**BOT1 Daily placement**

The process depends on two input files, one Vendor file **KMH510D1\_190423.csv** and one of the Client file **BER\_ACK\_20190423.xlsx, XTEND\_ACK\_20190423.xlsx** ..etc. These files will be generated every day in the respective folders.

Current process:

1. Every day user need to check for input files for current date in the respective folders.
2. If files are available, he needs to pick vendor file based on yesterday’s date and any one client file as current date file name.
3. He needs to create one excel file manually for output with four sheets MH File, BER File, MH Not BER and BER Not MH.
4. He needs to copy entire data in csv file and paste it into MH File sheet in output file.
5. He needs to copy entire data in BER file and paste it into BER File sheet in output file.
6. He needs to run VBA Macro to filter MH File sheet based on client file, if client file is BER he has to filter the data based on COS column containing BA, BE.
7. He needs to run another VBA Macro to compare above filtered MH File data with BER File sheet data based on common column ACCOUNT NUMBER.
8. Finally, he needs to write matched records from above criteria to MH Not BER and BER Not MH sheets.
9. The same process needs to repeat for the client files available for current date.

Issues:

1. Lots of Human effort
2. Processing time is more
3. Human errors
4. More workforce

RPA Process:

1. Using RPA, we develop one bot to perform above discussed process without human intervention.
2. The bot can be scheduled to run at specific time.
3. Bot looks into respective folders to get input files based on current date, it will get all the input files folders path from configuration file.
4. It uses inbuilt functionalities to filter and compare data and it generates output file.
5. Bot repeats the same process for all the client files available for current date from the given folder.

Advantages:

1. Bots can be scheduled
2. The processing time is very less
3. Greater productivity
4. less error prone
5. cost savings

**BOT5 Web Posting**

The process depends on one input excel file, containing patient data of different insurance companies These file will be generated multiple times on every day.

Current process:

1. Every day user need to check for input file for current date in the respective folder.
2. He needs to open the file and check for the name of the insurance company (AETNA, BLUECROSS...etc.) for all the records.
3. If it is AETNA, he needs to navigate to AETNA web portal and need to authenticate himself
4. He needs to start entering the data into AETNA portal from input file cell by cell from excel
5. Once he enters all the data of any one record he needs submit it to the AETNA site then he needs to start the pixcert application to submit entered data screen shot along with patient information.
6. Once he is done with above two processes Aetna and pixcert posting, he needs to write the status of record submission to Health Quest main frame application.
7. He needs to repeat the same process for all the records in input file.

Issues:

1. Lots of Human effort
2. Processing time is more
3. Human errors
4. More workforce

RPA Process:

1. Using RPA, we develop one bot to perform above discussed process without human intervention.
2. The bot can be scheduled to run at specific time.
3. Bot looks into respective folder to get input file based on current date, it will get all files path from configuration file.
4. If the file available for particular run, it opens the file check for insurance company type the it will navigate to related portal and starts entering the data after submitting the record to payer site takes the screen shot and it triggers the pixcert application it will submit it there.
5. Then bot triggers the Health quest application to submit the record status and it repeats the same process for all records in the file
6. Once it processes all the records it will close all the applications opened by the bot.

Advantages:

1. Bots can be scheduled
2. The processing time is very less
3. Greater productivity
4. less error prone
5. cost savings