Flipkart

GRiD 2.0

# Electronic Invoicing using Image Processing

**Team Name : SYS_MEET**

**Institute Name: KIET Group Of Institutions**

# Team members details

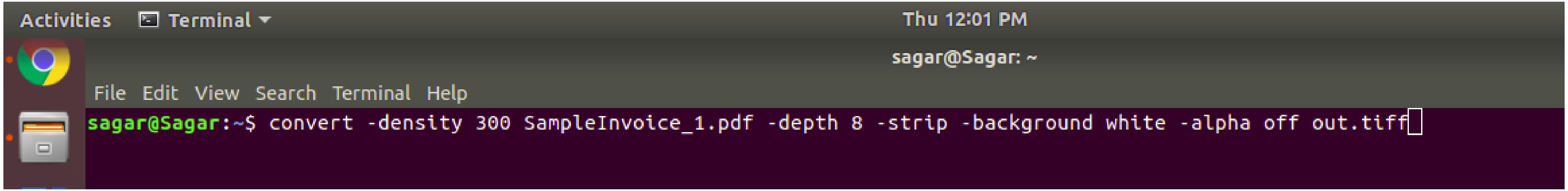
| | | | |
|---|---|---|---|
| **Team Name** | SYS_MEET | | |
| **Institute Name** | KIET Group Of Institutions | | |
| **Team Members >** | **1 (Leader)** | **2** | **3** |
| **Name** | Sagar Guglani | Sarthak Gupta | Yogesh Bhatia |
| **Batch** | 2017-21 | 2017-21 | 2017-21 |

## Objective:

The soul purpose of the project is to develop a cross platform that understands the required details out of a sample invoice and reflect the Text data on a given SpreadSheet format.

## Step 1:

Detecting the type of input file (Image (png/jpeg) or PDF )

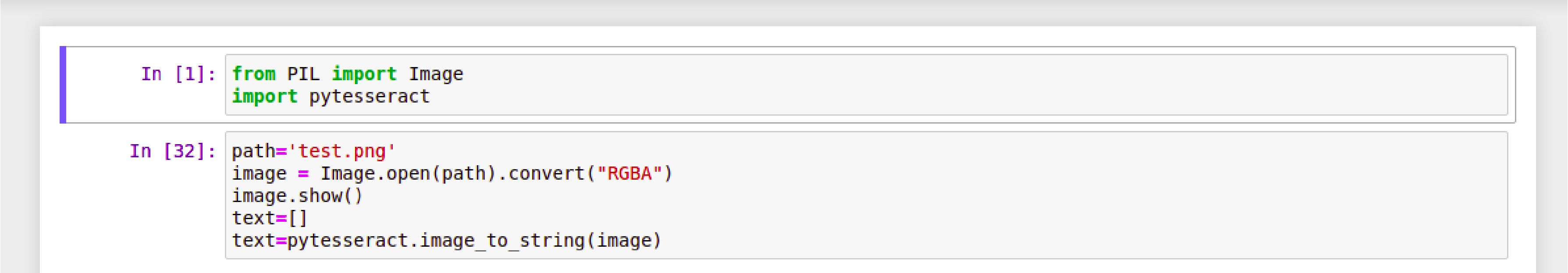


```
sagar@Sagar:~$ convert -density 300 SampleInvoice_1.pdf -depth 8 -strip -background white -alpha off out.tiff
```

The above ImageMagick command detects the format of our input and converts it into .tiff image format of **layered images.**

The tiff format is an image format used sepecially for multiple image storage in a single file for accessing it through layers.

**Step 2:**
Converting the image to RGB format for text detection and clear white backgrounds



```
In [1]: from PIL import Image
        import pytesseract
```

```
In [32]: path='test.png'
         image = Image.open(path).convert("RGBA")
         image.show()
         text=[]
         text=pytesseract.image_to_string(image)
```

**Step 3:**
Detection of text from Image format(.tiff) by Tesseract Library

**Step 4:**
Filtering the data for the desired information exporting to SpreadSheet using Pandas Library.

# Functionalities of Product

- **Product’s USP**: The accuracy of data extraction from any image using PyTesseract is **nearly 100%** as it requires very high resolution image of the range( 200-800 ppi density ).

- **Technology used**: The base is Python3 with the use of libraries like -->
  - ***PyTesseract*** ( a module to use Tesseract functions in Python consol/IDE )
  - ***ImageMagick-6***
  - **Pandas**
  - **PIL/pillow** ( image import in Python )

- There are **no** licensed S/W’s or modules to be used.

- The PyTesseract is an OS independent stand alone library, that does not require a particular template to extract data from an Image. Thus, any new template will raise no hurdle in the working of our model.

# Product Specifications

**> Programming Language:** Python3

> **Modules:** There are several modules required in this project :

- pdf to image conversion
- Image Parsing to layers
- data extraction
- filtering required data
- excel export

**>** The methodolgy to be used to intigrate all the models is yet to be decided, most probably at the coding stage. But the plan is heading towards **PackageImport** so that we may inherit the properties of one module as of a class, and the neccessary data is only visible to the user, hiding the irrelevant information. Taken into consideration the <u>*security of the system*</u>, so that **invoice inputs can’t be shared**

# Product Limitations

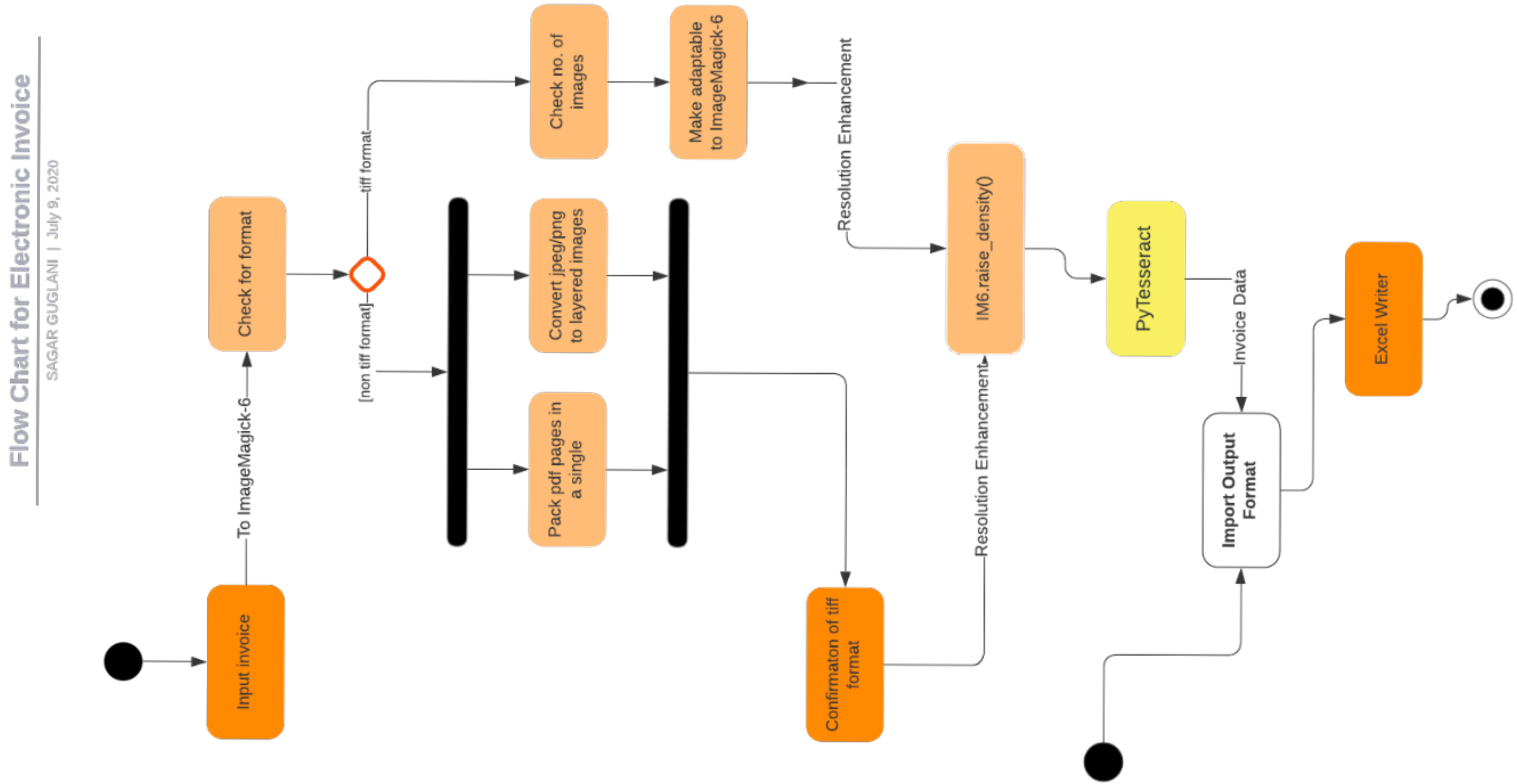
There are no specific limitations for this model,
but there stands a scope of **increased space complexity** of this idea as the ImageMagick library converts the input file to high resolution image.

A single A4 size sheet image may cost around 10-15 MB .

But this is still in favour of accuracy of the Data extracted from the Invoice input.

# Architecture

https://app.lucidchart.com/invitations/accept/ff78163a-17a4-4bc0-b7b6-120723f04f68



Flow Chart for Electronic Invoice

SAGAR GUGLANI | July 9, 2020

Input invoice

To ImageMagick-6

Check for format

tiff format

Check no. of images

Make adaptable to ImageMagick-6

Resolution Enhancement

[non tiff format]

Convert jpeg/png to layered images

Pack pdf pages in a single

Confirmaton of tiff format

Resolution Enhancement

IM6.raise_density()

PyTesseract

Invoice Data

Import Output Format

Excel Writer

# Execution Plan

1. *The basic on-paper prototype is ready.*
2. *Presentation of the model*
3. *The neccessary module files such as IM6 and Tesseract installation step.*
4. *Python3 coding of the project.*
5. *Use of Tkinter GUI to implement the project with user interface development.*
6. *Testing of provided sample invoice data.*
7. *Extracting more test resorces from kaggle.com .*
8. *Conversion of .py file to .exe file.*
9. *Submission*

Flipkart

GRiD 2.0