

---

# System Requirements Specification Index

For

## Job Search Application

Version 1.0

IIHT Pvt. Ltd.  
fullstack@iiht.com

## Contents

1	Business-Requirement:	3
1.1	Problem Statement:	3
1.1.1	Job Search Application	3
2.	Template Code Structure	
1.1	Job Controller	
3.	Resources AVAILABLE:	4
4	Suggested WIREFRAMES:	5
5	Business Validations	9
6	Considerations	9
7	Execution Steps to Follow	9

# 1 BUSINESS-REQUIREMENT:

---

## 1.1 PROBLEM STATEMENT:

The purpose of this application is to allow an admin to list out all jobs with different details for job seekers.

### 1.1.1 Job Search Application:

The Job Search Application allows you to:

1. Access home page
2. Should be able to add a new job.
3. It can have basic fields like Title, Description, Experience & Salary.
4. Should be able to get the list of jobs.

## 2. TEMPLATE CODE STRUCTURE:

### 2.1 JOB CONTROLLER

---

Method Exposed	Purpose
<code>listJobs()</code>	Should return page "list-jobs" with required data.
<code>showFormForAdd()</code>	Should return page "job-add" for adding a Job.
<code>saveJob()</code>	Should save a Job and return "job/list" with required data.
<code>showFormForUpdate()</code>	Should show job details in the page "job-add" to edit a job.
<code>showFormForDelete()</code>	Should delete a job and return "job/list" with required data.

## 3. RESOURCES AVAILABLE:

---

Description	View Pages Name	Remarks
Common UI		
Home Page	list-jobs	Contains a homepage which shows a list of all jobs along with options to add, edit and delete a job.
All Jobs	list-jobs	
Add a job	job-add	
Update a job	job-add	

### 3 SUGGESTED WIREFRAMES:

---

#### 1. Homepage – Visitor Landing Page



← ↻ ⓘ localhost:8081/job/list

## Job Management System

Post Job

Job Title	Job Description	Minimum Experience	Salary	Action
-----------	-----------------	--------------------	--------	--------

## 2. Create a Job



The screenshot shows a web browser window with the address bar displaying 'localhost:8081/job/showFormForAdd'. The page title is 'Job Management System'. Below the title, the section 'Post Job' is visible. The form contains four input fields: 'Job Title:', 'Job Description:', 'Minimum Experience:', and 'Salary(in INR):'. A 'Save' button is located below the salary field. At the bottom of the form, there is a blue hyperlink labeled 'Back to Job List Page'.

← ↻ ⓘ localhost:8081/job/showFormForAdd

### Job Management System

#### Post Job

Job Title:

Job Description:

Minimum Experience:

Salary(in INR):

[Back to Job List Page](#)

### 3. Edit a Job

← ↻ ⓘ localhost:8081/job/showFormForUpdate?jobId=2

## Job Management System

### Post Job

Job Title:

Job Description:

Minimum Experience:

Salary(in INR):

[Back to Job List Page](#)

## 4 BUSINESS VALIDATIONS

---

1. Id must be of type id.
2. JobTitle value not null, min 2 and max 40 characters.
3. Experience value not null
4. Job Description value not null and min 2 and max 40 characters.
5. Salary value not null.

## 5 CONSIDERATIONS

---

The Code template already contains skeleton methods for service and controller layer. Please write your logic in it.

## 6 EXECUTION STEPS TO FOLLOW

---

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu  
(Three horizontal lines at left top) → Terminal → New Terminal
3. To build your project use command:  
**mvn clean package -Dmaven.test.skip**
4. To launch your application:  
**java -jar springboot-job-service-0.0.1-SNAPSHOT.war**
5. This editor Auto Saves the code
6. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use **CTRL+Shift+B**-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
7. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.



8. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.
9. This is a web-based application, to run the application on a browser, use the internal browser in the workspace. Click on the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.

**Note: The application will not run in the local browser**

10. Default credentials for MySQL:
  - a. Username: **root**
  - b. Password: **pass@word1**
11. To login to mysql instance: Open new terminal and use following command:
  - a. **sudo systemctl enable mysql**
  - b. **sudo systemctl start mysql**
  - c. **mysql -u root -p**

The last command will ask for password which is '**pass@word1**'
12. Mandatory: Before final submission run the following command:  
**mvn test**
13. You need to use **CTRL+Shift+B** - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.