**GitHub Link: -** [**https://github.com/uday-rm20/You-tube-Clone**](https://github.com/uday-rm20/You-tube-Clone)

**Routes and HTTP methods to be used for Client/Server Side Operations:**

* To Sign up a user –

POST - <http://localhost:5100/auth/signUp>

* To Login a user to get JWT token-

POST - <http://localhost:5100/auth/login>

* To logout a user-

POST - <http://localhost:5100/auth/logout>

* To add/post a video -

POST - <http://localhost:5100/api/video>

* To get all videos in the database-

GET - <http://localhost:5100/api/allvideo>

* To get a video by its Id –

GET -<http://localhost:5100/api/getVideoById/:id>

* To go the channel of a particular user –

GET - <http://localhost:5100/api/:userId/channel>

* To post/add a comment to a video–

POST - <http://localhost:5100/commentApi/comment>

* To get a comment by videoId –

GET - <http://localhost:5100/commentApi/comment/:videoId>

* To Delete a Comment-

DELETE - <http://localhost:5100/commentApi/comment/:id>

* To edit/Update a comment –

PUT - <http://localhost:5100/commentApi/comment/:id>

Note: id is the unique object ID given by Mongo DB.

**Setup:**

* Build a Nodejs project for back-end
* Build a React project for front-end using vite
* Install the required packages –
* npm install to install node modules
* Express
* Mongoose
* React
* React-router-dom
* Jsonwebtoken
* Bcryptjs
* Cors
* Axios
* react-toastify
* Start the Front-end in an integrated terminal using command –

npm run dev

* Start the Back end in another integrated terminal using command –

npm start

{because back end is using nodemon package for active listening to changes, note: include "start": "nodemon server.js" in package.json to use nodemon}

* If not using nodemon start back end by running command –

node main\_server\_file.js   
{In my case its server.js i.e.=> node server.js}

**Features and Usage:**

* Homepage contains videos in the database in a grid format.
* Functional Search bar used to filter out videos using Titles.
* Signup feature – users can register in the app by providing the necessary details.
* Login feature – after registering user can login by using there credentials to use application and avail some user authorized features.
* Logout feature – user can logout of their account by clicking on logout.
* A dedicated video player page when clicking on a certain video it opens to a new page showcasing comment section and playing the selected video.
* Add video feature – an logged in user can upload a video and view their videos in their dedicated channel page.
* Edit and delete options for comment section – user can edit or delete their comments by choosing relevant options.

For Sample user experience:

* Use the below credentials in the log In form -
* Username – Bharat
* Password – 1903
* Results of Application:





