HDBFS\_Project Details:-

As discussed : -

* Actual requirement:
  + Need to find whether the Insurance MEF form is collected/not using per-defined context in the pdf file before processing the agreement
  + This is going to used by Insurance team for processing their back logs in quick manner.
  + 18- 50 pages approx & size of the each document would be around 2 mb max
  + Approx 35k agreement documents were processed per month currently
  + Only to identify few set of context in the image/pdf file in the first phase and store those information into db/excel file
  + Enhancing the OCR capabilities based out on outcomes (expected results will be 100%).

PFB bprg team comments for the requested queries:

* Will there be a separate folder designated for each individual's documents?
  + Common Folder will be used for picking the source document/Or from DMS api directly/Internal Portal
* How many documents are expected to undergo optical character recognition (OCR) for a single person?
  + Purely a batch process, no manual intervention is required
* What is the anticipated number of documents to be scanned in a single processing instance?
  + 1k documents/day
* Considering the monthly usage of this system, what is the overall count of documents?
  + approx 50k per month, current back log would be around 1l
* What will be the documents source location?
  + From DMS/ Internal boss Portal
* What will be the image quality of the document?
  + Undefined
* List of all fields that need to be extracted form the documents?
  + Extract Loan number from MEF form, Master policy Number, "Aditya Birla Sun Life Insurance Company Limited CAPITAL"
* Any time frame within which the OCR should be completed?
  + Need more clarity
* What will be the languages of text in the documents?
  + Currently in English

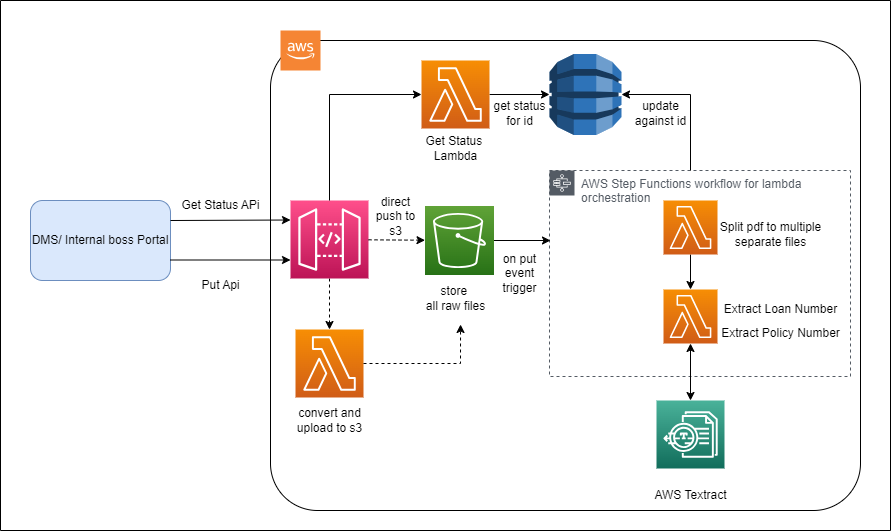
**Introduction: -**

Nowadays working on documents and getting valuable data manually is time-consuming to overcome that work we are creating an automated pipeline that will work on documents and give us the valuable data. This whole process must be Automated so for that we are using AWS services and Python scripts.

**Aim objective: -**

The aim of the project is to create an Automated Pipeline that will get invoked on a certain event and it will work on the data with the help of AWS services and Python scripts to extract the loan number, Policy number from documents and store them in the database.

**Architecture: -**



**Implementation: -**

**Step 1: -** pushing data from DMS/Internal boss portal to the S3 bucket by using API.

Created Api using API and Applied it on Postman, through PUT request pushed data file in s3 bucket

**Step 2: -** When the file gets stored in s3, based on this event Step function gets invoked and the Step function contains two Lambdas

To invoke the Step function, Enabled the AWS Event bridge on Bucket and then created a Rule

**Step Function: -**

**First Lambda: -**

The first lambda contains a script that will access the file from s3 and will create a new folder in the same pdf folder with the separated pages of the pdf

modules used in script: - “Boto3” to interact with AWS services(s3), “PyPDF4” to do operations on pdf

**Second Lambda: -**

After completion of the first Lambda, the second lambda will get invoked which contains a script that accesses the folder that contains the separated pages of the file and iterates through the folder and analyzes each page of the file, and extracts the text of the page and identifies the fields and stores those fields in dynamo db table.

Modules used in the script: - “Boto3” to interact with AWS services(s3), “PyPDF4” to do operations on pdf, “Textract” to extract the text from the file, “regex” to identify the fields

Third lambda: -

After completing the whole process and data gets stored in the dynamo db table there is one more lambda which gives the status of the process the data got stored in the table or not

Explanation: -

Data or Files are stored on DMS/Internal boss portal So by using API we are importing files and storing them in S3 Bucket, when files get stored in S3, because of this event step function will get invoked, step function contains Two Lambdas which are going to work on files.

first lambda will access the file from s3 and will create a new folder in the same pdf folder with the separated pages of the same pdf, the same lambda will work on each file.

The second lambda will get invoked,

The second lambda will access each folder of pdf containing separated pages. The lambda will extract the text from each page and then use regex (regular expression) to identify the task of the field that will be performed on that text.

After identifying the fields, those fields will be stored in the dynamo db for that dynamo db table will be created and those fields will get stored in each cell of table.

Problem statement: -

* + 1. If we upload data in S3 using API Gateway then u can do it with a time-only one file but if u want to upload a folder that contains multiple files, then for that u can create a zip and u can store that folder in s3 using API Gateway but then u must create the lambda function which contains a script which unzips the file from s3 and stores it back in s3
    2. First lambda will work on each customer of data and separate or it will work on only one customer of data and invoke the second lambda then the second lambda will work

API Gateway: -

click “Build” Rest API > choose the protocol = REST > create new api = NEW API >

API name= any name > endpoint type= regional

click Actions > Create resources = resource name= Bucket name(this is variable API in which are going to give the bucket name) > resource path (give curly brackets as placeholder) > create resource for filename same process with filename variable > actions >create method > put>aws services > aws region > aws service > http method > action type(use path overide)>path override({bucket}/{filename}) execution role(IAM role acess ARN) > connect handling(covert to binary) > save > integration request> url path parameter> add path> name:- bucket and mapped from:- method.request.path.bucket > tick click > same for filename(name :- filename and mapped for:- method.request.path.filename) > method execution > settings > binary media types > add binary media types(application/pdf or \*/\* for all) > save changes > actions > deploy API > deployment stage(new stage) > stage name(any) > deploy > copy URL > go Postman > + sign > paste API (give /bucket\_name/filename.pdf(by the name file is going store) > make rerquest put > body (binary)> select file > SEND > output(1)

1. API to S3 single file Done
2. s3 event step function gets triggered Done
3. step function: -
4. first lambda: -

separation function Done

1. second lambda: -
2. text Extraction
3. Storing in dynamo db

Implementation: -

First step: -

Import data from DMS and store data in S3

Second step: -

Separating each page of pdf and storing those separated pages in the same folder of the same customer

First Lambda: -

Modules used: -

* + 1. BOTO3: - To Access AWS S3
    2. PYPDF4: -To read the pdf and create a new pdf
    3. Io: - To convert the byte to int

Third step: -

Accessing each separate page of the pdf and extracting the Loan number, Master Policy, “Aditya Birla Sun Life Insurance Company Limited Capital”

Fourth step: -

Create a table of dynamo db and store the fields of each customer in cell of table