

OMR Bulk Analyzer from PDF (Web Application)

Summary

As CBSE has started taking Board examination for class X and XII in two phases i.e., one phase consisting of OMR sheets and other as subjective paper. Every school will conduct Mock examination for them. For OMR sheets particularly, Schools will either need dedicated hardware, or they're checking sheets manually. We can see that first option is very expensive and other one is taking unnecessary time.

We can build a web application which will have certain features, without requiring extra time, we can scan, analyze and export the result in required format as quickly as possible.

This idea will be very helpful for schools, tuition center where they don't need to buy hardware, or evaluate them manually, will save a lot of time or money or both for institutes.

Project Flow

1. We will upload a Sample OMR sheet (Few fields will be editable, like school name, class etc.) so that you can download a customized OMR for examination.
2. After examination, collect all sheets for particular batch, scan all the OMR as document and build a PDF file using any scanner like (Microsoft lens, Adobe Scanner).
3. Upload the PDF on the web application, it will check for pdf format and then start analyses of each page of pdf.
 - Either web application is not able to scan the page properly, due to bad orientation, it will store the page number of pdf and proceed ahead, we will return the page numbers of bad orientation pages
 - If page is of good orientation, we will proceed ahead and analyses the result.
4. In end we will export the result the result of analyses in CSV or JSON format as required.
5. We can use parallel processing too for analyses for multiple PDF files, or multiple pages of PDF for faster compute time.

Technical Stack

Web application: Frontend -> Using templates, html, JavaScript, CSS or sass.

Backend -> Flask or Django(For multi-processing)

PDF Scanning -> We can use **PyMuPDF** library of python for extraction of images from each page of pdf and further using **OpenCV** for analyses of image. We can use **Pandas** for storing result.

Hosting -> As we have sessions of using **Microsoft Azure**, will try to learn and host on Microsoft services, but in case something fails, I have experience of using **AWS EC2** instances, will have a backup there. Will use Services like **Nginx** and **Gunicorn** for production level deployment, with **SSL** certificate from **LetsEncrypt**