

Flipkart 

GRID 2.0

Electronic Invoicing using Image Processing

Team Name : noobEz

Institute Name: Indian Institute of Technology, Ropar

Team members details

Team Name	noobEz		
Institute Name	Indian Institute of Technology , Ropar		
Team Members >	1 (Leader)	2	3
Name	Praful Gupta	Rohit Tuli	
Batch	2018-2022 CSE (3rd year)	2018-2022 CSE (3rd Year)	

Functionalities of Product

Electronic Invoicing poses many difficulties due to difference in invoice templates, use of different words for the same thing in different invoices, low quality scans etc. There is a need to build a robust, platform independent and easy to use software that can intelligently process relevant information from the scans across every possible template.

- Our model would **not** require the format of the table data, resulting in high scalability and minimal human interaction in the processes.
- There would be an option to manually check the output before finalising which ensures high reliability.
- Our software will use state of the art models (either custom implementation or pretrained as required) to ensure high accuracy and less errors.
- Our software will have a website and Android/IOS app for easy usage.
- It would be easily integratable into existing SaaS software using IBM Watson.
- It will be able to detect scans which are not vertical or horizontal, manual cropping would not be required.

Product Specifications

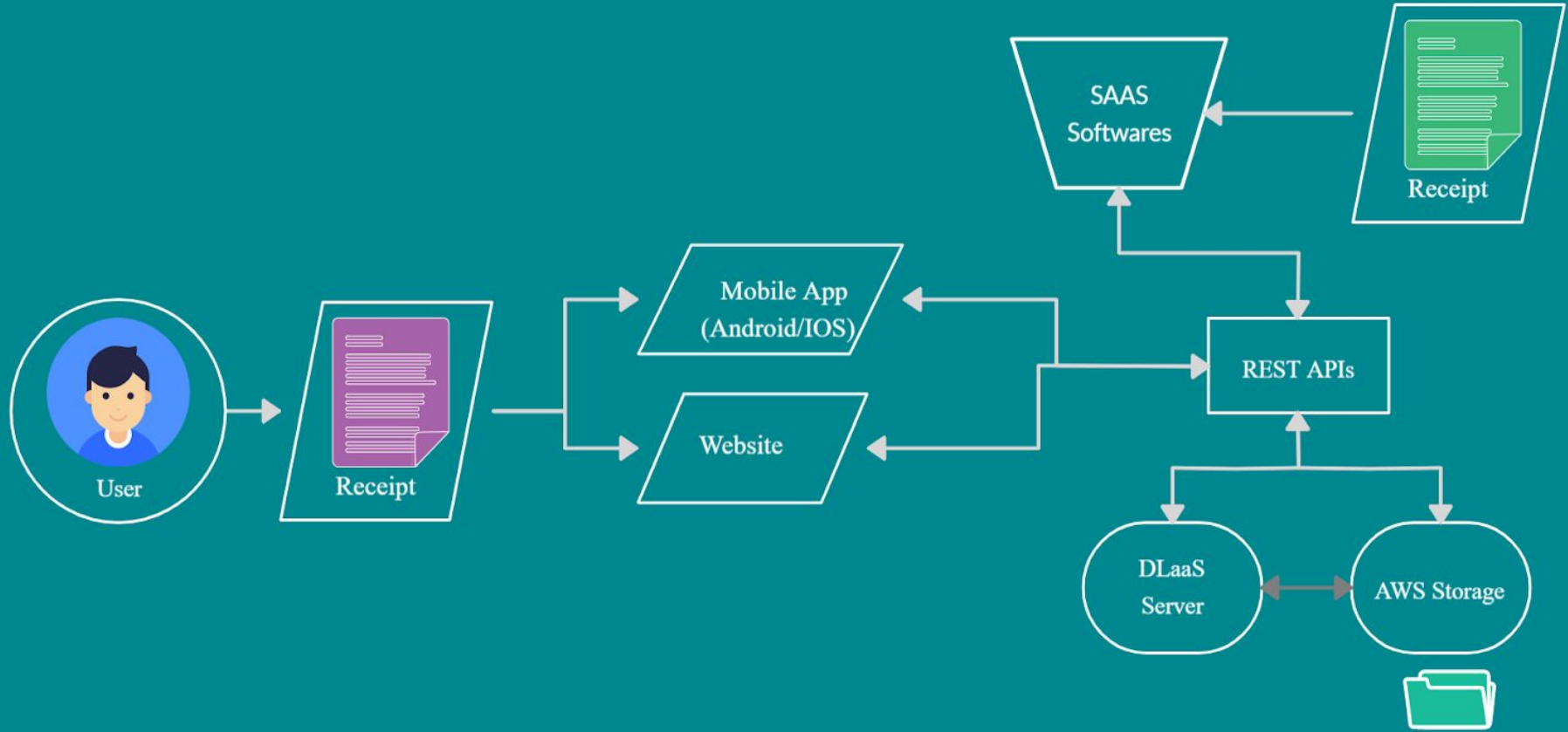
Our proposed solution has 3 main components further explained in details in upcoming slides.

1. Frontend- Website and Android /iOS App that will be used for uploading the document and retrieving required response.
2. Backend- To process the scans using the proposed Deep Learning Model and Apis to integrate with existing SaaS softwares for both uploading the document to storage and downloading appropriate response.
3. Composite Deep Learning Model pipeline-
 - I. Removing Noise from Image and making it suitable for text recognition
 - II. Text Detection and Recognition (OCR)
 - III. Mapping of labels with values, identifying data in tables etc and saving to Json File

Product Limitations

1. Large datasets would be required to train the model, we would use open datasets online for this purpose.
2. We have not supported handwritten bills or bills where text is not visible properly even after preprocessing.

Architecture Flow



Brief on Programming Module

Technology Stack

Tech Details	Mobile App (Frontend)	Website (Frontend)	Deep Learning Model	Server (Backend)
Programming Language Used	Dart	JavaScript	Python	Python , JavaScript
Software Modules Used	Flutter SDK	React.js	Keras , Tensorflow	IBM Watson (DLaaS) AWS , Node.js

Integration Methodology with existing SAAS softwares - REST Apis From the IBM Watson Cloud (DLaaS) Server will be used to get processed data from any existing SAAS software after uploading PDF / Image of Receipt to server.

Execution Plan

Detection Model (Keras, Tensorflow):

1. **Preprocessing of the scanned bill-** To allow for easy detection of data from scans we will perform Cropping, Rotating, Greyscaling, Adaptive Gaussian Thresholding and Noise Removal. Resizing of Image will be performed at this step to make the width and height multiple of 32 (required for EAST Model)
2. **Text Detection-** As the receipts may not be perfectly scanned, some text may be difficult to detect. Moreover many different OCR techniques perform poorly when the receipt has different Fonts, colors and images in it. We experimented with various pretrained models, and we have finally settled on our custom implementation of EAST (Efficient accurate scene text detector) model. [Paper Link](#).
3. **Text Recognition-** We will use the text bounding box coordinates from the EAST model as an Input to Convolutional Recurrent Neural Network (CRNN) model to which we will add a spatial transformer network layer to rectify the text.
4. Any bounding box in which the confidence score for text detection is less than 70 percent will be sent for manual verification to the user, if such an option is selected.

Detection model continued..

6. Gridding of Text- Based on the location of bounding boxes of the detected text, rearrange them into the form of a table by grouping nearby x coordinates into a row and all different y coordinates into different columns.

7.Final Step Use CascadeTabNet model ([Paper Link](#)) to convert the data to CSV file.

Frontend:

1. **Mobile Application** - For easy and faster development Google's cross platform Flutter SDK will be used to develop application for both Android and IOS device. Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia and the web from a single codebase.
 - 1.1 . **Uploading Receipt** - REST Api would be used to upload Receipt to the AWS Storage and retrieve the document URI. Document can be of PDF or any Image format.
 - 1.2 . **Final Response** - The document URI retrieved from the AWS server would be sent to the IBM Watson server using another REST Api where the receipt would be processed using the predefined Pipeline and perform the required action , the required data could be retrieved for the csv file obtained from the api response.
2. **Website** - For responsive and fast website development Facebook's React web framework would be used to develop the website. React is an open-source JavaScript library for building user interfaces. It is maintained by Facebook and a community of individual developers and companies.

The REST apis would be same for all platform as well and these Api can support any other **SAAS** software.

Backend:

1. **Storage** - To have scalability and reliability Amazon S3 Storage Service would be used to store all the receipt and CSV files, An Amazon S3 bucket is a public cloud storage resource available in Amazon Web Services' (AWS) Simple Storage Service (S3), an object storage offering. Amazon S3 buckets, which are similar to file folders, store objects, which consist of data and its descriptive metadata. REST API would be used to upload to the S3 Bucket which can be downloaded in the Pipeline IBM Watson server.
 - 1.1. Uploading Document Api - Restful Api to upload the required document to the S3 bucket.
 - 1.2. Downloading Document Api - Restful Api to download the required document from specific S3 bucket.
2. **File Processing Pipeline** - To have a fast and scalable Deep Learning Pipeline execution IBM Watson DLaaS would be used to maintain the Jupyter Notebook containing the Deep Learning pipeline along with the model.
 - 2.1 Pipeline Api - Restful Api to send the AWS Stored Document Url to the Watson server and get appropriate csv and other detailed required response

The REST apis would be same for all platform as well and these Api can support any other **SAAS** software.

Flipkart



GRID 2.0