1. **Project Title: -** Document Analyzer
2. **Problem Definition: -** Creating an application which can extract data from any document (hardcopy) and store them.
3. **Project Description: -**

Basically, our application is going to extract data from the document and store it in whichever format the user wants it to be stored.

* Amazon Textract uses Optical Character Recognition (OCR) technology to automatically detect printed text, handwriting and numbers in a scan or rendering of document, such as a legal document or can of a book.
* **Document Text Detection: -** It can help to detect all the words and hence the sentences from the document. This can be use to further process or get the information from the form
* **Form Extraction: -** Amazon textract enables you to detect key- value pairs in document image automatically so that you can retain the inherent context of the document without any manual intervention.
* **Table Extraction: -** Amazon textract preserves the composition of data stored in tables during extraction. This is helpful for the documents that are largely composed of structured data, such as financial reports or medical records that have column names in the top row of the table followed by rows of individual entries
* **Handwriting Recognition: -** Many document such as medical intake forms or employment application contain both handwritten and the printed text. Amazon Textract can extract printed text and handwriting from document in English, whether is free form text or text embedded in tables and forms.

1. **Technologies**: -
   * Frontend: C#, Windows Forms
   * Backend: Python, JSON, APIs, Docker
   * AWS: Lambda, Elastic Container Registry (ECR), API Gateway, Textract (OCR Service).
2. **Usage of Technologies: -**

* **C#:** It will be used to built the Graphical Interface of the application.
* **Windows Forms:** It is open-source graphical class library used to develop GUI using C#.
* **Python:** It will be used to write a serverless application on AWS Lambda where the file gets processed (converted to images, analyzing from AWS Textract).
* **JSON:** It will be used to handle the response from API call.
* **APIs:** API will be used in our software for uploading and downloading the documents.
* **Docker:** It is an open-source containerization platform. The docker will be used to create an image for our python code and setup the environment.
* **AWS ECR:** It is a fully managed container registry that makes it easy to store, manage, share, and deploy our container image. We will create a container using Docker and store our Python code in AWS ECR.
* **AWS Lambda:** It is a service used to deploy serverless applications by writing code and private container image. So, we will use it to deploy our Docker image from AWS ECR to AWS Lambda. And hence the serverless application is deployed.
* **AWS Textract:** It is a service which can be used to analyze documents using OCR technology. It will be used in our process which will be written in python.
* **AWS API Gateway:** It is a service used to create an API. This will help in our software to upload document and request AWS Lambda. After execution we will get response in json format.

1. **Process: -**

* User uploads document.
* Document is sent to AWS Lambda with API call using AWS API Gateway Service.
* Document is then converted to images.
* The images are then analyzed one by one and the data gets stored.
* After all the images are analyzed, the response will be sent using AWS API Gateway as json format.
* Now user will be notified that the document is analyzed.
* The user is now able to download the analyzed document.