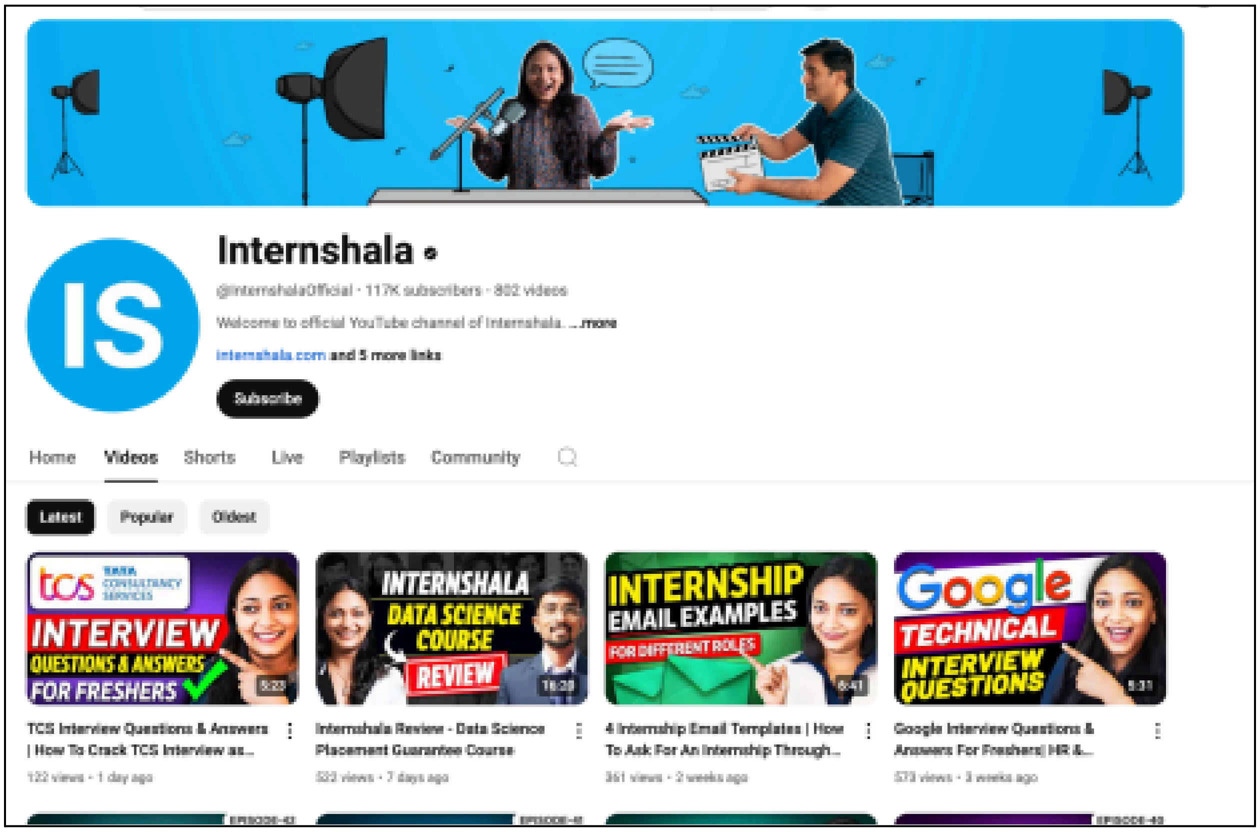
Name
Anshika Agarwal

Handle
@AnshikaAgarwal-h9c

By clicking Create Channel you agree to YouTube's Terms of Service. Changes made to your name and profile picture are visible only on YouTube and not other Google services. Learn more

Cancel   Create channel

Sample Channel Page


Internshala
@InternshalaOfficial · 117K subscribers · 802 videos
Welcome to official YouTube channel of Internshala. ...more
internshala.com and 5 more links
Subscribe
Home
Videos
Shorts
Live
Playlists
Community
Latest
Popular
Oldest
TCS Interview Questions & Answers | How To Crack TCS Interview as...
122 views · 1 day ago
Internshala Review - Data Science Placement Guarantee Course
523 views · 7 days ago
4 Internship Email Templates | How To Ask For An Internship Through...
361 views · 2 weeks ago
Google Interview Questions & Answers For Freshers| HR &...
573 views · 3 weeks ago

6.

6. **Responsive Design:**
    - ○ Ensure the application is fully responsive across devices.

Sample data for Channel:

{ channelId: "channel01", channelName: "Code with John", owner: "user01", description: "Coding tutorials and tech reviews by John Doe.", channelBanner: "https://example.com/banners/john_banner.png", subscribers: 5200, videos: ["video01", "video02"], }

Back-End (Node.js, Express)

1. **API Endpoints:**
    - ○ **User Authentication:**
        - ■ Sign up, login, and token-based authentication using JWT.
    - ○ **Channel Management:**
        - ■ API to create a new channel and fetch any information from that channel.
    - ○ **Video Management:**
        - ■ API to fetch, update, and delete videos.

   - **Comments:**
      - API to add and fetch comments.
2. **Database (MongoDB):**
   - Store users, videos, channels, and comments in MongoDB collections.
   - Store file metadata (e.g., video URL, thumbnail URL) in the database.

Technologies to Use:

- **Frontend:** React, React Router, Axios
- **Backend:** Node.js, Express.js, MongoDB
- **Authentication:** JWT (JSON Web Tokens)
- **Database:** MongoDB (MongoDB Atlas or local instance)
- **Version Control:** Git

Submission Requirements:

- Source code uploaded to a GitHub repository.
- A detailed README file explaining the project setup, features, and usage.
- A short video demo showcasing the application's features.

Rubrics for the YouTube Clone Assignment (Out of 400 Marks)

Category	Criteria	Marks
Front-End (React)		**170**
Home Page UI/UX	**Header, Sidebar, Video Grid, Filters**	**40**
User Authentication	**Registration, JWT-based login, Sign-in functionality**	**40**

Video Player Page	Video player, comments section, like/dislike buttons	50
Channel Page	Ability to create/edit/delete videos, channel page layout	40
Back-End (Node.js & Express)		120
API Design	Proper design of routes for authentication, video management, and comments	40
Data Handling (MongoDB)	Storing and fetching data correctly for users, videos, comments, and channels	40
JWT Integration	Secure JWT-based authentication and protected routes	40
Search & Filter Functionality		40

Search by Title	Working search functionality on the home page	20
Filter by Category	Proper filter Implementation based on video category	20
Responsiveness		30
Mobile/Tablet/Desktop Layout	Proper layout and functionality across different screen sizes	30
Code Quality & Documentation		40
Code Structure	Proper folder structure, clean code, adherence to coding best practices	20
Documentation	Clear comments, README file explaining project setup and usage	20

Rubric Breakdown:

1. Front-End (React) – 170 Marks:

- **Home Page UI/UX (40 Marks)**: Evaluate the completeness of the header, sidebar, video grid, filter buttons, and responsiveness on various devices.
- **User Authentication (40 Marks**): Ensure proper implementation of user registration, JWT-based login, and sign-in functionality.
- **Video Player Page (50 Marks)**: Check for the functioning video player, comment management, and interactive buttons (like/dislike).
- **Channel Page (40 Marks):** Evaluate the user's ability to create, edit, and delete their videos and the proper channel page layout.

2. Back-End (Node.js & Express) – 120 Marks:

- **API Design (40 Marks):** Assess the routing and functionality for users, videos, and comments, ensuring best practices are followed.
- **Data Handling (MongoDB) (40 Marks)**: Check how well the data is stored and retrieved, including video metadata, user info, comments, and channel details.
- **JWT Integration (40 Marks):** Ensure that JWT-based authentication is secure and all protected routes work correctly.

3. Search & Filter Functionality – 40 Marks:

- **Search by Title (20 Marks):** Verify the search bar functionality works correctly for searching videos by title.
- **Filter by Category (20 Marks):** Ensure category-based filters are functional and correctly implemented.

4. Responsiveness – 30 Marks:

- **Mobile/Tablet/Desktop Layout (30 Marks):** Test how well the app adapts across different screen sizes, ensuring all features are usable and the layout remains consistent.

5. Code Quality & Documentation – 40 Marks:

- **Code Structure (20 Marks):** Check for clean, well-organized code with proper folder structure, adhering to best practices.
- **Documentation (20 Marks):** Ensure the project is well-documented with clear comments and a README explaining the setup, usage, and purpose of the Project.

Points To Be Remembered

1. **In Sample Data for video:**
   You can add a videoUrl key to store the video URL in string format.
   {“videoUrl”: “https://example_video.com”}

2. **User Authentication:**
   - ❖ Allow users to register and log in with:
     ■ Username ■ Email ■ Password
   - ❖ Use JWT for authentication
     Apply proper validation (username, email, password) to all input fields and display relevant error messages on the UI. After successful registration, the user should be automatically redirected to the login page.

3. **Search and Filter Functionality:**
   Filter buttons should be implemented, and filters should work based on category. And at least 6 filter buttons should be added. Uploaded videos should be dynamically added to the homepage and displayed when clicking on the relevant filter button.

4. **Video Player Page:** Implement complete functionality for the Like and Dislike buttons.

Read Carefully Before Submission

- Make sure to use ES Modules (import/export) instead of CommonJS (require/module.exports).
- Avoid using Create React App (CRA) — it's deprecated. Instead, use tools like Vite.
- *Do not send node modules.*
- Both backend and Frontend are mandatory for this project.
- Your GitHub repository should reflect proper commit history (at least 30 commits).
- Individual commits for the Frontend and Backend sections.
- If MongoDB Compass is used locally, seed your database or provide export files so the evaluators can test features.

- Ensure that user registration/login works — authentication is foundational to most features. If Authentication will not work properly, then the marks will deduct on those functionalities on which it is relying on.
- A working channel page is required to manage videos. You need to implement Create, Read, Update, and Delete (CRUD) operations for the videos from the channel page.
- A fully functional video player page is required, allowing users to watch videos and post comments on them. You need to implement full Create, Read, Update, and Delete (CRUD) operations for comments directly from the video player page.
- The project should reflect originality. Copied code (especially without understanding) may result in significant mark deduction.
- Since this is a YouTube clone project, it's important to replicate the core features and overall user experience effectively. Minimal styling and missing functionalities may contribute to mark deductions.