

Build a Blog Website and Deploy it on Amazon EKS with MySQL

Objective:

You are tasked with developing a blog website using Node.js and deploying it on Amazon Elastic Kubernetes Service (EKS). The website should allow users to create, read, update, and delete blog posts, with data stored in a MySQL database.

Requirements:

Design and implement a blog website using Node.js. The website should have the following features:

- User registration and authentication.

- User roles (e.g., admin, author, reader) with different permissions.

- CRUD operations for blog posts (Create, Read, Update, Delete).

- User comments on blog posts.

- Display of blog post categories and tags.

- Search functionality for blog posts.

- Responsive design for mobile and desktop devices.

- Use a Node.js web framework of your choice (e.g., Express.js, Koa.js, Hapi.js) to build the website.

- Set up an Amazon RDS instance running MySQL to serve as the database for the blog website.

- Containerize the application using Docker.

- Set up an Amazon EKS cluster and configure the necessary networking components.

- Deploy the application on the EKS cluster using Kubernetes manifests (deployment, service, ingress, etc.).

- Connect the application to the MySQL database for storing and retrieving blog data.

- Implement proper error handling and validation for user inputs.

- Write unit tests and/or integration tests for critical parts of the application.

- Document the steps required to set up and run the application locally for development and testing purposes.

- Provide documentation on how to deploy the application on Amazon EKS, including any necessary Kubernetes configuration files or scripts.

Bonus Points (Optional):

- Implement user profile pages and allow users to customize their profiles.

- Add pagination and sorting options for the blog post listing page.

- Implement social sharing functionality for blog posts.

- Use a front-end framework/library like React or Vue.js to build the user interface.

- Configure logging and monitoring for the deployed application.

- Implement CI/CD pipelines to automate the build and deployment process.

Submission:

Please provide a link to a GitHub repository containing your project code, including the Docker configuration, Kubernetes deployment files, tests, and any other relevant documentation.

Note:

Feel free to use any additional libraries, frameworks, or services that you find suitable for the task. However, make sure to explain your design choices and document any dependencies or setup requirements.

Good luck with the assignment! If you have any questions, feel free to ask.