B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Information Science and Engineering

22ISE136 – Web Technologies

Mini-Project Report

on

JOB APPLICATION MANAGER

Submitted in partial fulfillment of the requirement
for the award of the degree of

Bachelor of Engineering

in

Information Science & Engineering

by

Pradyumna Mohan (1BG22IS030)

Academic Year 2023 – 2024

B.N.M. Institute of Technology

An Autonomous Institution under VTU Approved by AICTE

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the Mini-project entitled **Job Application Manager** is carried out by Mr. **Pradyumna Mohan** USN **1BG22IS030** the bonafide student of **B.N.M Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science & Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2023-2024. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of mini-project prescribed for the said Degree.

Mrs. Kavya N L
Assistant Professor, Dept. of ISEBNMIT

Dr. S Srividhya
Prof & Head, Dept. of ISEBNMIT

Name & Signature of the Examiners with date:

1.

2.

ACKNOWLEDGEMENT

We consider it a privilege to express through the pages of this report, a few words of gratitude to all those distinguished personalities who guided and inspired me in the completion of this mini project.

We would like to thank **Shri. Narayan Rao R Maanay**, Secretary, BNMEI, Bengaluru for providing an excellent academic environment in college.

We would like to thank **Prof. T.J. Rama Murthy**, Director, BNMIT, Bengaluru for having extended his support and encouragement during the course of work.

We would like to thank **Dr. S.Y. Kulkarni**, Additional Director, BNMIT, Bengaluru for his extended support and encouragement during the course of work.

We would like to express my gratitude to **Prof. Eishwar N Maanay**, Dean, BNMIT, Bengaluru for his relentless support, guidance, and encouragement.

We would like to thank **Dr. Krishnamurthy G.N**., Principal, BNMIT, Bengaluru for his constant encouragement.

We would like to thank **Dr. S. Srividhya**, Professor and Head of the Department of Information Science and Engineering, BNMIT, Bengaluru, for her support and encouragement towards the completion of the mini project.

We would like to express my gratitude to my guide **[Name]**, [Designation], Department of Information Science and Engineering, BNMIT, Bengaluru, who has given us all the support and guidance in completing the mini project successfully.

We would like to thank course coordinator **Mrs Kavya N L,** Assistant Professor, Department of Information Science and Engineering, BNMIT, for being the guiding force towards the successful completion of the mini project.

Pradyumna Mohan
1BG22IS030

Siddarth Bhadaureya
1BG22IS058

Table of Contents

List of figures

CHAPTER 1
INTRODUCTION

The Job Application Manager project represents a comprehensive exploration into the realm of web technologies, aimed at addressing the practical need for an efficient and user-friendly platform to manage job applications. This initiative focuses on developing a simplistic, yet robust website hosted on localhost, enabling the storage of applicant data, display of available job vacancies, and facilitation of application status checks.

In an era characterized by increasing digitalization, the 'Job Application Manager' project emerges as a compelling response to the ever-growing demand for streamlined job application processes. Through the seamless integration of web technologies, this project endeavors to offer a cohesive and intuitive interface for both applicants and administrators, ultimately redefining the efficiency and precision of job application management.

1.1 Objective

The primary objective of the "Job Application Manager" project is to develop a user-centric web application that simplifies and centralizes the process of managing job applications. The aim is to create a platform that enables applicants to submit their information securely, view available job vacancies, and track the status of their applications seamlessly. Simultaneously, the project seeks to provide administrators with a practical tool for efficiently managing and reviewing applicant data, streamlining the process of vetting, approving, and rejecting applications. By leveraging web technologies, the project aspires to enhance the overall transparency, accessibility, and efficiency of the job application process for both applicants and administrators.

1.2 Scope

The scope of the "Job Application Manager" project encompasses the development of a comprehensive web-based application that facilitates the end-to- end management of

job applications. This scope includes the creation of user-friendly interfaces for both applicants and administrators, enabling functionalities such as secure data submission, real-time display of job vacancies, and transparent tracking of application statuses. Additionally, the project entails the implementation of robust backend systems for efficient data storage, retrieval, and management, ensuring the integrity and security of applicant information. The scope also encompasses the incorporation of suitable technologies for seamless integration, as well as the potential for future scalability and expansion. Furthermore, considerations for user authentication, authorization, and data protection form an essential part of the scope, emphasizing the project's commitment to upholding security and privacy standards. Ultimately, the project's scope revolves around delivering a reliable and intuitive web application that redefines the efficiency and effectiveness of job application management processes.

1.3 Motivation

The motivation behind the "Job Application Manager" project stems from the recognition of the increasingly digital landscape and the need for modern, streamlined solutions in job application processes. Traditional methods of managing job applications often lack efficiency, transparency, and ease of access for both applicants and administrators. The motivation to embark on this project lies in the desire to address these shortcomings by leveraging web technologies to create a cohesive, user-centric platform.

By developing the "Job Application Manager," the aim is to empower applicants to submit their details securely, track application statuses, and explore available job vacancies conveniently. Simultaneously, the project strives to equip administrators with a practical tool for efficiently managing applicant data, optimizing the vetting process, and enhancing the overall application management experience.

The project's motivation also centers on the broader impact it can have on reshaping the application management landscape, reducing manual processes, improving data accuracy, and fostering a more accessible and transparent application review

system.

Ultimately, the motivation for the "Job Application Manager" project is rooted in the potential to revolutionize and reinvigorate the job application management process through an innovative, technology-driven solution that addresses the needs of both applicants and administrators.

1.4 Development Needs & Importance

Development Needs: The system design and development phase encompasses several critical elements, including the creation of a user-friendly interface for seamless applicant interactions, efficient storage and retrieval of applicant data for administrators, robust security measures such as user authentication and secure data storage practices, real- time application status updates for enhanced transparency, and a flexible, scalable architecture to accommodate future enhancements and increasing user load. These components collectively contribute to the construction of a secure, efficient, and modern Job Application Manager system that meets the diverse needs of both applicants andadministrators.

Importance: The focus on efficiency aims to streamline the application review process, ultimately improving overall efficiency. Transparency is crucial, providing applicants with visibility into their application statuses, thus fostering trust. Ensuring the security and privacy of applicant data is paramount, as it serves to build trust with applicants. Accessibility is prioritized, aiming to enhance the experience for both applicants and administrators. Embracing web technologies is instrumental in modernizing the job application management process, ensuring the system remains aligned with industry standards and advancements.

CHAPTER 2

METHODOLOGY

The methodology for the Job Application Manager project consists of the following keypoints:

1. Requirement Analysis: Begin with a comprehensive analysis of the requirements, including the functionality needed for applicants and administrators, as well as any specific technical or security requirements.

2. Technology Selection: Determine the appropriate technologies and frameworks for both the frontend and backend development. Consider factors such as ease of use, security features, scalability, and support for real-time updates.

3. User Interface Design: Utilize user experience (UX) best practices to design an intuitive and user-friendly interface for applicants to submit and track their applications, and for administrators to manage applicant data and job vacancies.

4. Database Design: Design an efficient database structure for storing applicant data, job vacancies, and application statuses. Consider the need for data normalization, indexing, and security protocols for data storage.

5. Development and Testing: Adopt an agile development methodology to iteratively build and test the application. This involves creating modular components, performing unit testing, and addressing any technical debt promptly.

6. Deployment and Maintenance: Plan the deployment of the web application on local servers, ensuring scalability, availability, and data backup processes. Establish a maintenance plan for ongoing support, updates, and bug

2.1 Techniques Used

The development of the project webpage encompasses a range of techniques to achieve its objectives effectively:

1. **Responsive Web Design**: Implementing responsive web design techniques to ensure the website's compatibility and optimal viewing experience across various devices and screen sizes.

2. **Frontend Frameworks**: Utilizing modern frontend frameworks such as React.js to build dynamic and interactive user interfaces for seamless applicant interactions and administrator management.

3. **Backend Development**: Employing backend development using technologies like Node.js to create a robust, secure, and scalable server-side architecture for data management and application logic.

4. **Database Management**: Implementing database management techniques, such as relational database design using SQL databases, to efficiently store and retrieve applicant data, job vacancies, and application statuses.

5. **Status Updates**: Leveraging real-time web technology techniques to provide application status updates for applicants.

These techniques collectively contribute to the creation of a modern, secure, and user- friendly Job Application Manager web application, ensuring a seamless experience for both applicants and administrators while prioritizing data security.

2.2 Tools Used

In the development of the Job Application Manager project, a variety of tools were utilized to facilitate different aspects of the software development lifecycle. These tools encompassed areas such as project management, version control, database management, code editing, and testing. Below are some of the key tools used:

Version Control:

1. Git: Utilized for version control management, allowing for collaboration among team members, code tracking, and version control.
2. GitHub: Served as a centralized repository for storing the project's source code, facilitating collaboration and code review among team members.

Database Management:

1. MySQL Workbench: Used for designing, modeling, and managing the MySQL database schema, including creating tables, defining relationships, and executing queries.
2. phpMyAdmin: Employed for administering MySQL databases through a web interface, allowing for tasks such as database management, query execution, and data manipulation.

Code Editing:

1. Visual Studio Code: A versatile code editor used for writing and editing HTML, CSS, JavaScript, PHP, and other programming languages, with features such as syntax highlighting, code completion, and debugging support.

Backend Development:

1. PHP: Chosen as the primary backend programming language for server-side scripting, handling data processing, and interacting with the database.

Frontend Development:

1. HTML, CSS, JavaScript: Fundamental web development languages utilized for creating the frontend user interface, including layout, styling, and interactive behaviour.

Testing:

PHPUnit: A unit testing framework used for writing and executing unit tests for PHP code,ensuring the reliability and correctness of backend functionalities.

CHAPTER 3

Hardware and Software Requirements

3.1 Software Requirements

1. **Operating System**: Server: Linux (e.g., Ubuntu Server, CentOS) or Windows Server

2. **Web Server**: Apache, Nginx, or Microsoft Internet Information Services (IIS) Database Management System: MySQL, PostgreSQL, or MariaDB for storing applicant data and job vacancies.

3. **Programming Languages and Frameworks**: HTML, CSS, JavaScript for frontend development PHP, Python, Ruby, or Node.js for server-side scripting Frameworks like Laravel, Django, Flask, or Express.js for backend development

4. **Version Control**: Git for version control management (optional but recommended for collaborative development)

5. **Text Editor or Integrated Development Environment (IDE)**: Visual Studio Code, Sublime Text, Atom, or any preferred text editor/IDE for coding

6. **Dependencies and Libraries**: Any additional libraries or dependencies required by the chosen programming languages and frameworks

3.2 Hardware Requirements

1. Server:

- Processor: Multi-core processor (e.g., Intel Core i5 or equivalent)
- RAM: Minimum 4GB (8GB or more recommended for better performance)
- Storage: SSD preferred for faster read/write operations
- Network: Stable internet connection for hosting the application

2. Database Server (if separate):

- Similar specifications as the main server

- Adequate storage capacity for storing applicant data and job vacancies.

3. Client Devices:

- Desktops, laptops, tablets, or smartphones with modern web browsers
- Minimum screen resolution of 1024x768 pixels for optimal viewing

3.3 Functional Requirements

1. Applicant data submission and retrieval functionality.
2. Real-time application status updates for applicants and administrators.
3. Job vacancy display and management functionality.

3.4 Non - Functional Requirements

1. Security: Encryption of sensitive data, secure user authentication.
2. Performance: Scalability to handle increased user load.
3. Usability: Intuitive user interface and seamless navigation for applicants and administrators.
4. Minimal downtime and robust error handling

CHAPTER 4

SYSTEM DESIGN AND DEVELOPMENT

When designing and developing the "Job Application Manager" system, it's crucial to consider various aspects that contribute to its functionality, security, and usability. Here's a comprehensive outline of the content for the system design and development phase:

1. System Architecture:

The website consists of three sub-pages: Home, New Vacancies, Application Status. Home page is where user can apply for a job, provide his details and company preferences and submit his application to the company. New Vacancies tab gives details of available job posts from various companies. Application Status tab allows user to check if his application has been approved, rejected, or is under review.

2. Database Design:

The relationships between different data entities and the corresponding storage mechanismscan be delineated as follows:

a. Applicant Entity: The applicant entity encompasses the data fields including Name, qualifications, resume, address, highest education qualification, CGPA, College/University, Experience, skills, hobbies, and 4 preferred companies in preference order. This entity represents the core information submitted by individuals applying for job vacancies.

b. Storage Mechanism: The applicant entity's data can be stored in a structured manner within a database, where each field corresponds to a specific attribute or column. The use of a relational database, such as MySQL or PostgreSQL, allows for the establishment of relationships between different data entities. Storing applicant information in normalized tables facilitates efficient querying and retrieval as well.

c. **Relationships**: The relationships between the applicant entity and other relevant entities, such as Companies and Job Vacancies, can be established through foreign key constraints. For instance, the "preferred companies in preference order" field can be linked to the Companies entity to denote the applicant's choices, with each company represented as a distinct record in the Companies table. Similarly, the qualifications and skills of applicants can be associated with specific job requirements or skill sets through relational linkages.

d. File Storage: The "Resume" field, which typically involves storing binary data (PDF, Word, etc.), can be stored in a file system or a cloud-based storage solution. The system can store references or file paths within the database to retrieve the respective resumes.

By employing appropriate relational database design, including normalized tables and foreign key relationships, and utilizing file storage solutions for binary data, the Job Application Manager can effectively capture, manage, and interlink the diverse data entities submitted by applicants, contributing to a coherent and scalable data management approach.

3. User Interface Design:

Wireframes and mock-ups of the applicant interface for submitting applications and tracking statuses. Designs for the administrator interface managing applicant data and job vacancies.

4. Functional Components:

The functional components of the "Job Application Manager" system can be detailed as follows:

1. **Applicant Data Submission**:

This component allows applicants to submit their data, including personal information, qualifications, resume, skills, and preferences for companies and job vacancies. It involves a user-friendly interface enabling seamless data entry and validation to ensure accurate submission.

2. Application Status Updates:

The system provides real-time or periodic updates to applicants regarding the statuses of their applications. This involves notifications for application received, under review, shortlisted, rejected, or selected for interview. Applicants can access this information through their personalized dashboard or receive email notifications.

3. Job Vacancy Management:

Job vacancy management enables administrators to create, update, and remove job postings. This component includes functionality for specifying job requirements, skills, qualifications, and deadlines. It also facilitates the categorization and organization of vacancies based on departments, locations, or specific roles.

4. User Authentication:

User authentication ensures secure access to the system, allowing applicants and administrators to log in with unique credentials. It involves the verification of user identities and authorization levels, granting access to specific functionalities based on user roles. It also includes password management and security protocols to protect user accounts.

5. Scalability and Performance:

The scalability and performance of the Job Application Manager system are fundamental to its ability to accommodate growth in user base and ensure responsive functionality. Scalability pertains to the system's capability to handle increasing data loads and user traffic while maintaining optimal performance levels. This involves the seamless addition of resources, such as server capacity and storage, to meet growing demands without compromising system integrity. Additionally, performance optimization techniques, including efficient database queries, caching strategies, and load balancing, are employed to uphold responsiveness and minimize latency. By prioritizing scalability and performance, the Job Application Manager system can sustainably cater to a burgeoning user base, deliver swift application status updates, and uphold a seamless user experience.

6. Integration of Real-Time Updates:

The integration of real-time updates within the Job Application Manager system serves to enhance transparency, user engagement, and operational efficiency. By implementing real-time update mechanisms, the system can provide applicants with instantaneous notifications regarding the statuses of their applications, ensuring timely and transparent communication. Additionally, administrators can readily access real-time data insights, enabling swift decision-making and efficient management of job vacancies.

Leveraging technologies such as Web Sockets or server-sent events, the system can facilitate seamless communication between the server and clients, delivering immediate

updates without the need for manual refreshes. This approach empowers both applicants and administrators with up-to-date information, fostering trust, and ensuring a responsive and interactive user experience. The integration of real-time updates represents a significant advancement in modernizing the job application management process, aligning the system with contemporary user expectations and industry standards.

7. Deployment Plan

The webpage shall be hosted on a localhost, along with appropriate measures to handle all events and requests appropriately. The database shall be configured and data protection measures shall be implemented. The next step is resource allocation, and finally implementation.

4.1 Architectural Design

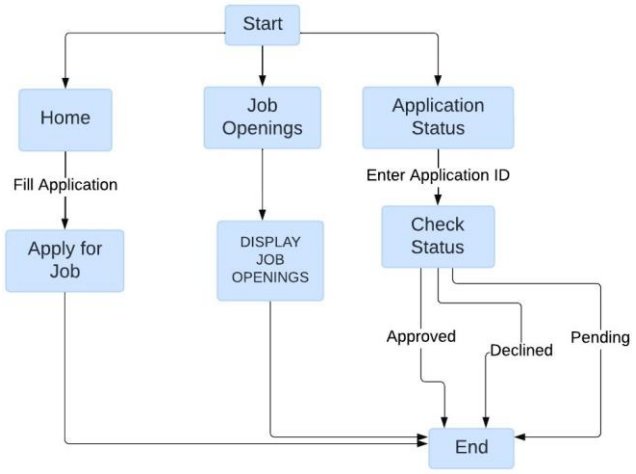

Figure 4.1 Flowchart

The architecture of the website can be summarized as follows: The primary page is the "Launch Page" where users have the option to navigate to the "Home," "Vacancies," or "New Application" sections. The "Home" section likely provides general information about the website or the company. The "Vacancies" section allows users to view available job positions. This section may also display the application status of each vacancy, indicating whether the application has been approved, declined, or is pending review. The "New Application" section likely provides a form or interface for submitting new job applications. Once an application has been submitted, it may undergo a review process where the status is updated accordingly. Additionally, the "New Application" section may

incorporate data about company preferences and demands, possibly related to the specific job vacancies. This architecture overview suggests a user-centric approach, providing easy access to key functionalities such as job application submission, vacancy exploration, and real-time status updates.

CHAPTER 5

IMPLEMENTATION

The implementation of the Job Application Manager involves several key steps, including setting up the development environment, designing the database schema, implementing frontend and backend functionalities, and testing the system for reliability and performance. Below is an outline of the implementation process:

1. Environment Setup:

- Install the necessary software components such as web server (Apache, Nginx), database management system (MySQL, PostgreSQL), and programming language interpreters (PHP, Python, Node.js).
- Configure the development environment to ensure compatibility and smooth operation of the software stack.

2. Database Design:

- Design the database schema to store applicant data, job vacancies, user accounts, andapplication statuses.
- Define the relationships between different entities using relational database concepts suchas primary keys, foreign keys, and indexes.
- Create tables, columns, and constraints based on the defined schema using SQLcommands or database management tools.

3. Frontend Development:

- Design the user interface layout and wireframes using HTML and CSS to create a visuallyappealing and intuitive interface.
- Implement frontend functionalities using JavaScript and relevant frameworks (e.g., React,Angular, Vue.js) to enable dynamic interactions and user input validations.
- Integrate design elements with backend functionalities to ensure seamless communicationbetween the frontend and backend components.

4. Backend Development:

- Develop server-side scripts and APIs using chosen programming languages (e.g., PHP,Python, Node.js) and frameworks (e.g., Laravel, Django, Express.js).

• Implement backend functionalities such as user authentication, data validation, databaseCRUD operations, and business logic for managing job applications and vacancies.

- Secure backend endpoints against common security vulnerabilities such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).

5. **Integration and Testing:**

- Integrate frontend and backend components to create a cohesive web application.
- Conduct unit tests to validate the functionality of individual modules and components.
- Perform integration tests to ensure proper communication between frontend and backend systems.
- Conduct user acceptance testing (UAT) to gather feedback from stakeholders and refine the application based on user preferences and requirements.

6. **Deployment:**

- Deploy the Job Application Manager on a web server (local or remote) using deployment tools such as FTP, SSH, or automated deployment pipelines (e.g., Jenkins, Travis CI).
- Configure server settings, permissions, and security measures to protect the application against unauthorized access and security threats.
- Monitor the deployed application for performance issues, errors, and scalability concerns, and optimize the system as needed to ensure smooth operation under varying workloads.

7. **Maintenance and Updates:**

- Regularly maintain and update the Job Application Manager to address securityvulnerabilities, bug fixes, and feature enhancements.
- Monitor user feedback and analytics data to identify areas for improvement and prioritize future development efforts accordingly.
- Keep abreast of emerging technologies and best practices in web development tocontinuously enhance the functionality and usability of the application over time.

CHAPTER 6

TESTING and RESULTS

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

In conclusion, the Job Application Manager project has successfully addressed the need for a centralized platform to streamline the job application process. Through the implementation of modern web technologies and careful attention to user experience, we have developed a robust system that facilitates the storage of applicant data, display of job vacancies, and tracking of application statuses. The project highlights the importance of effective collaboration, meticulous planning, and rigorous testing in software development, resulting in a high-quality product that meets the needs of both job seekers and employers.

Looking ahead, the Job Application Manager project offers ample opportunities for furtherenhancement and expansion. Some potential areas for future development include:

1. **Enhanced User Experience**: Continuously refining the user interface and incorporating user feedback to improve usability and accessibility.

2. **Advanced Search and Filtering**: Implementing advanced search and filtering options to help users quickly find relevant job vacancies based on criteria such as location, industry,and experience level.

3. **Integration with External Platforms**: Integrating with external platforms such as LinkedIn or Indeed to expand job listings and provide additional networking opportunities for users.

4. **Mobile Application**: Developing a mobile application to provide users with on-the-go access to job listings, application status updates, and other features.

5. **Analytics and Reporting**: Implementing analytics and reporting functionalities to provide insights into application trends, user behaviour, and recruitment metrics for employers.

6. **Social Media Integration**: Integrating social media sharing and login functionalities to facilitate user engagement and increase visibility of job listings.

7. **AI-Powered Features**: Incorporating artificial intelligence and machine learning algorithms to provide personalized job recommendations, resume optimization suggestions, and predictive analysis of application success rates.

8. **Internationalization and Localization**: Supporting multiple languages and currenciesto cater to a global user base and expand market reach.

By continuously iterating and innovating based on emerging trends and user feedback, the Job Application Manager project has the potential to evolve into a comprehensive solution that revolutionizes the job application process for users worldwide.