

SMART SEARCH ENGINE - FINAL REPORT

**CS 6359 Section 001
Object-Oriented Analysis & Design
Spring 2022**

| | |
|---------------------------|---------------------------|
| Instructor: | Dr. Lawrence Chung |
| Teaching Assistant | Kirthy Kolluri |

SMART-SEARCH ENGINE GITHUB LINK:

<https://github.com/krishnaty3/SmartSearchEngineDocs.git>

TEAM MEMBERS

- ❖ YELISETTI KRISHNA TEJA (NET-ID : KXY200016)
- ❖ SATYA SOMEPALLI (NET-ID : SXS190436)
- ❖ SANJANA PENMETSA (NET-ID : SXP190149)
- ❖ PREETHAM RAO GOTTUMUKULA (NET-ID : PXG210001)
- ❖ HARSHITH RAVIPROLU (NET-ID : HXR180005)
- ❖ SAKETH DASAVATHINI (NET-ID : SXD190016)
- ❖ B MOUNIKA (NET-ID : MXB210007)
- ❖ THOTA JAYASHREE SANTHOSHI (NET-ID : JXT210011)
- ❖ SAI PRANAV REDDY DONTIDI (NET-ID : SXD200125)
- ❖ APUROOP PARAVADA (NET-ID : AXP210033)
- ❖ YOGESH BALA (NET-ID : YXB200007)
- ❖ SRINATH REDDY MAVILLAPALLY (NET-ID : SXM210047)
- ❖ KIRAN DEVARAJ RAJ (NET-ID : KXR190038)

Contents

| | |
|---|----|
| INTRODUCTION..... | 3 |
| BASIC SUMMARY | 3 |
| PROJECT OVERVIEW | 3 |
| DEFINITIONS, ACRONYMS, AND ABBREVIATIONS..... | 3 |
| Abbreviations..... | 4 |
| TOOLS/TECHNIQUES USED | 4 |
| LIMITATIONS..... | 5 |
| FUNCTIONAL REQUIREMENTS | 5 |
| NON-FUNCTIONAL REQUIREMENTS | 6 |
| CONTRIBUTION OF TEAM MEMBERS | 7 |
| BASIC FLOW IN PHASE-I..... | 10 |
| BASIC FLOW IN PHASE-II | 10 |
| USECASE TEMPLATES | 11 |
| UML DIAGRAMS | 15 |
| USECASE DIAGRAM | 16 |
| CLASS DIAGRAM..... | 16 |
| SEQUENCE DIAGRAMS..... | 17 |
| STATE TRANSITION DIAGRAM-1 | 20 |
| STATE TRANSITION DIAGRAM-2 | 21 |
| ACTIVITY DIAGRAM-1..... | 22 |
| ACTIVITY DIAGRAM-2..... | 23 |
| TEST CASES..... | 24 |
| TEST-CASE RELATED SCREENSHOTS: | 29 |
| USER-MANUAL..... | 34 |
| PROTOTYPE IMPLEMENTATION..... | 38 |
| REFERENCES | 44 |
| IMPORTANT DATES..... | 44 |

INTRODUCTION

BASIC SUMMARY:

Smart Search Engine is a movie-based search engine where a user can search for movies using genre(s) or movie name(s) or actor name(s) or any word(s) related to the movies. The most relevant results (in this project: most relevant movie related Wikipedia-URLs based on the word(s)/query searched) will be displayed at the top. This search engine also supports multi-phrase query/words search (example: multiple-genres based search). User has an option to select AND/OR clause while searching a multi-phrase query. User can also click on a specific link displayed in the results and navigate to that link for more details on that result. User will be displayed with a two-line description below every Wikipedia-URL in the results. User will have an option of navigating through multiple results using the pagination functionality implemented. If user searches for any non-English words or with a query of no relevance to Movies, then he/she will be redirected to an error page. User can continue searching until he or she wants to exit or close the browser.

PROJECT OVERVIEW:

For the project to run properly, an Admin and a preprocessor python module are the most important parts of the Smart Search Engine.

Admin: Admin is the one who collects relevant URLs and tries to expand the database. After successful collection of URLs, admin saves or updates the already saved URLs for the pre-processor to process. Admin's task is to collect URLs, save or update it in the File system and then run the pre-processor.

Pre-processor: Pre-processor is a python module in the project which scrapes through all the URLs stored by admin in a single file. Pre-processor primarily saves the Term along with their positions in a respective document.

DEFINITIONS, ACRONYMS, AND ABBREVIATIONS:

Definitions:

- 1. Search Interface:** The user interface consists of a search bar where the user searches a keyword/query related to Movies. User has an option of selecting AND/OR clause while searching for a multi-phrase query. While results are being displayed, User can navigate through results using Pagination and User can also specifically click on a link to explore more.

- 2. Crawler and Indexing:** It is a backend component where the crawler will crawl through the websites available, and the content on the websites will be indexed for faster information retrieval.
- 3. Database:** Database is our backend component which would store information of different websites along with the associated keywords.
- 4. Dataset:** The dataset is a collection of data containing different website URLs, the title, description about the websites, and different keywords that relate to a particular URL.
- 5. TF-IDF:** In information retrieval, tf-idf (also TF*IDF, TFIDF, TF-IDF, or Tf-idf) is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is frequently used as a weighting factor in information retrieval, text mining, and user modeling searches.

Abbreviations:

- UML - Unified Modeling Language
- URL - Uniform Resource Locator
- TF-IDF - Term Frequency - Inverse Document Frequency
- NLP - Natural Language Processing

TOOLS/TECHNIQUES USED:

We used multiple tools/techniques for our search engine:

- Flask (Web framework)
- Beautiful Soup (Web scraper)
- Word Stemming (Words to their root form)
- TF-IDF Vectorizer (Term frequency and Document frequency)
- HTML, CSS, Bootstrap and Javascript
- Page ranking using scores (To display URLs in the decreasing order of their relevancies)
- Python for back-end coding
- Star-UML software for UML diagrams

LIMITATIONS:

- User can search for only English phrases which should be preferably related to Movies to get a few search results.
- For the search engine to work, the pre-processor must have been run at least once by the admin before the application is hosted.
- Some Wikipedia URLs might have been expired so admin should regularly run the preprocessor to avoid this issue.

FUNCTIONAL REQUIREMENTS:

Following are the functional requirements which are covered in the Smart Search Engine project:

- ★ **Domain specific search Engine:** One word/multiple words search which will fetch results from a database which has movie-related information. User has to search something related to movies, for example, genre(s), movie name, director name, actor names(s) or actress name(s), etc. For example, user can search for ‘drama’ or ‘horror’ or even ‘drama horror’ together.
- ★ **Logical AND/OR Implementation:** Logical OR based search was implemented in the first phase of the project. In phase-2 of the project, Logical AND based search was added along with OR based search. User will have a drop-down list present in the User Interface where user can select either ‘AND’ or ‘OR’ and then search. If user doesn’t select any clause in specific, then the default OR-clause will be taken into consideration while displaying the results. For example, if we search for ‘drama horror’ and choose the ‘AND’ option, then the links which mandatorily matches both the words will get displayed in the results, that is a Wikipedia link containing both the words in it. Another example would be ‘drama horror’ with ‘OR’ clause which will display all the results with either of the words present including ‘AND’ case results too.
- ★ **Ranking/Ordering of URLs:** In this project, we are displaying the most relevant URLs first or at the top. Here, the relevancy is being calculated based on TF-IDF scores of the words searched in the displayed URLs. For example, if user searches for ‘drama’, then the most relevant URL will be a link with the highest TF-IDF score of the word ‘drama’ in the corpus of URLs. Here, results will be displayed in the decreasing order of TF-IDF scores.
- ★ **Filter Stop-words:** In this project, stop words will be omitted from the query which user searches for. For example, if user searches for ‘Drama Horror is’ then ‘is’ here is a stop-word which will be filtered. Examples of some stop-words are: ‘the’, ‘and’, ‘or’, ‘do’, ‘for’, etc.
- ★ **Stemming:** Stemming of the word means bringing the word to its root form. Here, the word(s) being searched will be stemmed to their root forms and based on the root forms of the words, it will be compared with the database and relevant results (URLs) will be

outputted. For example, if the user searches for ‘Flying’ then the suffix ‘ing’ will be removed and the input will be brought to the root/stemmed word which is ‘Fly’.

- ★ **Auto-suggestion:** In phase-2 of the project, autosuggestion of the input has been implemented. When user searches for something, he/she will be suggested with a query of word(s) aligning with the words he/she is typing. For example, if user starts the search by typing ‘dr’, then the user will be suggested with a query of word(s) such as drama, drama movies, drama film, drama horror, etc.
- ★ **Pagination:** In the phase-2 of the project, pagination has been implemented for better UI support. A maximum of 10 results will be displayed per page and all other results will be distributed accordingly (to next pages). For example, if there are a total of 32 results, then there will be 4 pages to navigate where the first three pages will have 10 results per page and the last page will have the last 2 results.
- ★ **404 Error page:** In the phase-2 of the project, Page-404 has been implemented where if there are no results based on the user search, then user will be navigated to the Results-not-found page. This functionality has been added for better UI implementation compared to Phase-1 of the project. For example, if user searches for an irrelevant word ‘aaabbbcddeeg’ and since there’s no match in the movie database with the mentioned word, the browser will display ‘No Results found’ Page.

NON-FUNCTIONAL REQUIREMENTS:

Following are the non-functional requirements which are covered in the Smart Search Engine project:

- ★ **URLs-addition:** To expand the movie database, 1000 URLs have been scrapped and added to the movie database in the phase-2 of the project to fetch multiple and more relevant results based on user-search.
- ★ **Localization:** The project has been locally hosted and can be hosted on any local system or machine. A readme file along with a minimum package requirements file will guide on how to run this project or how to deploy this project locally.
- ★ **Maintenance:** In the phase-2 of the project, logging has been implemented which will help resolve the issues quickly as each and every module implementation will be logged into a file specifically (this file will log the flow of events or function calls happening in the back-end) through which problems can be understood or analyzed.
- ★ **Removal of outdated URLs & Addition of new URLs (Consistency of the Database):** To remove the expired URLs and add new URLs in the movie database, admin can run the pre-processor frequently (for example, weekly once) which will scrape the newly added links to expand the database and the expired links in the existing database will be removed.

In this way, consistency of the database can be retained. Note: This step also comes under maintenance of the project.

- ★ **Friendly User-Interface:** In the phase-2 of the project, the user interface of the Smart Search Engine has been changed compared to phase-1 where user will have an additional option to use the and/or clause while searching for multiple words. Pagination has also been implemented in phase two for better navigation among the links. To show the throughput, number of results along with time taken to display the results are also being displayed in the UI in phase two.
- ★ **Portability & Usability:** As previously mentioned in the Localization non-functional requirement, Smart Search Engine will have an option of portability where the project can be easily deployed on any local machine following a guide on how to setup the project using the read-me file and the requirements file. Smart Search Engine is very easy to use in nature taking both admin and user into consideration.
- ★ **Low-Latency and High-Throughput:** In phase-2 of the project, Low latency and High Throughput is maintained while displaying the results. Although the movie database has been expanded and multiple URLs have been scrapped, the low latency is still maintained and as mentioned previously the user interface will show the time taken to display the number of results which supports the high-throughput and low-latency functionality.

CONTRIBUTION OF TEAM MEMBERS:

Each and every team member was equally involved in every phase of the project including Preliminary Project Plan, Requirements Specification, UML Diagrams, Coding/Implementing the project, PowerPoint Presentation and creating initial and final reports. Everyone was invited to a teams meeting (twice/thrice a week) where every member was assigned a specific task and had to present a solution by the next meeting. Every team member responded in a timely way and the project evolved in a timely manner. Every sub-team used to internally connect to discuss their tasks in detail. Everyone shared their technical knowledge for this project and implemented the project in a modular fashion where code was divided into small Python code tasks and that's how a final code was implemented at the back end. User-Interface was developed in a timely fashion where every sub-team shared their knowledge of the front-end to come up with an efficient and friendly interface. Every member was involved while drawing UML diagrams to have all perspectives, suggestions and multiple feedbacks about the visual modelling.

Below are the three sub-teams of 13 members along with major responsibilities of every team:

SUB-TEAM I:

- 1.Yelisetti Krishna Teja- Group Leader
- 2.Satya Somepalli
- 3.Sanjana Penmetsa
- 4.Preetham Rao Gottumukula

Major Responsibilities of Sub-Team I:

- Expanded the database: Collected additional URLs for phase-II of the project
- PPT-I AND II preparation: Took part in creating first and second presentation
- Updated Sequence Diagrams: Involved in instance sequence diagrams in phase-II
- State Transition-Diagrams: Involved in state transition diagrams during Phase-II
- User-Interface implementation: Updated the user Interface and implemented it in View templates for successful rendering of every page.
- Python module changes: Implemented & updated page-ranking algorithm, updated the efficiency of python code from Phase-I to phase-II.

SUB-TEAM II:

- 1.Harshith Raviprolu
- 2.Saketh Dasavathini
- 3.B Mounika
- 4.Thota Jayashree Santhoshi

Major Responsibilities of Sub-Team II:

- Updated class-diagram: Involved in updating the class diagram from Domain model to Design model in phase-II.
- Updated sequence diagrams: Involved in instance sequence diagrams in phase-II
- Pre-processor Python module changes: Updated the preprocessor python module for faster and efficient scraping.
- Activity Diagrams: Involved in Activity Diagrams in phase-II
- State-Transition diagrams: Involved in State-Transition Diagrams in phase-II

SUB-TEAM III:

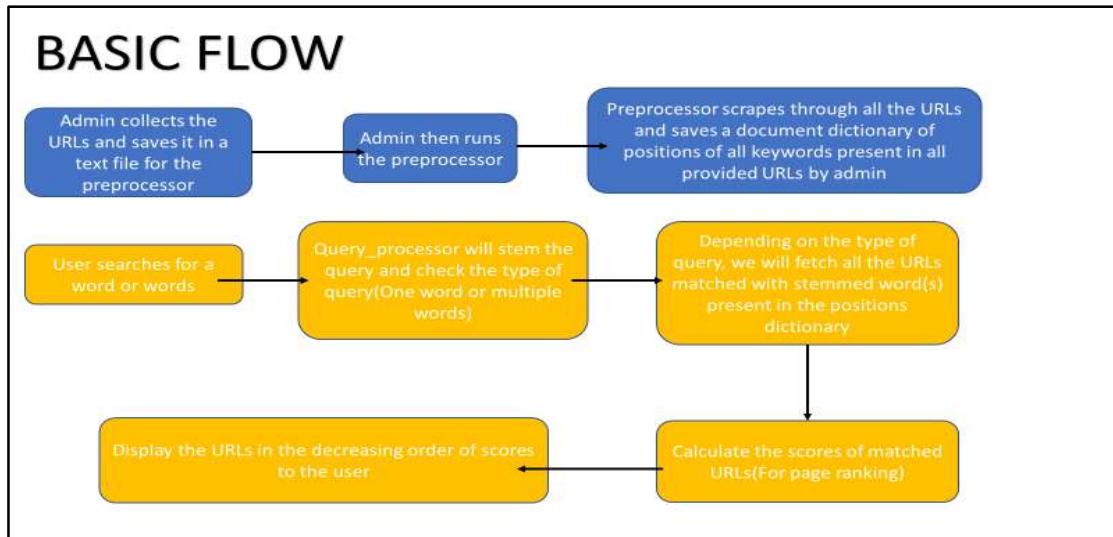
- 1.Sai Pranav Reddy Donthidi
- 2.Apuroop Paravada
- 3.Yogesh Bala
- 4.Srinath Reddy Mavillapally
- 5.Kiran Devaraj Raj

Major Responsibilities of Sub-Team III:

- Basic UML diagrams in Phase-I: Involved in basic UML diagrams like Class Diagram and Sequence diagram along with Basic Flow
- Updated usecase diagrams: Updated usecase diagrams in phase-II
- Pre-processor Python module changes: Updated the preprocessor python module for faster and efficient scraping.
- Updated class-diagram: Involved in updating the class diagram from Domain model to Design model in phase-II.
- Pre-processor python module changes: Updated the preprocessor python module for faster and efficient scraping.
- PPT preparation: Involved in Both the PPT-presentations
- Routes setting in Flask: Involved in setting up the routes for successful rendering of all the html templates.

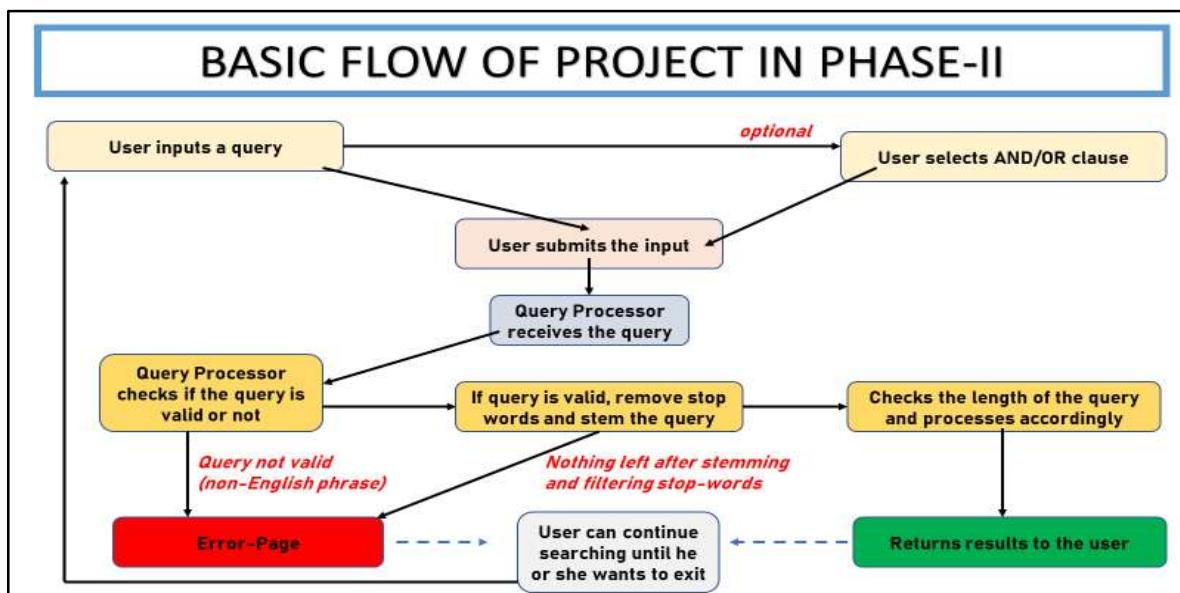
BASIC FLOW IN PHASE-I

Following diagram explains the basic flow of Smart-Search Engine in Phase-I of the project:



BASIC FLOW IN PHASE-II

Following diagram explains the basic flow of Smart-Search Engine in Phase-II of the project:



USECASE TEMPLATES

Following are the important usecases in Smart-Search Engine and have been discussed in detailed using the usecase templates.

| USECASE TEMPLATE-1 | |
|--------------------|--|
| Usecase | inputs the query |
| Description | User will give input in search box |
| Preconditions | Smart Search Engine Application is running |
| Postconditions | User will be displayed with autosuggestions |
| Actors Involved | User |
| Assumptions | Application has been locally hosted |
| Steps | <ol style="list-style-type: none">1.User will give a query input2.Auto Suggestion feature suggests the input query3.User can either select a suggestion or manually type the whole query |
| Issues | User can search only for English words preferably related to Movies. (Since our search engine's specific domain is Movie URLs) |

| USECASE TEMPLATE-2 | |
|--------------------|---|
| Usecase | performs search |
| Description | User clicks enter or clicks on the submit button to perform search. |
| Preconditions | <ol style="list-style-type: none">1.User has preferably given an input(English words)2.User has selected AND/OR clause(optional) |
| Postconditions | User will be displayed with results or error-404 page based on the query |

| | |
|-----------------|---|
| Actors Involved | User, Query Processor |
| Assumptions | 1.User has given a set of English word(s) related to our movie database. 2.Locally hosted. |
| Steps | 1.User will click on submit button or press enter after query search 2.Query Processor will process the query. 3.Query Processor will return the results. 4.User can view the results through the User Interface, i.e., browser. |
| Variations | User can specifically select AND/OR clause before searching multi-phrase query |
| Issues | User can search only for English words preferably related to Movies. (Since our search engine's specific domain is Movie URLs) |

| USECASE TEMPLATE-3 | |
|--------------------|--|
| Usecase | receive the query |
| Description | The Query Processor processes the input given by user and returns search results taking into consideration AND/OR clause. |
| Actors Involved | User, Query Processor, FileSystem. |
| Preconditions | 1.User has preferably given an input (English words) 2.User has selected AND/OR clause(optional) 3.Preprocessor has been run at least once and the database is being maintained in FileSystem. |
| Postconditions | 1. Query processor renders either error-404 page or results page based on query 2.User will be displayed with results or error-404 page based on the query |

| | |
|-------------|---|
| Assumptions | <ol style="list-style-type: none"> 1.Preprocessor has already processed the documents and created a database for query processor. 2.User has given English words. 3.Project has been hosted locally. |
| Steps | <ol style="list-style-type: none"> 1.Receives the query from user. 2.After removal of stop words and stemming, it decides the type of query. 3.Depending on type of query, it performs either one_word_query operation or multi_phrase_query operation. 4.Returns results after successful processing. 5.User views the results through browser. |
| Variations | If user would have given AND/OR clause then query processor will fetch results based on that specifically in multi phrase query operation only. |
| Issues | <ol style="list-style-type: none"> 1.User can search only for English words preferably related to Movies. (Since our search engine's specific domain is Movie URLs) 2.Some movie URL's (wiki links) might have expired. |
| Notes | To maintain this database, admin will/can preprocess regularly |

| USECASE TEMPLATE-4 | |
|--------------------|---|
| Usecase | gets output |
| Description | User gets to view search results on browser. |
| Actors Involved | User, Query Processor, FileSystem |
| Preconditions | Search engine is running on the local machine. |
| Postconditions | Able to view search results on browser. Able to continue searching until user wants to exit. Able to use the pagination functionality |

| | |
|-------------|---|
| Assumptions | Application has been locally hosted |
| Steps | <ol style="list-style-type: none"> 1. User gives search query. 2. If results exists then browser will display the search results. 3. If no results found then error-404 page is displayed. |

| USECASE TEMPLATE-5 | |
|--------------------|---|
| Usecase | run preprocessor |
| Description | Admin will run the preprocessor to create a database. |
| Actors Involved | Admin, Preprocessor, FileSystem. |
| Preconditions | Admin should collect relevant URLs. |
| Postconditions | Resulting Database stored to FileSystem. |
| Steps | <ol style="list-style-type: none"> 1. Admin collects URLs related to movies. 2. Admin will store this in FileSystem. 3. On running the preprocessor, the system will scrape all the URLs and creates a database for the Query Processor to later use. 4. Database is saved to FileSystem. |
| Issues | Admin can only store URLs containing English Language. Admin can only collect relevant Wikipedia URLs as our scraper can scrape only Wikipedia URLs efficiently. |

UML DIAGRAMS

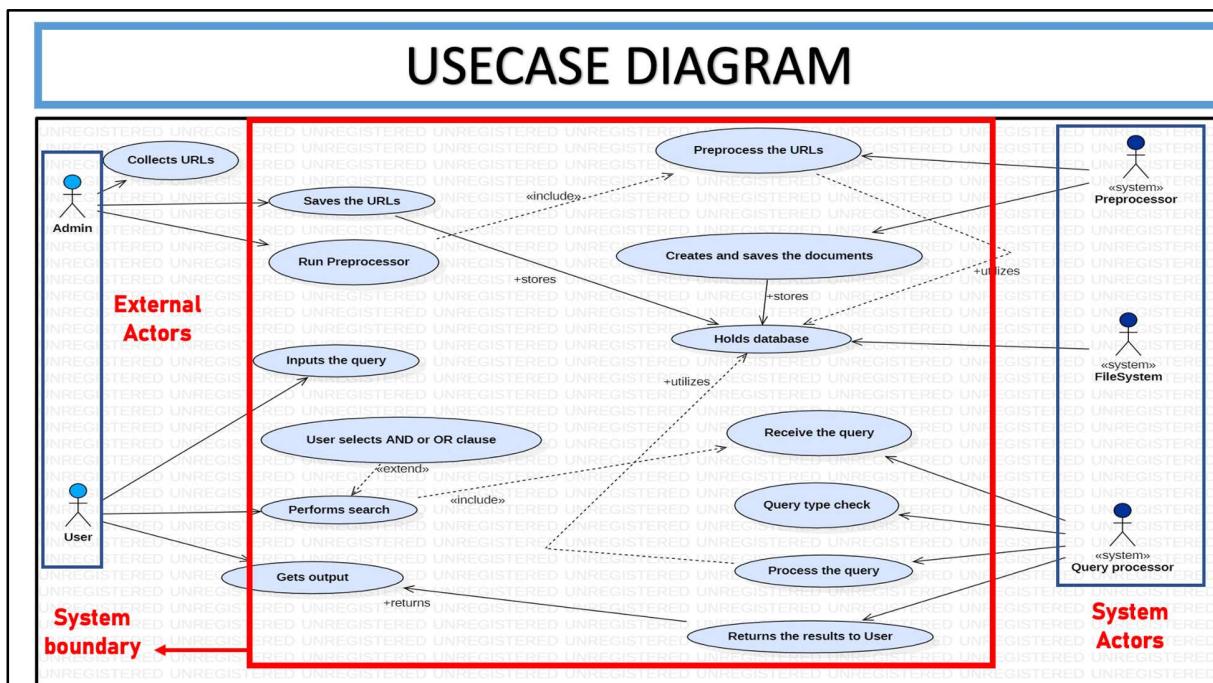
In phase-1 of the project, following UML diagrams were drawn:

- **Class Diagram:** A basic class diagram to understand the main and primary external or system actors was drawn.
- **Usecase Diagram:** A usecase diagram which covered only the main usecases in the proposed project was drawn.
- **Sequence diagram:** Two sequence diagrams were drawn where the basic flow of events between admin and pre-processor were shown in one of the sequence diagrams and in the second sequence diagram (main sequence diagram), the basic flow of events between user and the SearchEngineModule (system actor) were shown.

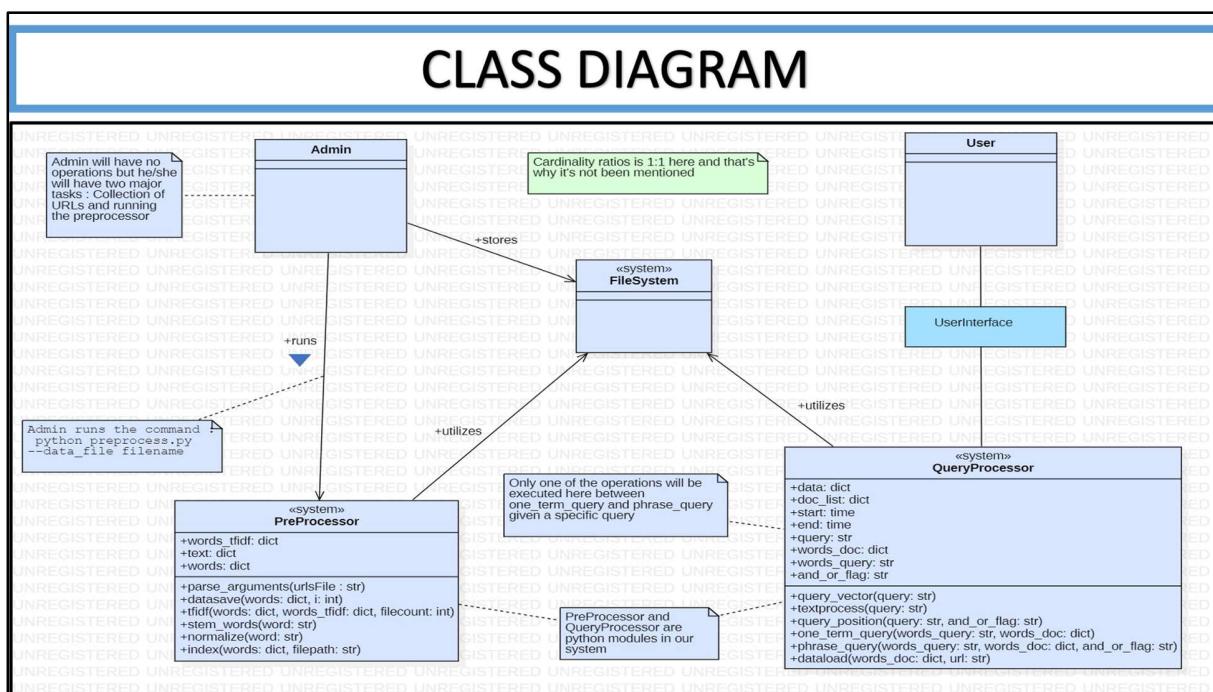
In phase-2 of the project, following UML diagrams were drawn or updated:

- **Class Diagram:** An updated class diagram which explains the relationships between different classes more clearly drawn/presented in the phase-two of the project.
- **Usecase Diagram:** A detailed usecase diagram which covers all the actors and their specific usecases drawn/presented in the phase-two of the project.
- **Sequence diagram:** Sequence diagrams updated accordingly and additional sequence diagrams drawn/presented in the phase-2 of the project to help understand the flow more clearly.
- **State-transition diagrams:** State-transition diagrams describe all of the states that an object can have, the events under which an object changes state (transitions), the conditions that must be fulfilled before the transition will occur (guards), and the activities undertaken during the life of an object (actions). State-transition diagrams are very useful for describing the behavior of individual objects over the full set of use cases that affect those objects. In the phase-2 of the project, a state-transition diagram is drawn/presented which explains the dynamic understanding of the flow of the Smart Search Engine.
- **Activity diagrams:** Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc. Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. In the phase-2 of the project, an activity diagram is drawn/presented.

USECASE DIAGRAM



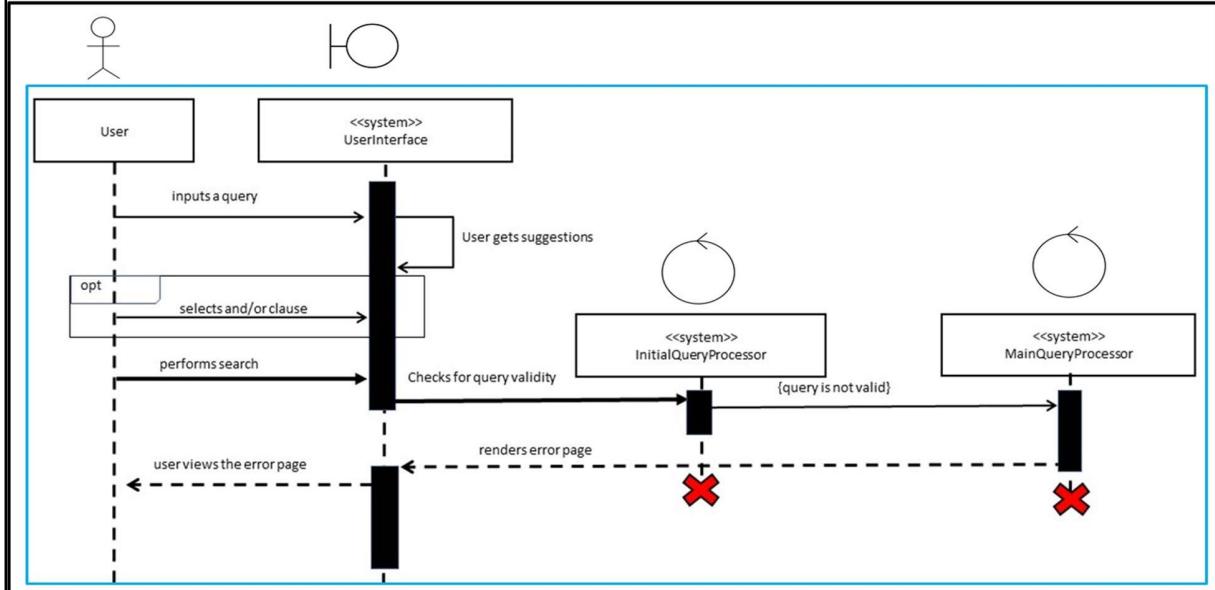
CLASS DIAGRAM



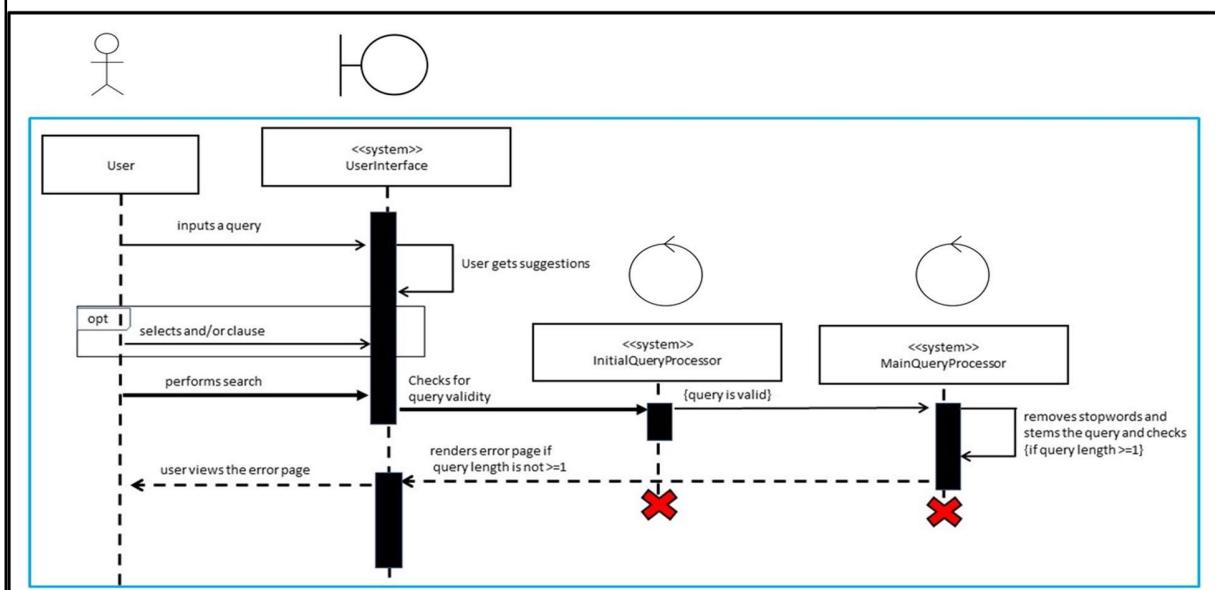
SEQUENCE DIAGRAMS

Following sequence diagrams show multiple scenarios of Smart Search Engine.

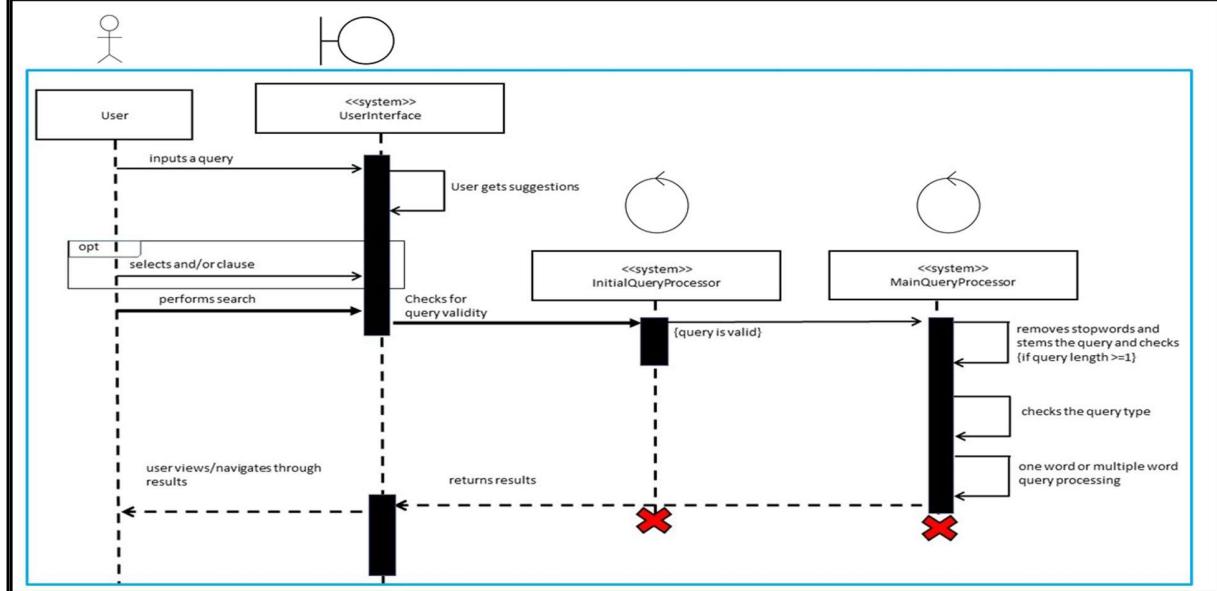
SEQUENCE DIAGRAM-1(INVALID INPUT)



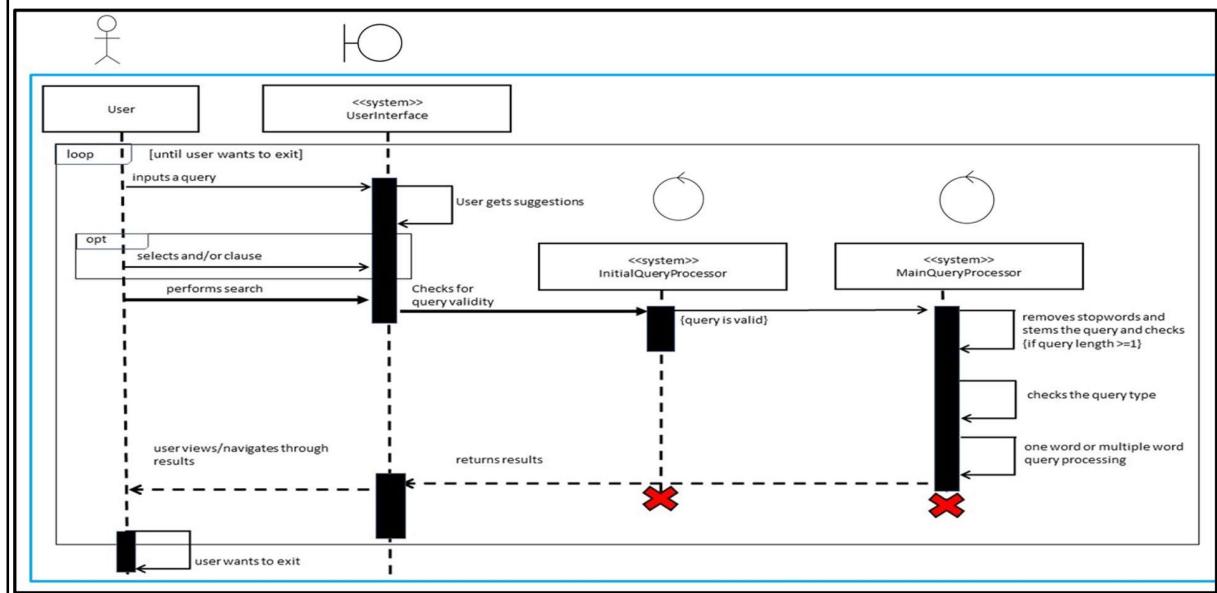
SEQUENCE DIAGRAM-2(IRRELEVANT INPUT)



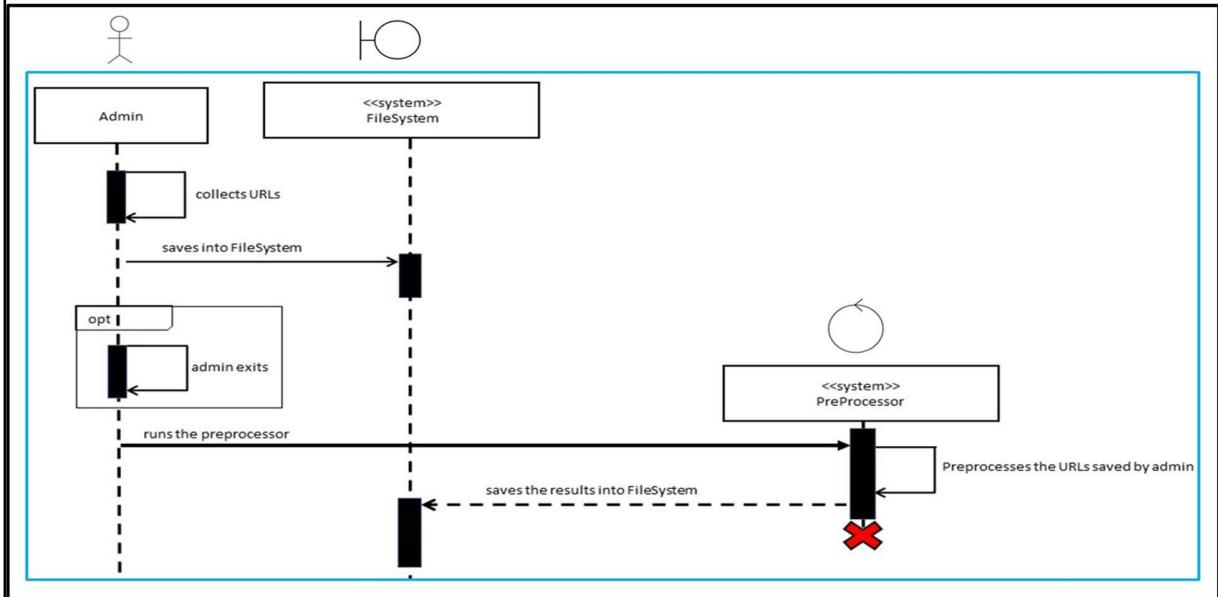
SEQUENCE DIAGRAM-3(SUCCESSFUL SEARCH)



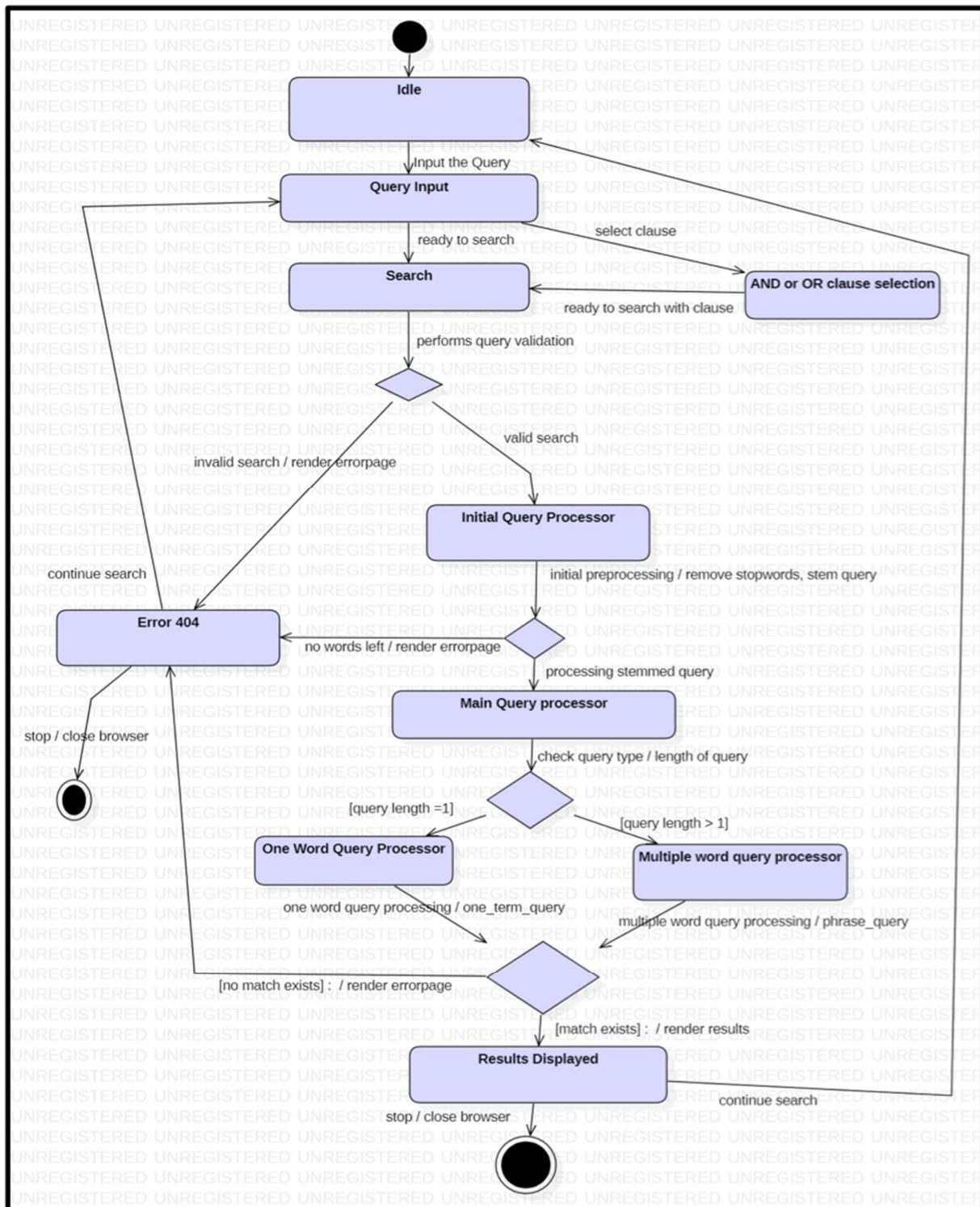
SEQUENCE DIAGRAM-4(LOOP FRAME ADDITION)



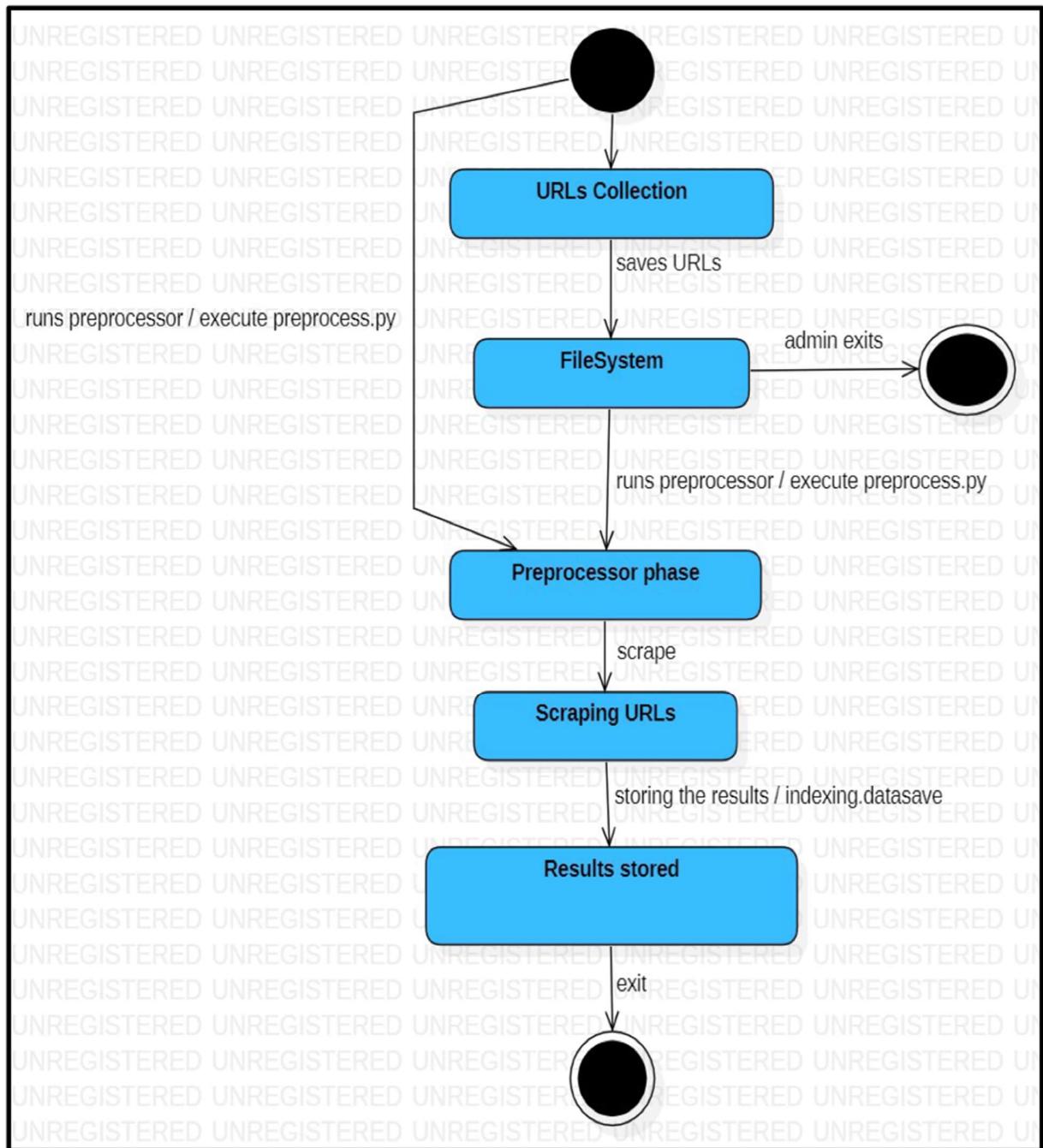
SEQUENCE DIAGRAM-5(ADMIN & PREPROCESSOR)



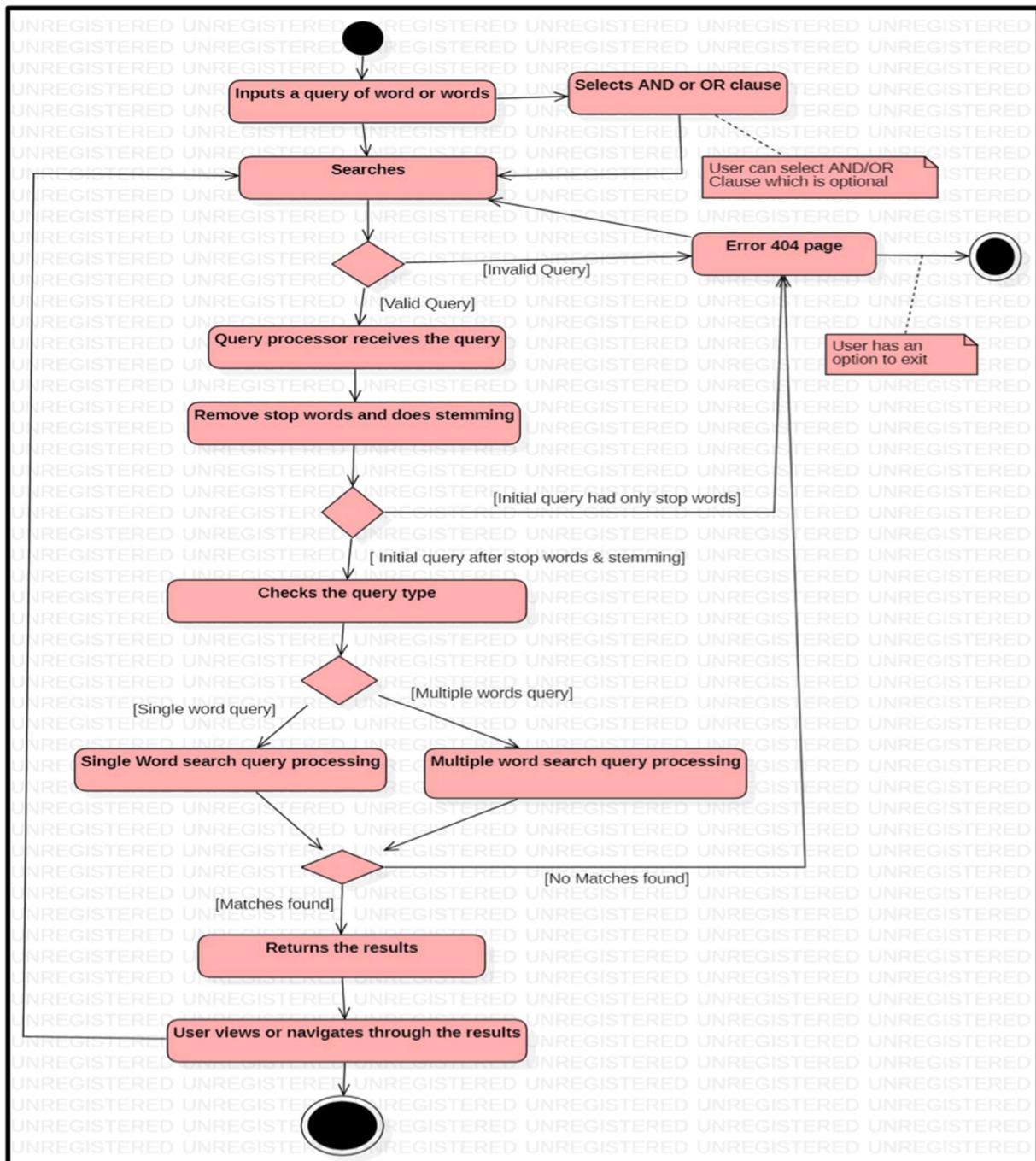
STATE TRANSITION DIAGRAM-1



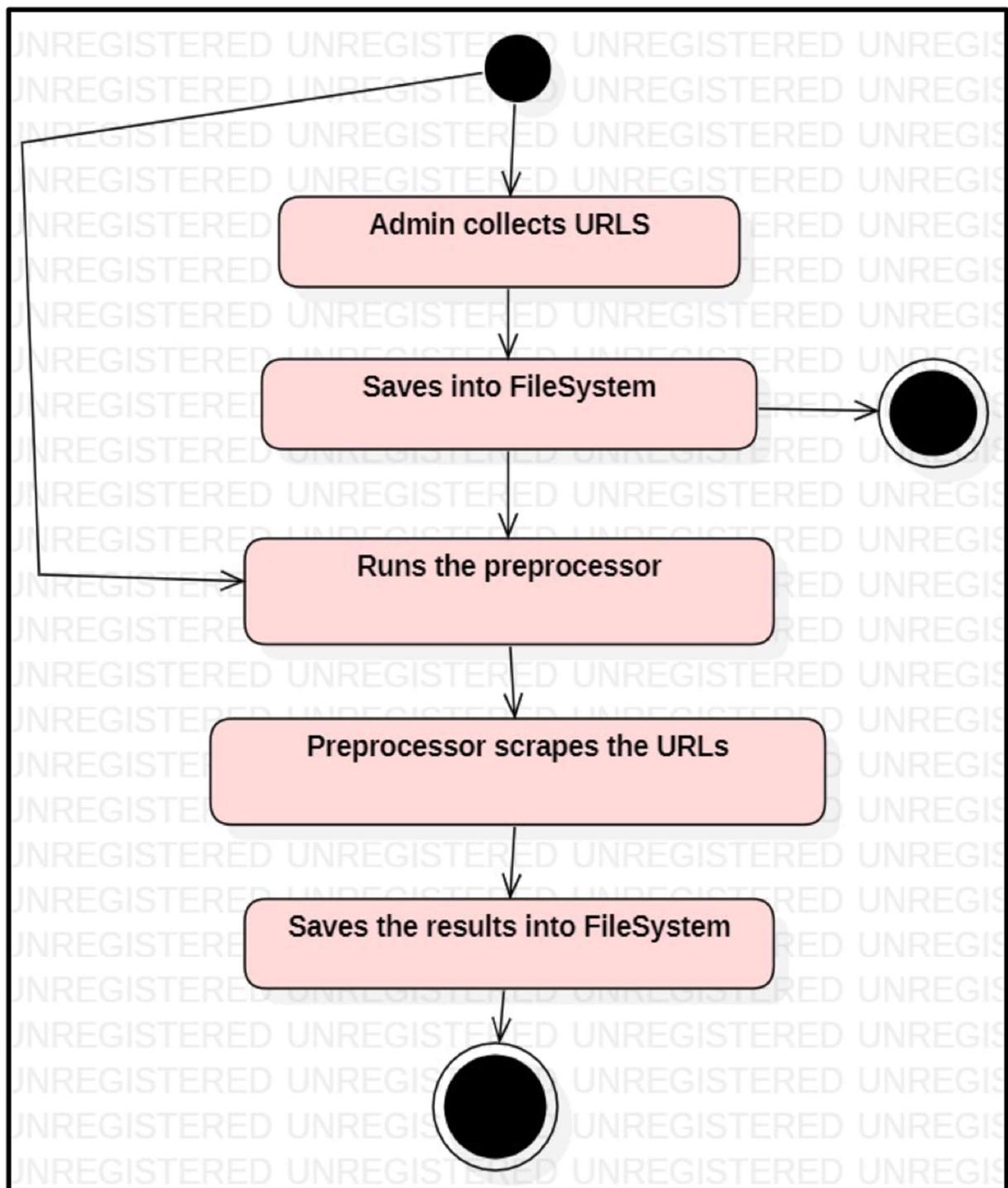
STATE TRANSITION DIAGRAM-2



ACTIVITY DIAGRAM-1



ACTIVITY DIAGRAM-2



TEST CASES

Following are few of the major test cases in this project:

| Test Case-ID | Pre-condition | Input | Output | Post-condition |
|--------------|--|--|--|--|
| TCID1 | System has been hosted and the application is running on the local machine | User starts giving an input query('Dr') | User should be suggested with a few queries related to Movies starting with 'Dr'. | User can select the query from the autosuggestions or can manually type the whole query and the value in the input box will be stored there itself for the user to search. |
| TCID2 | System has been hosted and the application is running on local machine | User starts giving an input query which is/are non-English words(á) | User won't be suggested with any query as it is a non-English input. | User can select the query from the autosuggestions or can manually type the whole query and the value in the input box will be stored there itself for the user to search. |
| TCID3 | 1. System has been hosted and the application is running on the local machine. 2. Pre-processor has been run at least once and the database is being maintained | User clicks on submit after querying an English input('action' word) | User is shown few results relevant to the query. User can use the pagination functionality. User can see the number of results and time taken to process | 1. User should be presented with a few Wikipedia URLs along with two-line description in decreasing order of their relevancy(Page-ranks) or should be shown an |

| | | | | |
|-------|--|---|--|--|
| | | | <p>those results. User has an option to continue search.</p> | <p>error 404-page if it's an invalid search or if no matches are found for the query in the database. User should be able to continue searching until he or she wants to exit.</p> <p><i><u>Below conditions are not applicable if error-404 page is shown :</u></i></p> <ol style="list-style-type: none"> 2. User can navigate through results using Pagination functionality(page numbers) 3. User can click on links and will be redirected to that link in a new tab. 4. User will also be displayed with the number of results and time taken to display those results. |
| TCID4 | <p>1. System has been hosted and the application is running on the local machine.</p> <p>2. Pre-processor has been run at least once and the database is</p> | User clicks on submit after querying a non-English input(á) | <p>User is displayed an error-404 page.</p> <p>User has an option to continue searching.</p> | <p>1. User should be presented with a few Wikipedia URLs along with two-line description in decreasing order of their relevancy(Page-</p> |

| | | | | |
|-------|---|--|---|--|
| | being maintained | | | <p>ranks) or should be shown an error 404-page if it's an invalid search or if no matches are found for the query in the database. User should be able to continue searching until he or she wants to exit.</p> <p><u><i>Below conditions are not applicable if error-404 page is shown :</i></u></p> <p>2. User can navigate through results using Pagination functionality(page numbers)</p> <p>3. User can click on links and will be redirected to that link in a new tab.</p> <p>4. User will also be displayed with the number of results and time taken to display those results.</p> |
| TCID5 | <p>1. System has been hosted and the application is running on the local machine.</p> <p>2. Pre-processor has been run at</p> | User selects AND clause and clicks on submit to search for a 'horror comedy' | User is displayed with Movies' links that come under both Horror and comedy. User can use the | <p>1. User should be presented with a few Wikipedia URLs along with two-line description in decreasing order</p> |

| | | | | |
|-------|--|---|---|---|
| | least once and the database is being maintained | | <p>pagination functionality.</p> <p>User can see the number of results and time taken to process those results.</p> <p>User has an option to continue search.</p> | <p>of their relevancy(Page-ranks) or should be shown an error 404-page if it's an invalid search or if no matches are found for the query in the database. User should be able to continue searching until he or she wants to exit.</p> <p><i><u>Below conditions are not applicable if error-404 page is shown :</u></i></p> <ol style="list-style-type: none"> 2. User can navigate through results using Pagination functionality(page numbers) 3. User can click on links and will be redirected to that link in a new tab. 4. User will also be displayed with the number of results and time taken to display those results. |
| TCID6 | 1. System has been hosted and the application is running on the local machine. | User selects OR clause and clicks on submit to search for a 'horror comedy' | User is displayed with Movies' links that come under either Horror or | 1. User should be presented with a few Wikipedia URLs along with two-line |

| | | | | |
|--|---|--|---|--|
| | <p>2. Pre-processor has been run at least once and the database is being maintained</p> | | <p>comedy or both Horror and comedy genres. User can use the pagination functionality. User can see the number of results and time taken to process those results. User has an option to continue search.</p> | <p>description in decreasing order of their relevancy(Page-ranks) or should be shown an error 404-page if it's an invalid search or if no matches are found for the query in the database. User should be able to continue searching until he or she wants to exit.</p> <p><i><u>Below conditions are not applicable if error-404 page is shown :</u></i></p> <p>2. User can navigate through results using Pagination functionality(page numbers)</p> <p>3. User can click on links and will be redirected to that link in a new tab.</p> <p>4. User will also be displayed with the number of results and time taken to display those results.</p> |
|--|---|--|---|--|

TEST-CASE RELATED SCREENSHOTS:

TCID1:

| Input | Output |
|---|---|
| User starts giving an input query('Dr') | User should be suggested with a few queries related to Movies starting with 'Dr'. |



TCID2:

| Input | Output |
|---|--|
| User starts giving an input query which is/are non-English words(á) | User won't be suggested with any query as it is a non-English input. |



TCID3:

| Input | Output |
|---|--|
| User clicks on submit after querying an English input ('action' word) | User is shown few results relevant to the query. User can use the pagination functionality. User can see the number of results and time taken to process those results. User has an option to continue search. |

The screenshot shows a web browser window titled "Smart Engine Search Page" with the URL "127.0.0.1:5000/search". The search bar contains the word "Action". Below the search bar, there is a button labeled "Choose AND/OR" and a "Search" button. The search results section displays the following information:

About 229 results (0.49346160888671875 seconds)

[The Lord of the Rings](https://en.wikipedia.org/wiki/The_Lord_of_the_Rings_(1978_film))
Early in the Second Age of Middle-earth, elven smiths forge nine Rings of Power for mortal men, seven for the Dwarf-Lords, and three for the Elf-Kings. Soon after, the Dark Lord Sauron makes the One Ring, and uses it to attempt to conquer Middle-earth.

[Predator](https://en.wikipedia.org/wiki/Predator_(film))
A spacecraft flies near Earth and releases a bright object which enters the atmosphere. In the Val Verde jungle, US Army Major Alan "Dutch" Schaefer and his Special Forces team — medic Mac Elliot, tracker Billy Sole, gunner Blain Cooper, explosives expert Jorge "Poncho" Ramirez, and radio operator Rick Hawkins — are tasked by the CIA with rescuing an official held hostage by insurgents.

[John Carter](https://en.wikipedia.org/wiki/John_Carter_(film))

The screenshot shows a web browser window titled "Smart Engine Search Page" with the URL "127.0.0.1:5000/search". The search bar contains the word "Jin Roh". Below the search bar, there is a button labeled "Choose AND/OR" and a "Search" button. The search results section displays the following information:

[Jin Roh](https://en.wikipedia.org/wiki/Jin_Roh)
The story is set in a parallel 1950s Japan, in which Germany has conquered Japan. It focuses on Kazuki Fuse, a member of the elite Kerberos Panzer Cops, a metropolitan antiterror unit.

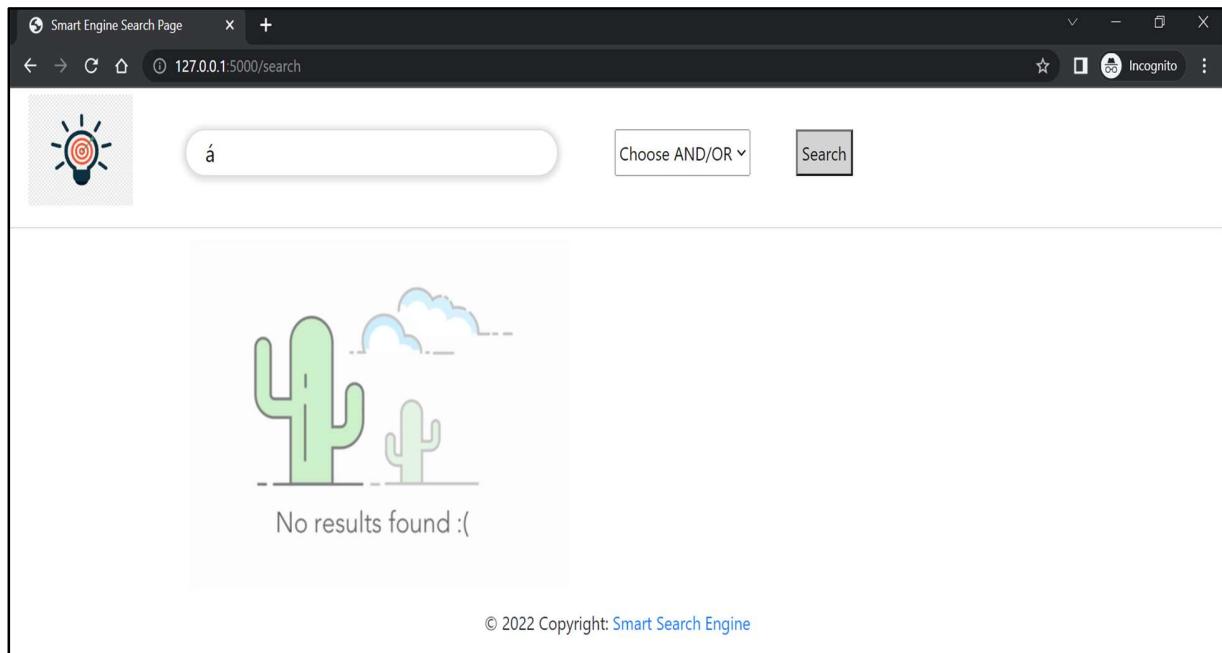
[Hana yori Dango Final](https://en.wikipedia.org/wiki/Hana_yori_Dango_Final)
Tsukasa Dōmyōji (Jun Matsumoto) and Tsukushi Makino (Mao Inoue) are driving through Nevada. Weeks prior, Tsukasa announced to the world that he was engaged to Tsukushi.

Pagination controls at the bottom: « < 1 2 3 4 5 ... 22 23 > »

© 2022 Copyright: Smart Search Engine

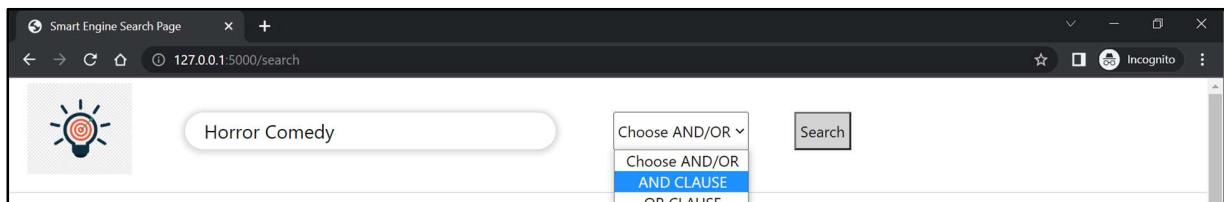
TCID4:

| Input | Output |
|---|---|
| User clicks on submit after querying a non-English input(á) | User is displayed an error-404 page. User has an option to continue searching. |



TCID5:

| Input | Output |
|--|--|
| User selects AND clause and clicks on submit to search for a 'horror comedy' | User is displayed with Movies' links that come under both Horror and comedy. User can use the pagination functionality. User can see the number of results and time taken to process those results. User has an option to continue search. |

A screenshot of the same search interface after performing the search. The search bar still shows "Horror Comedy". The dropdown menu now shows "AND CLAUSE". Below the search bar, the text "About 14 results (0.06481432914733887 seconds)" is displayed. Two results are listed:

https://en.wikipedia.org/wiki/Killer%27s_Moon
Killer's Moon
A coach full of schoolgirls breaks down in the Lake District, forcing the girls to take shelter for the night in a remote hotel. Meanwhile, strange and macabre things are happening to the locals (and their pets) and it is revealed that four escaped mental patients- Mr.

https://en.wikipedia.org/wiki/Piranha_II:_The_Spawning
Piranha II: The Spawning
Off the coast of a Caribbean island, a young couple flee a hotel to have sex in the sea. But they swim into a sunken wreck which is also a piranha lair and they are both killed and eaten by the unseen piranha.

A screenshot of the search results page for the term "Soapdish". The URL "https://en.wikipedia.org/wiki/Soapdish" is shown at the top. The search term "Soapdish" is bolded. The page content describes the plot of the TV show "The Sun Also Sets", mentioning Celeste Talbert and Montana Moorehead. At the bottom of the page, there is a navigation bar with page numbers "« 1 2 »".

<https://en.wikipedia.org/wiki/Soapdish>
Soapdish
Celeste Talbert, the long-time star of the embattled daytime drama *The Sun Also Sets*, is targeted by her ambitious co-star Montana Moorehead; Montana connives to supplant Celeste as the show's star by promising sexual favors to its producer, David Seton Barnes. To make the audience hate Celeste's character, Montana and David come up with a last-minute plot change in which she will accidentally kill a young, destitute deaf-mute, played by the newly-cast Lori Craven.

« 1 2 »

© 2022 Copyright: Smart Search Engine

TCID6:

| Input | Output |
|---|--|
| User selects OR clause and clicks on submit to search for a ‘horror comedy’ | User is displayed with Movies’ links that come under either Horror or comedy or both Horror and comedy genres. User can use the pagination functionality. User can see the number of results and time taken to process those results. User has an option to continue search. |

The screenshot shows a web browser window for 'Smart Engine Search Page' at 127.0.0.1:5000/search. The search bar contains 'Horror Comedy'. A dropdown menu next to it says 'Choose AND/OR' with 'OR CLAUSE' selected. A 'Search Movies' button is also visible. Below the search bar, there's a small icon of a lightbulb. The main content area displays search results:

About 382 results (0.48598456382751465 seconds)

[https://en.wikipedia.org/wiki/Cat_People_\(1942_film\)](https://en.wikipedia.org/wiki/Cat_People_(1942_film))
Cat People
At the Central Park Zoo in New York City, Serbian-born fashion designer Irena Dubrovna makes sketches of a black panther. She catches the attention of marine engineer Oliver Reed, who strikes up a conversation.

[https://en.wikipedia.org/wiki/Halloween_\(1978_film\)](https://en.wikipedia.org/wiki/Halloween_(1978_film))
Halloween
In 1963, on Halloween night in Haddonfield, Illinois, Michael Myers, dressed in a clown costume and mask, inexplicably stabs his older sister Judith to death with a kitchen knife in their home. He is subsequently hospitalized at Warren County's Smith's Grove Sanitarium.

https://en.wikipedia.org/wiki/Yakshiyum_Njanum
Lankeshwari
The movie engages with social issues through the horror and fantasy genres. Spadikam George is the lead actor of this film.

[https://en.wikipedia.org/wiki/Spellcaster_\(film\)](https://en.wikipedia.org/wiki/Spellcaster_(film))
Spellcaster
Winning a trip to Italy through a contest, Jackie (O'Grady) and her brother Tom (Pruett) join other winners Myrna (Demson), Yvette (Lind), Terri (Ulrich), Harlan (Zorek) and Tony (Modugno). They settle into the rooms of an old castle, together with Cassandra (Bailey), a rock star playing along in the contest.

« 1 2 3 4 5 ... 38 39 »

© 2022 Copyright: Smart Search Engine

USER-MANUAL

Below is a reference to the Read-me file which has also been attached to the project folder. This Read-me file explains in detail about the steps we need to follow to setup our project and make it run on a local machine.

```
# Smart Search Engine using Python, Html, CSS, JS and Flask.  
Search Engine -> Developed using Flask application to integrate python backend with html, js frontend.  
## Modules Used  
NLTK      - for NLP procedures  
Beautiful Soup - for scraping URL
```

Installations

Way 1 to run the project: Install virtual environment and use it to run the flask application.

Install with pip:

```
...  
$ python3 -m ensurepip --upgrade  
$ python3 -m pip install --user virtualenv  
$ python3 -m venv env  
$ source env/bin/activate  
$ pip install -r requirements.txt  
$ python routes.py  
...  
...
```

Way 2 to run the project: Use pre used virtual environment to run the flask application.

Install with pip:

```
...  
$ source env/bin/activate  
$ pip install -r requirements.txt  
$ python routes.py  
...  
...
```

Flask Application Structure

```
...  
. |--- env  
|--- 1000Movie.csv
```

```
├── README.md
├── data
│   ├── path.json
│   ├── positions.json
│   ├── summaries.json
│   ├── text_doc.json
│   └── tfidf_index.json
├── indexing.py
├── package.json
├── preprocess.py
├── query_process.py
├── requirements.txt
├── routes.py
└── static
    ├── css
    │   ├── index.css
    │   ├── main.css
    │   └── search.css
    ├── images
    │   ├── 404_error.webp
    │   ├── logo.png
    │   └── search.png
    └── js
        ├── script.js
        └── suggestions.js
├── summarisation.py
└── templates
    ├── error404.html
    ├── index.html
    └── search.html
└── text_extract.py
```

```
```
```

```
Flask Configuration
Example
```
app = Flask(__name__)
```

```
app.config['DEBUG'] = True
```

```
'''
```

```
## Run Flask
```

```
#### Run flask for develop
```

```
'''
```

```
$ python routes.py
```

```
'''
```

In flask, Default port is `5000`

Main Index Page: `http://127.0.0.1:5000/`

Result Page: `http://127.0.0.1:5000/search`

```
## Reference
```

Official Website

- [Flask](<http://flask.pocoo.org/>)
- [Flask Extension](<http://flask.pocoo.org/extensions/>)
- [Documentation](<https://flask.palletsprojects.com/>)
- [Changes](<https://flask.palletsprojects.com/changes/>)
- [PyPI Releases](<https://pypi.org/project/Flask/>)
- [Source Code](<https://github.com/pallets/flask/>)
- [Issue Tracker](<https://github.com/pallets/flask/issues/>)
- [Website](<https://palletsprojects.com/p/flask/>)
- [Twitter](<https://twitter.com/PalletsTeam>)
- [Chat](<https://discord.gg/pallets>)

Tutorial

-[Flask Overview] (<https://www.slideshare.net/maxcnunes1/flask-python-16299282>)

- [In Flask we trust](<http://igordavydenko.com/talks/ua-pycon-2012.pdf>)

- [Wiki Page](<https://github.com/tsungtwu/flask-example/wiki>)

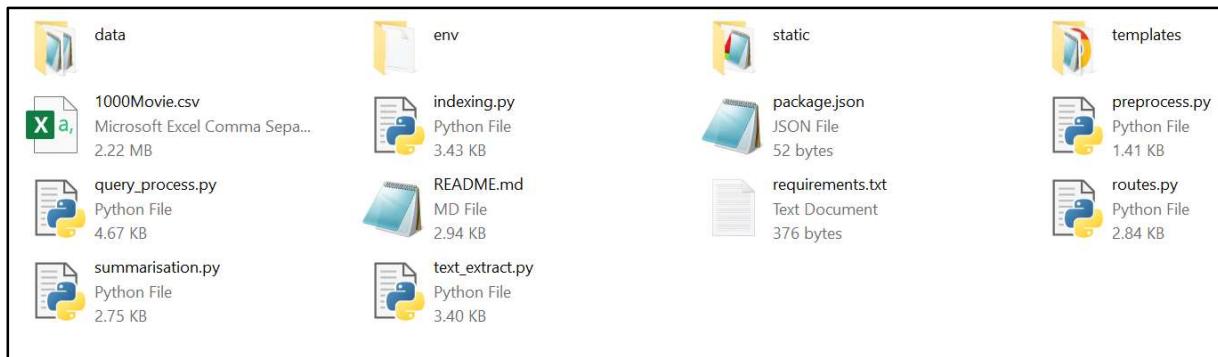
```
## Changelog
```

- Version 2.1 : Search Engine with advancements in User Interface.
- Version 2.0 : Search Engine with advanced search functionality with AND/OR clause included for multi phrase input query.

- Version 1.0 : Search Engine with basic functionality to search and get movie url's related to the searched input query.

Let's try to understand the above steps in more detailed manner via screenshots:

Below is our project folder where we can see all the required files to run the project.



We need to run the following command to activate the virtual environment.

```
PS C:\Users\krish\OneDrive\Desktop\env\Scripts> .\activate  
(env) PS C:\Users\krish\OneDrive\Desktop\env\Scripts>
```

Flask application is run using the following command.

```
(env) PS C:\Users\krish\OneDrive\Desktop\SearchEngine00ADV-Final> python routes.py
```

This indicates that the Smart Search Engine has been locally hosted using Flask application.

```
* Restarting with stat
[nltk_data]  Downloading package punkt to
[nltk_data]    C:\Users\krish\AppData\Roaming\nltk_data...
[nltk_data]      Package punkt is already up-to-date!
[nltk_data]  Downloading package stopwords to
[nltk_data]    C:\Users\krish\AppData\Roaming\nltk_data...
[nltk_data]      Package stopwords is already up-to-date!
[nltk_data]  Downloading package words to
[nltk_data]    C:\Users\krish\AppData\Roaming\nltk_data...
[nltk_data]      Package words is already up-to-date!
* Debugger is active!
* Debugger PIN: 134-844-852
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

PROTOTYPE IMPLEMENTATION:

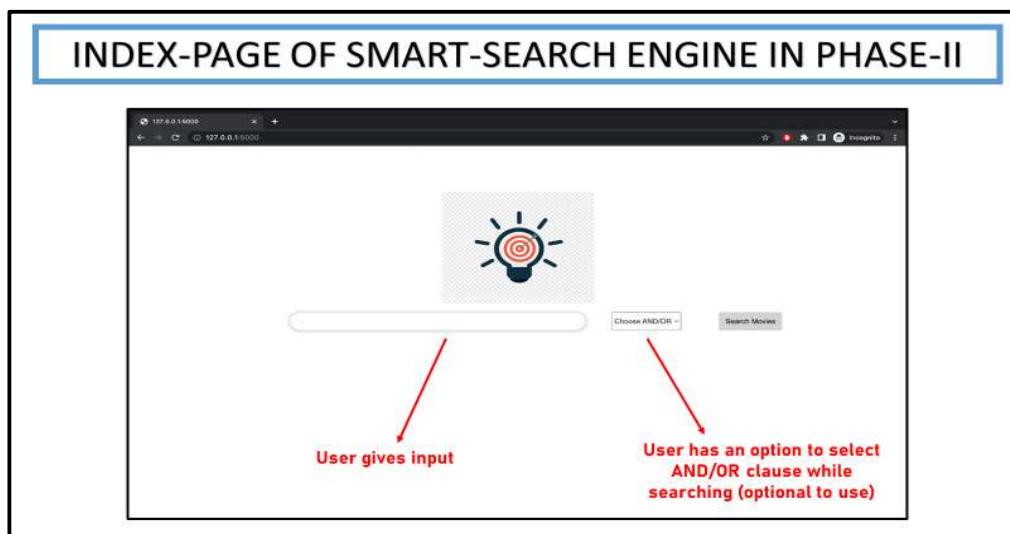
After running the project using the above steps, let's have a look at demo screenshots of major functionalities in our Smart Search Engine.

- 1) Below is a screenshot of the 1000 URLs file which is being pre-processed and creates our database.

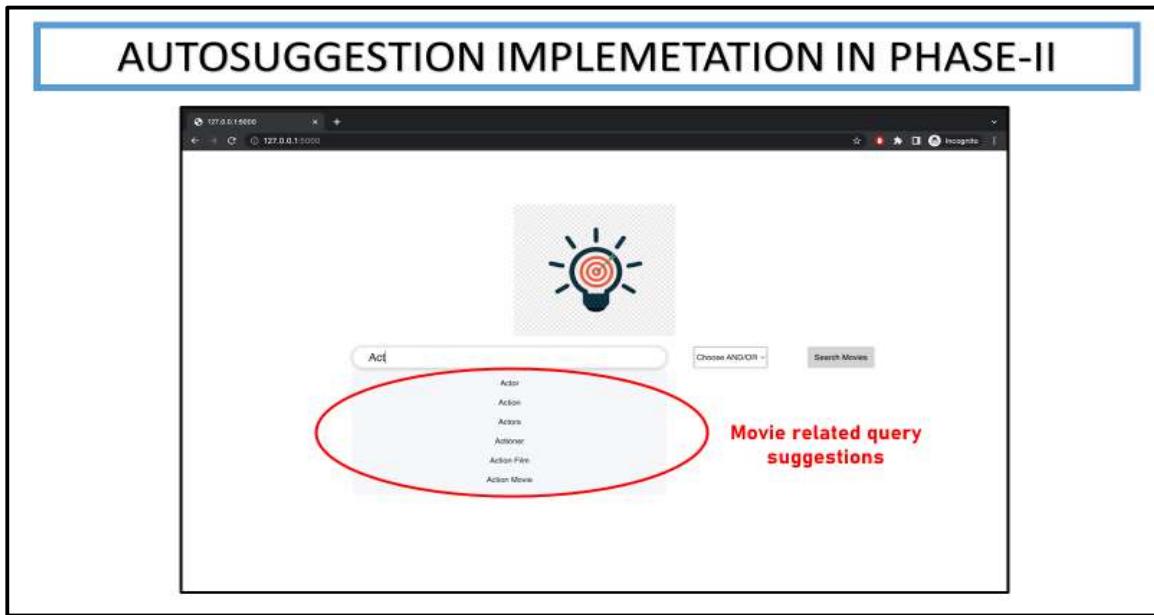
Admin will collect all the relevant URLs for the pre-processor and save it.
Following is a screenshot of how the file looks :

| Genre | Wiki Page | Plot |
|----------------------------|---|---|
| unknown | https://en.wikipedia.org/wiki/Kansas_Saloon_Smashers | A bartender is working at a saloon, set |
| unknown | https://en.wikipedia.org/wiki/Love_by_the_Light_of_the_Moon | The moon, painted with a smiling face |
| unknown | https://en.wikipedia.org/wiki/The_Martyred_Presidents | The film, just over a minute long, is co |
| unknown | https://en.wikipedia.org/wiki/Terrible_Teddy_the_Grizzly_King | Lasting just 61 seconds and consistin |
| unknown | https://en.wikipedia.org/wiki/Jack_and_the_Beanstalk_(1902_film) | The earliest known adaptation of the |
| unknown | https://en.wikipedia.org/wiki/Alice_in_Wonderland_(1903_film) | Alice follows a large white rabbit dow |
| western | https://en.wikipedia.org/wiki/The_Great_Train_Robbery_(1903_film) | She enters a kitchen, in which there is |
| comedy | https://en.wikipedia.org/wiki/The_Suburbanite | The film opens with two bandits break |
| unknown | https://en.wikipedia.org/wiki/The_Little_Train_Robbery | ing into a bank. They are stopped by a |
| unknown | https://en.wikipedia.org/wiki/The_Night_Before_Christmas_(1905_film) | Prep-processor will scraper these URLs |
| short | https://en.wikipedia.org/wiki/Dream_of_a_Rabbit_Fiend_(1906_film) | Scenes are introduced using titles of |
| short action/crime western | https://en.wikipedia.org/wiki/From_Leadville_to_Aspen:_A_Hold-Up_in_the_Rockies | The Rabbit Fiend gorges on Welsh rarebit |
| short film | https://en.wikipedia.org/wiki/Kathleen_Mavourneen_(1906_film) | The film features a train traveling through |
| biographical | https://en.wikipedia.org/wiki/Daniel_Boone_(1907_film) | Irish villager Kathleen is a tenant of Captain Boone's daughter befriends an Indian |
| comedy | https://en.wikipedia.org/wiki/How_Brown_Saw_the_Baseball_Game | Before heading out to a baseball game, Brown |
| comedy | https://en.wikipedia.org/wiki/Laughing_Gas_(film)#1907_film | The plot is that of a black woman going to |
| drama | https://en.wikipedia.org/wiki/The_Adventures_of_Dolly | On a beautiful summer day a father and his |
| drama | https://en.wikipedia.org/wiki/The_Black_Viper | A thug accosts a girl as she leaves her |

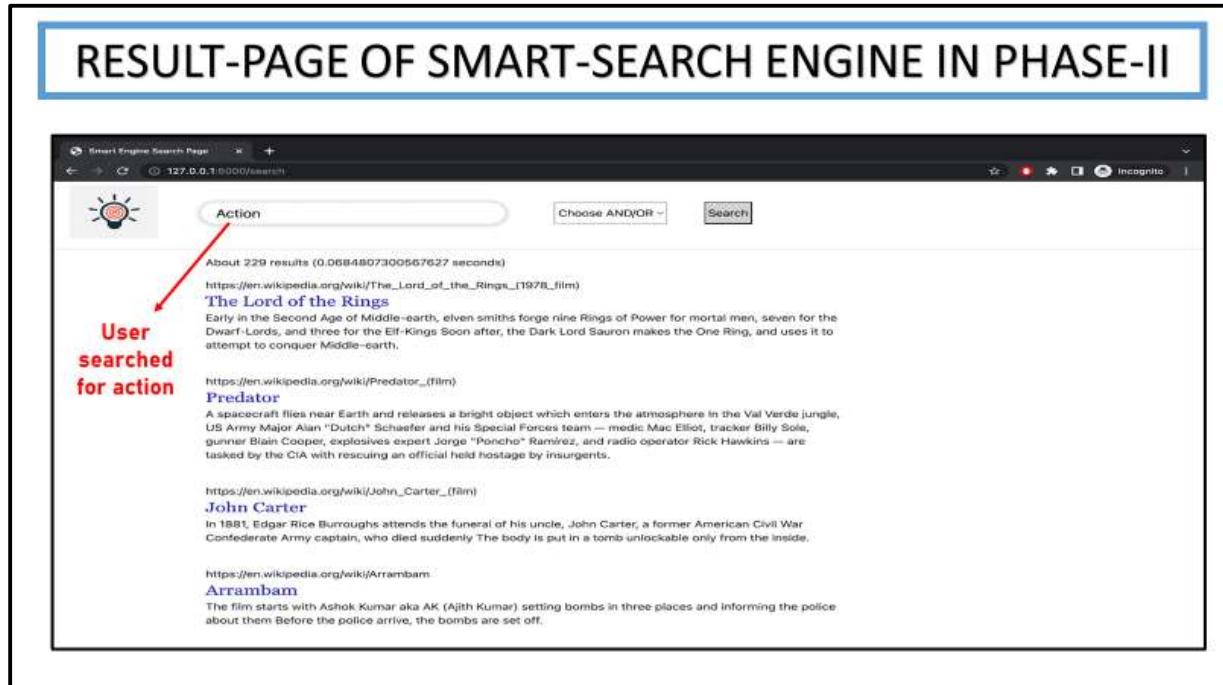
- 2) Below is a screenshot of the index page of the Smart Search Engine. The main index page URL is: **127.0.0.1:5000**



3) Below is a screenshot of the autosuggestion functionality.



4) Below is a screenshot of results after user performs a successful search with the query: 'Action'.



5) Below is a screenshot of AND-Clause functionality.

AND-CLAUSE IMPLEMENTATION IN PHASE-II

AND CLAUSE

About 14 results (0.009011983873459961 seconds)

[Killer's Moon](https://en.wikipedia.org/wiki/Killer%27s_Moon)
A coach full of schoolgirls breaks down in the Lake District, forcing the girls to take shelter for the night in a remote hotel. Meanwhile, strange and macabre things are happening to the locals (and their pets) and it is revealed that four escaped mental patients- Mr.

[Piranha II: The Spawning](https://en.wikipedia.org/wiki/Piranha_II:_The_Spawnning)
Off the coast of a Caribbean island, a young couple flee a hotel to have sex in the sea. But they swim into a sunken wreck which is also a piranha lair and they are both killed and eaten by the unseen piranhas.

[The Texas Chainsaw Massacre 2](https://en.wikipedia.org/wiki/The_Texas_Chainsaw_Massacre_2)
In 1986, thirteen years after the events of the first film, two high school seniors, Buzz (Barry Kironyon) and Rick (Chris Doubleday), race along a desolate stretch of Texas highway, en route to the Texas-OU football game at the Dallas Cowboys Bowl. Heavily intoxicated, they use their car phone to call and harass on-air radio DJ Vanita "Stretch" Brock (Caroline Williams).

[Burke & Hare](https://en.wikipedia.org/wiki/Burke_%26_Hare_(1971_film))
In Edinburgh in the 1820s, surgeon Dr Knox (Harry Andrews) employs graverobbers Burke and Hare (Derren Nesbitt and Olynn Edwards) to supply fresh corpses for his anatomical lectures at the medical academy. When graveyard supplies run low, the industrious pair turn to murder to keep the business going.

Movies that come under both Horror and comedy genres

6) Below is a screenshot of OR-Clause functionality.

OR-CLAUSE IMPLEMENTATION IN PHASE-II

OR CLAUSE

About 382 results (0.112878742380178 seconds)

[Cat People](https://en.wikipedia.org/wiki/Cat_People_(1942_film))
At the Central Park Zoo in New York City, Serbian-born fashion designer Irena Dubrovna makes sketches of a black panther. She catches the attention of marine engineer Oliver Reed, who strikes up a conversation.

[Halloween](https://en.wikipedia.org/wiki/Halloween_(1978_film))
In 1963, on Halloween night in Haddonfield, Illinois, Michael Myers, dressed in a clown costume and mask, inexplicably stalks his older sister Judith to death with a kitchen knife in their home. He is subsequently hospitalized at Warren County's Smith's Grove Sanitarium.

[Killer's Moon](https://en.wikipedia.org/wiki/Killer%27s_Moon)
A coach full of schoolgirls breaks down in the Lake District, forcing the girls to take shelter for the night in a remote hotel. Meanwhile, strange and macabre things are happening to the locals (and their pets) and it is revealed that four escaped mental patients- Mr.

[House of Horrors](https://en.wikipedia.org/wiki/House_of_Horrors)
Struggling sculptor, Marcel De Lange (Martin Kosleck) is depressed about the events going on in his life, and decides to commit suicide. But just as he's about to kill himself, he spots and saves a madman, named "The Creeper" (Rondo Hatton) from drowning.

Movies that come under either of Horror or comedy genres

- 7) Below is a screenshot of Hyper-link functionality and the description being displayed below every URL in the results.

HYPERLINK & DESCRIPTION OF URL IMPLEMENTATION IN PHASE-II

User has an option to navigate to the URL by clicking on it

Two line description

- 8) Below screenshots explain in short about how the Smart Search Engine's page-rank algorithm works.

PAGE-RANKING ALGORITHM IMPLEMENTATION IN PHASE-II

How do we conclude that Lord of Rings is most relevant result for the query Action?????

PAGE-RANKING ALGORITHM IMPLEMENTATION IN PHASE-II

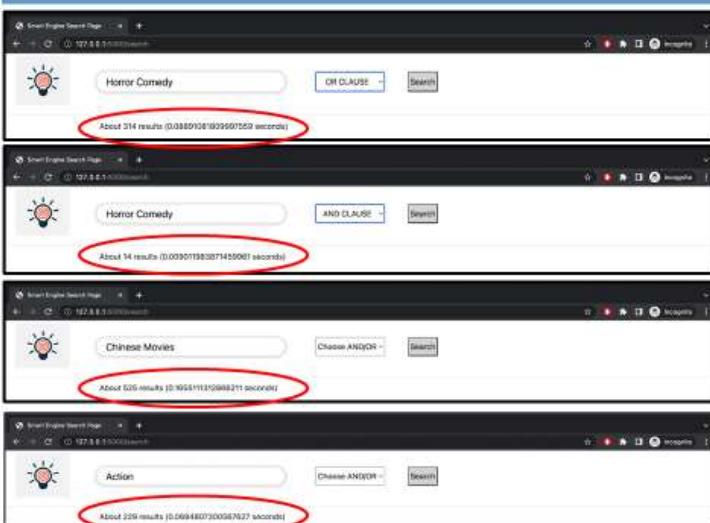
Single Phrase Query
Action
[https://en.wikipedia.org/wiki/The_Lord_of_the_Rings_(1978_film)], [2.7733236186612713, 'The Lord of the Rings', 'Early in the Second Age of Middle-earth, eleven valiant Forges nine Rings of Power for mortal men, seven for the ~~Dark~~ ~~Lord~~, and three for the E~~I~~-Kings! Soon after, the Dark Lord Sauron makes the One Ring, and uses it to attempt to conquer Middle-earth!], [[https://en.wikipedia.org/wiki/Predator_\(film\)](https://en.wikipedia.org/wiki/Predator_(film))], [2.189462788186712, 'Predator', 'A spacecraft flies near Earth and releases a bright object which enters the atmosphere in the Val Verde jungle, US Army Major Alan "Dutch" Schaefer and his Special Forces team – medic Mac El list, tracker Billy Rose, gunner Blain Cooper, explosives expert Jorge "Punch" Ramirez, and radio operator Rick Hawkins – are sent to the jungle with resupplying an off-base camp by insurgents!'], [<https://en.wikipedia.org/wiki/Arakan>], [2.239013584552052, 'Arakan'], 'The film starts with Ashok Kumar as Akh (Akhil Kumar) setting bombs in three places and informing the police about them. Before the police arrive, the bombs are set off!', [<https://en.wikipedia.org/w/index.php?title=Arapachis&oldid=209617605>], 'Arapachis', 'Ranavajin Tyengar alias Amil is an orthodox Brahmin and a strait-forward consumer protection advocate living in Triplicane, Chennai. He expects everyone to follow the law and prosecutes those who violate it!'], ([https://en.wikipedia.org/wiki/The_Killer_\(1980_film\)](https://en.wikipedia.org/wiki/The_Killer_(1980_film))), [2.1141977930088094, 'The Killer', 'Niranjan AH Jong is on his last job for the Hong Kong Triad, but accidentally damages the eyes of a young nightclub singer named Jennie with a muzzed shotgun shot. He is forced to flee Hong Kong and go to India to hide. He is tracked down by the right-hand man of the Triad, who is assisted by his son (Randy).'], ([https://en.wikipedia.org/wiki/Randy_\(2014_film\)](https://en.wikipedia.org/wiki/Randy_(2014_film))), [2.14147930088094, 'Randy'], 'The film opens in a store in Hyderabad in a night with Krishna (Vishnu Manchu) saving Sirisha (Shanti Srivastava) from some eye teasers and accompanying her to the hotel she stays in the journey, they both come to know that they hail from Rayalaseema region of Andhra Pradesh!'], (https://en.wikipedia.org/wiki/Cardfight!!_Vanguard), [11.97348929195983, 'Cardfight!! Vanguard The Movie!'], 'Season 1vNaVtchi Sendou is a timid young boy in his third year of junior high school. The one thing that keeps him going is his trading card Blaste r Blade from Cardfight!! Vanguard, a trading card game that takes place on a different planet called "Cray" and is popular throughout the world!'], (https://en.wikipedia.org/wiki/Jin_Roh), [11.885876149646832, 'Jin Roh'], 'The story is set in a parallel 1958 s Japan, in which Germany has conquered Japan. It focuses on Kasumi Reina, a member of the elite Kerberos Pantom Corps, a "metropolis" of secret agents. She is assigned to infiltrate the Yakuza. Her partner is Kurogane (Yuki Matsui), and Tsukuhiko Makino (Mao Inoue) are driving through Nevada weeks prior, Tsukuhiko announced to the world that he was engaged to Tsukushi!'], 127.0.0.1 - - [24/Apr/2022 19:23:42] "POST /search HTTP/1.1" 200 -
127.0.0.1 - - [24/Apr/2022 19:23:42] "GET /static/css/search.css HTTP/1.1" 208 -

- We can see that Lord of Rings has the highest page-rank
- and we can also see that the results are sorted in decreasing order of page-rank,
i.e., results are displayed in decreasing order of relevancy(page-rank in our case).

• Reference - Screenshot of the terminal

- 9) Below screenshot shows the latency and throughput of the Smart Search Engine.

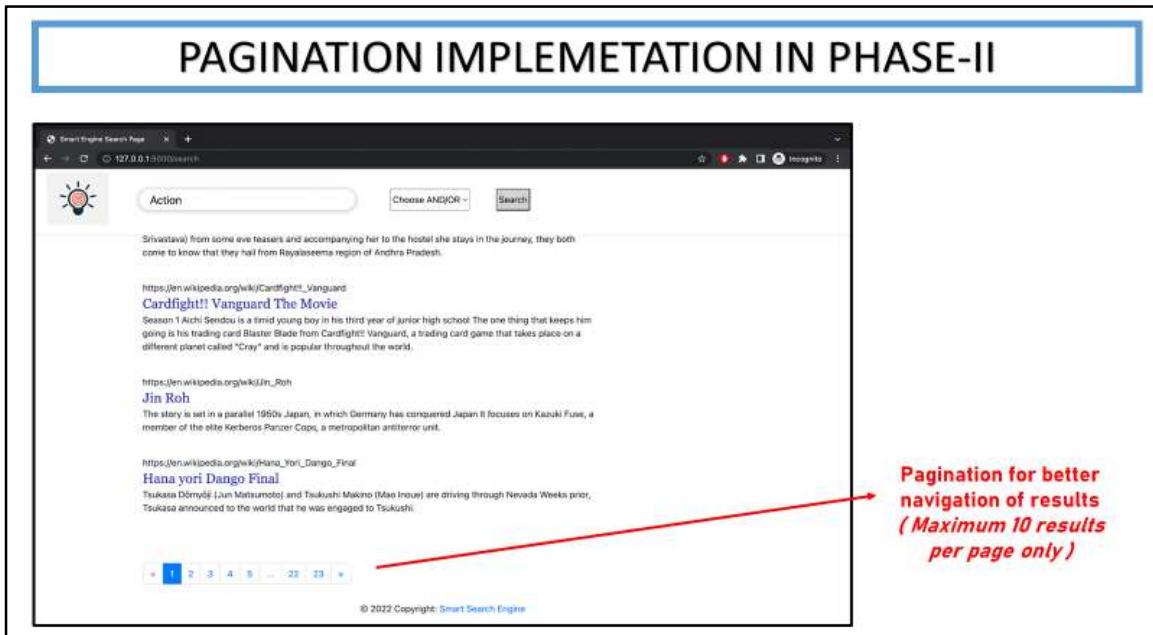
LOW LATENCY & HIGH THROUGHPUT IMPLEMENTATION IN PHASE-II



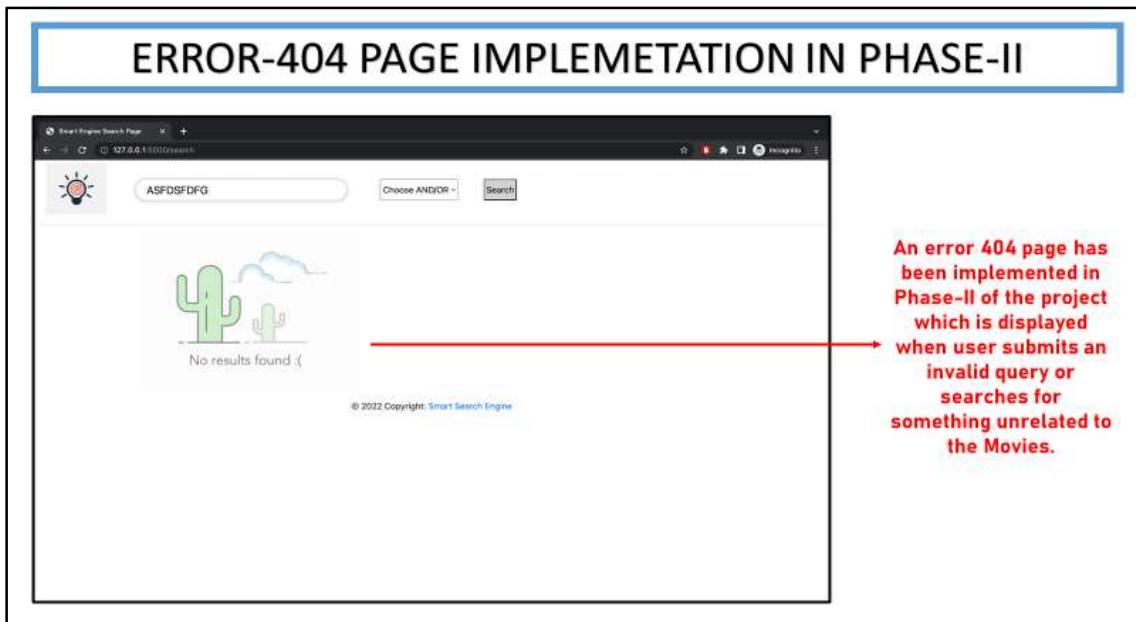
An additional functionality showing number of results as well as time taken to fetch those results is being displayed in Phase-II of the project.

Note: We can observe the low latency(time taken) and the number of results being fetched after trying to find a match from 1000 URLs(high throughput)

10) Below screenshot shows the pagination function implementation demo.



11) Below screenshot shows the Error-404 page for invalid results or results for query with no relevance to movies or our movie database.



REFERENCES:

- <https://towardsdatascience.com/how-to-build-a-search-engine-9f8ffa405eac>
- https://www.mckinsey.com/~media/mckinsey/dotcom/client_service/High%20Tech/PDFs/Impact%20of%20Internet%20technologies%20search%20final2.aspx
- <https://www.elastic.co/blog/elasticsearch-5-0-0-released>
- <https://www.google.com/search?q=components+in+search+engine&oq=compo&aqs=chrome.0.69i59j69i57j69i60l2.3466j0j7&sourceid=chrome&ie=UTF-8>
- <https://medium.com/analytics-vidhya/build-your-semantic-document-search-engine-with-tf-idf-and-google-use-c836bf5f27fb>
- <https://www.webnotes.com/what-are-different-types-of-search-engines/>

IMPORTANT DATES:

| Description of Deliverable | Submission Date |
|---|------------------------|
| Preliminary Project Plan | 2nd Feb 2022 |
| Interim Project I: PPT | 9th March 2022 |
| Final Project I submission | 23rd March 2022 |
| Interim project II | 18th April 2022 |
| Project Phase-2 PPT | 27th April 2022 |
| Final Project II submission (along with Final report) | 27th April 2022 |