READMe\_PAD\_Project Components (Databricks)

**Overview:**

The script is designed to process text files, specifically targeting the extraction of "Project Components" sections. It reads each file, applies regular expressions to locate the start and end of the desired sections, and captures the content in between. The script first attempts to remove the "Table of Contents" to facilitate cleaner data extraction. It then employs a series of pattern matches to accurately capture the "Project Components" section, accounting for variations in document structure. The extracted text, along with corresponding project IDs, is then saved to a CSV file. This script is particularly useful for aggregating detailed project component information from multiple text documents in an organized and efficient manner.

**Running the Code**:

There are two cells that can be run. Run the first one if the files are year 2018 and onwards. Run the second part if the files are from before the year 2018. Please follow the steps below.  
  
1) Set Up a Databricks Workspace:

* If you don't already have a Databricks account, create one.
* Log in to your Databricks workspace.

2) Import Data:

* Upload Your Data: Since Databricks operates in a cloud environment, you need to upload your (downloaded) .txt files to a cloud storage that Databricks can access, like AWS S3, Azure Blob Storage, or Databricks DBFS (Databricks File System).
* Modify File Paths: Update the folder\_path and output\_path in your script to point to the locations where your files are stored in the cloud environment.

3) Create a Notebook:

* In your workspace, create a new notebook.
* Choose Python as the language for the notebook.

4) Adapt the Script:

* Copy the Python script into cells in the notebook.
* If your script requires any external libraries (like re for regular expressions, which is built-in for Python), you need to install them. This can be done either by installing the libraries to the cluster or by running %pip install library-name commands in the notebook cells.

5) Run the Notebook:

* Execute the Notebook: Run the cells in your notebook. You can run them individually or all at once, depending on how you've organized your script.
* Monitor the Execution: Watch the cell outputs for any errors or messages. Databricks provides detailed logs that can help in troubleshooting.

6) Accessing Output:

* Once the script runs successfully, the output CSV file will be saved to the specified path in your cloud storage.
* You can then download the CSV file or use Databricks to directly analyze the data within the platform.