



Addis Ababa Institute of Technology

Center of Information Technology and Scientific  
Computing

Department of Software Engineering

Job portal

Software Requirements Specification

**PREPARED BY: -**

- **Abel Assefa**
- **Elias Andualem**
- **Meron Terefe**
- **Ruth Abayneh**
- **Zelalem Getahun**

**ADVISOR: - Mr. Nathnael Aregaw**

March 2018

## Revision History

Date	Description	Author	Comments
06/05/2017	Version 1.0	AEFFMZ	1 <sup>ST</sup> round revision
11/05/2017	Version 2.0	AEFFMZ	2 <sup>ND</sup> round revision
14/06/2017	Version 3.0	AEFFMZ	Final revision
19/03/2018	Version 4.0	AEMRZ	Post final revision

## Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Name	Title	Date	Signature
Fitsum Alemu	Advisor		
Nathnael Aregaw	Advisor		

## Table of Contents

<b>DOCUMENT APPROVAL.....</b>	<b>II</b>
<b>LIST OF TABLES .....</b>	<b>IV</b>
<b>LIST OF FIGURES .....</b>	<b>V</b>
<b>DEFINITIONS, ACRONYMS, AND ABBREVIATIONS.....</b>	<b>VI</b>
<b>DECLARATION.....</b>	<b>VII</b>
<b>1. INTRODUCTION.....</b>	<b>1</b>
1.1 PURPOSE.....	1
1.2 SCOPE .....	1
1.3 OVERVIEW .....	1
<b>2. GENERAL DESCRIPTION .....</b>	<b>2</b>
2.1 PRODUCT PERSPECTIVE.....	3
2.2 PRODUCT FUNCTIONS .....	3
2.3 USER CHARACTERISTICS .....	3
2.4 GENERAL CONSTRAINTS.....	3
2.5 ASSUMPTION AND DEPENDENCIES.....	3
<b>3. SPECIFIC REQUIREMENTS .....</b>	<b>4</b>
3.1 EXTERNAL INTERFACE REQUIREMENTS.....	4
3.1.1 User Interfaces .....	4
3.1.2 Hardware Interfaces.....	12
3.1.3 Software Interfaces .....	12
3.1.4 Communications Interfaces .....	12
3.2 Functional Requirements.....	13
3.3 USE CASES.....	16
3.4 NON-FUNCTIONAL REQUIREMENTS .....	27
3.4.1 Performance.....	27
3.4.2 Reliability .....	27
3.4.3 Availability .....	27
3.4.4 Security.....	27
3.4.5 Maintainability .....	27
3.4.6 Portability .....	27
3.5 INVERSE REQUIREMENTS .....	27
3.6 DESIGN CONSTRAINTS.....	28
3.7 LOGICAL DATABASE REQUIREMENTS.....	28
3.8 OTHER REQUIREMENTS.....	30
<b>4. CHANGE MANAGEMENT PROCESS.....</b>	<b>30</b>
<b>REFERENCES.....</b>	<b>31</b>

## List of Tables

Table 3.2.1: Functional Requirement 01...	4
Table 3.2.2: Functional Requirement 02...	5
Table 3.2.3: Functional Requirement 03...	6
Table 3.2.4: Functional Requirement 04...	7
Table 3.2.5: Functional Requirement 05...	8
Table 3.2.6: Functional Requirement 06...	9
Table 3.2.7: Functional Requirement 07...	10
Table 3.2.8: Functional Requirement 08...	11
Table 3.2.9: Functional Requirement 09...	12
Table 3.3.1: Use Case #1	17
Table 3.3.2: Use Case #2	17
Table 3.3.3: Use Case #3	18
Table 3.3.4: Use Case #4	19
Table 3.3.5: Use Case #5	19
Table 3.3.6: Use Case #6	20
Table 3.3.7: Use Case #7	21
Table 3.3.8: Use Case #8	22
Table 3.3.9: Use Case #9	23
Table 3.3.10: Use Case #10	24
Table 3.3.11: Use Case #11	24
Table 3.3.12: Use Case #12	25
Table 3.3.13: Use Case #13	25
Table 3.7.1: Logical Database	27

## List of figures

Figure 2.1 job seekers user interface flow diagram .....	4
Figure 2.2 employers' user interface flow diagram .....	5
Figure 2.3 Job portal home page .....	6
Figure 2.4 Job seekers registration page .....	7
Figure 2.5 Employers registration Page .....	8
Figure 2.6 Job seekers home page .....	9
Figure 2.7 Employer home page .....	10
Figure 2.8 Contact us page.....	11
Figure 2.9 About us page .....	12
Figure 2.10 Use case diagram .....	16

## **Abbreviations**

**SRS** - Software Requirements Specification

**HTML** - Hyper Text Markup Language

**HTTP** - Hyper Text Transfer Protocol

**JS** - JavaScript

**CSS** - Cascading Stylesheet

## DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included. We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

**Group AEMRZ , 19/03/2018**

## 1. Introduction

### 1.1 Purpose

The purpose of this document is to provide a basis for developing the software design for the "Job Portal" project. This document will exemplify the complete declaration for the development of the system. It will give a detailed description of the requirements. It will also explain the features of the system, the interfaces, product functions, and the system's interactions. After reading this document it is expected that the reader has a clear understanding of what the project is about as a whole, the purpose of the project and how the delivered software is supposed to meet its requirement. This document is intended for both the stakeholders and the developers of the system and will be proposed to the project advisors for its approval.

### 1.2 Scope

The project will try to help any user with any skill set seeking job either employed or unemployed to find job. The system will have the following functionality:

- ➔ Create account both Employers and candidate Employee
- ➔ Place jobs only by Employers
- ➔ Search jobs only by candidate Employee
- ➔ Apply for jobs only by candidate Employee

It will be compatible with different screen size devices (desktop, tablet or mobile).

### 1.3 Overview

This manuscript encompasses and holds information about the requirements that are needed by job portal to function. It is organized accordingly on the General descriptive section of this document. The job portal is compared to the features of similar systems that are already implemented.

It consists of functional requirements with respect to the user of the system and the expectations that are taken to accomplish to install the system. The document also contains use case diagrams which describe the main function the job portal does.

One of the most vibrant concepts when anyone needs to think about requirements are limitations. Thus, this manuscript has limits and holdups for the system implementations as well. Hence, there are user interface diagrams so any reader of this SRS document can foresee the image of the ultimate display.



## 2. General Description

### 2.1 Product Perspective

Job portal, is an online system that provide job seekers a platform for finding a right and a satisfactory job according to their skill sets (computer skills, technical skills, communication skills, research skills, problem-solving skills, etc....).

An online job finding system has been implemented in Ethiopia in earlier days; those systems provided jobs but doesn't include all level of society.

- a. **Jumia jobs:** - There are no details about the employer and no job list for daily workers. [1]
- b. **Upwork:** - Is an online job platform with fewer options to choose and doesn't include all level of individuals. [2]
- c. **Ethio jobs:** - Is an online job platform with main issue on security because a person can sign up with fake account. [3]
- d. **Egeza:** - Is an online job platform without detailed description about the employer. [4]
- e. **Addis jobs:** - Is an online job platform with little difficulty to approach the jobs (not user friendly). [5]
- f. **Qefira:** - Is an online job platform without different categories and lack of detailed information about the employer. [6]

Job portal is a system that will save the energy of employees and makes it available at anytime and anywhere. the system has alternatives with additional features that will include part time jobs and individuals to hire employees.

The product is systematically considered other online job finding systems to create the optimal vision for the job portal. The important feature that makes the job portal different from those site is that it is a system for all level of society.

### 2.2 Product Functions

At the end of development job portal is expected to serve both the employee and employer. The users will create an account, employee will be able to search or select from categories(lists) and apply for jobs and employer can post jobs and see those who had applied for the place. The portal will be compatible with different screen size devices (desktop, tablet or mobile).

## 2.3 User Characteristics

There are two types of users that interact with the system: the employee and the employer, each of the users have a different approach to use the system, so they each have different requirement on signing up.

The users should have a basic understanding of the English or Amharic languages to use the system, Basic interaction skills with a computer is needed, it is considered that users of job portal have a basic knowledge on how to work on the internet and how to access a website.

## 2.4 General Constraints

While developing Job portal platform, there will be constraints that will limit the system from reaching its full performance.

**A) Reliability Requirement-** The system will go offline when an error occurred until maintenance is done on the system.

**B) Safety and security considerations-** Since job portal will be responsible for handling personal work information of the users.

## 2.5 Assumptions and Dependencies

One assumption about the product is that the people who use the platform has a basic knowledge of using a website, internet and also understanding of the language that system will work.

To run the job portal, system hardware and a software applications are needed. The system will be able to run in any device that has an internet browser. since the portal system is web based there will be a prerequisite of an internet connection and browser and it is assumed that the user has access to attired internet connection.

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

Job portal has a home page which allows members to login and new users to register. For forgotten password, the system has resetting option. Members are classified in two major groups as job seekers and employers. Each group has different interface and are prototyped as follows.

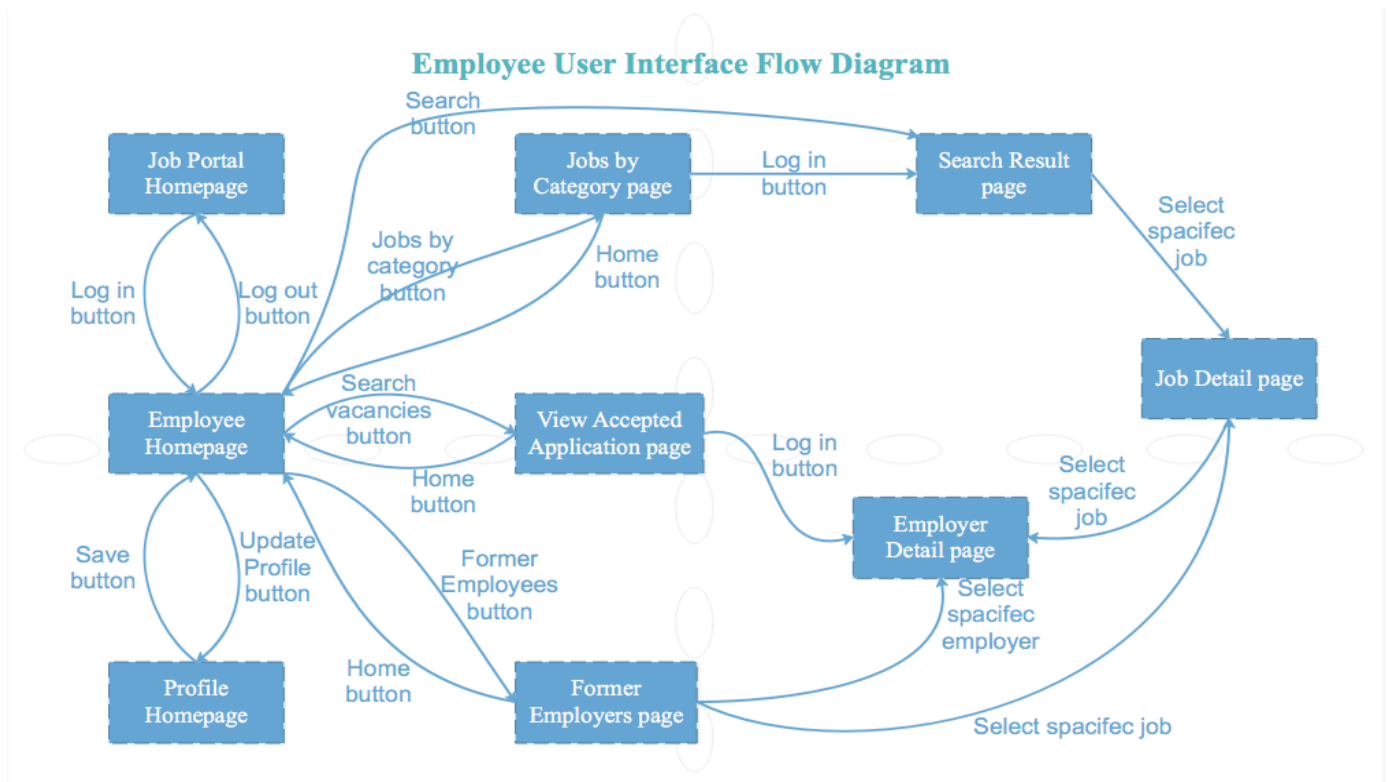


Figure 2.1 job seekers user interface flow diagram

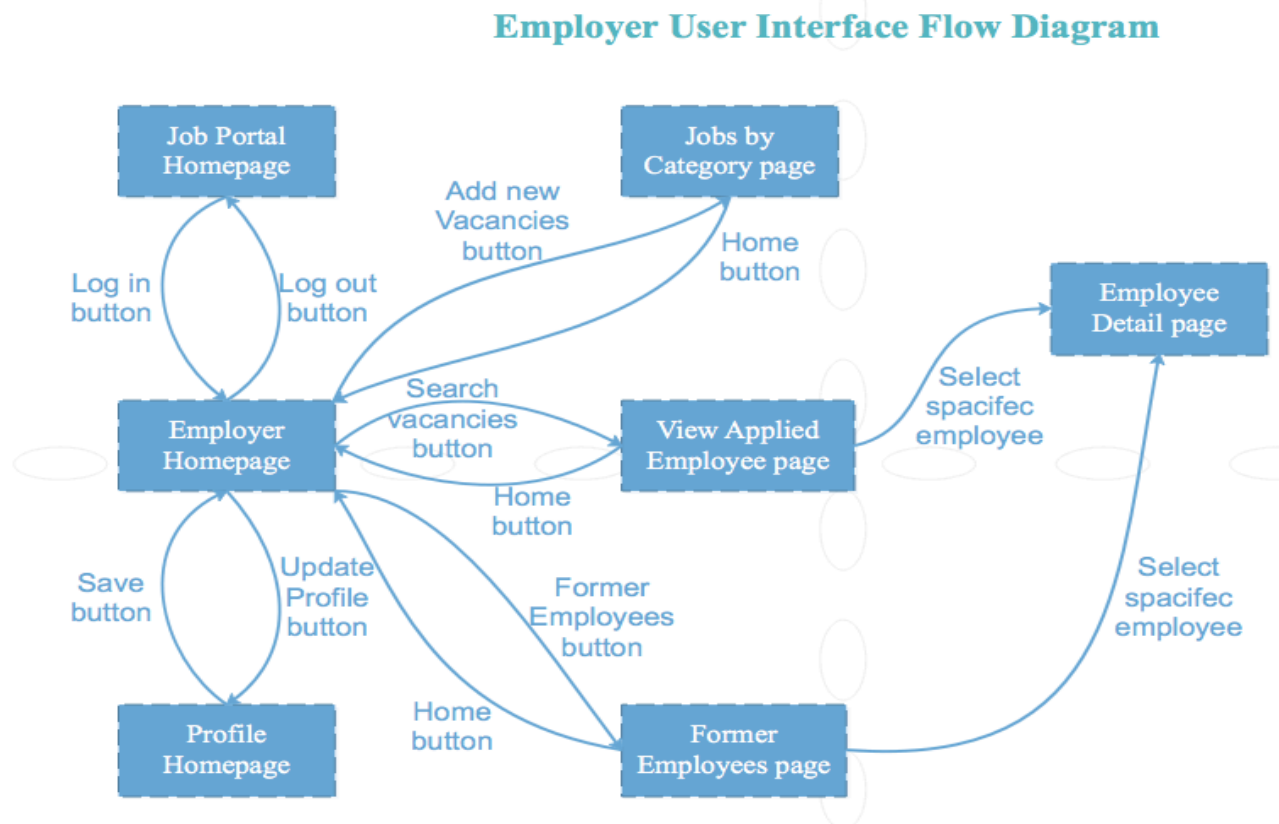


Figure 2.2 employers' user interface flow diagram

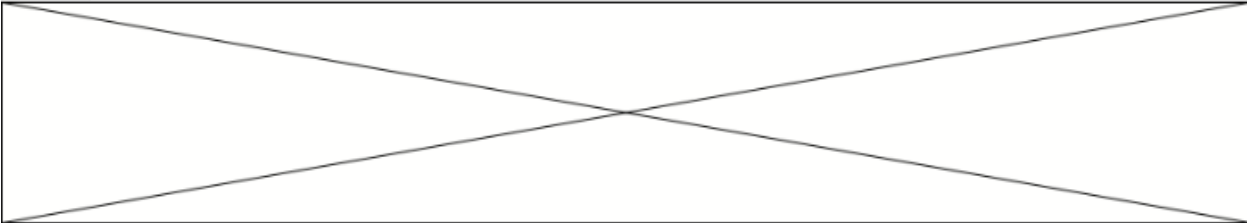
<a href="#">Home</a> <a href="#">About Us</a> <a href="#">Contact Us</a>	
	
<b>Login Form</b>	<b>Content</b>
<p>User Name</p> <input type="text"/>	
<p>Password</p> <input type="password"/>	
<p>Remember Me</p> <input type="checkbox"/>	
<input type="button" value="Log in"/>	
<p><a href="#">Forget your Password?</a></p>	
<p><a href="#">Create an account</a></p>	
<b>Footer</b>	

Figure 2.3 Job portal home page

Home	User Registration
About Us	<div>User Name:<input type="text"/></div>
Contact Us	<div>Password:<input type="password"/></div>
	<div>Confirm Password<input type="password"/></div>
	<div>Email Address<input type="text"/></div>
	<div>Date of Birth:<input type="text"/></div>
	<div>Sex<input type="text"/></div>
	<div>Phone<input type="text"/></div>
	<div>Postal/Zip Code<input type="text"/></div>
	<div>Country<input type="text"/></div>
	<div>Region<input type="text"/></div>
	<div>City<input type="text"/></div>
	<div>Address<input type="text"/></div>
	<div>About Me<input type="text"/></div>
	<div>Register</div>

Footer

Figure 2.4 Job seekers registration page

Home	User Registration
About Us	
Contact Us	
	Employer Name: <input type="text"/>
	Password: <input type="text"/>
	Confirm Password <input type="text"/>
	Email Address <input type="text"/>
	Industry Category <div>Real State</div>
	Company <input type="text"/>
	City <input type="text"/>
	Address <input type="text"/>
	Region <input type="text"/>
	Country <input type="text"/>
	Phone <input type="text"/>
	Fax <input type="text"/>
	Postal/Zip Code <input type="text"/>
	<div>Register</div>
Footer	

Figure 2.5 Employers registration page

Home

About Us

Contact Us

Welcome User

Log out

Search vacancies

Jobs by category

Jobs by Location

Accepted applications

Former employers

Update Profile

Apply for your dream jobs

Job Type

Location

Skill Required

Educational qualification

SEARCH

Footer

Figure 2.6 Job seekers home page

[Home](#)
[Profile](#)
[Messages](#)

Welcome User
[Log out](#)

Add new Vacancies

[Update Vacancies](#)  
[Accept Applied Employee](#)  
[Update Profile](#)  
[View former Employees](#)

Job Name

Job Type(optional)

Temporal ▼

Skill Required

No of Employees

Minimum Rate/h (Birr) (optional)

Maximum Rate/h (Birr) (optional)

Minimum Salary/Month (Birr) (optional)

Maximum Salary /Month (Birr) (optional)

Description

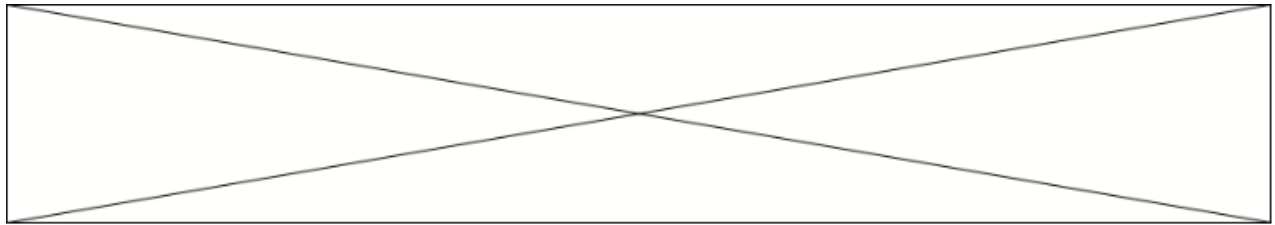
Footer

Figure 2.7 Employer home page

Center Of Information Technology and Scientific Computing

91 Page





## Contact Us

First Name

Last Name

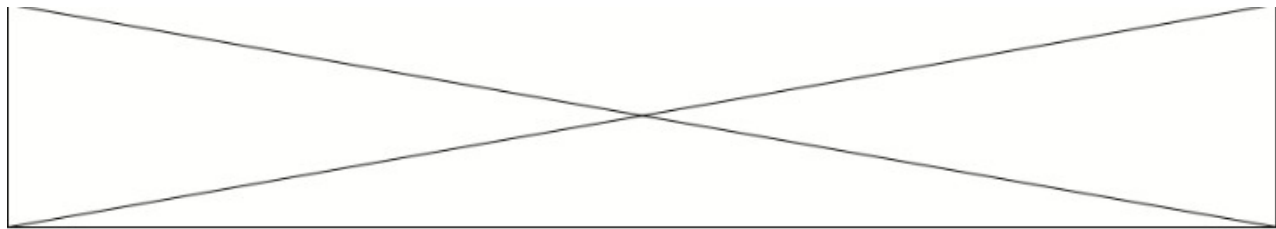
E-Mail

Message

SEND

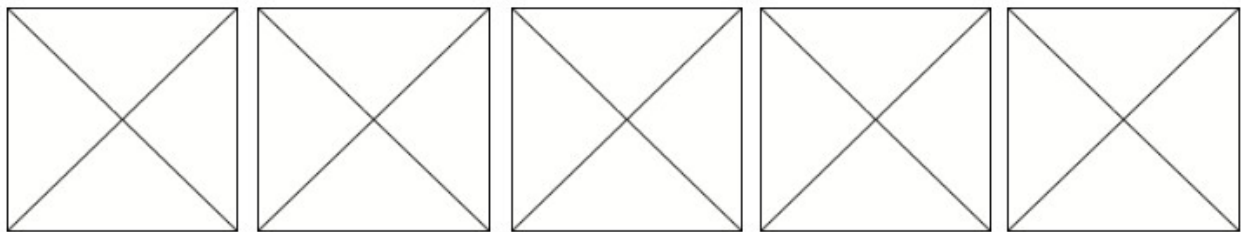
Footer

Figure 2.8 Contact us page



## Who we are?

## Our Clients



Footer

Figure 2.9 About us page

### 3.1.2 Hardware Interfaces

The system is not dependent on the hardware specifications.

### 3.1.3 Software Interfaces

This job portal is intended as a web-based system. Therefore, it is not much dependent on the specifications of the desktop, tablet PC or mobile browser. It would be accessed as long as the browser can support HTML 5 and be capable of running JavaScript.

### 3.1.4 Communications Interfaces

The system will use HTTP protocol for all communications over the internet.

## 3.2 Functional Requirements

**Table 3.2.1: Functional Requirement 01**

<b>ID</b>	FR 01
<b>Name</b>	Registering for an account
<b>Summary</b>	The system gives account to job seekers and employees
<b>Description</b>	The user can request an account that gives him/her the privilege to access or post vacancies. The system will then provide the user with a form to be filled. After checking the inserted form with the existing database, the system will be able to assign new account to users. The database will then be updated with a new information containing Password and user name.
<b>Dependency</b>	None
<b>References</b>	UC-01 and 02

**Table 3.2.2: Functional Requirement 02**

<b>ID</b>	FR 02
<b>Name</b>	View vacancies
<b>Summary</b>	The system lists job vacancies to job seekers and employers
<b>Description</b>	The job seeker can request the system to show him/her list of vacancies using different categories or search mechanism.
<b>Dependency</b>	Functional requirement 01
<b>Reference</b>	-

**Table 3.2.3: Functional Requirement 03**

<b>ID</b>	FR 03
<b>Name</b>	Apply on vacancies
<b>Summary</b>	The system allows job seekers to apply.
<b>Description</b>	The job seeker can apply for a job after viewing vacancies using their own account.
<b>Dependency</b>	Functional Requirement 02
<b>References</b>	UC-05

**Table 3.2.4: Functional Requirement 04**

<b>ID</b>	FR 04
<b>Name</b>	View Accepted Application
<b>Summary</b>	The system shows the accepted application by the Employer
<b>Description</b>	The employee can see his/her status of application on the system
<b>Dependency</b>	Functional requirement 03
<b>References</b>	UC- 07

**Table 3.2.5: Functional Requirement 05**

<b>ID</b>	FR 05
<b>Name</b>	Update Profile
<b>Summary</b>	The system allows the both the employee and the employer to update his/her profile
<b>Description</b>	Both the employee and employer can update their personal information whenever he/she has updates. Then the system updates automatically.
<b>Dependency</b>	Functional requirement 01
<b>References</b>	UC- 12 and 13

**Table 3.2.6: Functional Requirement 06**

<b>ID</b>	FR 06
<b>Name</b>	View Former Employer
<b>Summary</b>	The system allows the employee to see posts by the former Employers.
<b>Description</b>	Employee can view posts by former Employers in order to see if there are new posts under their previous company where they were hired.
<b>Dependency</b>	Functional requirement 09
<b>References</b>	UC- 08

**Table 3.2.7: Functional Requirement 07**

<b>ID</b>	FR 07
<b>Name</b>	Add new vacancies

<b>Summary</b>	The system allows the Employers to add new vacancies
<b>Description</b>	Employers will post new vacancy announcements so that, the job seekers can see and follow the steps in order to get the job.
<b>Dependency</b>	Functional requirement 01
<b>References</b>	UC- 06

**Table 3.2.8: Functional Requirement 08**

<b>ID</b>	FR 08
<b>Name</b>	Accept Applied Employee
<b>Summary</b>	The system allows the employers to accept applications.
<b>Description</b>	The system will list the employees that applied then the employer accepts applications that fulfil the given requirements
<b>Dependency</b>	Functional requirement 07
<b>References</b>	UC- 07

**Table 3.2.9: Functional Requirement 9**

<b>ID</b>	FR 9
<b>Name</b>	View Former Employees
<b>Summary</b>	Employer can view former employee lists
<b>Description</b>	The Employer may want to contact one of his/her good former employee again for some reasons.
<b>Dependency</b>	Functional requirement 09
<b>References</b>	UC- 09

### 3.3 Use Cases

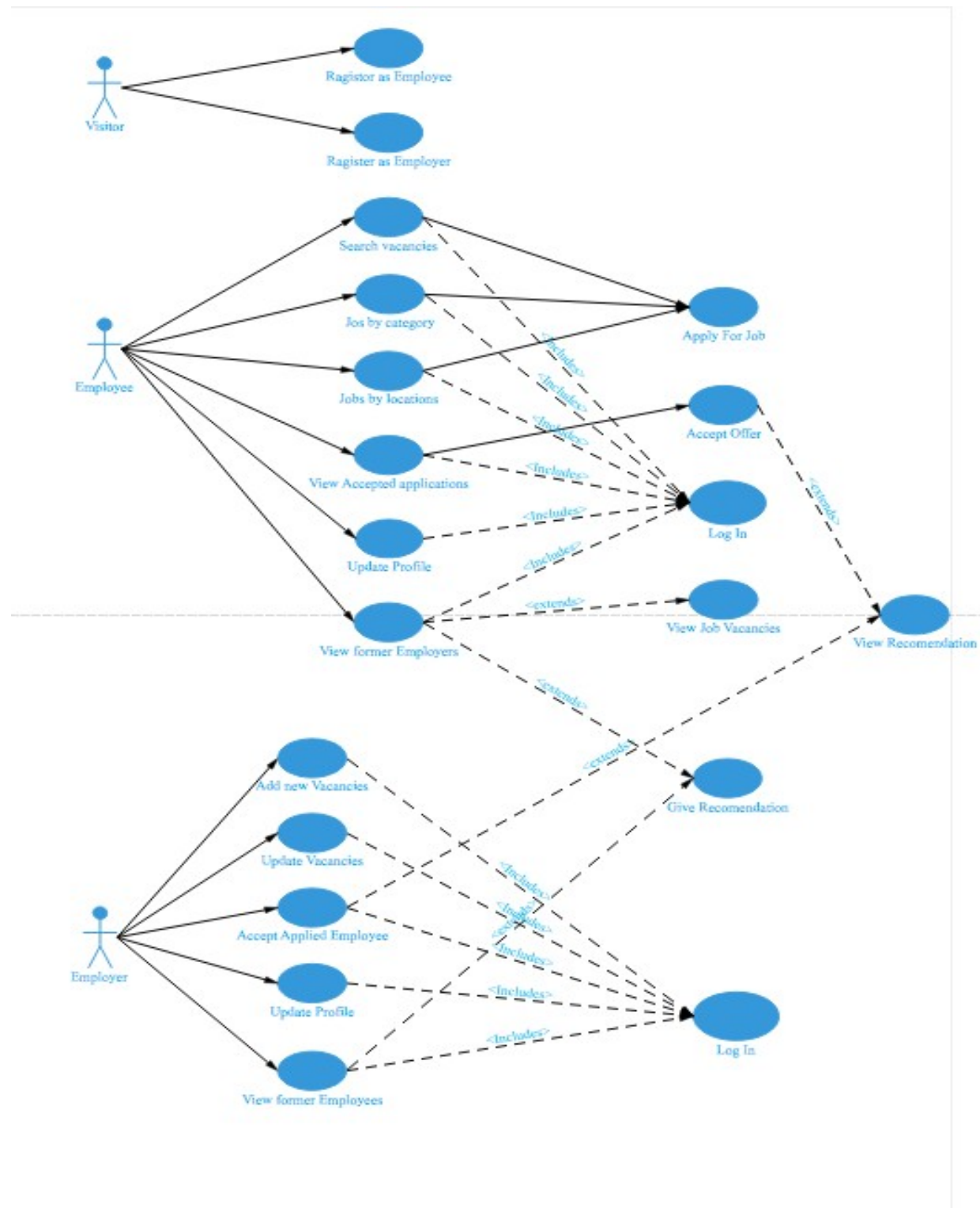


Figure 2.10 Use case diagram

### 3.3.1 Use Case #1

Table 3.3.1 Use Case #1

Use case name	<b>UC-01: Register Employee</b>
Goal	Employee registers to the job portal.
Primary Actors	Employee.
Preconditions	The user must be above age 14.
Post-conditions	Employee registers successfully.
Failure	Employee needs to register.
Trigger	The employee already has an account.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Employee clicks on sign up button.</li> <li>2. The system will display the registration form.</li> <li>3. Employee fills registration form.</li> <li>4. Employee clicks submit.</li> <li>5. The system will provide term and conditions.</li> <li>6. Employee accept the terms and conditions.</li> <li>7. The system displays success message</li> </ol>
Extension	<p>4A. Confirmation dialog box not displayed.</p> <p>4A1. The employee will fill registration form again.</p> <p>7A. Success message not displayed.</p> <p>7A1. The system will tell the user to accept the terms and conditions.</p>

### 3.3.2 Use Case #2

Table 3.3.2 Use Case #2

Use case name	<b>UC-02: Register Employer</b>
Goal	Employer registers to the job portal.
Primary Actors	Employer

Preconditions	The user must be above age 18.
Post-conditions	Employer registers successfully.
Failure	Employee fails to registers.
Trigger	The employer needs to register.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. Employer clicks on sign up button.</li> <li>2. The system will display the registration form.</li> <li>3. Employer fills registration form.</li> <li>4. The system display confirmation dialog.</li> <li>5. The system will provide term and conditions.</li> <li>6. Employer accept the terms and conditions.</li> <li>7. The system displays success message.</li> </ol>
Extension	<p>4A. Confirmation dialog box not displayed.</p> <p>4A1. The employer will fill registration form again.</p> <p>7A. Success message not displayed.</p> <p>7A1. The system will tell the employer to accept the terms and conditions.</p>

### 3.3.3 Use Case #3

Table 3.3.3 Use Case #3

<b>Use case name</b>	<b>UC-03: Employee logs in</b>
Goal	Employee logs in to his account.
Primary Actors	Employee.
Preconditions	UC-03
Post-conditions	Employee logs in successfully to his homepage.
Failure	Employee fails to log in.
Trigger	The user doesn't have an account.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. User opens job portal website.</li> <li>2. The user click log in button.</li> </ol>



	<ol style="list-style-type: none"> <li>3. The system will display login form.</li> <li>4. The user enters username and password.</li> <li>5. The system verifies user name and password.</li> <li>6. The system displays employee Homepage screen.</li> </ol>
Extension	5A. Homepage screen not displayed.  5A1. The system will return back login screen.

### 3.3.4 Use Case #4

Table 3.3.4 Use Case #4

<b>Use case name</b>	<b>UC-04: Employer logs in</b>
Goal	Employer logs in to his account.
Primary Actors	Employer.
Preconditions	UC-02
Post-conditions	Employer logs in successfully to his homepage.
Failure	Employer fails to log in.
Trigger	The user is accessing job portals home page
Main Success Scenario	<ol style="list-style-type: none"> <li>1. User opens job portal website.</li> <li>2. The system will display login form.</li> <li>3. The user enters username and password.</li> <li>4. The system verifies user name and password.</li> <li>5. The system displays employee Homepage screen.</li> </ol>
Extension	5A. Homepage screen not displayed.  5A1. The system will return back login screen.

### 3.3.5 Use Case #5

Table 3.3.5 Use Case #5

<b>Use case name</b>	<b>UC-05: Apply for Job</b>
Goal	Employee applies for particular job.
Primary Actors	Employee.

Preconditions	UC-03
Post-conditions	The user submits Application successfully.
Failure	Employee fails to submit the Application.
Trigger	Employee need to apply for job.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user clicks the view vacancies option.</li> <li>2. The system returns search form.</li> <li>3. The user fills search criteria.</li> <li>4. The user selects search button.</li> <li>5. The system returns Search result.</li> <li>6. The user selects his/her choice from the results.</li> <li>7. The System returns application form.</li> <li>8. The user fills application form.</li> <li>9. The user selects submit button.</li> <li>10. The system display massage” Application submitted successfully”.</li> </ol>
Extension	<p>5A. Search result not found.</p> <p>5A1. The system returns with massage” No search result”.</p> <p>9A. Application not submitted.</p> <p>9A1. The System returns application form again.</p>

### 3.3.6 Use Case #6

Table 3.3.6 Use Case #6

<b>Use case name</b>	<b>UC-06: Add new vacancy</b>
Goal	Upload job vacancy into system.
Primary Actors	Employer
Preconditions	UC-04
Post-conditions	Employer uploads the new vacancy successfully.
Failure	Employer fails to uploads the vacancy.

Trigger	Employer need to upload new vacancy.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects Add new vacancies option.</li> <li>2. The system returns vacancy upload form.</li> <li>3. The user fills vacancy criteria.</li> <li>4. The user selects upload button.</li> <li>5. The system uploads job into the database.</li> <li>6. The system display massage” job uploaded successfully”.</li> </ol>
Extension	<p>4A. Job not uploaded.</p> <p>4A1. The System returns upload form again.</p>

### 3.3.7 Use Case #7

Table 3.3.7 Use Case #7

<b>Use case name</b>	<b>UC-07: Accept Applied Employee.</b>
Goal	Employer hires new employee.
Primary Actors	Employer
Preconditions	UC-02, UC-04
Post-conditions	Employer accepts applicant job seekers.
Failure	Employer rejects applicant job seekers.
Trigger	Employee need to accept some of applied employees.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects accept applied employee option.</li> <li>2. The system returns applied employees list.</li> <li>3. The user selects one employee from the list.</li> <li>4. The System returns detailed information about selected employee with others recommendation.</li> <li>5. The user accepts the applied employee.</li> <li>6. The system display massage” message successfully sent”.</li> </ol>
Extension	2A. No employee Applied.

	<p>2A1. The system returns with message” No employee applied yet”.</p> <p>4A. Employee detail not found.</p> <p>4A1. The system returns applied employees list.</p>
--	---

### 3.3.8 Use Case #8

Table 3.3.8 Use Case #8

<b>Use case name</b>	<b>UC-08: Send new vacancy info for former employees.</b>
Goal	Employer requests his former employees to work in his new job.
Primary Actors	Employer
Preconditions	UC-02 and UC-07
Post-conditions	Send open vacancy information to former employee.
Failure	The system returns no former employee.
Trigger	Employee need to send open vacancy information to former employee.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects view former employees option.</li> <li>2. The system returns former employees list.</li> <li>3. The user selects one employee from the list.</li> <li>4. The System returns detailed information about selected employee with others recommendation.</li> <li>5. The user selects send message button.</li> <li>6. The system display message” message successfully sent”.</li> </ol>
Extension	<p>2A. No former employee founds.</p> <p>4A1. The system returns applied employees list.</p>

--	--

### 3.3.9 Use Case #9

Table 3.3.9 Use Case #9

<b>Use case name</b>	<b>UC-09: Apply on former employer job vacancy.</b>
Goal	Employee applies on available job vacancy at his former employer office.
Primary Actors	Employee
Preconditions	UC-02 and UC-07
Post-conditions	Employer views former employees list.
Failure	The system returns no former employer.
Trigger	Employee need to applies on open job vacancy of former employer.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects view former employers option.</li> <li>2. The system returns former employers list.</li> <li>3. The user selects one employer from the list.</li> <li>4. The System returns detailed information about selected employee with others recommendation.</li> <li>5. The user selects view job vacancy button.</li> <li>6. The system returns available vacancies result.</li> <li>7. The user selects his choice from the results.</li> <li>8. The System returns application form.</li> <li>9. The user fills application form.</li> <li>10. The user selects submit button.</li> <li>11. The system display message” Application submitted successfully”.</li> </ol>
Extension	<p>2A. No former employer founds.</p> <p>2A1. The system returns with message” No employer found”.</p> <p>11A. Application not submitted.</p> <p>11A1. The System returns application form again.</p>

**3.3.10 Use Case #10**

Table 3.3.10 Use Case #10

<b>Use case name</b>	<b>UC-010: Employer writes recommendation</b>
Goal	Employer writes recommendation for his employees.
Primary Actors	Employer
Preconditions	UC-02 and UC-07.
Post-conditions	Employer writes recommendation for employee they have hired.
Failure	The system returns no former employee.
Trigger	Employer need to write recommendation for former employee.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects view former employees option.</li> <li>2. The system returns former employees list.</li> <li>3. The user selects one employee from the list.</li> <li>4. The System returns detailed information about selected employee with others recommendation.</li> <li>5. The user selects writes recommendation button.</li> <li>6. The System returns recommendation form.</li> <li>7. The user fills recommendation form.</li> <li>8. The user selects submit button.</li> <li>9. The system display massage” Recommendation uploaded successfully”.</li> </ol>
Extension	<p>2A. No former employee founds.</p> <p>4A1. The system returns applied employees list.</p> <p>9A. Application not submitted.</p> <p>9A1. The System returns recommendation form again.</p>

**3.3.11 Use Case #11**

Table 3.3.11 Use Case #11

<b>Use case name</b>	<b>UC-11: Employee writes recommendation</b>
Goal	Employee writes recommendation for his employers.
Primary Actors	Employee.
Preconditions	UC-02 and UC-07.
Post-conditions	Employee writes recommendation for his former company.
Failure	The system returns no former employer.
Trigger	Employee need to write recommendation for former employer.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects view former employers option.</li> <li>2. The system returns former employees list.</li> <li>3. The user selects one employer from the list.</li> <li>4. The System returns detailed information about selected employer with others recommendation.</li> <li>5. The user selects writes recommendation button.</li> <li>6. The System returns recommendation form.</li> <li>7. The user fills recommendation form.</li> <li>8. The user selects submit button.</li> <li>9. The system display message” Recommendation uploaded successfully”.</li> </ol>
Extension	<p>2A. No former employer founds.</p> <p>4A1. The system returns applied employees list.</p> <p>9A. Application not submitted.</p> <p>9A1. The System returns recommendation form again.</p>

**3.3.12 Use Case #12**

Table 3.3.12 Use Case #12

<b>Use case name</b>	<b>UC-012: Employee updates his profile</b>
Goal	Employee updates his profile.
Primary Actors	Employee.
Preconditions	UC-02 and UC-07.
Post-conditions	The system updates the profile.
Failure	Employee inserts invalid impute.
Trigger	Employee need to update his profile.
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects update profile option.</li> <li>2. The system will display Profile detail.</li> <li>3. Employee Selects the information he wants to edit.</li> <li>4. The system makes the element editable.</li> <li>5. Employee edits the information.</li> <li>6. The user selects save changes button.</li> <li>7. The system display massage” Profile updated successfully”.</li> </ol>
Extension	7A. Profile not updated. 9A1. The System returns profile page form again.

**3.3.13 Use Case #13**

Table 3.3.13 Use Case #13

<b>Use case name</b>	<b>UC-13: Employer updates his profile</b>
Goal	Employee updates his profile.
Primary Actors	Employer.
Preconditions	UC-02 and UC-07.
Post-conditions	The system updates the profile.
Failure	Employer inserts invalid impute.
Trigger	Employer need to update his profile.



Main Success Scenario	<ol style="list-style-type: none"> <li>1. The user selects update profile option.</li> <li>2. The system will display Profile detail.</li> <li>3. Employee Selects the information he wants to edit.</li> <li>4. The system makes the element editable.</li> <li>5. Employee edits the information.</li> <li>6. The user selects save changes button.</li> <li>7. The system display massage” Profile updated successfully”.</li> </ol>
Extension	<p>7A. Profile not updated.</p> <p>9A1. The System returns profile page form again.</p>

### 3.4 Non-Functional Requirements

#### 3.4.1 Performance

The response time of a search should only take a minute. Selecting an element in the result list should only take a click.

#### 3.4.2 Reliability

The system should alert the users of errors. Consequently, the system should delete or remove fake accounts and be able to update its database promptly as any change occurs.

#### 3.4.3 Availability

If hosted the system should operate twenty-four hours a day, seven days a week. If the internet service gets disrupted while sending information to the server, the information should be sent again.

#### 3.4.4 Security

The system will provide accounts to users which is useful to protect it from unauthorized access. Proper login mechanism should be used to avoid personal information leaking. More than three attempts at login will automatically lock the account.

#### 3.4.5 Maintainability

The system should be malleable enough for changes and be proficient of any modifications that would be applied to it.

### 3.4.6 Portability

The system is expected to be compatible with most current web browsers and devices. User can access the system at any time and any place where there is internet accessibility.

### 3.5 Inverse Requirements

Job portal doesn't provide any service other than connecting employers and employees. The system doesn't recommend any jobs to apply or advising what kind of job to post. It does not specify anything about the chance of getting hired. The system doesn't provide any mobile (android, ios, windows) application. It won't accept any users that are not registered as a valid member.

### 3.6 Design Constraints

Are limits that hold the platform to reach its highest capacities, constraints are boundaries that occur because of unattainability of employers not providing resources(jobs) that are required to complete a category.

Other likely constraint is an online user, jobs posted on the platform expire and apparent vacancies are already closed,

internet is the main constraint the platform will become unavailable when there is no connection, another is that when the website needs maintenance it will become unavailable.

### 3.7 Logical Database Requirements

A logical database is a related information stored on physical disk files and hard drives on a computer.

This database will keep a database constituting. The information stored on the database will be accessed at almost every possible interface with the user and by almost every internal system.

The system will allow a user to sign up for account and release posts. These submitted accounts and posts will be stored in a database and will be accessed if the user searches for posts or user accounts who applied for the space. additionally, the database must store all registered user accounts personal information and detail information about the employers.

Table 3.7.1 Logical Database

Data	Attributes	Use
1. Employee data	<ul style="list-style-type: none"> <li>Employee id</li> <li>Name</li> <li>Password</li> <li>Email</li> <li>Age</li> </ul>	<p>This table is used to verify the users of Job portal using the user name and password.</p> <p>The table is also used to store additional information which are used for purpose of</p>

	<ul style="list-style-type: none"> <li>• Phone</li> <li>• Country</li> <li>• Region</li> <li>• City</li> <li>• Registration Date</li> <li>• About</li> </ul>	detail information about employee.
2. Employer data	<ul style="list-style-type: none"> <li>• Employer id</li> <li>• Name</li> <li>• Password</li> <li>• Email</li> <li>• Country</li> <li>• Region</li> <li>• City</li> <li>• Address</li> <li>• Phone</li> </ul>	<p>This table is used to verify the users of Job portal using the user name and password.</p> <p>The table is also used to store additional information which are used for purpose of providing the detail information about employer.</p>
3. Company	<ul style="list-style-type: none"> <li>• Company id</li> <li>• Company name</li> <li>• Company Profile</li> <li>• Business Stream</li> <li>• Establishment date</li> <li>• Company Website</li> <li>• Phone</li> <li>• Fax</li> <li>• Country</li> <li>• Region</li> <li>• City</li> <li>• Address</li> <li>• Employer id</li> </ul>	This table is used to store give company details and contact info for appropriate employees.
4. Employee Educational Detail	<ul style="list-style-type: none"> <li>• Educational Detail id</li> <li>• Institute or University name</li> <li>• major</li> <li>• Certificate or Degree name</li> <li>• Starting date</li> <li>• Completion date</li> <li>• GPA</li> <li>• Employee id</li> </ul>	This table is used to prove educational details of employees for employers.
5. Skill set	<ul style="list-style-type: none"> <li>• Skill set id</li> <li>• Skill set name</li> </ul>	

6. Employee Skill set	<ul style="list-style-type: none"> <li>Employee id</li> <li>Skill set id</li> <li>Detail</li> </ul>	This table is used to provide list values of Skill requirement custom list for employee searching for job vacancies.
7. Experience Detail	<ul style="list-style-type: none"> <li>Employee id</li> <li>Job id</li> <li>Employer id</li> <li>Starting date</li> <li>End date</li> <li>Recommendation</li> </ul>	This table is used to store recommendation given by employers to employees and the data is used for showing recommendation of the employees.
8. Employer Detail	<ul style="list-style-type: none"> <li>Employer id</li> <li>Employee id</li> <li>Job id</li> <li>Starting date</li> <li>End date</li> <li>Recommendation</li> </ul>	This table is used to store recommendation given by employees to employer and the data is used for showing recommendation of the employer.
9. Job	<ul style="list-style-type: none"> <li>Job id</li> <li>Job Type</li> <li>Created date</li> <li>Job description</li> <li>Job location</li> <li>is active</li> <li>Skill Required</li> <li>Educational qualification required</li> <li>Employer id</li> </ul>	This table is used to store uploaded job vacancies by employers and the data is used for showing search result for job vacancy.
10. Image	<ul style="list-style-type: none"> <li>Owner</li> <li>Owner id</li> <li>Image Detail</li> </ul>	This table is used to provide images of Employees, employers, Companies and uploaded CVs of employees.

### 3.8 Other Requirements

For our system to hit its target successfully, users should be able to use it to its full capacity. This will be possible with some intuitive skill that will develop over time. To fasten this, we have prepared a ‘help’ item in the job portal header menu that will guide users through the website. Since our system is web base application, files of the system will be placed in a folder and are in ‘.html’, ‘.CSS’, ‘.JS’, ‘.PHP’ and other web formats.

## 4. Change Management Process

During this project, our team is trying to follow the water fall model which makes it harder to be flexible regarding requirements but to make sure there will not be a lot of change after this we have tried to understand the requirements as much as possible so that we can easily understand the problem at hand and know what should be done to solve the problem, but since there could be better choices than what we thought change is expected in the SRS document.

For any change, the project scope and requirements shall be discussed by all the members of the team. A change will be made only when the majority of the team agrees on the change. In this case, the SRS document shall be updated by the team members in order to reflect the changes, and a date of change shall be noted in the file. If this change request is made by the client or anyone outside of the team, he or she will have to contact the team. If a change request is made by a team member, he or she can raise it during the weekly team meeting.

## References

1. Jumia jobs, (may 13,2017), time(7:00 ] local time)
2. Upwork, (may 13,2017) time(7:00 local time)
3. Ethio jobs, (may 13,2017) time(7:00 local time)
4. Egeza, (may 13,2017) time(7:00 local time)
5. Addis jobs, (may 13,2017) time(7:00 local time)
6. Qefira, (may 13,2017) time(7:00 local time)