SYNOPSIS
Report on

Project Craft Connect

By

Shadman Ahmed 2300290140166
Nikunjay 2300290140109
Sahil Singh 2300290140152

Session:2024-2025 (III Semester)

Under the supervision of
Ms. Divya Singhal

KIET Group of Institutions, Delhi-NCR, Ghaziabad

DEPARTMENT OF COMPUTER APPLICATIONS
KIET GROUP OF INSTITUTIONS, DELHI-NCR, GHAZIABAD-201206
(2024-2025)

ABSTRACT

The Craft Connect Website is a comprehensive web-based solution that facilitates the recruitment process by connecting job seekers and employers on a single platform. The portal offers features such as job listings, candidate search, resume uploads, application tracking, and interview scheduling, making it easier for employers to find qualified candidates and for job seekers to discover job opportunities. The platform supports user authentication for both job seekers and employers, ensuring that profiles and job postings are managed securely. With an emphasis on user experience, the website incorporates filtering mechanisms for job search, notifications for new job postings, and employer-specific dashboards to manage job listings and applicants. Developed using the MERN stack (MongoDB, Express, React, Node.js), this project guarantees scalability, reliability, and responsiveness, supporting a large number of users concurrently. The platform also integrates with third-party APIs to offer additional functionalities like LinkedIn-based resume uploads and job matching algorithms, providing a modern and comprehensive solution for the recruitment industry.

TABLE OF CONTENTS

INTRODUCTION

The Craft Connect Website is designed as an all-in-one recruitment solution aimed at simplifying the hiring process for employers and the job search process for candidates. In today's fast-paced job market, job seekers need an efficient and convenient way to discover job opportunities, while employers require effective tools to manage and evaluate potential candidates. This job portal addresses both needs by offering a highly interactive platform that allows users to create profiles, apply for jobs, post job listings, and track applications with ease.

The platform incorporates the latest web technologies, utilizing the MERN stack for a fully responsive and scalable web application. MongoDB serves as the database for storing job listings, candidate information, and application data. Express and Node.js power the backend, providing secure and efficient routing and data handling, while React ensures a dynamic and engaging front-end experience for users. The project also places a strong emphasis on security, implementing robust authentication mechanisms for employers and job seekers using JWT (JSON Web Tokens).

The ultimate goal of the project is to provide a streamlined, user-friendly job portal that meets the needs of all stakeholders in the hiring process, from small businesses to large enterprises, and from entry-level job seekers to experienced professionals.

Literature Review

With the advent of digital technologies, job portals have become the go-to method for recruiting and job searching. Several platforms such as LinkedIn, Indeed, and Glassdoor dominate the market, offering varied features for both job seekers and employers. However, there remain certain challenges in the recruitment industry, including difficulties in filtering job candidates, lack of personalized job recommendations, and inadequate tools for managing the hiring pipeline.

Research has shown that user-friendly interfaces, personalized job recommendations, and secure profile management significantly enhance the job search experience. Existing platforms often fall short in areas of customization and ease of use for smaller employers and candidates seeking niche jobs. This project addresses those gaps by providing intuitive features such as advanced filtering options, personalized job alerts, and easy-to-use dashboards for both employers and job seekers.

Studies have also highlighted the importance of data security in job portals, as sensitive information like resumes and job applications need to be protected from unauthorized access. The Job Portal Website incorporates industry-standard security practices, such as data encryption, secure APIs, and role-based access control, to ensure that user data remains private and protected.

Project Objective

The primary objective of the Craft Connect Website is to develop an efficient and user-friendly online platform that connects job seekers and employers, streamlining the recruitment process. Key objectives include:

1. Connecting Job Seekers with Employers: Provide a robust platform that allows job seekers to discover relevant job opportunities and apply for positions based on their skills, experience, and preferences.

2. Improving the Hiring Process for Employers: Enable employers to post job listings, search through candidate profiles, and manage applications from a centralized dashboard.

3. Security and Data Protection: Implement secure authentication and data handling practices to protect user information and maintain the confidentiality of sensitive data.

4. Enhanced User Experience: Design a user interface that is intuitive and easy to navigate, making it simple for job seekers to search and apply for jobs, and for employers to review applications and schedule interviews.

5. Scalability and Performance: Develop the platform using the MERN stack to ensure that it can handle a large number of users concurrently without compromising performance.

6. Customization and Flexibility: Offer customization options for both employers and job seekers, including tailored job recommendations, job alerts, and candidate search filters.

Hardware Requirements:

1. **Server:**

   Processor (CPU): Multi-core, 2.5 GHz or higher.

   Memory (RAM): 16 GB or more.

   Storage: 500 GB SSD for fast data access.

   Bandwidth: 1 Gbps or higher for smooth user access.

2. **Backup and Redundancy:**

   Automated backups and redundancy systems to prevent downtime (e.g., failover setup, cloud-based backups).

3. **Database Server:**

   DBMS: MongoDB to store user data, job postings, and applications.

4. **Security Infrastructure:**

   Firewall: To prevent unauthorized access.

   SSL Certificates: For secure data transmission.

Software Requirements

1. **Operating System:**

   - A server-grade operating system like Linux (e.g., Ubuntu, CentOS) or Windows Server, chosen based on the server hosting provider's compatibility and system administrators' expertise.

2. **Web Server:**

   - A web server software such as Apache, Nginx, or Microsoft Internet Information Services (IIS) to serve web pages and handle HTTP requests.

3. **Database Management System (DBMS):**

   - MongoDb or an equivalent DBMS for creating and managing the relational database where project data and user information will be stored.

4. **Programming Languages:**

   - JavaScript for server-side scripting to build the web application's functionality.

   - HTML, CSS, and JavaScript for front-end web development.

5. **Development Frameworks:**

   - Bootstrap framework for creating responsive and user-friendly web interfaces.

6. **Security Measures:**

   JWT for secure authentication.

7. **Version Control:**

   - A version control system (e.g., Git) for tracking changes in the source code, enabling collaboration among developers, and facilitating code management.

8. **Development and Testing Tools:**

   - Integrated development environments (IDEs), code editors, and debugging tools for software development and testing.

9. **Monitoring and Analytics:**

   - Tools for monitoring server performance, error tracking, and user analytics to identify and resolve issues and improve the platform's efficiency.

10. **Documentation and Collaboration Tools:**

    - Documentation software (e.g., Confluence or Wiki) for maintaining project documentation.

These hardware and software requirements serve as the foundation for setting up the "Project Craft Connect" platform. They ensure the reliability, security, scalability, and performance of the system, creating a robust environment for facilitating project collaboration within educational institutions.

Proposed Time Duration

Week 1-2: Planning and Design

- Define project scope and objectives.

- Quickly create a basic system design.

- Select essential technologies and frameworks.

Week 3-6: Development and Testing

- Set up a minimal development environment.

- Code core functionalities.

- Perform rapid testing and debugging.

Week 7-8: User Testing and Refinement

- Start user testing with a small group.

- Gather feedback and make immediate adjustments.

- Continue refining features.

Week 9-10: Documentation and Deployment

- Create basic documentation.

- Deploy a core version with critical features.

- Configure essential security measures.

Week 11-12: Ongoing Maintenance

- Monitor platform performance and user feedback.

- Prioritize bug fixes and critical improvements.

- Continuously assess effectiveness.

This highly accelerated schedule focuses on delivering a functional core platform quickly, with plans for ongoing improvement based on user feedback and needs.

REFERENCES/ Bibliography

1. MongoDB Documentation. "MongoDB Manual", MongoDB, Inc. Available at: https://docs.mongodb.com/manual/

2. React.js Official Documentation. "A JavaScript library for building user interfaces", Meta (formerly Facebook). Available at: https://reactjs.org/docs/getting-started.html

3. Node.js Documentation. "Node.js v18.x Documentation", Node.js Foundation. Available at: https://nodejs.org/en/docs/

4. Express.js Guide. "Express.js Documentation", Express.js Foundation. Available at: https://expressjs.com/en/starter/installing.html

5. JWT Authentication. "JSON Web Token Introduction", Auth0. Available at: https://jwt.io/introduction/

6. Nginx Documentation. "Nginx: A High Performance Web Server", NGINX, Inc. Available at: https://nginx.org/en/docs/

7. API Testing with Postman. "Postman Learning Center", Postman, Inc. Available at: https://learning.postman.com/

8. Bootstrap Framework Documentation. "Bootstrap Documentation", Twitter, Inc. Available at: