JOB NAVIGATOR

UNIVERSITY OF INFORMATION TECHNOLOGY

SEPTEMBER 2024

Course Name: CST-6108 Enterprise Applications Development using Java

Project Title: Job Navigator Website

Group Members

1. **Thant Zin Win**
    - Student ID: CS -
    - Email: @gmail.com
2. **Lin Let Win**
    - Student ID: CS -
    - Email: @gmail.com
3. **Lin Let Shwe Yi Khin**
    - Student ID: CS -
    - Email: @gmail.com
4. **Yin Mon Win**
    - Student ID: CS -
    - Email: @gmail.com

Supervisor: Dr. Win Win Thant

Department: Faculty of Computer Science

Submission Date:

Contents

Introduction 3
Project Planning and Requirement Analysis 3
System Design and Architecture 3
Development Environment Setup 3
Screen Layout and Design 3
Conclusion and Reflection 3
Appendices. 3

Introduction

- Overview
- Project Scope
- Objectives

Project Planning and Requirement Analysis

- Functional vs. Non-Functional Requirements
- Creating Use Cases and User Stories

System Design and Architecture

- System Architecture Design
- Database Design: ER diagrams

Development Environment Setup

- Tools and Technologies
- Development Environment
- Coding Standards

Screen Layout and Design

Conclusion and Reflection

- Project Summary
- Challenges and Solutions
- Lessons Learned
- Future Enhancements

Appendices.

- Screen Design
- References

Chapter-1

Introduction

1.1 Overview

Our Job Navigator website is a comprehensive online job platform designed to bridge the gap between job seekers and employers. In today's competitive job landscape, there is a growing need for platforms that not only provide job listings but also enable effective networking, professional development, and streamlined recruitment processes. Our platform aims to fulfill this need by offering a robust, user-friendly solution that supports both job seekers and employers in their respective journeys.

For job seekers, the platform will allow the creation of detailed profiles that highlight their skills, experience, and education. They will have access to an extensive database of job listings, with search functionality that allows filtering by criteria such as job name, company name, location and work place types. Additionally, job seekers will be able to apply directly through the platform and track their applications. Our website aims to help individuals identify and pursue career goals.

For employers, a suite of tools is offered to them to manage the recruitment process efficiently. Employers can create job vacancies, view applicant profiles, review applications and manage recruitment processes.

The comprehensive profile of a company including job positions and what benefits are provided for the employees are presented in the company’s profile so job seekers can easily browse their desired company.

The platform is designed to streamline the hiring process by providing employers with the ability to manage their recruitment activities all in one place. It will support various user interactions, including user registration, profile creation, job posting, job search, and application tracking. The project emphasizes delivering a user-friendly, secure, and scalable solution.

1.2 Project Scope

The scope of our Job Navigator project encompasses the design, development, and deployment of a job portal that caters to both job seekers and employers. The platform will allow job seekers to register, create professional profiles, upload resumes, search for job opportunities, and apply for positions. Employers will have the ability to post job listings, search for suitable candidates, and manage applications. The project will also include developing an administrative interface for managing user accounts, job listings, and site content. Key considerations in the project scope include security, scalability, and usability. The website will be developed to handle a growing user base and expanding job listings while ensuring that sensitive user data is protected through encryption and secure authentication mechanisms.

1.3 Objectives

The primary objectives of our Job Navigator website are as follows:

- Develop a User-Friendly Interface: We ensure that the website's interface is intuitive and easy to navigate for both job seekers and employers.
- Implement Core Functionality: We aims to provide essential features such as user registration, job posting, job search, and application management, ensuring they function efficiently and meet user needs.
- Ensure Security: We have implemented password hashing as a core security measure to protect user login credentials. While the platform's security focuses on this feature, future enhancements may include additional security mechanisms like data encryption and protection against common vulnerabilities.
- Scalability: We design the system architecture to be scalable, accommodating future growth in terms of users, job postings, and additional features.
- Cross-Platform Accessibility: The website is accessible on various devices, including desktops, tablets, and smartphones, providing a responsive design that adjusts to different screen sizes.
- Improved Job Search Facility: We offer advanced search functionality, and application tracking, enabling job seekers to quickly and easily find job opportunities that match their skills and preferences.
- Streamline the recruitment process: For employers, we provide a suite of tools to manage job postings, applicant profiles, and recruitment processes in one place, reducing time and cost to.

  By meeting these objectives, our project aims to create a powerful and versatile job platform that addresses the needs of both job seekers and employers, providing a valuable tool for professional networking and recruitment.

Chapter 2

Project Planning and Requirement Analysis

2.1 Functional vs. Non-Functional Requirements

Functional Requirements

- User Registration and Login: Users (both job seekers and employers) can register and log in securely.
- Job Seeker Profile Management: Job seekers can create and update their professional profiles, including personal details, education, work experience, and resumes.
- Company Profile Management: Employers can create and update their company profiles, including company information, culture, benefits, and values.

- Job Search: Users can search for jobs using various filters such as location, job type, and industry.
- Job Posting: Employers can post job vacancies, including job descriptions, and requirements.
- Application Management: Job seekers can apply for jobs, and employers can view and manage applications.
- Notifications: Job seekers can receive notifications when their applications are accepted.
- Payment Gateway Integration: Employers can pay for job postings through a secure payment gateway.

Non-Functional Requirements

- Performance: The system should handle a large number of concurrent users without significant delays.
- Security: User passwords are protected securely using a strong hashing algorithm called BCrypt to prevent unauthorized access.
- Usability: The platform should have a user-friendly interface, ensuring that new users can navigate and use core features (e.g., job search, profile creation) without needing extensive training or help.
- Reliability: Ensure high availability of the system with minimal downtime.
- Scalability: The architecture should support future growth in the user base and data.
- Availability: The system should be available 24/7, with a minimum uptime of 99.9%.
- Compatibility: The platform should be compatible with different browsers, devices, and operating systems to ensure a seamless user experience across different platforms.

2.2 Creating Use Cases and User Stories

Use Cases

User stories

Registration
As a job seeker, I want to register an account so that I can start searching and applying for jobs.
As an employer, I want to create an account so that I can post job listings and review candidates.
As an admin, I want to manage user accounts so that I can ensure that the system remains secure and functional.

Logging In
As an employer, I want to securely log in so that I can manage job postings and view applicant profiles.
As a job seeker, I want to log into my account so that I can browse job vacancies.
If I'm using the app and forget my password, I want a simple way to reset it.
As a user, I want to see an error message if I enter incorrect login details, so that I know when my login attempt has failed.

Job Search and Posting
As a job seeker, I want to search for jobs by title, job type (full time, internship or part-time) and work type (on-site, remote, hybrid) so that I can find relevant job opportunities.
As a job seeker, I want to filter jobs by location, salary, and job type so that I can find the most suitable positions quickly.
As a recruiter, I want to post a job vacancy, so I can find a new team member.
As an employer, I want to view applications for my job postings so that I can manage the applications.

Application Management
As a job seeker, I want to apply to a job just by clicking the job postings so that it will save my time.

User Profile Management
As a job seeker, I want to create and update my profile with personal information, skills, and work experience so that employers can easily view my qualifications.
As an employer, I want to update my company profile so that job seekers can learn more about my company.

Notification System
As an employer, I want to receive notifications when I am accepted.

Cross-Platform Accessibility
As a keyboard-only user, I want to access the Job Navigator platform from my mobile device so that I can search and apply for jobs without needing a personal computer.

Admin Management
As an admin, I want to manage job postings and user accounts so that I can remove any inappropriate content or block abusive users.
As an admin, I want to see the reviews given by the users so that I can make future refinements.

Chapter - 3

System Design and Architecture

3.1 System Architecture Design

At a high level, the system architecture can be divided into three main layers: presentation, application, and data access.
Presentation Layer: The presentation layer is responsible for handling user input and displaying the output to the user. This layer consists of:

Web Controllers: Handle HTTP requests and responses, and interact with the application layer to perform business logic.
JSP templates: Used for the user interface.
CSS and JavaScript: Used for styling and client-side scripting.

Application Layer: The application layer is responsible for performing business logic and interacting with the data access layer. This layer consists of:
Service Classes: Encapsulate business logic and interact with the data access layer.
Business Logic: Implemented in the service classes, this layer performs tasks such as job searching, profile creation, and matching.

Data Access Layer: The data access layer is responsible for interacting with the database and performing CRUD (Create, Read, Update, Delete) operations. This layer consists of:
JdbcTemplate: Used for interacting with the database, performing CRUD operations, and executing SQL queries.
DAO (Direct Access Object) Classes: Encapsulate database interactions and provide a layer of abstraction between the application layer and the database.

3.2 Database Design: ER diagrams

Chapter – 4

Development Environment Setup

4.1 Tools and Technologies

The development of the Job Navigator website will require the following tools and technologies:

1. Java EE: For building the core application.
2. Spring MVC: For implementing the MVC architecture and handling HTTP requests and responses.
3. Apache Tomcat: Server for deploying the web application.
4. MySQL: A relational database for storing users' information.
5. JDBC Template: For interaction with MySQL database.
6. Eclipse IDE: For writing and managing the codebase.
7. GitHub: For Collaborative development.
8. Figma: For the systematic and consistent frontend designs.

4.2 Development Environment

The development environment will be set up to support Java EE and Spring MVC without Maven. Configuration files such as web.xml and spring-servlet.xml will be manually created and managed. The project structure will include directories for Java resources, web content (JSP files, CSS, JavaScript), and configuration files. Testing and deployment will be carried out using Tomcat.

4.3 Coding Standards

Java naming conventions: Follow the standard Java naming conventions for classes, methods, and variables.
Code Documentation: Use clear and concise comments to provide clear explanations of their purpose and functionality of class and method.
Code Organization: Structure the code logically, separating concerns into different layers (e.g., controllers, models, DAOs).
Error Handling: Use try-catch blocks to handle exceptions and errors, and provide informative error messages.

Chapter – 5

Screen Layout and Design

Home Page:

Registration and Login Pages:

Profile Pages:

Job Searching Page:

Chapter – 6

Conclusion and Reflection

6.1 Project Summary

The Job Navigator website is a comprehensive online platform designed to connect job seekers with potential employers. The project aims to provide a user-friendly interface for job seekers to search and apply for jobs, and for employers to post job openings and manage applications.

The platform was built using Java EE, Spring MVC, and JDBC Template without Maven, focusing on delivering core functionalities such as user registration, job posting, job search, and application management.

Throughout the project, emphasis was placed on creating a user-friendly interface, ensuring security, and designing a scalable architecture. The development process involved a thorough analysis of both functional and non-functional requirements, followed by the design of a

multi-tiered system architecture that effectively separates concerns between the presentation, business logic, and data access layers.

Database design played a critical role in the project, with the creation of ER diagrams that mapped out key entities and their relationships, ensuring that the platform could efficiently manage user data, job listings, and application processes. The development environment was set up to handle the project's specific needs, with tools such as Tomcat Server and Eclipse IDE being utilized to streamline the coding and deployment processes.

6.2 Challenges and Solutions

1. Adaptation to New Technologies
   Challenge: Since not all team members were familiar with some of the technologies used, such as Spring MVC and JDBC Template, there was a learning curve that initially slowed down development.
   Solution: We individually learnt these technologies by ourselves and then shared our knowledge each other. This collaborative learning approach not only reduced the learning time but also encouraged team members, even the novices, to become more confident in their abilities.

2. Direct Message System
   Challenge: Our website will notify the users if they are accepted for a job position. However, further interaction between job seekers and employers such as direct email system will not be available.
   Solution: To facilitate seamless communication between job seekers and employers, a direct message system can be integrated into the website. This system will enable users to send and receive private messages, allowing for more personalized and efficient communication.

3. Personalized Job Recommendation
   Challenge: Our website offers the filtering mechanism but not the recommendation system.
   Solution: To provide a more tailored experience for job seekers, an AI-driven personalized job recommendation system can be integrated into the website.

6.3 Lessons Learned

The development of the Job Navigator website has provided valuable lessons and insights that can be applied to future projects. Some of the key takeaways include:

1. Data Management is critical.
   Effective data management is essential for ensuring fast data retrieval and processing and for maintaining the integrity of the system.

2. User Experience is crucial.

A user-friendly interface and responsive design are essential for ensuring a positive user experience.

3. Modular work breakdown is the key.
   Balancing multiple tasks like design, coding, testing, and documentation within the project timeline makes the system less complexed and saves time and effort.

4. Clear Requirements and Planning is important
   Unambiguous requirements are the key to success for our project. By spending more time refining both functional and non-functional requirements, the development process becomes smoother.

5. New Technologies are learnt.
   Throughout this project we came into familiar with the structure of Java EE for building scalable applications and Spring MVC for implementing the Model-View-Controller design pattern. This hands-on experience helped us grasp core concepts like dependency injection, annotation-based configuration, and handling web requests efficiently.

6.4 Future Enhancements

1. Direct Messaging System: To communicate more efficiently, facilitating a smoother hiring process.
2. Video Interview Integration: To provide a more convenient and efficient way for job seekers and employers to conduct interviews, reducing the need for in-person meetings.
3. AI-based job recommendation: To support job seekers with more accurate and relevant job suggestions, increasing the likelihood of successful job placements.
4. Integration with Social Media and Professional Networks: To enable job seekers to leverage their existing online presence, providing a more comprehensive view of their skills and experience.

Chapter – 7
Appendices

7.1 Screen Design
7.2 References