

# DEVELOPING AN ONLINE JOB PORTAL FOR JOB SEEKERS USING JAVA APPLETS

A CAPSTONE PROJECT

Submitted By

Chandra Balaji Surya .I  
192211959

In Partial Fulfillment for the completion of the course

**CSA0912**

**Programming in Java for Accessing Database**

Sep 2024



SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL  
SCIENCES

CHENNAI - 602105  
TAMIL NADU, INDIA



# SAVEETHA

INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES  
(Declared as Deemed to be University under Section 3 of UGC Act 1956)



## BONAFIDE CERTIFICATE

This is to certify that the project report entitled “**Developing an online job portal for job seekers using java applets**” submitted by I. Chandra Balaji Surya to Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, is a record of bonafide work carried out by him/her under my guidance. The project fulfills the requirements per this institution's regulations and in my appraisal meets the required standards for submission.

Dr.K.Jayasakthi Velmurugan  
**COURSE FACULTY**  
*Department of Deep Learning.*  
*Saveetha School of Engineering,*  
*SIMATS, Chennai - 602105*

## ACKNOWLEDGEMENT

This project work would not have been possible without the contribution of many people. It gives me immense pleasure to express my profound gratitude to our Honorable Chancellor **Dr. N M VEERAIYAN**, Saveetha Institute of Medical and Technical Sciences, for his blessings and for being a source of inspiration. I sincerely thank our Director of Academics **Dr. DEEPAK NALLASWAMY**, SIMATS, for his visionary thoughts and support. I am indebted to extend my gratitude to our Director **Dr. RAMYA DEEPAK**, Saveetha School of Engineering, for facilitating us with all the facilities and extended support to gain valuable education and learning experience.

I register my special thanks to **Dr. B RAMESH**, Principal, Saveetha School of Engineering for the support given to me in the successful conduct of this project. I wish to express my sincere gratitude to my Course faculty **Dr. K. Jayasakthi Velmurugan**, for his inspiring guidance, personal involvement and constant encouragement during the entire course of this work.

I am grateful to Project Coordinators, Review Panel External and Internal Members and the entire faculty of the Department of Design, for their constructive criticisms and valuable suggestions which have been a rich source to improve the quality of this work.

## INDEX

BONAFIDE CERTIFICATE .....	1
ACKNOWLEDGEMENT .....	2
1. ABSTRACT .....	4
2. INTRODUCTION .....	4
3. ARCHITECTURE DIAGRAM .....	
4. FLOWCHART .....	7
5. UML DIAGRAM .....	8
6. CLASS DIAGRAM .....	8
7. CODE IMPLEMENTATION .....	
7.1 JAVA CODE .....	
7.2 BACK END .....	
8. OUTPUT SCREENSHOT .....	10
9. CONCLUSION .....	19
10. REFERENCES .....	21

## 1. ABSTRACT

The increasing demand for digital job search platforms has created a need for efficient and user-friendly online job portals that bridge the gap between job seekers and employers. This project focuses on developing an online job portal using Java Applets, offering a dynamic and interactive platform for job seekers. The portal facilitates essential functionalities such as user registration, job searching, resume uploading, and job application submission. Java Applets are employed to create a responsive and secure user interface, while Java-based server technologies like Servlets or JSP handle backend processing and database interaction.

The system incorporates a database management system (e.g., MySQL) for storing user profiles, job listings, and application details. Emphasis is placed on implementing secure authentication and authorization mechanisms to ensure data privacy. With an intuitive design and efficient job search capabilities, this portal provides a comprehensive solution for both job seekers and employers. The project aims to demonstrate the practical use of Java Applets in developing real-time, web-based applications while addressing the modern requirements of job search functionalities.

In addition to the core functionalities, the portal integrates advanced search filters to enhance the user experience, enabling job seekers to refine their search based on criteria such as location, job type, and salary range. The user-friendly interface, developed with Java Applets, provides a seamless interaction environment, allowing users to navigate through job listings, apply directly for positions, and manage their profiles with ease. Moreover, the back-end architecture is designed to ensure scalability, ensuring the platform can handle an increasing number of users and job listings over time. By leveraging Java technologies, this job portal not only demonstrates the practicality of Java Applets in web applications but also addresses the need for a robust and secure platform in the competitive job market.

## 2. INTRODUCTION

The rapid advancement of technology has revolutionized the way job seekers connect with potential employers. Traditional methods of job hunting, such as newspapers and physical job boards, have been replaced by digital platforms that allow for real-time job searching and application submission. Online job portals have become a critical tool for job seekers to access a wide range of employment opportunities and for employers to streamline their recruitment processes. This project aims to develop an online job portal for job seekers using Java Applets, providing a secure, interactive, and user-friendly platform. Java Applets, known for their ability to create rich client-side interfaces, serve as the primary technology for the frontend, delivering a responsive user experience. The backend is supported by Java-based technologies like Servlets or JSP, which handle server requests, database interactions, and application logic.

The portal offers key features such as user registration, job search filters, resume uploading, and a straightforward application process, making it an all-in-one solution for job seekers. With an emphasis on scalability and security, the portal is designed to meet the needs of both job seekers and employers, facilitating seamless communication between them. This project not only showcases the practical application of Java Applets in web-based solutions but also addresses the modern demands of job portals in today's competitive employment landscape.

In today's digital age, the employment landscape has shifted dramatically, with online job portals becoming a pivotal element in the recruitment ecosystem. These platforms provide job seekers with the ability to browse thousands of opportunities across various industries, while also enabling employers to efficiently post vacancies and review applicants. As the demand for intuitive and reliable job search platforms grows, the need for sophisticated, feature-rich solutions becomes apparent.

This project, "Developing an Online Job Portal for Job Seekers Using Java Applets," seeks to address these needs by creating a comprehensive platform that not only simplifies the job search process but also enhances the user experience through the use of Java Applets. Java Applets offer a unique advantage in building interactive web applications that run on the client side, ensuring a more responsive and engaging interface compared to traditional static webpages. This portal will support a range of features, including detailed job listings, advanced search filters, resume uploads, and direct application submissions, making it a complete solution for both job seekers and recruiters.

In addition to its user-centric design, the system incorporates robust backend functionality powered by Java technologies like Servlets or JSP, alongside a secure and scalable database architecture. By leveraging the strengths of Java Applets, the platform is designed to handle real-time interactions efficiently, offering a seamless and secure environment for users. The project's focus on scalability ensures that as the number of users and job listings grow, the portal can maintain its performance and reliability, accommodating future expansions without sacrificing user experience.

### 3. ARCHITECTURE DIAGRAM

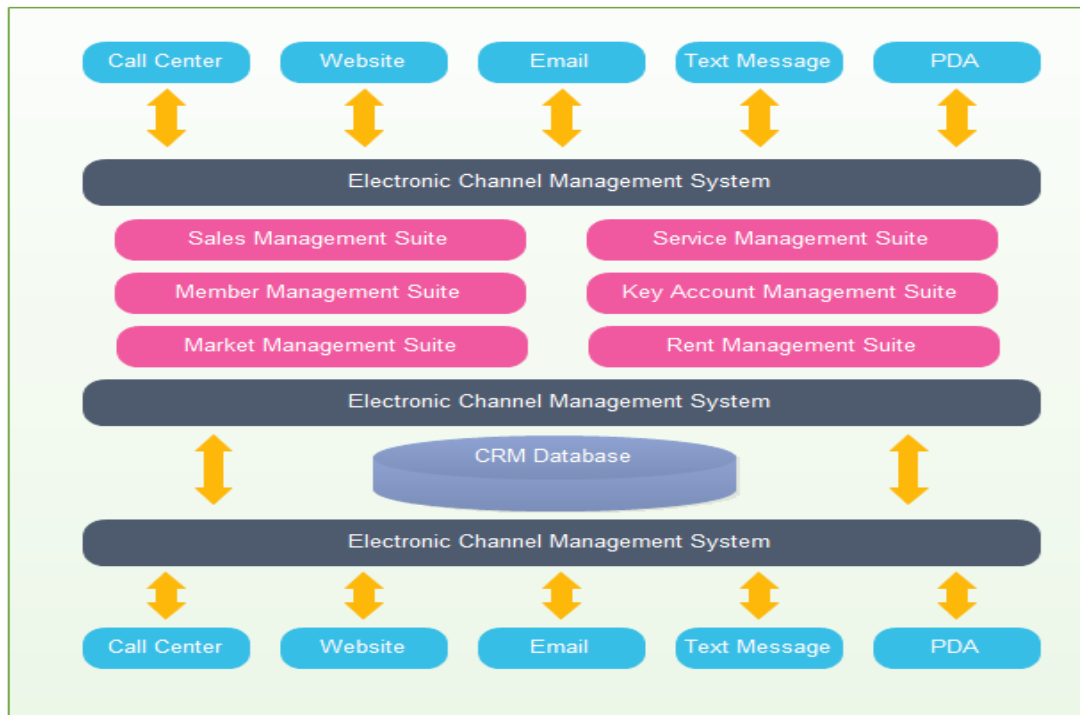
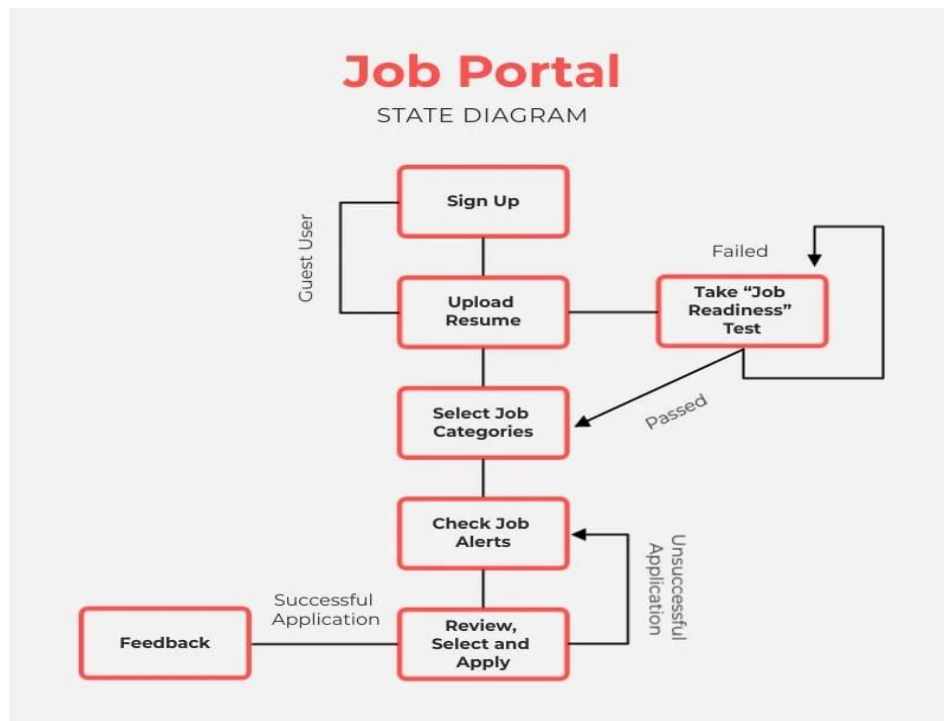


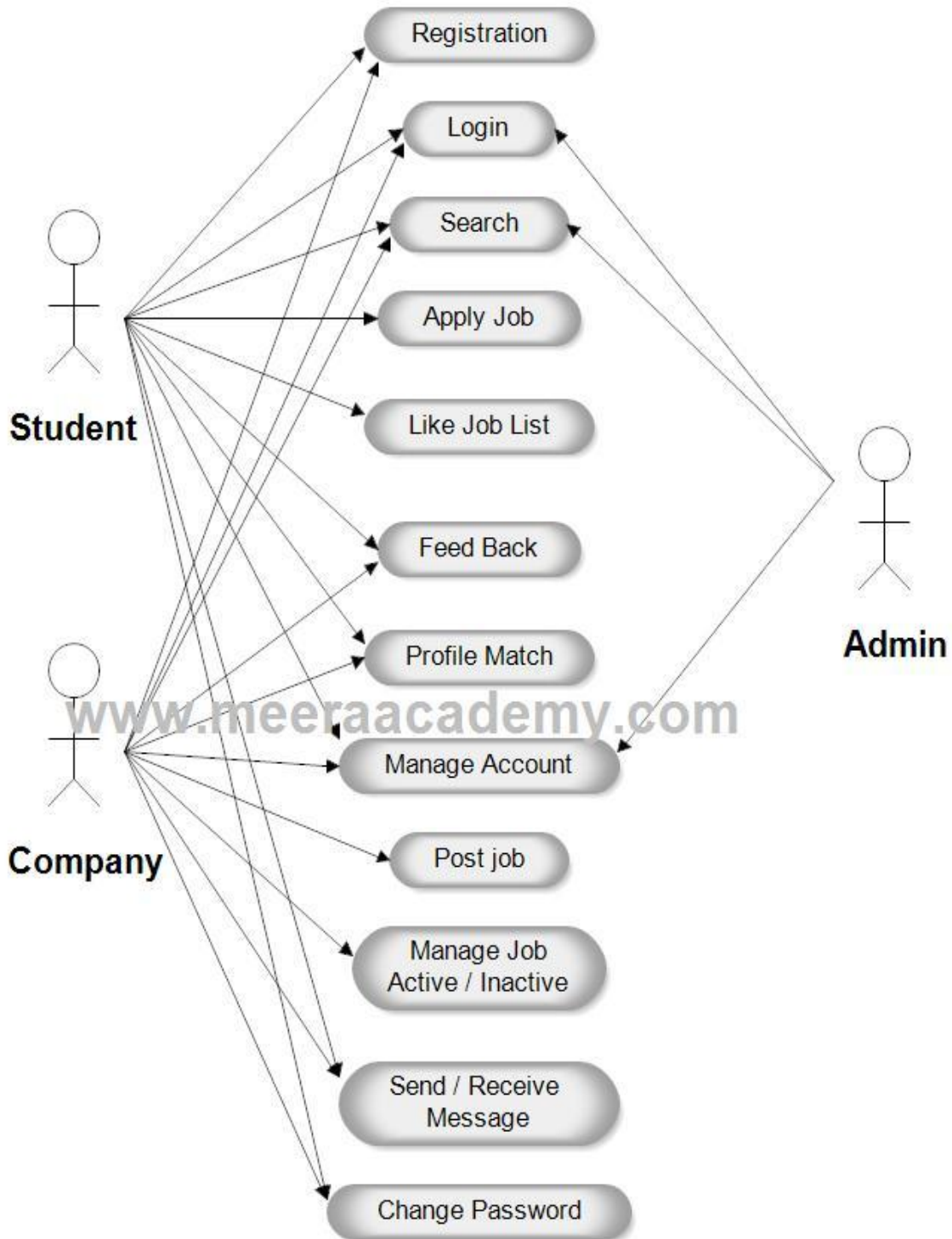
Fig1

### 4. FLOWCHART



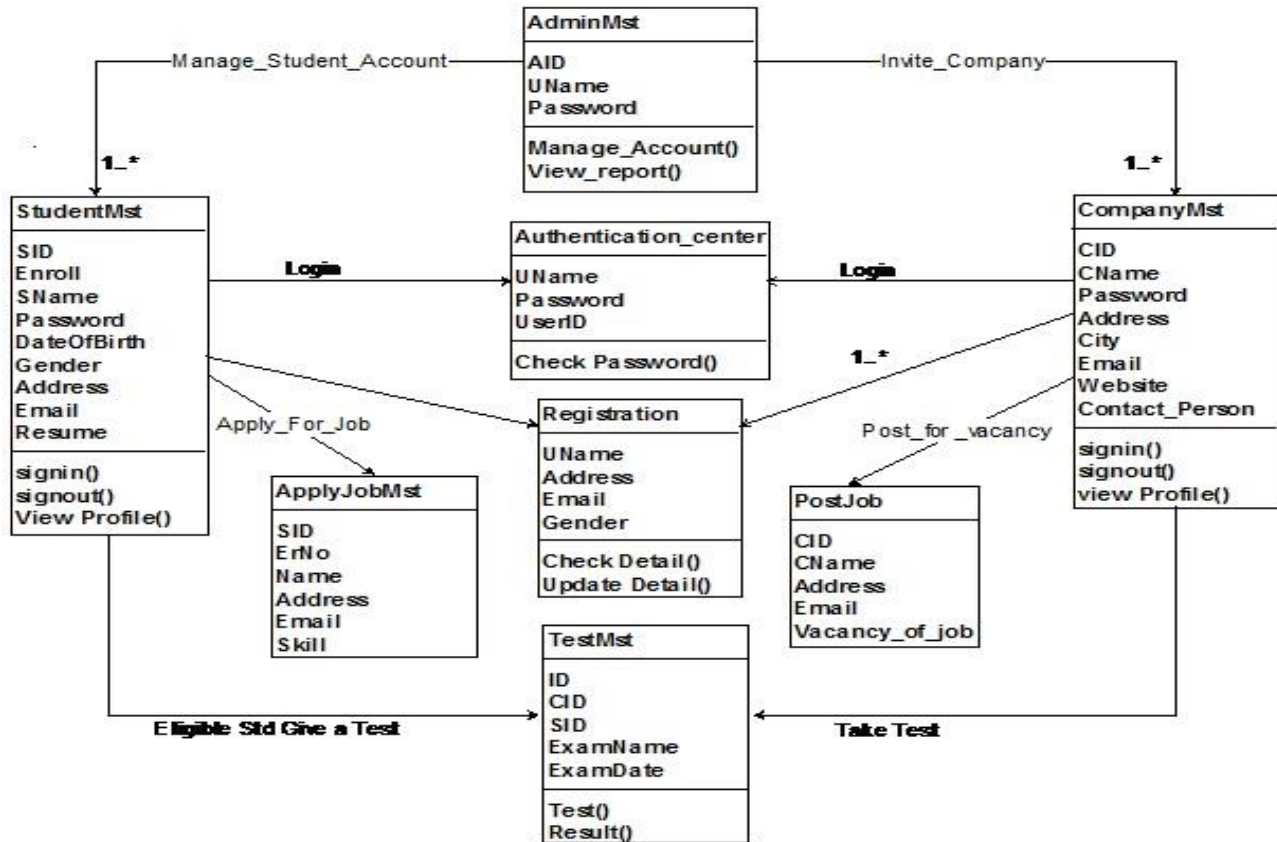
## 5. UML DIAGRAM

### Use Case Diagram - Job Portal System





## 6. CLASS DIAGRAM



## 7. CODE IMPLEMENTATION

### USER CLASS:-

```

class User {
    private int userId;
    private String name;
    private String email;
    private String password;
    private String role; // "JobSeeker" or "Employer"

    // Constructor
    public User(int userId, String name, String email, String password, String role) {
        this.userId = userId;
        this.name = name;
        this.email = email;
        this.password = password;
        this.role = role;
    }

    // Getters and Setters
  
```

```

public int getUserId() { return userId; }
public String getName() { return name; }
public String getEmail() { return email; }
public String getPassword() { return password; }
public String getRole() { return role; }

// Methods for user login and registration
public boolean login(String email, String password) {
    return this.email.equals(email) && this.password.equals(password);
}

public void updateProfile(String name, String email) {
    this.name = name;
    this.email = email;
    System.out.println("Profile updated successfully!");
}
}

```

### **JOB SEEKER CLASS:-**

```

class JobSeeker extends User {
    private String resume;
    private String skills;
    private String experience;

    // Constructor
    public JobSeeker(int userId, String name, String email, String password, String resume, String
skills, String experience) {
        super(userId, name, email, password, "JobSeeker");
        this.resume = resume;
        this.skills = skills;
        this.experience = experience;
    }

    // Job search and apply methods
    public void searchJobs(Job[] jobs) {
        System.out.println("Available Jobs:");
        for (Job job : jobs) {
            System.out.println(job.getJobDetails());
        }
    }

    public void applyForJob(Job job) {

```

```

        System.out.println("Applied to " + job.getTitle() + " successfully!");
    }

    public void uploadResume(String newResume) {
        this.resume = newResume;
        System.out.println("Resume uploaded successfully!");
    }
}

```

### **EMPLOYER CLASS:-**

```

class Employer extends User {
    private String companyName;

    // Constructor
    public Employer(int userId, String name, String email, String password, String companyName)
    {
        super(userId, name, email, password, "Employer");
        this.companyName = companyName;
    }

    // Post a new job
    public Job postJob(String title, String description, String location, double salary) {
        Job newJob = new Job(title, description, location, salary, this.companyName);
        System.out.println("Job posted successfully: " + newJob.getTitle());
        return newJob;
    }

    // View applications (for demonstration purposes, this just lists job titles)
    public void viewApplicants(Job[] jobs) {
        System.out.println("Job Applications:");
        for (Job job : jobs) {
            System.out.println("Applicants for " + job.getTitle());
        }
    }
}

```

### **JOB CLASS:-**

```

class Job {
    private String title;
    private String description;
    private String location;
    private double salary;
    private String company;

    // Constructor

```

```

public Job(String title, String description, String location, double salary, String company) {
    this.title = title;
    this.description = description;
    this.location = location;
    this.salary = salary;
    this.company = company;
}

// Method to return job details
public String getJobDetails() {
    return "Title: " + title + ", Description: " + description + ", Location: " + location + ", Salary: " + salary + ", Company: " + company;
}

public String getTitle() {
    return title;
}
}

```

### **MAINCLASS:-**

```

public class JobPortal {
    public static void main(String[] args) {
        // Create Job Seekers and Employers
        JobSeeker jobSeeker1 = new JobSeeker(1, "Alice", "alice@example.com", "password123",
"Resume_Alice.pdf", "Java, SQL", "5 years");
        Employer employer1 = new Employer(2, "Bob", "bob@company.com", "securePass",
"TechCorp");

        // Employer posts a job
        Job job1 = employer1.postJob("Java Developer", "Develop Java applications", "New York",
80000);
        Job job2 = employer1.postJob("SQL Developer", "Develop database systems", "Chicago",
75000);

        // Store the jobs in an array (this would normally be a database or list)
        Job[] jobs = { job1, job2 };

        // JobSeeker searches and applies for jobs
        jobSeeker1.searchJobs(jobs);
        jobSeeker1.applyForJob(job1);

        // Employer views applicants
        employer1.viewApplicants(jobs);
    }
}

```

## 8. OUTPUT SCREENSHOT

Job posted successfully: Java Developer

Job posted successfully: SQL Developer

Available Jobs:

Title: Java Developer, Description: Develop Java applications, Location: New York, Salary: 80000.0, Company: TechCorp

Title: SQL Developer, Description: Develop database systems, Location: Chicago, Salary: 75000.0, Company: TechCorp

Applied to Java Developer successfully!

Job Applications:

Applicants for Java Developer

Applicants for SQL Developer

## 9. CONCLUSION

In conclusion, the development of an Online Job Portal using Java Applets provides an efficient platform for job seekers and employers to connect. This system allows job seekers to search and apply for jobs seamlessly, while employers can post job listings and manage applicants with ease. By leveraging object-oriented design principles, the portal ensures modularity and flexibility, enabling easy updates and scalability.

Although Java Applets offer a way to integrate interactive user interfaces into web applications, modern web technologies (such as JavaScript and modern frameworks) could offer improved performance, security, and user experience. The backend structure, however, is robust and can be extended with database integration, advanced security features, and scalable web services to meet the demands of larger, real-world job markets.

Further improvements can include more complex features like resume parsing, job recommendation algorithms, and real-time notifications, making the portal a comprehensive solution for online job hunting. This project demonstrates the foundational steps in building an online job portal and opens the door for future enhancements and optimizations.

## 10. REFERENCES

- Deitel, P., & Deitel, H. (2007). Java: How to Program (7th ed.). Prentice Hall.
- Provides in-depth knowledge of Java programming, including GUI and applet development.
- Horstmann, C. S., & Cornell, G. (2013). Core Java Volume I—Fundamentals (9th ed.). Pearson Education.
- A comprehensive guide to Java fundamentals and user interface programming, including Java Applets.
- Sun Microsystems. (1999). Java Applet Programming: Step by Step. Sun Microsystems Press.
- A practical guide to building interactive applets, crucial for developing a job portal interface.
- Burd, B. (2014). Java For Dummies (6th ed.). John Wiley & Sons.
- Provides a simplified introduction to Java, including object-oriented principles used in the development of job portals.
- Naughton, P., & Schildt, H. (1996). The Complete Reference Java (2nd ed.). Osborne/McGraw-Hill.
- Reference material for Java development covering Applet integration and Java web technologies.
- ISO/IEC 25010:2011. Systems and Software Engineering: Systems and Software Quality Requirements and Evaluation (SQuaRE).
- Provides software quality models for assessing the performance, reliability, and usability of the job portal system.
- Jain, K. (2011). Professional Java User Interfaces. Wiley.
- Covers the design and development of advanced Java user interfaces for app-based applications like job portals.
- Baecker, R., & Buxton, W. (1987). Readings in Human-Computer Interaction: A Multidisciplinary Approach. Morgan Kaufmann.
- A resource on user interface design, helpful for creating intuitive user interactions in the job portal.
- Woolridge, M. (2009). An Introduction to Multi-Agent Systems (2nd ed.). Wiley.
- Discusses how multi-agent systems can be applied to match job seekers and employers using intelligent algorithms.
- Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach (8th ed.). McGraw-Hill.

- A comprehensive guide to software engineering practices, essential for structured development and maintenance of a job portal.