**The project has three parts:**

**First part :**

1. Read 10 files (.txt)
2. Apply tokenization
3. stemming

**Second part :**

1. Build positional index and displays each term as the following :

<***term****,* number of docs containing ***term***; ***doc1***: position1, position2 … ;

***doc2***: position1, position2 … ;

etc.>

1. Allow users to write phrase query on positional index and system returns the matched documents for the query.

**Third part :**

1. Compute term frequency for each term in each document.
2. Compute IDF for each term.
3. Displays TF.IDF matrix.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Term | Doc1 | Doc2 | Doc3 | Doc4 | Doc5 | Doc6 | Doc7 | Doc8 | Doc9 | Doc10 |
| Term1 |  |  |  |  |  |  |  |  |  |  |
| Term2 |  |  |  |  |  |  |  |  |  |  |
| Term3 |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

1. Compute cosine similarity between the query and matched documents.
2. Rank documents based on cosine similarity.