

# Youtube - Clone

Steps --

```
Npx create-react-app youtube-clone
```

```
npm install -D tailwindcss
```

```
npx tailwindcss init
```

Configure path – config tailwind

---

```
/** @type {import('tailwindcss').Config} */
module.exports = {
  content: ["./src/**/*.{html,js}"],
  theme: {
    extend: {},
  },
  plugins: [],
}
```

---

Index.file -

```
@tailwind base;
@tailwind components;
@tailwind utilities;
```

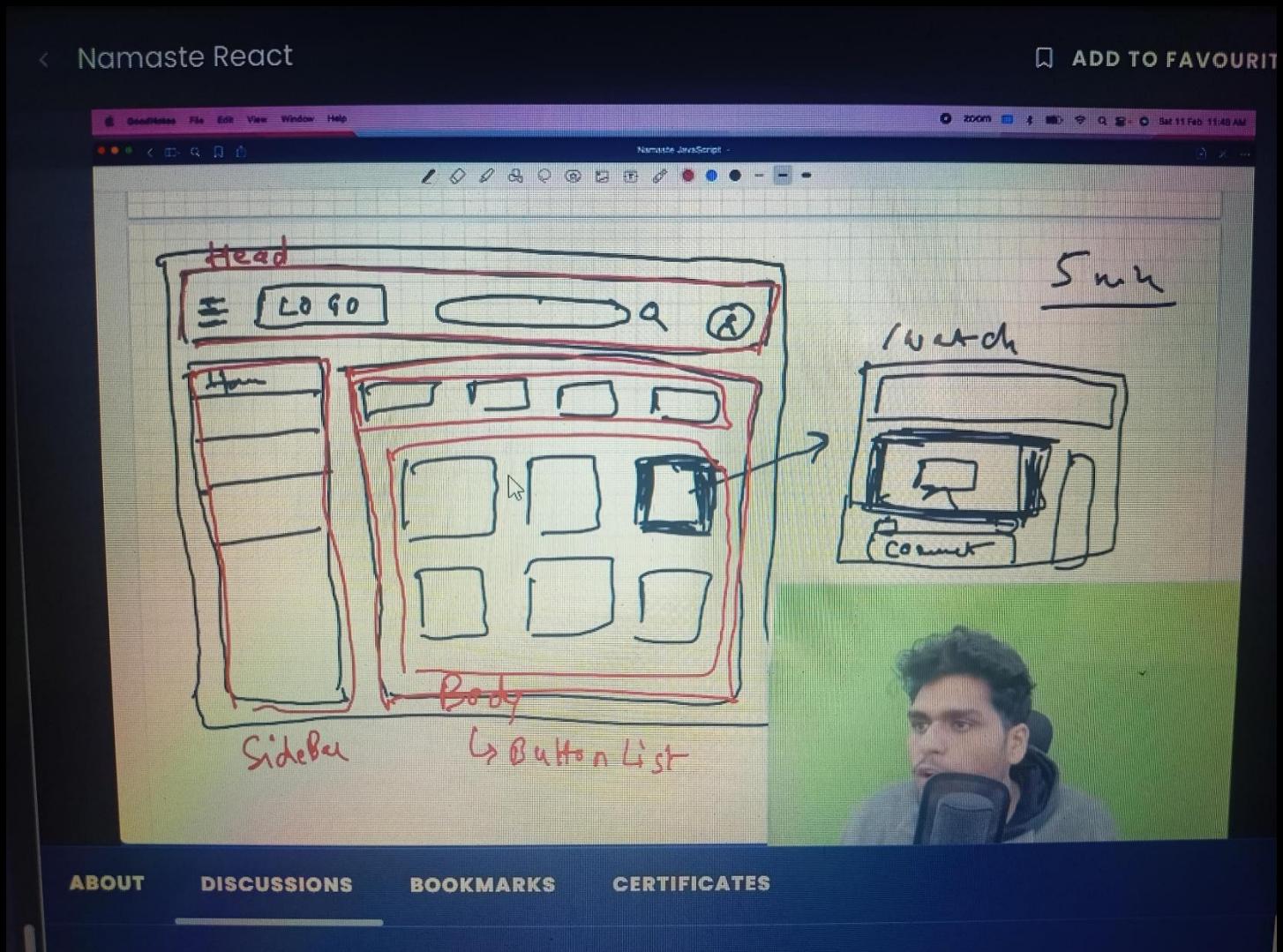
<React.StrictMode> is a development mode feature in React that helps with identifying potential problems in the application. It activates additional checks and warnings for its descendants. It is not a component that renders anything, but rather a tool that highlights potential issues in the code. It is recommended to use it during development and testing, but not in production.

Machine coding interview – requirements clarification .

We have to ask interviewer about the features we need to build.

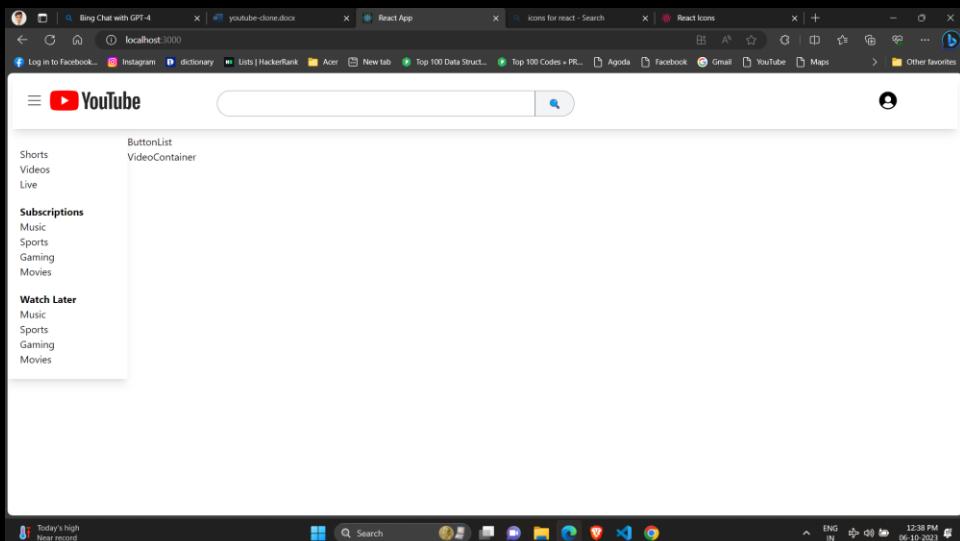
Ask about tech stack – what are techs we will use – like tailwind or css. Redux or use State?

Next 5 mins in planning things – low level design . Discuss your approach.



After this you have to say which cmd u would use , I use npx create-react-app.

reportWebVitals is a function that is used to report performance metrics of a web page to a third-party service or to the developer's own analytics system.



Now I want to write a functionality where if I click on menu bar button the side bar should toggle.

How to build this??

Using use state??

And example -

```
import React, { useState } from 'react';
const App = () => {
  const [sidebarVisible, setSidebarVisible] = useState(false);
  const toggleSidebar = () => {
    setSidebarVisible(!sidebarVisible);
  };
  return (
    <div className="flex">
      <div className={`sidebar ${sidebarVisible ? 'visible' : 'hidden'}`}>
        {/* Sidebar content */}
      </div>
      <div className="content">
        <button onClick={toggleSidebar}>Toggle Sidebar</button>
        {/* Main content */}
      </div>
    </div>
  );
};
export default App;
```

But we wont use this because we need to think big and we r building utube with many functionalities.

So we can build a it using redux store.

Npm Install @reduxjs/toolkit and npm i react-redux

Using Redux :

First thing I have to do is --- > create redux Store – it's a central repository system where can store the data . And inside the store there will be many slices of particular data.

## 1) Create redux store

Create this in utils .

It will be created with **configureStore**.

```
import { configureStore } from "@reduxjs/toolkit";

const store = configureStore({

});

export default store;
```

## 2) Create Slice

Create a slice with api – **createSlice** :

This takes in parameter **name , initialState , reducers**

Inside reducers – there are states and actions through which we can manipulate data.

```
import { createSlice } from "@reduxjs/toolkit";

const appSlice = createSlice({
  name: "app",
  initialState: {
    isMenuOpen: true,
  },
  reducers: {
    toggleMenu: (state) => {
      state.isMenuOpen = !state.isMenuOpen;
    },
    closeMenu: (state) => {
      state.isMenuOpen = false;
    },
  },
});

export const { toggleMenu, closeMenu } = appSlice.actions;
export default appSlice.reducer;
```

Once I created the slice import in into store.

```
import { configureStore } from "@reduxjs/toolkit";
import appSlice from "./appSlice";
```

```

const store = configureStore({
  reducer: {
    app: appSlice,
  },
});
export default store;

```

### 3) Give Provider to app.js

Once you do this , we need to go back into the app.js and we need to provide our store to it.

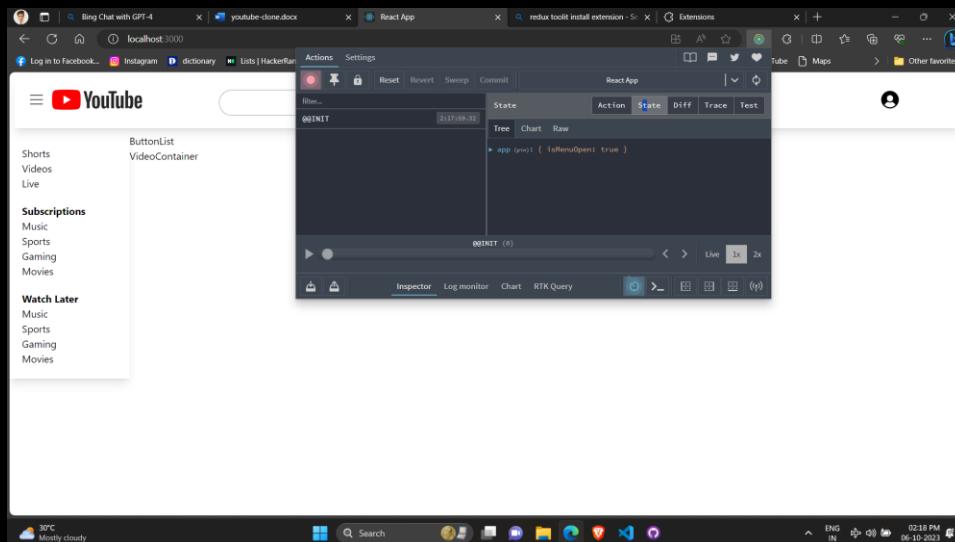
To do that wrap the main div with <provider store = {appStore}>

Need to import appStore from utils.

```

function App() {
  return (
    <Provider store={store}>
      <div>
        <Head/>
        <Body/>
      </div>
    </Provider>
  );
}
export default App;

```



Once we are done with these steps we can check in the redux extension , if the states are visible or not (initial state set to true)

If it is these then its working fine.

----- Commit 1 -----

So now I am trying to toggle the menu bar.

Now that we have state ready lets go to button and write action handler.

<button

```
onClick={()=>toggleMenuHandler}

  className="border border-gray-400 px-5 py-2
  rounded-r-full bg-gray-100">
  <img alt="Search icon" />
</button>
```

As soon as button is clicked , toggleMenuHandler is called . Need to define it.

What does it do??

```
const toggleMenuHandler = () => {
  console.log("toggle menu");
It calls dispatcher here//
```

```
}
```

For that to work need to define dispatcher first : = `const dispatch = useDispatch();`  
UseDispatch is a hook that comes from react redux library.

Then dispatch can be called with appropriate action .

```
import React from "react";
import { useDispatch } from "react-redux";
import { toggleMenu } from "../utils/appSlice";

const Head = () => {
  const dispatch = useDispatch();

  const toggleMenuHandler = () => {

    dispatch(toggleMenu())
  }
}
```

Now that state is changing from ture to false vice verca we can use this to implement it in sidebar so that I will be available only if state is True.

```
const Sidebar = () => {

  const isMenuOpen = useSelector((Store) => Store.app.isMenuOpen);
```

To do that , to get the value of the state ,

We have defined isMenu open in our slice right. So we need to get that value.

To do that we need to subscribe to our store which can be done by useSelector hook.

Now lets create buttons.

Created button :

```
import React from "react";
const Button = ({ name }) => {
  return (
    <div>
      <button className="px-5 py-2 m-2 bg-gray-200 rounded-lg">{name}</button>
    </div>
  );
};
export default Button;
Used destructuring
```

Now I will pass a list of props:

```
import React from "react";
import Button from "./Button";

const list = ["All", "Live", "Gaming"]; use it for map function instead of the whole list
const ButtonList = () => {
  return (
    <div className="flex">
      <Button name="All" />
      <Button name="Gaming" />
      <Button name="Songs" />
      <Button name="Live" />
      <Button name="Soccer" />
      <Button name="Cricket" />
      <Button name="Cooking" />
      <Button name="Cricket" />
      <Button name="Valentines" />
      <Button name="Cricket" />
      <Button name="Cooking" />
      <Button name="Cricket" />
    </div>
  );
};

export default ButtonList;
If I dint use {name} like this then I would have to use props.name
-----same thing can be achived with help of map.
```

```
const list = [
  "All",
  "Live",
```

```

"Gaming",
"Songs",
"Soccer",
"Cricket",
"Cooking",
"Valentines",
// add more items here
];
const ButtonList = () => {
return (
<div className="">
{list.map((name , index) => (
<Button key={index} name={name} />
))}
</div>
);
};
export default ButtonList;

```

Now lets move on to video container.

Now lets use live api to fetch the data. Also we will not get CORS exception because we will get legit api – free api for a limit.

Need to create a key to use the api – My key

**Google Api Key :** -----

AlzaSyA0diAmgUiVwmrm\_-Xrm93PJTVCli2iTk

---

To get api of videos we need to follow some link – search youtube api videos list

<https://developers.google.com/youtube/v3/docs/videos/list>

Lets show most popular videos on the page.

GET  
[https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&chart=mostPopular&regionCode=US&key=\[YOUR\\_API\\_KEY\]](https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&chart=mostPopular&regionCode=US&key=[YOUR_API_KEY]) HTTP/1.1

Authorization: Bearer [YOUR\_ACCESS\_TOKEN]

Accept: application/json

This is the api but we nly need to take whats useful to us

Now where will I call this api ? In my vid container.

With use effect we should call this.

```
YOUTUBE_VIDEOS_API =  
https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&chart=mostPopular  
&regionCode=US&key=[YOUR_API_KEY]
```

This is a hardcoded api so we need to put it in constants.

So we can is at last we need to append our API key

#### My Constants :

---

```
const GOOGLE_API_KEY = "AlzaSyA0diAmgUiVwmrm_-Xrm93PJTVCli2iTk";
```

```
export const YOUTUBE_VIDEOS_API =
```

```
"https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&chart=mostPopular  
&regionCode=US&key=" +
```

```
GOOGLE_API_KEY;
```

---

In useeffect -

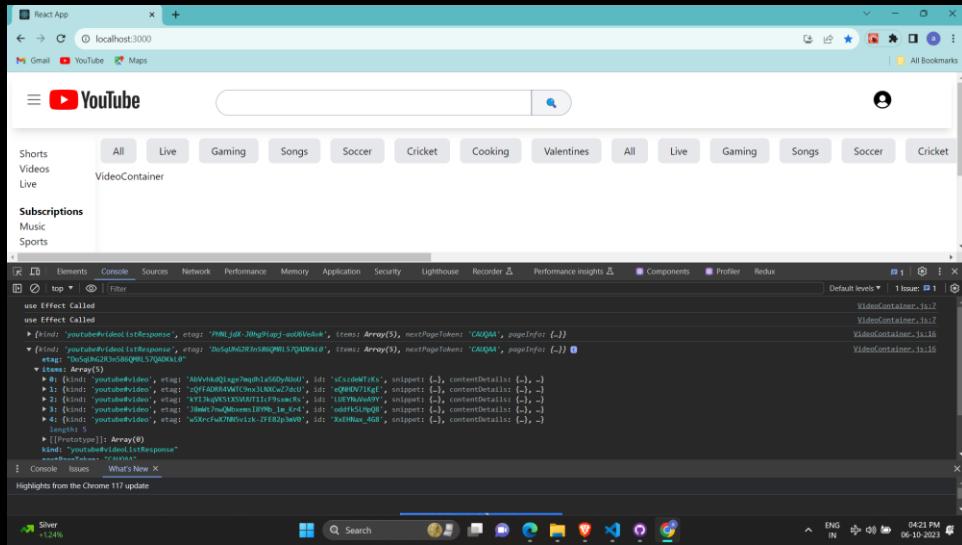
```
useEffect(()=>{  
  console.log("use Effect Called")  
  getVideos()  
} , [])
```

Now we will write the code to get the videos with fetch api

I RAN INTO AN ISSUE : - where youtube API was not enabled

- 3) Visit the Google Cloud Console.
- 4) Select your project from the dropdown on the top.
- 5) Navigate to APIs & Services > Library.
- 6) Search for YouTube Data API v3 and select it.
- 7) Click on ENABLE.

This is the solution and I was able to fetch correctly after this.



Now it works but it's showing 2 times why? Because of strictmode – it runs few more queries , if you remove strictmode it will also go away.

Success , now we got data , but how to extract all these things? How to use them?

I will create a state variable. So that once the json got data It needs to be updated to our variable so that we can loop over them.

```
const VideoContainer = () => {  
  
  const [videos, setVideos] = useState([])  
  
  useEffect(()=>{  
    getVideos()  
  
  } , [])  
  
  const getVideos = async () => {  
    const data = await fetch(YOUTUBE_VIDEOS_API);  
    const json = await data.json();  
  }  
};
```

```

    setVideos(json.items);
};

return (
  <div>
    <VideoCard info={videos[0]} />
  </div>
)
}

```

This is how we can extract the data and pass it over as props . Right now I am getting only one card but I will do it with map so that I can have multiple cards.

```

<div>
  {videos.map((video) => () => <VideoCard info={video} />)}
</div>

```

But I am getting only 5 videos , so I need to use something from api – maxresults.

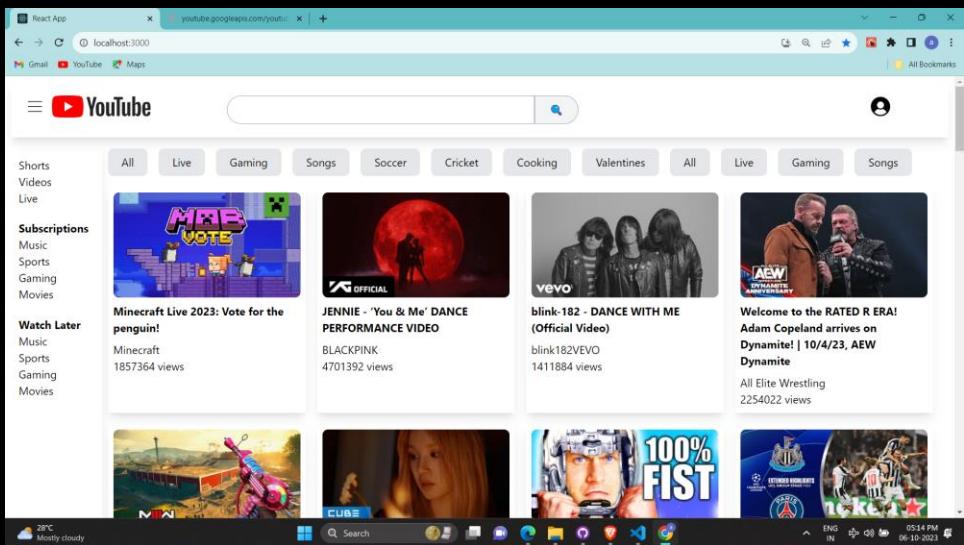
Slight modification in the api:

```
const GOOGLE_API_KEY = "AlzaSyA0diAmgUiVwmrm_-Xrm93PJTVCli2iITK";
```

```
export const YOUTUBE_VIDEOS_API =
```

```
"https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&chart=mostPopular
&maxResults=50&regionCode=US&key=" +
```

```
GOOGLE_API_KEY;
```



This is the ui : yaaaaay

---

Now lets install routing

### \*\*\* Npm I react-router-dom

How to create router ??

CreateBrowserRouter API in app.js

```
const appRouter = createBrowserRouter([
  {
    path: "/",
    element: <Body />,
    children: [
      {
        path: "/",
        element: <MainContainer />,
      },
      {
        path: "watch",
        element: <WatchPage />,
      },
    ],
  },
]);

function App() {
  return (
    <Provider store={store}>
      <div>
        <Head />
        <RouterProvider router={appRouter} />
    </div>
  );
}

export default App;
```

```

        </div>
    </Provider>
);
}
export default App;

```

So this is the rout , why I have done like this??

See , the boy will load either maincontainer or watchpage which depends on the path.

So inside body we have given children.

Also you can see head remains same so we do not need to give routerprovider inside that.

```

const Body = () => {
  return (
    <div className="flex">
      <Sidebar />
      <Outlet />
    </div>
  );
};

```

Also body also got changed.

```

const Body = () => {
  return (
    <div className="flex">
      <Sidebar />
      <Outlet />
    </div>
  );
};

```

This outlet is dynamic it takes whichever is being loaded. This comes from RRD.

-----now lets build watchpage , suppose if I click on a videocard , it will get the id and goto its watchpage.

```

<div className="flex flex-wrap">
  {videos.map((video) => (
    <Link to={`/watch?v=${video.id}`}>
      <VideoCard key={video.id} info={video} />
    </Link>
  ))}
</div>

```

```
);
```

This link is a routing mechanism given to us by RRD.

Now lets read the id of each video that we click , and once we get the id we can go to that path .

To read id – lets use `useSearchParams` to get v – id value ,

Once I get that I will provide iframe to load the video , but we have given link to watch page with id , so it will open in another page.

```
<iframe
  width="1200"
  height="600"
  src={"https://www.youtube.com/embed/" + searchParams.get("v")}
  title="YouTube video player"
  frameBorder="0"
  allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share"
  allowFullScreen
></iframe>
```

Also the close menu is given inside `useEffect` because when we open another page it should automatically close , so once it loads it will close because we have given `state = false` in dispatch action.

```
import React, { useEffect } from "react";
import { useDispatch } from "react-redux";
import { useSearchParams } from "react-router-dom";
import { closeMenu } from "../utils/appSlice";
const WatchPage = () => {
  const [searchParams] = useSearchParams();
  console.log(searchParams.get("v"));
  const dispatch = useDispatch();
  useEffect(() => {
    dispatch(closeMenu());
  }, []);
  return (
    <div className="px-5">
      <iframe
        width="1200"
        height="600"
```

```
src={"https://www.youtube.com/embed/" + searchParams.get("v")}  
title="YouTube video player"  
frameBorder="0"  
allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share"  
allowFullScreen  
></iframe>  
</div>  
);  
};  
export default WatchPage;
```

-----commit4-----