

# BLUBBER DOCUMENTATION

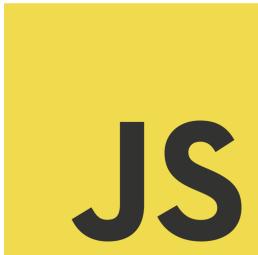
## OBJECTIVE

Design and implement a web-based blogging platform where users can create, publish, and manage their blog posts. The platform should allow users to register, log in, create blog posts, view and edit their posts, and display the published posts to the public.

Submitted By - **Chandra Mani Pandey**  
[chandra.pandey@mca.christuniversity.in](mailto:chandra.pandey@mca.christuniversity.in)  
**7023754529**  
2247256  
5 MCA - B  
**Full - Stack Developer Assignment**

<b>BLUBBER DOCUMENTATION</b>	<b>1</b>
1. Tech Stack Used - NodeJS, ReactJS, Express, MySql	1
2. Implementation	6
A. Database	6
B. Client	7
C. Server	8
D. Version Control	9
How to run the Source Code	9
What I had to work on still	9
New features that can be added	10
Screenshots	10

## 1. Tech Stack Used - NodeJS, ReactJS, Express, MySql



JavaScript is a popular choice for full-stack web development due to its versatility and widespread adoption. Its client-side capabilities enable dynamic and interactive user interfaces, while advancements like Node.js allow for server-side development in the same language. The language's asynchronous features contribute to efficient handling of tasks, and the rich ecosystem of libraries and frameworks streamlines development. Being part of a large and active community ensures ample support and resources. For someone aspiring to be a Full Stack Web Developer like you, choosing JavaScript aligns well with the demands of the industry and provides a comprehensive skill set for end-to-end web application development.



Choosing Node.js for a full-stack blogging website offers several advantages. Node.js is a server-side JavaScript runtime built on the V8 JavaScript engine, and its non-blocking, event-driven architecture makes it particularly well-suited for real-time applications like blogs.

### The advantages of using Node.js include:

**Single Language for Frontend and Backend:** Node.js allows you to use JavaScript on both the server and client sides, promoting code consistency and reusability. This can streamline development and make it easier to manage the entire application.

**Scalability:** Node.js is known for its ability to handle a large number of concurrent connections with high throughput. This makes it suitable for applications with potential scalability requirements, ensuring that your blogging website can handle increasing user traffic efficiently.

**Vast Ecosystem:** Node.js has a rich ecosystem of modules and packages available through npm (Node Package Manager). This allows you to leverage a wide range of

tools and libraries to enhance the functionality of your blogging website without starting from scratch.

**Asynchronous I/O:** Node.js is designed to be non-blocking and asynchronous, making it well-suited for applications that involve many I/O operations, such as reading and writing to a database. This can lead to improved performance and responsiveness.

**However, it's essential to consider potential disadvantages:**

**Callback Hell:** The asynchronous nature of Node.js can sometimes lead to callback hell, where deeply nested callbacks can make the code harder to read and maintain. This issue can be mitigated by using features like Promises or `async/await`.

**Learning Curve:** If you're new to asynchronous programming or server-side development, there might be a learning curve when working with Node.js. However, given your aspirations as a software engineer and your interest in computer science, this could also be seen as an opportunity for skill development.

**Less Suitable for CPU-Intensive Tasks:** Node.js may not be the best choice for CPU-intensive tasks due to its single-threaded nature. If your blogging website involves heavy computations, you might need to offload such tasks to other technologies or services.



Opting for Express.js, a web application framework for Node.js, brings several benefits when developing a full-stack blogging website:

**Minimalistic and Flexible:** Express.js is known for its minimalistic design and flexibility, providing a lightweight framework that allows developers to choose their preferred libraries and tools. This can be advantageous when tailoring your blogging website to specific requirements.

**Robust Routing:** Express.js offers a robust routing system that simplifies the handling of HTTP requests. This makes it particularly well-suited for structuring the backend of a blogging website, handling various routes for creating, updating, and displaying blog posts.

**Middleware Support:** Express.js emphasizes middleware, allowing developers to integrate additional functionality seamlessly. This is valuable for tasks like authentication, error handling, and request processing, enhancing the overall functionality of your blogging platform.

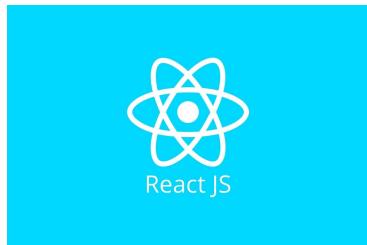
**Large Community and Ecosystem:** Similar to Node.js, Express.js has a large and active community, ensuring access to a wealth of resources, tutorials, and third-party middleware via npm.

**Alternative frameworks** for Node.js include:

**Koa.js:** A more modern and lightweight alternative to Express.js, Koa emphasizes middleware composition and is designed to be more expressive and robust.

**Nest.js:** A framework for building scalable and maintainable server-side applications, Nest.js is built on TypeScript and offers a modular and structured approach to web development.

**Sails.js:** Aimed at making it easy to build custom, enterprise-grade Node.js apps, Sails.js provides a full-featured MVC framework, making it suitable for larger and more complex projects.



Choosing React.js for the frontend of your full-stack blogging website comes with several advantages:

**Component-Based Architecture:** React.js follows a component-based architecture, allowing you to build reusable and modular UI components. This promotes code organization and makes it easier to manage and scale your frontend codebase.

**Virtual DOM:** React utilizes a virtual DOM, which optimizes the rendering process by updating only the parts of the actual DOM that have changed. This results in improved performance and a smoother user experience, especially crucial for dynamic content in a blogging website.

**Declarative Syntax:** React uses a declarative syntax, making it more straightforward to understand and write UI components. This can enhance productivity and readability, especially for developers new to the framework.

**Large Ecosystem and Community:** React has a vast ecosystem and a thriving community. This means access to a wide range of libraries, tools, and community support, facilitating development and problem-solving.

### **Comparing React.js with other well-established frontend frameworks:**

**Angular:** Angular is a more opinionated and feature-rich framework compared to React. While it comes with a steeper learning curve, it provides a more comprehensive solution out of the box, including a built-in router and a powerful CLI. React, on the other hand, offers more flexibility and is often favored for its simplicity and ease of integration.

**Vue.js:** Vue.js strikes a balance between the flexibility of React and the opinionated nature of Angular. It has a more gradual learning curve and is known for its simplicity. React's virtual DOM and large community might be seen as advantages, but Vue.js is praised for its ease of integration and simplicity.



**Relational Database Management System (RDBMS):** MySQL is a widely used open-source RDBMS, providing a structured and organized way to store and retrieve data. Its relational nature allows for efficient management of complex relationships within your blogging website's data.

**Scalability:** MySQL is known for its scalability, making it suitable for projects that may experience growth in terms of data volume and user interactions. It can handle large databases and high traffic, ensuring my blogging website remains responsive as it grows.

**Community Support:** MySQL has a robust and active community, resulting in extensive documentation, forums, and online resources. This community support is valuable for troubleshooting, learning, and staying updated on best practices.

**ACID Compliance:** MySQL adheres to ACID (Atomicity, Consistency, Isolation, Durability) properties, ensuring data integrity and reliability. This is crucial for applications like a blogging website where accurate data representation is paramount.

**NoSQL Alternatives:** Depending on your specific requirements, a NoSQL database like MongoDB might be considered. NoSQL databases offer more flexibility in handling unstructured data, which can be beneficial for certain types of applications. However, this flexibility comes at the cost of sacrificing some of the structured querying capabilities inherent in MySQL.

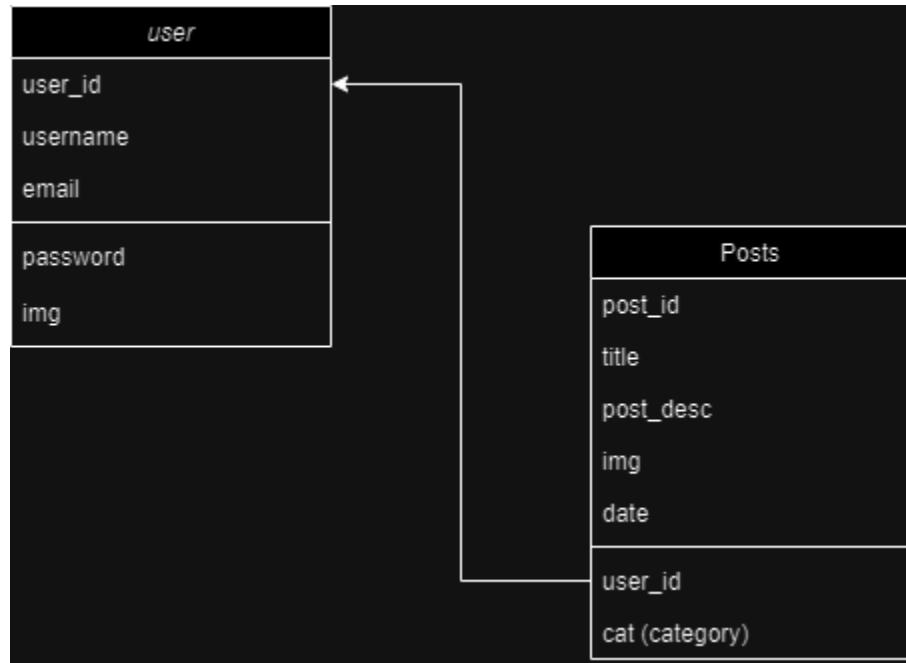
**Comparing MySQL with other well-established database systems:**

**PostgreSQL:** Similar to MySQL, PostgreSQL is an open-source RDBMS. It is known for its advanced features, extensibility, and support for complex data types. MySQL is often favored for its ease of use, while PostgreSQL may be chosen for projects with more demanding requirements.

**MongoDB:** MongoDB is a popular NoSQL database that stores data in flexible, JSON-like documents. It is schema-less, providing more flexibility in handling diverse data types. MySQL, being a relational database, offers a structured approach suitable for applications with well-defined data relationships.

## 2. Implementation

### A. Database



A simple implementation of the relational database model with only two tables,

1. Users - user table is used to store the user related information like user\_id, username, name, etc.
2. Posts - posts table is used to store the detailed information about a post created by a user, the user is connected to the post by a foreign key with the users table to recognize which posts belong to what users.

### B. Client

#### a. Dependencies -

```
"dependencies": {  
    "@testing-library/jest-dom": "^5.16.4",  
    "@testing-library/react": "^13.1.1",
```

```
"@testing-library/user-event": "^13.5.0",
"axios": "^1.3.4",
"moment": "^2.29.4",
"react": "^18.0.0",
"react-dom": "^18.0.0",
"react-icons": "^4.12.0",
"react-quill": "^2.0.0",
"react-router-dom": "^6.9.0",
"react-scripts": "^5.0.1",
"sass": "^1.60.0",
"web-vitals": "^2.1.4"
```

b. Configuration -

```
"scripts": {
  "start": "react-scripts start",
  "build": "react-scripts build",
  "test": "react-scripts test",
  "eject": "react-scripts eject"
},
```

c. Core Features -

Core features in this blogging website's client are -

- I. Login and Registration functionality.
- II. JWT based Authentication
- III. Home page displaying all the posts present on the website from all the users
- IV. Category based post filtering
- V. A single page for displaying everything about a single post selected by the user.
- VI. Implementation of Write functionality, for users to be able to write a new post.
- VII. A simple sidebar menu when a single post is opened, displaying the posts from the same category as the selected post.

## C. Server

a. Dependencies -

```
"dependencies": {
  "bcryptjs": "^2.4.3",
```

```
"body-parser": "^1.20.2",
"cookie-parser": "^1.4.6",
"cors": "^2.8.5",
"express": "^4.18.2",
"jsonwebtoken": "^9.0.0",
"multer": "^1.4.5-lts.1",
"mysql": "^2.18.1",
"nodemon": "^2.0.22"
```

b. Configurations -

```
"scripts": {
  "test": "echo \\\"Error: no test specified\\\" && exit 1",
  "start": "nodemon index.js"
}
```

c. Core Features -

- I. Creating database models for the data to be stored.
- II. Configuring the database connection
- III. Configuring CORS to access data from the client side
- IV. Configuring Multer to store the images from the client side
- V. Configuration of Authentication routes for Registering a new user, logging a user, and logging out a user from the Website.
- VI. Creating the routes for GET, POST, PUT, and DELETE queries for the data storage, access, updating, and deleting.

## D. Version Control

Using Git/Github for Version control and hosting of source code.

## How to run the Source Code

- Clone the git repository, [https://github.com/Chandra-MP/blog\\_website.git](https://github.com/Chandra-MP/blog_website.git)
- Open the folder, then open 2 separate terminals and navigate to client and server folders respectively.
- Run, `npm start` on both the terminals to start the development build of the project.

## What I had to work on still

1. In the source code -
  - Client side validation
  - More robust UI
  - Ability to select a article in multiple categories
  - Able to save the post as a draft and pickup from where it was left
  - A character based search for the articles
  - Better way to store the uploaded images on the server
  - Proper security check and implementing more secure ways
  - Ability for the users to like/dislike the posts
  - Real time updates to the users when a post is made on their account
  - Ordering of the posts on certain criterias
2. Hosting
  - Hosting the website on cloud servers like AWS or GCP

## New features that can be added

1. Ability for the users of the website to comment.
2. A more robust and custom embedded text editor
3. Ability for the users to customize their feed according to their preferences.

## Screenshots

localhost:3000/post/55

The screenshot shows a web page from the BLUBBER website. At the top, there is a navigation bar with the BLUBBER logo, followed by categories: ART, SCIENCE, FOOD, DESIGN, TECHNOLOGY, a 'Write' button, and a 'Login' link. Below the navigation, there is a large image of a painting titled 'Portrait of Countess Livia da Porto Thiene and her Daughter Deidamia' by Veronese. The painting depicts a woman in a red dress and a young girl in a green dress. To the left of the painting, there is a title and a short description of the painting. A 'Read More' button is visible. The URL 'localhost:3000/post/55' is shown at the bottom left.

**Portrait of Countess Livia da Porto Thiene and her Daughter Deidamia**

Veronese's majestic, full-length portrait of the countess and her eldest daughter Deidamia, born in 1545, was originally accompanied by one of her husband Count Issepo (Giuseppe) da Porto and their son Leonida (now in the Palazzo Pitti, Florence).

[Read More](#)

The screenshot shows the BLUBBER login page. The page has a teal header with the BLUBBER logo and the tagline 'BLABBER ALL YOU WANT'. Below the header, there is a 'Login' button. A form box contains fields for 'username' and 'password', and a 'Login' button. Below the form, there is a link 'Dont have an account? [Register](#)'.

BLUBBER

BLABBER ALL YOU WANT

Login

username

password

Login

Dont have an account? [Register](#)

# BLUBBER

BLABBER ALL YOU WANT

## Register

username

email

password

confirm-password

[Register](#)

Already have an account? [Login](#)

BLUBBER  
BLABBER ALL YOU WANT

ART SCIENCE FOOD DESIGN TECHNOLOGY Write chandra Logout

## Portrait of Countess Livia da Porto Thiene and her Daughter Deidamia

Veronese's majestic, full-length portrait of the countess and her eldest daughter Deidamia, born in 1545, was originally accompanied by one of her husband Count Issepo (Giuseppe) da Porto and their son Leonida (now in the Palazzo Pitti, Florence).

[Read More](#)





surya  
Posted 5 hours ago

## 4 Effective Ways to Manage Food Cravings

When those dreaded snack cravings come knocking, it can be hard to stay on track with healthy eating. Factors such as stress and boredom lead us to crave unhealthy foods that are high in sugar or salt content. Good thing there's an easy way out! One of the best strategies is drinking plenty of water — thirst often tricks us into thinking we're hungry but a glass of H2O helps reduce these feelings before they get too overwhelming.

### Other posts you may like



**4 Effective Ways to Manage Food Cravings**

[Read More](#)



**The use of bitter flavor in your diet and food and its**

Title

---

Normal B I U H E Tx

**Publish**

Status: Draft

Visibility: Public

Upload Image

[Save as a Draft](#) [Publish](#)

**Category**

Art  
 Science  
 Technology  
 Food  
 Design