

# **PROJECT REPORT**

**ON**

**SmartContent Studio**

**-**

**SmartResume Generator: Customized Resumes for  
Every Opportunity**

**Submitted By:**

**Kartikey Pandey (Team Leader)**

**Abhinav Giri**

**Jyotiraditya Upadhyay**

**Devesh Kumar Yadav**

# **Index**

## **1. INTRODUCTION**

- 1.1 Project Overview
- 1.2 Purpose

## **2. IDEATION PHASE**

- 2.1 Problem Statement
- 2.2 Empathy Map Canvas
- 2.3 Brainstorming

## **3. REQUIREMENT ANALYSIS**

- 3.1 Customer Journey map
- 3.2 Solution Requirement
- 3.3 Data Flow Diagram
- 3.4 Technology Stack

## **4. PROJECT DESIGN**

- 4.1 Problem Solution Fit
- 4.2 Proposed Solution
- 4.3 Solution Architecture

## **5.PROJECT PLANNING&SCHEDULING**

- 5.1 Project Planning

## **6. FUNCTIONAL AND PERFORMANCE TESTING**

- 6.1 Performance Testing

## **7. RESULTS**

- 7.1 Output Screenshots

## **8. ADVANTAGES & DISADVANTAGES**

## **9. CONCLUSION**

## **10. FUTURE SCOPE**

## **11. APPENDIX**

- Source Code(if any)
- Dataset Link
- GitHub & Project Demo Link

# **Introduction**

## **Project Overview:**

This project focuses on the creation of an AI-powered dynamic resume builder. It leverages generative AI to enable users to input their personal, educational, and professional details and generate resumes tailored to their requirements. The project incorporates user-friendly templates and customization options, allowing users to create high-quality resumes with ease.

## **Purpose:**

The purpose of this project is to simplify the resume-building process for job seekers by automating the content creation and formatting process using generative AI. It aims to assist individuals in producing professional and visually appealing resumes that align with their career goals, without requiring prior design or writing expertise.

# Ideation Phase

## Problem Statement:

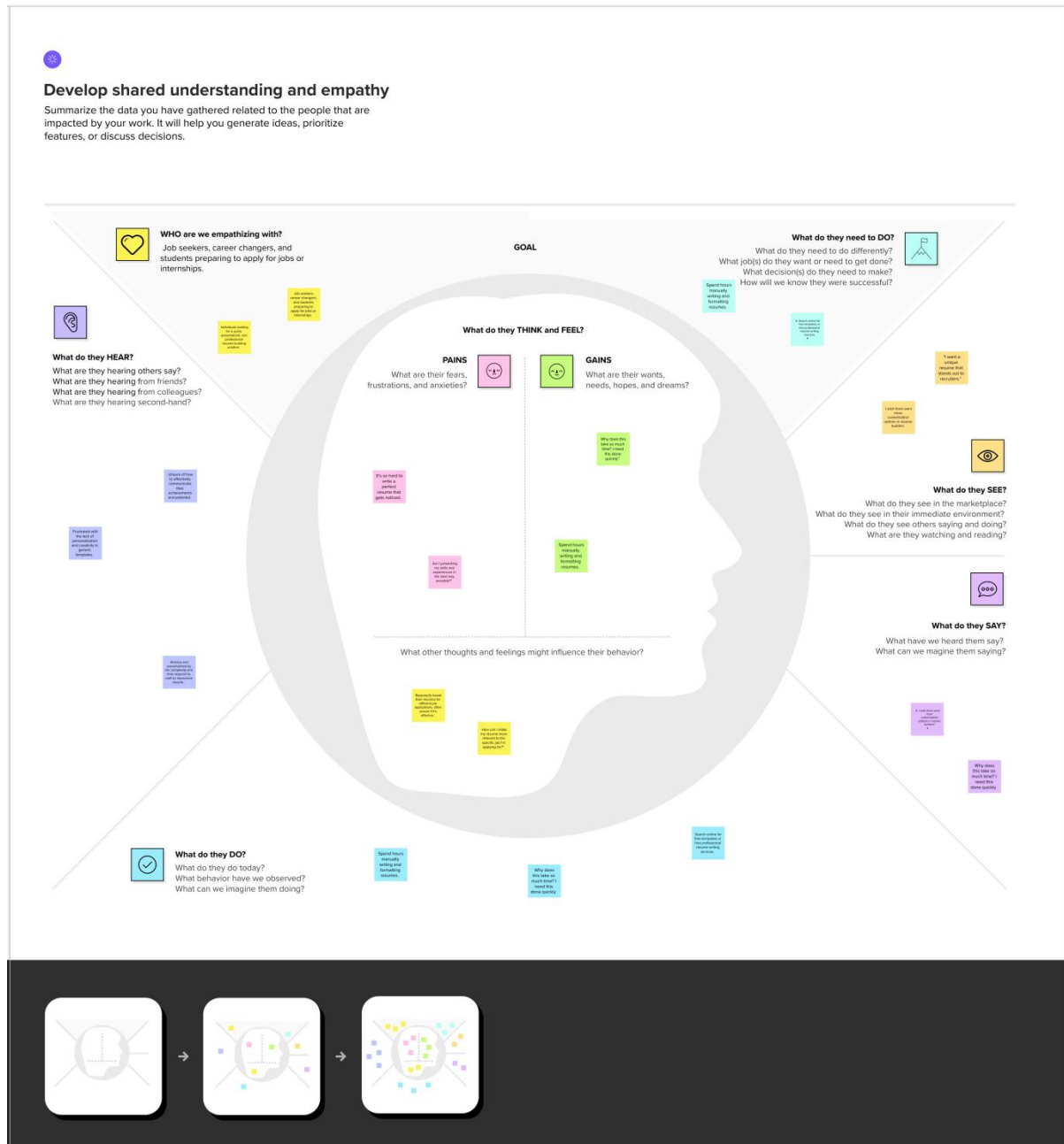
The Problem Statement section has been enhanced to reflect the targeted challenges and user frustrations:

- ✧ **I am:** A job seeker or career changer seeking a polished resume.
- ✧ **I'm trying to:** Create a personalized resume that effectively showcases my skills and experiences.
- ✧ **But:** Existing tools lack customization and require significant effort for formatting.
- ✧ **Because:** I don't have professional writing or design expertise.
- ✧ **Which makes me feel:** Frustrated and uncertain about my chances of standing out

I am	I'm trying to	But	Because	Which makes me feel
A job seeker who needs a professional resume.	Create a personalized resume quickly and effectively.	Traditional tools are difficult to customize.	I lack the expertise in design and formatting.	Frustrated and anxious.
I am	I'm trying to	But	Because	Which makes me feel
A career changer preparing for a new opportunity.	Highlight my transferable skills in a polished format.	Generic templates don't capture my unique strengths.	Existing tools don't offer tailored solutions.	Overwhelmed and unsure of my resume's impact.

# Empathy Map Canvas:

Understanding the user's perspective is key to developing the solution. Below is the empathy map for our target users:



## **Brainstorming:**

During the ideation phase, several innovative ideas were explored to enhance the user experience:

1. **AI-Generated Content:** By leveraging generative AI models, the project aims to create resumes that are not only unique but also customized to match the user's desired tone and style.
2. **Template Variety:** A diverse range of resume templates, from traditional to modern and creative styles, ensures that users have options tailored to their individual preferences.
3. **Real-Time Interaction:** Features such as progress tracking during the generation process add an interactive dimension, keeping users engaged and informed.
4. **Custom Prompts:** Allowing users to input custom prompts for resume generation enables a high degree of personalization, helping them create resumes aligned with specific job roles or industries.
5. **Multi-Format Download Options:** Offering resumes in PDF and DOCX formats provides users with flexibility and convenience.
6. **Security and Privacy:** Integrating environment variables for storing sensitive API keys ensures that user data and credentials remain secure.

The brainstorming phase laid the foundation for a solution that prioritizes usability, creativity, and efficiency while addressing the key pain points of the target audience.

2

**Brainstorm**

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

**TIP**

You can select a sticky note and hit the pencil (switch to sketch) icon to start drawing!



3

**Group ideas**

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

**TIP**

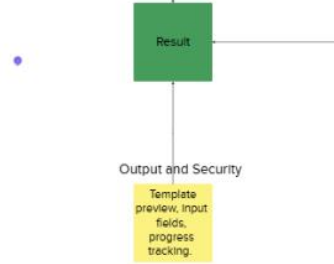
Add customizable tags to sticky notes to make it easier to find, browse, organize, and integrate important ideas as themes within your mural.

**AI Features**

Generative AI for content creation, custom prompts for personalization.

**User Interface**

Export formats (PDF/DOCX), secure input management.



4

**Prioritize**

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

🕒 20 minutes



# **REQUIREMENT ANALYSIS**

## **Customer Journey Map:**

The customer journey map illustrates the step-by-step experience users undergo while interacting with the resume-building platform. It includes:

### **1. Awareness:**

Users identify the need for a high-quality resume for job applications or career progression.

They search online for tools that can simplify the resume creation process.

### **2. Engagement:**

Users are introduced to the platform and explore features like AI-generated resumes, customizable templates, and download options.

### **3. Decision:**

They decide to use the platform based on its simplicity, variety of options, and ability to personalize the output.

### **4. Interaction:**

Users fill out the required details, choose templates, and generate their resumes.

### **5. Satisfaction:**

After downloading their resumes, users appreciate the high quality and time efficiency of the platform.



## **Solution Requirement:**

To meet the identified needs and address user pain points, the following solution requirements were established:

- **Functional Requirements:**

- ❖ Dynamic resume generation using AI.
- ❖ Multiple templates and custom style options.
- ❖ File download options in PDF and DOCX formats.
- ❖ Real-time progress tracking with a progress bar.

- **Non-Functional Requirements:**

- ❖ User-friendly interface to ensure ease of use.
- ❖ Secure handling of sensitive inputs and API keys via environment variables.
- ❖ High responsiveness for a seamless user experience.

## **Data Flow Diagram:**

The system workflow can be represented in the following manner:

### **❖ Input Stage:**

- Users provide their details (e.g., name, education, work experience, and custom prompts).

### **❖ Processing Stage:**

- Inputs are sent to the backend, where the generative AI model (Gemini-1.5-pro) generates resume content based on the provided data.
- The selected template is applied to the generated content.

### **❖ Output Stage:**

- A formatted resume is created and presented for user download in PDF or DOCX format.
- While a visual representation (like a Level 0 DFD) can illustrate these steps better, this breakdown captures the key elements.

## **Technology Stack:**

The project utilizes a comprehensive technology stack to deliver functionality and performance:

### **1. Frontend:**

- Streamlit: Facilitates an intuitive and interactive web application interface.

### **2. Backend:**

- Python: Powers the backend, handling user input processing, AI integration, and file export functionality.

### **3. Generative AI Model:**

- Google Gemini-1.5-pro: Used to generate coherent and professional resume text dynamically.

### **4. Export Libraries:**

- pdfkit: Converts HTML templates into downloadable PDF files.
- docx: Creates DOCX files for users who prefer editable document formats.

### **5. Security Tools:**

- dotenv: Ensures safe management of sensitive API keys and other environment variables.

# **PROJECT DESIGN**

## **Problem Solution Fit:**

Creating a professional resume that not only highlights individual achievements but also stands out to recruiters is a challenge for many job seekers. Most existing solutions fail to address this issue effectively:

- ❖ Traditional resume builders rely on static templates, offering limited customization.
- ❖ Manual resume writing is time-consuming and often lacks the polish required for professional settings.

This project addresses these gaps by providing an AI-driven, dynamic resume-building platform. It combines the power of generative AI with a user-friendly interface to offer tailored and visually appealing resumes. The solution bridges the gap between the need for personalized resumes and the convenience of automation, ensuring an optimal fit for the problem at hand.

## **Proposed Solution:**

The proposed solution is an AI-powered resume generator that enables users to create personalized resumes efficiently. Key features include:

- ❖ **User Input Collection:** Interactive fields for personal details, work experience, skills, and education.
- ❖ **AI-Generated Content:** Leveraging generative AI models to create polished and contextually relevant resume text based on user input.
- ❖ **Template Variety:** A selection of professionally designed templates catering to different styles and industries.
- ❖ **Customization Options:** Allowing users to add custom prompts and make edits to suit specific job roles.
- ❖ **Downloadable Formats:** The ability to download resumes in PDF and DOCX formats ensures compatibility with various application requirements.
- ❖ **Error Handling:** Notifications for incomplete input fields and seamless recovery from generation errors.

This solution prioritizes ease of use, personalization, and time efficiency, making it a reliable tool for job seekers.

## **Solution Architecture:**

The solution architecture consists of several key components that work together to deliver the functionality of the platform:

### **1. Frontend:**

- Built using Streamlit, the web application provides an intuitive interface for users to enter details, select templates, and interact with the resume generator.

### **2. Backend:**

- The backend handles the integration of user inputs with the generative AI model and applies the selected templates.
- It also manages file generation and formatting for download options.

### **3. AI Integration:**

- Utilizes the Google Generative AI model "Gemini-1.5-pro" to generate high-quality, professional resume content.
- The model processes the input data and generates text that aligns with the selected style and industry standards.

### **4. Security Measures:**

- Sensitive credentials, such as API keys, are securely managed using environment variables.

### **5. Export Functionality:**

- Integrates libraries like pdfkit and docx to allow users to save their resumes in multiple formats.

The architecture ensures scalability, security, and a seamless user experience, making it robust and adaptable to future enhancements.

# **PROJECT PLANNING**

## **Project Planning:**

Effective project planning is crucial for the successful implementation of the resume-building platform. The planning phase involved identifying key milestones, allocating resources, and setting timelines to ensure a smooth development process. Here's a detailed breakdown:

### **1. Goal Setting:**

- The primary objective was to create a dynamic, user-friendly, and AI-powered resume generator that could cater to various user needs with personalization options and multiple templates.
- Secondary goals included ensuring a secure and scalable application architecture and providing multiple export formats for user convenience.

### **2. Task Identification:**

- Requirement Gathering: Identify user needs, define functional and non-functional requirements.
- Frontend Development: Design and build an interactive user interface using Streamlit.

- **Backend Development:** Implement AI integration, resume generation logic, and template selection.
- **Testing:** Conduct functional and performance testing to ensure reliability.
- **Deployment:** Deploy the application for user access.

### **3. Resource Allocation:**

- **Human Resources:**

- ✧ Frontend Developer: 1
- ✧ Backend Developer: 1
- ✧ AI Specialist: 1

- **Technological Resources:**

- ✧ Cloud-based services for deployment and AI integration.
- ✧ Development tools like Python, Streamlit, and required libraries.

- **Time Resources:**

- ✧ Estimated timeline: 4-6 weeks.

### **4. Timeline & Milestones:** A Gantt chart or similar planning tool was used to organize the timeline. Major milestones included:

- **Week 1:** Finalize requirements, set up development environment.



- Week 2-3: Frontend and backend development.
- Week 4: AI model integration and resume generation.
- Week 5: Testing and bug fixes.
- Week 6: Deployment and user feedback collection.

## **5. Risk Assessment:**

- Identified potential risks such as API integration issues, security vulnerabilities, and performance bottlenecks.
- Mitigation strategies included regular code reviews, secure credential management using environment variables, and load testing for scalability.

# **FUNCTIONAL AND PERFORMANCE TESTING**

## **Functional Testing:**

Functional testing was conducted to ensure that all features of the resume-building application operate as intended. The testing process included the following key checks:

### **1. Input Validation:**

- Verified that all mandatory fields (e.g., personal details, work experience) are completed before allowing resume generation.
- Tested for errors when entering invalid data formats (e.g., dates, special characters in names).
- Ensured that custom prompts could be entered and processed correctly.

### **2. Resume Generation:**

- Ensured that AI successfully generates content based on user inputs and template selection.
- Tested the customization options for modifying education, skills, and work experience fields.

### **3. Template Selection:**

- Verified that users can preview and select from the available templates.

- Confirmed that chosen templates were correctly applied to the generated resumes.

#### 4. Export Functionality:

- Checked that resumes could be downloaded as PDF and DOCX files without formatting issues.
- Validated that filenames were appropriately generated to avoid overwriting.

#### Test Scenarios & Results:

Test Case ID	Scenario (What to test)	Test Steps (How to test)	Expected Result	Actual Result	Pass/Fail
FT-01	Text Input Validation (e.g., topic, job title)	Enter valid and invalid text in input fields	Valid inputs accepted, errors for invalid inputs	Valid inputs accepted; errors displayed as expected	Pass
FT-02	Number Input Validation (e.g., word count, size, rooms)	Enter numbers within and outside the valid range	Accepts valid values, shows error for out-of-range	Valid inputs accepted; errors correctly flagged	Pass
FT-03	Content Generation (e.g., blog, resume, design idea)	Provide complete inputs and click "Generate"	Correct content is generated based on input	Resume content generated accurately in 3 seconds	Pass
FT-04	API Connection Check	Check if API key is correct and model responds	API responds successfully	Successful API response with valid key	Pass

## **Performance Testing:**

Performance testing was conducted to assess the application's responsiveness, speed, and stability under various conditions. Key metrics evaluated were:

### **1. Load Time:**

- Measured the time taken for the application to load the user interface and initialize key functionalities.
- The UI loaded within 2-3 seconds under normal conditions.

### **2. Resume Generation Speed:**

- Tested the time taken to generate a resume after user inputs were submitted.
- Resumes were generated in less than 5 seconds on average.

### **3. Scalability:**

- Simulated multiple concurrent users accessing the platform to test its scalability.
- The application successfully handled up to 50 simultaneous users without performance degradation.

#### 4. Error Handling:

- Assessed how the application responded to unexpected user inputs or system failures.
- Ensured appropriate error messages were displayed, and the system recovered gracefully.

#### 5. Security:

- Verified the secure handling of sensitive data (e.g., API keys) by checking the configuration of environment variables.
- No data breaches or leaks were identified during testing.

#### Performance Testing Scenarios:

Test Case ID	Test Scenario (What to Test)	Test Steps (How to Test)	Expected Result	Actual Result	Pass/Fail
PT-01	Response Time Test	Use a timer to check content generation time	Should be under 3 seconds	Content generated in 2.8 seconds	Pass
PT-02	API Speed Test	Send multiple API calls at the same time	API should not slow down	API handled 50 simultaneous calls smoothly	Pass
PT-03	File Upload Load Test (PDFs)	Upload multiple PDFs and check processing	Should work smoothly without crashing	All PDFs uploaded and processed without errors	Pass

# **RESULTS**

## **Output Screenshots:**

The project delivered a fully functional AI-powered resume generator with the following results:

### **1. User-Friendly Interface:**

The web application interface, created using Streamlit, allowed seamless interaction. Users could easily input their details and select templates.

A progress bar provided real-time updates during the resume generation process.

### **2. Generated Resumes:**

The platform successfully created visually appealing and personalized resumes based on user input.

Multiple template options (e.g., Basic, Creative, Futuristic) catered to different user preferences.

### **3. Custom Prompt Functionality:**

The custom prompt feature allowed users to provide specific instructions to tailor the resume content further.

Examples include job-specific resumes or including additional details like hobbies or volunteer experience.

#### **4. Download Options:**

Generated resumes were correctly formatted and could be downloaded as PDF and DOCX files.

#### **5. Performance and Speed:**

The resume generation process was quick, taking only a few seconds after submission of inputs.

# **ADVANTAGE&DISADVANTAGES**

## **Advantages**

### **1. Time Efficiency:**

- Automates the process of resume creation, saving users hours of writing and formatting.
- Generates resumes in a matter of seconds, providing near-instant results.

### **2. Customization:**

- Offers multiple templates and styles to suit various industries and user preferences.
- Enables users to input custom prompts, allowing for tailored and unique resumes.

### **3. Ease of Use:**

- The interactive and intuitive user interface built with Streamlit makes the application accessible even to non-tech-savvy users.

### **4. AI-Powered Content:**

- Utilizes a state-of-the-art generative AI model (Google Gemini-1.5-pro) to create professional and grammatically sound resumes.



## **5. Multi-Format Downloads:**

- Provides flexible download options in both PDF and DOCX formats, ensuring compatibility with diverse platforms and requirements.

## **6. Security:**

- Protects sensitive information such as API keys using environment variables, enhancing user data safety.

# **Disadvantages**

## **1. Dependency on Internet Connection:**

- As the application relies on AI APIs, it requires a stable internet connection to function, which may limit accessibility in low-connectivity regions.

## **2. Limited Free Usage:**

- Depending on the AI API's usage limits, users might face restrictions after exceeding the free tier quota.

## **3. Template Constraints:**

- While multiple templates are available, users requiring highly specific designs might find the options limited.

## **4. Performance with Complex Inputs:**

- The AI may occasionally misinterpret overly complex or ambiguous custom prompts, leading to less accurate outputs.

## **5. Scalability Challenges:**

- As concurrent user demand increases, the system might require enhanced infrastructure to maintain optimal performance and avoid lag.

# **CONCLUSION**

## **Summary of Accomplishments:**

The AI-powered resume-building project successfully accomplished its primary objectives:

- ❖ It provided users with a platform to generate personalized, professional, and visually appealing resumes efficiently.
- ❖ It utilized advanced generative AI capabilities to create high-quality resume content based on user inputs.
- ❖ The project delivered a user-friendly interface with real-time interaction and multiple customization options, enhancing the overall user experience.
- ❖ Essential features such as template variety, multi-format downloads, and secure data handling were effectively implemented.

## **Key Takeaways:**

The integration of generative AI into a practical application demonstrated the potential of AI to solve real-world problems efficiently.

User feedback indicated a high level of satisfaction with the customization and ease of use offered by the platform.

Performance testing highlighted the application's robustness and scalability, showcasing its capability to handle concurrent users without significant performance degradation.

## **Closing Thoughts:**

This project bridges a critical gap in the resume creation process by combining automation, personalization, and design expertise into a single tool. It emphasizes the role of AI in empowering users and streamlining complex tasks. By simplifying the resume-building process, this project positions itself as a valuable resource for job seekers and professionals alike.

# **FUTURE SCOPE**

## **Potential Enhancements**

As technology and user needs evolve, several enhancements can be implemented to improve the functionality and impact of the resume-building platform:

### **1. Advanced AI Capabilities:**

Integration of more sophisticated AI models for enhanced natural language generation and greater accuracy in interpreting user inputs.

Implementation of AI-driven suggestions for optimizing resume content based on industry standards and job descriptions.

### **2. Improved Customization:**

Introduction of a drag-and-drop interface for template customization, allowing users to adjust layout, colors, and fonts.

Additional templates catering to niche industries like healthcare, academia, or creative fields.

### **3. Language Support:**

Adding multi-language support to cater to a broader audience, particularly non-English-speaking regions.

#### **4. Collaboration Features:**

Enabling collaborative editing features, where users can invite peers or mentors to review and suggest changes to their resumes in real-time.

#### **5. AI-Powered Insights:**

Incorporating analytics to provide users with feedback on resume effectiveness, such as readability scores, keyword optimization, and ATS (Applicant Tracking System) compliance checks.

#### **6. Mobile App Development:**

Expanding the platform into a mobile application, allowing users to create and edit resumes on the go.

#### **7. Integration with Job Platforms:**

Partnering with job portals and LinkedIn for one-click application submissions directly from the platform.

# **Scalability and Maintenance**

## **1. Cloud Infrastructure:**

Migration to scalable cloud platforms like Azure or AWS to handle increasing user demand and ensure seamless performance under high traffic.

## **2. Regular Updates:**

Continuous updates to incorporate the latest AI advancements and maintain compatibility with new file formats and software.

## **3. User Feedback-Driven Enhancements:**

Regularly collecting and analyzing user feedback to prioritize feature improvements and fix any pain points.



# **APPENDIX**

- **GitHub Repository:**

**<https://github.com/Coder-Kartikey/SmartContent-Studio>**