



## **Jobs for You**

SRS Document

<b>Jobs for You -Employment Portal</b>	<b>3</b>
<b>Introduction</b>	<b>3</b>
<b>Management Summary</b>	<b>3</b>
<b>Key Assumptions</b>	<b>3</b>
<b>High Level Architecture</b>	<b>3</b>
<b>Functional Requirements</b>	<b>4</b>
Use Case Diagrams	<b>5</b>
Use Cases	<b>6</b>
<i>Upload Masters</i>	<b>6</b>
<i>Register for Job Card</i>	<b>7</b>
<i>Employment Request</i>	<b>7</b>
<i>View Consumer Details</i>	<b>8</b>
<i>Post your complaint</i>	<b>8</b>
<i>View Employment Statistics</i>	<b>9</b>
<i>View Dashboard</i>	<b>9</b>
<b>Logical Object Model</b>	<b>10</b>
<b>Database Design Guidelines</b>	<b>11</b>
<b>Testing Approach</b>	<b>11</b>
<b>Suggested Technical Reading</b>	<b>12</b>

## Disclaimer

This Software Requirements Specification document is a guideline. The document details all the high level requirements. The document should be used as a guideline by the students to design the Solution Architecture for the project. The document also describes the broad scope of the project and high level logical object model. But while developing the solution if the developer has a valid point to add more details being within the scope specified then it can be accommodated after consultation.

# Jobs for You -Employment Portal

## Introduction

The purpose of this document is to define scope and requirements of recruitment of youth in rural india - Jobs for You that will deploy practices as underlined in NREG scheme of Government.

This document should be used by the development team to architect the solution the project.

## Management Summary

This information system for the benefit of under privileged is being developed to provide comprehensive and relevant information to consumers, managers, administrators and general public . The key objectives are

1. Facilitate management of the available resources and information for authorized personnel. The NREG has been notified in most backward districts of India.
2. Provide employment to those who demand employment within 15 days of their demand.
3. Effective monitoring using this system.
4. Eliminate manual information systems that are difficult to be maintained.

The proposed solution will be designed & developed to run on IBM WebSphere Application Server and IBM DB2 Universal Database in a 2-tier architecture.

## Key Assumptions

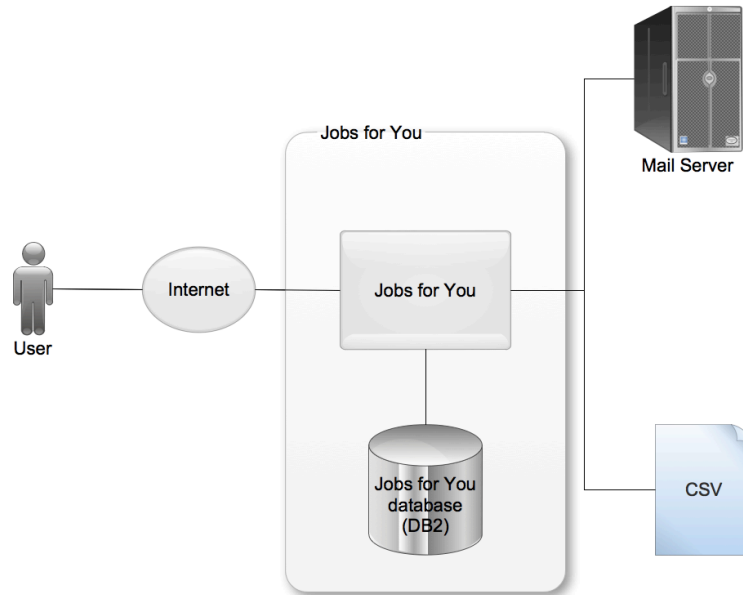
1. The National Rural Employment Guarantee Act is vast in terms of implementation and requires enormous amount of information capture and its dissemination using various methodologies. Only select features have been included in the scope of current project to meet the project timelines.
2. The project owner should read about the NREG act/scheme, browse information system created for rural employment.

## High Level Architecture

Job for You high level architecture is illustrated through the context diagram shown below. It will have following categories of users:

1. Members (Protege)

2. Mentors
3. Volunteers (Providers of Job for You website)
4. Administrator



*Job for You Context Diagram*

Job for You	Consumers can register, check wait time for employment and their employment history by the scheme and the money earned. Consumers can view employment opportunities by skills and capabilities. Information about poverty alleviation schemes and health, election registration etc., and other information like ration cards, ration shops, vocational education etc. They can also provide feedback of employers under NREGA
Job for You Database	Centralized Repository that need to be maintained to enable the system to operate efficiently to provide accuracy and swift response. Will have data for states, districts, blocks, Employers, Consumer, Employment history, Users of the system
CSV	Master data such as States, Districts, Blocks, Employers, Opportunities, are uploaded via CSV. Transactional data such as Job Cards, Employer money transfer data are also uploaded via CSV (assuming its being received from external systems not in the ambit of this system)
Mail Server	All notifications are routed through the Mail Server

## Functional Requirements

The high level functional requirements for the Job for You are outlined in the Use Case diagram described in this section.

Job for You will provide a secure user-id/password based secured login mechanism to access its services. The details of this are not outlined here. The development team is expected to create these keeping in mind the general practices followed by

the web applications. Login will be a prerequisite to use Job for You. Internal users will be provided user id/password pair separately.

On logging in user gets to view the options as per their role.

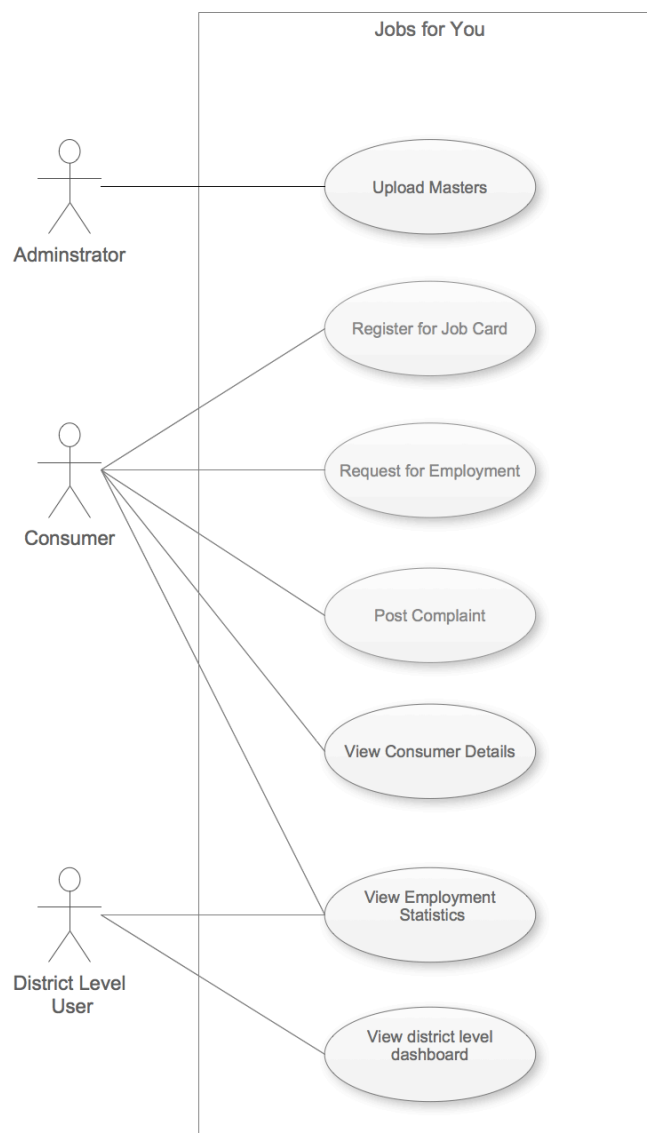
**Consumer** – Register, Request for employment, Check wait time for employment. Employment history by the scheme and the money earned. Other employment opportunities by skills and capabilities. Checking employment history of others that have the same name and address in the local area. Log in complaint.

**District level** – View registered users, their employment history, payments received, feedback. Checking funds disbursal in the local area. Aggregation of feedback under this scheme. Employment criteria met /not-met.

**Any external user** - Listing of funds usage in specific areas and announced projects. Employment statistics.

## Use Case Diagrams

The following figure illustrates the Use Case diagram for the system. The MIS use cases are not detailed here.



*Use Case Diagram*

## Use Cases

### Upload Masters

Use Case Element	Description
Number	UC.01
Application	<p>Masters in Job for You are uploaded using CSV Format</p> <p>State Master contains State id, State Name.</p> <p>District Master contains District Id, District Name, State id.</p> <p>Block Master contains Block id, Block Name, District id.</p> <p>User master will have user id, user name, type(Consumer / Internal), State id, District id, Block id.</p> <p>(user access will be governed by the location parameters, information related to their state and below hierarchy will be accessible all the users)</p> <p><b>By District level admin on as is basis</b></p> <p>Job card master will have Job card id, user id, valid from date, valid till date, government id reference, type of government id proof submitted(ration card, adhaar card)</p> <p>Employers master will have Employer id, Employer name, Business type(manufacturing, education, services oriented, other institutions)</p> <p>Opportunity master will have opportunity id, description, number of days, per day wages, status, state id, district id, block id.</p>
Use Case Name	Upload Masters
Primary Actor	Administrator
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the <b>Upload Masters</b> menu item on the landing page
Basic Flow	<p>System prompts for the file name to be uploaded. Standard file upload dialog is presented to select a file from the local system.</p> <ul style="list-style-type: none"> <li>The selected file data is uploaded in the related tables; if an existing record is encountered, the old details are replaced with the new details.</li> </ul>
Alternate Flow	<ul style="list-style-type: none"> <li>In event of incorrect CSV format, system gives an error and NO data is uploaded.</li> <li>Operation is cancelled</li> </ul>
Output	System displays the number of records uploaded. It also highlights the number of records updated (i.e. already existing ones being replaced)

**Register for Job Card**

Use Case Element	Description
Number	UC.02
Application	Consumer registers for the Job Card
Use Case Name	Register
Primary Actor	Consumer
Secondary Actor	None
Pre-condition	User has not registered earlier in the system.
Trigger	User clicks on the <b>Register</b> link on the landing page
Basic Flow	<ul style="list-style-type: none"> <li>The system displays a form to take the following inputs</li> <li>Name, House #, Village, District, mobile number, email id, user name(not more than 8 characters - Alpha numeric), Current employment status(Employed, Self-employed, Dependent), State, District, Block are selected via pickers populated based on the hierarchy of information being entered.</li> <li>Click on submit saves the request and moves it to district level user's queue for processing.</li> <li>(offline processing in this project, only approved job card ids and details are uploaded by the admin person at district level).</li> <li>An alert is displayed to the user to collect job card after 3 days from the district office counter.</li> </ul>
Alternate Flow	None
Output	Alert display

**Employment Request**

Use Case Element	Description
Number	UC.03
Application	Consumer after receiving job card number logs in using the job card id.
Use Case Name	Employment Request
Primary Actor	Members
Secondary Actor	None
Pre-condition	Authorized user is logged in to the system
Trigger	User clicks on the <b>Employment Request</b> link on the landing page



Use Case Element	Description
Basic Flow	<ul style="list-style-type: none"> <li>The system displays consumer's information like name, location, mobile number etc. if the member has logged in to the system. The system displays the list of employers and opportunities(Number of days, wages) available in the consumer's area.</li> <li>User selects the opportunity to pursue further and submits the request</li> <li>The system notifies the employer about the member's interest in the opportunity. This request is to be processed and job provided within 15 days from the date of application.</li> <li>A request id is generated which the user can keep referencing to on viewing the status</li> </ul>
Alternate Flow	None
Output	Notification to employer for response awaited.

### View Consumer Details

Use Case Element	Description
Number	UC.04
Application	District level users view consumer request details
Use Case Name	View Consumer Details
Primary Actor	District level user
Secondary Actor	Volunteer
Pre-condition	None
Trigger	The user clicks on the <b>Consumer Details</b> link on the landing page
Basic Flow	<ul style="list-style-type: none"> <li>The list of consumers awaiting job card issue are displayed. The user can select any one of them and view credentials and process the application.</li> <li>Job card number is generated. An email notification to the applicant is sent.</li> <li>The rejected applicant is also notified for furnishing additional details via email.</li> </ul>
Alternate Flow	None
Output	Email to consumer for checking the response.

### Post your complaint

Use Case Element	Description
Number	UC.05
Application	The consumers have to right to provide feedback to the managing authorities at district or above level via this system
Use Case Name	Post Complaint
Primary Actor	Consumer
Secondary Actor	None
Pre-condition	User is an authorized role in the system
Trigger	The user clicks on the <b>Post Complaint</b> link on the landing page.

Use Case Element	Description
Basic Flow	<p>The system displays list of Categories to select from.(Multi-select is possible) e.g. (No opportunities, payment not received, Conduct issues with employer, fund transfer to incorrect account etc.)</p> <p>A form is displayed to capture the complaint:</p> <p>Date, Details of complaint (Text box)</p> <p>The entries are saved and sent as complaint from the Job card holder.</p>
Alternate Flow	Pressing Cancel abandons operation, no database gets affected
Output	None

### View Employment Statistics

Use Case Element	Description
Number	UC.06
Application	Consumers and other users can view the performance in terms of employment data
Use Case Name	View employment statistics
Primary Actor	All users
Secondary Actor	None
Pre-condition	None
Trigger	The user clicks on the <b>View Employment Statistics</b> link on the landing page.
Basic Flow	<ul style="list-style-type: none"> <li>System displays a list of employment opportunities generated for each state, number of consumers applied and number of consumers got jobs.</li> <li>This data can be viewed month wise or for a period</li> </ul>
Alternate Flow	None
Output	None

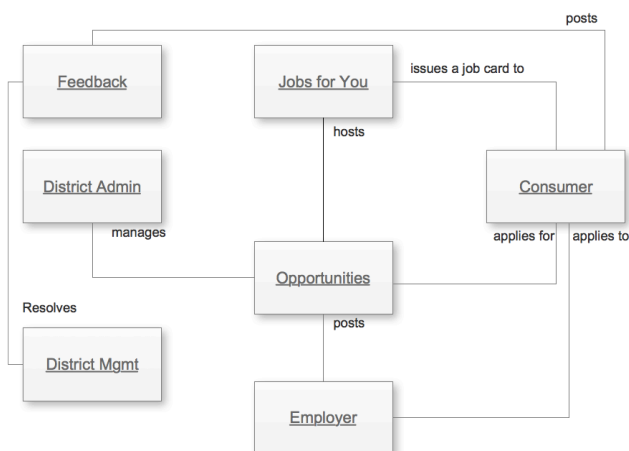
### View Dashboard

Use Case Element	Description
Number	UC.07
Application	District Level user get a complete snapshot of the scheme implementation in their area.
Use Case Name	View Dashboard
Primary Actor	District Level User
Secondary Actor	None
Pre-condition	None
Trigger	The user clicks on the <b>View Dashboard</b> link on the landing page.

Use Case Element	Description
Basic Flow	<ul style="list-style-type: none"> <li>• System displays the following sections of data in a snapshot view</li> <li>• Total job card enrollments, % of unemployed</li> <li>• Total opportunities, open, closed,</li> <li>• Employment requests not processed, rejected</li> <li>• Payment issues reported by the consumer as a complaint</li> <li>• Opportunity issues reported by the consumer as a complaint</li> <li>• User can click on any of the information and get to the actual data page view.</li> </ul>
Alternate Flow	None
Output	None

## Logical Object Model

A high level logical object model of the system is shown below. During technical design it will be transformed into a physical model covering all system entities. Such a diagram will include their relationship and its cardinality.



*Logical Object Model*

1. Jobs for you hosts content on Jobs for the rural youth.
2. Job opportunities are posted on the Jobs for you portal by employers in a area.
3. The consumer is able to access information related to their district.
4. A consumer registers on Job for You and gets a job card.
5. A job card holder requests for employment on the opportunities posted on Jobs for you.
6. Tracking of employment figures is done at district level.
7. The consumer posts complaint regarding job requests, employer and payments for district level managements action.

## Database Design Guidelines

This involves the transformation of the use cases, state diagrams, and logical object model into detailed and optimized physical database table designs.

Typically persistent classes will map to table(s) with their attributes as columns of the table. In some cases a high level object may map in to a master-child table. Invoice is one such example where it maps in to "invoice\_header" and "invoice\_line\_item" table.

Associations between two persistent objects are realized as foreign keys to the associated objects. A foreign key is a column in one table that contains the primary key value of the associated object.

Similarly, a standard technique in relational modeling is to use an intersection entity to represent many-to-many associations. Following is a broad checklist for physical database database design:

1. Database must be properly normalized except those instances where de-normalization help improves performance. This option must be used with special care.
2. All persistent classes that use the database for persistency must map to database structures.
3. Many-to-many relationships must have an intersecting table.
4. Primary keys should be defined for each table, unless there is a performance reason not to define a primary key.
5. Indexes should be defined to optimize access.
6. Data and referential integrity constraints should be defined.

## Testing Approach

Quality of the software can be achieved with basic hygiene and consistency followed during design and development of User Interface(UI), Navigation, Validations as per the business process requirement.

To ensure the project delivers acceptable quality to the customer, its important to create a checklist of the conventions to be followed across. Common checks as below are for your reference during design and development:

Common Checks	Validation Type
Page Title is valid for the feature being provided on the page	UI
Order of the Data Entry Fields is logical as per the functionality being provided by the feature	UI
Order of the Display only Fields makes viewing and understanding easy for the user	UI
Spellings and Correctness of Label for the Data Entry and Display fields	UI
The labels are not wrapping onto another row thereby adding a blank row on the page	UI
The fields with drop down are displayed in single row instead of drop down coming on the next row	UI
Data Entry field basic validations are working i.e Text field /Numbers / Dates allow data for their type only	Functional
The dates are following a standard format dd/mmm/yyyy on all forms	UI
The color scheme of all forms i.e headers labels , alerts, entry fields are uniform throughout the application	UI

Common Checks	Validation Type
The action buttons for a New Data Entry Form are uniform for all forms that is allowing data entry	UI
The action buttons are performing the desired action e.g. "submit" is creating a new record if there are no errors and recording all the input fields, whereas 'cancel' is not creating a new record in the database	Functional
The links provided on the forms are opening correctly.	Functional
The data feed mechanism for Read and Write files is generating a log with count of entries.	Navigation

## Suggested Technical Reading

The project is aimed at making the student understand concepts of Design and Development using IBM Rational tools, WebSphere Application Server and DB2 Database. The following reading reference is easy to understand and should be read to get a clear understanding of capabilities of the tools and how you would leverage them to execute a project.

Technical Reference	URL to access
RAD - Tackling challenges of software development with Rational Application Developer for WebSphere Software	<a href="http://www.ibm.com/developerworks/rational/library/08/0926_ackerman-mahate/index.html">http://www.ibm.com/developerworks/rational/library/08/0926_ackerman-mahate/index.html</a>
IBM Education Assistant - Rational Application Developer 7.5	<a href="http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/index.jsp?topic=/com.ibm.iea.rad_v7/rad/rad75.html">http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/index.jsp?topic=/com.ibm.iea.rad_v7/rad/rad75.html</a>
RSA-Overview of Rational Software Architect for WebSphere Software Version 7.5	<a href="http://www.ibm.com/developerworks/rational/library/08/0926_arnold/index.html">http://www.ibm.com/developerworks/rational/library/08/0926_arnold/index.html</a>
Using the new features of UML Modeler in IBM Rational Software Architect Version 7.5	<a href="http://www.ibm.com/developerworks/rational/library/08/0926_diu/index.html">http://www.ibm.com/developerworks/rational/library/08/0926_diu/index.html</a>
Rational Technical Library	<a href="http://www.ibm.com/developerworks/rational/library/">http://www.ibm.com/developerworks/rational/library/</a>