***Smart Search Engine***

***Preliminary Project Plan***

**PROJECT MEMBERS:**

YELISETTI KRISHNA TEJA (NET-ID : KXY200016)

SATYA SOMEPALLI (NET-ID : SXS190436)

SAKETH DASAVATHINI (NET-ID : SXD190016)

HARSHITH RAVIPROLU (NET-ID : HXR180005)

SAI PRANAV REDDY DONTHIDI (NET-ID : SXD200125)

APUROOP PARAVADA (NET-ID : AXP210033)

SANJANA PENMETSA (NET-ID : SXP190149)

THOTA JAYASHREE SANTHOSHI (NET-ID : JXT210011)

B MOUNIKA (NET-ID : MXB210007)

PREETHAM RAO GOTTUMUKULA (NET-ID : PXG210001)

YOGESH BALA (NET-ID : YXB200007)

SRINATH REDDY MAVILLAPALLY (NET-ID : SXM210047)

KIRANDEVARAJ RAJ (NET-ID : KXR190038)

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

**Submitted for: CS 6359.001**

**Phase 0**

**1. Introduction**

**1.1 Project overview**

This is a project plan to be used for the implementation of a Smart Search Engine which is an information retrieval software. We are developing a system that would retrieve the topmost results (website URLs) associated with the keyword or a query when a user searches for it. We are building the system based on Object-oriented software architecture. Our system has a search interface, where users can search using keywords. The backend of the system consists of a crawler that will traverse through the collection of documents, decode the text, and produce a search index to be stored in the database. Indexing on the data would result in faster information retrieval. The data stored will contain different website Urls, the title, description about the websites, and different keywords that relate to a particular url. Whenever a user searches for a keyword by going to our search engine, the algorithm will consider that keyword entered, find the appropriate matches to the keyword, and return the results. In order to avoid the result overflow, we are putting a threshold and returning only the top results. The user can click on the results and view the websites and content associated with the searched term.

**1.2 Project deliverables**

The following are the deliverables for this project:

a) Preliminary Project Plan ----------------------- Deliverable 0

b) Requirements Elicitation –---------------------- Deliverable 1

c) Requirements Analysis ----------------------- Deliverable 2

d) Architectural Design ----------------------- Deliverable 3

e) Object/Component Design ---------------------- Deliverable 4

f) Coding ---------------------- Deliverable 5

g) Testing ---------------------- Deliverable 6

**1.3 Evolution of this document**

This is a preliminary document with the scope of the project. We will keep developing and revising the document as we gather more requirements, reviews, comments and specifications upon detailed research.

### **1.4 References**

* <https://towardsdatascience.com/how-to-build-a-search-engine-9f8ffa405eac>
* <https://www.mckinsey.com/~/media/mckinsey/dotcom/client_service/High%20Tech/PDFs/Impact_of_Internet_technologies_search_final2.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.webnots.com/what-are-different-types-of-search-engines/>

**1.5 Definitions, acronyms, and abbreviations**

**Definitions:**

**1. Search Interface:** The user interface consists of a search bar where the user searches a keyword.

**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

**2. Project organization**

### **2.1 Process model**

We are developing our model based on the UML diagrams for the project. UML diagrams provide an efficient way for modeling and designing, to visualize the project and understand the classes and the relationship between classes before implementation. Hence, UML diagrams will act as a foundation for our project programming. We will be using the agile model for our project development where our team will assess the project and feedback in regular meetings called sprints or iterations.

**2.2 Organizational structure**

The members involved in developing this project are:

* YELISETTI KRISHNA TEJA
* SATYA SOMEPALLI
* SAKETH DASAVATHINI
* HARSHITH RAVIPROLU
* SAI PRANAV REDDY DONTHIDI
* APUROOP PARAVADA
* SANJANA PENMETSA
* THOTA JAYASHREE SANTHOSHI
* MOUNIKA B
* PREETHAM RAO GOTTUMUKULA
* YOGESH BALA
* SRINATH REDDY MAVILLAPALLY
* KIRAN RAJ

For the first deliverable, YELISETTI KRISHNA TEJA will be the Team Leader for the project.

### **2.3 Organizational boundaries and interfaces**

Team lead and sub-group leads will be responsible to communicate with everyone during every deliverable and regular meetings will be conducted to discuss the progress and any limitations encountered. Every individual team member will be involved in the project and will be solely responsible for his/her given tasks.

### **2.4 Project responsibilities**

All the team members will be involved in all phases of the project life cycle : Research, Requirement Analysis and Gathering, UML Diagrams, Development, Testing & Documentations.

**3. Managerial process**

**3.1 Management objectives and priorities**

The main objective of the team is to work on a timely basis and distribute the work evenly among all the group members and sub-groups. Since we all belong to different domains or backgrounds taking our technical experience or exposure into consideration, we would like to have group discussions together so that everyone is up to date regarding each and every step during the project development phases.

### **3.2 Assumptions, dependencies, and constraints**

Assumptions:

* Search Engine: Upon keyword or query search, it will return the URLs whose context is relevant to the searched terms and the URLs are sorted based on the ranking algorithms.
* We will be using TF-IDF Vectorizer for our search engine implementation as it would return the most effective results.

Dependencies & Constraints:

* One of the major constraints will be distributing the work evenly among everyone.
* The team will have to strictly abide by the given time constraints and due dates.
* Efficient and accurate working of the project functionalities will be a priority during all the deliverables.

### **3.3 Risk management**

Risk management is defined as forecasting and evaluation of risks together with the identification of procedures to avoid or minimize the impact. Following are the strategies we will use to minimize or avoid the risk:

* Proper antivirus software will be used.
* Every team member will have a local version or backup of the project.
* The team will maintain the working version histories.

### **3.4 Monitoring and controlling mechanisms**

* Each phase of the project will be monitored and led by the main team and lead and the subgroup leads for that respective deliverable.
* Team Lead will correspond with everyone on a timely basis and update the Preliminary Project Plan document if there are any changes.
* Work will be distributed evenly among the sub-teams and due dates will be followed strictly for the timely progress of the project.
* All documents related to the project will be posted and updated in the GitHub link provided for reference.

**4. Technical process**

**4.1 Methods, tools, and techniques**

The UML modeling which implies the creation of UML diagrams will be done using the “Rational Rose” tool. Python programming language will be used to develop the algorithms. There are two reasons: One being all our team members are more familiar with python compared to other languages and also python offers easy, effective interfaces and vast support of libraries that would help us to efficiently build our search engine. Our project team is using Microsoft Teams for easy team member communication. All the project documents will be posted [**here**](https://github.com/krishnaty3/SmartSearchEngineDocs.git)on GitHub repository.

**4.2 Software documentation**

The following software documents will be developed:

* Preliminary Project Plan
* Requirements Specification
* UML Diagrams
* Python code snippets

### **4.3 Project support functions**

* The team will connect immediately whenever we encounter any issue regarding the code, technical assumptions, or any type of failure/limitations to the project.
* The team will collectively come up with the best approaches and ideas to deliver the best quality.
* Project-related documentation will be up to date and will be parallelly updated in the GitHub link.
* The timely progress of the project will be monitored to meet the specific deadlines without fail.
* We will make sure to follow the same coding guidelines to be on the same page.

### 

**5. Work elements, schedule, and budget**

This project is scheduled to be completed by April 27th, 2022 for the final demo. Here is the outline of the timeline of the deliverables:

Deliverable 1 due by 02/14/2022 - 02/16/2022

Deliverable 2 due by 02/28/2022 - 03/02/2022

Deliverable 3 due by 03/09/2022

Deliverable 4 due by 03/16/2022

Deliverable 5 & 6 due by 04/25/2022 – 04/27/2022