1. **INTRODUCTION**

**1.1 OVERVIEW OF THE PROJECT**

To retrieve information about the query which is given by the user in depth manner by finding synonym, hypernym and hyponym of the query given by the user that is pre-processed. We use customized stop list to pre-process the user’s query for removal of stop words. Further, successor count root words are found by “successor stemming algorithm” for eliminating the suffix of the given query. There are four techniques used in this stemming process to stem the words. They are cut-off, peak and plateau, complete word method and entropy. For the obtained root word, we find hypernym, hyponym and synonym using “Natural Language Processing”. Then for those root words, term frequency and inverse document frequency are calculated and based on that calculations, we have stored website links. These links are retrieved when the user selects the word to know in detail. Images are also stored for perceptual view. We have additionally added a feature “bookmark” that saves the websites both locally and globally. So that they are browser independent and can also be used in all types of browsers without importing it.

**1.2 SCOPE OF THE PROJECT**

The aim of the project is to develop a search engine which can be used for exploratory search for retrieving unawarefields for different kind of users using ontology clustering and to provide a browser independent book marking system. The search engine existing now will provide good results only if the keywords are given correctly in search query. So we are going to develop a search engine which allows user to search even if they doesn’t know exact keywords using ontology clustering and multi term search and to provide a bookmarking system independent of browsers. The purpose of project is to provide an exploratory search for users to know about different fields using Hypernym, Hyponym, Synonym of the keywords given in search query and to allow them to bookmark globally and locally by storing bookmarks in database.

**1.3 OBJECTIVE OF THE PROJECT**

The search engine existing now will provide good results only if the keywords are given correctly in search query. So we are going to develop a search engine which allows user to search even if they doesn’t know exact keywords using ontology clustering and multi term search and to provide a bookmarking system independent of browsers.

* 1. **EXISTING SYSTEM**

Smart Crawlerperforms site-based searching for center pages with the help of search engines, avoiding visiting a large number of pages. A matching query style is unsuitable to investigate information in unfamiliar fields and to learn new associations and knowledge with relevance to a query. When users cannot specify accurate search words, the search results are often useless. Furthermore, as information becomes more ubiquitous and demands for various searches grow, there is an increasing need to support search behaviours beyond simple lookup.In the second stage, Smart Crawlerachieves fast in-site searching by excavating most relevant links with an adaptive link-ranking. To eliminate bias on visiting some highly relevant links in hidden web directories, we design a link tree data structure to achieve widercoverage for a website.

* 1. **NEED FOR THE PROJECT**

The problem in the existing system is that the users investigate unfamiliar fields the search would not be efficient and the system uses “smart crawler method” which avoids users from visiting large number of web pages. As the information becomes more ubiquitous and demands for various searches grow, there is an increasing need to support search behaviours beyond simple lookup. In current search engine the bookmarking system is dependent, so it can’t be used in other search engines.

**1.6 PROPOSED SYSTEM**

We proposed an efficient and flexible search scheme that supports both multi-keyword ranked search and synonym hyponym and hypernym based search. To implement multi-keyword search and result ranking, Vector Space Model (VSM) is used to build document index, where each document is expressed as a vector and each dimension value is the “Term Frequency (TF) weight” of its corresponding keyword. A new vector is also generated in the query phase. The document has the same dimension with document index and each dimension value is the “Inverse Document Frequency (IDF) weight.” To compute similarity of one document in a large dimension to search a query we use “Cosine measure” that calculates the angle between them. Tree base index structure is used to improve search efficiently and balanced by Binary tree.  The searchable index tree is constructed with the document index vectors, So the related documents can be found by traversing the tree.

**2.LITERATURE SURVEY**

The purpose of the Literature Survey is to give the brief overview and also to establish complete information about the reference papers. The goal of Literature Survey is to completely specify the technical details related to the main project in a concise and unambiguous manner.

**2.1 SURVEYS**

[1] ChangJun Jiang introduces an indexing network model and five related normal forms to advance the field. As a basic model, indexing network organizes and manages various information service resources through analyzing the relationships among them. Along with the enormous amount of information service resources on the Internet, it is increasingly necessary to consider the urgent problems it has brought, such as diversity, heterogeneity, disorder, and redundancy.

[2] Michael T. Mills and Nikolaos G. Bourbakis presents the functional components, performance, and maturity of graph-based methods for natural language processing and natural language understanding and their potential for mature products. This paper presents a summary of such graph-based methods that are found in recent technical publications plus an analysis of their component functions and their maturity calculated from information found in the referenced papers.

[3] Hung Chim and Xiaotie Deng propose a phrase-based document similarity to compute the pair wise similarities of documents based on the Suffix Tree Document (STD) model. This conclusion sufficiently explains why the phrase-based document similarity works much better than the single-word tf-idf similarity measure.

[4] Adam L. Kaczmarek concerns clustering-by-directions algorithm. The algorithm introduces a novel approach to interactive query expansion. It is designed to support users of search engines in forming Web search queries. It focuses on a novel approach to facilitating users in forming queries. The approach is based on clustering-by directions (CBD) algorithm which was introduced by the author in.

[5] Jianxin Li, Chengfei Liu and Rui Zhou ,Wei Wang proposes a new keyword search approach which basically utilizes the statistics of underlying XML data to decide the promising result types and then quickly retrieves the corresponding results with the help of selected promising result types. Keyword search enables inexperienced users to easily search XML database with no specific knowledge of complex structured query languages and XML data schemas

[6]Hengshu Zhu, Enhong Chen,Hui Xiong, Huanhuan Cao, Jilei Tian proposes an expert finding framework based on the authority information in the target category as well as the relevant categories for ranking user authority.A scalable method for measuring the relevancies between categories through topic models, which takes consideration of both content and user interaction based category similarities is developed here.

[7] RajendraKumarRoul, Omanwar Rohit Devanand, S. K. Sahay proposed a new mechanism called Tf-Idf based Apriori for clustering the web documents. It then rank the documents in each cluster using Tf-Idf and similarity factor of documents based on the user query.

[8] Heiko Paulheimin discusses a framework of adding background knowledge from Linked Open Data to a given data mining problem in a fully automatic, unsupervised manner. It introduces the FeGeLOD framework and its latest implementation, the RapidMinerLinked Open Data extension.

[9] A novel probabilistic retrieval model is presented by Ho Chung Wu. It forms a basis to interpret the TF-IDF term weights as making relevance decisions. The signiﬁcance of interpreting TF-IDF in this way is the potential to: (1) establish a unifying perspective about information retrieval as relevance decision-making; and (2) develop advanced TF-IDF-related term weights for future elaborate retrieval models.

[10] Moritz Tenorth*,* Alexander Clifford Perzylo, ReinhardLafrenz, and Michael Beetz reports on the formal language developed for encoding information and present approaches to solve the inference problems related to finding information, to determining if information is usable by a robot, and to grounding it on the robot platform. ROBOEARTH is intended to be a web community by robots for robots to autonomously share descriptions of tasks they have learned, object models they have created, and environments they have explored.

**2.2 FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase is put forth with very general plan for the project and some cost estimates. During system analysis, the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the user.

**Economical feasibility**

Search engine organizations will get benefited as they can use our system for all kind of users who are using their search engine. Economic feasibility attempts 2 weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system.

**Social feasibility**

Social feasibility addresses the influences that the proposed project may have on the social system in the project environment. All kind of users will be benefited even if they doesn’t know exact keywords they can use our system for searching as wordnet tool is used to get proper keywords.

**Resource feasibility**

The system is said to be beneficial, if the proper resource is utilized in the concerned project. The resource or requirements includes software or hardware requirements needed to perform the operations in the project. This evolves the migration of existing to incorporate with the new systems using impact analysis.

**3.SYSTEM DESIGN SPECIFICATION**

Design is a meaningful engineering representation of something that is to be built. System Design is a process through which the requirements are translated into a representation of the software. Design provides us with representations of software that can be assessed for quality.

**3.1 SYSTEM ARCHITECTURE DESIGN**

**Fig 3.1 Architecture of proposed system**

To retrieve information about the query which is given by the user in depth manner by finding synonym, hypernym and hyponym of the query given by the user that is pre-processed. We use customized stop list to pre-process the user’s query for removal of stop words. Further, successor count root words are found by “successor stemming algorithm” for eliminating the suffix of the given query. There are four techniques used in this stemming process to stem the words. They are cut-off, peak and plateau, complete word method and entropy. For the obtained root word, we find hypernym, hyponym and synonym using “Natural Language Processing”. Then for those root words, term frequency and inverse document frequency are calculated and based on that calculations, we have stored website links. These links are retrieved when the user selects the word to know in detail. Images are also stored for perceptual view. We have additionally added a feature “bookmark” that saves the websites both locally and globally. So that they are browser independent and can also be used in all types of browsers without importing it.

**ARCHITECTURE DESCRIPTIONS**

There are nine components present in our project which is given in detail as follows, the nine components are

1. User interface
2. Input key term
3. Term pre processing
4. Term similarity generation
5. Ontology clustering
6. Web server
7. Bookmark
8. Database
9. Jelastic cloud environment

**USER INTERFACE**

To know in-depth details about certain keywords and its meaning, the user gives query in the search engine where the interface acts as an intermediate between the user and the system. The user can even search about unaware fields that he wants to gain more knowledge about it. To retrieve information, the user must have registered by giving his details that is stored in secured manner. The registration process, it contains: user name, password, mail id and mobile number. For any enquiries, they can contact us for help. The user must login for accessing the search engine.

**INPUT KEY TERM**

User can give the input key terms to gain more knowledge about the certain field. All the key terms can be stored in the database with their related meanings so that it is easy to retrieve the similar datas related to the given key term. Each and every details are stored in the database.

**TERM PROCESSING**

For the given query, pre processing is done to obtain the root word from the given query. So that it can retrieve the datas related to that root word. Time efficiency is achieved by using this technique. For this pre processing we have used two algorithms, they are

* Success variety stemming algorithm
* Stopping algorithm

**SUCCESS VARIETY STEMMING ALGORITHM**

For stemming less weight-age words we are using successor variety stemming algorithm as it is efficient and find correct root words using different strategies stated below:

1. Using the **cutoff method**, some threshold value is selected for successor varieties and a limit is identified whenever the threshold value is reached. The problem with this technique is how to select the threshold value. If it is too small, incorrect splits will be made; if too large, correct splits will be missed.
2. With the **peak and plateau method**, a segment split is made after a letter whose successor variety exceeds that of the letter immediately preceding it and the letter immediately following it. This method removes the need for the threshold value to be selected.
3. In the **complete word method**, a split is made after a segment if the segment is a complete word in the corpus.
4. The **entropy** method takes advantage of the distribution of successor variety characters. The method works as follows. Let |Dαi| be the number of words in a text body beginning with the i length sequence of letters. Let |Dαij| be the number of words in Dαi with the successor j. The probability that a member of Dαi has the successor j is given by

**αi  .**

These four strategies are combined together for efficiency and it is the best stemming algorithm by performance wise.

**STOP WORD LIST**

**Customised Stop Word List**

Combination of [Snowball stop word](http://snowball.tartarus.org/algorithms/english/stop.txt) list -Terrier stop word list and minimal stop word list together and make a customized list for efficient removal of stop words.

i)[Snowball stop word](http://snowball.tartarus.org/algorithms/english/stop.txt)list – this stop word list is associated with the Snowball Stemmer

ii)[Terrier stop word](https://bitbucket.org/kganes2/text-mining-resources/downloads)list – this is a simple and précised stop word list associated with the Terrier package.

iii)[Minimal stop word](https://bitbucket.org/kganes2/text-mining-resources/downloads) list – this is a stop word list that made consisting of determiners, coordinating conjunctions and prepositions and some question words.

**TERM SIMILARITY GENERATION**

After pre processing, term similarity generation can be done by generating the similar words from the obtained root word. For this process, k means clustering is used where datas that have same meaning are stored in a single cluster.

**K-MEANS CLUSTERING ALGORITHM**

For finding similarity between keywords we are using k-means clustering algorithm as it is simple and more efficient for searching purpose.

Let  X = {x1,x2,x3,……..,xn} be the set of data points and V = {v1,v2,…….,vc} be the set of cluster centers.

1) Randomly select *‘c’* cluster centers.

2) Compute the distance between each seed point and centers of clusters which we selected randomly

3) Assign the seed point to the cluster center whose distance from the cluster center is less than that of all the cluster centers.

4) Recalculate the new cluster center like finding mean for seed points.

5) Recalculate the distance between each seed point and new obtained centers

6) If no seed point was reassigned then stop, otherwise repeat from step 3.

This  algorithm  aims at  minimizing  squared error function given by:

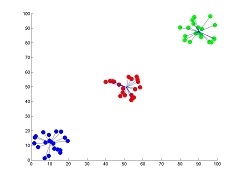
**=**

where,

*‘||xi- vj||’* is the Euclidean distance between *xi* and *vj.*

*‘ci’* is the number of seed points in *ith* cluster.

*‘c’* is the number of cluster centers (mean)



**Fig: an example for forming clusters using similarity**

**Advantage**

Relatively efficient: O(tknd), where n is # objects, k is # clusters, d is # dimension of each object, and t  is # iterations. Normally, k, t, d << n. Fast and robust too.

**ONTOLOGY CLUSTERING**

Here, based on the term similarities, it shows synonym, hypernym and hyponym of the root word that is pre processed. Vector space algorithm is used for this process where the shortest distance between the similar words are measured .

**VECTOR SPACE MODEL**

For ranking websites we are using vector space model as it is efficient and involves cosine similarity function to find similarity between keywords and links.

1.Find term frequency for keywords using Robinson tf (okapitf):

2. Find inverse document frequency for website links using below formula

**IDF(t)=log(N/Nt)**

where, N= #of docs

Nt= #of docs containing term t

3. Compute cosine similarity for tf and idf like below,

**.868**

4. Compute tf-idf base similarity using below formula,

**WEB SERVER**

The primary function of a web server is to store, process and deliver [web pages](https://en.wikipedia.org/wiki/Web_page) to [clients](https://en.wikipedia.org/wiki/Client_%28computing%29). The communication between client and server takes place using the [Hypertext Transfer Protocol (HTTP)](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol). Pages delivered are most frequently [HTML documents](https://en.wikipedia.org/wiki/HTML), which may include [images](https://en.wikipedia.org/wiki/Image), [style sheets](https://en.wikipedia.org/wiki/Style_sheet_%28web_development%29) and [scripts](https://en.wikipedia.org/wiki/JavaScript) in addition to text content. The server also gives details about the data that the user wants to gain knowledge about. The server stores and maintains data in the database.

**BOOKMARK**

In the context of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), a bookmark is a [Uniform Resource Identifier](https://en.wikipedia.org/wiki/Uniform_Resource_Identifier) (URI) that is stored for later retrieval in any of various storage formats. Bookmarks are called favorites or Internet shortcuts in [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer), and by virtue of that browser's large [market share](https://en.wikipedia.org/wiki/Market_share), these terms have been synonymous with bookmark since the [first browser war](https://en.wikipedia.org/wiki/First_browser_war). Bookmarks are normally accessed through a menu in the user's web browser, and [folders](https://en.wikipedia.org/wiki/Directory_%28file_systems%29) are commonly used for organization. In addition to bookmarking methods within most browsers, many external applications offer bookmark management.

Local and global book marking can be done for the page user using and it is very helpful as it is not browser dependent and book marks are stored in database. This can overcome the drawback in real browsers since in real browsers can only book mark in one particular browser and if it is needed in another browser it has to import the book marks. Instead of that we can save book marks in database or user’s folders.

**DATABASE**

A database is an organized collection of [data](https://en.wikipedia.org/wiki/Data_%28computing%29). It is the collection of [schemas](https://en.wikipedia.org/wiki/Database_schema), [tables](https://en.wikipedia.org/wiki/Table_%28database%29), [queries](https://en.wikipedia.org/wiki/Query_language), reports, [views](https://en.wikipedia.org/wiki/View_%28SQL%29), and other objects. The data are typically organized to model aspects of reality in a way that supports [processes](https://en.wikipedia.org/wiki/Process_%28computing%29) requiring information, such as modeling the availability of rooms in hotels in a way that supports finding a hotel with vacancies. A database management system (DBMS) is a [computer software](https://en.wikipedia.org/wiki/Computer_software) application that interacts with the user, other applications, and the database itself to capture and analyze data. A general-purpose DBMS is designed to allow the definition, creation, querying, update, and administration of databases. Well-known DBMSs include [MySQL](https://en.wikipedia.org/wiki/MySQL), [PostgreSQL](https://en.wikipedia.org/wiki/PostgreSQL), [MongoDB](https://en.wikipedia.org/wiki/MongoDB), [MariaDB](https://en.wikipedia.org/wiki/MariaDB), [Microsoft SQL Server](https://en.wikipedia.org/wiki/Microsoft_SQL_Server), [Oracle](https://en.wikipedia.org/wiki/Oracle_Database), [Sybase](https://en.wikipedia.org/wiki/Sybase), [SAP HANA](https://en.wikipedia.org/wiki/SAP_HANA), [MemSQL](https://en.wikipedia.org/wiki/MemSQL) and [IBM DB2](https://en.wikipedia.org/wiki/IBM_DB2).

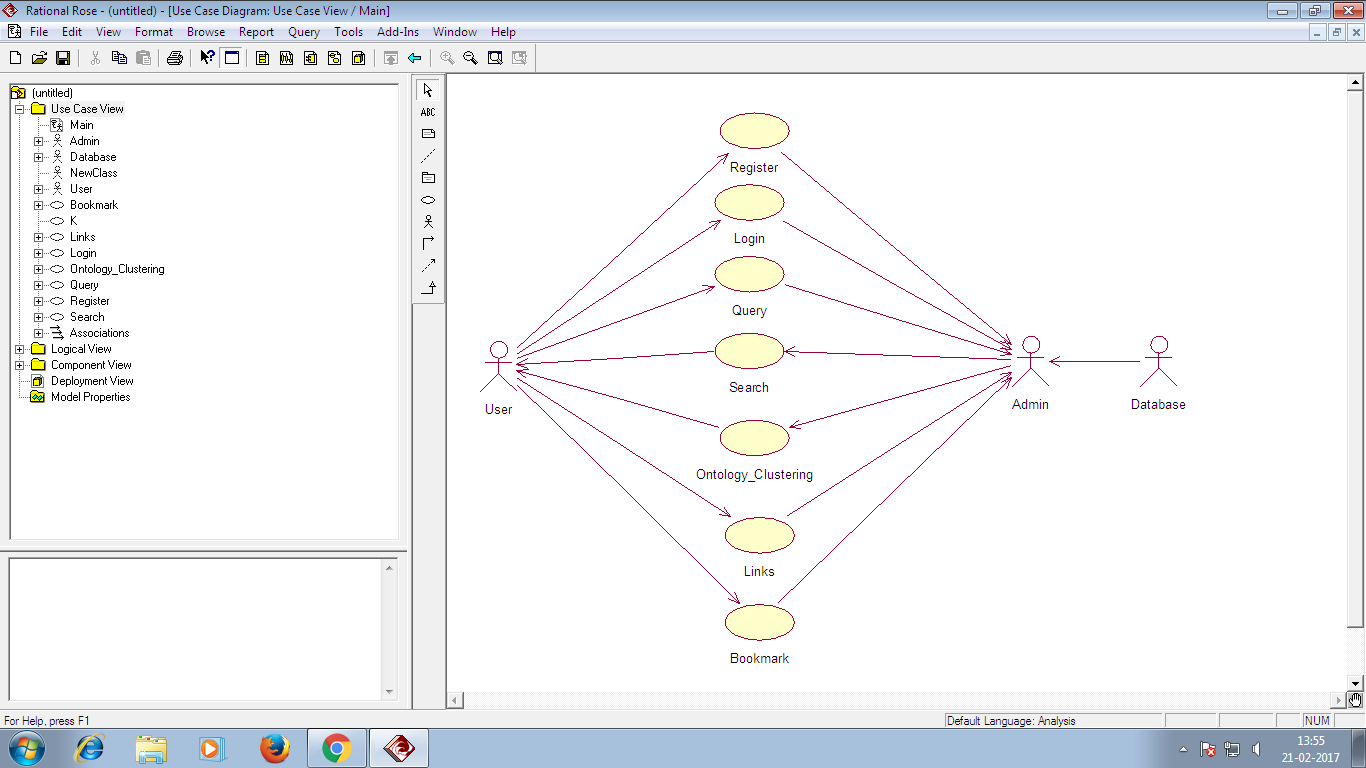
**JELASTIC CLOUD ENVIRONMENT**

Jelastic ([acronym](https://en.wikipedia.org/wiki/Acronym) for Java Elastic) is an unlimited [PaaS](https://en.wikipedia.org/wiki/PaaS) and Container based [IaaS](https://en.wikipedia.org/wiki/IaaS) within a single platform that provides high availability of applications, automatic vertical and horizontal scaling via containerization to [software development](https://en.wikipedia.org/wiki/Software_development) clients, enterprise businesses, [DevOps](https://en.wikipedia.org/wiki/DevOps), [System Admins](https://en.wikipedia.org/wiki/System_administrator), Developers, OEMs and [web hosting](https://en.wikipedia.org/wiki/Web_hosting) providers. The company has developed technologies for moving [Java](https://en.wikipedia.org/wiki/Java_%28programming_language%29), [Node.js](https://en.wikipedia.org/wiki/Node.js), [Ruby](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29), [Python](https://en.wikipedia.org/wiki/Python_%28programming_language%29), [.NET/Windows](https://en.wikipedia.org/wiki/.NET_Framework) and [PHP](https://en.wikipedia.org/wiki/Php)-based [applications](https://en.wikipedia.org/wiki/Computer_applications) onto the cloud, solving the task of application delivery and simultaneously providing orchestration of multi-container environments in public, hybrid and private clouds. Multi-cloud feature provides the ability to achieve higher availability through geo-distribution among different data centers or clouds, easily relocate the projects to the superior hardware with the help of environment migration, choose between higher quality or more cost affordable hardware and host applications with the trusted cloud vendors.

**3.2 UML DIAGRAMS:**

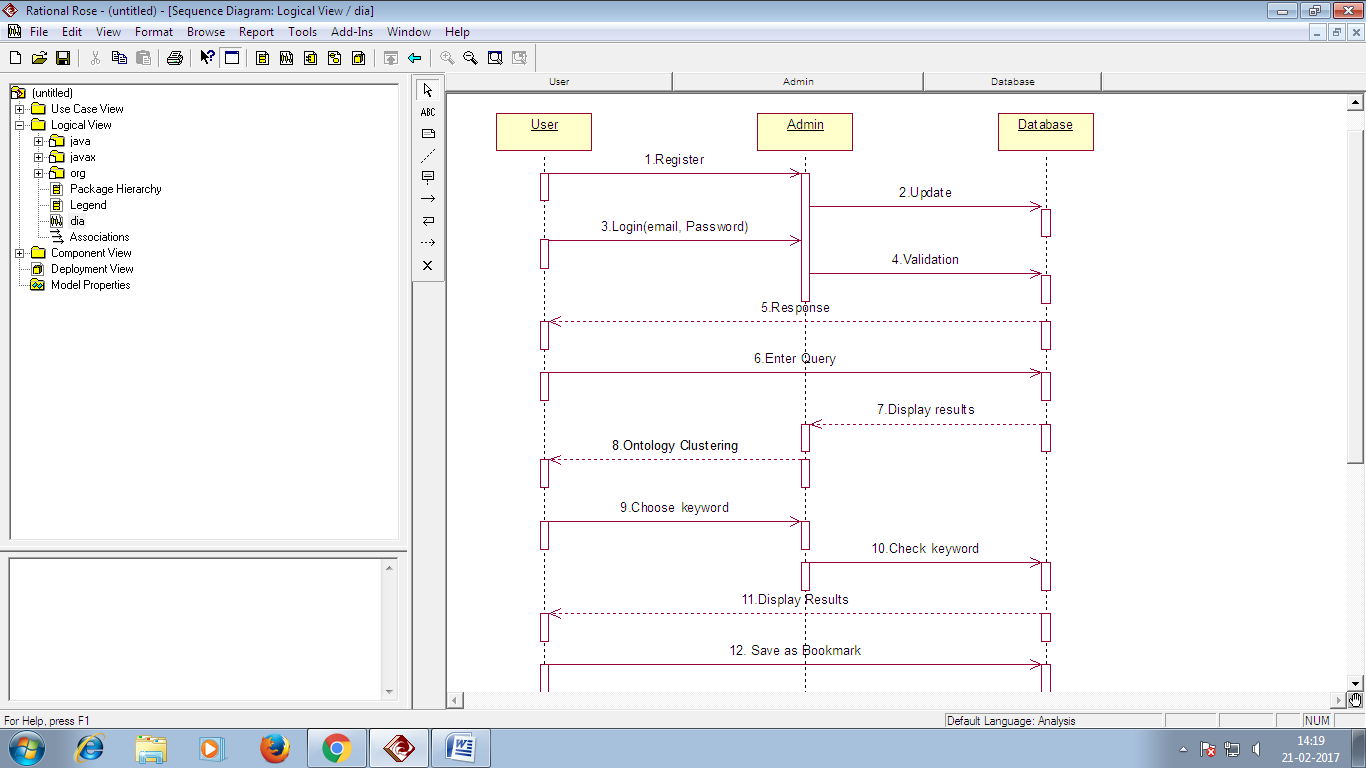
**3.2.1USE CASE DIAGRAM:**

Use case diagrams are usually referred as behavior diagram which is used to describe a set of actions that some system can perform in collaboration with one or more external users of the system.Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors,their goals and their dependencies.



**3.2.2SEQUENCE DIAGRAM:**

**Sequence diagram** is the most common kind of interaction diagram, which focuses on the [message](http://www.uml-diagrams.org/interaction-message.html) interchange between a number of [lifelines](http://www.uml-diagrams.org/sequence-diagrams.html#lifeline). Sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.



**3.2.3CLASS DIAGRAM :**

A class diagram in the UML is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, and the relationships between the classes. Private visibility hides information from anything outside the class partition. Public visibility allows all other classes to view the marked information. Protected visibility allows child classes to access information they inherited from a parent class.

user



input key term



Term Preprocessing



term similarity generation()



Ontology clustering()

Data base



inputtermsets



tables()



views()



triggers()

web server



user interface



database



userspcification()



db details()

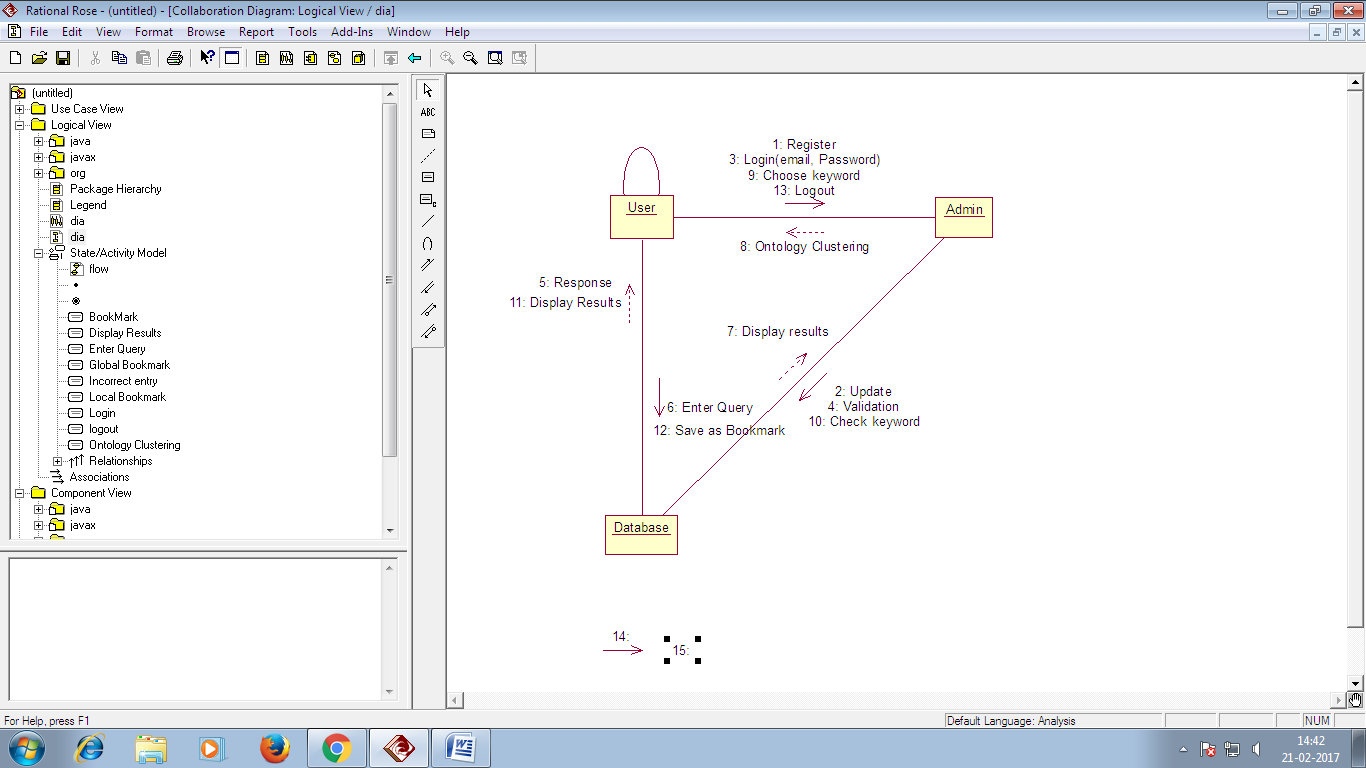
**3.2.4STATE CHART DIAGRAM:**

Activity diagram is basically a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deals with all type of flow control by using different elements like fork, join etc.

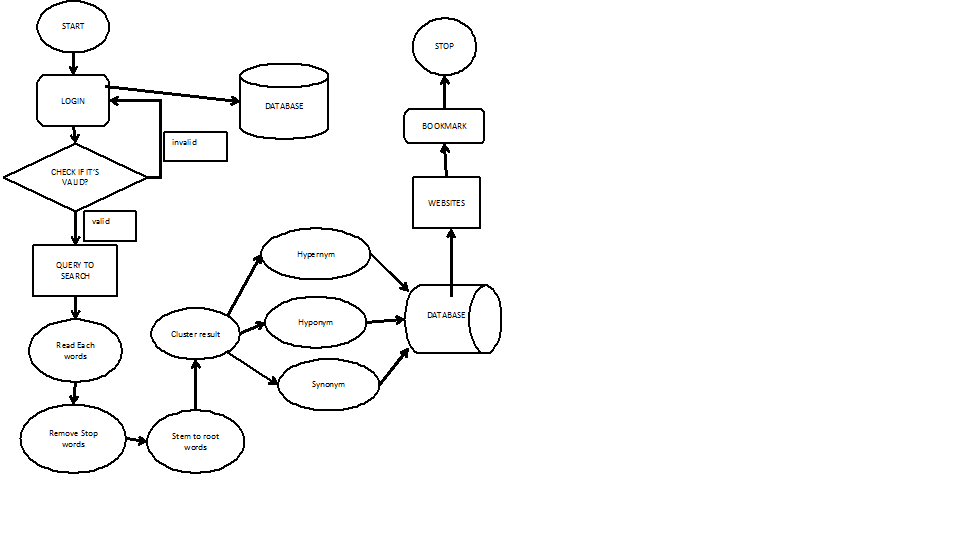


**3.2.5 COLLABORATION DIAGRAM:**

A collaboration diagram show the objects and relationships involved in an interaction, and the sequence of messages exchanged among the objects during the interaction. The collaboration diagram can be a decomposition of a class, class diagram, or part of a class diagram. it can be the decomposition of a use case, use case diagram, or part of a use case diagram. The collaboration diagram shows messages being sent between classes and object (instances).



**3.3 OVERALL OF SYSTEM DIAGRAM**



We proposed an efficient and flexible search scheme that supports both multi-keyword ranked search and synonym hyponym andhypernymbased search. To implement multi-keyword search and result ranking, Vector Space Model (VSM) is used to build document index, where each document is expressed as a vector and each dimension value is the “Term Frequency (TF) weight” of its corresponding keyword. A new vector is also generated in the query phase. The document has the same dimension with document index and each dimension value is the “Inverse Document Frequency (IDF) weight.” To compute similarity of one document in a large dimension to search a query we use “Cosine measure” that calculates the angle between them. Tree base index structure is used to improve search efficiently and balanced by Binary tree.  The searchable index tree is constructed with the document index vectors, So the related documents can be found by traversing the tree.

**4. MODULE DESCRIPTIONS**

**4.1 MODULES**

Software is divided into separately named and addressable components called modules that are integrated to satisfy problem requirements. Modularity is the single attribute of software that allows a program to be intellectually manageable.

Exploratory search for retrieving unaware fields for users usingontology clustering is divided intosix different modules. They are

* User Interface
* Data Preprocessing
* Ontology Clustering
* Multi-term Search
* Cluster the Most Relevant Content
* Book marking

**4.2 MODULES DISCRIPTION:**

**4.2.1 USER INTERFACE:**

New user register to retrieve information in our project. Registering contains User name, password, mail-ID and phone number. Mail-ID is used for login as it’s an unique field and if the user forgets his password a mail is sent to the user for change of password in a secured manner. The user can enter the web-page after the verification of the user. After login, web user can enter the search space page. In this environment, the user searches the content from the web server. This Search Space is the interface for the user and web server. The input is given as a query by user for gaining knowledge by searching relevant information.

**4.2.2 DATA PRE-PROCESSING:**

For data pre-processing we use two main methods namely,

* CUSTOMIZED STOP LIST:

For the given query by user, stop word is filtered. Stop words means removal of less weighted words like ‘who’, ‘what’, ‘how’, ‘a’, ‘is’, ‘the’, etc which is done by pre-processing. It is controlled and implemented using customized stop list.

* SUCCESSOR STEMMING ALGORITHM:

After stopping process is done, another process of stemming is implemented. Stemming of word is the removal of suffix to get the root word. Stemmers employ a lookup table which contains relations between root forms and normal forms. To stem a word, the table is queried to find a matching word. If a matching word is found, the associated root form is returned. Eg: A stemming algorithm reduces the words "running” and "runner" to the root word, "run".

**4.2.3 ONTOLOGY CLUSTERING:**

Words ending in ’nyms’ are often used to describe different classes of words, and the relationships between words.

**Synonym**: A word which have similar meaning.

For example, one synonym of sad is "[gloomy](http://www.yourdictionary.com/gloomy)" however, this word carries quite a negative connotation. Depending on the circumstance you can use it, but if you just want to say that someone is "down," then another synonym such as "blue" or "unhappy" would be more applicable.

**Hypernym**: A word that has a more general meaning than another.

For example, the word “color” is a hypernym for red, green, blue, brown, etc.,

**Hyponym**: A word that has specific meaning.

For example, it is the specific meaning of hyponym like red, green, blue, brown, etc.

**4.2.4 MULTI TERM SEARCH:**

The term "multiple word search" is applied to written words that can have more than one use or definition, and the user can choose the word from different meanings. “Homographs**”** are words that are written the same, but have different meanings and pronunciations. Example :*lead* (the metal) and *lead* (the verb for going ahead). Without the sound difference, it may need to be clarified.

**4.2.5 CLUSTER THE MOST RELEVANT CONTENT:**

Clustering sentences by keywords. SIMBA produces a generic summary. Thus, the keywords that represent the global topic within the collection of texts are identified. The candidate keywords list contains common and proper names. It is built considering the lemma of the words, to ensure that the words in the collection are unique. There-after, the list is ordered considering the score of each word. in the clusters that represent the first set of keywords. These sentences are considered more significant than the others, since they address the main topics conveyed by the collection of texts.

**4.2.6 BOOKMARK:**

Local and global book marking can be done for the page user using and it is very helpful as it is not browser dependent and book marks are stored in database. This can overcome the drawback in real browsers since in real browsers we can only book mark in one particular browser and if we need that in another browser we need to import the book marks. Instead of that we can save book marks in database or user’s folders.

**5.SYSTEM REQUIREMENTS SPECIFICATION**

The purpose of the Software Requirement Specification is to produce the specification of the analysis task and also to establish complete information about the requirement, behavior and other constraints such as functional performance and so on. The goal of Software Requirement Specification is to completely specify the technical requirements for the software product in a concise and unambiguous manner.

**5.1 SOFTWARE REQUIREMENTS:**

* + Operating System : Windows 8 or Higher
  + Languages used : Java,J2EE(JSP, Servlet), HTML ,

CSS,Javascript

* + Tools : JDK 1.7, Net Beans 7.0.1, PhpMyAdmin,

Jelastic cloud environment

* + Backend : MySQL
  1. **HARDWARE REQUIREMENTS:**
  + Processor : Pentium Dual Core 2.3 GHz
  + Hard Disk : 250 GB or Higher
  + Ram : 4 GB
  1. **LANGUAGE SPECIFICATION**

This project is implemented using Java,Servlet,JSP,HTML,CSS,Javascript and Mysql is used as backend.

**JAVA**

Java is Platform Independent. Java is an object-oriented programming language developed initially by James Gosling and colleagues at Sun Microsystems. It implements a strong security model, which prevents compiled Java programs from illicitly accessing resources on the system where they execute or on the network. Popular World-Wide Web browsers, as well as some World-Wide Web servers and other systems implement Java interpreters. These are used to display interactive user interfaces, and to script behavior on these systems.

**SERVLET**

Servlet technology is used to create web application (resides at server side and generates dynamic web page).Servlet technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was popular as a server-side programming language. But there was many disadvantages of this technology. There are many interfaces and classes in the servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponseetc. It is a web component that is deployed on the server to create dynamic web page.

**JSP**

JavaServer Pages (JSP) is a technology for developing web pages that support dynamic content which helps developers insert java code in HTML pages by making use of special JSP tags, most of which start with <% and end with %>.A JavaServer Pages component is a type of Java servlet that is designed to fulfill the role of a user interface for a Java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and embedded JSP actions and commands. Using JSP, you can collect input from users through web page forms, present records from a database or another source, and create web pages dynamically.JSP tags can be used for a variety of purposes, such as retrieving information from a database or registering user preferences, accessing JavaBeans components, passing control between pages and sharing information between requests, pages etc.

**HTML**

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages.

* Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus the link available on a webpage are called Hypertext.
* As its name suggests, HTML is a Markup Language which means you use HTML to simply "mark up" a text document with tags that tell a Web browser how to structure it to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

**CSS**

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

**JAVASCRIPT**

Javascript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities. JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers. The [ECMA-262 Specification](http://www.ecma-international.org/publications/index.html) defined a standard version of the core JavaScript language.

* JavaScript is a lightweight, interpreted programming language.
* Designed for creating network-centric applications. Open and cross-platform.
* Complementary to and integrated with Java. Complementary to and integrated with HTML.

**MYSQL**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. It is developed, marketed, and supported by MySQL AB, which is a Swedish company. MySQL is released under an open-source license. So you have nothing to pay to use it. It is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages. It uses a standard form of the well-known SQL data language. It works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc. It works very quickly and works well even with large data sets.

**6. IMPLEMENTATION**

**6.1 SAMPLE CODE:**

**Login.java**

importjava.io.IOException;

importjava.io.PrintWriter;

importjava.sql.\*;

importjavax.servlet.RequestDispatcher;

importjavax.servlet.ServletException;

importjavax.servlet.http.HttpServlet;

importjavax.servlet.http.HttpServletRequest;

importjavax.servlet.http.HttpServletResponse;

public class Login extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throwsServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

String user=request.getParameter("user").trim();

String pass=request.getParameter("pass").trim();

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/deepsearch","root","root");

Statement st=con.createStatement();

ResultSetrs=st.executeQuery("select \* from user");

while(rs.next())

{

String us=rs.getString("mailid").trim();

String ps=rs.getString("password").trim();

if(user.equals(us) &&pass.equals(ps))

{

System.out.println("Success");

response.sendRedirect("1search.jsp");

}

else

{

response.sendRedirect("login.jsp");

}

}

}

catch(Exception e)

{

System.out.println(e);

} finally {

out.close();

}

}

**Preprocess.java**

importjava.io.FileOutputStream;

importjava.io.IOException;

importjava.io.PrintStream;

importjava.io.PrintWriter;

importjavax.servlet.ServletException;

importjavax.servlet.http.HttpServlet;

importjavax.servlet.http.HttpServletRequest;

importjavax.servlet.http.HttpServletResponse;

importjavax.servlet.http.HttpSession;

public class Preprocess extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throwsServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

HttpSession session=request.getSession(true);

try {

String key=request.getParameter("key");

session.setAttribute("Rinp", key);

String ppkey="";

String keya[]=key.split(" ");

for(int i=0;i<keya.length;i++)

{

String swr=keya[i];

if(!swr.equalsIgnoreCase("how") && !swr.equalsIgnoreCase("a") && !swr.equalsIgnoreCase("and")&&!swr.equalsIgnoreCase("he")&&!swr.equalsIgnoreCase("i")

&&!swr.equalsIgnoreCase("in")&&!swr.equalsIgnoreCase("is")&&!swr.equalsIgnoreCase("it")&&!swr.equalsIgnoreCase("of")

&&!swr.equalsIgnoreCase("that")&&!swr.equalsIgnoreCase("the")&&!swr.equalsIgnoreCase("to")&&!swr.equalsIgnoreCase("was")

&&!swr.equalsIgnoreCase("all")&&!swr.equalsIgnoreCase("are")&&!swr.equalsIgnoreCase("as")&&!swr.equalsIgnoreCase("at")

&&!swr.equalsIgnoreCase("be")&&!swr.equalsIgnoreCase("but")&&!swr.equalsIgnoreCase("for")&&!swr.equalsIgnoreCase("had")

&&!swr.equalsIgnoreCase("him")&&!swr.equalsIgnoreCase("his")&&!swr.equalsIgnoreCase("not")&&!swr.equalsIgnoreCase("on")

&&!swr.equalsIgnoreCase("one")&&!swr.equalsIgnoreCase("said")&&!swr.equalsIgnoreCase("so")&&!swr.equalsIgnoreCase("they")

&&!swr.equalsIgnoreCase("we")&&!swr.equalsIgnoreCase("with")&&!swr.equalsIgnoreCase("you")&&!swr.equalsIgnoreCase("adout")

&&!swr.equalsIgnoreCase("an")&&!swr.equalsIgnoreCase("back")&&!swr.equalsIgnoreCase("been")&&!swr.equalsIgnoreCase("before")

&&!swr.equalsIgnoreCase("big")&&!swr.equalsIgnoreCase("by")&&!swr.equalsIgnoreCase("call")&&!swr.equalsIgnoreCase("came")

&&!swr.equalsIgnoreCase("can")&&!swr.equalsIgnoreCase("come")&&!swr.equalsIgnoreCase("could")&&!swr.equalsIgnoreCase("did")

&&!swr.equalsIgnoreCase("do")&&!swr.equalsIgnoreCase("down")&&!swr.equalsIgnoreCase("first")&&!swr.equalsIgnoreCase("from")

&&!swr.equalsIgnoreCase("get")&&!swr.equalsIgnoreCase("go")&&!swr.equalsIgnoreCase("ad")&&!swr.equalsIgnoreCase("her")

&&!swr.equalsIgnoreCase("here")&&!swr.equalsIgnoreCase("if")&&!swr.equalsIgnoreCase("into")&&!swr.equalsIgnoreCase("just")

{

if(ppkey.equals(""))

{

ppkey=ppkey+swr;

}

else if(!ppkey.equals(""))

{

ppkey=ppkey+" "+swr;

}

}

}

catch(Exception e)

{

out.println(e);

}

finally {

out.close();

}

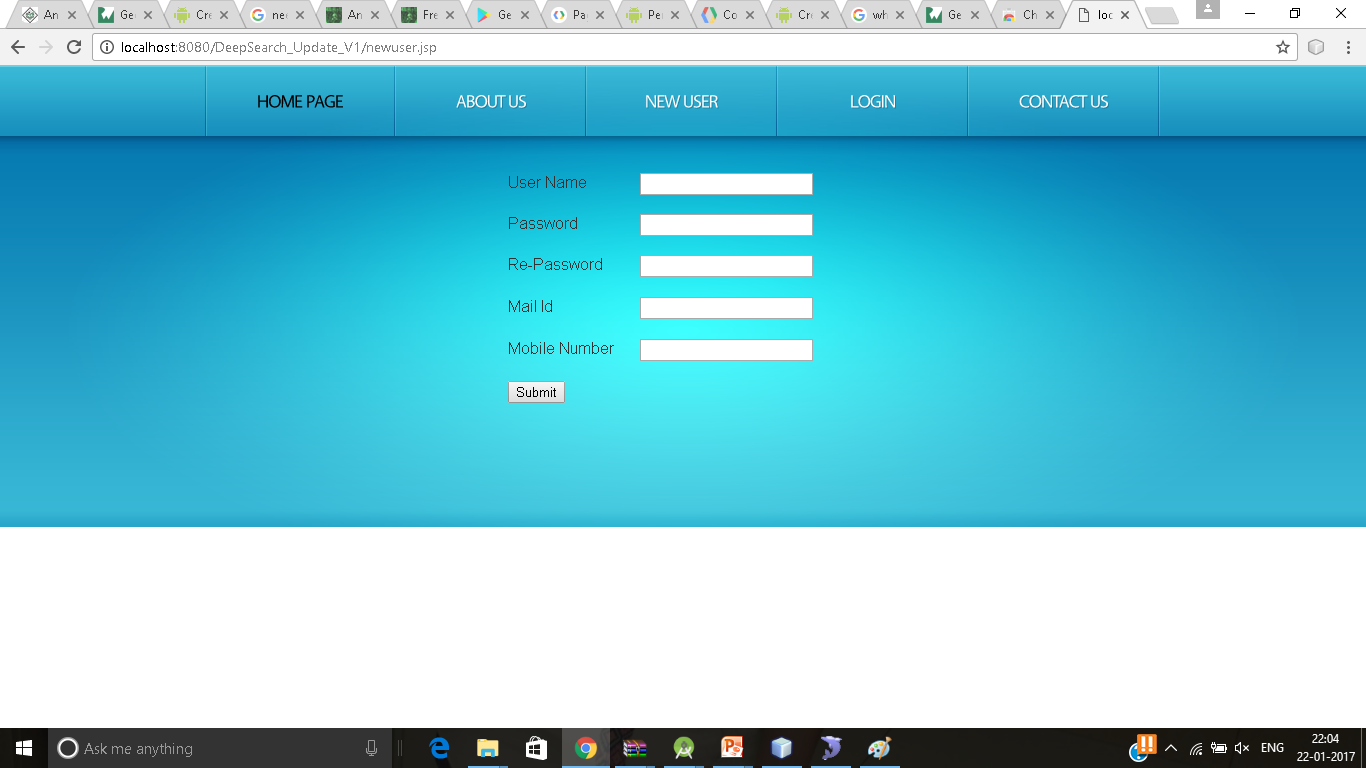
}

**6.2 SAMPLE SCREEN SHOTS**

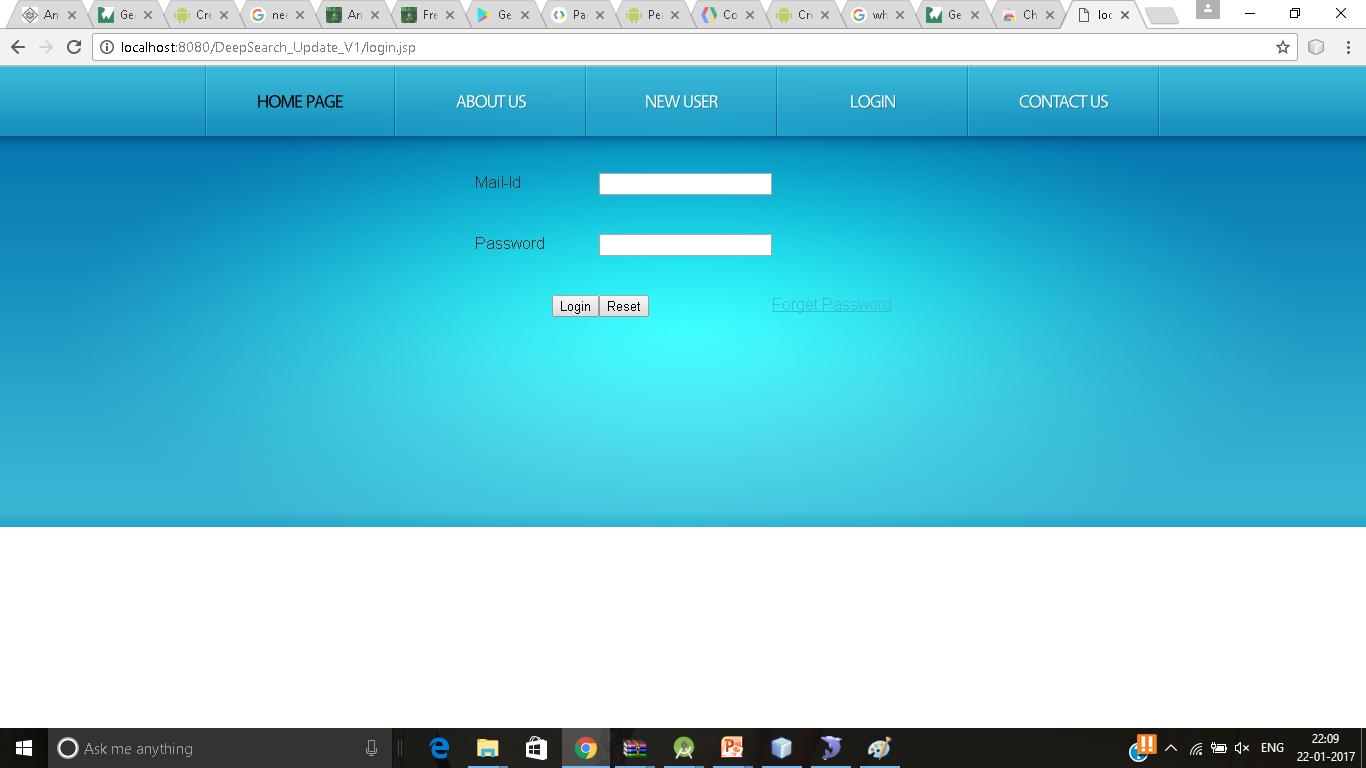
**6.2.1HOME PAGE:**

****

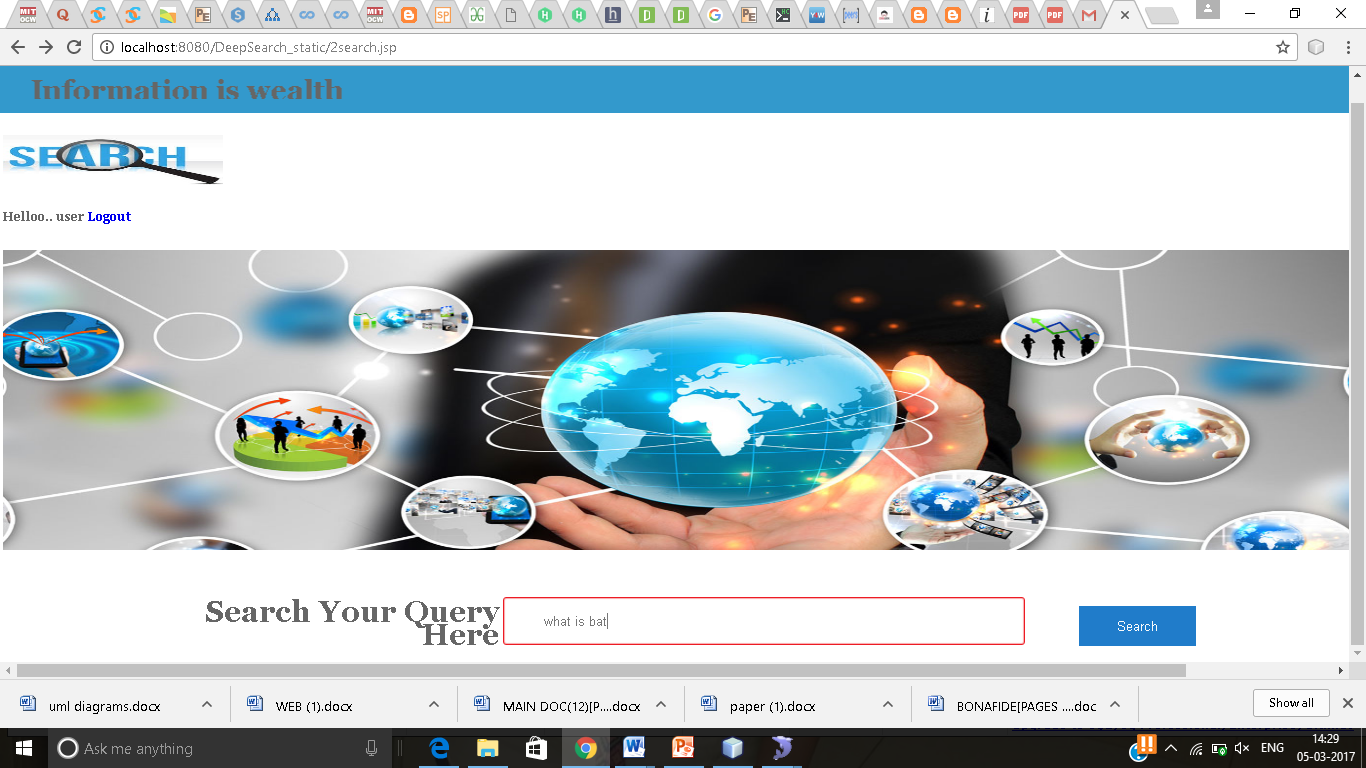
**6.2.2 REGISTRATION:-NEW USER**

****

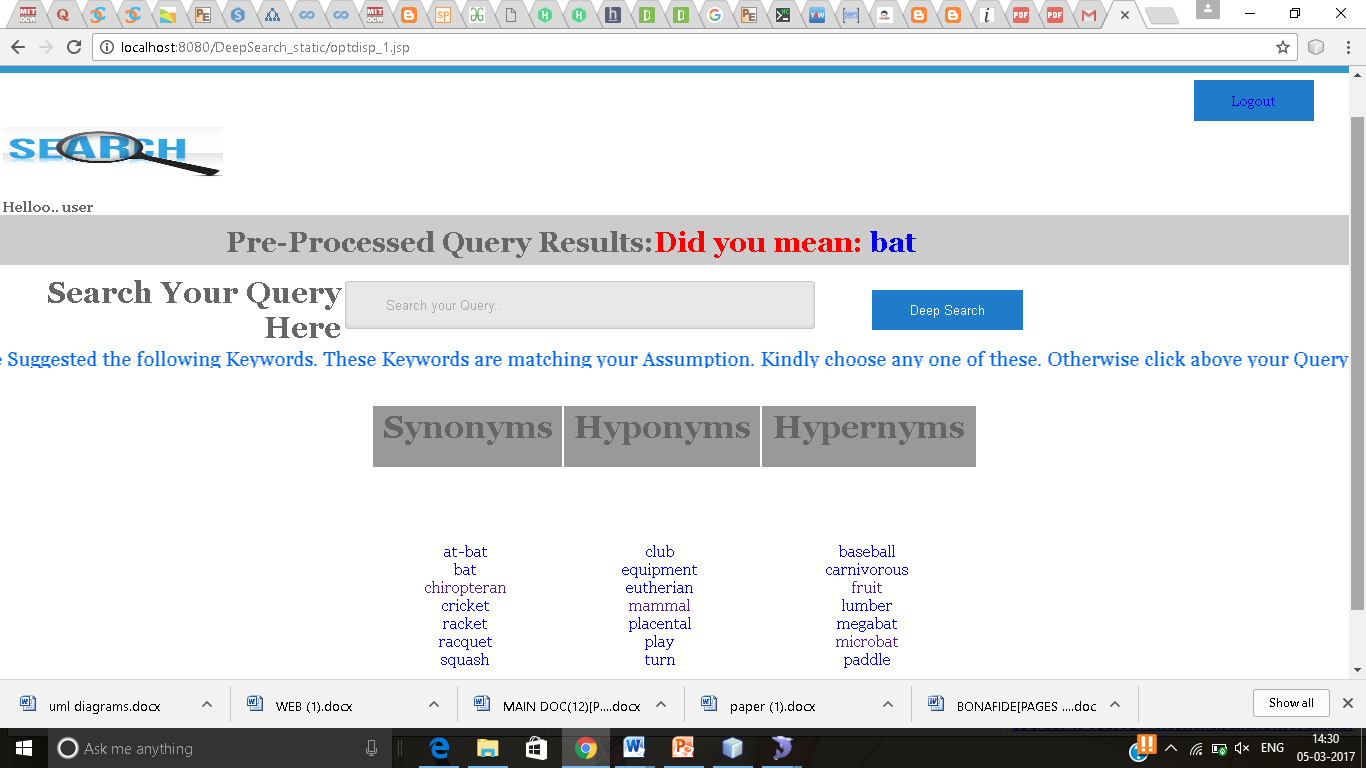
**6.2. 3LOGIN:**

****

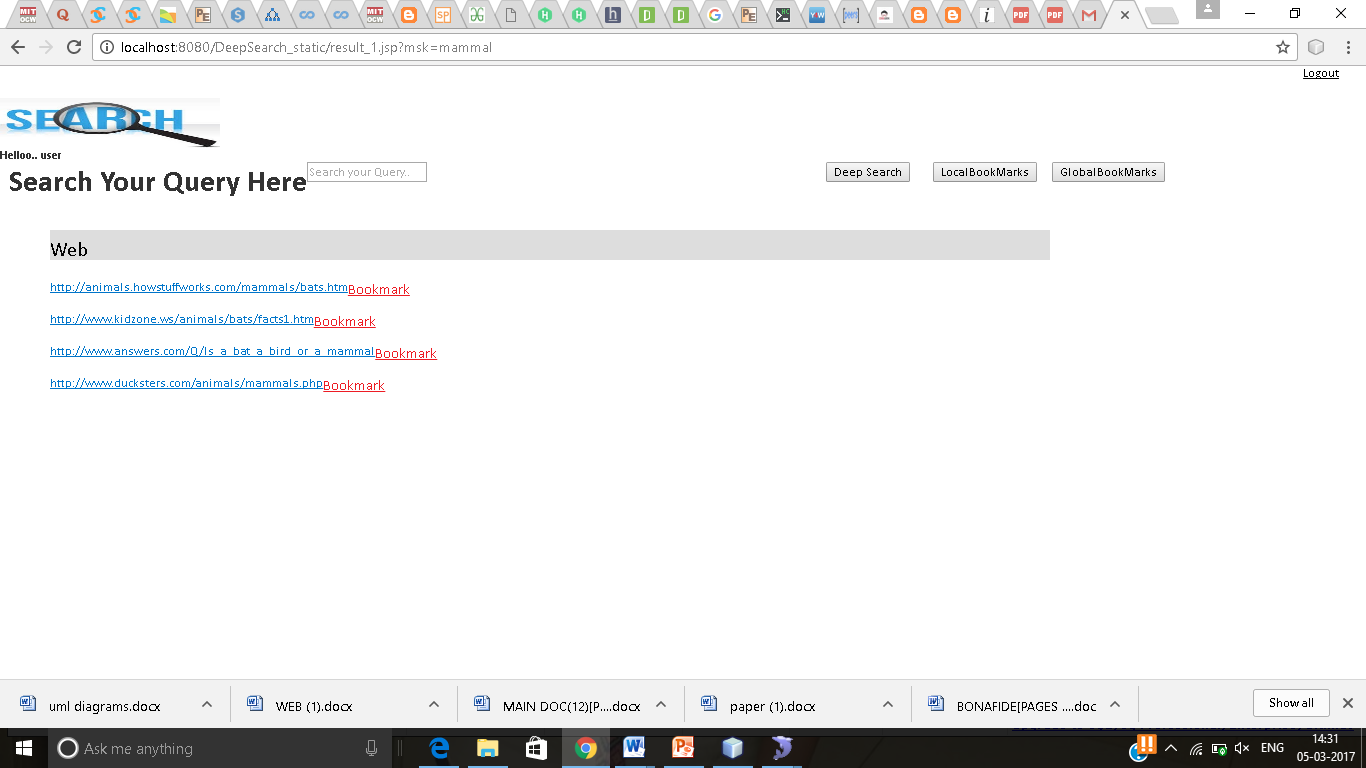
**6.2.4 SEARCH ENGINE:**



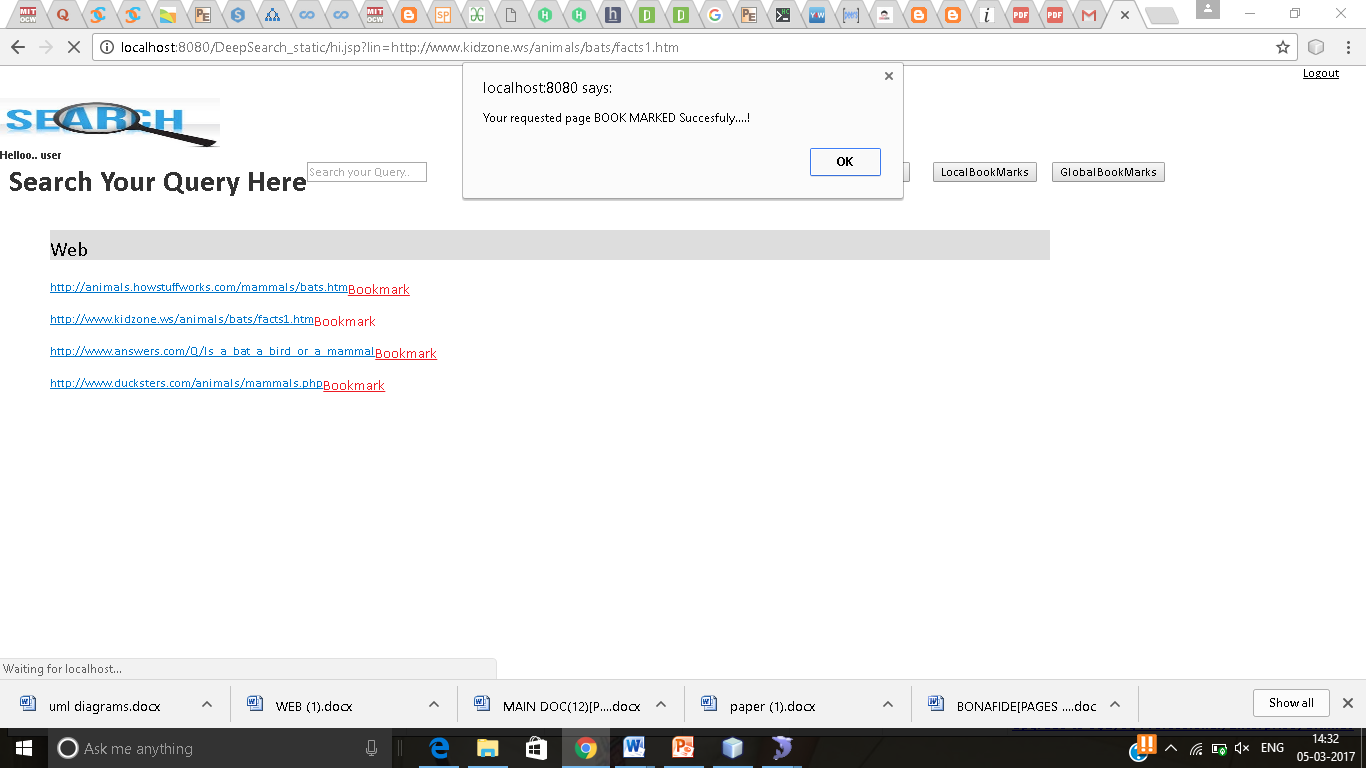
**6.2.5 EXPLORATORY SEARCH:**



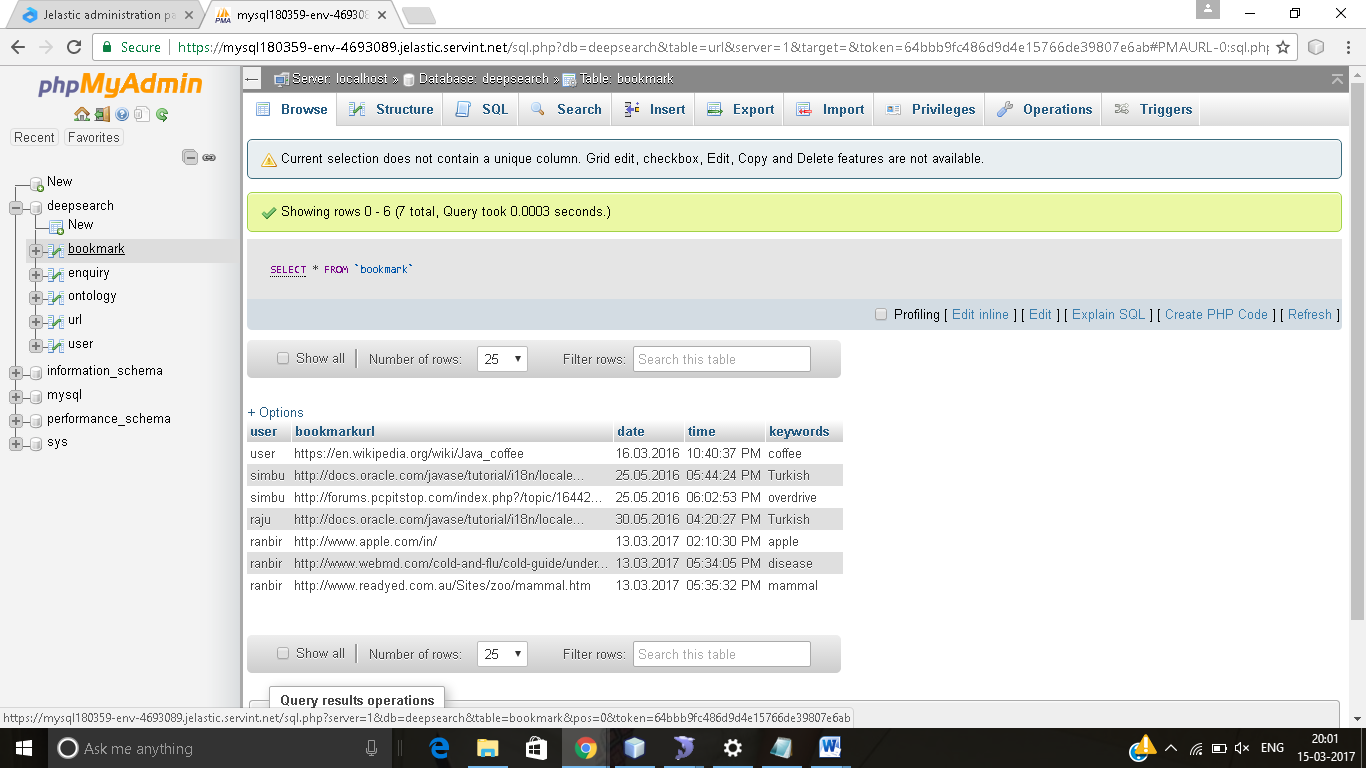
**6.2.6 WEBSITES:**



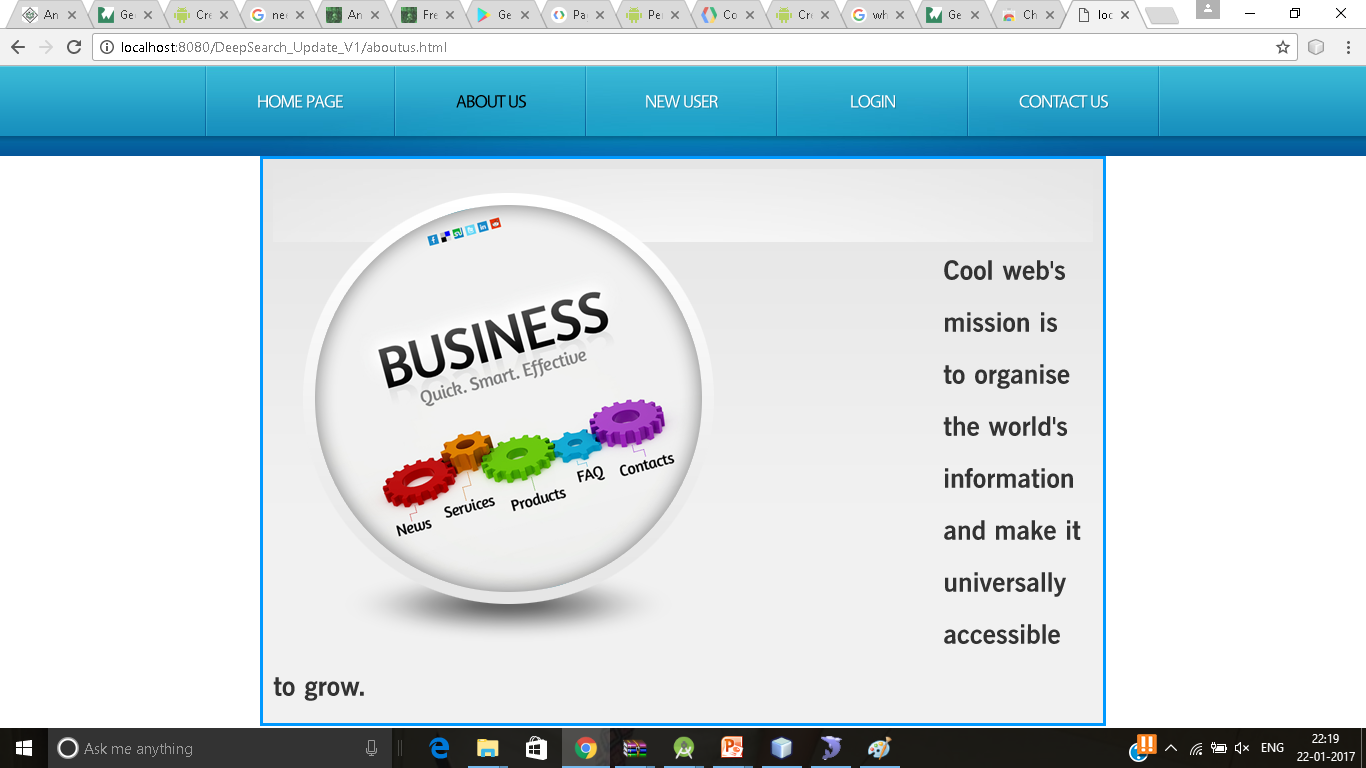
**6.2.7 BOOKMARKING:**



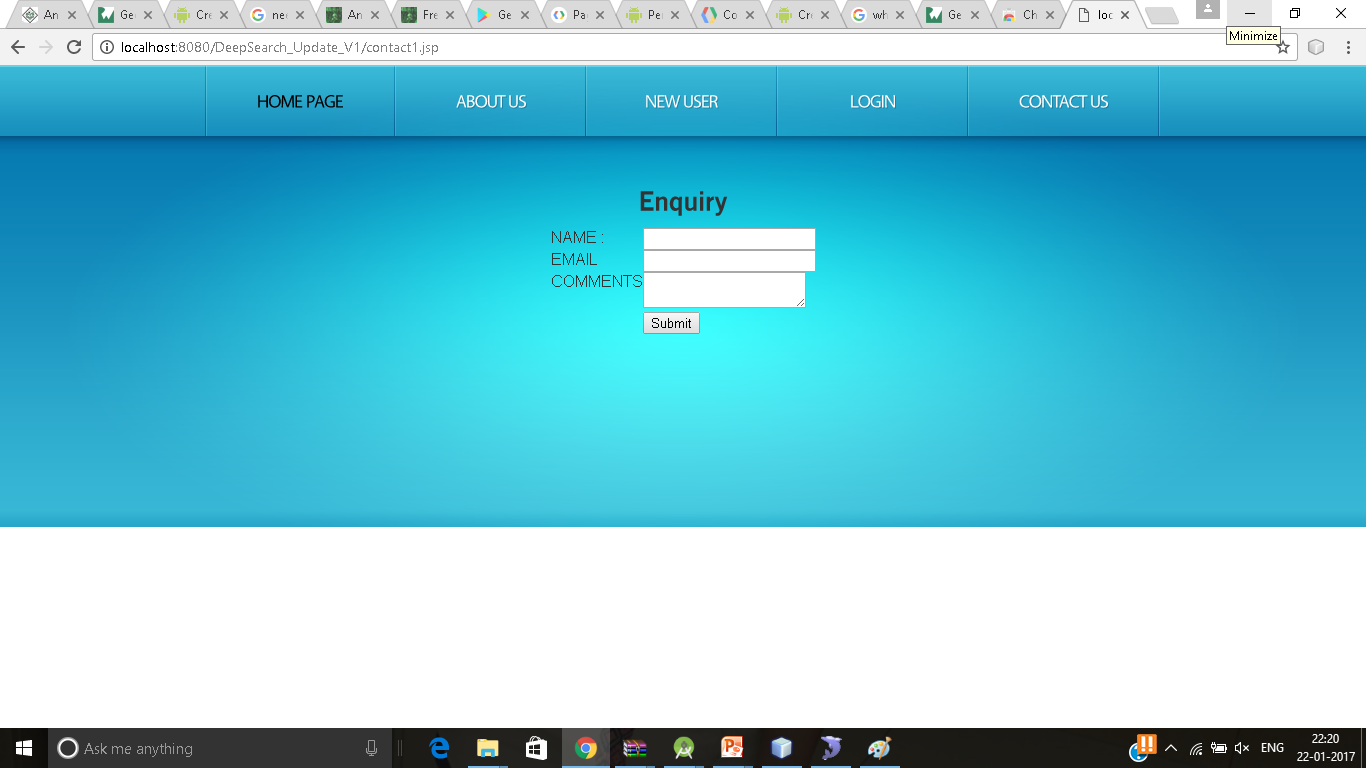
**6.2.8 BOOKMARKING MANAGER:**



**6.2.9 ABOUT US:**

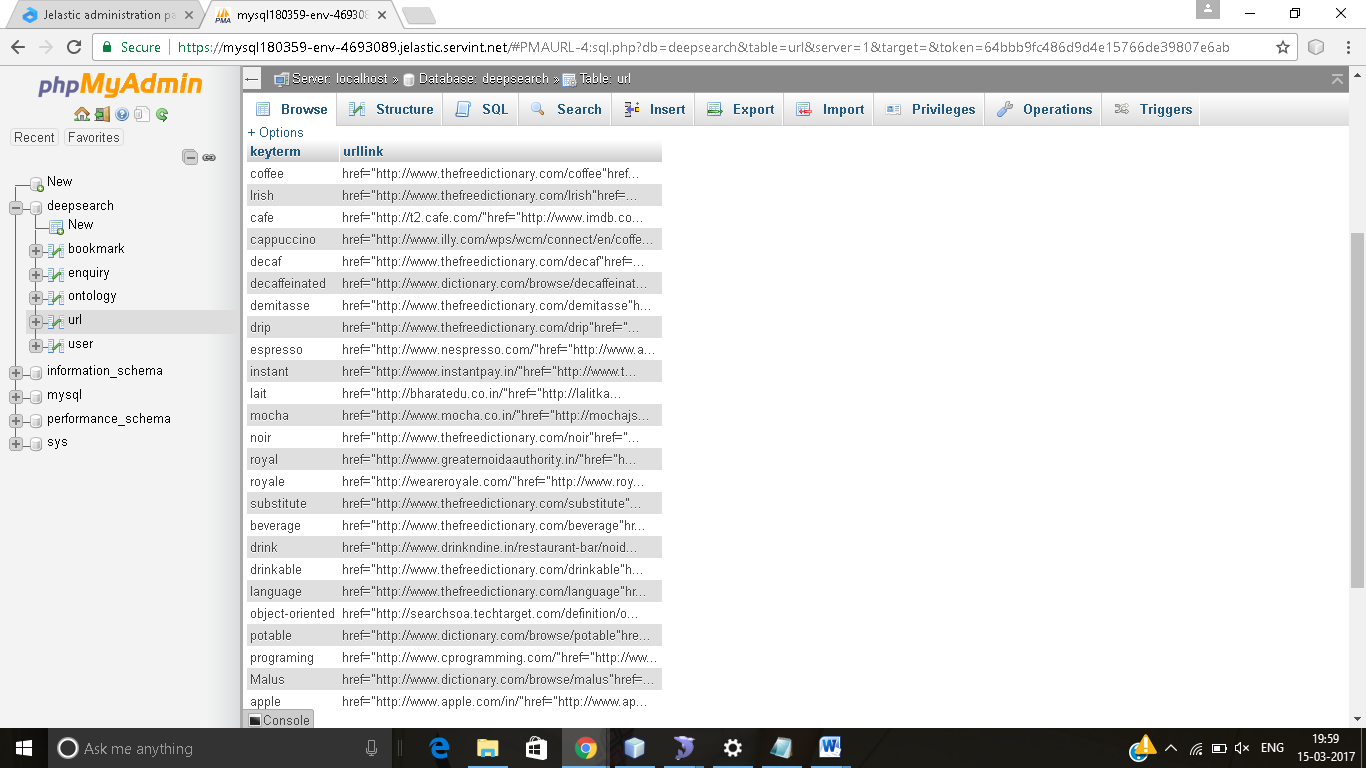
****

**6.2.10 ENQUIRY:**

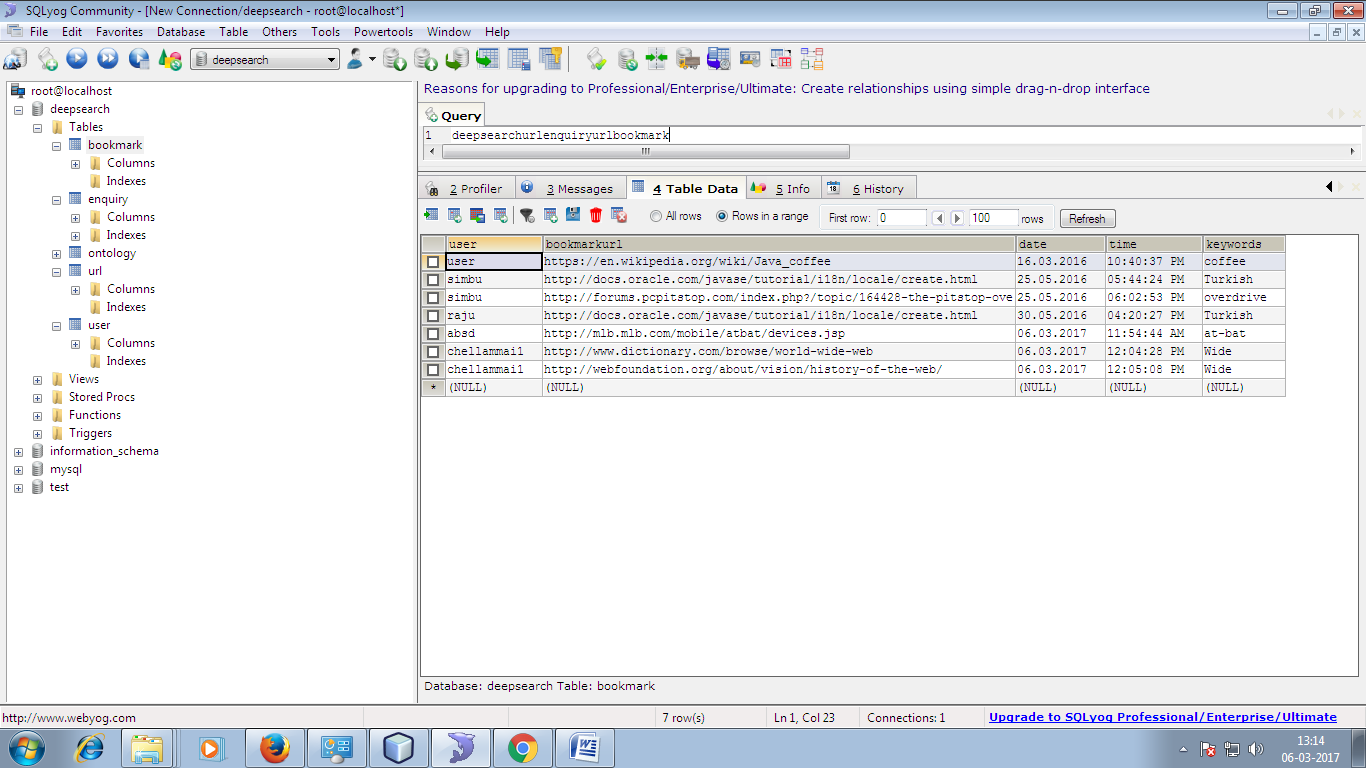
****

**6.3 SNAP SHOTS FOR DATABASE**

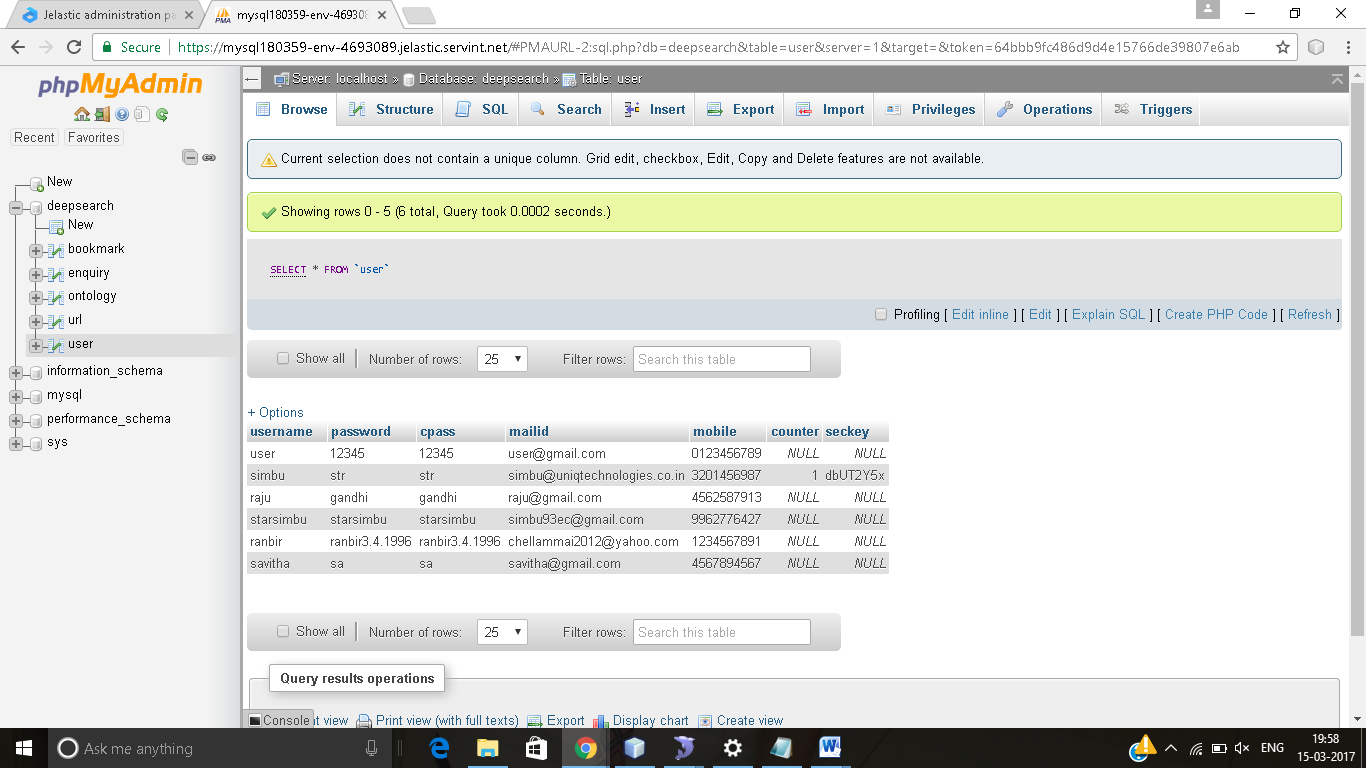
**6.3.1 SCREENSHOT FOR ONTOLOGY:**



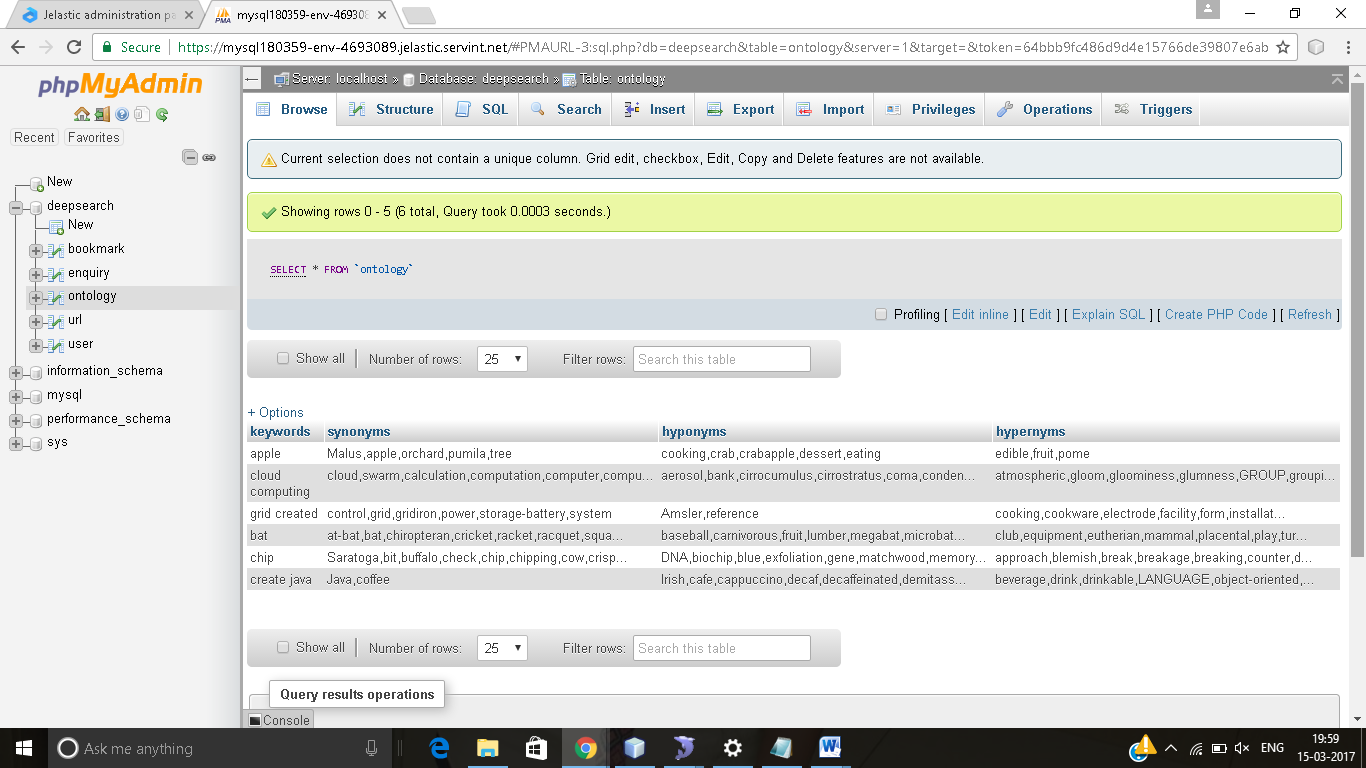
**6.3.2 SCREENSHOT FOR BOOKMARK DATABASE**



**6.3.3 SCREENSHOT FOR USER DATABASE**



**6.3.4 SCREENSHOT FOR KEYWORDS DATABASE**



**7. TESTING AND MAINTENANCE**

**7.1 TESTING OBJECTIVES**

Testing is a set of activities that can be planned in advance and conducted systematically. For this reason a template for software testing, a set of steps into which we can place specific test case design techniques and testing methods should be defined for software process. Testing often accounts for more effort than any other software engineering activity. If it is conducted haphazardly, time is wasted, unnecessary effort is expanded, and even worse, errors sneak through undetected.

**Type of Testing**

There are two type of testing according their behaviors

1. Unconventional Testing

572. Conventional Testing

**Unconventional Testing**

Unconventional testing is a process of verification which is doing by SQA team. It is a prevention technique which is performing from begging to ending of the project development. In this process SQA team verifies project development activities and insuring that developing project is fulfilling the requirement of the client or not. In this testing the SQA team follows these methods:

1. Peer review

2. Code walk and throw

3. Inspection

4. Document Verification

**Conventional Testing**

Conventional Testing is a process of finding the bugs and validating the project. Testing team involves in this testing process and validating that developed project is according to client requirement or not. This process is a correction technique where testing team find bugs and reporting to the development team for correction on developed project built.

**7.2 TEST CASE DESIGN**

**Unit Testing**

The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect. Each unit is tested separately before integrating them into modules to test the interfaces between modules. Unit testing has proven its value in that a large percentage of defects are identified during its use. In the company as well as seeker registration form, the zero length username and password are given and checked.

Also the duplicate username is given and checked. In the job and question entry, the button will send data to the server only if the client side validations are made. The dates are entered in wrong manner and checked. Wrong email-id and web site URL (Universal Resource Locator) is given and checked.

**Integration Testing**

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus the system testing is a confirmation that all is correct and an opportunity to show the user that the system works.

**Validation Testing**

The final step involves Validation testing, which determines whether the software function as the user expected. The end-user rather than the system developer conduct this test most software developers as a process called “Alpha and Beta Testing” to uncover that only the end user seems able to find. The compilation of the entire project is based on the full satisfaction of the end users. In the project, validation testing is made in various forms. In question entry form, the correct answer only will be accepted in the answer box. The answers other than the four given choices will not be accepted.

**7.3 TESTING STRATEGIES**

A number of software testing strategies have been proposed in the literature. All provide the software developer with a template for testing and all have the following generic characteristics:

1. Testing begins at the component level and works “outward” toward the integration of the entire computer-based system.
2. Different testing techniques are appropriate at different points in time.
3. The developer of the s/w conducts testing and for large projects, independent test group.

**7.3.1 Integration Testing:**

The strategies for integrating software components into a functioning product include the bottom-up strategy, the top-down strategy and to ensure that modules will be available for integration into the evolving software product when needed. The integration strategy dictates the order in which modules must be available and thus exerts a strong influence on the order in which modules are written, debugged and unit tested.

**7.3.2 White Box Testing:**

It is just the vice versa of the Black Box testing. There we do not watch the internal variables during testing. This gives clear idea about what is going on during execution of the system. The point at which the bug occurs were all clear and were removed.

**7.3.3 Black Box Testing:**

In this testing we give input to the system and test the output. Here we do not go for watching the internal file in the system and what are the changes made on them for the required output.

**7.3.4 Interface Testing**

The Interface Testing is performed to verify the interfaces between sub modules while performing integration of sub modules aiding master module recursively.

**7.3.5 Module Testing**

Module Testing is a process of testing the system, module by module. It includes the various inputs given, outputs produced and their correctness. By testing in this method we would be very clear of all the bugs that have occurred.

**7.3.6 Smoke testing**

Smoke testing refers to physical tests made to closed systems of pipes to test for leaks. By metaphorical extension, the term is also used for the first test made after assembly or repairs to a system, to provide some assurance that the system under test will not catastrophically fail. After a smoke test proves that "the pipes will not leak, the keys seal properly, the circuit will not burn, or the software will not crash outright”,] the system is ready for more stressful testing. The term smoke testing is used in several fields, including [electronics](http://en.wikipedia.org/wiki/Electronics),  [computer software](http://en.wikipedia.org/wiki/Computer_software) development,  [plumbing](http://en.wikipedia.org/wiki/Plumbing),  [woodwind](http://en.wikipedia.org/wiki/Woodwind)  repair,  [infectious disease](http://en.wikipedia.org/wiki/Infectious_disease) control, and the [entertainment industry](http://en.wikipedia.org/wiki/Entertainment_industry).

**7.4 Maintenance**

The objectives of this maintenance work are to make sure that the system gets into work all time without any bug. Provision must be for environmental changes which may affect the computer or software system.

This is called the maintenance of the system. Nowadays there is the rapid change in the software world. Due to this rapid change, the system should be capable of adapting these changes. In our project the process can be added without affecting other parts of the system. Maintenance plays a vital role.

The system will able to accept any modification after its implementation. This system has been designed to favor all new changes. Doing this will not affect the system’s performance or its accuracy. This is the final step in system life cycle. Here we implement the tested error-free system into real-life environment and make necessary changes, which runs in an online fashion.

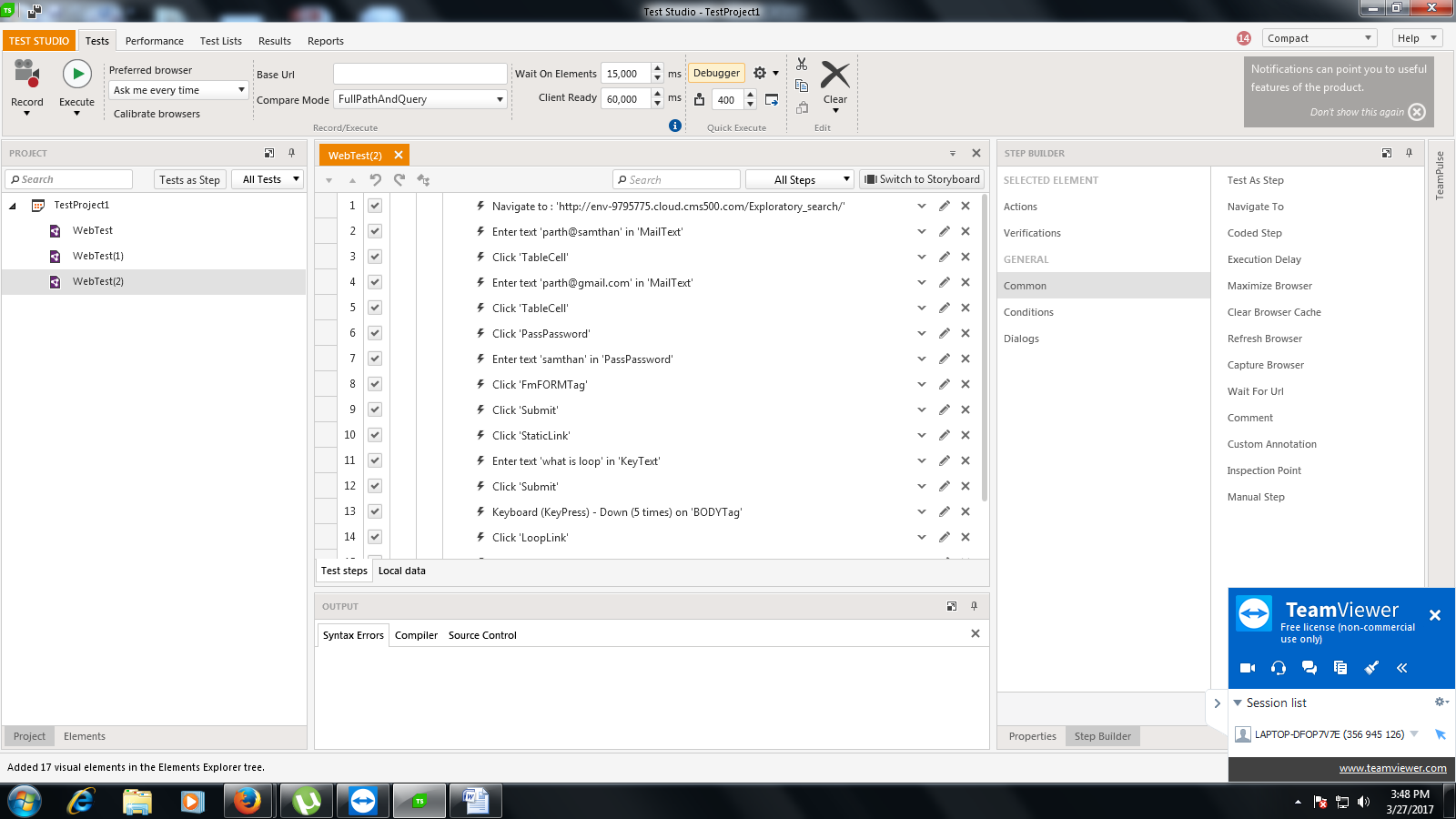
As a rule, system testing takes, as its input, all of the "integrated" software components that have successfully passed [integration testing](http://en.wikipedia.org/wiki/Integration_testing) and also the software system itself integrated with any applicable hardware system(s).

The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

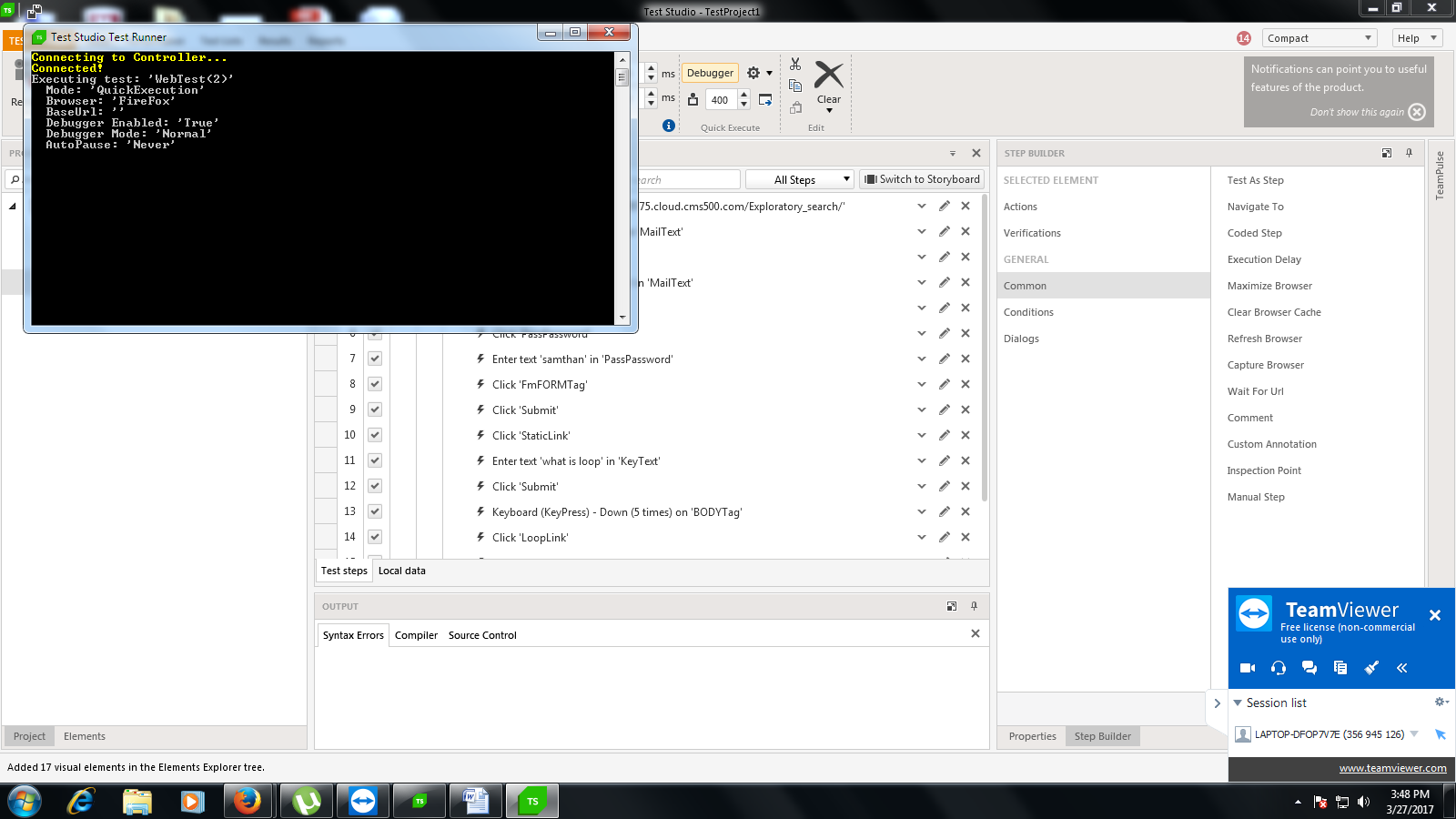
**7.5 SCREENSHOTS FOR TESTING**

Testing can be done using the TEST STUDIO software. **Telerik Test Studio** is a Windows-based software testing tool for web and desktop[[1]](https://en.wikipedia.org/wiki/Test_Studio#cite_note-desktop_applications-1) [functional testing](https://en.wikipedia.org/wiki/Functional_testing),[[2]](https://en.wikipedia.org/wiki/Test_Studio#cite_note-Functional_testing-2) [software performance testing](https://en.wikipedia.org/wiki/Software_performance_testing), [load testing](https://en.wikipedia.org/wiki/Load_testing)[[3]](https://en.wikipedia.org/wiki/Test_Studio#cite_note-load_testing-3) and [mobile application testing](https://en.wikipedia.org/wiki/Mobile_application_testing) developed by [Telerik](https://en.wikipedia.org/wiki/Telerik).[[4]](https://en.wikipedia.org/wiki/Test_Studio#cite_note-Telerik-4)[[5]](https://en.wikipedia.org/wiki/Test_Studio#cite_note-developed_by_Telerik-5) The tool ships with a plugin for Visual Studio and a standalone app that use the same [repositories](https://en.wikipedia.org/wiki/Source_code_repository) and file formats. Test Studio supports [HTML](https://en.wikipedia.org/wiki/HTML), [AJAX](https://en.wikipedia.org/wiki/AJAX), [Silverlight](https://en.wikipedia.org/wiki/Silverlight), [ASP.NET MVC](https://en.wikipedia.org/wiki/ASP.NET_MVC), [JavaScript](https://en.wikipedia.org/wiki/JavaScript) and [WPF](https://en.wikipedia.org/wiki/Windows_Presentation_Foundation). Test Studio supports test execution in [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer), [Firefox](https://en.wikipedia.org/wiki/Firefox), [Safari](https://en.wikipedia.org/wiki/Safari_%28web_browser%29) and [Chrome](https://en.wikipedia.org/wiki/Google_Chrome).

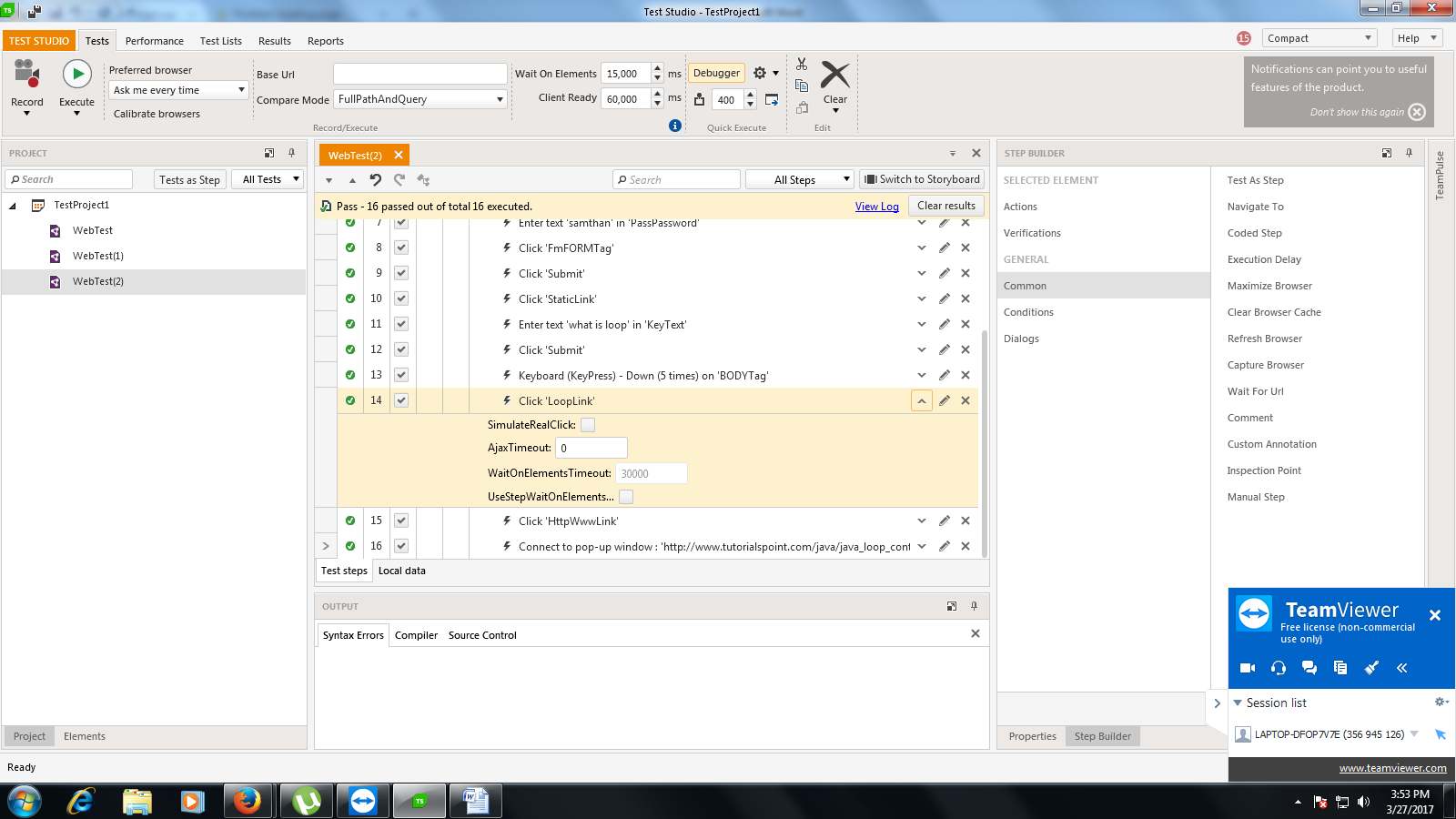
**7.5.1 SCREENSHOT FOR RECORDING**



**7.5.2 SCREENSHOT FOR TEST RUNNER**



**7.5.3 SCREENSHOT FOR SUCCESSFUL TEST CASES**



**7.6 TEST CASE TEMPLATE**

A test case can have the following elements. Note, however, that normally a test management tool is used by companies and the format is determined by the tool used.

**7.6.1 TEST CASE SAMPLE**

|  |  |
| --- | --- |
| Test Suite ID | TS001 |
| Test Case ID | TC001 |
| Test Case Summary | To verify that it retrieve the unaware fields. |
| Related Requirement | RS001 |
| Prerequisites | 1. User is authorized. 2. Query should be given |
| Test Procedure | 1. Enter the word to search. 2. Click Deep Search. |
| Test Data | 1. loop |
| Expected Result | **SYNONYMS**  [circuit](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=circuit)  [closed](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20closed)  [cringle](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20cringle)  [eyelet](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20eyelet)  [grommet](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20grommet)  [grummet](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20grummet)  [iteration](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20iteration)  [loop](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20loop)  [loop-the-loop](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20loop-the-loop)  [topology](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20topology)  **HYPONYMS**  [IUD](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=IUD)  [Jordan](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20Jordan)  [airplane](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20airplane)  [camp](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20camp)  [circle](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20circle)  [circuit](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20circuit)  [clique](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20clique)  [closed](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20closed)  [computer](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20computer)  [coterie](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20coterie)  [curve](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20curve)  [device](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20device)  [electric](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20electric)  [electrical](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20electrical)  [fastener](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20fastener)  [fastening](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20fastening)  [fingerprint](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20fingerprint)  [fixing](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20fixing)  [flight](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20flight)  [holdfast](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20holdfast)  [ingroup](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20ingroup)  [inner](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20inner)  [intrauterine](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20intrauterine)  [maneuver](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20maneuver)  [network](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20network)  [pack](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20pack)  [physical](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20physical)  [process](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20process)  [program](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20program)  [programme](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20programme)  [simple](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20simple)  [topology](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20topology)  **HYPERNYMS**  [band](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=band)  [belt](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20belt)  [bight](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20bight)  [circuit](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20circuit)  [drip](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20drip)  [inside](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20inside)  [lobe](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20lobe)  [loop](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20loop)  [noose](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20noose)  [outside](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20outside)  [parallel](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20parallel)  [running](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20running)  [shunt](http://env-9795775.cloud.cms500.com/Exploratory_search/result_1.jsp?msk=%20shunt) |
| Actual Result | 1. If the specified field is valid, the result is as expected. 2. If the specified field is invalid, nothing happens; the expected message is not displayed |
| Status | success |
| Remarks | This is a sample test case. |
| Test Environment | * OS: Windows Y * Browser: Chrome N |

**8. CONCLUSION AND FUTURE ENHANCEMENT**

**8.1 CONCLUSION**

Thus, we hereby conclude by saying that our project helps to search about unaware fields in an efficient manner. It is feasible for all kinds of user. Finding synonym, hypernym and hyponym for keywords by natural language processing is a challenging task as we have to eliminate irrelevant words and find only for similar useful words. Further to make stemming process more effective we defined a new combined strategy using existing strategies for successor stemming algorithm which is somewhat complex but gives high performance in retrieving relevant links.

**8.2 FUTURE ENHANCEMENT**

Information retrieval through images additional to keywords can be done in order to make searching more interactive and user friendly.We can make information retrieval more secured by displaying only websites which is of high security like https protocol websites alone. And for websites which is not secured we have to get security key through Kerberos mechanism. This way we can provide a secured search engine to user and hide vulnerable websites. Some highly advanced encryption algorithms can be used for links which gets user’s credentials or involving e-payment. Further we can also store data in hybrid cloud if it exceeds some threshold limit.

**9. REFERENCES**

[1] C. J. Jiang et al., “An indexing network model for information servicesand its applications,” in Proc. 6th IEEE Int. Conf. Serv. Orient.Comput.Appl., Koloa, HI, USA, 2013, pp. 290–297.

[2] M. T. Mills and N. G. Bourbakis, “Graph-based methods for natural language processing and understanding—A survey and analysis,”IEEE Trans. Syst., Man, Cybern., S

[3] H. Chim and X. Deng, “Efficient phrase-based document similarity for clustering,” IEEE Trans. Knowl. Data Eng., vol. 20, no. 9,pp. 1217–1229, Sep. 2008.yst., vol. 44, no. 1, pp. 59–71,Jan. 2014.

[4] L. Kaczmarek, “Interactive query expansion with the use of clustering-by-directions algorithm,” IEEE Trans. Ind. Electron., vol. 58,no. 8, pp. 3168–3173, Aug. 2011.

# [5] Li, J., Liu, C., Zhou “XML keyword search with promising result type recommendations”World Wide Web (2014) ,19January 2014, Volume 17, [Issue 1](http://link.springer.com/journal/11280/17/1/page/1), pp 127–159

[6] Zhu, H., Chen, E., Xiong“Ranking user authority with relevant knowledge categories for expert finding” World Wide Web (2014) September 2014, Volume 17, [Issue 5](http://link.springer.com/journal/11280/17/5/page/1), pp 1081–1107

[7][R.K. Roul](https://arxiv.org/find/cs/1/au:+Roul_R/0/1/0/all/0/1), [O. R. Devanand](https://arxiv.org/find/cs/1/au:+Devanand_O/0/1/0/all/0/1), [S.K. Sahay](https://arxiv.org/find/cs/1/au:+Sahay_S/0/1/0/all/0/1)“Web document clustering and ranking using tf-idf based apriori approach “IJCA Proceedings on ICACEA, No. 2, p. 34 (2014)

[8]HeikoPaulheim,”Exploiting Linked Open Data as Background Knowledge in Data Mining” ceur-ws.org/Vol-1082,(2013)

[9] [HoChungWu](http://dl.acm.org/author_page.cfm?id=81423595171&coll=DL&dl=ACM&trk=0&cfid=725108723&cftoken=30062602),”Interpretingtf-idf term weights as making relevance decisions “ACM Transactions on Information Systems (TOIS) ,Volume 26 Issue 3,June 2008 ,Article No.13

[10] M. Tenorth, A. C. Perzylo, R. Lafrenz, and M. Beetz, “Representation and exchange of knowledge about actions, objects, and environments in the RoboEarth framework,” IEEE Trans. Autom. Sci. Eng., vol. 10,no. 3, pp. 643–651, Jul. 2012.

**APPENDIX**

**PUBLICATION DETAILS**

**PAPER TITLE** EXPLORATORY SEARCH FOR RETRIEVING

UNAWARE FIELDS FOR USERS USING

ONTOLOGY CLUSTERING

**JOURNAL NAME** ISIRSET journals

**PUBLICATION** Ess and ess research publications

**MONTH AND YEAR** March 2016

**UNIVERSAL PAPER ID** IJ60303049