My web application is combination of Ruby on Rails for the backend, React for the frontend, Redux Toolkit for state management and some other additional libraries.

**Key Features**

* **User Authentication**: Secure user registration and login processes with encrypted password handling.
* **Blog Management**: Create, edit, and delete blogs with ease. Add rich text content using Draft.js.
* **Categorization**: Organize blogs into categories for easy navigation.
* **Likes and Comments**: Engage with the community by liking and commenting on blogs.
* **Admin Panel**: Exclusive admin privileges for content moderation.

**1]Authentication and Authorisation:**

**User Authentication**: Secure user registration and login processes with encrypted password handling. JWT (JSON Web Token) authentication ensures data security.

**My Blogging Platform**  incorporates user authorization to ensure that users can interact with the application securely and efficiently. The authorization system in this application is designed to provide the following functionalities:

**1. User-Specific Operations**

Users are granted specific privileges over the blog posts they create:

* **Edit**: Users can edit the content and details of their own blog posts.
* **Update**: Users can update the content of their blog posts.
* **Delete**: Users can delete their own blog posts if they choose to remove them from the platform.

**2. Viewing Other Users' Blog Posts**

While users can manage their own posts, they have view-only access to blog posts created by other users. This means they can read, like, and comment on posts from fellow bloggers.

**Implementation**

The authorization system relies on a combination of backend and frontend logic. Here's an overview of how it works:

* **Backend**: The Ruby on Rails backend uses authentication tokens (JWT) to identify users and determine their roles and permissions. When a user attempts to edit, update, or delete a blog post, the backend checks if the user making the request is the author of that post. If they are, the request is authorized; otherwise, it's denied.
* **Frontend**: The React frontend adapts to the user's role. Users see "Edit" and "Delete" buttons on their blog posts, which are handled by the frontend components.

These buttons are hidden or disabled for blog posts authored by others, ensuring that users can only interact with their content.

**2]Blog Categories**

My Blogging Platform offers a convenient way for users to explore and discover blogs by category. Whether you're interested in technology, spirituality, sports, or personal stories, there's something for everyone. Here's how you can access blogs based on different categories:

**1. Personal Blogs**

* **Description**: Personal blogs are a place for users to express themselves, share their thoughts, experiences, and stories with a personal touch.
* **Access**: Users can access their own personal blogs, which they've created themselves.

**2. Sports Blogs**

* **Description**: Sports enthusiasts can dive into a world of sports-related blogs. Stay updated with the latest scores, game analysis, and athlete stories.
* **Access**: Users can view all sports-related blogs available on the platform.

**3. Spiritual Blogs**

* **Description**: Find inner peace and spiritual guidance through a collection of blogs focused on spirituality, mindfulness, and personal growth.
* **Access**: Users can explore a variety of spiritual blogs authored by fellow users.

**4. Technology Blogs**

* **Description**: Stay informed about the ever-evolving world of technology. Technology blogs cover the latest trends, gadgets, and tech innovations.
* **Access**: Users can read technology-related blogs contributed by tech enthusiasts.

**Implementation**

Accessing blogs by category is straightforward:

* **Personal Blogs**: Users automatically have access to their personal blogs upon login. These blogs are listed in the "Personal" category.
* **Sports, Spiritual, and Technology Blogs**: Users can navigate to the respective categories . Here, they can browse and read blogs from a wide range of topics.

**3]Interacting with Blogs**

My Blogging Platform allows users to engage with blogs by adding comments and expressing their appreciation through likes. Here's how you can actively participate in the community:

**1. Adding Comments**

Share your thoughts, insights, or questions by adding comments to blogs.

**2. Liking Blogs**

Show your appreciation for a well-written blog by liking it.

**4]Pagination**

My Blogging Platform offers a seamless browsing experience, allowing you to explore a vast collection of blogs. The pagination feature ensures you can find the content you're looking for easily. Pagination divides blogs into manageable pages, making it effortless to browse through numerous articles.

**5]Search bar**

My Blogging Platform empowers you to find the content you're looking for effortlessly with its advanced search functionality. Here's how you can make the most of it .The search bar enables you to find blogs matching specific keywords or topics.

**6]Admin Privileges**

My Blogging Platform offers special privileges to administrators, allowing them to manage user accounts and content effectively. Here's an overview of what administrators can do:

**1. User Management**

Administrators have the authority to manage user accounts, including promoting users to admin status and deactivating accounts.

**2. Blog Management**

Administrators can oversee all blogs on the platform, enabling them to delete, edit, or update any blog content.

**3. Content Moderation**

Admins play a vital role in ensuring the quality and integrity of content on the platform. They can moderate comments, ensuring that discussions remain respectful and on-topic.

**4. Role Assignment**

Administrators have the authority to change a user's role, promoting them to admin or reverting them to a regular user as needed.

Attached some ss below

A login screen with white and orange text

Description automatically generated

A screenshot of a login form

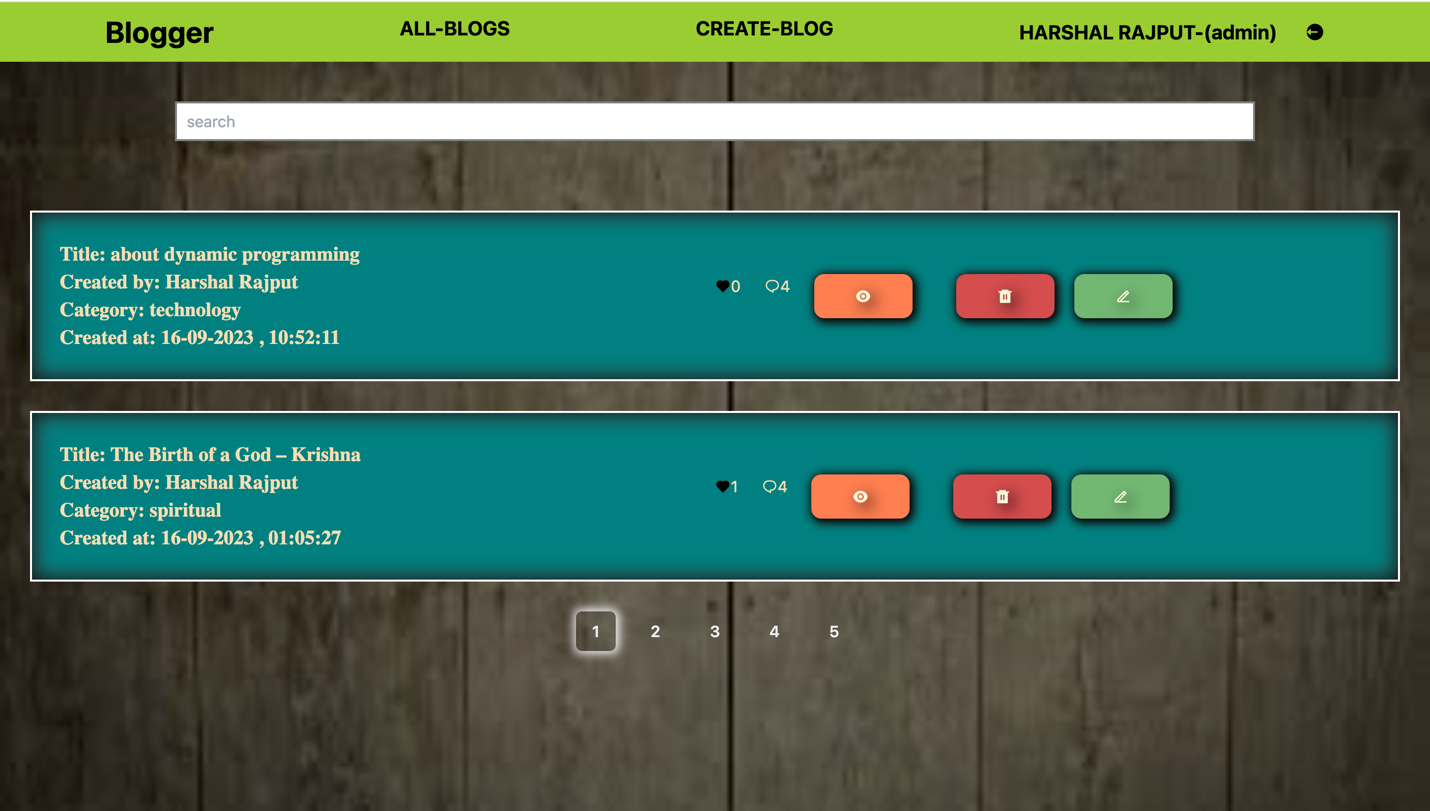
Description automatically generated

A screenshot of a website

Description automatically generated

A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated