**Final Project Report: Resume Builder using React js**

**Prepared by: Bytexl**

**Educator: Kajal Sharma**

Project Title: Resume Builder (Mern stack)

**Executive Summary**

The Resume Builder project, developed with React.js, is an intuitive and efficient tool that enables users to create, customize, and download professional resumes. Built to enhance user experience, it utilizes a component-driven architecture to deliver a dynamic, interactive interface that guides users through each section of their resume. The project features customizable templates, real-time previews, and a streamlined form input for sections such as personal information, work experience, education, skills, and more.

Keys

**User-Friendly Interface**: Responsive design with React ensures smooth, seamless navigation on both desktop and mobile devices.

**Real-Time Preview**: Provides immediate feedback on how the resume will look, improving usability and reducing the need for multiple downloads.

**Customizable Templates**: Offers various style and format options, allowing users to select designs that best reflect their professional image.

**PDF Download Capability**: Utilizes a third-party library to convert the final resume into a downloadable PDF, ensuring ease of sharing and printing.

**Reusable Components**: Leveraging React components, the codebase is modular and maintainable, facilitating quick updates and feature additions.

1. **Tools and Technologies Utilized**

**Frontend:**

* **React.js**
* Material UI and icons
* React-router-dom
* Spdf, for downloading resume in pdf
* Redux for state management
* React Router
* HTML5 & CSS3

**2. Project Overview :**

The Resume Builder application, utilizing React.js and Node.js, demonstrates effective use of modern web development practices. It successfully combines a responsive, user-friendly interface with scalable backend functionality, providing a robust solution for generating professional resumes. Areas for enhancement include performance optimization, advanced security features, and expanded PDF generation capabilities to meet professional-grade application standards.

This project not only highlights technical skills in JavaScript, React, and Node but also shows an understanding of user-centered design and full-stack application development.3. System Requirements

**3.Software Requirements**

- Node.js (with npm)

- React.js (latest version)

Compatible browser (Google Chrome or Mozilla Firefox)

Hardware Requirements

- Minimum RAM: 4GB

- Processor: Dual-core or better

- Operating System: 64-bit

**4. Functional Requirements**

* 1.User Registration and Authentication
* User Profile Management
* Resume Creation Workflow
* Real-Time Resume Preview
* 5.Export and Download Options
* 6.Resume Templates and Customization
* 7. Data Privacy and Security
* User Interface Requirements
* Interactive Help and Tooltips
* Security and Privacy Features
* Consistent Design and Theming
* Accessible and User-Friendly Design

6. Inputs and Outputs

Inputs: User data (personal info, education, work experience, skills, etc.), customizations (template choice, section order), user actions (add/remove sections, save drafts).

Outputs: Real-time resume preview, downloadable PDFs/DOCX files, user data stored in the backend for future edits, and analytics on user interactions.

7.Potential for Other Applications

The resume builder is not just a standalone tool for creating resumes.

It can be expanded into a comprehensive career management and job search platform, offering real-time feedback, networking, career growth, and personal branding features, all in one place such as

-Job Search & Application Management

-Skills Assessment & Certification

-Job Market Insights & Analytics

8. Test Case Design

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Case Description | Test Steps | Expected Result |  |  |  |
| TC001 | Verify that the user can add personal details in the resume builder. | 1. Open the resume builder.  2. Fill in name, email, phone number, and address fields.  3. Click "Save". | The personal details are saved successfully and displayed in the preview section. |  |  |  |
| TC002 | Verify that the user can add work experience to the resume. | 1. Navigate to the "Work Experience" section.  2. Fill in job title, company name, start and end dates.  3. Click "Save". | The work experience is added and displayed in the "Work Experience" section of the resume. |  |  |  |
| TC003 | Verify that the user can delete a work experience entry. | 1. Add a work experience.  2. Click on the "Delete" button next to the added experience. | The work experience is removed from the list and the preview section. |  |  |  |
| TC004 | Verify that the "Generate Resume" button creates a downloadable file. | 1. Fill out all the required sections (personal details, education, work experience).  2. Click "Generate Resume". | A PDF file containing the completed resume is generated and downloaded. |  |  |  |
| TC005 | Verify that an error message is shown when the user tries to generate a resume without filling out mandatory fields. | 1. Leave one or more mandatory fields empty.  2. Click "Generate Resume". | An error message appears asking the user to fill out all required fields. |  |  |  |
| TC006 | Verify that the user can edit an existing work experience entry. | 1. Add a work experience.  2. Click on the "Edit" button.  3. Modify job title, company name, or dates.  4. Save the changes. | The work experience is updated correctly, and changes are reflected in the preview section. |  |  |  |
| TC007 | Verify that the user can preview the resume before generating it. | 1. Fill out all sections.  2. Click on "Preview Resume". | The user can see a preview of the completed resume with all details filled in. |  |  |  |
| TC008 | Verify that the "Education" section accepts and saves multiple entries. | 1. Add multiple educational entries.  2. Fill in degree, institution name, start and end dates for each.  3. Click "Save". | All educational entries are saved and displayed in the preview section of the resume. |  |  |  |
| TC009 | Verify that the resume builder does not allow invalid email format in personal details. | 1. Enter an invalid email address (e.g., "invalid-email").  2. Try to save personal details. | The system prompts an error message stating "Invalid email address". |  |  |  |
| TC010 | Verify that the user can log out of the resume builder. | 1. Log into the resume builder application.  2. Click on the "Logout" button. | The user is logged out and redirected to the login screen. |  |  |  |

**9. Future Enhancements**

Future enhancements for a Resume Builder Application using React.js and Node.js can significantly expand its functionality, transforming it into a comprehensive career development platform. One of the key upgrades could be the integration of AI-driven resume optimization, where the system analyzes resumes in real-time and offers personalized feedback to improve content, formatting, and keyword usage based on job descriptions. Additionally, real-time collaboration features could be introduced, allowing multiple users, such as mentors or peers, to work together on a resume, providing seamless version control and feedback mechanisms. Incorporating job search and application tracking capabilities would allow users to directly apply to job listings, track application status, and even receive AI-based job recommendations based on their resume data.

**10. References**

-"Building Scalable Web Applications with React and Node.js"

-A Full-stack Application for Personal Resume Creation Using React.js and Node.js"

-"Building Interactive User Interfaces with React: A Focus on Dynamic Forms"

-Implementing PDF Generation in Web Applications Using React and Node.js"

**11. Project Reflection**

project demonstrated the power of full-stack development, allowing real-time data handling, personalized resume generation, and PDF export features. Looking ahead, integrating AI-driven suggestions and enhancing scalability for larger user bases could further elevate the app's usability and impact in the competitive job market.

Technical Challenges

-**State Management**

- **Cross-Browser Compatibility**

- **Backend Scalability**

- **Security and Authentication**

**12.Personal Development**

Resume Building Application helped me grow in problem-solving, user-centered design, and data security, while improving my collaboration and project management abilities in a team setting. This project has been a significant step in advancing both my coding expertise and understanding of real-world application development.