Project Requirement Document

Project Name: **Techforce Buddy** (Your AI powered chatbot)

Technology Stack: Java 21, Spring Boot, Hibernate, MySQL, OpenNLP/Deeplearning4j, Thymeleaf, JavaScript, JWT Authentication

1. Project Structure

- **Frontend**: Separate project using **Thymeleaf** for rendering UI elements. It will communicate with the backend using **JavaScript** for sending HTTP requests.
- **Backend**: Separate project using **Spring Boot** for managing the core business logic, including authentication, processing chatbot queries, and handling user data.
- **Communication**: The frontend will send requests to the backend using JavaScript (AJAX or Fetch API). The backend will respond with **JSON** data.

2. Login Page (Frontend)

Description:

The login page will allow users to enter their **username** and **password** for authentication. JWT (JSON Web Token) will be used to secure the login.

UI Elements:

- Username input field
- Password input field
- Login button
- Register button

Functionality:

- JavaScript will send a POST request to the backend's authentication API.
- Backend will authenticate the user and return a JWT token.
- Store the JWT token in local storage or cookies for session management.

- Unauthorized requests will be rejected with an appropriate message.

3. Registration Page (Frontend)

Description:

This page will allow new users to create an account by entering their details.

UI Elements:

- First Name (Text Field)
- Last Name (Text Field)
- Email (Text Field)
- Password (Password Field)
- Register Button

Functionality:

- Form data will be validated using JavaScript.
- A POST request will be sent to the backend for registration.
- Backend will store user details and return a confirmation message.

4. Chat Interface (Post-Login) - Frontend

Description:

Once the user is logged in, they will be presented with the **Techforce Buddy** chatbot interface, where they can enter queries and receive responses.

UI Elements:

- Search bar (Input field for user queries)
- Response area (Textarea for displaying chatbot responses)

Functionality:

- The user will type their query in the search bar and click the **Submit** button.
- **JavaScript** will send the query to the backend using an HTTP POST request.

- The backend will analyze the query using policy documents stored in the system.
- Using AI libraries (OpenNLP/Deeplearning4j), the backend will process the policy content and generate an appropriate response.
- The response will be returned to the frontend in JSON format and displayed in the textarea below the search bar.

5. Backend

- **Authentication**:
  - REST API for login and registration.
  - Use **JWT authentication** to secure all endpoints.
  - Validate incoming requests with JWT token and handle authorization.
- **Chatbot Functionality**:
  - REST Controller to handle chatbot requests.
  - Accept requests in JSON format containing the user's query.
  - Analyze the content of policy documents based on the query.
  - Use **OpenNLP/Deeplearning4j** to prepare a response based on the policy content.
  - Return the response as JSON to the frontend.

6. Technology Stack

- **Frontend**:
  - **Thymeleaf**: For rendering HTML views and dynamic content.
  - **JavaScript**: For handling user interactions and making HTTP requests.

- **Backend**:
  - **Java 21**: Backend logic and API development.
  - **Spring Boot**: To build RESTful services.
  - **Hibernate**: ORM to interact with the MySQL database.
  - **MySQL**: Database for storing user credentials and policy documents.
  - **OpenNLP / Deeplearning4j**: AI component for processing user queries and policy content.
  - **JWT Authentication**: To secure user login and manage sessions.