

**TRIBHUVAN UNIVERSITY**

**Institute of Science and Technology**

**“Job Recommendation System.”**

**A Project Report**

**Submitted to:  
CSIT Department   
Bhaktapur Multiple Campus  
Dudhpati, Bhaktapur**

**In partial fulfillment of the requirement for the  
Bachelor's Degree  
in  
Computer Science and Information Technology**

**Submitted By**Chetan Raj Budhathoki (T.U. Roll No. 20232/075)  
Bimal Shrestha (T.U. Roll No. 20227/75)  
Nabin Bhandari (T.U. Roll No. 20247/075)

**May 2023**

**Certificates**

Shape, arrow

Description automatically generated

**Supervisor Recommendation**

This is to certify that the report entitled “Job Recommendation System using Cosine Similarity Algorithm” has been prepared under my supervision by Chetan Raj Budhathoki (20232/075), Nabin Bhandari (20247/075), and Bimal Shrestha (20227/075) in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Information Technology.

I have read and approved the report and recommend it for evaluation. The report meets the required standards for the degree program and demonstrates a high level of understanding and competence in the field of computer science and information technology.

…………………..……………..…  
**Saroj Maharjan**Supervisor/Visiting Faculty  
CSIT Department  
Bhaktapur Multiple Campus  
Dudhpati-02, Bhaktapur

Shape, arrow

Description automatically generated

**Certificate of Approval**

This is to certify that this project prepared by Chetan Raj Budhathoki *(20232/075), Nabin Bhandari (20247/075), and Bimal Shrestha (20227/075)* entitled “Job Recommendation System using Cosine Similarity Algorithm” in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Information Technology has been well studied. In our opinion, it is satisfactory in the scope and quality of a project for the required degree.

|  |  |
| --- | --- |
| …………………..………..…  **Saroj Maharjan**  Supervisor/Visiting Faculty  CSIT Department  Bhaktapur Multiple Campus  Dudhpati-02, Bhaktapur | …………………..………..…  **Sushant Paudel**  Department Head  CSIT Department  Bhaktapur Multiple Campus  Dudhpati-02, Bhaktapur |
| …………………..………..…  **Internal Examiner**  CSIT Department  Bhaktapur Multiple Campus  Dudhpati-2, Bhaktapur | …………………..………..…  **External Examiner**  Central Department of Computer Science and IT  Tribhuvan University  Kritipur, Nepal |

**Acknowledgment**

The success and final outcome of this assignment required a lot of guidance and assistance from many people and we feel extremely fortunate to have got all this all along the completion of our assignment work. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them. We take this opportunity to express our profound gratitude and deep regards to our supervisor **Mr. Saroj Maharjan** for his exemplary guidance, monitoring and constant encouragement throughout the course of this thesis. The blessing, help and guidance given by him from time to time shall carry us a long way in the journey of life on which we were about to embark.

We could not have completed this project without help from our college “Bhaktapur Multiple Campus” which gave us supportive hands before us academically, also regarding other activities related to Information Communication Technology as well as extracurricular activities that we got to participate in. It gave us a family-like environment.

We also take this opportunity to express a deep sense of gratitude to Mr. Sushant Poudel, coordinator, BMC College, for his cordial support, valuable information and guidance, which helped us in completing this task through various stages.

**Project Team**Chetan Raj Budhathoki  
Nabin Bhandari  
Bimal Shrestha

**Abstract**

Job searching has become increasingly challenging in recent times, with manual job portals requiring users to spend countless hours searching for job vacancies, leading to inefficiency and time wastage. To tackle this problem, this report introduces a job recommendation system that utilizes content-based filtering and cosine similarity algorithms to suggest relevant job opportunities to job seekers based on their user profile and job vacancy requirements. The system aims to improve the job search experience for both job seekers and employers by providing a personalized and efficient approach to job searching.

This report discusses the design, implementation, and evaluation of the job recommendation system. The system was developed using PHP and a MYSQL database was used to store the job listings. The user interface was designed to be user-friendly and easy to navigate. A user study was conducted to evaluate the system's effectiveness, and the results showed that the system successfully recommended relevant jobs to job seekers.

The job recommendation system has the potential to significantly impact the job market by allowing employers to better manage their resources and receive applications from interested job seekers, while also providing job seekers with a more efficient way to search for relevant job opportunities. Overall, this system is a valuable tool for job seekers and employers alike, and has the potential to transform the job search experience.

**Table of Contents**

[**List of Figures**8](#_Toc134117496)

[**List of Tables**9](#_Toc134117497)

[**List of Abbreviations**10](#_Toc134117498)

[**Chapter 1: Introduction**12](#_Toc134117499)

[**1.1**121212](#_Toc134117500)

[**1.2**131313](#_Toc134117501)

[**1.3**141414](#_Toc134117502)

[**1.4**141414](#_Toc134117503)

[**1.4.1 Scope**14](#_Toc134117504)

[**1.4.2 Limitation**14](#_Toc134117505)

[**1.5**151515](#_Toc134117506)

[**1.6**171717](#_Toc134117507)

[**1.6.1**171717](#_Toc134117508)

[**1.6.2**171717](#_Toc134117509)

[**1.6.3**171717](#_Toc134117510)

[**1.6.4**171717](#_Toc134117511)

[**1.6.5**171717](#_Toc134117512)

[**1.6.6**181818](#_Toc134117513)

[**Chapter 2: Background Study and Literature Review**19](#_Toc134117514)

[**2.1**191919](#_Toc134117515)

[**2.1.1**191919](#_Toc134117516)

[**2.1.2Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.**](#_Toc134117517)

[**2.2**202020](#_Toc134117518)

[**2.2.1Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.**](#_Toc134117519)

[**Chapter 3: System Analysis**22](#_Toc134117520)

[**3.1**222222](#_Toc134117521)

[**3.1.1**222222](#_Toc134117522)

[**3.1.2**252525](#_Toc134117523)

[**3.1.3**272727](#_Toc134117524)

[**Chapter 4: System Design**29](#_Toc134117525)

[**4.1**292929](#_Toc134117526)

[**4.2**303030](#_Toc134117527)

[**Chapter 5: Implementation and Testing**40](#_Toc134117528)

[**5.1**404040](#_Toc134117529)

[**5.1.1 Tools Used**41](#_Toc134117530)

[**5.1.2 Implementation Details of Modules**42](#_Toc134117531)

[**5.2**434343](#_Toc134117532)

[**5.2.1 Test Cases for Unit Testing**43](#_Toc134117533)

[**5.2.2 Test Cases for System Testing**43](#_Toc134117534)

[**5.3**474747](#_Toc134117535)

[**Chapter 6: Conclusion and Future Recommendations**48](#_Toc134117536)

[**6.1**484848](#_Toc134117537)

[**6.2**494949](#_Toc134117538)

[**References**50](#_Toc134117539)

# **List of Figures**

[**Figure 1 Methodology diagram of Project**15](#_Toc134117620)

[**Figure 2 Use case diagram of the systemError! Bookmark not defined.**](#_Toc134117621)

[**Figure 3 Gantt chart of project**26](#_Toc134117622)

# **List of Tables**

[**Table 1 Table of the task schedule**26](#_Toc134117579)

# **List of Abbreviations**

|  |  |
| --- | --- |
| HTML | Hyper Text Markup Language |
| CSS | ..Cascading Style Sheets |
| JS | JavaScript |
| CV | Curriculum Vitae |
| PHP | Hypertext Preprocessor |
| SQL | Structured Query Language |
| DBMS | Database Management System |
| TF-IDF | Term Frequency-inverse document frequency |
| ER | Entity Relationship |

# **CHAPTER ONE INTRODUCTION**

## **Introduction**

A Recommendation System is a subclass of information filtering system that seeks to predict the "preference" a user would give to an item. In most of the existing job portal systems, users or job seekers have to manually search for the different job vacancies and apply for those job vacancies. The Job Recommendation System has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some case, it reduces the hardships faced by the existing system. Moreover, this system is designed for particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error messages while entering invalid data. No formal knowledge is required for the user to use this system. Thus, by this all it proves that it is user friendly.

Job Recommendation System, as described above can lead to error free, reliable and fast management systems. It can assist the user to concentrate on their own other activities rather than concentrating on record keeping.

Every organization, whether big or small, has challenges to overcome and manage the information of vacancy, job, vacancy, job candidate, interview schedule. Through this system, every organization can have their own personal account, post the jobs, and view the applications sent by different job seekers for those jobs. And also, this system is very useful for those people who are seeking job opportunities. This system will ultimately allow employers to better manage resources. This system is equally advantageous for job seekers as they can manually search for the jobs in which they are interested and apply for that job. Another important feature of this system is it recommends the job seeker, a list of relevant jobs matching the content of user profile and job specification of job vacancies using content-based filtering and cosine similarity algorithm. Thus, this unique feature of the system makes it different from existing job portal system.

## **Problem statement**

In existing Job Portal Systems such as merojob.com, indeed.com, SimplyHired.com etc., job seekers have to manually search for the job vacancies that are suitable to them also some of the existing systems do not have a resume upload mechanism.

So our motive is to solve this problem by introducing the system in which job seekers can register their account and upload their resume and create their own profile based on their experience and skills. According to their profile, our system suggests the list of the jobs that are relevant to them.

Relevant job recommendations are therefore crucial for a good user experience. Here we present a method to compute the similarities between user profiles and job specification related to different jobs using cosine similarities and provide a list of jobs that are according to their similarities.

## **Objectives**

The general objective of the project is to develop a web-based application where different organizations can post job vacancies and job seekers can search and apply for the job suitable for them as well as this system will also recommend the job relevant to their job. Another objective is to build the application program to reduce the manual jobs of managing the jobs and vacancy.

## **Scope and Limitation**

### **1.4.1 Scope**

Our future plan for this system is to ensure that the website is available, accessible and reached to every Nepali job seeker and recruiter. In the near future this recommendation system will fulfill the necessity in a speedy manner.

The scope of the system can be as follow:  
• Maintain job seeker and employer records.  
• Provide customized job postings.  
• Maintain job posting details.  
• Maintain the job applications lists.

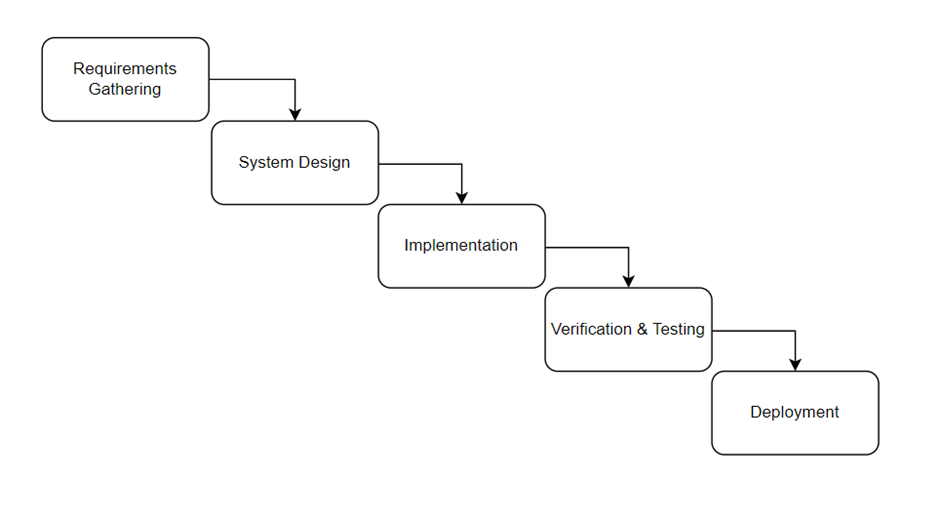
**1.4.2 Limitation**The limitations of the project are listed below:

1. Limited data: A job recommendation system relies heavily on data to generate recommendations. If there is limited data available, the system may not be able to provide accurate or relevant job recommendations.
2. Data sparsity: Cosine similarity can be limited by data sparsity, where there are not enough data points to accurately calculate similarity scores between job seekers and job requirements. This can lead to inaccurate or irrelevant job recommendations.
3. Inability to handle context: Cosine similarity is a context-free measure of similarity and may not be able to handle context-specific job requirements. For example, a job that requires a certain level of experience may be recommended to job seekers with similar skills but without the required experience.

## **Development Methodology**

Since the requirements were pretty much clear from the beginning and timeframe was allocated likewise, we found waterfall model as the best fit to this system.

Here’s how waterfall model was applied:

**Figure 1: Methodology diagram of Project**

**1. Requirements Gathering**: The first step in the waterfall model is to gather all the requirements for the job recommendation system. We tried understanding the user requirements, business goals and system constraints. Clearly defining the objectives of the system and the kind of users interacting with the system like job seekers,recruiters, understanding their specific requirements and the level of personalization needed. Data sources were identified that will be used to provide job recommendations such as job postings, candidate profiles and user behaviour data.

**2. Design:** Once the requirements were identified we designed the system architecture which involved creating a high level design for the job recommendation system where we defined the data models, algorithms, and the overall system architecture. We also considered the user interface where that will be used by users to easily interact with the system.

**3. Implementation:** In this phase, the job recommendation system was built according to the design specifications. We implemented the algorithms and data models, as well as the user interface and other features that were specified in the design phase. We also conducted thorough testing to ensure the system is functioning as intended.

**4. Testing:** In the testing phase, we conducted comprehensive testing to ensure that the job recommendation system is free of bugs and meets all the requirements. We conducted unit testing, integration testing, and system testing to ensure that the system is working as expected.

**5. Deployment:** Once the job recommendation system was thoroughly tested and validated, we deployed it in a production environment. This involved installing the system on the production servers and configuring it for use, by end-users.

**6. Maintenance:** After the job recommendation system was deployed, we have been constantly monitoring it to ensure that it's working properly. Any bugs that may arise will be fixed and necessary updates to the system will be provided to keep it up-to-date with changing user needs.

## **Report Organization**

The report is divided into seven chapters along with their descriptions, and also there include some additional sections of references and a conclusion.

### **Chapter 1: Introduction**

In this chapter, the project “Job Recommendation System” is described. This chapter discusses the questions: what is the project about, what problems does the project address, what are the goals of the project, etc.?

### **Chapter 2: Background Study and Literature Review**

In this chapter, the underlying concept and theories used in the project are discussed. As the project is to create a system (web application) As the project is to create a system (web application), that focuses on recommending jobs based on the cosine similarity between contents from different parties. Different research article done on recommending systems are mentioned and discussed.

### **Chapter 3: System Analysis**

In this chapter, the analysis of the system is done in various aspects such as requirement analysis, various feasibility analyses, and also a brief analysis of the overall system is done.

### **Chapter 4: System Design**

In this chapter, the main system design of the system is discussed and various respective system designs drawn are discussed. As the system is object-oriented based on the class, object, state, sequence, and activity diagrams, components diagrams, deployment diagrams, and also the major algorithm used in the project segment intersection, linear interpolation, and multi-layer perceptron neural network are discussed.

### **Chapter 5: Implementation and Testing**

In this chapter, how the project was conducted, what programming languages, CASE tools, and technologies were used for completing the project are discussed, and also what testing methodologies and test cases conducted during the implementation period are discussed.

### **Chapter 6: Conclusion and Recommendation**

In this chapter the summary of the overall project, what was achieved, and what future enhancement in the project can be done for the project is discussed.

# **CHAPTER TWO BACKGROUND STUDY AND LITERATURE REVIEW**

## **Background Study**

The fundamental theories, general concepts, and terminologies related to the project are discussed below:

### **Fundamental Concepts, Theories and Terminologies**

* + - 1. **Job Matching:** Job matching refers to the process of matching job seekers with job postings that best fit their skills, experience, and preferences. Job recommendation systems use a variety of techniques, such as machine learning and candidate matching algorithms, to provide accurate and relevant job recommendations.

* + - 1. **Personalization:** Personalization refers to the ability of job recommendation systems to provide personalized job recommendations based on a user's skills, experience, and preferences. Personalization can help increase user engagement and satisfaction with the system.
      2. **Collaborative Filtering:** Collaborative filtering is a technique that leverages user behavior data to identify patterns and similarities between users and recommend items that users with similar behavior have enjoyed. In the context of job recommendation systems, collaborative filtering can help identify jobs that are similar to ones that a user has previously viewed or applied for.
      3. **Content-Based Filtering:** A recommendation technique that recommends items based on user’s past behavior or interactions with the system, such as job searches, job applications, and resume uploads.
      4. **Machine Learning:** Machine learning is a type of artificial intelligence that enables computer systems to learn and improve from experience, without being explicitly programmed. Machine learning algorithms are used in job recommendation systems to analyze user data and provide personalized job recommendations.
      5. **Candidate Matching Algorithms:** Candidate matching algorithms are used in job recommendation systems to match job seekers with job postings based on their skills, experience, and preferences. These algorithms use a variety of techniques, such as collaborative filtering and content-based filtering, to provide accurate and relevant job recommendations.

Overall, job recommendation systems rely on these fundamental theories to analyze user data and provide personalized job recommendations that match a user's skills, experience, and preferences.

## **Literature Review**

Recommender Systems have become an important research field since the emergence of the first paper on collaborative filtering in the mid-1990s. In general, these systems are stated as the support systems which help users to find content, products, or services (such as books, movies, music, TV programs, and websites). By gathering and examining suggestions from other users, which means reviews from various establishments, and users.

These systems are broadly classified into collaborative filtering (CF) and content-based filtering (CB). Content-Based recommender system [1] tries to guess the features or behavior of a user given the item’s features, he/she reacts positively to.

In 2008 [2] Kleanthi Lakiotaki, Stelios Tsafarakis, and Nikolaos Matsatsinis proposed UTA-Rec. UTARec is a Recommender System that incorporates Multiple Criteria Analysis methodologies. The system’s performance and capability of addressing certain shortfalls of existing Recommender Systems is demonstrated in the case of movie recommendations. UTARec’s accuracy is measured in terms of Kendall’s tau and ROC curve analysis and is also compared to a Multiple Rating Collaborative Filtering (MRCF) approach. Juan A.

Mucheol Kim & Sang Oh Park [3] in 2011 proposed an intelligent movie recommender system with a social trust model. The proposed system is based on a social network for analyzing social relationships between users and generating group affinity values with user profiles. In experiments, the performance of this system is evaluated with precision-recall and F-measures.

In 2012 Punam Bedi, Ravish Sharma [4] proposed a Trust based Ant Recommender System (TARS) that produces valuable recommendations by incorporating a notion of dynamic trust between users and selecting a small and best neighborhood based on the biological metaphor of ant colonies.

Along with the predicted ratings, displaying additional information for explanation of recommendations regarding the strength and level of connectedness in the trust graph from where recommendations are generated, items and number of neighbors involved in predicting ratings can help active users make better decisions. Also, new users can highly benefit from pheromone updating strategy known from ant algorithms as positive feedback in the form of aggregated dynamic trust pheromone defines ‘‘popularity’’ of a user as recommender over a period of time. The performance of TARS is evaluated using two datasets of different sparsity levels viz. Jester dataset and Movie Lens dataset (available online) and compared with traditional Collaborative Filtering based apps convince users about the product. It is an agent-based recommender system. It combines a hybrid recommender system with automated argumentation.

Recommender system is one of the applications which is being used by many vectors and online service providers to believe the necessity of online users. Thus, the recommender system is presented as an intelligent system, which identifies the user category on the basis of user information and then user interest analysis. Once such information is obtained, in the second stage, the analysis is performed to obtain the similarity group respective to necessity products and services. To perform such an analysis there are some existing techniques such as content based as well as collaborative recommender systems.

# **CHAPTER THREE SYSTEM ANALYSIS**

## **System Analysis**

System analysis was conducted to investigate the system and its components for effective goal determination of the system. In the system analysis, we conducted a requirement analysis of the project to gather functional and non-functional requirements to get a clear idea of the functionalities needed for the system, also feasibility study of the project was conducted to determine whether the project is possible or not before starting it and lastly after conducting the feasibility study basic system analysis was done where we drew system analysis diagrams.

### **Requirement Analysis**

Firstly, during the system analysis phase, we first conducted the requirement analysis of the project where the functional and non-functional requirements were listed down. Proper requirement analysis was conducted to get a clear picture of what system functionalities are required and no missing functionalities arise during the middle of the project.

* + - 1. **Functional Requirements**

Functional requirements are the specific features or capabilities that a system must have in order to fulfill its intended purpose. In the context of a job recommendation system, some possible functional requirements might include:

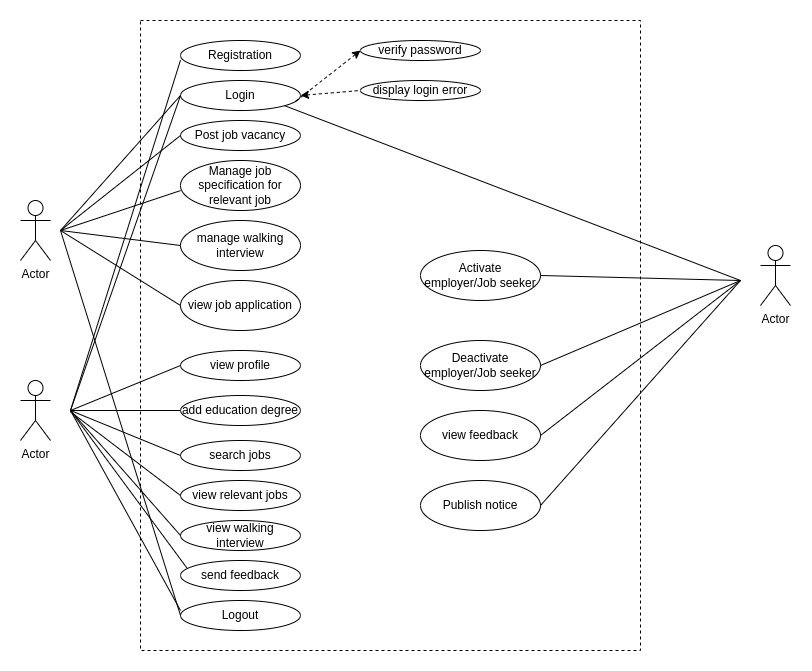
1. User Registration: The system should allow job seekers to create an account and fill out a profile that includes personal information, work experience, education, and skills.

2. Job Search: The system should allow job seekers to search for job opportunities based on keywords and job type.

3. Job Matching: The system should use algorithms and data analysis techniques to match the job seeker’s qualifications with the job requirements provided by the employers.

4. Employer Registration and Job Posting: The system should allow employers to register and create a profile that includes information about their company and job opportunities. Employers should be able to post job openings, including job requirements, qualifications, and job descriptions.

5. Feedback and Evaluation: The system should allow job seekers and employers to provide feedback and evaluate the job recommendations and job placements.



**Figure: 2 Use case diagram of the system**

#### **Non-Functional Requirements**

Non-functional requirements are not directly related to the functions performed by the system. Simultaneously, non-functional requirements may relate not only to the software system itself: some may relate to the technological process of creating software.

1. Usability: The system should be easy to use and intuitive for both job seekers and employers, with clear navigation and user-friendly interfaces.

2. Performance: The system should provide fast response times and be able to handle a high volume of traffic and user data.

3. Security: The system should be secure, protect user data, and prevent unauthorized access or breaches.

4. Scalability: The system should be able to handle a growing number of users and job opportunities without sacrificing performance or user experience.

5. Reliability: The system should be reliable, available 24/7, and minimize downtime or system failures.

6. Accessibility: The system should be accessible to all users, including those with disabilities, and support multiple languages.

7. Compatibility: The system should be compatible with multiple web browsers, operating systems, and devices.

8. Compliance: The system should comply with Nepalese labor laws and regulations and provide job recommendations that comply with them.

9. Maintainability: The system should be easy to maintain and upgrade, with clear documentation and efficient code.

### **Feasibility Analysis**

After the proper requirement gathering was done, we conducted the feasibility analysis of the project to identify whether the project is feasible in every aspect and whether the project is possible to proceed in each aspect before starting the major work of the project. A feasibility study is an evaluation of a proposed project or system to determine whether it is practical, viable, and worth pursuing. In the context of a job recommendation system, a feasibility study might consider a range of factors, including technical, economic, operational, and legal considerations. Various feasibility analyses are discussed below:

#### **Technical**

From a technical perspective, a job recommendation system is feasible as long as the necessary technologies and infrastructure are available or can be developed. Since this system is able to handle large amounts of data and traffic, use similarity algorithms and data analysis techniques, and provide fast response times and reliable performance, this is technically feasible.

#### **Operational**

Operational feasibility refers to whether the job recommendation system can be integrated into the existing business processes and workflows. The system is easy to use and intuitive for both job seekers and employers, and provide support and training as needed. The system cab also be made compatible with other relevant systems or applications, such as applicant tracking systems or social media platforms.

#### **Economic**

The economic feasibility of a job recommendation system depends on the cost-benefit analysis. The system development and implementation costs, maintenance costs, and other associated costs should be justified by the benefits that the system provides. The benefits may include improved job seeker-employer matching, increased efficiency and productivity, and reduced recruitment costs for employers. This system can at least self-sustain and is economically feasible.

#### **Feasibility**

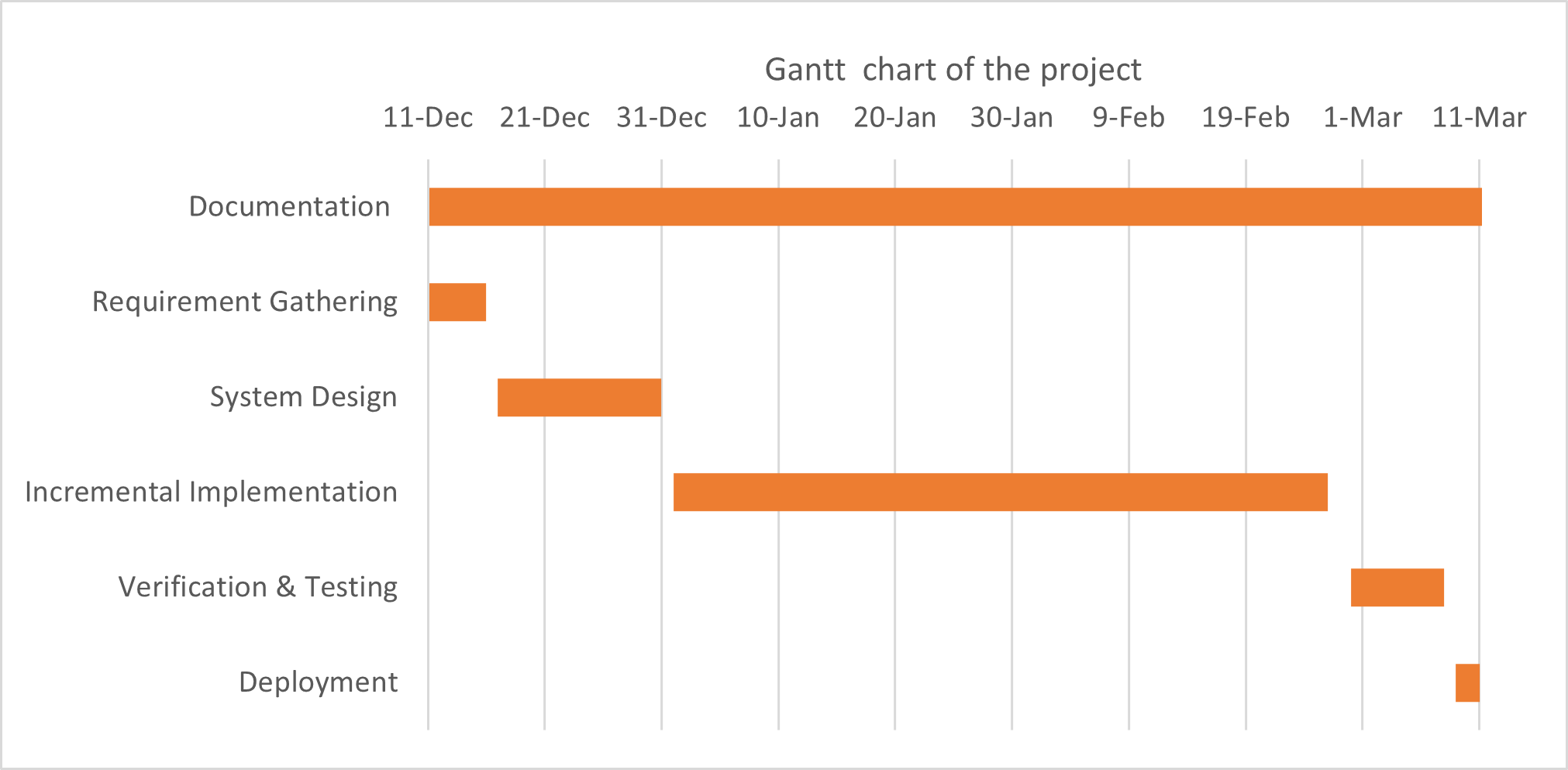
Since this recommendation system comply with Nepalese labor laws and regulations, including data protection and privacy laws, and provide job recommendations that comply with them, it can be seem to be feasible legally as well. Legal feasibility also includes ensuring that the system does not discriminate against any specific group of people based on race, gender, religion, or other characteristics.

Overall, a job recommendation system is feasible as long as the necessary resources are available, and the system meets the technical, economic, legal, and operational requirements. The benefits of such a system can be significant, providing a more efficient and effective way to match job seekers with suitable job opportunities.

The schedule of the project was analyzed by drawing the Gantt chart through Microsoft Excel.

**Table 1 Table of the task schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Start Date** | **Duration (days)** | **End Date** |
| Documentation | 11-Dec | 92 | 13-Mar |
| Requirement Gathering | 11-Dec | 5 | 16-Dec |
| System Design | 17-Dec | 14 | 31-Dec |
| Incremental Implementation | 1-Jan | 56 | 26-Feb |
| Verification & Testing | 28-Feb | 8 | 8-Mar |
| Deployment | 9-Mar | 2 | 11-Mar |



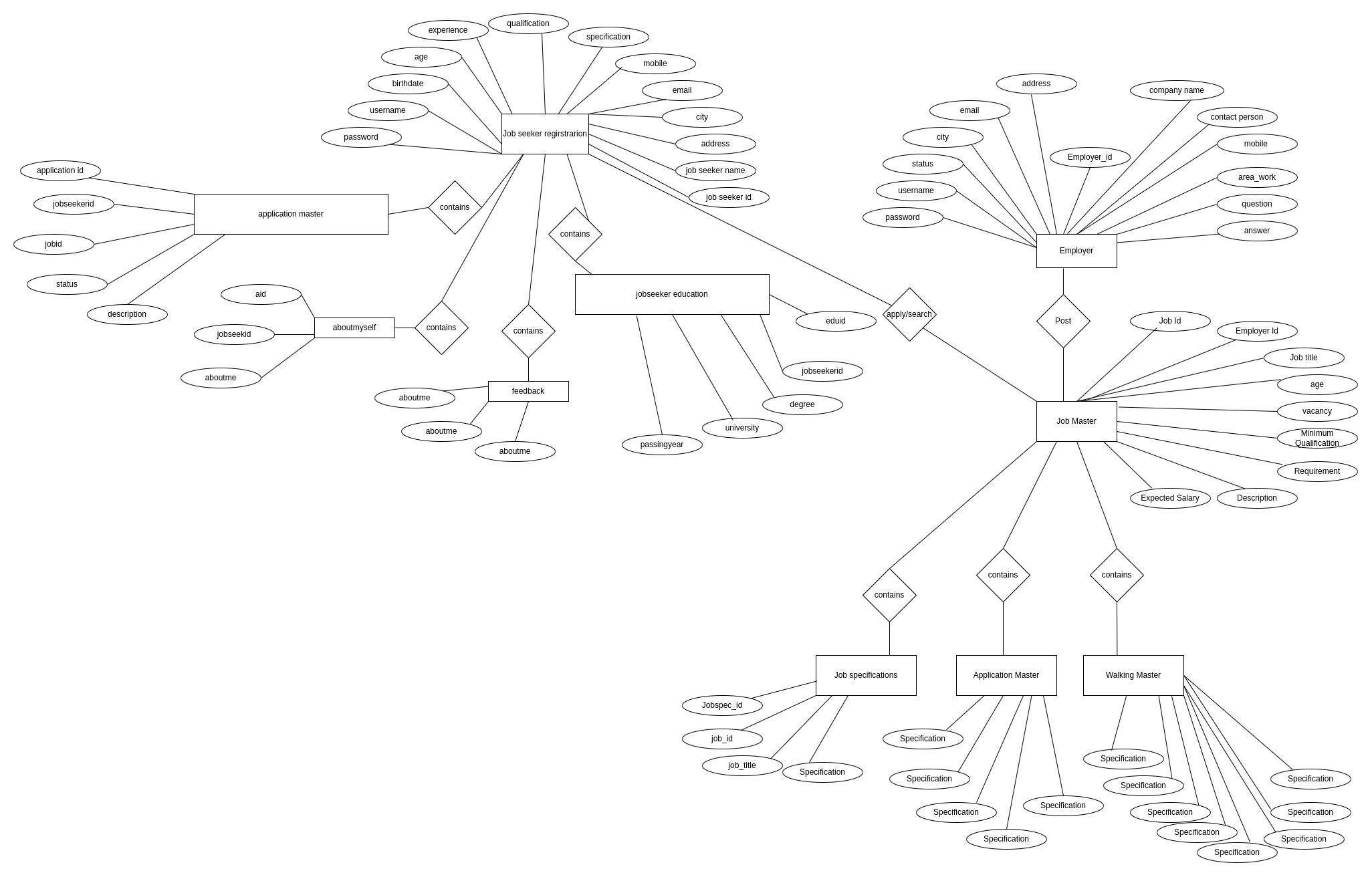
**Figure 2 Gantt chart of the project**

While analyzing the Gantt chart of the project it was found that the project can be completed on a fixed schedule so the project was found feasible on the time aspect.

### **Analysis**

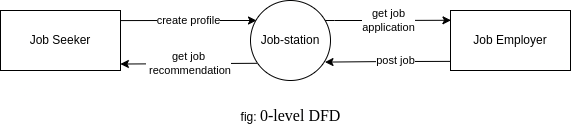
After we conducted the feasibility study, we identified that the project was feasible from every aspect and there may not arise any major hindrance during the execution of the project. After we got the assurance that the project was feasible, we conducted a thorough system analysis of the project before starting the implementation of the project. When conducting the system analysis, the team decided to implement the project through structured approach though Object Oriented is applied for the algorithm part. Also in the analysis period, various system diagrams like ER Diagram, Use case Diagram and DFD Diagram were sketched by the team to analyze the system to be developed.

#### **Data modeling(ER Diagram)**

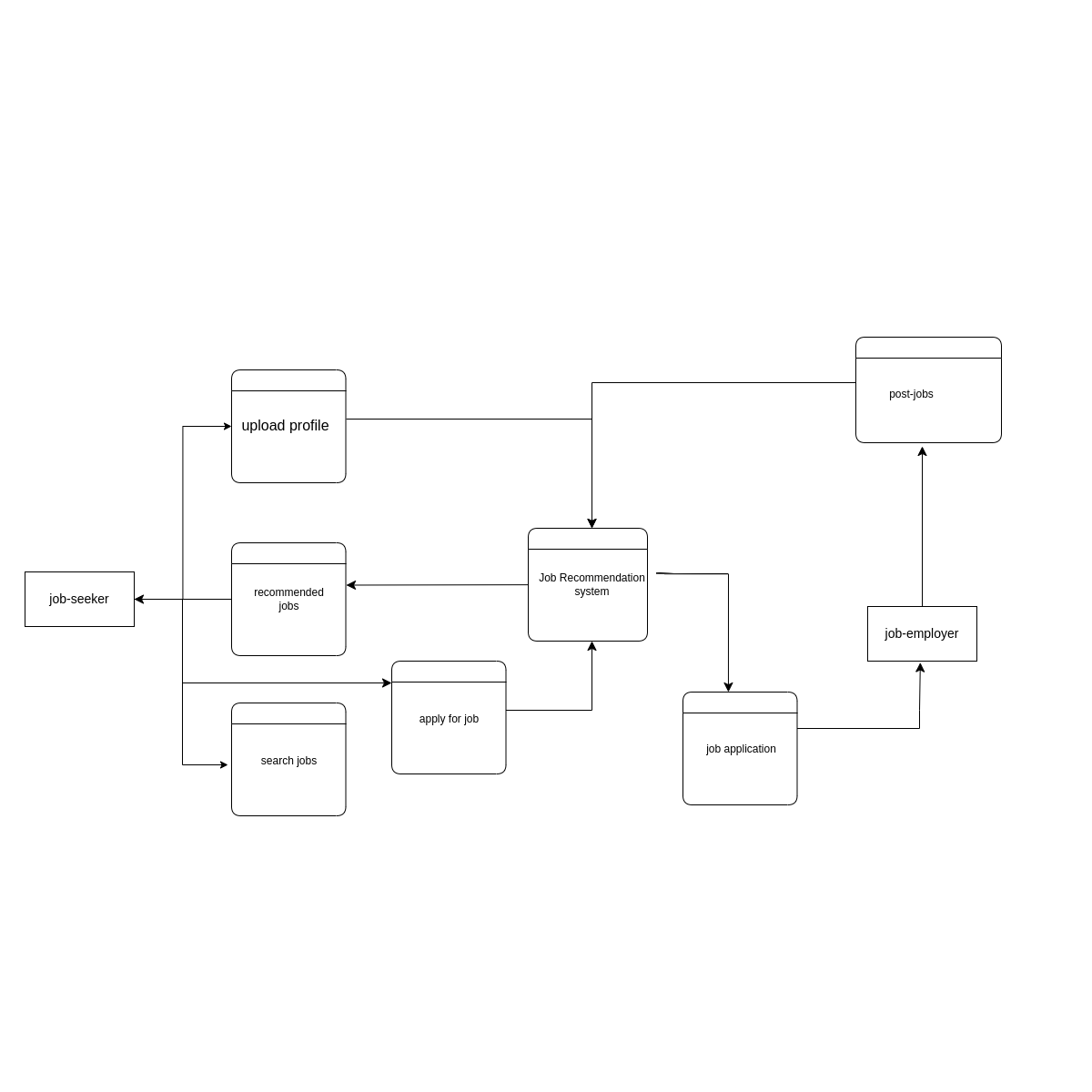


**Figure 3 . ER Diagram**

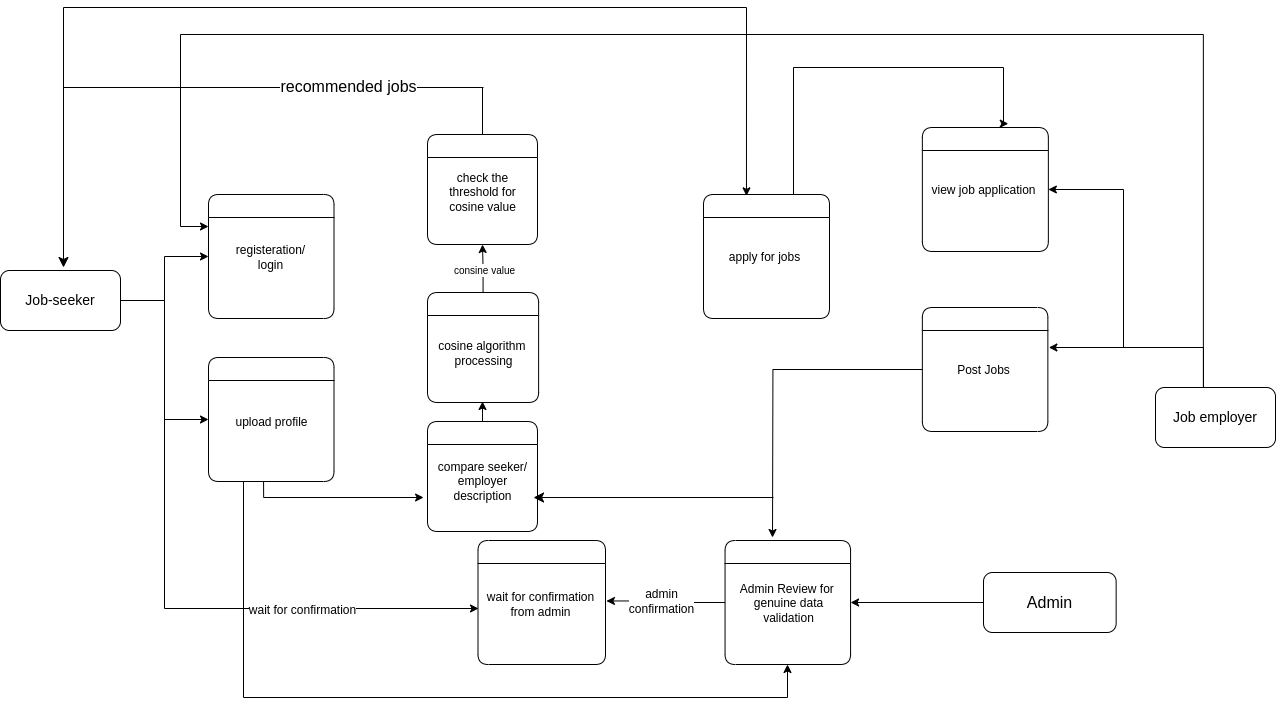
* + - 1. **Process modeling using DFD**



**Figure 4. 0-level DFD**



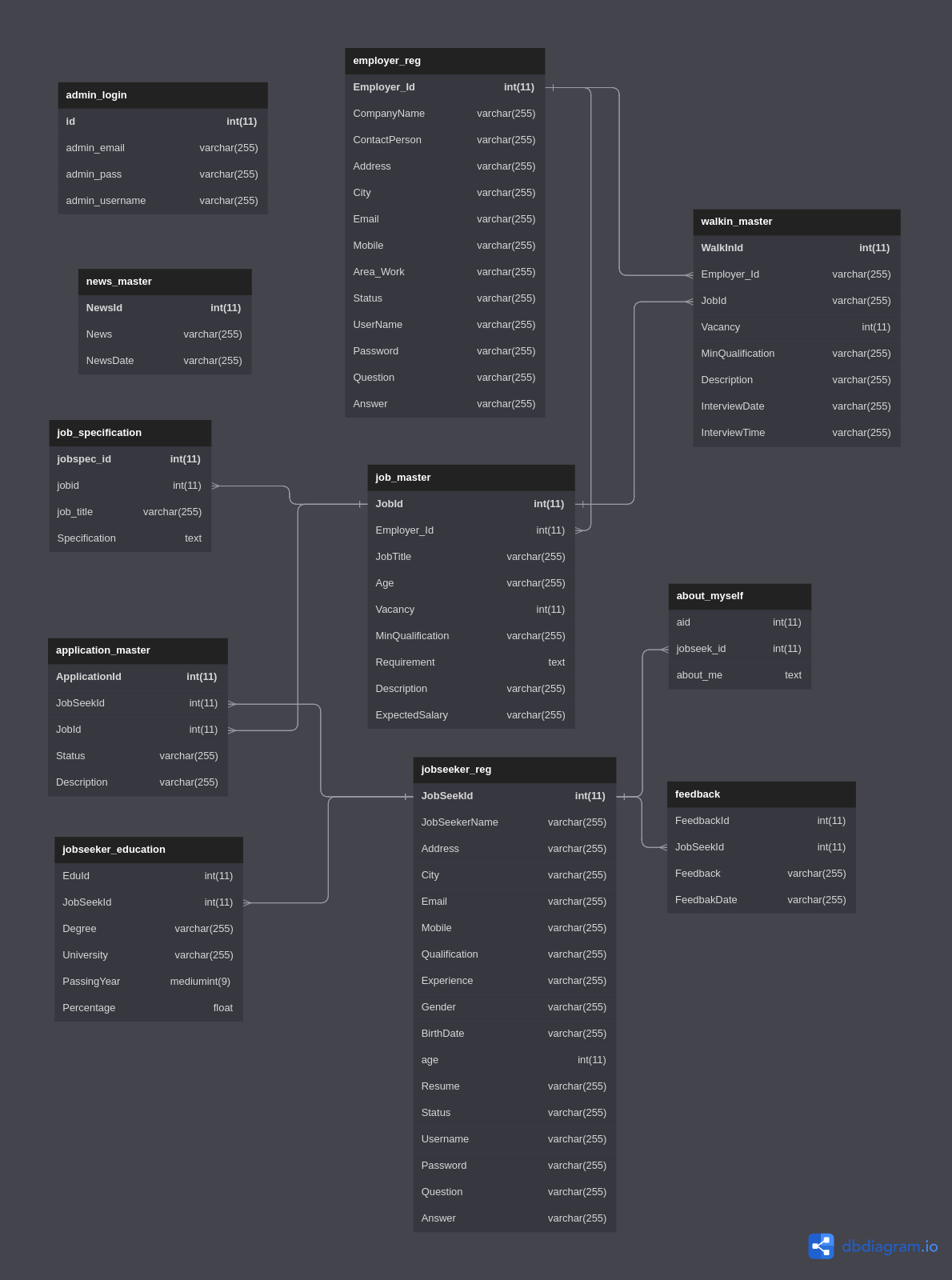
**Figure 5. 1-level DFD**



**Figure 6. 2-level DFD**

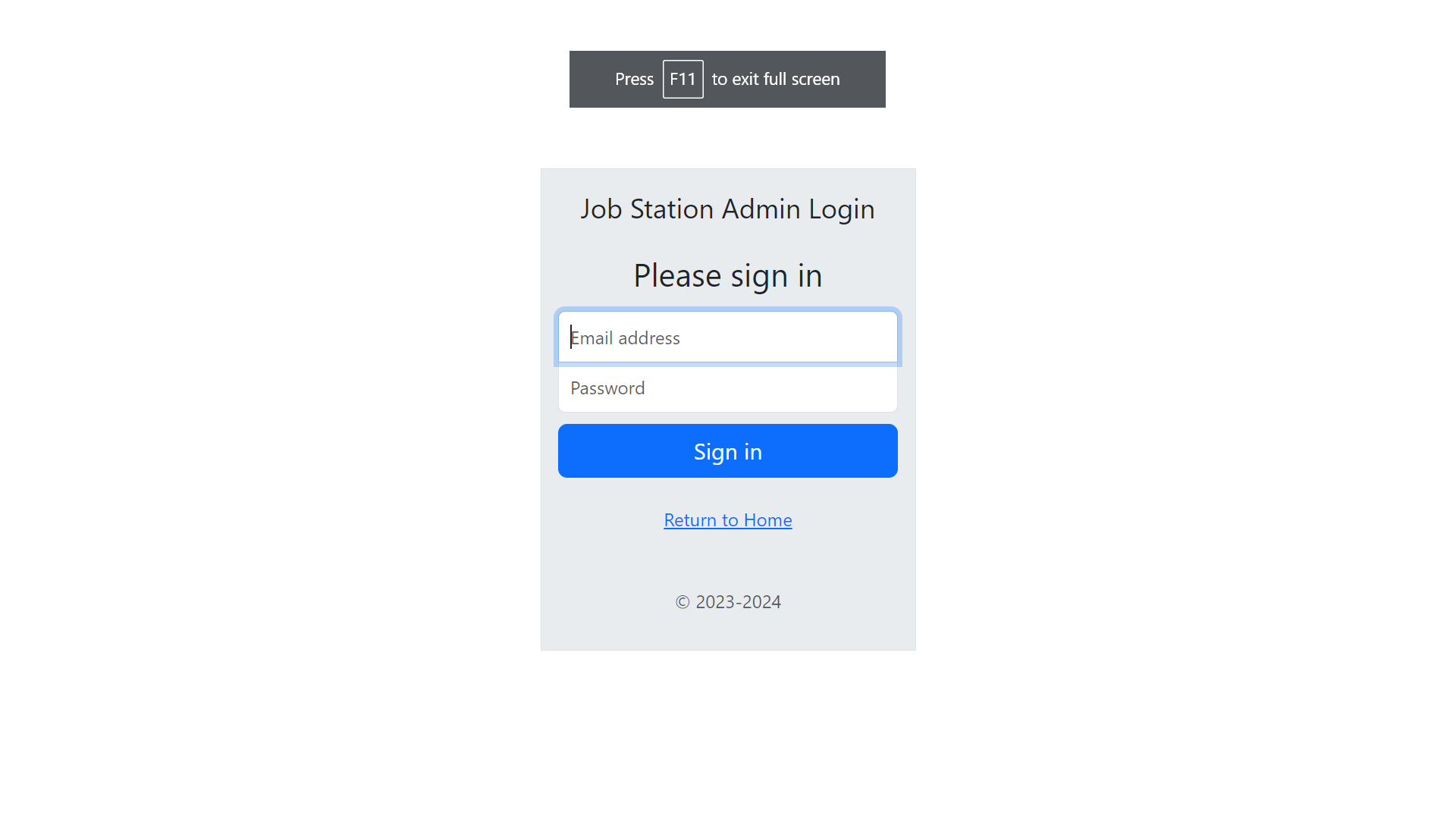
**CHAPTER FOUR  
SYSTEM DESIGN**

## **Design**

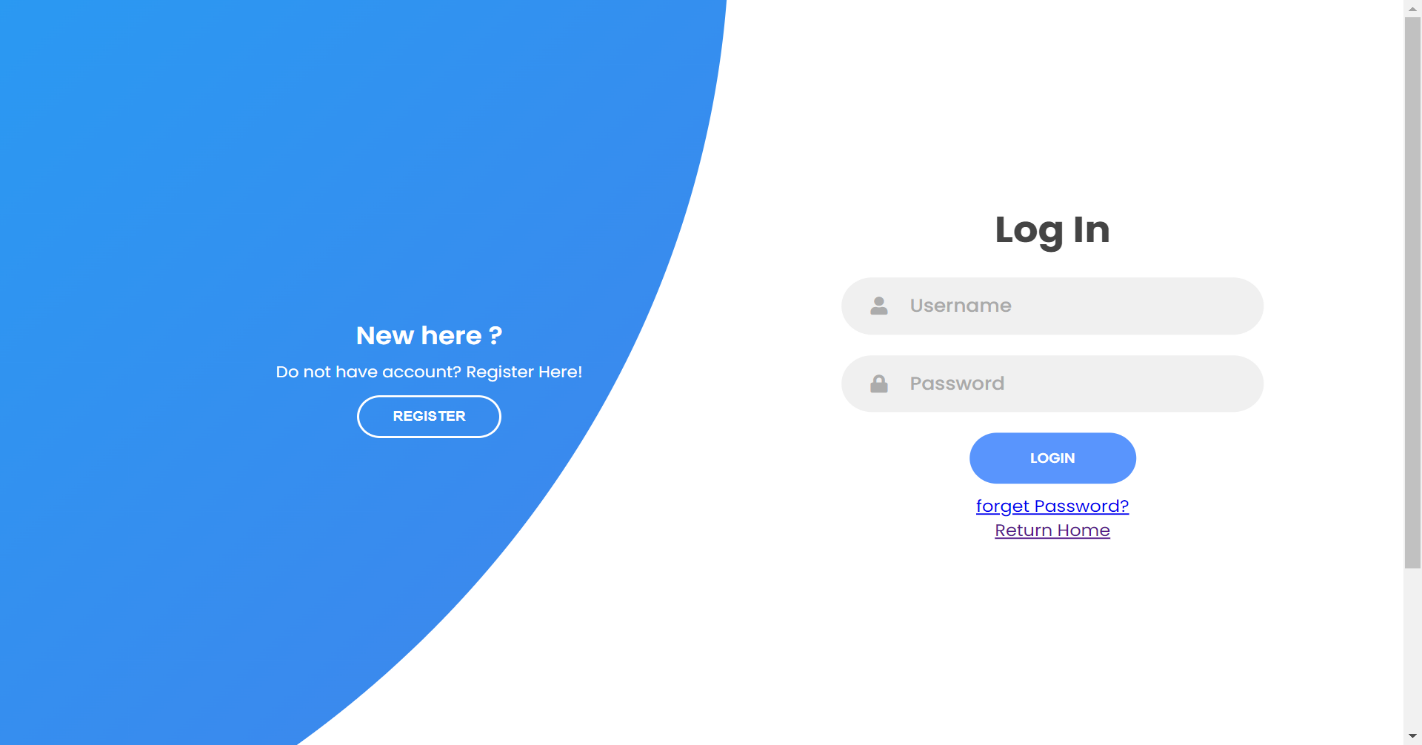
**4.1.1 Database Design**

**Figure 7. Database Design**

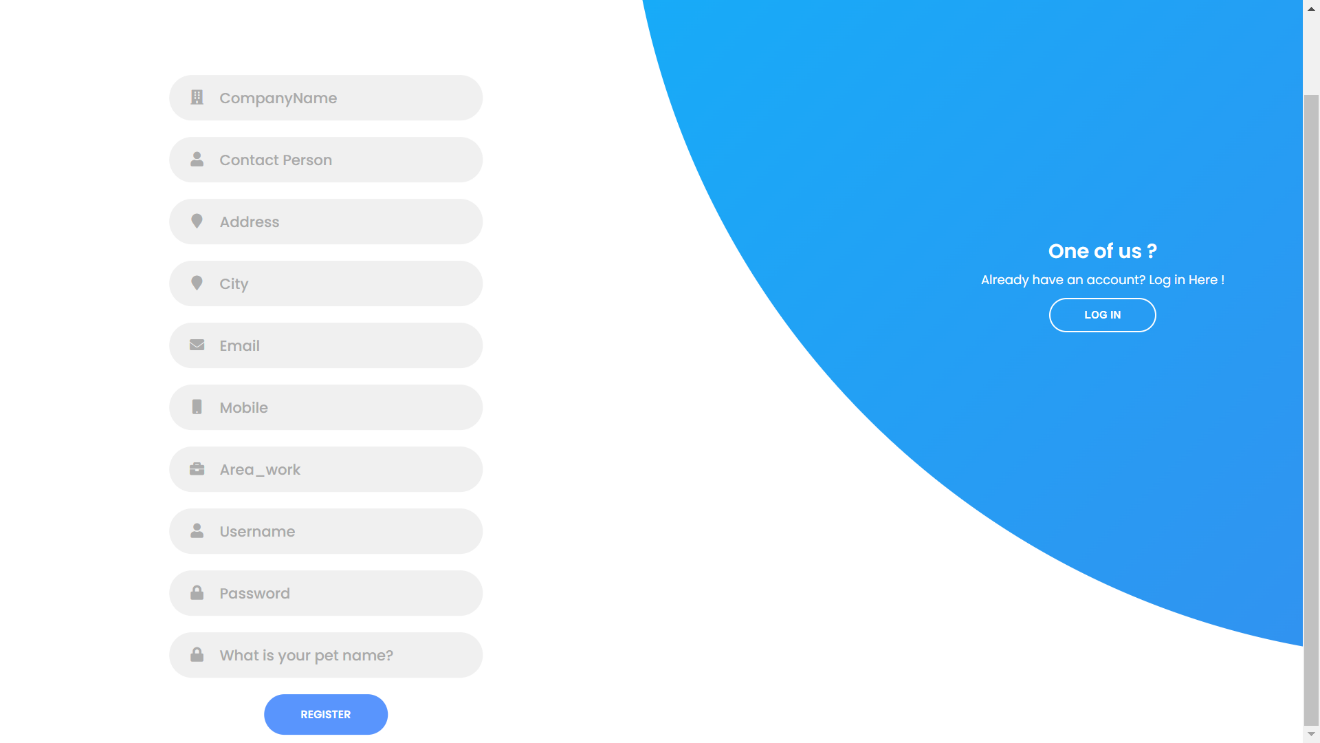
**4.1.2 Forms and Report Design**

****

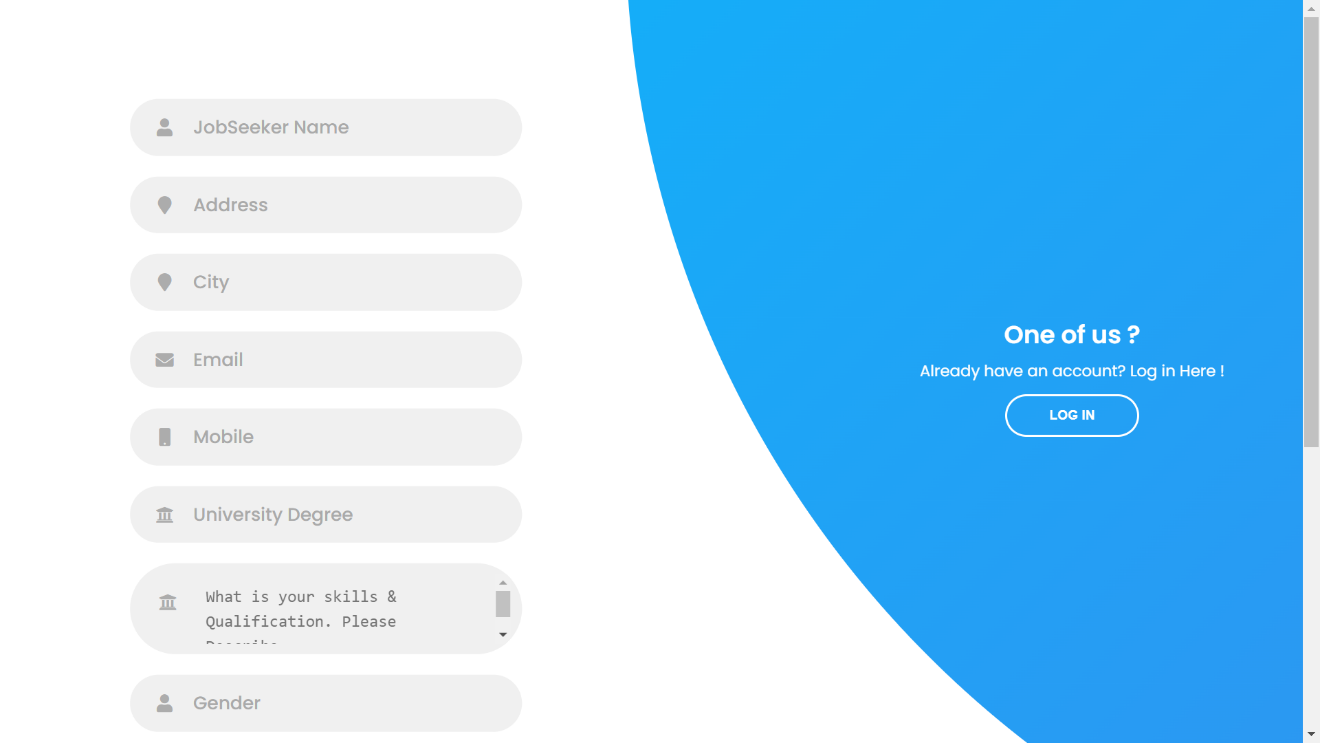
**Figure 8. Admin Login Form**

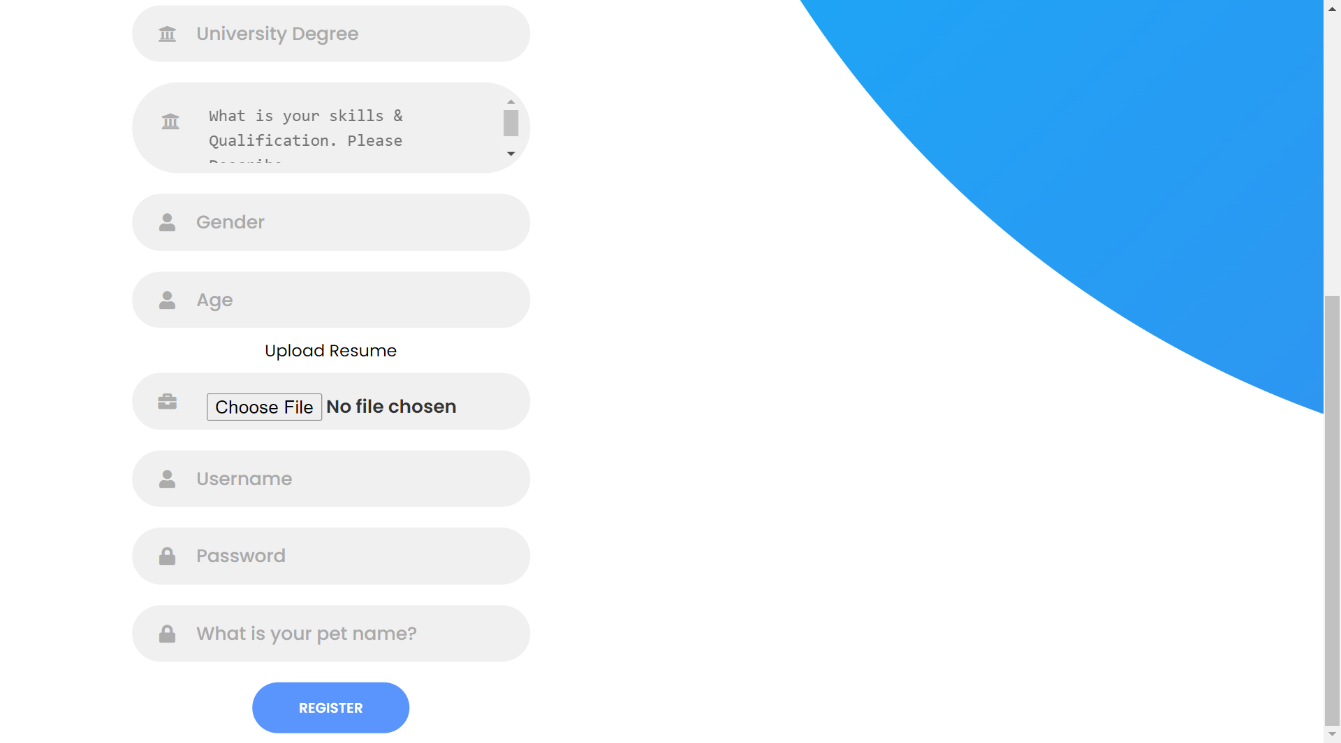
****

**Figure 8. Employer Login Form**

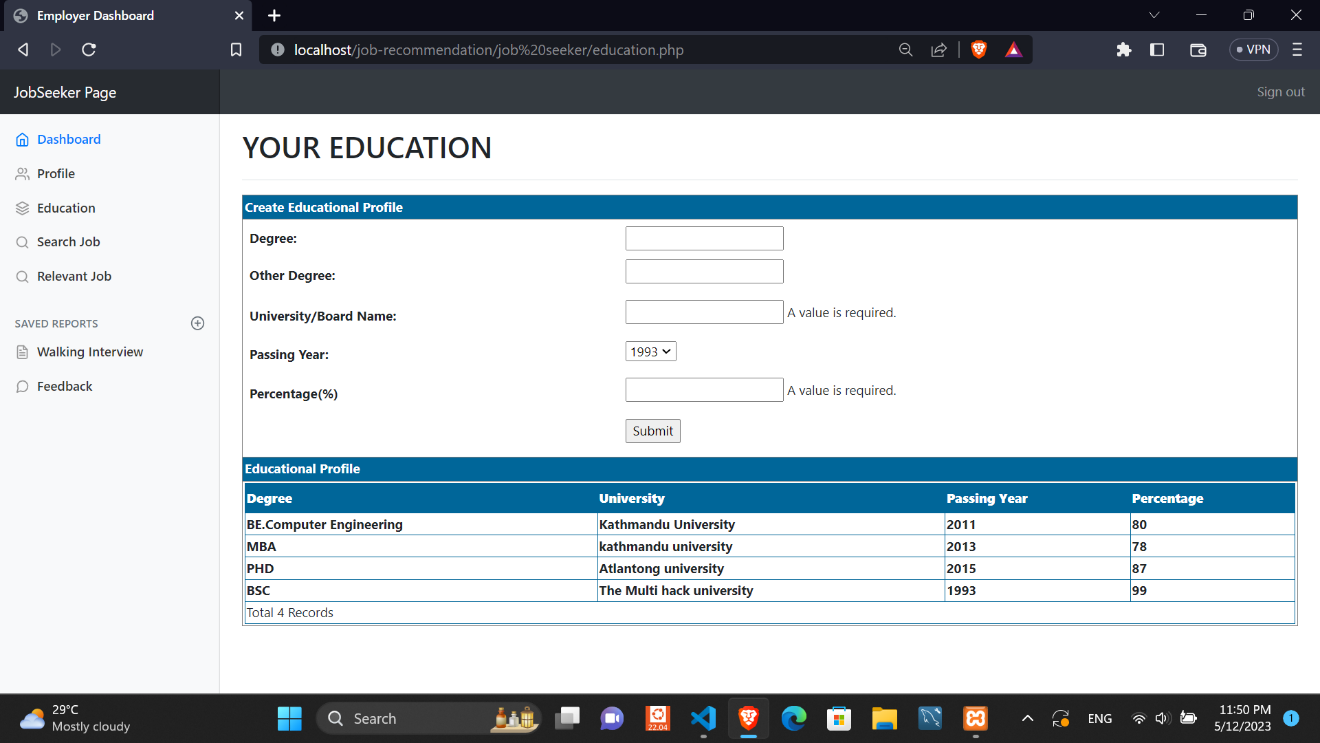
****

**Figure 8. Employer Registration Form**

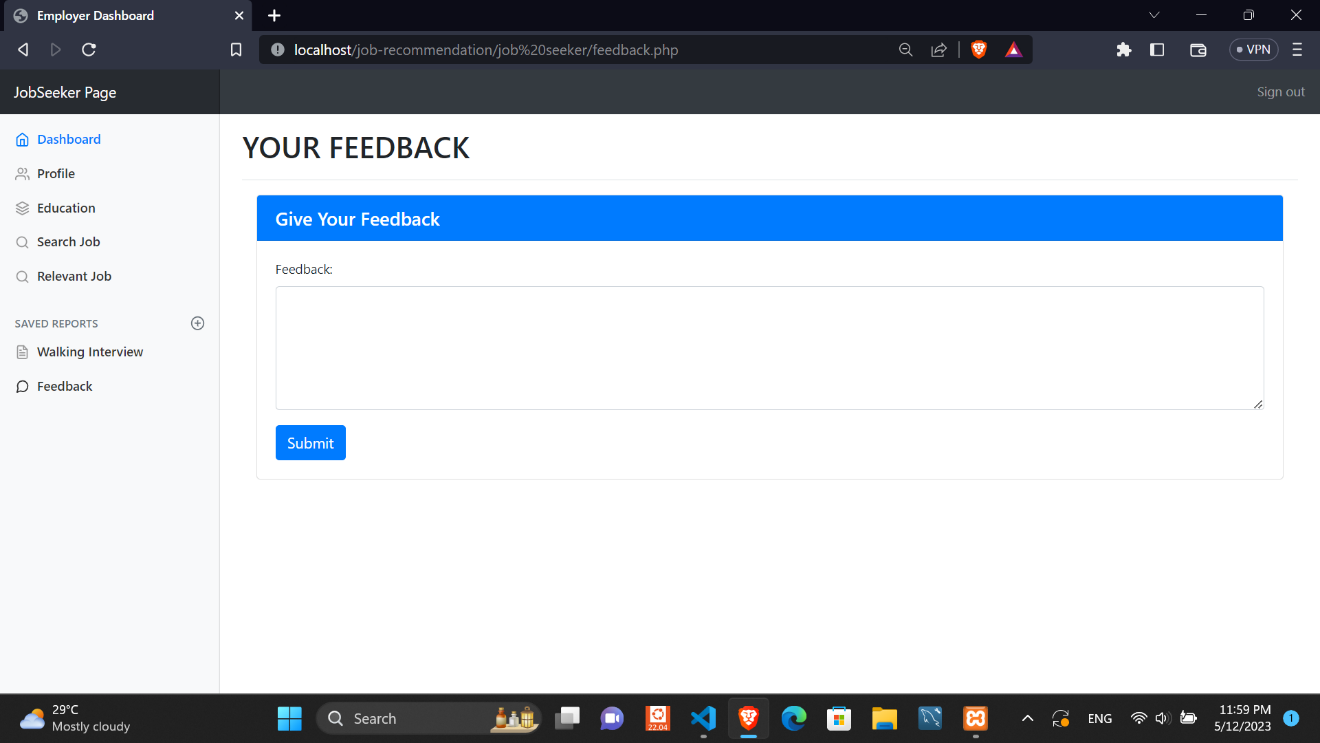
****

****

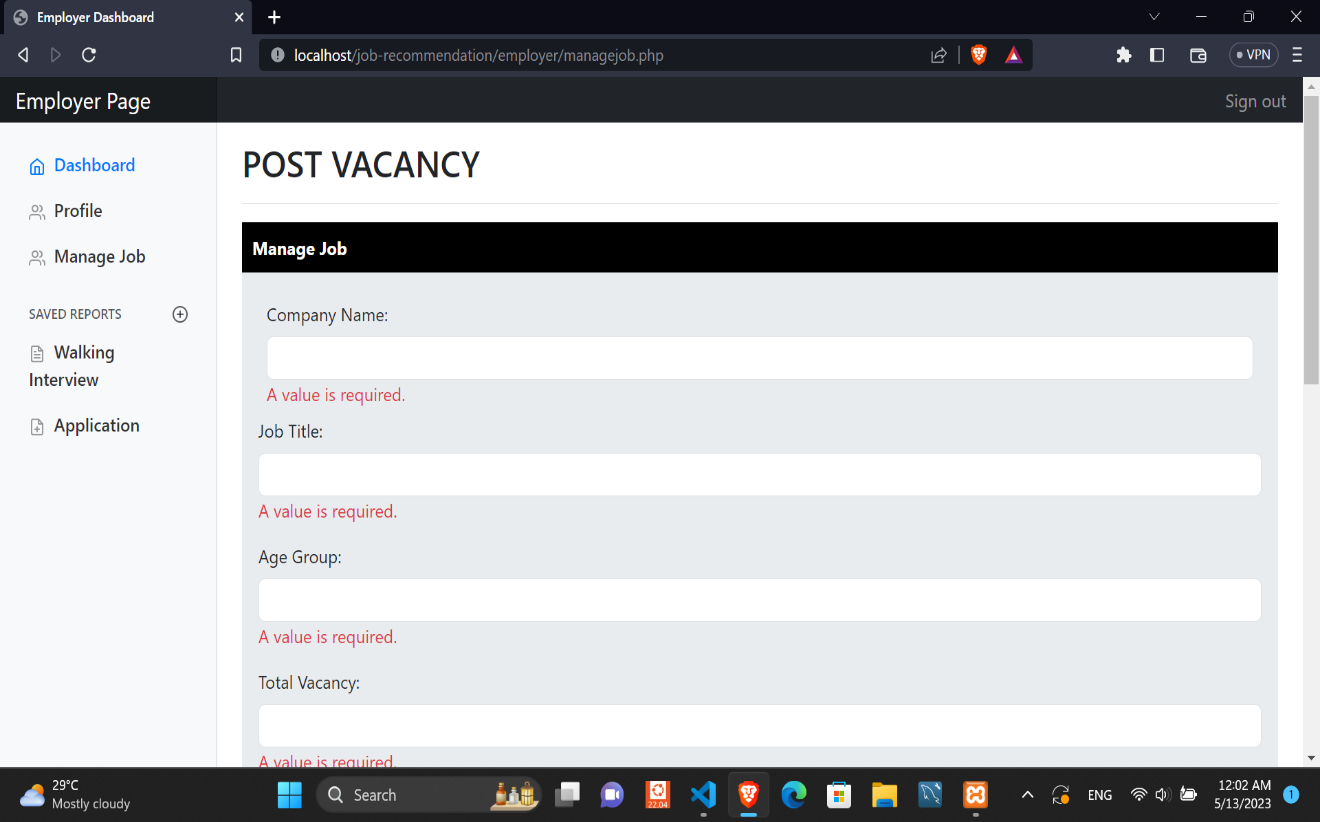
**Figure 9. Job Seeker Registration Form**

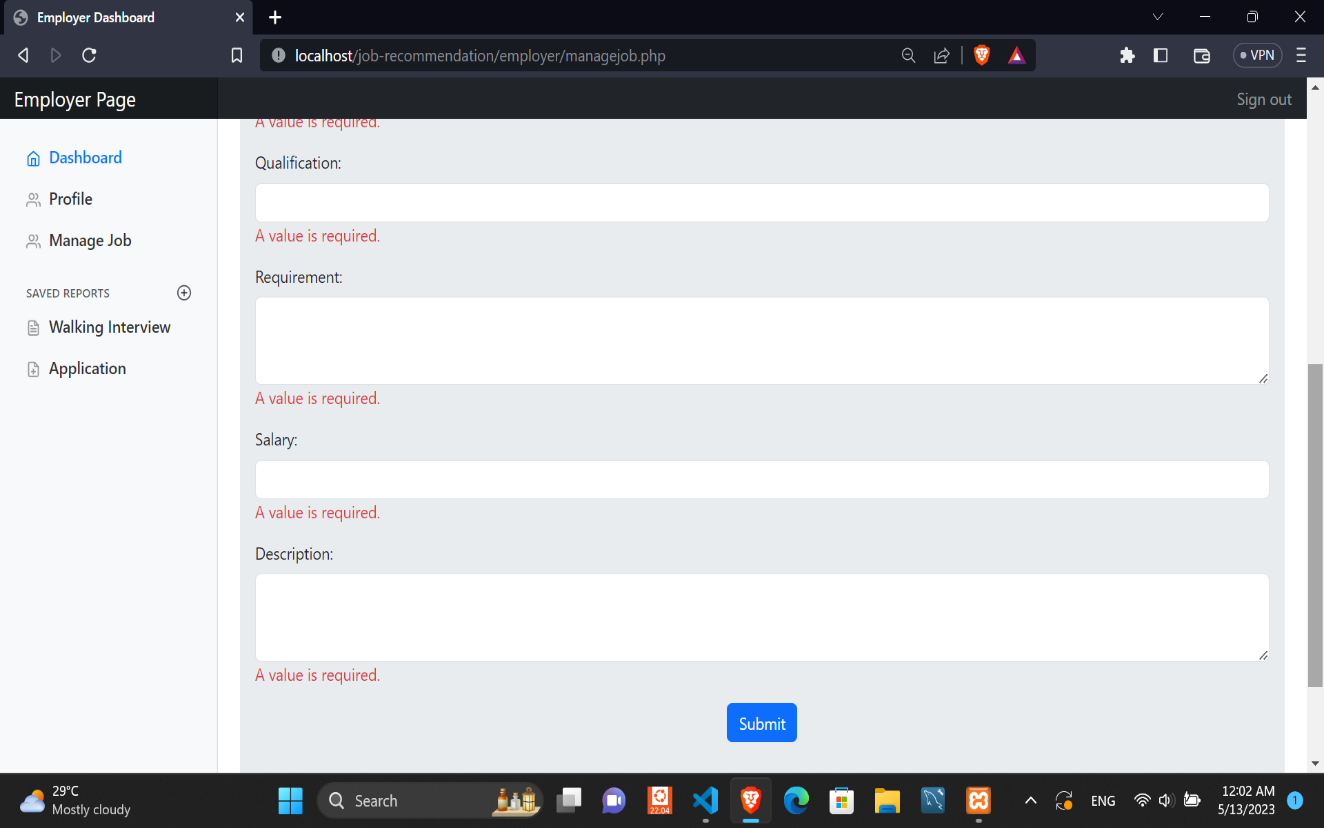
****

**Figure 9. Job Seeker Education Form**

****

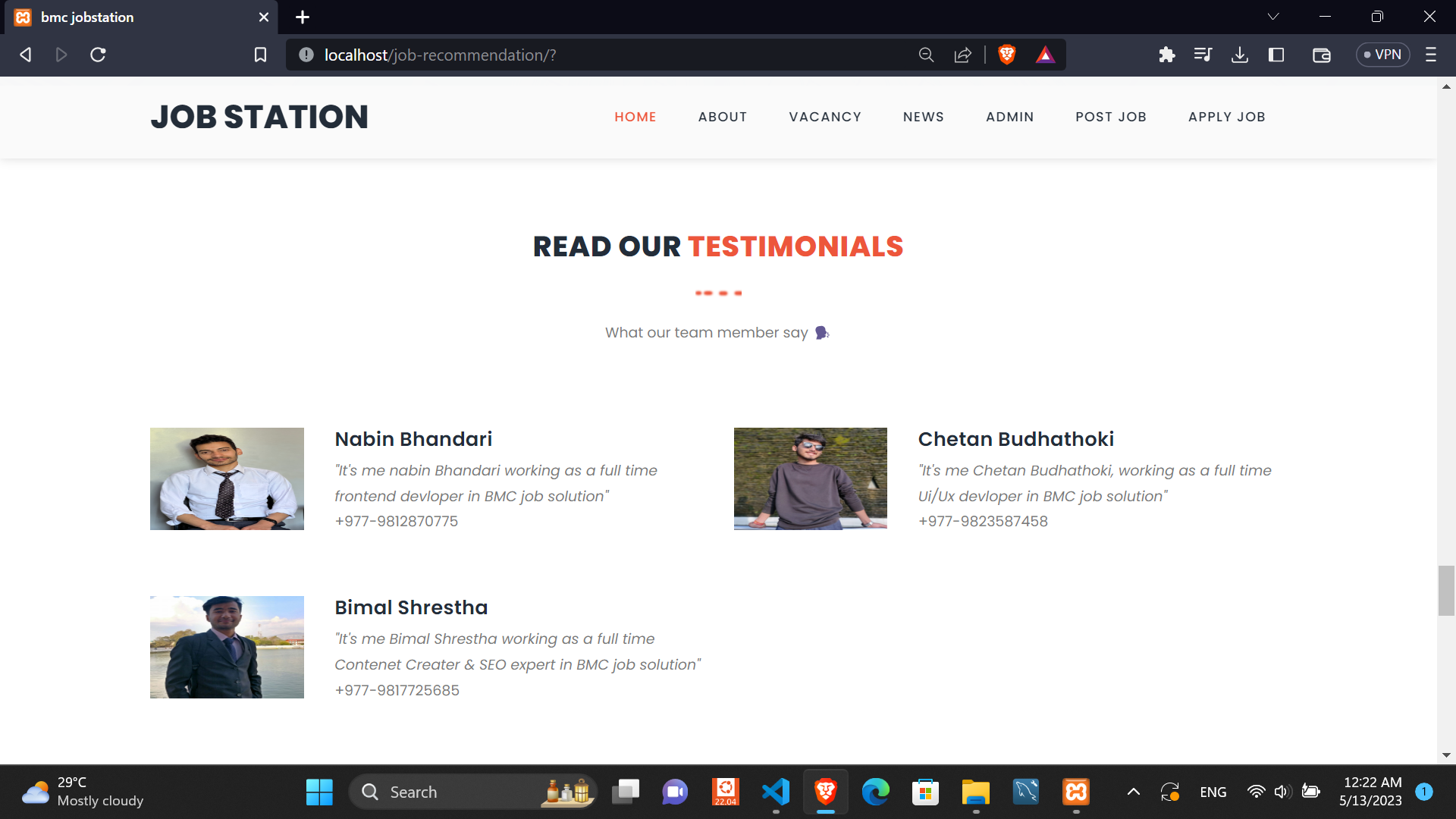
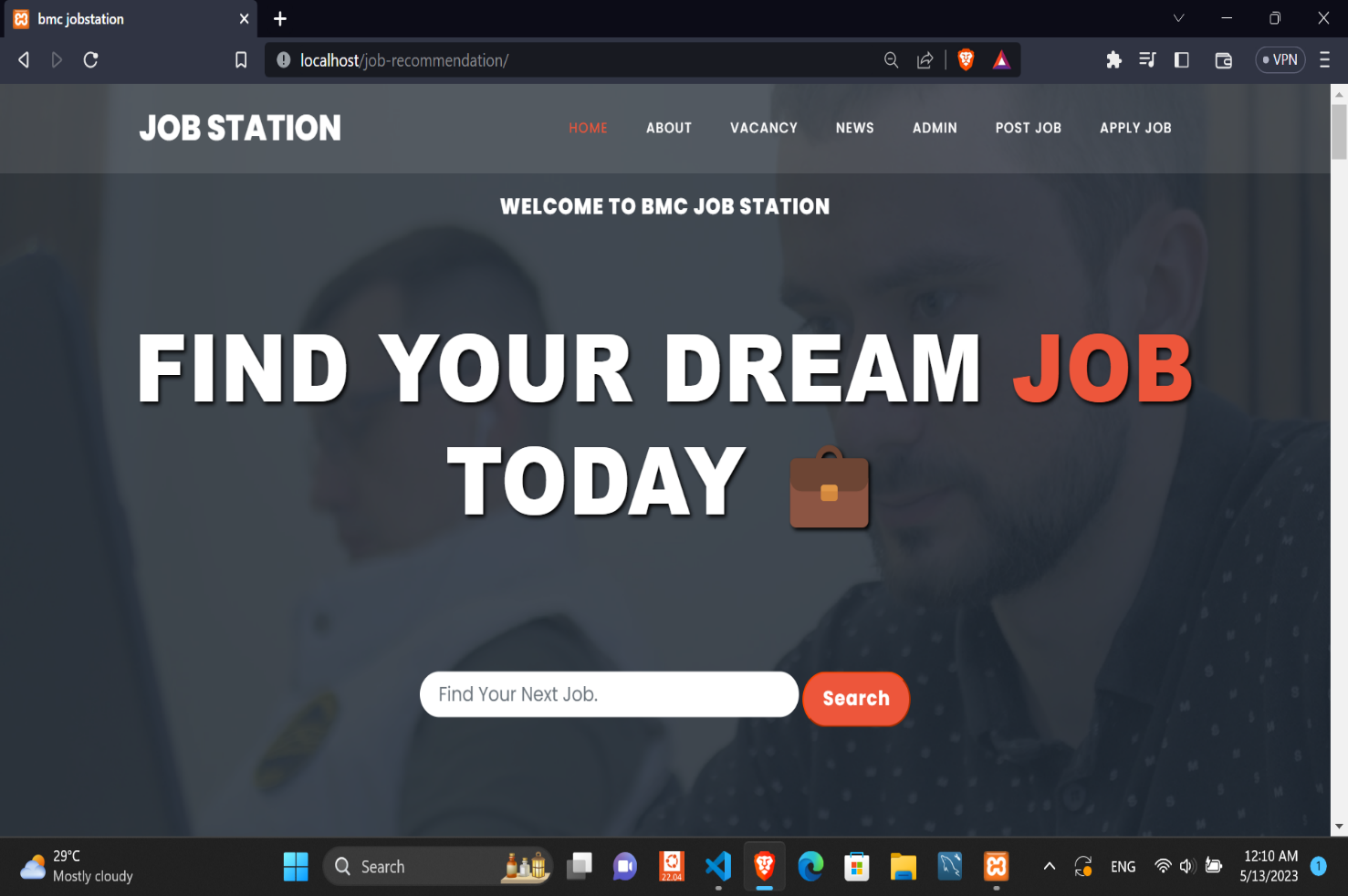
**Figure 9. Job Seeker Feedback Form**

****

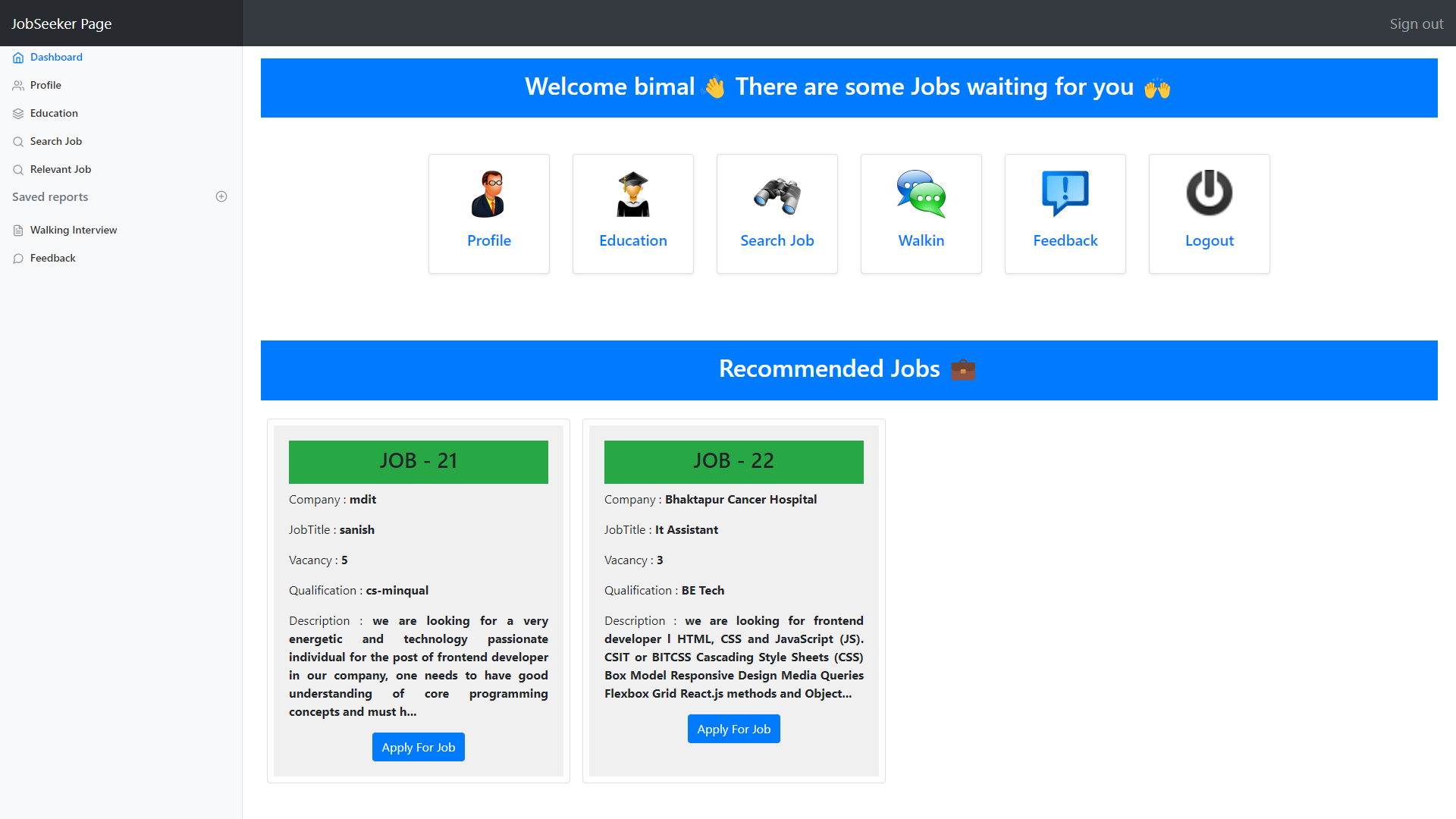
****

**Figure 10. Job Employer new Job post Form**

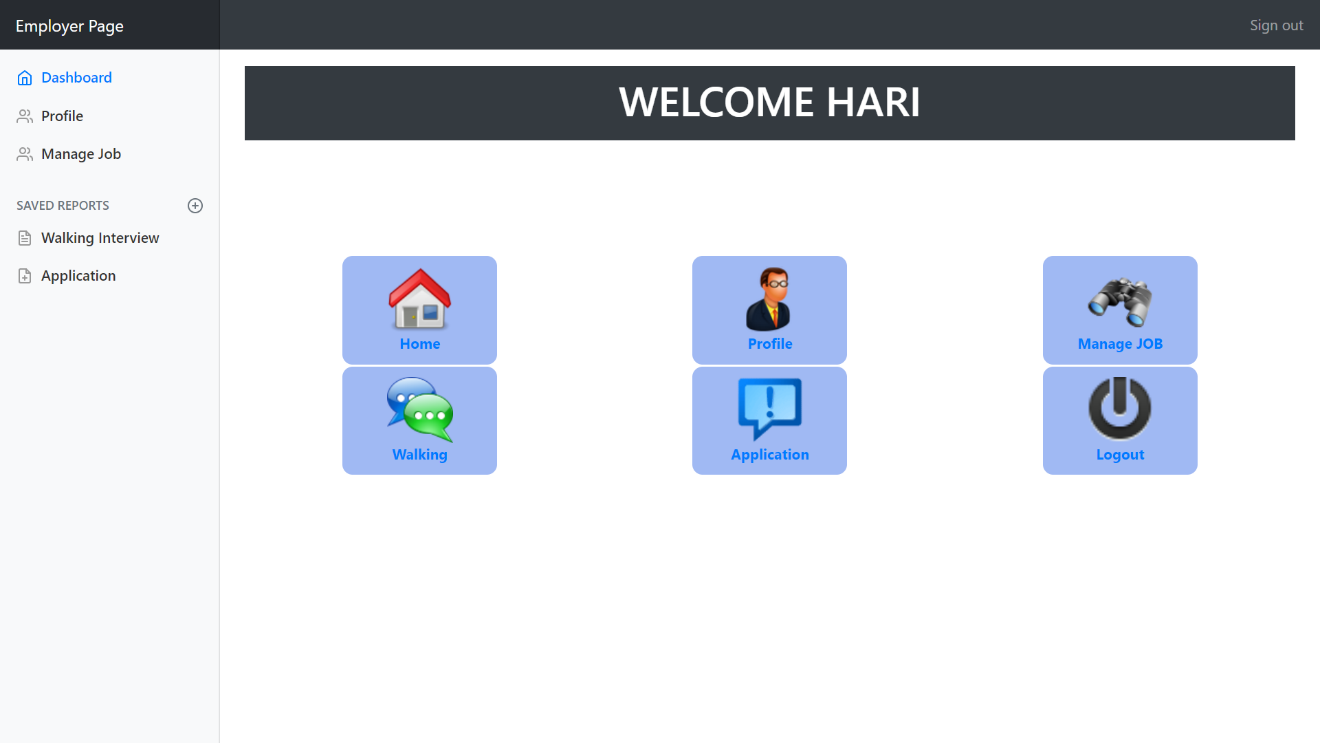
**4.1.3 Interface Design**

****

**Figure 10. Home page**

****

**Figure 11. JobSeeker Hompage**

****

**Figure 11. Employer Hompage**

* 1. **Algorithm Details**

**Cosine Similarity**

This concept is used in this project to calculate the score between two document vectors. The similarity between two vectors is defined by the angle between them. If the angle between two vectors are zero and thus cosine=1, representing the perfect match. If those two vectors are perfectly dissimilar, then the angle between the vectors is perfect 90 degree and cosine=0, representing the perfect dissimilar. The inputs required for cosine similarity calculation must be represented in the form of a Vector Space Model. A Vector Space Model is the mathematical structure formed by a collection of vectors.

In this recommendation system we have implemented the cosine similarity between job seeker’s bio data and job specifications that is related to different jobs. For calculating cosine similarity, we have to convert these two texts (bio, specification) into document vectors. In order to obtain document vectors, we use the concept of term frequency.

## **4.2.1 Term Frequency**

Term frequency is a measure of how often a term is found in the collection of documents. A reasonable scoring mechanism is computed with document terms. It counts the frequency of the terms that match between the query terms and the document term list which is denoted by tf(t, d).

## **4.2.2 Calculation Process**

Here we consider,  
text1=bio data  
text2= job specification

Using the explode function where the delimiter is space.  
$text1 = Computer Engineer BscCSIT HTML CSS JS  
$text2 = Front End Developer HTML CSS JS BscCSIT

Conversion of text into Array:  
$array\_text1=[‘Computer’, ‘Engineer’,’BscCSIT’,’HTML’,’CSS’,’JS’]  
$array\_text2=[‘Front’, ‘End’,’Developer’,’HTML’,’CSS’,’JS’,’BscCSIT’]

Step 1:  
Removing duplicate terms from array\_text1 using function  
array\_unique($array\_text)

Step2:  
Assigning key values for $array\_test1 using function

a$= $a = array\_fill\_keys($array\_text1, 0) with key value zero.

$a=Array(

[computer]=>0

[Engineer]=>0

[BscCSIT]=>0

[HTML]=>0

[CSS]=>0

13[JS]=>0

Step 3:

Now, it again assign the key value for $array\_text1 and $array\_text2 which is 1 and it is added

with $a.

1. $array\_text1=array\_fill\_keys($a,1)+$a

2. $arrary\_text2=arrary\_fill\_keys($b,1)+$a

Output for $array\_text1:

array([computer]=>1

[Engineer]=>1

[BscCSIT]=>1

[HTML]=> 1

[CSS] => 1

[JS] =>1

Output for $arrary\_text2:

[Computer]=> 0

[Engineer] =>0

[BscCSIT]=> 1

[HTML] =>1

[CSS]=>1

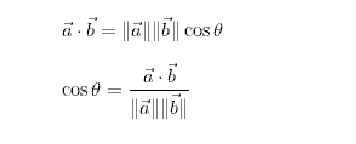
[JS]=>1

[front]=>1

[End]=>1

[Developer]=>1

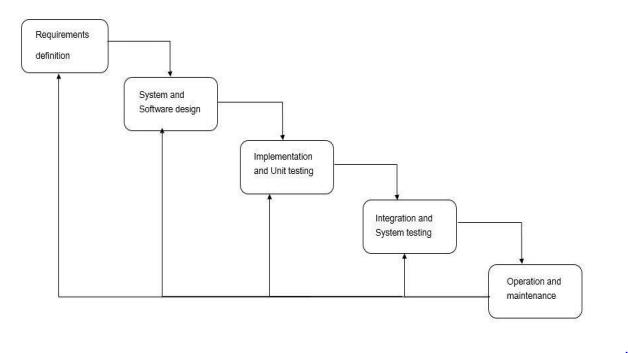
Thus, we compute cosine similarity using output of array text 1 and array text 2 using following

formula:

If similarity is greater than 0.5 then we will suggest corresponding job vacancy to the jobseeker.

# **CHAPTER FIVE IMPLEMENTATION AND TESTING**

## **Implementation**



**Figure 8. Waterfall model**

The waterfall model is a classical model used in the system development life cycle to create a system with a linear and sequential approach. It is termed as waterfall because the model develops systematically from one phase to another in a downward fashion. This model is divided into different phases and the output of one phase is used as the input of the next phase. Every phase has to be completed before the next phase starts and there is no overlapping of the phases. For our project we used the Waterfall Model because our project is a medium scale project and all the

requirements needed for our project were clearly known. Firstly, we analyzed the requirements for our project and then we designed our system. After this we moved to the implementation phase. Once we are done with the implementation phase, we will move towards the testing phase of the product. The ready product will be then deployed to the users and the maintenance will be done by the team with the passing time.

### **5.1.1 Tools Used**

The coding phase of the software development life cycle is concerned with the development of code that will implement the design. Coding is the most valuablepart of developing software. The code is written in a formal language called programming language. Coding usually involves front-end and back-end tools. A different front end and back-end tools used in the Online Job Portal System are finely discussed below.

* + - 1. **Frontend**The front end is the part of the system where normal users communicate with the system. The front end is related to the user interface. The front end consists of different coding tools where codes are written to develop the system. The Online Job Portal System uses the following front-end tools.

• HTML

HTML (Hypertext Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. "Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

• CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

* + - 1. **Backend**

Back end is the part of the system that normal users don’t want to know about. Back end possesses a database. The following tools are contained in the back end of the Online Job Portal System.

• PHP

PHP is an acronym for “Hypertext Preprocessor” and is a server-side programming language used for web development. It is a widely used, open-source scripting language. The PHP language has its roots in C and C++ so it is easy to learn and manipulate. PHP can run on both UNIX and Windows servers. In the development of this job portal system, PHP was chosen because it is dynamic, cheap and easy to set up, secure, fast, reliable and can interact with many different relational database management systems (DBMS).

• MYSQL

MySQL is one the world’s most popular open-source database management systems. It is a free-to-use, opensource database that enables users to create databases and perform various forms of data manipulations and database administration. MySQL was chosen for the development of this system because of its advanced features like the following;

▪ Data Security  
▪ On-Demand Scalability  
▪ High Performance  
▪ Complete workflow control  
▪ Easy to learn and use

## **5.3** **Testing**

Testing is a process of executing a program with the intent of finding bugs that makes the application fail to meet the expected behavior. Regardless of the development methodology, the ultimate goal of testing is to make sure that what is created does what it is supposed to do. Testing plays a critical role for assuring quality and reliability of the software. We have included testing as a part of the development process. The test cases should be designed with maximum possibilities of finding the errors or bugs. Various levels of testing are as follows.

### **5.3.1 Testing Levels**

**System Testing:**

System testing is performed on a completely integrated system to see if it meets the requirements.

### **5.3.2 Test Cases:**

A test case is a set of rules or conditions to check if the system or one of its features works in accordance with the requirement. It is a document with a set of details which includes, set of test data, expected results, actual results, environment information and so on. We have designed and executed a few test cases to check if the application meets the functional requirements.

Below are the test cases for Online Job Recommendation System application.

|  |  |  |  |
| --- | --- | --- | --- |
| Test Module | Test Case | Expected Result | Test Result |
| ADMIN | Provide valid login credentials | User successfully logged in and directed to the admin dashboard page | PASS |
| ADMIN | Enters invalid login credentials | Displays error message | PASS |
| ADMIN | Upon successful login, click on the ‘Manage Employer’ tab | Display the list of employers whose account registration is to be verified | PASS |
| ADMIN | Upon Successfull login, click on the ‘Manage Job Seeker ‘ tab | Display the list of seekers whose account registrationis to be verified | PASS |
| ADMIN | Upon Successfull login, click on the ‘Public No ‘ tab | Can publish new news and edit and delete existing news | PASS |
| ADMIN | Upon Successfull login, click on the ‘Feedback‘ tab | Can view the feedbacks send by employers and seekers | PASS |
| EMPLOYER | Provide details for registration | Employer registration successfully submitted | PASS |
| EMPLOYER | Upon successful login,click on ‘Manage Jobs’  tab | Employer posts jobs with therequired details and view the details of all the posted jobs | PASS |
| EMPLOYER | Upon successful  login, click on ‘Walking Interview’ tab | Employers post job vacancies that has to be attended physically and also can view the list of previously posted jobs | PASS |
| EMPLOYER | Upon successful login, click on ‘Profile’ tab | Can view as well as edit information about themselves | PASS |
| EMPLOYER | Upon successful login, click on ‘APPLICATION’ tab | Can view the list of applications send by applicants or PASS seeker for a particular jobs as well as can reply all later message to the applicant | PASS |
| JOBSEEKER | Provide details for registration | Job Seeker registration successfully submitted | PASS |
| JOBSEEKER | Enters invalid login credentials | Enters invalid login credentials | PASS |
| JOBSEEKER | Upon successful login, click on ‘ Profile’ tab | List details of jobseeker | PASS |
| JOBSEEKER | Upon successful login, click on ‘Search Jobs’ tab | Can search for different job vacancies that are  available and also view status of previously applied jobs | PASS |
| JOBSEEKER | Upon successful login, click on ‘Walking Interview’ tab | Can view the list of vacancies which should be attended physically | PASS |
| JOBSEEKER | Upon successful login, click on ‘Education’ tab | Can add their educational degrees | PASS |
| JOBSEEKER | Upon successful login, click on ‘Feedback’ tab | Can send feedbacks to administrator | PASS |

## **Table 1:UnitTestCases for application**

Test Objectives:Successfully submits the job application

|  |  |  |  |
| --- | --- | --- | --- |
| TEST CONDITION | INPUT SPECIFICATION | OUTPUT SPECIFICATION | PASS/FAIL |
| The user is currently on the jobs page | User clicks on the “APPLY” tab against a job post | Prompts a message as “Successfully applied to this job” | PASS |

**Table 2: Test case for applying jobs**Test Objectives:Successfully submits the job application

|  |  |  |  |
| --- | --- | --- | --- |
| TEST CONDITION | INPUT SPECIFICATION | OUTPUT SPECIFICATION | PASS/FAIL |
| The job seeker is currently logged into their account | The job seeker clicks on search job tab | Lists all the jobs that are applied | PASS |

**Table 3: Test case for listing all jobs**

## **Result Analysis**

# **CHAPTER SIX CONCLUSION AND FUTURE RECOMMENDATIONS**

## **Conclusion**

The major goal of this project is to provide a web-based system where Job Seekers can find the relevant jobs based on their profile specifications and employers can manage and select the right candidate for their jobs by viewing their profile and resumes.

It is a web-based application developed using Html, CSS, JavaScript, Bootstrap for its front end, PHP for its backend and MySQL for databases.

People are unable to get access to job opportunities due to inefficient distribution of information on job offers but now the Internet has made a huge impact on knowledge management and information dissemination all over the world. It is expected to use less budget and less resources and can be easily access by every user because of its user-friendly environments and also the system is not so complex for users. Any person who has access to the Internet can access information anywhere in the world. This system aims and objective is to provide a platform where recruiters and job seekers can communicate.

The development and implementation of the system provides the following features:  
• User authentication  
• CV or resume upload  
• Feedback mechanism  
• Find relevant jobs

## **Future Recommendations**

This project fulfills the primary requirements of the job seekers and employers. It can be extended in several ways. It can provide recommendations and email updates for new job postings based on the job seeker’s search history. Since the job seekers might be interested in building a strong resume, it can provide tips and information for the same. It can also provide templates for building the Resumes which might interest most applicants. The mobile application is developed fulfilling the functionalities of job seekers, it can be extended to support functionalities of Employers as well.

In this system, Cosine Similarity is calculated using the term frequency of each term in job seeker biodata. But Term frequency suffers from critical problems as all the terms are considered equally important. In fact, Certain terms have little or no selective power in determining relevance. And also terms which appear very few in number have higher probability in determining the relevance. So, to overcome this problem we have to scale down the term weight of the terms with high collection frequency. For this we can use TF-IDF weighting instead of term frequency which can improve the outcome of our system.

# **References**

BIBLIOGRAPHY

|  |  |
| --- | --- |
|  | 1. Abhjit Roy, “Introduction to recommender system-1:Content-Based Filtering And Collaborative Filtering”.[Online] Available:https://towardsdatascience.com/introduction-to-recommender-systems-1- 971bd274f421.  [Accessed 09 Feb 2023].  2. K. Lakiotaki, ”UTA-Rec: A Recommender System based on Multiple Criteria Analysis,” RecSys’08, pp. 219-225, 2008.  3. S. O. P. Mucheol Kim, ”Group affinity based social trust model for an intelligent movie recommender system,” Multimedia Tools and Applications, vol. 64, no. 2, pp. 505-516, May, 2013.  4. R. S. Punam Bedi, ”Trust based recommender system using ant colony for trust computation,” Expert Systems with Applications: An International Journal, vol. 39, no. 1, pp. 1183-1190, January, 2012. |
|  |  |