**Python based application to extract text from PDF files**

The script has the following functionalities:

Convert batch PDFs from cloud bucket into text with html format and save them to local directory.

**PLATFORMS NEEDED:**

PyCharm or any other equivalent IDE(NOT MANDATORY)

Google cloud platform with billing enabled.

**SOFTWARE NEEDED:**

PYTHON3 functionality and all the necessary libraries which are mentioned in the requirements.txt

**SETUP:**

Download the GITHUB repo zip file from the source.

Unzip the file and open the project in pycharm.

Pycharm community edition is a free IDE for python development.

Create a new virtual environment in pycharm project ide.

After that enter **pip install -r requirements.txt**

This will install all the required libraries to run the OCR project.

**GOOGLE CLOUD SETUP**

In this project we are using google cloud vision api for batch pdf processing.

Google cloud vision for PDFs to document conversion is only possible via Google cloud storage bucket.

The output json file will be processed online and our program parses this json file to generate the html file in our local directory.

Before starting with this project we need a google cloud account with billing enabled.

Under the projects section, select your project and enable API which is named google cloud vision.

After that you’ve created a service account key. And paste that service account key in the service-account-json file into the python project file.

Requirements:

* Gcloud with vision api and billing enabled.
* List of pds to be converted.

First off, we have to upload all the files into the cloud storage bucket.

You can create a storage bucket with a name of your own choice and copy the root path of the bucket.

Eg: gs://test\_ocr

This root path is editable and also customisable via python code. You can uncomment the input code in the python file to have a dynamic root path. I.e, prompt the user to enter the root path of the cloud bucket. (OR) hardcode it into the code by assigning the variable the root path.

Second, we’ve to mention the URI of the pdf file in the storage bucket.

Eg: gs://test\_ocr/agreements/rental\_agreements.pdf

This should be given as an input when the program prompts for “enter the file location”

After taking these inputs the program runs the api and creates a json file with parsed data under the same folder name which matches the file.

The program also creates an HTML file in the local directory of the PC with the name that matches the “file name.pdf”

After the setup run the main.py file.

All the created html output will be opened in the browser automatically and the files can be found in the root directory of the project.