**<Online Job>**

**System Requirements Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Name** | **ID** | **Group** |
| 1 | Donia Yasser Saied | 20201230 | S5 |
| 2 | Sarah Tareq Abd El-Alim | 20201087 | S5 |
| 3 | Reham Ashraf Fathee | 20201079 | S5 |
| 4 | Aziz Samy Aziz | 20200327 | S5 |
| 5 | Mohammed Samy Hmam | 20190442 | S5 |
| 6 | Belal Ahmed Gharib | 20200769 | S5 |

Project contact member email : st473614@gmail.com

Project contact member mobile : 01141723072

TA. <Eng: Ahmed Abd El-Aziz>

# Purpose:

* The system should be user-friendly system and aim to streamline the Job Search process, providing a comprehensive solution for both applicants and employers.
* Facilitate effective and direct communication between employers and applicants.
* Help job seekers to find Varies job opportunities relevant to their skills.

# Scope:

* Creating a comprehensive solution for both employers and job seekers.
* **Features:**
  + User authentication and personalized profiles create and manage for job seekers.
  + Job listings categorized by industries, positions, and locations.
  + Detailed information about job listings including company profiles, job descriptions, and application instructions.
  + Job apply, application submission, and feedback tracking.
  + Search and filter jobs.
  + Integration with a messaging system for direct communication between employers and applicants.
* Support a mobile responsive design to enable users to engage with the platform on various devices.
* Allowingusers to stay connected to job opportunities while on the move.

# Audience:

(1) Job Seekers:

- Individuals looking for job opportunities.

(2) Employers:

- Organizations or companies that want to hire qualified candidates for their

available jobs.

# Definition and abbreviation:

1. IDE (Integrated Development Environment):
   * + Is a software application that provides facilities to programmers for software development.
2. HTML (Hypertext Markup Language):
   * + The standard markup language for creating web pages and applications.
3. CSS (Cascading Style Sheets):
   * + A style sheet language used for describing the look and formatting of a document written in HTML or XML.
4. JS (JavaScript):
   * + is the main programming language of the Web, a versatile scripting language primarily used for front-end web development.
5. JSON (JavaScript Object Notation):
   * + Data-interchange format.
6. UI (User Interface):
   * + The visual elements and interactions through which users interact with the software.
7. UX (User Experience):
   * + The overall experience and satisfaction that users have while interacting with the software, including factors such as ease of use, efficiency, and enjoyment.
8. API (Application Programming Interface):
   * + A set of rules and protocols that allow different software applications to communicate and interact with each other.
9. SQL (Structured Query Language):
   * + Used in programming for managing data in a relational database.

# System Overview

## Problem Statement

1. Limited reach: physical visits to companies limited the number of job opportunities that a job seeker could access. This meant that they may miss out on potential opportunities that were not aware of through their network.
2. Manual effort: Submitting applications and tracking the status of each application required a significant amount of manual effort. Job seekers had to fill out paper applications or type up cover letters and resumes for each position they were interested in. They also had to keep track of where they had applied and follow up with each company individually.
3. Time consuming : The manual nature of the job search process made it inefficient. Job seekers had to spend a lot of time and energy on administrative tasks rather than focusing on finding the right job for them
4. Lack of transparency: Without an online system for submitting applications, it was difficult for job seekers to know the status of their applications. They often had to wait weeks or even months without any updates from the companies they applied to.

## 

## Proposed Solution

## Advanced Search and Filtering: The program provides a search and filtering feature to ease your search for your specialization amid the abundance and diversity of job fields.

1. Efficient Application Process : The platform provides easy job application; you can simply choose the job category you are interested in, view all available jobs, and apply by submitting your CV.

## Mobile Accessibility :The program enables you to interact and access it through mobile devices, ensuring user-friendly accessibility for all users.

1. Transparency in Job Postings: The program offers all job details, including salary and working conditions, to ensure transparency in job postings.
2. Application Status: The platform allows job seekers to track the status of their applications.

## Review Application : The program offers a feature for self-improvement by allowing employers to provide feedback, giving you insight into the reasons for job rejection.

1. Communication Tools: Employers can communicate with applicants through the platform, facilitating interview scheduling, sharing feedback, and providing updates on the application status.

## 

# Assumptions and/or constraints:

* **Assumptions:**

1. Internet Connectivity:

- The software assumes that users will have access to a stable internet connection to

use the application effectively.

1. User Device:

- It is assumed that users will have access to devices such as computers, laptops,

tablets, or smartphones with compatible web browsers or a dedicated mobile

application to access the online job recruitment software.

1. Data Accuracy:

- It is assumed that the information provided by employers, such as job listings and

company profiles are accurate and up-to-date.

1. User Engagement:

- The assumption is that job seekers will actively engage with the software, regularly

updating their profiles and actively applying for job listings.

* **Constraints:**

1. Time Constraints:

- The development and deployment of the software may be subject to time

limitations or project deadlines, impacting the implementation of certain features or

functionalities.

1. Budget Constraints:

- Software development and maintenance may be constrained by budget limitations,

affecting the available resources for development, infrastructure, and ongoing

support.

1. Security Constraints:

- The software must adhere to security best practices to protect user data, including

personal information and resumes, from unauthorized access or breaches.

1. Scalability Constraints:

- The software should be designed to handle a growing user base and increase job

listings without significant performance degradation.

1. Legal and Regulatory Compliance:

- The software must comply with relevant laws and regulations related to data

protection, privacy, equal employment opportunity, and other employment-related

regulations.

1. Compatibility Constraints:

- The software should be compatible with a wide range of web browsers, operating

systems, and device types to ensure accessibility for users.

1. Organizational Constraints:

- Development may be subject to organizational constraints, such as internal

policies, procedures, and approval processes, which may impact the development

timeline and decision-making.

# System Requirements

## Functional Requirements:

**Job Seekers:**

1. **Create a profile:**

. Include his personal information (Name, age, address, etc), skills, experiences, and aspirations.

1. **Manage profile:**

. He can edit his information to make it updated all the time.

1. **Search about a job:**

. He can search for a specific job by its (type-salary range-required skills).

1. **Exploring diverse job listings:**

. Diverse job listings are categorized by industries, positions, and locations and each job presents detailed information, including company profiles, job descriptions, and application instructions.

1. **Apply for jobs:**

. Users can apply for multiple positions effortlessly and submit the form.

1. **Notify of relevant job openings:**

. User can filter jobs by their (type- salary range-required skills) and the system will notify them by available jobs ensuring that they stay informed about opportunities aligning with their career goals.

1. **Receive feedback:**

. Receive feedback on the status of the application is provided by the recruiter, ensuring transparency and reducing uncertainties.

1. **Communicate with employer:**

. System enabling direct communication between employers and applicants. This facilitates clarifications, interview scheduling, and overall engagement throughout the recruitment process.

**Employers:**

1. **Create job postings:**

. Create job post including job information’s like (Salary-location-Required skills-time-description).

1. **Manage job postings:**

. employer can edit any information about the job to make it updated all the time.

1. **Review applicant profile:**

. review applicant profiles to choose the best candidates for the job and communicate with them and examine who are suitable for that position.

1. **Communicate with candidates:**

. After reviewing applicant profiles the company give them feedback to inform them they are rejected or accepted to inform them about time of interview.

## Non-Functional Requirements:

1. Performance:

- The application should provide fast response times to ensure a smooth user

experience (e.g: average response time taken to load pages or perform

searches 2 seconds).

1. Reliability:

- The application should have high availability and minimal downtime to ensure

users can access it when needed (e.g: the system must be available 98% of

the time).

1. Usability:

- The application should have an intuitive and user-friendly interface, allowing

users to navigate and perform tasks easily, the average time to complete

common tasks efficiently and without confusion (e.g: creating a profile,

applying for a job, adding job listings) must be done within 3 min.

1. Maintainability:

- The software should be designed and developed in a way that allows for easy

maintenance and quick resolution of issues Mean Time to Repair (MTTR) must

be within a few days.

1. Compatibility:

- The application should be compatible with a wide range of web browsers to

ensure accessibility for users (e.g., support for major web browsers like

Chrome, FireFox, and Microsoft Edge).

- The software should be compatible with various devices, including desktops,

laptops, tablets, and smartphones, to provide a consistent experience and

should support different screen sizes and resolutions for various devices.

# Use Case Model

## Use Case Diagram:

## 

## 

## 8.2 Use Case narratives

* **E.g., you may not do LogIn Use case table as it is simple functionality**
* **Flow of events should be very detailed**
* **Using below table template, for *two of the use cases* (from the use case diagram)**

|  |  |
| --- | --- |
| * **Name : Apply for job** |  |
| **Actor :Job Seeker** |  |
| **Entry Condition :**  **(1) He should be registered in the system,**  **(2) His profile should be established**  **(3) job information should be provided** | **2** |
| **Exit Condition :(1) He applied for his desired job,**  **(2) A feedback on the application status is provided from the recruiter** |  |
| **Flow Of Events :1-job seeker creates an account,**  **2- System checks the authentication,**  **3- He edits and manages its profile,**  **4- He explores job listing and its details using searching and filtering,**  **5- He applies job in which he is interested**  **6- Submitting the app**  **7 – Receiving app status** |  |
| **Alternative Events :If he remembered that he entered any wrong information in the submitted app, then he would return to step 4 up to the last step** |  |

# 

|  |  |
| --- | --- |
| **Name : Communicate with potential candidate** |  |
| **Actor : Employer** |  |
| **Entry Condition :**  **(1) He should create a job post and manage it**  **(2) He should be received applications from job seekers and reviewed their profiles** | **2** |
| **Exit Condition :­­ If he found the candidate as appropriate person for the announced position** |  |
| **Flow Of Events :**  **(1) He creates an account,**  **(2) System checks the authentication,**  **(3) Company profile which he represents should be established**  **(4) He should create a job post then manage it after system authentication acceptance**  **(5) He should be received applications from job seekers and checked them**  **(6) He should be reviewed their profiles**  **(7) finally ,He should communicate with whom he finds appropriate for that position** | **4** |
| **Alternative Events : if the applicant ignores the recruiter ,recruiter returns to step 5 up to final step** |  |

# 9. Sequence Diagram

* **Provide 2 sequence diagrams [**complex use case scenarios**]**
* **Use ArgoUML or visual-paradigm to Draw the sequence diagram**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated