

**FATEC INTELIGÊNCIA ARTIFICIAL**

**GRUPO SOULCARE / FEELING AI**

**RELATÓRIO DE PESQUISA**

Integração do Flask com Web Scraping e Análise de Sentimentos no Projeto PROJAS

**Versão Final – Código Real do Projeto PROJAS**

**FATEC INTELIGÊNCIA ARTIFICIAL – 2025**

### 3. FUNCIONAMENTO TÉCNICO DO FLASK NO PROJETO PROJAS

O Flask atua como o núcleo de integração do PROJAS, coordenando a coleta das opiniões enviadas pelo site, o processamento automatizado por Web Scraping e a Análise de Sentimentos. Diferente de aplicações convencionais, o projeto não utiliza banco de dados; os resultados são processados e exibidos em tempo real, permitindo relatórios imediatos e simplificados.

#### 3.1 Coleta de Opiniões via Web Scraping

O módulo de scraping do PROJAS é acionado diretamente pelo Flask. Ele acessa o site com as opiniões enviadas e utiliza as bibliotecas **BeautifulSoup** e **requests** para capturar o texto e os autores das postagens.

```
from bs4 import BeautifulSoup
import requests
from csv import writer

url = 'https://projas.onrender.com/complaints'
result = requests.get(url)
soup = BeautifulSoup(result.text, 'html.parser')

with open('quotes.csv', 'w', newline='', encoding='utf-8') as f:
    thewriter = writer(f)
    # Pegando os dados
    for p in soup.find_all('p'):
        texto = p.get_text()
        for h5 in soup.find_all('h5'):
            autor = h5.get_text()
            formatação = [autor, texto]
            thewriter.writerow(formatação)
```

Neste código, o Flask pode acionar esse script internamente por meio de uma rota administrativa, integrando a coleta dos dados com o fluxo geral do sistema. A saída em CSV é usada para alimentar a análise de sentimentos de forma automatizada.

#### 3.2 Análise de Sentimentos Baseada em Estrelas

O PROJAS utiliza duas funções principais para classificar sentimentos: uma baseada na quantidade de estrelas atribuídas pelo cliente e outra pelo modelo **CardiffNLP** (análise semântica avançada).

```
# --- Função 1: Classificação por Estrelas ---
def analisar_sentimento(estrelas, score):
```

```

if estrelas >= 4:
    sentimento = 'Positivo'
elif estrelas <= 2:
    sentimento = 'Negativo'
else:
    sentimento = 'Neutro'
return sentimento, estrelas, score

# --- Função 2: Modelo CardiffNLP (Emoji) ---
def formatar_cardiffnlp(resultado):
    label = resultado['label'].lower()
    score = resultado['score']
    return label, score

```

Essas funções são chamadas diretamente pelas rotas Flask durante a execução. Assim, quando uma nova opinião é enviada ou coletada, o sistema avalia automaticamente o sentimento e produz a resposta em tempo real.

### 3.3 Integração com o Flask

O Flask é o responsável por orquestrar todo o fluxo entre o site, o scraping e a análise. A cada requisição, o Flask chama as funções correspondentes e exibe o resultado processado. Ele também controla as rotas que disparam cada uma das etapas, como mostrado abaixo:

```

@app.route('/enviar-opiniao', methods=['POST'])
def enviar_opiniao():
    estrelas = int(request.form['estrelas'])
    score = float(request.form['score'])
    sentimento, estrelas, score = analisar_sentimento(estrelas, score)
    return render_template('resultado.html', sentimento=sentimento,
estrelas=estrelas, score=score)

@app.route('/iniciar-scraping')
def iniciar_scraping():
    executar_scraping()
    return "Raspagem e análise concluídas com sucesso!"

```

### 3.4 Geração do Relatório Final

Ao final do processamento, o Flask utiliza **Jinja2** para renderizar o relatório HTML contendo a porcentagem de opiniões positivas, negativas e neutras. O relatório é exibido de forma clara e direta,

sem necessidade de armazenamento em banco de dados, refletindo os resultados mais recentes obtidos pela raspagem e pela análise automatizada.

### **Conclusão Técnica**

O Flask funciona como um **núcleo orquestrador** no PROJAS, coordenando a entrada de dados (comentários), a coleta externa (scraping), o processamento de sentimentos e a exibição dos resultados. Seu papel é garantir que todas as camadas se comuniquem de forma eficiente, mantendo o projeto leve, ágil e automatizado.

### **Referências Técnicas:**

GRINBERG, M. Flask Web Development. 3. ed. O'Reilly Media, 2023.

SOUZA, R. A. Flask e Python: Desenvolvimento de Aplicações Web Inteligentes. São Paulo: Novatec, 2024.

HASHTAG PROGRAMAÇÃO. Introdução ao BeautifulSoup – Raspagem de Dados com Python. YouTube, 2025.

CARDIFF UNIVERSITY. CardiffNLP Sentiment Analysis Model Documentation. 2024.