# Doc(s)Simplr

All docs need to classify !

Mohit R.Sojitra
K. D. Patel Deaprtment of Information Technology
Charusat , Changa .
Gujrat , India
17it109@charusat.edu.in

***Abstract*—When you throw a pile of Docs and resumes at us, we classify that docs into a proper manner to minimize your work while searching the perfect person of your demand. while doing so, you will be blessed with a properly arranged pile of docs available for download**

## I. Introduction

This project is complete based on simplifying the PDF docs. Based on your need. Ease of Use We here at Doc(S)implr classifies a zip full of Resumes into the categories making things super easy for recruiters of the company. We at Doc(s)implr also email the results obtained by our algorithms concluding what's your profile is upto or what are you good at! We classify that docs into a proper manner to minimize your work while searching. while doing so, you will be blessed with a properly arranged pile of docs available for download.

We are using natural language processing for the classification of the data and uses machine learning techniques to classifying the best resulted as you need by our system to import your company and quality of your employees.

Our system also introduced the login functionalities for those who wanted to save their resulted in our DynamoDB and MySQL database and gives results in front of you via web application.

## II. Purpose

Main purpose of developing this project is to classifying the PDFs of resumes for the recruiter to easily classify that which of applicants are best for the job in company or call for interview.

That means our web application is remove to burden of recruiters for reading and analyzing every resume they got.

Recruiters just need to worry about where and when to take the interviews of the eligible candidates and not of the sorting phase.

## III. Technologies

We have created web application using Flask for frontend for backend we are using python libraries. Here there are some of the important libraries we used.

### A. *Natural language processing*

Natural language processing are a line of linguistics, computer science, information engineering, and artificial intelligence that intersect with communication between computers and human languages, and especially how computers organize and process large amounts of natural language data.

### B. *Flask*

Flask is a lightweight WSGI application framework. It is designed to make implementation quick and easy, with the ability to scale access to complex applications. It started out as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Flask offers suggestions, but does not use any dependencies or project formats. It is up to the developer to select the tools and libraries they want to use. There are many extensions provided by the community that make adding new functionality easier.

### C. *PyPDF2*

The Pure-Python library is built as a PDF tool. It can:

- extracting document information (title, author,…)
- to separate text page by page
- compiling a page of text by page
- plant pages
- combining multiple pages into one page
- encryption and encryption of PDF files
- and more!

Called Pure-Python, it should work on any Python platform without relying on external libraries. It can also work more perfectly on StringIO objects than on file streaming, which allows PDF manipulation in memory. It is therefore a useful tool for websites that manage or manage PDFs.

### *D. SpaCy(en_core_web_sm)*

SpaCy v2.0 has new neural models for tagging, syncing and business recognition. The models are designed and used from scratch at the start of spaCy, to give you an unparalleled balance of speed, size and accuracy. A novel blurring strategy with elements of subordinate structure is used to support large words in small tables. Variable layers with residual connection, solution implementation and maxout connection are used, which provides much better performance than the standard BiLSTM solution.

en_core_web_sm is a small English model trained on written web text (blogs, news, comments), that includes vocabulary, vectors, syntax and entities.

### *E. DynamoDB*

Amazon DynamoDB is a fully managed proprietary NoSQL database service that supports key-value and document data structures and is offered by Amazon.com as part of the Amazon Web Services portfolio. DynamoDB exposes a similar data model to and derives its name from Dynamo, but has a different underlying implementation

## IV. FLOW OF WEB-APP

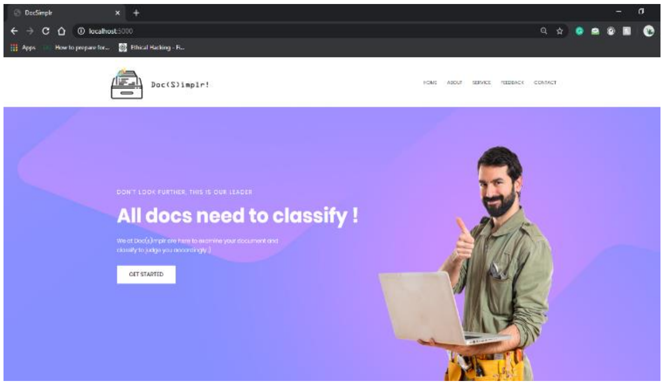
TABLE I. USER-PROCESS FLOW

| **Table Head** | **Doc-simplr website** |
|---|---|
| | ***Web-application flow*** |
| 1 | ***Upload your zip*** |
| 2 | ***Take a break*** |
| 3 | ***Results Generated*** |
| 4 | ***Results Emailed*** |

TABLE I. SYSTEM-PROCESS FLOW

| **Table Head** | **Doc-simplr website** |
|---|---|
| | ***System Flow*** |
| 1 | ***HTTP site request*** |
| 2 | ***Respond from flask & apache web server*** |
| 3 | ***Upload zip of PDF by user and saved in web server*** |
| 4 | ***Python script triggered and result generated using NLP*** |
| 5 | ***Result saved in foam of charts and emailed to the particular user.*** |

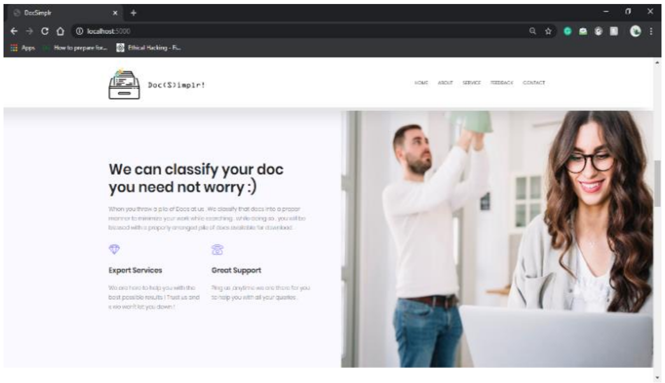
## V. SCREEN SHOT OF UI



All docs need to classify !

We at Doc(s)implr are here to examine your document and classify to judge you accordingly :)

5.0 INDEX PAGE



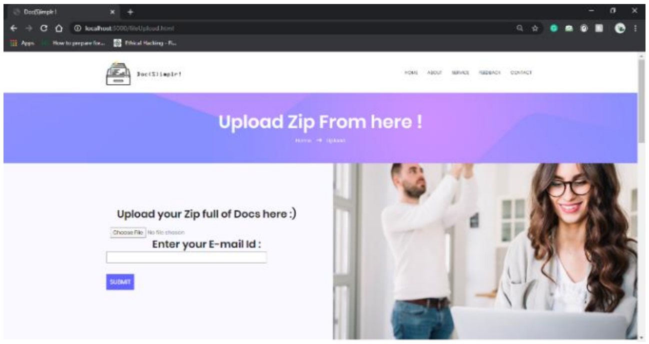
We can classify your doc you need not worry :)

When you throw a pile of Docs at us , We classify that docs into a proper manner to minimize your work while searching , while doing so , you will be blessed with a properly arranged pile of docs available for download .

We are here to help you with the best possible results ! Trust us and we won't let you down !

Ring us anytime we are there for you to help you with all your queries .

5.1 ABOUT

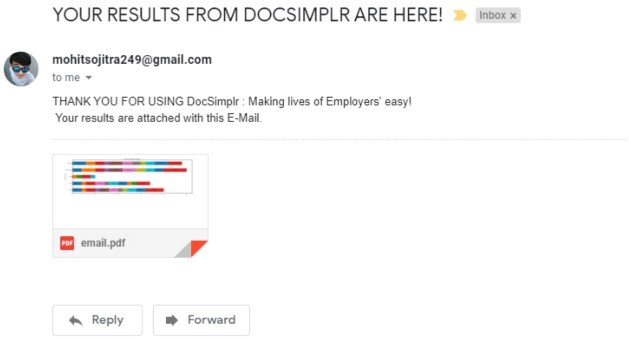


Upload Zip From here !

Upload your Zip full of Docs here :)

Enter your E-mail Id :

5.2 UPLOAD

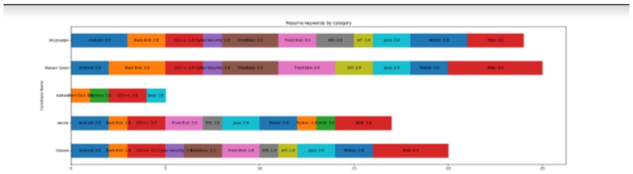

YOUR RESULTS FROM DOCSIMPLR ARE HERE!

mohitsojitra249@gmail.com

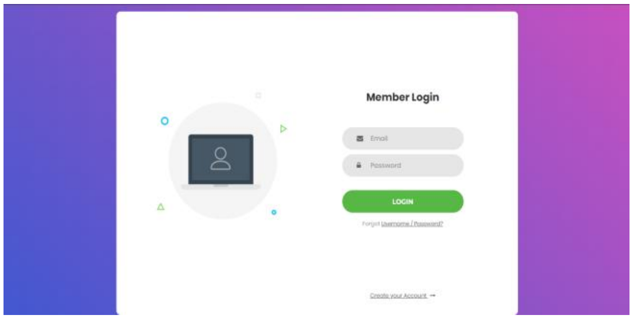
to me

THANK YOU FOR USING DocSimplr : Making lives of Employers' easy!
Your results are attached with this E-Mail.

5.3 GETTING MAIL FROM OUR APP



5.4 CLASSIED RESULTS OF PDF



5.5 LOGIN PAGE

## VI. QUALITY OF APPLCATION

- Easy to navigate, user friendly WebApp
- Available to users 24x7
- Proper classification of the docs
- Sending correct results to user
- Less Latency
- Sending results to correct user

## VII. FUTURE ENHANCEMENT

Provide authentication for two types of users: Registered and Guest users, so that premium registered users can store their docs for any future reference

Provide option of tag selection (i.e. classifications based on which tag the users need in their results)

## VIII. CONCLUSION

Developing an application which analyze the cv and resumes is very challenging task and yet we developed that using python, SMTP services. That gives result to the user via email back. And we learnt how to developed python-based web application using flask.

## IX. ACKNOWELDGEMENTS

This research and project development was encouraged as a part of academic curriculum by Smt. Kundanben Dinsha Patel department of Information Technology, CSPIT, CHARUSAT. We thank our faculties and reviewers whose expertise and support in the field provided and guided us, which greatly assisted our project. Our project reviewers, who provided us their valuable feedbacks and comments to improve our module, also played a very vital role in assisting this research. We thank Dr.Parth Shah, Head of department, Dept. of IT, CSPIT, CHARUSAT and prof. Nirav Bhatt, whose guidance and comments greatly improved the project. We also appreciate ourselves and all the colleagues, whose insight and expertise in the field, lead to the development and successful execution of the project.

## X. REFRENCES

1) https://pythonhosted.org/PyPDF2/
2) https://spacy.io/usage
3) https://flask.palletsprojects.com/en/1.1.x/
4) https://docs.python.org/3/library/zipfile.html
5) https://www.pythonforbeginners.com/code-snippets-source-code/using-python-to-send-email
6) https://www.freecodecamp.org/news/how-to-build-a-web-application-using-flask-and-deploy-it-to-the-cloud-3551c985e492/