

17.(BONUS) Machine Coding Interview & Building YouTube

00:00 - Intro to Youtube features

- Requirement Clarification
- Features
- Tech Stack
 - Redux
 - Tailwind
 - React Router DOM
 - Bundler
 - Jest, React-testing

08:00 - 17:37 => Interview Tips & Diagram (Mock UI)

19:18 - `npx create-react app my-youtube`

21:00 - Installed Packages Explanation

24:35 - `reportWebVitals.js`

25:45 - `<React.StrictMode>`

28:17 - Command - `npm i -D tailwind CSS`

31:10 - 3 command should be there in `App.css`

- `@tailwind base`
- `@tailwind components`
- `@tailwind utilities`

36:38 - Rough blueprint of what we need to build

- Head
- Body
- Sidebar
- MenuItem
- MainContainer
- ButtonList
- VideoContainer
- VideoCard

39:00 - Coding Starts

42:22 - Building Header Component

46:34 - Using CSS Grid

- Styling Header Component using Tailwind CSS

54:22 - Building Search bar

56:28 to 01:06:56 - Building Side Bar

01:09:06 - npm i @reduxjs/toolkit

- Npm i react-redux

01:11:00 -Till 01:15:20 -

Building Redux Store

01:16:40 - Toggle Functionality

01:19:20 - Cursor Pointer

01:21:30 - Subscribe to the store

01:22:35 - Coding Pattern: Early Return Pattern

01:25:25 - Let's build Main Container

- Button Component
- ButtonList Component
- Props Usage

01:31:25 - Asking question to interviewer (APIs or Hardcoded)

01:33:30 - YouTube APIs and YouTube API Keys

01:38:12 - API Call - Axios v/s Fetch

01:45:00 - useState Hook

01:47:45 - Tips: Don't use Map function at the programming

01:52:00 - Developing Video Card Component

01:58:00 - Calling 50 videos through API

02:01:00- "India" - Regional Code to get popular youtube videos in India

02:02:00 - Routing - npm react router-dom

02:07:00 - Report of explanation about Routing once again

02:10:46 - Start building “watchpage”

02:13:40 - make “home” as clickable functionality

02:19:00 - Usage of useSearchParams

02:21:06 till 02:27:00 - Making WatchPage video fullscreen

02:28:00 - API for Youtube “comments”

02:28:22 -Tips for machine Coding interview

END

18. (Bonus) Advanced Features in YouTube Project

00:00 - Higher Order Components

11:21 - Interview Question - Build Search Functionality, Search API

21:21 - Debouncing

30:00 - Google Search API

33:50 - YouTube Search Suggestions API

35:30 - Make an API Call

36:26 - Let’s build Debouncing

44:45 - Make an API call after every key press that if the difference between 2 API calls is < 200ms, decline the API call

55:00 - How many times API has been called ?

01:03:03 - Building Youtube Search Suggestions bar

01:24:50 - usage of “Data Structures” in Front-end

01:32:00 - Spread Operator in JavaScript [ES6]

01:36:00 - Creating A Store

01:44:00 - JavaScript Merge Objects (object.assign())

01:49:49 - We have achieved a well organized/optimized search [“debouncing”, “cache”] - features

01:55:20 - Developing LRU Cache(Last Recently Used)

01:59:00 - Interview Question

- N- level nested components(Reddit Example)
- Comments Component

19.(Bonus) Wrapping up YouTube Project

00:00 - API key expired - Added new key

- Building Live Chat in Youtube
- Live Chat - Infinite Scroll - Pagination
- Challenges -
- Data Layer(DL) - Get data live
- UI Layer - Update the UI

11:15 - Data Live

- Web Sockets - API Polling

16:50 - Does Gmail is webSocket or API Polling?\

22:50 - Youtube Comment section uses API Polling

41:33 - Why page does not freeze despite so much API calls?

49:20 - Building Chat bar

01:00:00 - Never write “map” as the first. Later after multiple components working fine.

01:03:00 - Coding Polling(Live)

- Don't forget to clear the interval or timeout (setInterval, SetTimeout)

01:05:52 - Lets build a redux store

Till 01:13:52 - Perfect mock example of writing Redux Code(Slice)

01:14:00 - use "map"

01:17:00 - "Overflow hidden"

01:18:00 - Random name Generator

01:20:40 - Random Strings/ Text Generator

01:23:45 - Reverse Scrolling - "Unshift" in Redux code - "Splice" usage

01:30:00 - Build a Input box to write a comment

01:37:30 - useState

01:41:00 - Prevent Default Function

01:47:00 - Push() instead of unShift()

01:52:05 - To make infinite Scroll

End

-----***-----***-----