**Research Project: YMS**

**Report**

**Introduction:**

In this research project, YMS came up with a problem of long hours and human efforts in generating elements of PDF file on their EMR system, so to automate this process Seneca team has proposed a solution by creating a new system named Universal PDF Reader. This report provides the detailed description of mechanism and benefits of using the universal PDF reader system.

**Problem description:**

According to current working strategy for generating PDF file elements on EMR is completely manual which consumes a lot of human efforts and time. Therefore, company doesn’t want to spend lot of time and efforts on this process which leads to the decision to automate this process.

**Proposed Solution:**

The solution proposed by Seneca team to this problem is universal pdf reader. The universal PDF reader system reads and scan the fillable pdf files as input and process them to give the html file of the interactive elements in pdf files, this html file can be used to upload the interactive elements of pdf file in EMR (Electronic medical record) system.

**Requirements of system:**

1. Installation of integrated development environment for python language
2. All the interactive elements of PDF file should be fillable
3. PDF file should not be protected

**Mechanism of system:**

This system is made up with following computations to decrease the human efforts in manually creating a pdf in EMR from the existing fillable PDF:

1. Reading Fillable PDF:

For reading the fillable PDF file, a python library pdf query was used because this library gives the tree structure of fillable PDF file which makes it easy to convert to xml format. This library not only gives the elements of pdf file in tree structure but also their properties such as their position in terms of right, left, top and bottom. Then this tree structure is saved in xml format. The reason to choose xml format to save tree structure is because xml document is always descriptive which makes it easy to parse and get the elements in case of tree structure.

1. Parsing xml to get properties of interactive elements:

The next step was to parse the xml and get the elements and their properties – their type ﻿(checkbox, text or text fill up), position (x0, y0, y1, x1) in form of data frame (csv file). This was done by ET library which drags down from root to the lowest layer in file and get the all the branches.

Therefore, a data frame was made containing all the elements and their properties.

1. Generating html script:

After this and html script was created using the data frame created in previous step. Using the type of element different syntax of html were used in script. The position properties were used to position the interactive elements in an order.

1. Linking html script to database of EMR:

To make this html work on EMR system, it should be linked with its database, so a database dictionary provided by company in form of csv file was used to link the elements with database of EMR system using html syntax. Therefore, output in form of a html file for EMR system is given by system.

**Regular updates to be made in system:**

The universal PDF reader system is linked to the database of EMR system so a regular update must be made in csv file of linked history which contains the name of field in pdf file and its database name. For new pdfs which might not have their database name stored in linked history so these new names must be updated from time to time for working with this system.

**Testing and Results:**

This system was tested on the various kinds of pdf files. The system has been tested on all the files given company and it pretty much worked well. There were cases when file was not completely fillable which resulted into error so file should be completely fillable. Sometimes the name of fields existing on document does not appears exactly same in html file, so this requires a manual effort to change that exceptional name of fields.

**Conclusion**:

No doubt there were some exceptional cases where the results were not as desired, but this system worked for 80% of given general pdf files and produced satisfying results. It is believed that the 70% of manual efforts will be decreased by this system.