

CODE FOR JOB SCOUT

```
import streamlit as st
import google.generativeai as genai
import os
import PyPDF2 as pdf
import json

st.set_page_config(layout="wide")

api_key = os.getenv("API_KEY")
genai.configure(api_key="AlzaSyBqIcx3wsVw2d1_ZegkknmhPW92gMZFJmY")

def get_gemini_response(input):
    model=genai.GenerativeModel('gemini-pro')

    response=model.generate_content(input,generation_config=genai.types.GenerationConfig(temperature
    =0))
    return response.text

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

## streamlit app

left, right = st.columns(2)
with left:
    st.title("JOB SCOUT")
    st.write("\n")
    st.subheader("Free Resume Score Generator: Scan Your Resume To Improve Your Resume ATS")
    st.write("Our ATS resume grader will optimize your resume for any job, score it, and provide
    suggestions for improvements to make it as effective as possible.....")

with right:
    img = "https://static.jobscan.co/blog/uploads/ats_3.png"
    width = 560
    st.image(img, width=width)
```

```
st.subheader("Optimize Your Resume In Minutes")
jd=st.text_area("Paste the Job Description")
uploaded_file=st.file_uploader("Upload Your Resume",type="pdf",help="Please uplaod the pdf")
```

```
#submit button
```

```
st.write("\n")
```

```
st.write("\n")
```

```
st.write("""
```

```
<style>
```

```
.stButton>button {
```

```
display: block;
```

```
margin: 0 auto;
```

```
height:60px;
```

```
width:300px;
```

```
background-color: red; /* Set background color */
```

```
color: white; /* Set text color */
```

```
font-size: 40px; /* Set font size */
```

```
padding: 10px 20px; /* Set padding */
```

```
border-radius: 28px; /* Set border radius */
```

```
border: black; /* Remove border */
```

```
cursor: pointer; /* Change cursor on hover */
```

```
}
```

```
.stButton>button:hover {
```

```
background-color:dark-orange; /* Change background color on hover */
```

```
}
```

```
</style>
```

```
""", unsafe_allow_html=True)
```

```
submit = st.button("Get Your Score")
```

```
st.write("\n")
```

```
st.write("\n")
```

```
st.write("\n")
```

```
st.write("\n")
```

```
st.write("\n")
```

```
st.write("\n")
```

```
left_column, right_column = st.columns(2)
```

```
# Content for first section
```

```
with left_column:
```

```
image="https://www.platinumresumes.com/wp-content/uploads/2017/07/bigstock-Skill-Level-85782647.jpg"
```

```
width = 500
```

```
st.write('\n')
```

```
st.write('\n')
```

```
st.image(image,width=width)
```



```

image =
"https://tse4.mm.bing.net/th?id=OIP.coDqbPUlxRoWaS_XDwi6EwHaDt&pid=Api&P=0&h=180"
width = 469
st.image(image, width=width)

with right_column:
    st.write('\n')
    st.write('\n')
    st.subheader("Get Immediate Suggestions")
    st.write("Based on your industry, our resume scanner will suggest relevant improvement to each
section of your resume.")
    st.write('\n')
    st.write("For example, you'll be given relevant skills to add and our AI-powered software will help you
add or improve your resume introduction.")

st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
st.subheader("-----RESULTS-----")

```

```

if submit:
    if uploaded_file is not None:
        text=input_pdf_text(uploaded_file)
        input_prompt=f"""
        Hey Act Like a skilled or very experience ATS(Application Tracking System), with extensive
        experience in evaluating resumes against job descriptions. Be a strict checker. Let's dive into the
        evaluation process based on the provided job description:

```

1. Get Resume Text: Extract the full resume text enclosed in triple backticks (``).
2. Evaluate Resume: Thoroughly analyze the entire resume text, considering the competitive job market.
3. Assign Percentage Based on JD Matching: Determine the percentage of match between the job description (JD) enclosed in triple quotes (""") and the resume text. This percentage indicates the likelihood of the candidate's resume being shortlisted for the job based on the matched text from the JD. If there is an exact match assign percentage greater than 85. If there is a good match assign % greater than 75. If the match is moderate assign % greater than 60 and if the match is not that good assign percentage around 30.
4. Identify Missing Keywords: List the missing keywords representing skills, technologies, or tools knowledge from the job description. These keywords should ideally be present in the resume. Separate the missing keywords with commas in a different section for easy identification and consideration.
5. Check Grammar and Spelling: Review the resume for grammar and spelling errors since human errors are common.
6. Highlight Grammar and Spelling Mistakes: Identify lines or words with incorrect grammar or spelling for correction.

8. Assess Work Experience: Examine the work experience section and provide suggestions for improvement in the format of old text -> suggested new text. Only suggest the lines where major improvement is required not all.
9. Find Opportunities for Quantifiable Figures: Identify areas in the resume where quantifiable figures (e.g., percentages, numbers) can be added, excluding lines already containing figures or percentages. Limit suggestions to lines where you are 100% confident in the format of old text -> suggested new text.
10. Maintain Consistency: Ensure a consistent output format for each section, regardless of the provided job description and resume.

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

Below is the desired output structure for each section:

```
JD Match in %
Missing Keywords in an endosed list comma-separated
Grammar and Spelling Errors
Experience Updates in the format of old text -> suggested new text
Lines where I can add quantifiable figures mostly in '%' or in numbers in the format of old text ->
suggested new text""
```

```
    response=get_gemini_repsonse(input_prompt)
    st.subheader(response)
    st.write("_____")
```

```
else:
    st.warning("Please upload a PDF file.")
```

```
st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
```

```
# Custom CSS for footer
footer_style = ""
<style>
.footer {
    width: 100%;
    background-color:skyblue;
    padding: 10px 10px;
    font-size: 15px;
    color: #000000;
    text-align:center;
}
.footer1 {
    width: 100%;
    background-color:skyblue;
    padding: 10px 10px;
```

```
font-size: 15px;
color: #000000;
text-align:center;
}
</style>
st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
st.write("\n")
st.markdown(footer_style, unsafe_allow_html=True)
st.markdown('<div class="footer">© 2024, JOBSCOUT, All Rights Reserved </div>',
unsafe_allow_html=True)
st.markdown('<div class="footer1">Terms of Use | Privacy Policy | Cookie Policy | Contact Us | About
Us | Careers</div>', unsafe_allow_html=True)
```