JOB SITE

Software Requirements

**SUBMITTED BY**

JOSMY JOSE

**SUPERVISED BY**

UTHARA D, IPSR SOLUTIONS PVT LTD

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | **Title Page No**   1. **INTRODUCTION**    1. Purpose 1    2. Scope 1    3. Definitions, Acronyms and Abbreviations 1    4. References 2    5. Overview 2 2. **OVERALL DESCRIPTION**    1. Product Perspective 3       1. System Interface 3       2. User interface 3       3. Hardware Interface 3       4. Software Interface 4       5. Communication Interfaces 4       6. Memory Constraints 4       7. Operations 4       8. Site Adaption Requirements 4    2. Product functions 5       1. Context Diagram 5       2. Use Case Diagrams 6       3. Use case descriptions 7    3. User Characteristics 8       1. Jobseeker 8       2. Employer 8       3. Admin 8    4. Constraints 8       1. User Interface Constraints 8       2. Hardware Constraints 8       3. Software Constraints 8       4. Data Management Constraints 9       5. Operational Constraints 9       6. Site Adaptation Constraints 9       7. Design Standards Compliance 9    5. Assumptions and dependencies 9    6. Apportioning of Requirements 9 3. **SPECIFIC REQUIREMENTS**    1. External interface 10       1. Web Server 10       2. Python Django Application . 10       3. MySQL Database 10    2. Functional Requirements 10    3. Performance Requirements 14    4. Logical database requirements 14    5. Design Constraints 14    6. Software System Attributes 15       1. Reliability 15       2. Availability 15       3. Security 15       4. Maintainability 15       5. Portability 16 | . |

**CHAPTER 1 : INTRODUCTION**

**1.1 Purpose**

The purpose of designing the "JOB SITE" is to give the job seekers a platform for finding a right and a satisfactory job according to their qualification. It also connects the job seekers with the major agencies. It also provides a job portal for Job Seekers to submit their CVs and apply for job posting and Employer can select the best Employees from Available CV based on their payment option selection. This is basically a Job portal where job Seeker applies for jobs and employer post jobs and select prospective applicant. The job portal is prepared to provide all categories of jobs and help to get various types of jobs. The main purpose of the job portal is to provide the facility to job seekers for getting a quick job. So, it enables applicants to search for jobs conveniently and to enable employers to find suitable candidates.

**1.2 Scope**

The Software Requirements Specification captures all the requirements in a single document. The online job Portal System that is to be developed provides the members with job information, online applying for jobs, and many other facilities. The basic scope of the project is given as under:

1. Maintain Job Seeker and Employer records
2. Maintain uploaded Resumes
3. Provide Customized Job Postings
4. Maintain Job Posting details and generate various reports
5. Maintain job seeker and employer job profile

**1.3 Definitions, Acronyms and Abbreviations**

**1.3.1 Administrator:**

admin manages the entire application and maintains the profile of applicants and employers. The registered account become valid only after admin approves it.

**1.3.2 Jobseeker**

This module provides functionalities for job seekers. Applicants can post their resumes

with personal and professional details. They can also update the resume as frequently as required. The applicant can also browse through the present vacancies available. Job seekers can also get mail alerts when their resumes are selected by employers.

**1.3.3 Employer**

In this section, employers can post vacancy details and update the details as and when necessary. Employers can search through applicant resumes based on different criteria.

**1.4 References**

1. Literature review on online job portal

<https://ijarcce.com/upload/2017/may-17/IJARCCE%2019.pdf>

2. <https://www.naukri.com/account/register/basicdetails>

3. <https://corp.freshersworld.com/post-job?src=homeheader>

4. <https://www.freshersworld.com/>

**1.5 Overview**

This is a working document and, as such, is subject to change. In its initial form, it is incomplete by definition, and will require continuing refinement. Requirements may be modified and additional requirements may be added as development progresses and the system description becomes more refined. This information will serve as a framework for the current definition and future evolution

**CHAPTER 2: OVERALL DESCRIPTION**

**2.1 Product Perspective**

The Online Job Portal System is a package to be used by agencies to improve the efficiency of business. The Online Job Portal System to be developed benefits greatly the members. The system provides jobs catalog and information to members and helps them decide on the jobs to apply. The Admin can keep the jobs catalog updated all the time so that the members (Job seekers and the agencies) get the updated information all the time.

The main users are users: Admin, Members who are the Job seekers and the agencies.

**2.1.1 System Interface**

Apache will be used as web server. The user inputs data via the web server using HTML forms. The actual program that will perform the operations is written in Python Django.

**2.1.2 User Interface**

The new system shall provide a very intuitive and simple interface to the user and the administrator, so that the user can easily navigate through pages, Inputs the data, get notifications when new jobs are posted by employer.

**2.1.3 Hardware Interface**

a) Server side the web application will be hosted on a web server which is listening on the web standard port, port 80.

b) Client-side Monitor screen – the software shall display information to the user via the monitor screen Mouse – the software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, command buttons and select options from menus. Keyboard – the software shall interact with the keystrokes of the keyboard. The keyboard will input data into the active area of the database.

**2.1.4 Software Interface**

a) Server side An Apache web server will accept all requests from the client and forward it accordingly. A database will be hosted centrally using MySQL.

b) Client side An OS which is capable of running a modern web browser which supports JavaScript and HTML5.

**2.1.5 Communication Interface**

The HTPP or HTTPS protocol(s) will be used to facilitate communication between the client and server.

**2.1.6 Memory Constraints**

Memory constraints will come into play when the size of MySQL grows to a considerable size.

**2.1.7 Operations**

The product shall have operations to protect the database from being corrupted or accidentally altered during a system failure.

**2.1.8 Site Adaption Requirements**

Not Applicable

**2.2 Product Functions**

**2.2.1 Context Diagram**

**Jobseeker**

Register/Login

View jobs/search jobs

Apply jobs/upload cv

Login,

Manange whole system,jobseeker,

employer,jobs

**Job Site**

**Admin**

**Employer**

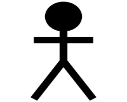
View/approve applicants

Register/Login

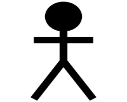
Post job

Search cv

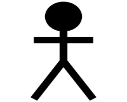
**2.2.2 Use Case Diagrams**

****

**EMPLOYER**



**ADMIN**

****

**JOBSEEKER**

View applicants

**2.2.3 Use case descriptions**

**Actors:**

The Actors of the system are Admin, Employer,

**Use cases:**

I have identified a set of use cases based on the functionalities and goals of the application.

• Register Account- This use case denotes a set of actions required for Employer and Job seeker to register with the application.

• Login- This use case denotes a set of actions required for Employer, Admin and Job seeker to login into the application.

• Add Job Vacancy- This use case denotes a set of actions required for Employer to post a job vacancy.

• Approve Job Post- This use case denotes a set of actions required for admin to change the status of the Job Post.

• View Applicants for a Job Post- This use case denotes a set of actions required for Employer to view the list of applicants for a particular job post.

• View Job Posts- This use case denotes a set of actions required for Employer to view all the jobs posted by the Employer.

• Search Job Posts- This use case denotes a set of actions required for Job Seeker to search available and active jobs.

• Apply for Job- This use case denotes a set of actions required for Job Seeker to apply for an available job vacancy.

• Manage users- this use case denotes a set of actions required for admin to manage both jobseeker and employer activities.

• Notify Jobseeker- This use case denotes that employer notify the jobseeker interest for their vacancy

• Manage website- This use case denotes a set of actions required for admin to manage the whole content of the web site.

• search cv- This use case denotes a set of actions required for employer to search candidate cv on the basis of certain criteria.

**2.3 User Characteristics**

**2.3.1 Jobseeker**

Job seekers will use this application for getting up-to-date job posts and applying for decent jobs.

**2.3.2 Employer**

Job provider will post those job circulars and can maintain the resumes and contact with the candidates.

**2.3.3 Admin**

Admin panel can monitor all the information.

**2.4 Constraints**

**2.4.1 User Interface Constraints**

Using this system is fairly simple and intuitive. A user familiar with basic browser navigation skills should be able to understand all functionality provided by the system.

**2.4.2 Hardware Constraints**

The system should work on most home desktop and laptop computers which support JavaScript and HTML5.

**2.4.3 Software Constraints**

The system will be intended to run on Firefox 4 and above, Google Chrome 10 and above and Internet Explorer 8 and above.

**2.4.4 Data Management Constraints**

System shall be able to interface with other components according to their specifications.

**2.4.5 Operational Constraints**

The system is limited by its operating server in terms of the maximum number of users it can support at a given time.

**2.4.6 Site Adaptation Constraints**

The component will be adapted to the overarching system at the conclusion of the system creation.

**2.4.7 Design Standards Compliance**

The system shall be implemented in Python Django.

**2.5 Assumptions and dependencies**

Our new system focuses on the features which are most important to the users of an vehicles along with introduction of some new features which other portals lacks.

**2.6 Apportioning of Requirements**

Integration of LDAP login m

**CHAPTER3: SPECIFIC REQUIREMENTS**

**3.1 External interface**

**3.1.1 Web Server**

• Apache will be used as web server:

• The user inputs data via the web server using HTML forms

• The web server executes the Python as a module and Python script retrieves the post data

if available.

• The web server receives information back from the Python script.

•The web server displays a HTML page as result to the end-user.

**3.1.2 Python Django Application**

The actual program that will perform the operations is written in Python Django. All data will be stored in a database.

**3.1.3 MySQL Database**

It’s an open source SQL database to store all data which communicates with the application on the server.

### 3.1 Functional Requirements

1. **As a** new user  
   **I want to** register my account  
   **So that,** I can get benefited by this website.  
   **Confirmation:**  
   –Open the application  
   –Click on SignUp  
   –Enter all the required information  
   –Click on SignUp
2. **(a) Success**  
   user will be directed to the login page  
   **(b) Failure**  
   –Email not match with regex, show error: Please provide a valid email  
   –Password is less than 6 digits, show error: Password must include at least 6 digits time.  
   –Any field left blank, show error: All the fields must be filled up.
3. **As a** user  
   **I want to** login into my account  
   **So that,** I can get benefited by this website.  
   **Confirmation:**  
   –Open the application.  
   –Click on login.  
   –Enter all the required information.  
   –Click on login.  
   **(a) Success**  
   Valid users will be directed to the login page.  
   -“Remember me” ticked – store cookie/automatic login next time.  
   –“Remember me” not ticked – force login next time.  
   **(b) Failure**  
   –incorrect email, show error: Incorrect email address.  
   –incorrect Password, show error: Incorrect password.  
   –Any field left blank, show error: Something went wrong.
4. **As a user**  
   **I want to** change my password.  
   **So that,** I can secure my account.  
   **Confirmation:**  
   –Open the "profile" tab  
   –Click on My Account  
   –Click on Change Password  
   –Enter the new password  
   –Enter the old password  
   –Click on Done  
   **(a) Success:**  
   -"Password has been changed successfully!"- pop up will show.  
   **(b) Failure:**  
   -Old password may not match with the given password [now], show error: Please provide your old password correctly.  
   -New password is less than 6 digits, show error: Password must include at least 6 digits time.  
   -Any field left blank, shows error: All the fields must be filled up.

### Jobseeker perspective

1. **As a** Jobseeker  
   **I want to** edit my profile.  
   **So that,** I can keep my profile up to date.  
   **Confirmation:**  
   -Open the homepage.  
   –Click on the "Profile" tab.  
   –Jobseekers can then view the previously saved version of their profile.  
   –They can also update their personal information.
2. **As a** Jobseeker  
   **I want to** upload my CV.  
   **So that,** I can apply according to the specification.  
   **Confirmation:**  
   -Open the homepage.  
   –Click on "Profile" tab.  
   –Click on "Upload CV" option.  
   –Job seekers can upload their CVs.
3. **As a** Jobseeker  
   **I want to** apply for different jobs.  
   **So that,** I can get a decent job.  
   **Confirmation:**  
   -Open the homepage.  
   –Click on "Jobs" tab.  
   –Choose a particular job.  
   –Click on "Apply Easily".  
   -Choose your recent CV.  
   -Click on "Apply".

### Employer perspective

1. **As a** Employer  
   **I want to** post for new jobs.  
   **So that,** I can get a decent candidate.  
   **Confirmation:**  
   -Open the homepage  
   –Click on "Add Jobs" tab  
   –Edit the add job field  
   -Click on "Post".
2. **As a** Employer  
   **I want to** view the application progress.  
   **So that,** I can keep track of the candidates applying.  
   **Confirmation:**  
   -Open the homepage  
   –Click on "Added jobs" tab  
   -Click on particular job  
   –Click on "candidates" -Job provider now can view the candidates.
3. **As a** Employer  
   **I want to** collect the CVs.  
   **So that,** I can make my decisions about a particular candidate.  
   **Confirmation:**  
   -Open the homepage  
   –Click on "Added jobs" tab  
   -Click on particular job  
   –Click on "candidates
4. –Click on "CV Bank"  
   -Job provider now can collect the CVs.
5. **As a** Employer  
   **I want to** edit my profile.  
   **So that,** I can show the applicants the details about my organization.  
   **Confirmation:**  
   -Open the homepage  
   -Click on "Profile" tab  
   -Jobseekers can then view the previously saved version of their profile.  
   -They can also update their personal information.

**3.3 Performance Requirements**

The system should support at least 200 concurrent users. This statement provides a general sense of reliability when the system is under load. It is important that a substantial number of users be able to access the system at the same time, since an academic portal is important to the courses that employ it. The times when the system will be under the most stress are likely during assignment submissions. Therefore, it must be able to handle at least 200 concurrent users.

**3.4 Logical database requirements**

All data will be saved in the database: user accounts and profiles, discussion data, messages etc. (except files which are stored on the disk.) The database allows concurrent access and will be kept consistent at all times, requiring a good database design.

**3.5 Design Constraints**

▪ The communication between the portal software and the database will be in SQL.

▪ The portal layout will be produced with HTML/CSS.

▪ The product will be written in Python Django.

▪ The output must be compatible with W3C XHTML 1.0

▪ The source code must follow the coding conventions of Python.

▪ System administrators must have access to comprehensive documentation.

**3.6 Software System Attributes**

The software consists of the following elements:

▪ The apache web server

▪ The Python Django application

▪ The MySQL database

▪ The database should remain consistent at all times in case of an error.

**3.6.1 Reliability**

The reliability of the overall program depends on the reliability of the separate components.

**3.6.2 Availability**

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. In case of a of a hardware failure or database corruption, a replacement page will be shown. Also, in case of a hardware failure or database corruption, backups of the database should be retrieved with the MySQL server and saved by the administrator.

**3.6.3 Security**

▪ Passwords will be saved encrypted in the database in order to ensure the user's privacy.

▪ The user's IP will be logged.

▪ The system will be protected against vulnerabilities such as SQL injection attacks.

**3.6.4 Maintainability**

MySQL is used for maintaining the database and the Apache server takes care of the site. In case of a failure, a re-initialization of the program is recommended.

**3.6.5 Portability**

The application is Linux-based and should be compatible with other systems. Apache, Python Django and MySQL programs are practically independent of the OS-system which they communicate with. The end-user part is fully portable and any system using any web browser should be able to use the features of the application.