

## Intern Developer Interview Task:

**Part A: Project Overview:** You are tasked with developing a robust API and design for an Employee Management System. The system should allow CRUD (create, read, update, and delete) operations on employee records, role assignment, searching, and status updates. Pay attention to the structure of your code, adherence to best practices, and the clarity of your documentation.

### Task Specifications:

#### 1. Employee CRUD Operations:

- Create endpoints to perform CRUD operations on employee records.
- Ensure a well-structured payload for creating and updating employees.

#### 2. Role Assignment:

- Implement an endpoint to allow admins to assign roles to employees.
- Valid roles: [manager, developer, design, scrum master].

#### 3. Search Functionality:

- Create endpoints to find employees by name and ID.

#### 4. Admin Dashboard:

- Provide an admin dashboard with endpoints to:
  - Retrieve total employees.
  - Retrieve total available roles.
  - Create and delete job roles.

#### 5. Status Updates:

- Implement an endpoint allowing admins to update the status of an employee (employed or fired).

#### 6. Data Consistency:

- When fetching employee details, ensure that the job role is populated along with other employee information.

### Things to Evaluate:

#### 1. Data Handling:

- Assess how data is returned from each endpoint.
- Evaluate the structure and organization of the payload.

#### 2. Success Handling:

- Evaluate how success scenarios are handled, providing detailed responses.
- Ensure proper HTTP status codes are returned upon success.

#### 3. Error Handling:

- Examine the approach to handling errors gracefully.
- Include meaningful error messages and appropriate HTTP status codes.

#### 4. Documentation:

- Document each endpoint comprehensively.
- Use any API client of your choice for documentation.

#### 5. Response and Status Codes:

- Ensure accurate usage of HTTP status codes.
- Clearly define response structures for different scenarios.

#### 6. Database and HTTP Methods:

- Use any SQL database for storage.
- Implement appropriate HTTP methods for each endpoint (GET, POST, DELETE, etc.).

**Additional Information:** Please include a brief explanation of your technical choices. You are free to use any programming language and framework of your choice.

**Part B:** Evaluate the Rulerise website as if you were a customer. Identify 2-3 areas where you would make changes to improve the user experience.

**NB:** Kindly submit your output via email to [careers@rulerise.com](mailto:careers@rulerise.com) within 72 hours of receiving this email. Your interview/presentation will be scheduled within 48 hours of receiving your submission.

Best of luck with the task! If you have any questions or need clarification, feel free to ask.