

# **CareerCraft: ATS-Optimized Resume Analyzer using Gemini Model**

**Team ID:** SWTID1719935402

**Team Members:**

Rueben Varghese Philip

Varun Donthi

Aryaman Abbi

Jayanth G

**Date:** 19/07/2024

# INDEX

S.No.	Contents	Page Number
1	Introduction	3
2	Project Initialization and Planning Phase	5
3	Model Development Phase	7
4	Model Optimization and Tuning Phase	11
5	Results	12
6	Advantages & Disadvantages	13
7	Conclusion	14
8	Future Scope	15
9	Appendix	16

# 1. Introduction:

## 1.1 Project Overview

An ATS (Applicant Tracking System) optimized resume analyser using the Gemini API model is designed to improve the likelihood of a resume passing through an ATS and reaching a human recruiter. Many companies use ATS to filter resumes before they reach hiring managers. An ATS-optimized resume analyser ensures that resumes are formatted and contain keywords that make them ATS-friendly. By using an AI model like Gemini, the process of optimizing and analysing resumes becomes faster and more accurate.

### Key Features:

The Gemini API can identify and suggest relevant keywords based on the job description and industry standards. It can provide guidance on how to format resumes to ensure they are ATS-compliant (e.g., using standard fonts, avoiding images, etc.). Resumes are scored based on their alignment with job descriptions, and specific feedback is provided to improve the score. Additionally, the analyser checks the resume's compatibility with various ATS software systems, ensuring a higher chance of passing the initial filter.

### How It Works:

The user uploads their resume and inputs the job description. The Gemini API analyses the resume, comparing it against the job description and ATS requirements. The tool then provides a score, feedback, and suggestions for improvement.

### Benefits:

This tool increases the chances of the resume reaching a recruiter by ensuring it passes through ATS filters. It automates the optimization process, saving time for job seekers. Moreover, it tailors the resume to specific job descriptions, enhancing its relevance.

### Use Cases:

Job seekers looking to improve their resume's chances of passing ATS filters can benefit from this tool. Career coaches can assist clients in optimizing their resumes, and recruitment agencies can ensure their candidates' resumes are ATS-compatible.

## **1.2. Objectives**

The objective of this project is to create a functional model of an ATS-optimized resume analyzer that takes advantage of Gemini. This prototype will allow users to upload resumes and receive analysis reports tailored to Applicant Tracking Systems, thereby increasing the efficiency and effectiveness of the recruitment process.

## 2. Project Initialization and Planning Phase

### 2.1 Problem Statement

In today's job market, a big issue for job seekers is developing resumes that not only highlight their qualifications but also pass through Applicant Tracking Systems. Employers utilise ATS software to speed up the recruiting process by sorting and rating resumes based on keywords, formatting, and other algorithmic factors. However, due to poorly optimised resumes, these methods frequently result in the rejection of many eligible individuals.

Job seekers, particularly those inexperienced with the specifics of ATS, struggle to ensure that their resumes are compatible with these systems. Common problems include incorrect formatting, a lack of relevant keywords, and structural errors, which can lead to their applications being screened out before they reach a person. This is a considerable barrier to employment, since even highly skilled people may not be considered for positions owing to technical errors in their resumes.

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
I'm responsible for collecting and analyzing the resume data that's been used for the hiring process of the company I'm working currently.	Resume Analyzer	Ensure to get the best information out of the resume data accurately that is relevant to the job description.	The analyzing software doesn't perform it's right.	Inaccuracies leading to confusion.	Finding difficulty in finding the right person for the job.
I'm responsible for overseeing the workforce whom we hired to work for the company.	Employer	Get good job performance and improve workforce.	Low job performance.	Some employees are not performing well at their	Should have made a better choice and it would have been beneficial for the company.

				job, most likely due to the poor choice of candidate during interview process.	
I'm responsible for analyzing the potential of the employee for being useful to the company.	HR Representative	Maintain good workplace environment by hiring good employees who have good attitude.	Some employee do seem to misbehaving resulting a nuisance in the workplace environment.	Due to bad attitude.	Should have made a better choice by observing their attitude and it would have been beneficial for the company.
I'm responsible for finding jobs and hoping to get a job in an esteemed organization.	Job Seeker	Create and maintain a good resume document that may entice the employers for the hiring process.	Mostly get rejected from many companies.	Most likely due to the resume document that I have designed even though I have to necessary qualifications to surpass their expectations and expect to hire me.	Should make a better and clean document by understanding what they use for the hiring process.

## 2.2 Proposed Solution

A tool that can assist job seekers in optimizing their resumes, specifically for ATS compatibility, is vitally required. This solution need to provide a thorough analysis as well as useful suggestions for enhancing resumes and making sure they meet the various requirements used by ATS. The improved capabilities of the Gemini AI Model may be used to create an ATS-Optimized Resume Analyzer in order to get around this problem. The application would assess resumes, point out areas that needed work in terms of formatting, keyword usage, and content organization, and offer detailed suggestions to help users increase the likelihood that their resumes would get past ATS filters. Job seekers would be able to more effectively showcase their qualifications in this way, improving their chances of being permitted for interviews.

Project Overview	
Objective	To create an optimized ATS Friendly Resume Analyzer using Google Gemini AI Model
Scope	Utilizing the Google's AI model helps us to save time by optimizing keyword matching, formatting and detection sections for enhancing the compatibility.
Problem Statement	
Description	Most of the resume documents get rejected automatically due to poor formatting, missing keywords and incompatibility.
Impact	Solving this problem will significantly improve the job application success rates by ensuring resumes are properly formatted, keyword-optimized, and ATS-compatible.
Proposed Solution	
Approach	Utilizing the Google's Generative AI model – Gemini by using it's API to create the application.
Key Features	Gemini AI is very advanced generative AI model and it is easy to integrate with any kind of application formats.

## Resource Requirements

Resource Type	Description	Specification/Allocation
Hardware		
Computing Resources	CPU/GPU specifications, number of cores	CPU – 6 core 12 threads AMD Ryzen or Intel Core CPU
Memory	RAM specifications	Minimum – 8 GB Recommended – 16 GB
Storage	Disk space for data, models, and logs	Required – 1 GB
Software		
Frameworks	Python frameworks	StreamLit UI
Libraries	Additional libraries	google.generativeai, os, PyPDF2, PIL
Development Environment	IDE, version control	Jupyter Notebook via Visual Studio Code
Data		
Data	Source, size, format	Few resume templates and examples



## 2.3 Initial Project Planning

- Gathering the necessary requirements that are needed for developing the project.
- Researching and experimenting with the current solutions available for developing the project.
- Evaluating the feasibility of integrating the modules that are required for the project.
- Exploring different resume documents in order to understand the standard format that is optimized for ATS resume analyzer.

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members	Sprint Start Date	Sprint End Date (Planned)
Sprint-1	Generative AI model creation	USN-1	As a user, I want to generate the accurate response for the user's requirements and compatibility with job description.	3	High	Varun	6th July 2024	7th July 2024
Sprint-1	Job description prompt	USN-2	As a user, I want to find the candidate that qualifies based on the job description that is provided by me.	2	High	Aryaman	8th July 2024	9th July 2024

Sprint-2	Website UI Design	USN-3	As a user, I want to browse the website with no errors.	2	Low	Varun	9th July 2024	10th July 2024
Sprint-2	Upload document format	USN-4	As a user, I want to make sure the document that I upload, should be compatible with the required format that is supported by the application.	1	Medium	Rueben	11th July 2024	12th July 2024
Sprint-2	Integration of website with Generative AI API	USN-5	As a user, I must ensure that the document that is uploaded must be analyzed by the Gemini AI model.	1	High	Jayanth	11th July 2024	12th July 2024

### 3. Model Development Phase

#### 3.1 Requirements Specification

A requirements text file is need to be created in order to install Python modules that are required to develop a functional code for this project.

i) Creation of requirements.txt file

The text file contains these following modules:

```
streamlit  
streamlit_extras  
google-generativeai  
python-dotenv  
PyPDF2  
Pillow
```

ii) Installation of the modules

The modules that will be used in this project are installed by performing this command in the Python terminal.

```
pip install -r requirements.txt
```

This process takes few minutes to be installed in an IDE.

#### 3.2 Initialization of Google API Key

The Gemini API key allows developers to safely access and incorporate Google's advanced AI models into their apps. Developers may authenticate and authorize their apps to utilize Gemini AI models by creating an API key in Google AI Studio and include it in API calls, resulting in improved security and scalable AI solutions.

The API key is generated by accessing the Google AI Studio. The key is generated by the user who is signed in with a personal account.

After API key is generated, the key is used for the initialization of the API by creating an '.env' file, where the API key is inserted into this file.

### 3.3 Interfacing with Pre-trained Model

To interface with the pre-trained model, an .py file is created named app, which will contain both the Gemini AI model and StreamLit UI code.

i) Load the Gemini Pro pre-trained model:

Importing required libraries and modules, such as dotenv, Streamlit, os, PyPDF2, GenerativeAI from Google, PIL (Python Imaging Library), and a custom module for providing vertical space to Streamlit.

The load\_dotenv() method is used to load environment variables from the .env file.

The Google API key is set up in the Generative AI module and is kept in the environment variable GOOGLE\_API\_KEY.

A Generative AI model object named "model" is built by using Google Gemini AI's Gemini Pro pre-trained model.

ii) Implement a function to get Gemini model's response and to read the contents of the PDF:

The function get\_gemini\_response takes an input as a text.

The model then generates a response using the input provided in a text format.

The response that is generated by the model, is then returned as a text.

The function input\_pdf\_text takes an input as the file uploaded by the user.

A PDF reader object is created using PyPDF2 module to convert the uploaded PDF file into a readable format. A 'text' variable is initialized as an empty string.

Using for loop, it reads every content from each page of the PDF file through each iterations and extracts the text from the file and stores it in the 'text'.

The stored content in 'text' is returned.

iii) Writing a prompt for Gemini model:

An ATS (Applicant Tracking System) is instructed in a multiline text called `input_prompt`. It defines the level of expertise needed to pass the ATS, including proficiency in a range of technical fields including data science and software engineering.

In a competitive job market, the applicant tracking system's (ATS) goal is to evaluate applicants based on job descriptions offered.

Three components are requested in the response: a summary of the profile, a list of keywords that are missing, and the percentage of the response that matches the job description.

### **3.4 Model Deployment using StreamLit:**

i) Integrate with Web Framework:

The web framework is designed using StreamLit and for this project, it is designed into four sections of the application.

a) Introduction:

The given code configures the page layout for a Streamlit application titled "Resume ATS Tracker".

It then generates a layout with two columns, with the first three times wider than the second.

The first column has the term "CareerCraft" as well as a header and markdown text presenting CareerCraft as an ATS-Optimized Resume Analyzer.

In the second column, a picture from a URL is shown, with the width adjusted to match the column.

Overall, the code generates a visually appealing interface for the CareerCraft programme, using text and graphics to explain its capabilities and benefits to users.

#### b) Offerings:

The given code uses Streamlit to generate a two-column layout, with the first column having an image and the second column containing a header followed by a list of offerings.

The title "Wide Range of Offerings" presents the list, which includes services such as ATS-Optimized Resume Analysis, Resume Optimisation, Skill Enhancement, and others.

Overall, this structure offers the application's functions in a simple and organised manner.

#### c) Resume ATS Tracking Application:

This code makes a two-column layout using Streamlit.

In the left column ('col1'), there is a heading urging visitors to begin their career journey, followed by a text box for copying the job description and a file uploader for uploading the resume.

When you click the "Submit" button, it obtains the text from the uploaded PDF resume, creates a response using the 'get\_gemini\_response' function based on the job description, and displays it as a subheader.

The right column ('col2') shows a picture that represents the career journey notion.

#### d) F.A.Q:

This code splits the page into two columns by calling Streamlit's 'st.columns()' method.

The right column ('col2') has a FAQ section containing questions and answers concerning CareerCraft. Each question and answer pair is given with 'st.write()'.

'avs.add\_vertical\_space()' adds vertical space between each question and answer pair.

An image is shown in the left column ('col1') by calling 'st.image()'.

## 4. Model Optimization and Tuning Phase

For Model Optimization and Tuning, the prompt is changed and fine-tuned to avoid hallucinations and variation in the overall matching percentage of the resume with the job description.

This allows the model to respond more accurately to the resume and give missing keywords relevant to the job description.

The model is optimized to extract relevant information such as job title, skills, experience, education, etc. It is also ensured that the summary is concise and informative.

The prompt is written and optimized to be as informative and pertinent to the job description

A screenshot of the input prompt used is given below

```
input_prompt="""
As an experienced ATS (Applicant Tracking System), proficient in the technical domain encompassing
Software Engineering, Data Science, Data Analysis, Big Data Engineering, Web Developer, Mobile App
Developer, DevOps Engineer, Machine Learning Engineer, Cybersecurity Analyst, Cloud Solutions Architect,
Database Administrator, Network Engineer, AI Engineer, Systems Analyst, Full Stack Developer, UI/UX
Designer, IT Project Manager, and additional specialized areas, your objective is to assess
resumes against provided job descriptions. In a competitive job market, your expertise is crucial
in offering top-notch guidance for resume enhancement. Assign precise matching percentages based on the JD
(Job Description) and identify any missing keywords with accuracy.

resume: {text}
description: {jd}

I want the response in the following structure:
The first line indicates the percentage match with the job description in multiples of 5 (JD).
The second line presents a list of missing keywords,
this should be included in context of the job description provided and
keywords not directly related to the job description should not be given as missing.
The third section provides a concise profile summary along with improvements required.

Mention the title for all the three sections.
While generating the response put some space between all the three sections
"""
```

## 5. Results

The application is deployed via the Python terminal and the StreamLit app is initialised by running this command

```
streamlit run app.py
```

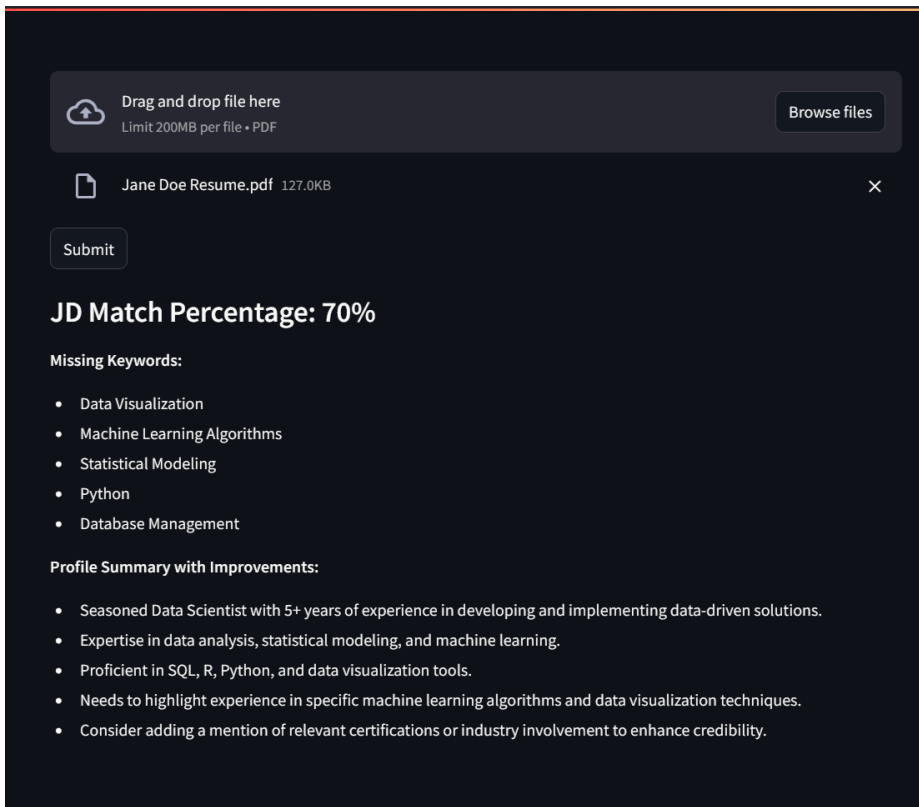
Once it is initialised, then StreamLit UI will initialise the webpage and the URL for this webpage.

Local URL: `http://localhost:xxxx`

The webpage is fully functional and it can perform the basic task.

First, type in the job title and job description that is required for matching the job requirements, then upload the resume which should be in a PDF format.

After uploading is done, click 'Submit' and it should take a while to generate the response for your resume description.



The screenshot displays the Streamlit application interface. At the top, there is a file upload section with a cloud icon and the text "Drag and drop file here" and "Limit 200MB per file • PDF". A "Browse files" button is located to the right. Below this, a file named "Jane Doe Resume.pdf" (127.0KB) is shown with a close button (X). A "Submit" button is positioned below the file list. The results section shows a "JD Match Percentage: 70%". Under "Missing Keywords:", there is a list of keywords: Data Visualization, Machine Learning Algorithms, Statistical Modeling, Python, and Database Management. Under "Profile Summary with Improvements:", there is a list of suggestions: Seasoned Data Scientist with 5+ years of experience in developing and implementing data-driven solutions, Expertise in data analysis, statistical modeling, and machine learning, Proficient in SQL, R, Python, and data visualization tools, Needs to highlight experience in specific machine learning algorithms and data visualization techniques, and Consider adding a mention of relevant certifications or industry involvement to enhance credibility.



## 6. Advantages and Disadvantages

### Advantages:

1. **Advanced NLP:** Gemini is a large language model (LLM) with powerful Natural Language Processing (NLP) features. This enables it to fully understand the intricacies of human language better than simpler keyword-matching algorithms, resulting in more accurate resume analysis for ATS compatibility.
2. **Flexibility:** Unlike rule-based systems, Gemini can accommodate a variety of writing styles and resume formats.
3. **Keyword Recognition:** Gemini can accurately detect relevant terms from both the resume and the job description. This allows you to personalize your resume to showcase skills and experiences that are relevant to the job criteria.
4. **Scalability:** Gemini's architecture makes it possible for it to effectively managing large volumes of resumes, which makes it suitable for companies that receive an immense number of applications.
5. **Integration Potential:** For a smooth workflow, the API can be integrated with currently in use applicant tracking systems (ATS).

### Disadvantages:

1. **Limited Explainability:** In contrast to rule-based systems, Gemini's decision-making process can be difficult to comprehend when it comes to presenting a score or recommendation. The lack of ability to see things clearly can make it difficult to troubleshoot errors or improve the analysis further.
2. **Data Dependency:** The level of quality and variety of the training dataset determines how well Gemini performs. The model's outputs may reflect biases found in the training dataset. It's critical to identify potential biases and take action to reduce them.
3. **API Access:** Depending on usage, access to Google's Generative AI API may be charged in order to use different Gemini models.
4. **Cost of Usage:** It can be computationally expensive to run Gemini model. Depending on your usage and pricing plan, this could result in higher costs.
5. **Fairness and Bias:** There is always a chance that AI models will be biased, even with mitigation techniques in place. To guarantee a fair and impartial review of each candidate's resume, monitoring and assessment must occur on a regular basis.

## 7. Conclusion

The Gemini AI model is notable for its adaptability and accuracy in managing complicated data jobs across sectors. By leveraging deep learning and natural language processing capabilities, companies can confidently extract important insights, optimise processes, and make data-driven choices. Its versatility makes it a valuable tool for improving operational efficiency and generating innovation in today's changing business environment.

To conclude, the future of ATS-optimized resume analysis using generative AI models such as Gemini appears to be very promising, even though there are still issues to be resolved. These models have the potential to facilitate a more sophisticated and nuanced evaluation of candidates, resulting in a more efficient and equitable recruitment process. By embracing AI responsibly and ethically, businesses can use its power to find top talent and make data-driven hiring decisions that benefit both the company and the candidates.

## 8. Future Scope

Gemini and other generative AI models have the potential to completely change the way resume analysis is done in Applicant Tracking Systems (ATS) going forward. Beyond just matching keywords, these models can analyze a resume's overall structure, writing style, and terminology usage to give a more comprehensive assessment of a candidate's qualifications. This could also be used to evaluate soft skills, such as communication, teamwork, and problem-solving skills, by examining linguistic patterns. AI can also enhance the candidate matching process by taking into account a candidate's complete background to find transferable skills that are valuable to the organization even if they aren't stated in the job description. By reducing prejudice in job descriptions and resumes, this can promote diversity and inclusion.

The employers' time can be saved by using AI to automate the shortlisting of resumes based on key requirements. It is even capable of proposing customized interview questions and subjects based on the job requirements and the candidate's characteristics. In an ideal future, AI-powered resume analysis would integrate with ATS, CRM, and other talent management systems to provide a seamless and effective hiring process. AI models would constantly learn from and evolve according to previous decisions, enhancing their ability to recognize the best candidates. But it's also important to address ethical issues with impartiality, fairness, and openness in decision-making. Moreover, as AI develops further, standards and laws may be required to guarantee its responsible application in the recruiting process.

Overall, generative AI models have the potential to transform resume analysis and the recruitment process; however, careful consideration of their limitations and responsible implementation are required.

## 9. Appendix

### 9.1 Source Code:

```
from dotenv import load_dotenv
import streamlit as st
from streamlit_extras import add_vertical_space as avs
import google.generativeai as genai
import os
import PyPDF2
from PIL import Image

load_dotenv()

genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))

model = genai.GenerativeModel('gemini-pro')

def get_gemini_response(input):
    response = model.generate_content(input)
    return response.text

def input_pdf_text(uploaded_file):
    reader = PyPDF2.PdfReader(uploaded_file)
    text = ''
    for page_num in range(len(reader.pages)):
        page = reader.pages[page_num]
        text += str(page.extract_text())
    return text

input_prompt="""
As an experienced ATS (Applicant Tracking System), proficient in the technical domain
encompassing
Software Engineering, Data Science, Data Analysis, Big Data Engineering, Web
Developer, Mobile App
Developer, DevOps Engineer, Machine Learning Engineer, Cybersecurity Analyst, Cloud
Solutions Architect,
Database Administrator, Network Engineer, AI Engineer, Systems Analyst, Full Stack
Developer, UI/UX
Designer, IT Project Manager, and additional specialized areas, your objective is to
assess
resumes against provided job descriptions. In a competitive job market, your expertise
is crucial
in offering top-notch guidance for resume enhancement. Assign precise matching
percentages based on the JD
(Job Description) and identify any missing keywords with accuracy.
```

```
resume: {text}
description: {jd}
```

I want the response in the following structure:

The first line indicates the percentage match with the job description in multiples of 5 (JD).

The second line presents a list of missing keywords, this should be included in context of the job description provided and keywords not directly related to the job description should not be given as missing. The third section provides a concise profile summary along with improvements required.

Mention the title for all the three sections.

While generating the response put some space between all the three sections  
"""

```
##streamlit UI
st.set_page_config(page_title="Resume ATS Tracker", layout="wide")

avs.add_vertical_space(4)

col1, col2 = st.columns([3,2])
with col1:
    st.title("CareerCraft")
    st.header("Navigate the Job Market with Confidence!")
    st.markdown("""<p style='text-align: justify;'>
        Introducing CareerCraft, an ATS-Optimized Resume Analyzer - your
ultimate
        solution for optimizing job applications and accelerating career
growth. Our innovative platform
        leverages advanced ATS technology to provide job seekers with valuable
insights into their resumes'
        compatibility with job descriptions. From resume optimization and
skill enhancement to
        career progression guidance, CareerCraft empowers users to stand out
in today's competitive job market.
        Streamline your job application process, enhance your skills, and
navigate your career path with confidence.
        Join CareerCraft today and unlock new opportunities for professional
success!
        </p>""", unsafe_allow_html=True)

with col2:
    st.image("https://cdn.dribbble.com/userupload/12500996/file/original-
b458fe398a6d7f4e9999ce66ec856ff9.gif", use_column_width=True)

avs.add_vertical_space(10)
```

```

#offerings
col3, col4 = st.columns([3,2])
with col4:
    st.header("Wide Range of offerings")
    st.write('ATS-Optimized Resume Analysis')
    st.write('Resume Optimization')
    st.write('Skill Enhancement')
    st.write('Career Progression Guidance')
    st.write('Tailored Profile Summaries')
    st.write('Streamlined Application Process')
    st.write('Personalized Recommendations')
    st.write('Efficient Career Navigation')

with col3:
    img1 = Image.open("images/icon1.png")
    st.image(img1, use_column_width=True)

avs.add_vertical_space(10)

col5, col6 = st.columns([3,2])
with col5:
    st.markdown("<h1 style='text-align: center;*>Embark on Your Career
Adventure</h1>", unsafe_allow_html=True)
    jd=st.text_area("Paste the Job Description")
    uploaded_file=st.file_uploader("Upload Your Resume", type="pdf", help="Please
uplaod the pdf")

    submit = st.button("Submit")

    if submit:
        if uploaded_file is not None:
            text=input_pdf_text(uploaded_file)
            response=get_gemini_response(input_prompt)
            st.subheader(response)

with col6:
    img2 = Image.open("images/icon1.png")
    st.image(img2, use_column_width=True)

avs.add_vertical_space(10)

col7, col8 = st.columns([2, 3])
with col8:
    st.markdown("<h1 style='text-align: center;*>FAQ</h1>", unsafe_allow_html=True)
    st.write("Question: How does CareerCraft analyze resumes and job descriptions?")
    st.write("""Answer: CareerCraft uses advanced algorithms to analyze resumes and
job descriptions,
            identifying key keywords and assessing compatibility between the

```

```

two.""")
    avs.add_vertical_space(3)
    st.write("Question: Can CareerCraft suggest improvements for my resume?")
    st.write("""Answer: Yes, CareerCraft provides personalized recommendations to
optimize your resume
                for specific job openings, including suggestions for missing keywords
and alignment with
                desired job roles.""")
    avs.add_vertical_space(3)
    st.write("Question: Is CareerCraft suitable for both entry-level and experienced
professionals?")
    st.write("""Answer: Absolutely! CareerCraft caters to job seekers at all career
stages, offering
                tailored insights and guidance to enhance their resumes and advance
their careers.""")

with col7:
    img3 = Image.open("images/icon1.png")
    st.image(img3, use_column_width=True)

```

## 9.2 GitHub and Project Demo Link:

1. [GitHub Link](https://github.com/realben1601/CareerCraft-ATS-Optimized-Resume-Analyzer-using-Gemini-Model) - <https://github.com/realben1601/CareerCraft-ATS-Optimized-Resume-Analyzer-using-Gemini-Model>
2. [Project Demo](https://drive.google.com/file/d/1Lw_4vJYxqqTZBYzF4C9SnXfD6I94v901/view?usp=sharing) - [https://drive.google.com/file/d/1Lw\\_4vJYxqqTZBYzF4C9SnXfD6I94v901/view?usp=sharing](https://drive.google.com/file/d/1Lw_4vJYxqqTZBYzF4C9SnXfD6I94v901/view?usp=sharing)