



MANIPAL UNIVERSITY
JAIPUR

Smart Resume System

MTE Minor Project Presentation

Submitted By: Gurbani Kaur (229302139)
Anshika Sethi (229302127)

PROBLEM STATEMENT

Manual resume filtering is inefficient and prone to error.

- No tailored systems for academic resume parsing
- Difficult to handle large volumes quickly
- Lacks fairness, consistency, and automation
- Recruiters often rely on subjective judgment

- **Example Scenario:**

Imagine a university receiving 500+ faculty applications. Manually screening each resume takes weeks, leading to delays in hiring. What if we could automate and rank candidates in minutes?



OUR SOLUTION:

- AI-powered resume filtering system that extracts & ranks faculty resumes
- Data-driven decision-making for selecting the best candidates
- Extract candidate information from resumes and match it with job descriptions using NLP
- Generate score-based, ranked outputs.

With our system, universities can now streamline faculty recruitment effortlessly!



TRADITIONAL HIRING

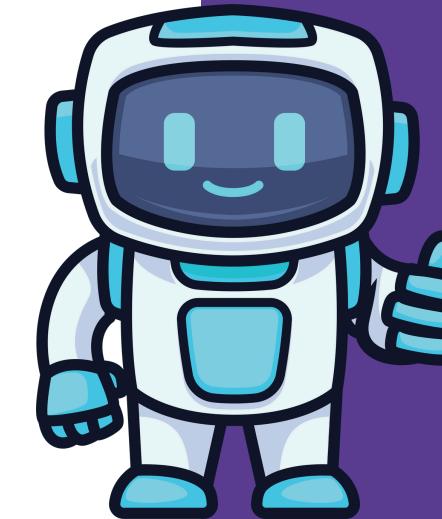
- Time-consuming manual resume screening
- High chances of human bias
- Overlooking qualified candidates
- Inconsistent evaluation criteria



SMART RESUME SYSTEM

Reduces manual effort by 70%, enabling quick faculty hiring

- Automates resume screening
- Ensures fair and objective selection
- Reduces hiring time significantly
- Matches candidates to university needs efficiently



PHASES/ TIMELINE

Research &
Planning

Implementing
Ranking Algorithm

Phase 1

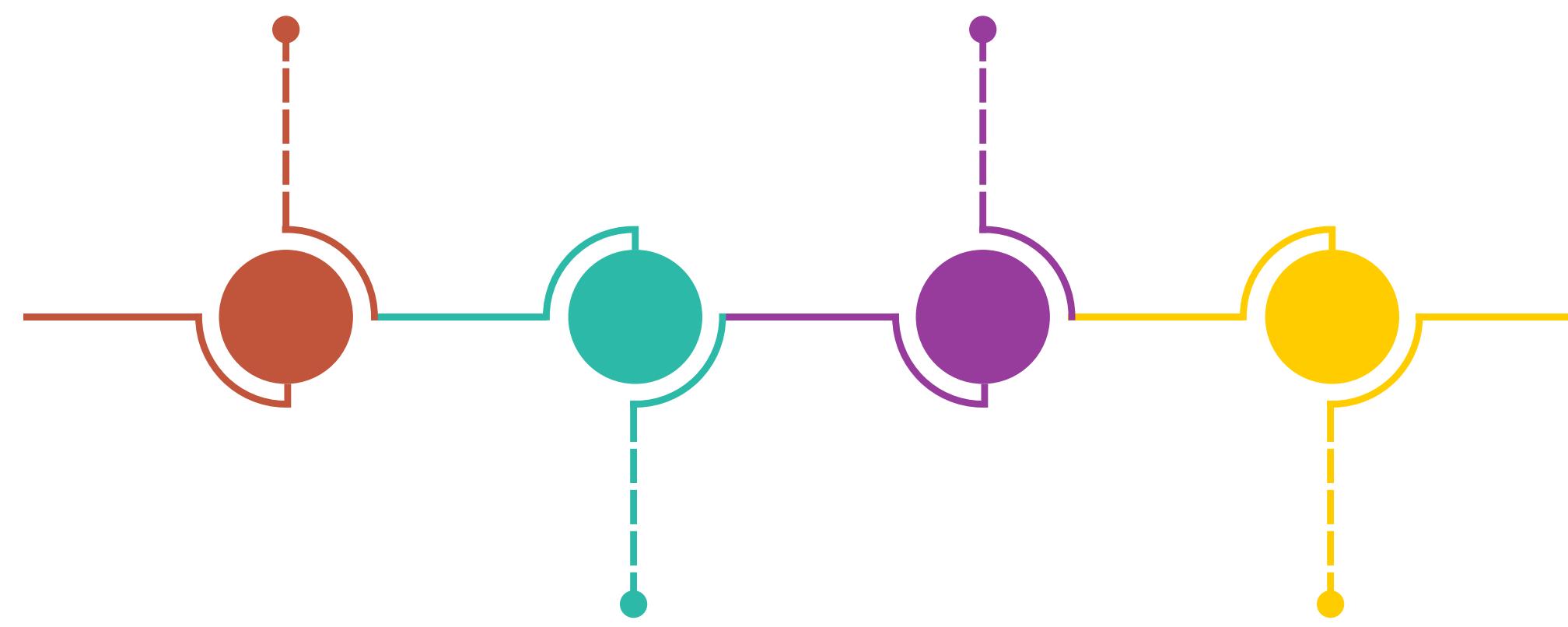
Phase 3

Phase 2

Phase 4

Data Extraction &
Processing

Finalizing Project



OBJECTIVES



Objective

Description

Web Interface

Simple platform for recruiters to upload and view results

Extract Data

Capture qualifications, research, experience, and skills

Rank Candidates

AI/ML-based scoring aligned with hiring needs

Data Storage

Secure SQL database management

ENGINEERING METHODOLOGY

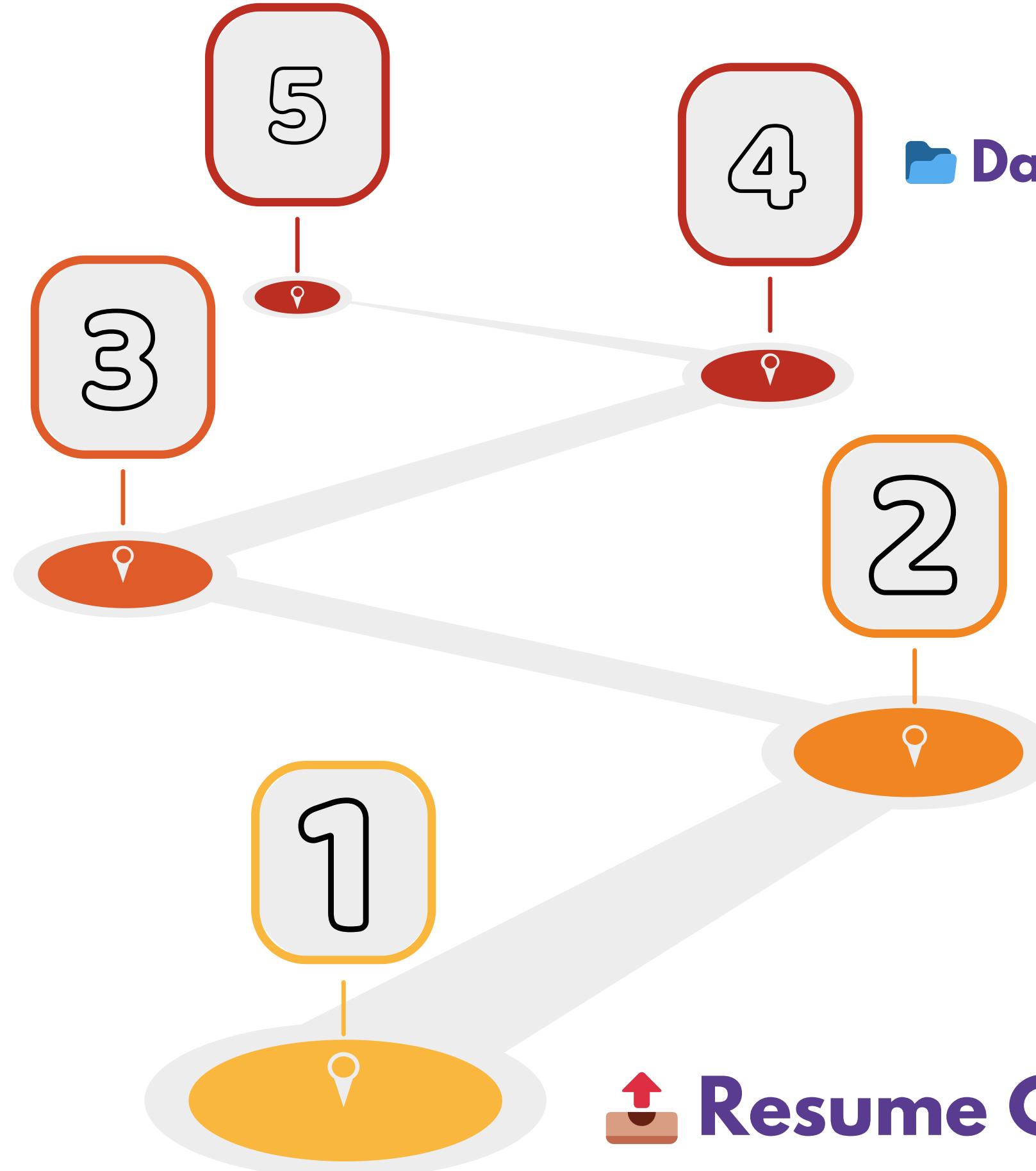
We followed the Agile development lifecycle to break the project into phases and iterate quickly.

Key Practices:

- Sprint-based feature development
- Weekly reviews and testing
- GitHub for code management
- Trello for task tracking



Ranking & Display Results



RUNNING PROGRAM IN TERMINAL

```
Last login: Thu Mar 27 00:43:47 on ttys012
gurbanikaur@Gurbanis-MacBook-Air ~ % cd /Users/gurbanikaur/Documents/smart_resume
gurbanikaur@Gurbanis-MacBook-Air smart_resume % source my_env/bin/activate
(my_env) gurbanikaur@Gurbanis-MacBook-Air smart_resume % python3 app.py
 * Serving Flask app 'app'
 * Debug mode: on
2025-03-27 01:33:43,001 - INFO - WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
2025-03-27 01:33:43,001 - INFO - Press CTRL+C to quit
2025-03-27 01:33:43,002 - INFO - * Restarting with stat
2025-03-27 01:33:44,207 - WARNING - * Debugger is active!
2025-03-27 01:33:44,217 - INFO - * Debugger PIN: 935-547-735
```

FRONTEND DEVELOPMENT

1

Upload Section

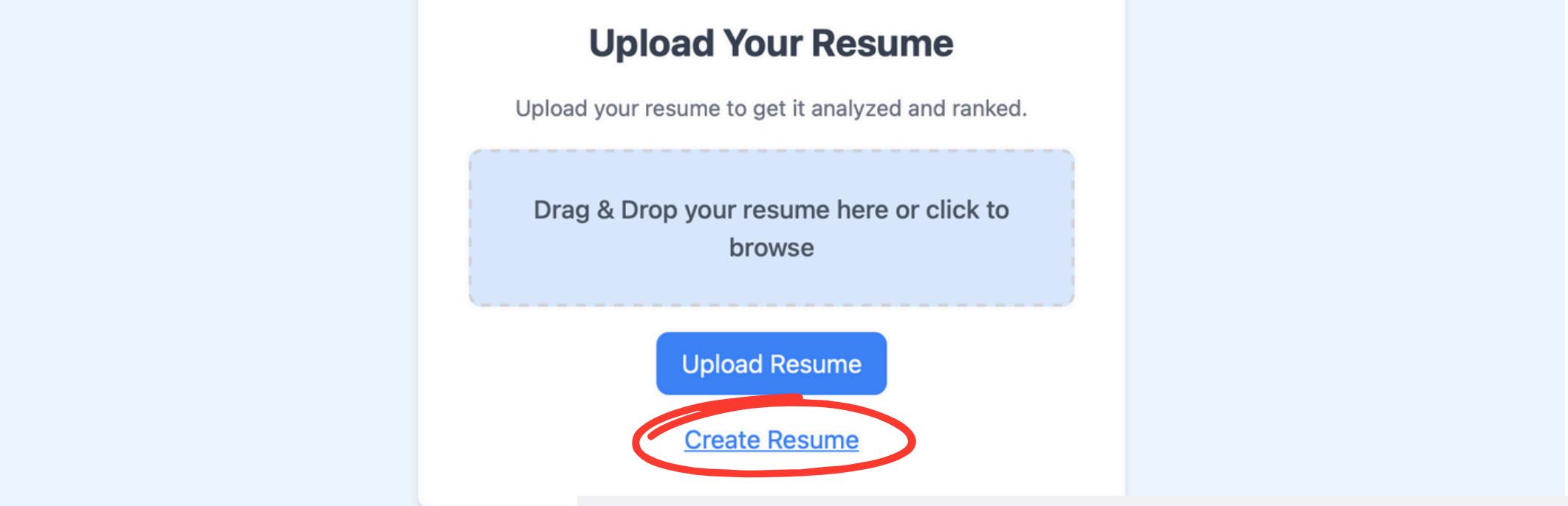
Upload Your Resume

Upload your resume to get it analyzed and ranked.

Drag & Drop your resume here or click to browse

[Upload Resume](#)

[Create Resume](#)



- **HTML**
- **CSS**
- **Tailwind CSS**

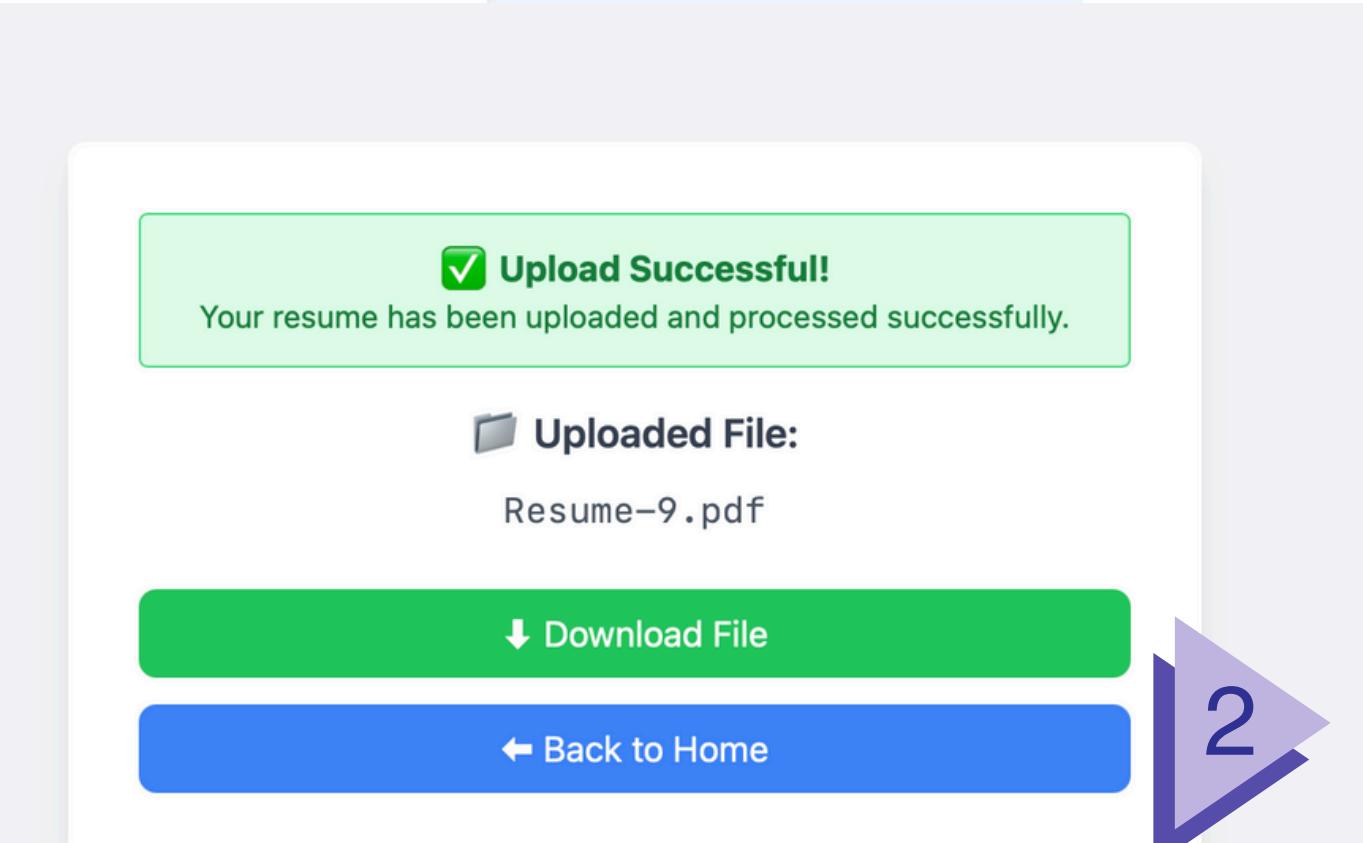
2

Upload Successful!
Your resume has been uploaded and processed successfully.

Uploaded File:
Resume-9.pdf

[Download File](#)

[Back to Home](#)



Upload Status
Download Uploaded File
Back to Home

RESUME BUILDER

User will enter the information to create a resume

Resume Builder

The screenshot shows a user interface for a resume builder. On the left, there is a large input area with fields for Full Name, Email, Phone, LinkedIn, Education, Experience, Skills, and Projects. Each field has a placeholder text below it. On the right, there is a preview panel titled "Your Name" which displays the entered information: Email (your.email@example.com), Phone (123-456-7890), and LinkedIn (@username). Below this, sections for Education, Experience, Skills, and Projects are shown with their respective placeholder texts. At the bottom, there is a green button labeled "Generate Resume" and a red button labeled "Download as PDF". A red circle highlights the "Download as PDF" button. Below these buttons is a file input field with the placeholder "Choose File" and the message "no file selected". At the very bottom is a red "Back to Home" button.

Full Name:

Email:

Phone:

LinkedIn:

Education:

Experience:

Skills:

Projects:

Your Name
Email: your.email@example.com
Phone: 123-456-7890
LinkedIn: @username

Education
Your education details...

Experience
Your experience details...

Skills
Your skills...

Projects
Your projects...

Generate Resume

Download as PDF

Choose File no file selected

Back to Home

User can preview details here!

Can download Resume from here!

File Name	Full Name	Email	Phone	LinkedIn	Education	Experience	Skills	Project
Gurbani_Resume.pdf	Gurbani Kaur Saluja		+91 9369062843		Bachelor's in Information Technology Sep 2022 - Present	Design, Data-Driven Interfaces, Machine		Smart Resume System
resume.pdf	GURBANI KAUR	gurbanikaur1702@gmail.com	+91) 9369062843		Bachelor's in Information Technology Sep 2022 - Present	Exchange Administrator, IAESTE INDIA, Karnataka, India August 2024 - Present		Smart Resume System
Resume-9.pdf	Gurbani Kaur	gurbanikaur1702@gmail.com	9369062843	https://www.linkedin.com/in/gurbani-kaur-5a6788264/	B.TECH Information Technology	Exchange Administrator IAESTE India	HTML, CSS, Javascript, Python	Smart Resume System
mariyuhs_resume.pdf	MARIYUSH GUPTA	mariyushgupta@gmail.com	+91) 6387504771		Bachelor's in Information Technology Sep 2022 - Present	President, IAESTE India LC MUJ Aug 2023 - Aug 2024	Technical & Digital Skills: Social Media Management, Pay-Per-Click (PPC) Advertising, Graphic	
Resume-10.pdf	Ahana jain	Ahana@gmail.com	8299860210	linkedin.com/in/your-profile	Computer science	President IAESTE	Leadership	Your project details...

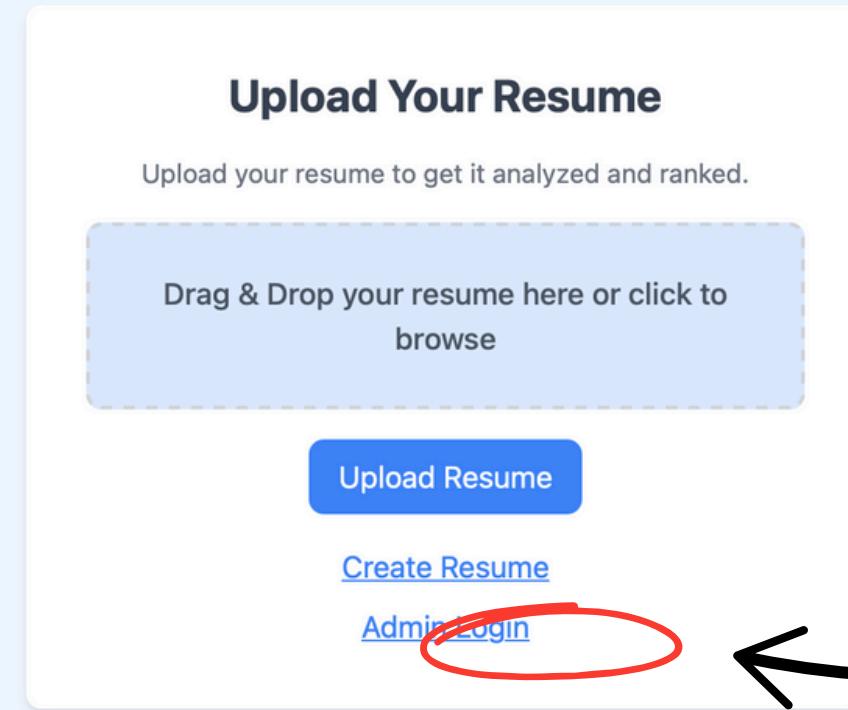
```

    index.html
    result.html
    upload_success.html
    upload.html
    uploads
        Gurbani_Resume.pdf
        mariyuhs_resume.pdf
        Resume-9.pdf
        Resume-10.pdf
        resume.pdf
    app.py
    resume_data.xlsx
    resume_extractor.py

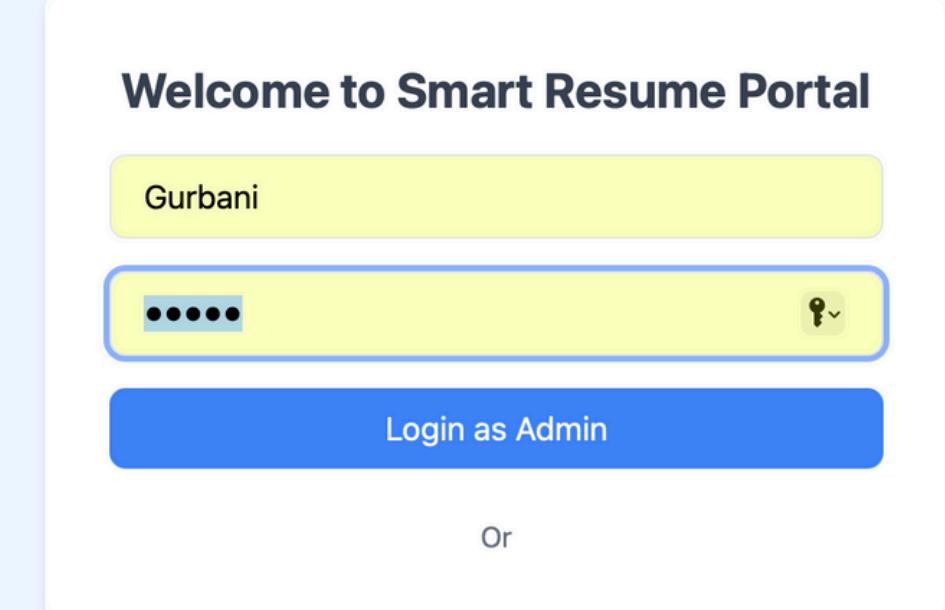
```

**Upload Resumes
are visible under
the uploads folder.**

**Extracted data
from resume !**



Added a admin login button in the homepage



Admin can login with its own id and password

Admin Dashboard

Enter Job Description:

Score Resumes

No results to show yet. Upload resumes and enter a job description to see scores.

Admin can enter the job description and get the score for uploaded resumes

Admin Dashboard

Enter Job Description:

Spring Boot are part of your everyday language, you'll fit right in.

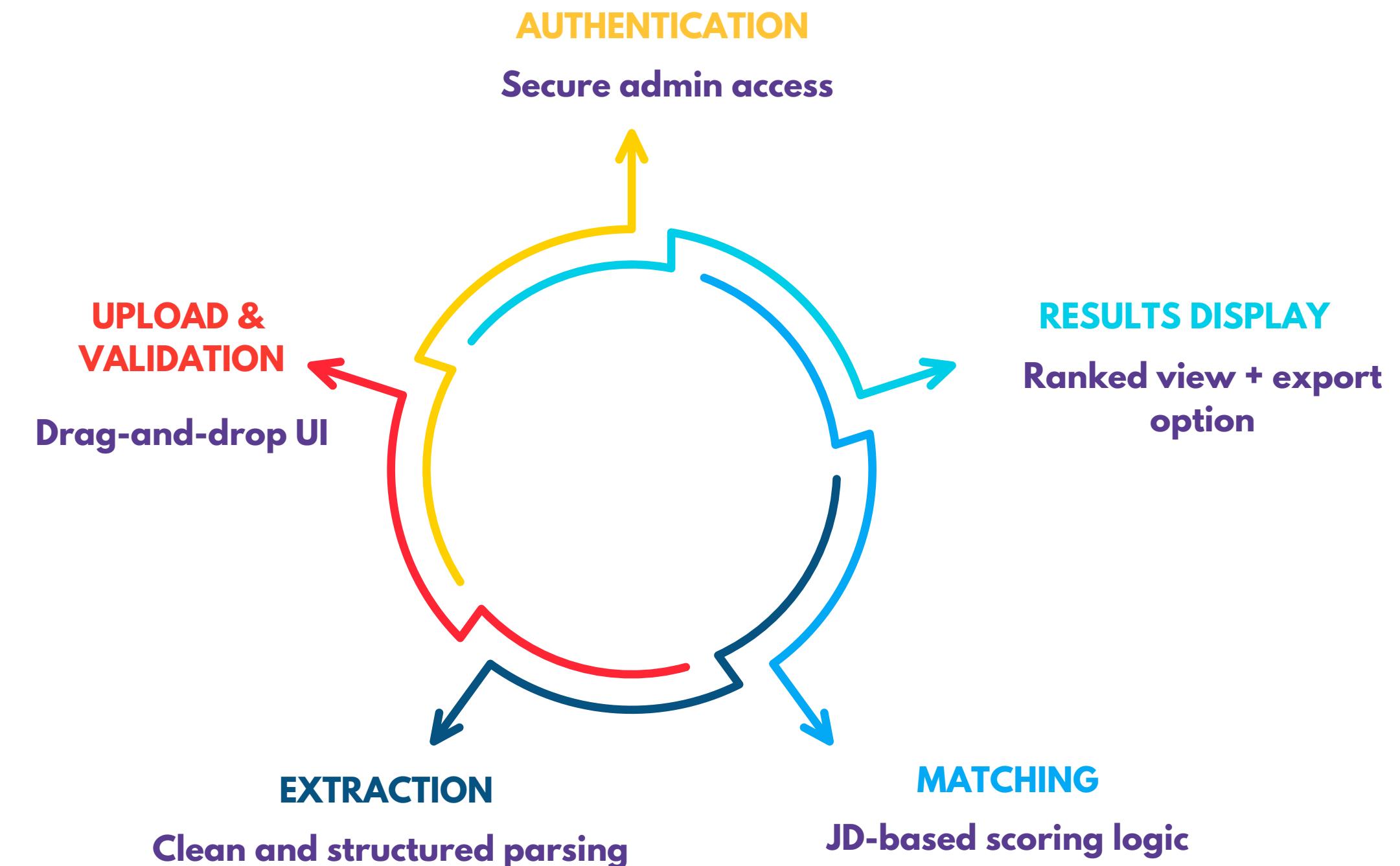
This isn't a hand-holding kind of role. We want someone who can take ownership, challenge assumptions, and get things done. You'll work with a team that values clarity, performance, and simplicity — not bloated code or over-engineered fixes. If you've got experience with CI/CD, cloud stuff (AWS/GCP), and know how to debug like a detective, let's talk.

Score Resumes

Resume Scores:

Rank	Name	Score	File
1	MARIYUSH GUPTA	1.8	View Resume
2	CONTACT	1.6	View Resume
3	Gurbani Kaur Saluja	1.4	View Resume
4	First Last CONTACT	1.2	View Resume
5	GURBANI KAUR SALUJA	1.0	View Resume
6	Nadia Delgado	0.7	View Resume

MODULE BREAKDOWN



OBSERVED

BENEFITS



Time-Efficient: Processes each resume in under 3 seconds

Fair & Consistent: Objective scoring minimizes human bias

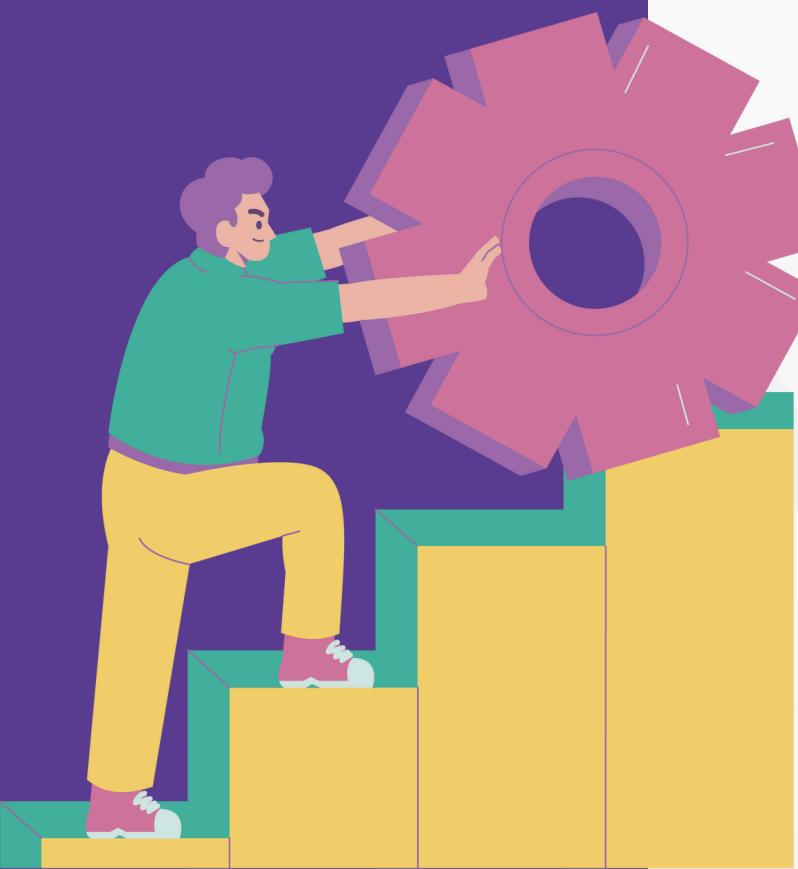
Modular Design: Ready for future upgrades like deep learning

Scalable: Handles bulk uploads without performance drop

Structured Data Export: Seamlessly generate Excel reports

CHALLENGES FACED

- **NLP-Based Extraction Issues** – Initially, we tried using Natural Language Processing (NLP) for resume data extraction, but it didn't yield accurate results.
- **Regex Approach** – Shifted to Regular Expressions (Regex), which improved extraction but wasn't effective for all resume formats.
- **Inconsistent Resume Structures** – Different resume formats made it difficult to extract data uniformly.
- **Fixed Format Implementation** – To ensure accurate extraction, we decided to enforce a fixed resume format, which provided better results.



KEY LEARNINGS



- **Adaptability in Approach** – Learned that different data extraction methods work better in different scenarios.
- **Structured Data Handling** – Implementing a fixed format helped streamline the extraction process and improve accuracy.
- **Balancing Flexibility & Accuracy** – While the fixed format works perfectly, ongoing improvements are helping extraction on general formats as well.
- **Continuous Improvement** – The latest trials show that extraction is working partially on general resumes but fully on the fixed format, leading to further refinements.

Future Scope

Smarter Matching: Integrate BERT or RoBERTa for contextual, AI-driven resume ranking.

Cloud Deployment: Host on AWS or Azure for remote access and scalability.

Analytics Dashboards: Visualize trends in skills, qualifications, and JD matching via Plotly/Dash.

Role-Based Access: Enable user-specific views for Dean, Reviewer, and Admin roles.

Candidate Feedback: Auto-generate improvement tips for lower-ranked resumes





THANK YOU
