Mastering MERN: Full Stack Integration

In this session, we'll cover key topics for connecting our React frontend to the backend APIs, including posting and fetching data, authentication, and best practices for production-ready full stack applications.

Connecting React to the Backend

1   **Make API Calls with Fetch or Axios**

Use the browser's native Fetch API or a library like Axios to make HTTP requests to your backend endpoints.

2   **Manage API Endpoints and Data Fetching**

Create reusable API service modules to handle endpoint URLs and data transformation.

3   **Integrate with React Components**

Fetch data in useEffect hooks and update your UI based on the response.

Posting Data to APIs

Create New Resources

Use HTTP POST requests to send data to your backend and create new database records.

Handle Form Submissions

Capture user input, prepare the data, and send it to the API when a form is submitted.

Validate and Sanitize Data

Ensure data integrity by validating all user input and sanitizing it before sending to the backend.
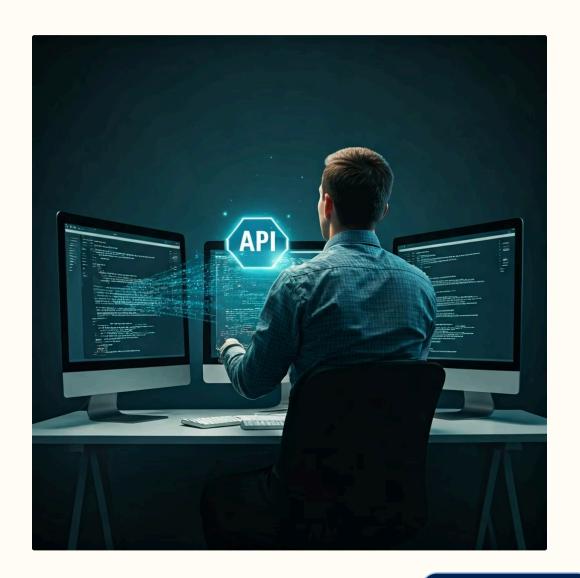
CORS

What is CORS?

Rendering Backend Data

Fetch and Display Data

Use React's state and hooks to fetch data from your backend APIs and render it in your components.

Implement loading and error states to provide a smooth user experience.

Authentication

1

Login Process

User logs in by sending credentials to the backend.

2

JWT Token

The backend validates the credentials and returns a JSON Web Token (JWT).

3

Storing the Token

The client stores the JWT in localStorage or a cookie.

4

Authenticated Requests

The client sends the JWT in the Authorization header for protected API requests.

5

Documentation

JWT.IO - JSON Web Tokens Introduction

Production Best Practices

1	**API Security**	Use JWTs, validate all inputs, and sanitize data to secure your API.
2	**CORS Configuration**	Properly configure CORS on your backend to allow requests from your frontend.
3	**Error Handling**	Gracefully catch and display user-friendly error messages.
4	**Folder Structure**	Use a modular folder structure with separate directories for API, components, and services.
5	**Follow Clean Code Practices**	**KISS, DRY, SOLID, YAGNI—A Simple Guide to Some Principles of Software Engineering and Clean Code**

Key Takeaways

- ## How to Connect React to Backend APIs

  Use Fetch or Axios to make API calls and integrate with React components.

- ## Implement Authentication with JWTs

  Manage the login flow and store JWT tokens securely to protect your API.

- ## Leverage Environment Variables

  Separate dev and prod environments and store sensitive data safely.

- ## Apply Production Best Practices

  Secure your API, optimize your build, and structure your project for maintainability.