**YOUTUBE CLONE USING REACT JS**

**A MINI - PROJECT REPORT**

***Submitted by***

**AJAY. M 821020104005**

**SIVAGANESH. K 821020104034**

**VASANTH SURIYA. P 821020104046**

**VIJAY. V 821020104047**

***In partial fulfilment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**K.S.K COLLEGE OF ENGINEERING AND TECHNOLOGY**

**DARASURAM-612 702**

**ANNA UNIVERSITY:CHENNAI 600 025**

**2023**

**ANNA UNIVERSITY:CHENNAI 600 025**

****

**BONAFIDE CERTIFICATE**

Certified that this Project report **“YOUTUBE CLONE USING REACT JS”**

Is the bonafide work of **“AJAY.M (821020104005), SIVAGANESH.K (821020104034), VASANTH SURIYA.P (821020104046), VIJAY.V(821020104047), ”** who carried out the Mini-Project during the academic year of 2022-2023.

**SIGNATURE SIGNATURE**

Mrs. K.PAPATHI, M.E Mr. P.PARTHIBAN, M,E

**HEAD OF THE DEPARTMENT SUPERVISOR**

ASSISTANT PROFESSOR ASSISTANT PROFESSOR

DEPARTMENT OF CSE DEPARTMENT OF CSE

K.S.K College of Engineering K.S.K College of Engineering

and Technology and Technology

Darasuram – 612 702 Darasuram – 612 702

Submitted for the project Viva Voce held on ………………………

INTERNAL EXAMINER EXTERNALEXAMINER

ACKNOWLEDGEMENT

First and foremost we wish to thank the God for guiding us in all ways to complete the project successfully.

We feel honoured to place our warm salutation to **K.S.K College Of Engineering and Technology** Darasuram and Department of CSE, which has given us the opportunity to obtain chaired goal of Beckoning professional.

We would like to extend our heartfelt thanks to our beloved **Chairman of our college MR.K.S.Krishnamoorthy** and to **Secretary of our college Mr.K.S.K.Duraimurugan,M.E,** and also enlarge our respectful I thanks to our **Managing Trustee Mrs.D.Swathy Duraimurugan,MBA,M.S(U.K)** who has given as an Excellence opportunity to exhibit our mode of creativity by this project.

We would like to avail this opportunity to express our gratitude and sincere thanks to our **Principal Dr.R.Rajavignesh,M.E.,Ph.D.,** who has provide all help in executing in this project successfully.

We take this opportunity to express our gratefulness and thanks to **Head of the Department and Project Co-ordinator Mrs.K.PAPATHI, M.E.,Assistant professor** Department of CSE to complete our course successfully.

We would like to avail this opportunity to express our sincere thanks to our Project Co-ordinator **Mr.P.Parthiban, M.E.., Assistant professor** Department of CSE who encouraged as in each and every step of this project to complete it successfully.

Finally we would like to use this opportunity to express our sincere thanks to our parents and all Teaching staff and Non-Teaching staff providing all help for successful completion of this project.

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **PARTICULARS** | **PAGE NO** |
| 1 | INTRODUCTION | 6 |
| 2 | ABSTRACT | 7 |
| 3 | SYSTEM DESIGN,  IMPLEMENTATION | 8 |
| 4 | SYSTEM CONFIGURATION  HARDWARE REQUIREMENTS SOFTWARE REQUIREMENTS | 11 |
| 5 | SOFTWARE SPECIFICATION HTML,CSS,JAVASCRIPT,REACT,RAPID API | 12 |
| 6 | CODINGS | 21 |
| 7 | SCREENSHOTS | 23 |
| 8 | TESTING | 28 |
| 9 | CONCLUSION | 29 |
| 10 | REFERNCES | 30 |

5

**1. INTRODUCTION**

The introduction of the YouTube clone mini project is a fully functional website that mimics the essential functionalities of the popular video sharing platform, YouTube. The completed project will include the following features:

* User registration and login: The website will allow users to register for an account and log in using their credentials.
* Video uploading: Registered users will be able to upload videos to the website.
* Video viewing: Users will be able to view videos uploaded by other users on the website.
* Video search: Users will be able to search for videos based on keywords.
* User profiles: Each registered user will have a profile page that displays their uploaded videos and other relevant information.
* Video comments: Users will be able to leave comments on videos.
* Security features: The website will include security features such as password hashing and cross-site scripting prevention.

The outcome of the project will also provide individuals with practical experience in web development concepts and techniques, including database design, front-end and back-end development, API integration, and user authentication. Individuals will also gain experience in project management, problem-solving, and debugging.

6

**2. ABSTRACT**

The YouTube clone mini project is a web development project that aims to provide individuals with an opportunity to learn and develop their web development skills by creating a simplified version of the popular video sharing platform, YouTube. The project involves building a website from scratch using various web development technologies such as HTML, CSS, JavaScript, React, and RapidAPI.

The completed project will include essential functionalities such as uploading videos, viewing videos, searching for videos based on keywords, and user authentication. It will also include security features such as password hashing and cross-site scripting prevention.

The project provides individuals with practical experience in web development concepts and techniques, including database design, front-end and back-end development, API integration, and user authentication. It also allows individuals to gain experience in project management, problem-solving, and debugging.

Overall, the YouTube clone mini project provides a fun and rewarding opportunity for individuals to develop their web development skills, gain practical experience, and showcase their capabilities to potential employers and clients

7

**3. SYSTEM DESIGN**

The YouTube clone mini project can be designed and implemented using the following components.

Front-end Development: The front-end development can be done using React, a popular JavaScript library for building user interfaces. The front-end development will include designing the user interface, implementing user authentication, and displaying video content.

Back-end Development: The back-end development can be done using PHP, a server-side scripting language used for web development. The back-end development will include implementing the database, video uploading functionality, and video search functionality.

Database Design: The project will require a database to store user data, video data, and other relevant information. The database design will include creating tables, setting up relationships between tables, and implementing data validation.

API Integration: The project will integrate with the RapidAPI, which provides access to YouTube's API. The API integration will enable the project to access video content and other features provided by YouTube.

Security Features: The project will include security features such as password hashing and cross-site scripting prevention. These features will help protect user data and prevent malicious attacks.

Testing and Debugging: The project will require testing and debugging to ensure that it works as expected and to identify and fix any bugs or issues.

8

**IMPLEMENTATION**

* Design the User Interface: The user interface should be designed to mimic the essential functionalities of YouTube, such as video playback, search functionality, and user authentication.
* Implement User Authentication: The user authentication system should be implemented to allow users to create accounts, log in, and access their profiles.
* Implement Video Upload Functionality: The video upload functionality should be implemented to enable users to upload videos to the platform.
* Implement Video Search Functionality: The video search functionality should be implemented to allow users to search for videos based on keywords or tags.
* Integrate with RapidAPI: The project should be integrated with the RapidAPI to enable access to YouTube's API and video content.
* Implement Database Design: The database design should be implemented to store user data, video data, and other relevant information.
* Implement Security Features: The project should include security features such as password hashing and cross-site scripting prevention to protect user data and prevent malicious attacks.

9

**EXISTING SYSTEM**

In the existing system viewers can upload videos, watch videos, commenting and interaction and do more things.

**Disadvantages of Existing System**

* Inappropriate Content
* Copyright Infringement
* Monetization Challenges
* Limited controls
* Algorithmic Bias

**PROPOSED SYSTEM**

In the proposed System would allow users to upload videos with feautures such as trimming, adding annotations and so on.copyright free content for all the videos.

**Advantages of Proposed System**

* AD free contents
* Less Data Usage
* Security
* Customization
* Performance

The outcome of the project will also provide individuals with practical experience in web development concepts and techniques, including database design, front-end and back-end development, API integration, and user authentication. Individuals will also gain experience in project management, problem-solving, and debugging, which are valuable skills in the tech industry.

Additionally, the completed project can serve as a portfolio piece to showcase their skills and capabilities to potential employers and clients. Overall, the outcome of the YouTube clone mini project is a valuable learning experience that can provide individuals with practical skills and knowledge in web development

10

**4. SYSTEM REQUIREMENTS**

**Hardware Requirement**

Processor **:** Intel Pentium i3

Speed **:** 2.4 GHZ

RAM **:** 2GB

Hard Disk **:** 500GB

Monitor **:** ACER Monitor

Mouse **:** Acer Mouse

Keyboard **:** 104 keys

**Software Requirements**

Operating System  **:** Windows 10

Front End **:** REACT JS

Back End **:** RAPID API

Coding Language **:** HTML, CSS , JAVASCRIPT

Code Editor **:** Visual Studio Code

11

**5. SOFTWARE DESCRIPTION**

**LANGUAGE SPECIFICATION: HTML**

HTML (Hypertext Markup Language) is a markup language used to create web pages and other types of content that can be displayed on the internet. HTML is a standardized language with a set of rules and specifications that must be followed to ensure compatibility across different web browsers and platforms. Here are some of the key specifications of HTML:

* Elements: HTML documents are made up of elements, which are enclosed in tags. Each element has a specific purpose, such as displaying text, images, or links. Tags are used to indicate the beginning and end of an element, and can also contain attributes that provide additional information about the element.
* Document Structure: An HTML document consists of several parts, including the doctype declaration, the head section, and the body section. The doctype declaration specifies the version of HTML being used, while the head section contains information about the document such as the title, metadata, and links to external resources. The body section contains the actual content of the document, such as text, images, and other media.
* HTML Versions: HTML has gone through several revisions over the years, with the most recent version being HTML5. Each version adds new features and functionality to the language, while also deprecating older features that are no longer used or supported.
* Semantic Markup: Semantic markup is a way of using HTML elements to describe the content of a web page in a way that is meaningful to both humans and machines. For example, using the <header> element to define the header of a page.

12

* Accessibility: HTML includes features that allow web developers to create accessible content for people with disabilities, such as alt attributes for images, and aria labels for non-text content. Following accessibility guidelines can help ensure that your content can be accessed and understood by the widest possible audience.

Overall, HTML is a versatile and powerful language that is essential for creating web content that is both functional and visually appealing. By following the specifications and best practices of the language, you can create web pages that are accessible, responsive, and compatible with a wide range of devices and browsers.

**HTML –Syntax**

HTML (Hypertext Markup Language) uses a specific syntax to create and structure web pages. Here are some of the key components of HTML syntax:

* Tags: HTML uses tags to define different parts of a web page. Tags are enclosed in angle brackets (< >) and come in pairs: an opening tag and a closing tag. The content that is enclosed between the opening and closing tags is affected by the tag.
* Elements: An HTML element consists of an opening tag, some content, and a closing tag. The content is the text, images, or other media that is affected by the tags.
* Attributes: HTML tags can have attributes, which are used to provide additional information about the element. Attributes are added to the opening tag and consist of a name and a value, separated by an equals sign. For example, the <img> tag might have an attribute like src="image.png", which specifies the URL of the image to be displayed.
* Comments: HTML comments are used to add notes to the code that are not displayed in the browser. Comments start with <!-- and end with -->.

13

* Document structure: HTML documents have a specific structure that consists of a doctype declaration, a head section, and a body section. The doctype declaration specifies the version of HTML being used, while the head section contains information about the document such as the title, metadata, and links to external resources. The body section contains the actual content of the document.
* Nesting: HTML tags can be nested inside one another to create more complex structures. The outermost tag is the parent, while the tags inside it are children.

**HTML CODE**

<!DOCTYPE html>

<html>

<head>

<title>My Web Page</title>

</head>

<body>

<h1>Welcome to my web page!</h1>

<p>This is some text on my page.</p>

<img src="image.png" alt="A picture of something.">

</body>

</html>

This code creates a basic HTML document with a title, heading, paragraph, and image.

14

**How to Save HTML pages**

To save your HTML page and run it in a web browser, follow these steps:

1. Create your HTML file and save it with a .html extension. You can use any text editor to create an HTML file, such as Notepad, Sublime Text, or Visual Studio Code.

2. Open the HTML file in a web browser to preview it. You can do this by double-clicking the file or right-clicking the file and selecting "Open With" and then choosing your preferred web browser.

3. If your HTML file references external files such as images, stylesheets, or JavaScript files, make sure they are all in the same folder as your HTML file or in a subfolder within that folder.

4. To run the HTML file, simply double-click it or open it in a web browser by right-clicking the file and selecting "Open With" and then choosing your preferred web browser.

If you want to share your HTML page with others, you can upload it to a web server or hosting service. Some popular options for hosting HTML files include GitHub Pages, Netlify, and Firebase Hosting. Once you have uploaded your file, you can share the link with others to view your HTML page in their web browser.

**LANGUAGE SPECIFICATION: CSS**

CSS (Cascading Style Sheets) is a language used for describing the presentation of web pages. It allows web designers to control the layout, formatting, and appearance of HTML documents, including things like colors, fonts, backgrounds, and positioning of elements on the page. CSS is used in conjunction with HTML to create visually appealing and well-structured web pages.

CSS consists of a set of rules or styles that are applied to specific HTML elements. These rules are written in a syntax that includes selectors, properties, and values. A selector is used to target a specific HTML element, and the properties and values are used to define the desired appearance of that element.

15

**CSS CODE WITH HTML**

Here's a simple example of CSS that changes the color and background of a button when it's hovered over:

**HTML:**

<button class="my-button">Click me!</button>

**CSS:**

.my-button {

color: white;

background-color: blue;

border: none;

padding: 10px 20px;

border-radius: 5px;

font-size: 16px;

}

.my-button:hover {

color: blue;

background-color: white;

}

In this example, the CSS is applied to the button element with the class "my-button". The first rule sets the initial style for the button, including its color, background color, padding, font size, and border radius. The second rule changes the color and background of the button when it's hovered over.

When the button is not hovered over, it will have white text on a blue background. But when it's hovered over, the text will change to blue, and the background will change to white.

16

**LANGUAGE SPECIFICATION: JAVASCRIPT**

JavaScript is a high-level, dynamic, and interpreted programming language that is widely used for developing web and mobile applications. It is a versatile language that can be used for both front-end and back-end development, and it is supported by all modern web browsers.

JavaScript was developed by Brendan Eich at Netscape in 1995 and has since evolved into one of the most popular programming languages in the world. It is often used in conjunction with HTML and CSS to create interactive and dynamic web pages.

JavaScript has a syntax similar to other C-style programming languages and consists of a set of statements that are executed in a specific order. Some key features of JavaScript include:

1. Variables: JavaScript allows you to declare variables using the `var`, `let`, or `const` keywords, which can store values of various types, including numbers, strings, objects, and arrays.

2. Functions: JavaScript functions are blocks of code that can be called and executed when needed. Functions can take parameters and return values, and they can be defined using the `function` keyword or as arrow functions.

3. Objects: In JavaScript, an object is a collection of key-value pairs, where the key is a string, and the value can be any data type. Objects can be created using object literals, constructor functions, or classes.

4. Control Flow: JavaScript supports several control flow statements, including `if`/`else` statements, `for` and `while` loops, `switch` statements, and more.

5. Events: JavaScript allows you to attach event handlers to HTML elements, such as clicks, mouse movements, and keyboard presses, allowing you to create interactive and dynamic web pages.

6. DOM Manipulation: JavaScript can be used to manipulate the Document Object Model (DOM), which represents the structure of an HTML document, allowing you to change the content and appearance of web pages dynamically.

17

**JAVASCRIPT CODE WITH HTML**

Here's a simple example of JavaScript that displays a message in an alert box when a button is clicked

**HTML:**

<button id="my-button">Click me!</button>

**JAVASCRIPT:**

const button = document.getElementById('my-button');

button.addEventListener('click', () => {

alert('Hello, World!');

});

In this example, the JavaScript code first selects the button element with the ID "my-button" using the `getElementById` method. Then, it adds an event listener to the button using the `addEventListener` method, which listens for a "click" event and executes the provided callback function when the button is clicked.

When the button is clicked, the callback function displays an alert box with the message "Hello, World!".

This is just a simple example of what you can do with JavaScript. It can be used to create much more complex and dynamic web applications, such as interactive forms, animations, and full-featured single-page applications.

**LANGUAGE SPECIFICATION : REACT JS**

ReactJS is a popular JavaScript library used for building user interfaces. It was developed by Facebook and released in 2013 as an open-source project. React allows developers to create reusable UI components that can be used to build complex web applications.

18

One of the key features of React is its use of a virtual DOM (Document Object Model), which is an in-memory representation of the actual DOM. React uses the virtual DOM to optimize the rendering of components by only updating the parts of the UI that have changed, rather than re-rendering the entire UI.

React also uses a declarative approach to programming, where developers describe what the UI should look like based on the current state of the application, rather than specifying how the UI should be updated. This makes it easier to reason about and maintain complex UIs.

Other features of React include:

1. Component-based architecture: React allows developers to build applications by breaking them down into small, reusable components, making it easier to manage and maintain code.

2. JSX syntax: React uses a syntax extension called JSX, which allows developers to write HTML-like code inside JavaScript. This makes it easier to create and manage UI components.

3. Unidirectional data flow: React uses a unidirectional data flow, where data flows from parent components to child components. This makes it easier to manage and maintain data in complex applications.

4. React Native: React can also be used to build native mobile applications for iOS and Android using a framework called React Native.

Overall, React is a powerful and flexible library that can be used to build a wide range of web applications, from simple static websites to complex single-page applications. Its popularity has led to a large community of developers and a rich ecosystem of tools and libraries.

19

**LANGUAGE SPECIFICATION: RAPID API**

RapidAPI is a platform that provides a unified interface to access and manage APIs (Application Programming Interfaces) from various providers. It was launched in 2014 and has since grown to become one of the largest API marketplaces, with over 35,000 APIs from more than 500 providers.

RapidAPI allows developers to search for and use APIs in a variety of categories, including weather, news, social media, finance, and more. Developers can test and integrate APIs directly from the RapidAPI platform, and also manage API keys and usage limits in a centralized dashboard.

Some of the key features of RapidAPI include:

1. API Marketplace: RapidAPI provides a marketplace where developers can discover and access APIs from a variety of providers.

2. API Management: RapidAPI allows developers to manage API keys and usage limits in a centralized dashboard.

3. API Testing: RapidAPI provides a testing console where developers can test and debug APIs before integrating them into their applications.

4. API Analytics: RapidAPI provides analytics and usage data for APIs, allowing developers to monitor and optimize API usage.

5. API Monetization: RapidAPI provides a platform for API providers to monetize their APIs by offering different pricing plans and revenue-sharing models.

Overall, RapidAPI provides a comprehensive platform for developers to discover, test, and integrate APIs from various providers. Its large API marketplace and centralized API management features make it a popular choice among developers for API integration.

20

**6. CODINGS:**

**Index.html**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <meta name="theme-color" content="#000000" />

    <meta name="description" content="Web site created using create-react-app" />

    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

    <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap" />

    <link rel="stylesheet" href="https://fonts.googleapis.com/icon?family=Material+Icons" />

    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

    <title>JSM Media</title>

  </head>

  <body>

    <noscript>You need to enable JavaScript to run this app.</noscript>

    <div id="root"></div>

  </body>

</html>

**index.js**

import React from 'react';

21

import ReactDOM from 'react-dom/client';

import App from './App';

import './index.css';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**fetchFromAPI.js**

import axios from 'axios';

export const BASE\_URL = 'https://youtube-v31.p.rapidapi.com';

const options = {

  params: {

    maxResults: 50,

  },

  headers: {

    'X-RapidAPI-Key': process.env.REACT\_APP\_RAPID\_API\_KEY,

    'X-RapidAPI-Host': 'youtube-v31.p.rapidapi.com',

  },

};

export const fetchFromAPI = async (url) => {

  const { data } = await axios.get(`${BASE\_URL}/${url}`, options);

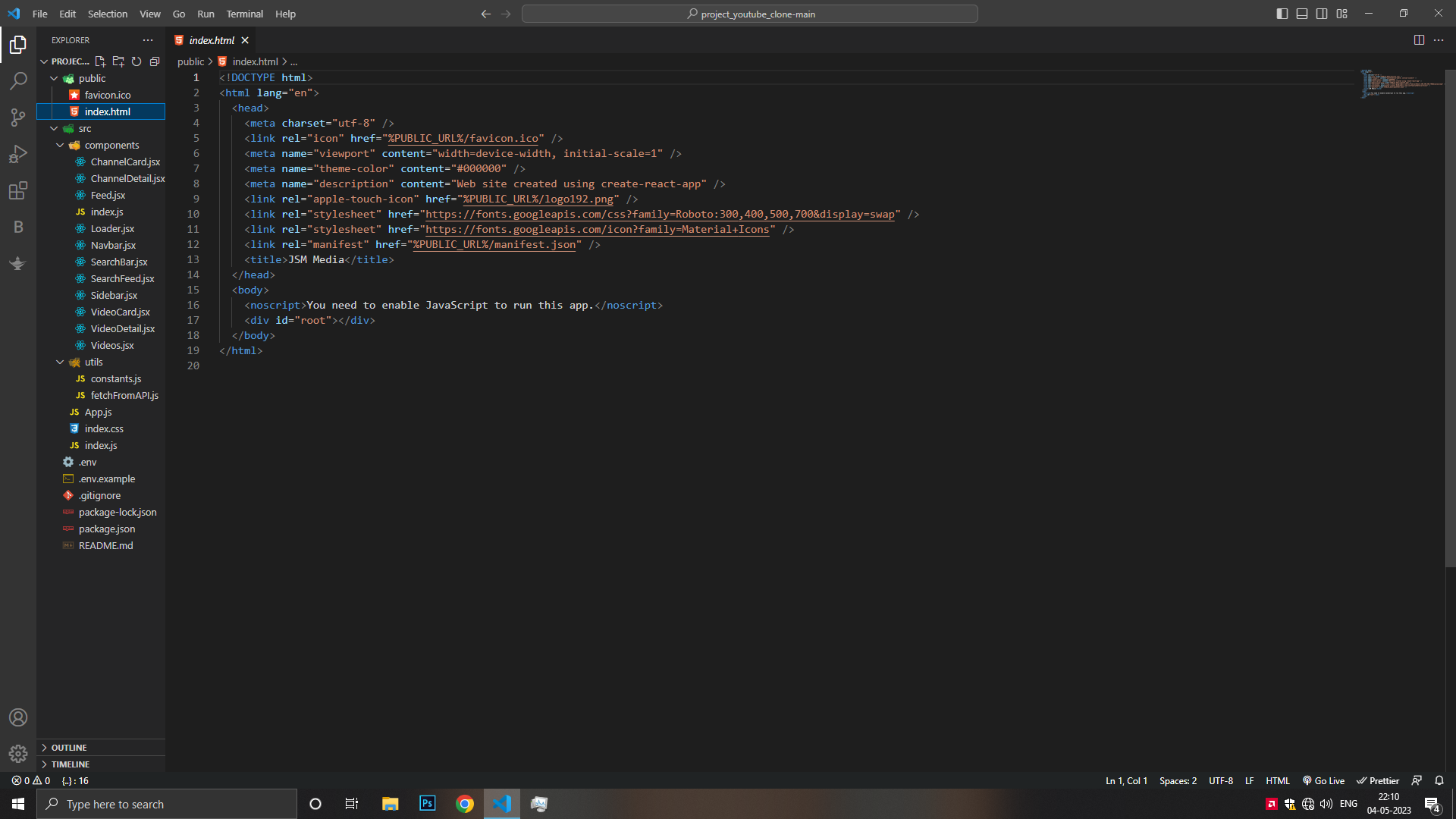
  return data;

};

22

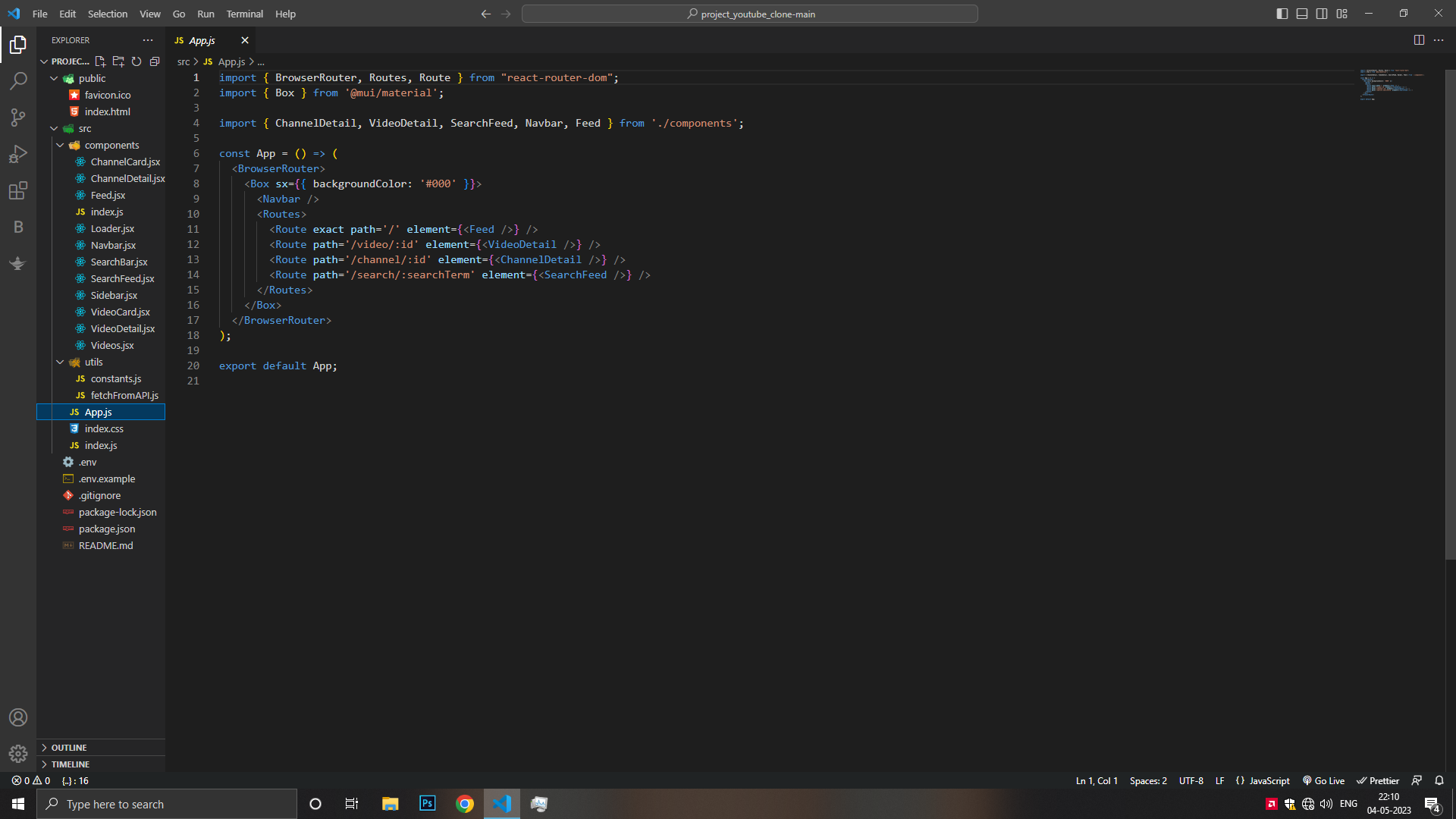
**7. SCREENSHOTS:**

**Index.html**

****

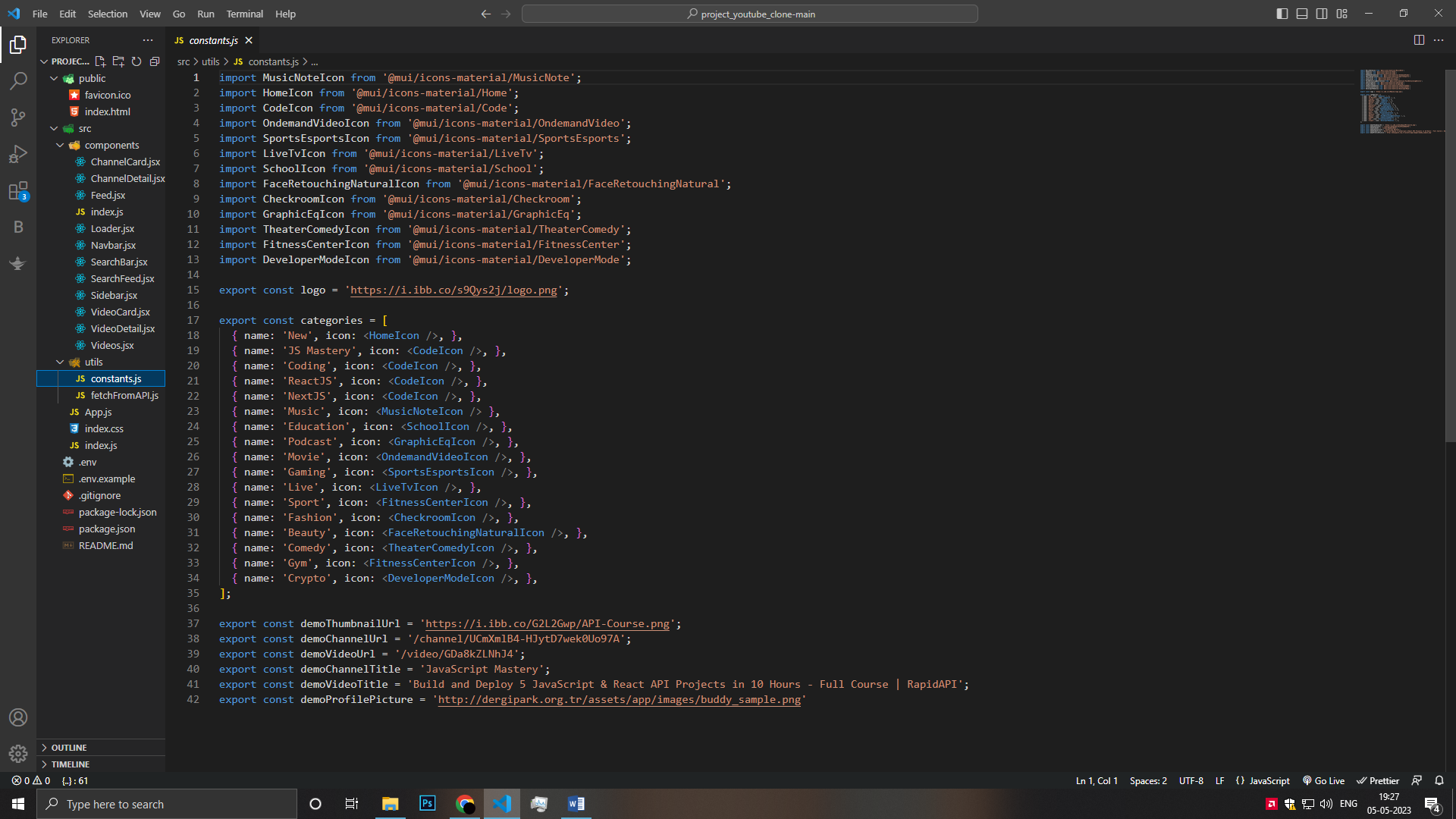
23

**App.js**

****

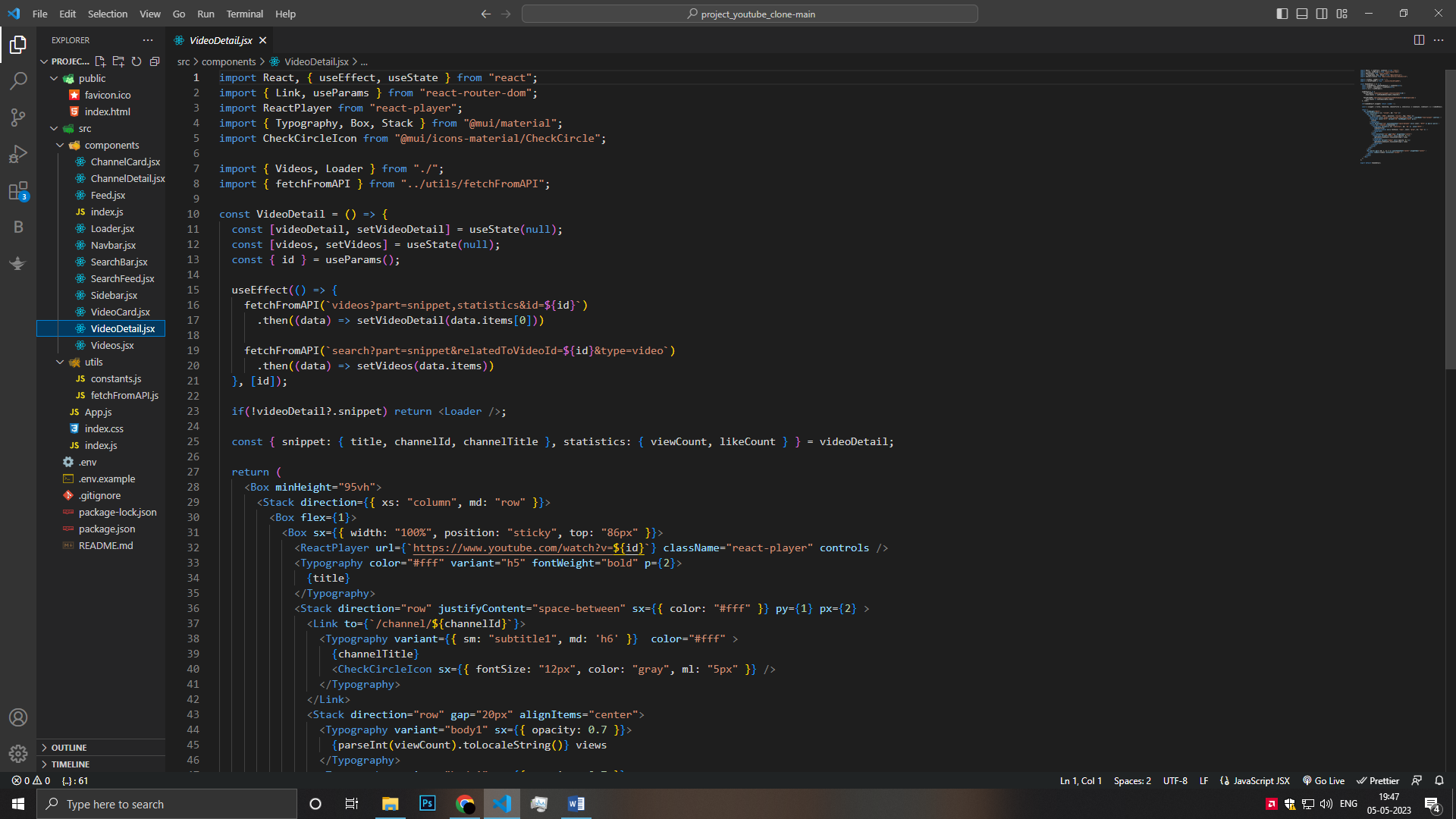
24

**Constant.js**

****

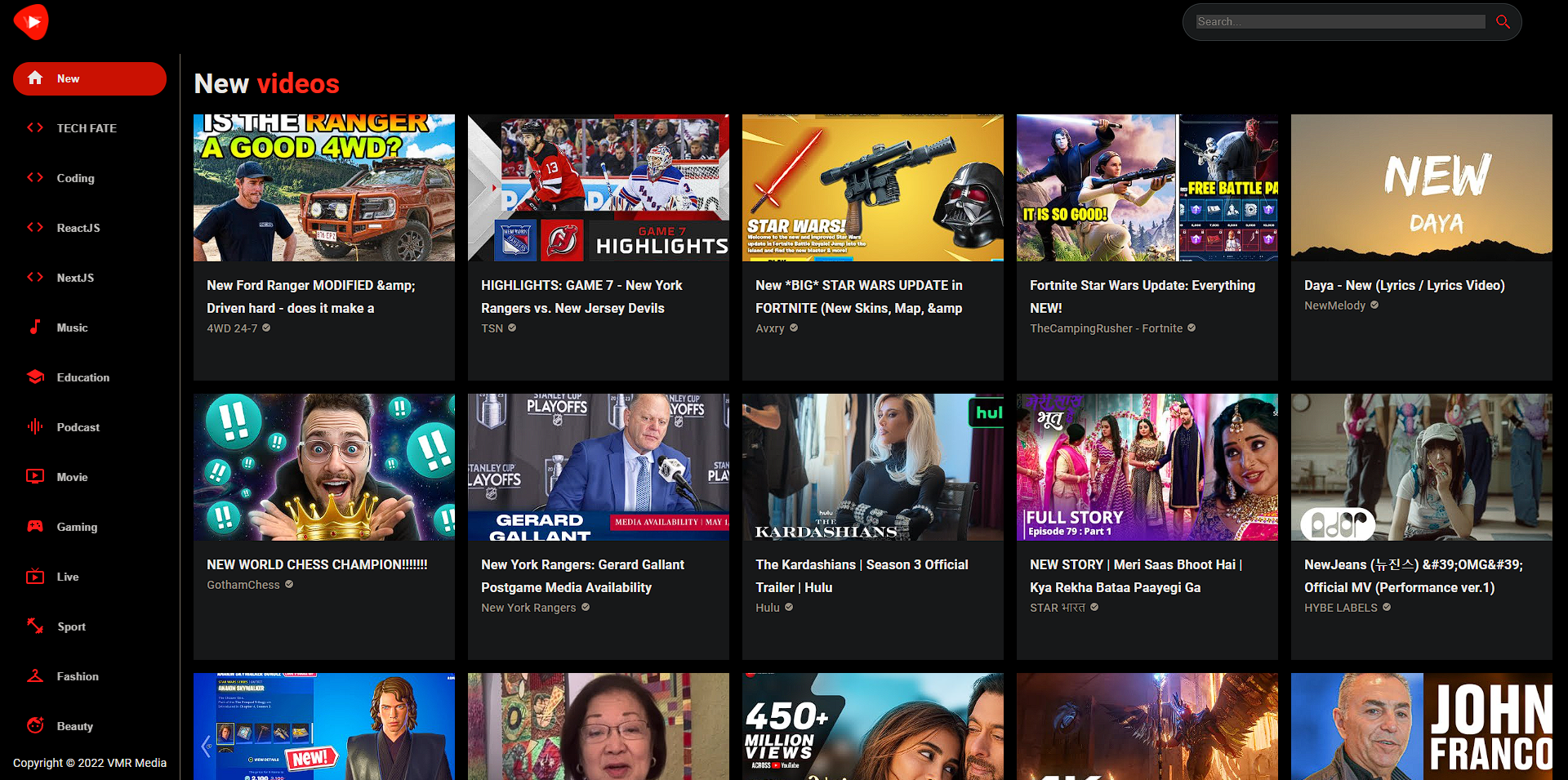
25

**VideoDetail.jsx**

****

26

**OUTPUT :**

****

27

**8. TESTING**

Testing a YouTube clone built with React, RapidAPI, and Material UI would involve a combination of unit tests, integration tests, and end-to-end tests.

Unit tests are used to test individual components in isolation, ensuring that they behave as expected. Integration tests are used to test how multiple components work together. End-to-end tests are used to test the entire application from the user's perspective.

Here are some of tests YouTube clone:

1. Unit test: Test that the video player component correctly renders a video.

2. Integration test: Test that the video search component works correctly with the video list component.

3. End-to-end test: Test that the user can search for a video, select it, and play it in the video player component.

4. Unit test: Test that the RapidAPI client correctly retrieves data from the YouTube API.

5. Integration test: Test that the Material UI components work correctly with the React components.

To perform these tests, you could use tools such as Jest for unit and integration tests, Cypress for end-to-end tests, and React Testing Library for testing React components. You could also use tools such as BrowserStack or Sauce Labs for cross-browser testing.

Overall, testing a YouTube clone built with React, RapidAPI, and Material UI would involve a combination of different types of tests, each with their own focus and purpose. By writing comprehensive tests, you can ensure that your application is reliable, robust, and performs as expected.

28

**9. CONCLUSION**

In conclusion, building a YouTube clone using React, RapidAPI, and Material UI can be a rewarding and challenging project for web developers. React provides a powerful and flexible framework for building user interfaces, while RapidAPI allows for easy integration with external APIs such as the YouTube API. Material UI provides a library of pre-built components that can be used to quickly create a consistent and polished user interface.

However, building a YouTube clone is not without its challenges. The YouTube API has strict usage limits and requires proper authorization to access certain data. Additionally, implementing user authentication and authorization can be complex and time-consuming.

To create a successful YouTube clone, it's important to focus on the user experience, testing, and security. You should also consider implementing features such as search, video playback, user comments, and social sharing. By following best practices and leveraging the strengths of these technologies, you can create a high-quality YouTube clone that demonstrates your skills as a developer and provides a valuable service to users.

Overall, building a YouTube clone using React, RapidAPI, and Material UI can be a great way to gain experience in web development and showcase your abilities to potential employers or clients.

By combining these technologies and best practices, developers can create a high-quality YouTube clone that showcases their skills and creativity. With the growing demand for video content on the web, building a YouTube clone can also be a valuable addition to a developer's portfolio or a great starting point for a new project.

29

**10. REFERENCES**

1. W3C HTML Specification: https://www.w3.org/TR/html52/
2. MDN Web Docs HTML: https://developer.mozilla.org/en-US/docs/Web/HTML
3. W3C CSS Specification: https://www.w3.org/Style/CSS/
4. React documentation: https://reactjs.org/
5. RapidAPI documentation: https://rapidapi.com/docs/
6. Material UI documentation: https://material-ui.com/
7. YouTube API documentation: https://developers.google.com/youtube/
8. Jest testing framework: https://jestjs.io/
9. React Testing Library: https://testing-library.com/docs/react-testing-library/intro/
10. Cypress testing framework: https://www.cypress.io/
11. BrowserStack cross-browser testing: https://www.browserstack.com/
12. Sauce Labs cross-browser testing: <https://saucelabs.com/>

30