Resumate

[*] Note: Sub-titles are not captured in Xplore and should not be used

1st Saud
Dept. of BSc in Computer Science and Engineering (BSCSE)
North South University
Dhaka, Bangladesh
email: saud@northsouth.com

2nd Anika Bushra
Dept. of BSc in Computer Science and Engineering (BSCSE)
North South University
Dhaka, Bangladesh
email: anika@northsouth.com

3rd Sakia Tabassum Ananna
Dept. of BSc in Computer Science and Engineering (BSCSE)
North South University
Dhaka, Bangladesh
email: sakia@northsouth.com

4th Rima Saha
Dept. of BSc in Computer Science and Engineering (BSCSE)
North South University
Dhaka, Bangladesh
email: rima@northsouth.com

***Abstract*—The AI Resume Builder and Rating System is designed to assist users in creating professional resumes efficiently. The system integrates artificial intelligence (AI) for content generation and machine learning (ML) for evaluating resumes. It allows automatic scoring, feedback, and PDF exports while providing an installable interface via PWA or Electron.js. This project demonstrates the intersection of web development, AI, and practical software deployment in a junior-level academic setting.**

***Index Terms*—Resume Builder, Artificial Intelligence, Machine Learning, Web Development, PWA**

I. Introduction

In today's competitive job market, the quality and presentation of a resume often determine whether a candidate progresses to the next stage of recruitment. However, crafting a professional and optimized resume can be challenging for students and job seekers who lack industry experience or knowledge of best practices. Traditional resume builders provide basic formatting assistance but fail to evaluate content quality or provide actionable feedback.

The proposed system leverages artificial intelligence (AI) and machine learning (ML) to overcome these challenges by analyzing resumes for linguistic quality, keyword relevance, and structure. By integrating AI-generated suggestions through the Google Gemini API, the system empowers users to create more impactful and tailored resumes. Furthermore, the web application is developed using modern frameworks such as Laravel and Vue.js, ensuring scalability, responsiveness, and ease of deployment across platforms.

II. Problem Statement

Existing resume builders primarily focus on template formatting without providing content-based evaluation or feedback. Users are left uncertain about the effectiveness of their resumes in meeting job-specific requirements. Moreover, manual review processes are time-intensive and often subjective, leading to inconsistent results.

The proposed solution introduces an AI-based resume rating and feedback mechanism that can objectively analyze resumes using data-driven models. By implementing TF-IDF similarity measures and machine learning classifiers, the system compares user resumes against job description datasets to provide a quantitative rating. The tool not only saves time but also enhances fairness and transparency in resume evaluation.

III. Project Objectives

The project's main objectives include:

Identify applicable funding agency here. If none, delete this.

- Resume Creation: Develop an interactive and responsive interface for building resumes using Laravel Blade and Vue.js.
- AI Content Assistance: Integrate Google Gemini API to generate text suggestions for resume sections such as summary, skills, and experience.
- Resume Rating System: Implement ML models using scikit-learn and TF-IDF to evaluate resumes and provide automated scores.
- PDF Export Feature: Enable users to download formatted resumes through Laravel DOMPDF integration.
- Cross-Platform Deployment: Package the application as a Progressive Web App (PWA) and Electron.js desktop app for offline accessibility.
- Version Control Integration: Utilize Git and GitHub for project collaboration and version tracking.

These objectives collectively ensure that the system provides both technical robustness and user-friendly functionality.

IV. Methodology

The development process follows a modular architecture that separates the system into distinct layers — **frontend**, **backend**, **AI layer**, **and database**.

A. System Architecture

- Frontend Layer: Built using Laravel Blade templates and Vue.js, the interface allows users to input and edit resume data dynamically.
- Backend Layer: Powered by Laravel PHP, it handles authentication, API routing, and integration between the AI and database layers.
- AI Layer: The Google Gemini API assists in generating human-like content suggestions, while a local ML model (built with scikit-learn) computes similarity scores and evaluates resumes.
- Database Layer: A MySQL database stores user profiles, resumes, and evaluation results.
- Output Layer: Produces a numerical score and qualitative feedback, which can be exported as a PDF report.

B. Data Processing Workflow

- User input is tokenized and vectorized using TF-IDF to capture term relevance.
- The model computes semantic similarity between user resumes and predefined “ideal” datasets.
- Feedback is generated based on low-scoring segments, suggesting improvements in phrasing or keyword density.

This multi-layered approach ensures modularity, maintainability, and scalability for future enhancements.

V. Developer Stack

The system utilizes a combination of modern web frameworks, artificial intelligence tools, and deployment technologies to ensure scalability, responsiveness, and ease of maintenance. Table I summarizes the technologies used across different layers of the application architecture.

TABLE I
Developer Stack and Technologies Used

Layer	Technology	Functionality
Frontend	Laravel Blade / Vue.js / Bootstrap	Form design, UI logic
Backend	Laravel PHP	Application routing & control
Database	MySQL	Resume & score storage
AI Generation	Google Gemini API	AI content creation
AI Rating	Scikit-learn + TF-IDF	Resume evaluation
PDF Export	Laravel DOMPDF	Resume download
Installable App	PWA / Electron.js	Desktop-like experience
Version Control	Git / GitHub	Team collaboration

VI. Expected Outcomes

The expected deliverables of this project include:

- A fully functional AI-powered resume builder with a user-friendly interface.
- Automated resume scoring and qualitative feedback system.
- PDF export capability for finalized resumes.
- Installable desktop and web-based application with offline features.
- Demonstration of AI integration in real-world web development scenarios.

These outcomes demonstrate how AI-driven applications can enhance employability tools for students and professionals alike.

VII. Conclusion

The AI Resume Builder and Rating System illustrates how web development and artificial intelligence can converge to solve real-world problems in career preparation. By combining Laravel, machine learning, and cloud-based AI services, the project establishes a scalable foundation for intelligent resume generation. The integration of TF-IDF and content generation models ensures that each resume is evaluated not just on design, but also on relevance and content quality. This project exemplifies how AI technologies can augment productivity tools in education and employment

References

[1] Laravel Documentation, “Laravel Blade and Vue.js Integration,” 2024.
[2] T. Otwell, “Laravel Framework Documentation,” 2023.
[3] MySQL Development Team, “MySQL Reference Manual,” 2024.
[4] Google AI Studio, “Google Gemini API Documentation,” 2025.
[5] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[6] barryvdh, “Laravel DOMPDF Package,” 2024.
[7] Electron Documentation, “Building Desktop Apps with Electron.js,” 2025.
[8] Git SCM, “Git Version Control Documentation,” 2024.