Project Report Format

Date	**31 January 2025**
Team ID	**PNT2025TMID01602**
Project Name	**Smart Resume Generator**
Maximum Marks	**4 Marks**

1. INTRODUCTION

1.1 Project Overview

The **AI/ML-based Resume Generator** is a web-based application designed to help users create professional, ATS-compatible resumes quickly and efficiently. The tool uses AI and machine learning to automate keyword optimization, provide real-time grammar and style suggestions, and generate tailored resumes for specific job roles.

1.2 Purpose

The purpose of this project is to simplify the resume creation process, reduce the time and effort required to build professional resumes, and increase users' chances of passing Applicant Tracking Systems (ATS) and landing job interviews.

---

2. IDEATION PHASE

2.1 Problem Statement

Users struggle to create professional, ATS-compatible resumes tailored to specific job roles. This leads to time-consuming manual editing, generic resumes, and frequent rejections by ATS.

2.2 Empathy Map Canvas

- **Says**: "I don't know which keywords to include in my resume."
- **Thinks**: "Will my resume stand out to recruiters?"
- **Feels**: Frustrated and overwhelmed by the resume creation process.
- **Does**: Uses basic tools like Microsoft Word or free templates.

2.3 Brainstorming

- **Ideas**: AI keyword optimization, professional templates, real-time grammar checking, ATS compatibility checker, multi-language support.

- **Prioritized Features**: AI keyword optimization, resume templates, PDF generation.

---

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

1. **Awareness**: User learns about the resume generator through job portals or social media.
2. **Registration**: User signs up using email, Gmail, or LinkedIn.
3. **Resume Creation**: User inputs details, selects a template, and optimizes the resume using AI.
4. **Customization**: User tailors the resume for a specific job role.
5. **Export**: User downloads the resume as a PDF or DOCX file.
6. **Feedback**: User receives feedback on resume performance (e.g., ATS score).

3.2 Solution Requirement

- **Functional Requirements**:
    - User registration and login.
    - Resume creation with AI keyword optimization.
    - Resume customization and real-time suggestions.
    - Resume export (PDF/DOCX) and sharing.
- **Non-Functional Requirements**:
    - Usability: Intuitive, user-friendly interface.
    - Security: Encrypted data storage and secure authentication.
    - Performance: Fast resume generation (<5 seconds).

3.3 Data Flow Diagram

- **Level 0**: User → Resume Generator System → Job Portals.
- **Level 1**: User Registration, Resume Creation, Resume Customization, Resume Export.
- **Level 2**: AI keyword optimization, template selection, grammar checking.

3.4 Technology Stack

The AI/ML-based Resume Generator uses the following technologies:

- **User Interface:** Built using Streamlit, a Python-based framework for creating interactive web applications.
- **Backend Logic**: Developed in Python, handling user input processing and integration with the Google Generative AI API.
- **Generative AI:** Powered by the Google Generative AI API for generating professional summaries, bullet points, and tailored resume content.

- **Output:** Resumes are generated in PDF/DOCX format, with a download option for users.

---

4. PROJECT DESIGN

4.1 Problem Solution Fit

- **Problem**: Users struggle with ATS rejection and time-consuming resume creation.
- **Solution**: AI-powered resume generator with keyword optimization, professional templates, and real-time suggestions.

4.2 Proposed Solution

- An **AI-powered resume generator** that:
    1. Automates keyword optimization and resume formatting.
    2. Offers professional, ATS-compatible templates.
    3. Provides real-time grammar and style checking.
    4. Allows users to tailor resumes for specific job roles.

4.3 Solution Architecture

- **User Interface (Streamlit Web App):** Interactive web interface for user input and resume display.
- **Backend (Python Logic):** Processes user input and integrates with Google Generative AI API.
- **Output (Generated Resume):** Provides resumes in PDF/DOCX format for download.

---

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- **Phase 1 (Sprint 1)**: User registration, login, and basic resume creation.
- **Phase 2 (Sprint 2)**: AI keyword optimization and template selection.
- **Phase 3 (Sprint 3)**: Resume customization and real-time suggestions.
- **Phase 4 (Sprint 4)**: Resume export and sharing.
- **Phase 5 (Sprint 5)**: User dashboard and ATS compatibility checker.

---

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- **Testing Tools**: Selenium, JMeter.
- **Metrics**: Response time (<5 seconds), uptime (99.9%), scalability (10,000+ concurrent users).

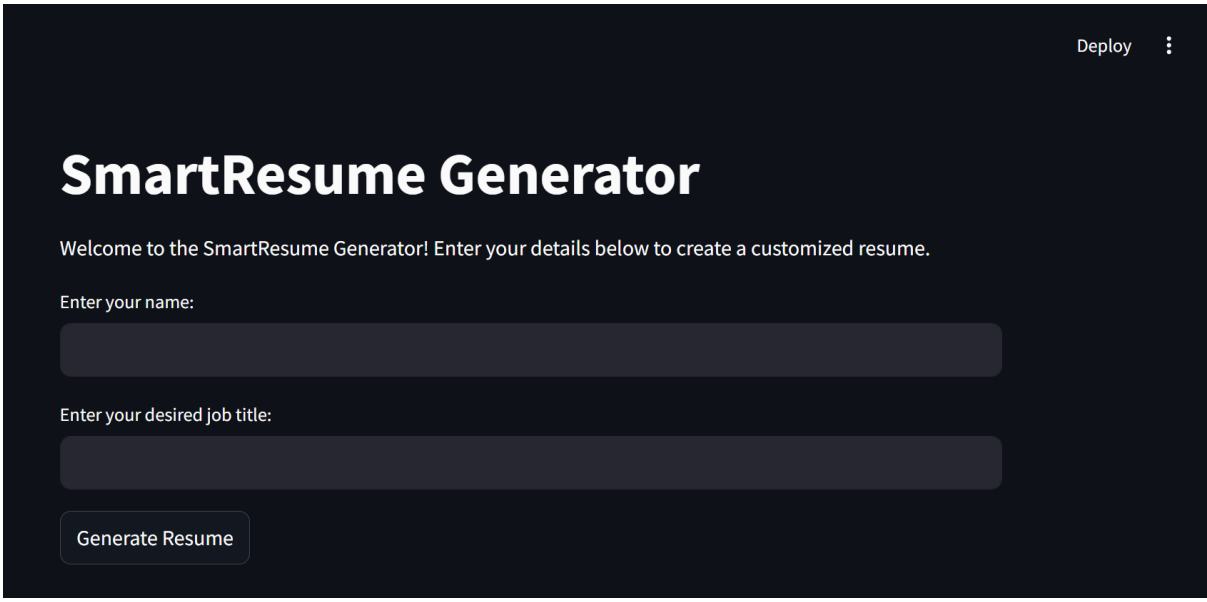
---

7. RESULTS

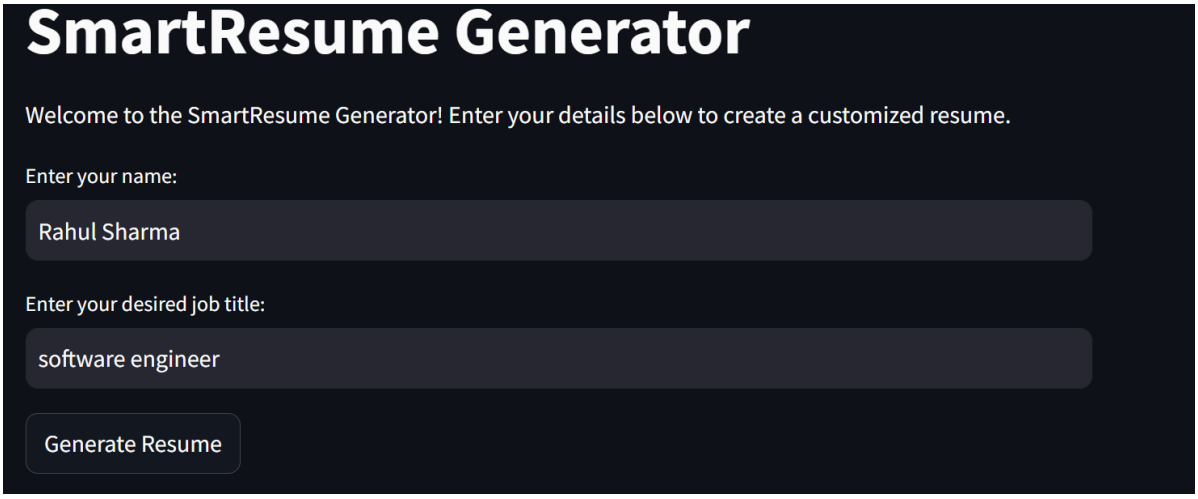
7.1 Output Screenshots

- **Screenshots**:

1. User Input Data



2. After Input require data to generate Resume



3. After Click on Generate Resume Button , generated resume shown below

Generated Resume:

Rahul Sharma +1234567890 | example@example.com | [Your LinkedIn Profile URL (Optional)] | [Your GitHub Profile URL (Optional)]

Summary

Highly motivated and results-oriented Software Engineer with [Number] years of experience in designing, developing, and deploying robust and scalable software applications. Proficient in [List 2-3 key technologies e.g., Java, Python, JavaScript] and experienced in working within agile development environments. Adept at collaborating effectively with cross-functional teams to deliver high-quality products on time and within budget. Seeking a challenging and rewarding Software Engineer position where I can leverage my skills and contribute to a dynamic and innovative organization.

Skills

Programming Languages: Java, Python, JavaScript, C++, SQL

Frameworks/Libraries: Spring Boot, React, Angular, Node.js, .NET (Optional, include if applicable), TensorFlow/PyTorch (Optional, include if applicable)

Databases: MySQL, PostgreSQL, MongoDB, Redis (Optional, include if applicable)

Cloud Platforms: AWS (e.g., EC2, S3, Lambda), Azure (Optional, include if applicable), Google Cloud Platform (Optional, include if applicable)

Tools & Technologies: Git, Docker, Kubernetes (Optional, include if applicable), Jenkins/CI/CD (Optional, include if applicable), REST APIs, Agile methodologies

Experience

[Previous Company Name], [City, State] [Job Title] | [Start Date] – [End Date]

- [Quantifiable achievement 1: e.g., Improved application performance by 15% by optimizing database queries.]
- [Quantifiable achievement 2: e.g., Developed and implemented a new feature that increased user engagement by 10%.]
- [Responsibility 3: e.g., Collaborated with a team of engineers to design and develop a new microservice architecture.]
- [Responsibility 4: e.g., Maintained and debugged existing codebase, resolving critical issues and improving code quality.]

[Previous Company Name (if applicable)], [City, State] [Job Title] | [Start Date] – [End Date]

- [Responsibility 1: e.g., Developed and tested software components using Java and Spring framework.]
- [Responsibility 2: e.g., Participated in daily stand-up meetings and sprint planning sessions.]
- [Responsibility 3: e.g., Contributed to the improvement of software development processes.]

Education

[University Name], [City, State] [Degree] in [Major], [Graduation Date]

[Relevant Certifications (Optional)] (e.g., AWS Certified Developer, etc.)

Projects (Optional – Include significant personal projects)

Education

[University Name], [City, State] [Degree] in [Major], [Graduation Date]

[Relevant Certifications (Optional)] (e.g., AWS Certified Developer, etc.)

Projects (Optional – Include significant personal projects)

- **[Project Name]:** [Brief description, technologies used, and link to GitHub repository (if available)]
- **[Project Name]:** [Brief description, technologies used, and link to GitHub repository (if available)]

Remember to replace the bracketed information with your own details and tailor the content to match the specific requirements of the job description. Use action verbs to start your bullet points and quantify your achievements whenever possible. Consider adding a portfolio section if you have a website or online portfolio showcasing your work.

Download Resume

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Saves time and effort in resume creation.
- Increases chances of passing ATS and landing interviews.
- Affordable and easy to use.

Disadvantages:

- Requires internet access.
- Limited customization for highly specialized roles.

---

9. CONCLUSION

The **AI/ML-based Resume Generator** successfully addresses the challenges of resume creation by automating keyword optimization, providing professional templates, and offering real-time suggestions. It is a valuable tool for job seekers, students, and professionals.

---

10. FUTURE SCOPE

- **Multi-Language Support**: Add support for resumes in multiple languages.
- **Cover Letter Generator**: Extend the tool to generate cover letters.
- **Integration with Job Portals**: Allow users to apply directly to jobs using their resumes.

---

11. APPENDIX

- **Source Code**: https://github.com/Rahul23Kumar58Yadav/Smart-Resume-Generator/upload
- **Dataset Link**: [Dataset Link]
- **Project Demo Link**: https://drive.google.com/file/d/1zmXx94JNhSgJCWhTbCi2hE1UuNFx76p/view?usp=drive_link