Welcome to the

CoGrammar

Tutorial: Full Stack Web Development

The session will start shortly...

Questions? Drop them in the chat. We'll have dedicated moderators answering questions.

CoGrammar

SKILLS FOR LIFE

SKILLS BOOTCAMPS

Department for Education

CoGrammar Presentation

Month 2024

Full Stack Web Development Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Full Stack Web Development Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: www.hyperiondev.com/support

- Report a **safeguarding** incident: www.hyperiondev.com/safeguardreporting

- We would love your **feedback** on lectures: [Feedback on Lectures]()

Skills Bootcamp
8-Week Progression Overview

Fulfil 4 Criteria to Graduation

✅ Criterion 1: Initial Requirements

Timeframe: First 2 Weeks
Guided Learning Hours (GLH): Minimum of 15 hours
Task Completion: First four tasks

Due Date: 24 March 2024

✅ Criterion 2: Mid-Course Progress

60 Guided Learning Hours

Data Science - **13 tasks**
Software Engineering - **13 tasks**
Web Development - **13 tasks**

Due Date: 28 April 2024

CoGrammar

Skills Bootcamp Progression Overview

✅ Criterion 3: Course Progress

Completion: All mandatory tasks, including Build Your Brand and resubmissions by study period end
Interview Invitation: Within 4 weeks post-course
Guided Learning Hours: Minimum of 112 hours by support end date (10.5 hours average, each week)

✅ Criterion 4: Demonstrating Employability

Final Job or Apprenticeship Outcome: Document within 12 weeks post-graduation
Relevance: Progression to employment or related opportunity

CoGrammar

Learning Objectives

- ❖ Implement state management using Context API in a React application.
- ❖ Secure a web application using JWT for authentication.
- ❖ Deploy and manage a React application using Vercel.

CoGrammar

Introduction to Building a Full Stack Blog Application

- The purpose of the blog app is to Create a platform for users to write, post, and manage articles.
- Features:
  - User registration and login.
  - Create, read, update, and delete (CRUD) blog posts.
  - Comment system (planned feature).

CoGrammar

Understanding State Management

- State management is the process of handling and updating data within a React application.
- It allows components to maintain their internal state and respond to user interactions effectively.
- It centralizes the state in large applications for easier data management and UI consistency

CoGrammar

What is State in React?

- ❖ In React, state refers to an object that represents the current condition of a component.
- ❖ Stateful components have the ability to hold and modify their state, which affects their rendering and behavior.

CoGrammar

How Does State Work?

- ❖ When a component's state changes, React automatically re-renders the component to reflect the updated state.
- ❖ Changes to state trigger a re-render of the component and its child components, ensuring that the UI stays in sync with the underlying data.

CoGrammar

Why Context API?

- Provides a way to pass data through the component tree without having to pass props down manually at every level.
- Use cases:
  - Managing user authentication state globally.
  - Sharing theme settings or user preferences across the application.

CoGrammar

JWT Authentication Overview

- What is JWT?
  - JSON Web Tokens (JWT) are a compact, URL-safe means of representing claims to be transferred between two parties.
- Benefits:
  - Facilitates secure data transfer.
  - Efficient for client-side storage and server-side verification.

Backend Setup

- Using Node.js and Express:
  - Set up a basic server with Express.
  - Create endpoints for user authentication and blog post management.
- Essential Middleware:
  - Use jsonwebtoken for creating and verifying tokens.
  - Use bcryptjs for password hashing.

User Registration Flow

- Flowchart:
  - User submits registration form → Validate input → Hash password → Store in database → Generate JWT → Return JWT.

User Login Flow

- Flowchart:
  - User submits login form → Validate input → Check email → Verify password → Generate JWT → Return JWT.

CoGrammar

Let's Breathe!

Let's take a small break before moving on to the next topic.

CoGrammar

Frontend Setup

- ❖ Creating the Project:
  - ➢ Initialize a new React project using create-react-app.
- ❖ Key Libraries:
  - ➢ Install and configure axios for API communication.
  - ➢ Setup routing using react-router-dom.

Integrating Context API

- ❖ Creating AuthContext:
  - ➢ Define AuthContext for global state management of user authentication.
- ❖ Usage:
  - ➢ Wrap the application root with AuthContext provider in index.js.

CoGrammar

Handling Authentication in Frontend

- Using Context:
  - Access authentication state using useContext hook in components.
- Examples:
  - Show or hide components based on authentication state.

Building the Blog Functionality

- CRUD Operations:
  - Implement forms and views for creating, reading, updating, and deleting blog posts.
- Security Considerations:
  - Secure routes using JWT to ensure only authenticated users can post, edit, or delete.

Deployment with Vercel

- ❖ Connecting to GitHub:
  - ➢ Push the code to a GitHub repository.
  - ➢ Link the repository to Vercel for deployment.
- ❖ Vercel Settings:
  - ➢ Configure environment variables such as API secrets in Vercel dashboard.

CoGrammar

Testing and Debugging Tips

- Common Issues:
  - CORS errors, JWT expiration handling, and route protection flaws.
- Debugging Tools:
  - Use browser developer tools, Postman for API testing, and React Developer Tools.

CoGrammar

Enhancing the Blog Application

- ❖ Adding Features:
  - ➢ Integrate a comment system for each blog post.
  - ➢ Implement like/dislike functionality.
- ❖ Scaling Considerations:
  - ➢ Optimize performance, consider serverless functions for backend.

CoGrammar

Summary

- **Key Takeaways:**
  - Utilization of React for building the frontend and Context API for managing global state such as user authentication across the application.
  - Implementation of secure user authentication using JSON Web Tokens (JWT), which includes user registration, login, and maintaining sessions securely.
  - Development of backend API routes using Node.js and Express to handle Create, Read, Update, and Delete (CRUD) operations for blog posts, with MongoDB as the database.
  - App Deployment process using Vercel, which includes setting up continuous deployment from a GitHub repository for seamless updates to the live application.

CoGrammar

Questions and Answers

CoGrammar

Thank you for attending

SKILLS FOR LIFE

SKILLS BOOTCAMPS

Department for Education

CoGrammar