Search Engine Optimization

Enrollment No. : 101315

Name of Student: Paru Sharma

Supervisor's Name: Prof. Dr. Pardeep Kumar

May-2014
Submitted in partial fulfillment of the Degree of Bachelor of Technology

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY
JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
WAKNAGHAT, SOLAN (H.P)

Table of Content

Certificate

This is to certify that project report entitled “Search Engine Optimization”, submitted by Paru Sharma in partial fulfillment for the award of degree of Bachelor of Technology in Computer Science Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma

Date: Dr. Pardeep Kumar

Assistant Professor (Senior Grade)

Acknowledgement

Apart from the efforts of me, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of the project.

I would like to show my greatest appreciation to Mr. Pardeep Kumar. I can’t say thank you enough for this tremendous support and help. I feel motivated and encouraged every time everytime we attend his meeting. Without his encouragement and guidance this project would not have materialized.

The guidance and support received from all the members who contributed to this project, was vital for the success of this project. I am thankfull for their constant support and help.

Summary

In this project i have made a database. Two tables are created one is login and the other one is search. Relevent data is added to these tables. I have also made a website and have applied page rank using SQL queries in java code. A search engine was created. There is a code for the login details in which SQL query is inserted. When we will run this code a page will be displayed where we have to enter the username and password. If we enter the username of the admin then we can insert the rows in the tables and if we enter the username of a user then we can search for a particular address and if the initially the counter of that address is 0 so it will increment to 1 if we hit the link. The highest counter means it has the highest rank among all and is frequently visited.

List of figures:-

Chapter 1: Introduction

The web creates new challenges for information retrieval. The amount of information on the web is growing rapidly, as well as the number of new users inexperienced in the art of web research. People are likely to surf the web using its link graph, often starting with high quality human maintained indices such as Yahoo! or with search engines. Human maintained lists cover popular topics effectively but are subjective, expensive to build and maintain, slow to improve, and cannot cover all esoteric topics. Automated search engine that rely on keyword matching usually return too many quality machines. To make matter worse, some advertisers attempts to gain people's attention by taking measures meant to mislead automated search engines. We are trying to build up a search engine which addresses many of the problems of existing systems. It will specially make heavy use of the additional structure present in hypertext to provide much higher quality search result.

Search engine technology has had to scale dramatically to keep up with the growth of the web. In 1994, one of the first web search engine, the World Wide Web Worm(WWWW) had an index of 110,000 web pages and web accessible documents. As of November 1997, the top search engines claim to index from 2 millions to 100 millions web documents. By the year 2000, a comprehensive index of the Web contained over a billion documents. At the same time, the number of queries search engines handle has grown incredibly too. In March and April 1994, the World Wide Web Worm received an average of about 1500 queries per day. In November 1997, Altavista claimed it handles roughly 20 million queries per day. With the increasing number of users on the web, and automated systems which querry search engines, it is likely that top search engines wile handle hundreds of millions of queries per day.

Creating a search engine which scales even to today's web presents many challenges. Fast crawling technology is needed to gather the web documents and keep them up to the date. Storage space must be used efficiently to store indices and optionally, the documents themselves. The indexing system must process hundreds of gigabytes of queries.

1.1. Need of Search Engine

Internet is the most popular medium to communicate with the visitors, customers and other businessmen for branding and promoting a business. Internet has changed the lives of all people of the world. Now everyone is depending on the internet to seek anything they need. It has become a very important part of our life.

Every day the number of internet users is increasing due to their own reasons. In such a world it becomes necessary for every business to market itself in the online world, as it is a popular medium that is used by almost everyone in the world. A business can get benefit by using internet as a medium to market the products and services it is offering.

Therefore all businesses are making efforts to build a large customer base by means of internet. For getting large customer base it is necessary to be indexed in the search engines like Google, Yahoo, Bing and MSN etc. as internet users search any information they need through search engines. So, for getting indexed in search engines, it is needed to make efforts like SEO that stands for Search Engine Optimization. Search Engine Optimization is a technique to improve the online visibility of a business by making its website to be appeared in the top results of search engines.

A business can hire an SEO firm to make its site user friendly and search engine friendly. If a site is user friendly then it increases the chances of being indexed in search engines. Therefore SEO should be included in a website.

1.2. History

During early development of the web, there was a list of web servers edited by Tim Berners-Lee and hosted on the CERN web server. One historical snapshot of the list in 1992 remains, but as more and more web servers went online the central list could no longer keep up. On the NCSA site, new servers were announced under the title "What's New!"

The very first tool used for searching on the Internet was Archie. The name stands for "archive" without the "v". It was created in 1990 by Alan Emtage, Bill Heelan and J. Peter Deutsch, computer science students at McGill University in Montreal. The program downloaded the directory listings of all the files located on public anonymous FTP (File Transfer Protocol) sites, creating a searchable database of file names; however, Archie did not index the contents of these sites since the amount of data was so limited it could be readily searched manually.

Soon after, many search engines appeared and vied for popularity. These included Magellan, Excite, Infoseek, Inktomi, Northern Light, and AltaVista. Yahoo! was among the most popular ways for people to find web pages of interest, but its search function operated on its web directory, rather than its full-text copies of web pages. Information seekers could also browse the directory instead of doing a keyword-based search.

Google adopted the idea of selling search terms in 1998, from a small search engine company named *goto.com*. This move had a significant effect on the SE business, which went from struggling to one of the most profitable businesses in the internet.

Search engines were also known as some of the brightest stars in the Internet investing frenzy that occurred in the late 1990s. Several companies entered the market spectacularly, receiving record gains during their initial public offerings. Some have taken down their public search engine, and are marketing enterprise-only editions, such as Northern Light. Many search engine companies were caught up in the dot-com bubble, a speculation-driven market boom that peaked in 1999 and ended in 2001.

By 2000, Yahoo! was providing search services based on Inktomi's search engine. Yahoo! acquired Inktomi in 2002, and Overture (which owned All the Web and AltaVista) in 2003. Yahoo! switched to Google's search engine until 2004, when it launched its own search engine based on the combined technologies of its acquisitions.

Microsoft first launched MSN Search in the fall of 1998 using search results from Inktomi. In early 1999 the site began to display listings from Looksmart, blended with results from Inktomi. For a short time in 1999, MSN Search used results from AltaVista were instead. In 2004, Microsoft began a transition to its own search technology, powered by its own web crawler (called msnbot).

Microsoft's rebranded search engine, Bing, was launched on June 1, 2009. On July 29, 2009, Yahoo! and Microsoft finalized a deal in whichYahoo! Search would be powered by Microsoft Bing technology.

Chapter 2: Search Engine Optimization

Search Engine Optimization refers to the collection of techniques and practices that allow a site to get more traffic from search engines (Google, Yahoo, Microsoft). SEO can be divided into two main areas: off-page SEO (work that takes place separate from the website) and on-page SEO (website changes to make your website rank better

Page Rank is a ranking system that previously was the foundation of the infamous search engine, Google. When search engines were first developed, they ranked all websites equally and would return results based only on the content and meta tags the pages contained. At the time, however, the Page Rank system would revolutionize search engine rankings by including one key factor: a site's authority.

To determine how important, or authoritative, a site was Google chose several big sites, such as cnn.com, dmoz.org, and espn.com. These sites were clear authorities, and Google figured that if these websites chose to link to another site (let's say site B), then site B would receive a piece of that site's authority. If site B were to link to another site (how about C), then site C would also receive a piece of authority, though much smaller.

Using this system of passing authority, Google would then count up how much authority a site had and give it a Page Rank from 0 to 10. The Page Rank system has become more complicated since then, but this is how it all started.

Who uses page rank?
When Page Rank first came out, only Google was using the technology, but as other search engines have seen how much it improved Google's accuracy, nearly every search engine has added the Page Rank system in to be at least part of their algorithm. In the past, while many of the search engines were still working on adding Page Rank to their search algorithm, some couldn't wait to make their own and instead signed deals with Google to have them power their results (Yahoo did this for quite some time).
Apart from search engines, SEOs (Search Engine Optimization specialists), link buyers, webmasters, marketers, and anyone interested in a site's value will often look to the Google Page Rank when trying to quickly determine the importance of a site.

Importance:-
When Google was in its childhood, Page Rank was the single most important factor for ranking well. However, as soon as the SEO community caught on to this, there was a great deal of people who found ways to artificially boost their clients' Page Rank. Those sites became more authoritative than Google thought they should be. Since then, Google and other search engines have constantly refined how important Page Rank is, and its importance has definitely declined through the years.
One tactic Google uses is to update Google Toolbar Page Rank values four times a year instead of every week, making it difficult for SEOs to know a site's real Page Rank. Another tactic is to prevent a site that has been known to sell links from passing any of its Page Rank (authority) on to sites that it links to. However, Google can't use that tactic too much because then they run the risk of preventing good sites from being ranked as they should be.
The process of increasing your final thoughts: page rank
Although not nearly as important as it used to be, Page Rank can still be the deciding factor that bumps your site to the top of the search engines. Not only that, but it is also a good indicator of which sites you should spend your most time trying to get links from. Page Rank is directly tied to link acquisition. Link acquisition is getting links from other sites, be it via natural or through link purchasing. We cover both of these topics in greater detail, and you should read each lesson to learn more about the benefits and drawbacks of each.

First, we will provide a high level discussion of the architecture. Then we will look into the major application, i.e., search engine’s three distinct parts:

- A web crawler that finds and fetches web pages.
- The indexer that sorts every word on a every page and stores the resulting index of words in a huge database.

- The query processor, which compares your search query to the index and recommends the documents that it considers more relevant.

2.1 What is Search Engine Optimization?

- SEO Stands for **S**earch **E**ngine **O**ptimization.
- SEO is all about optimizing a web site for Search Engines.
- SEO is the process of designing and developing a web site to rank well in search engine results.
- SEO is to improve the volume and quality of traffic to a web site from search engines.
- SEO is a subset of search engine marketing.
- SEO is the art of ranking in the search engines.
- SEO is marketing by understanding how search algorithms work and what human visitors might search.

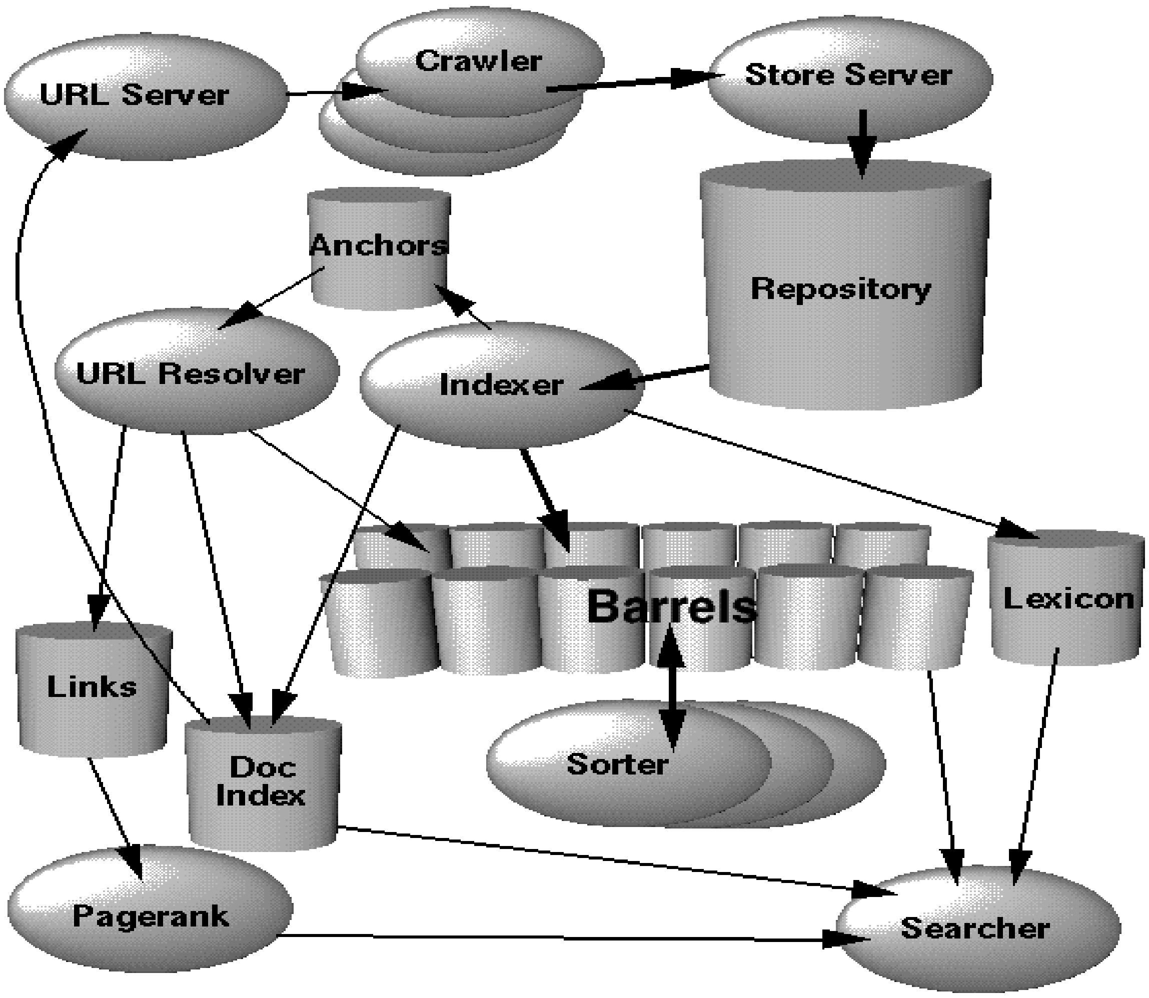


Fig 1: Basic Architecture of a Search Engine

The web crawling (downloading of web pages) is done by several distributed crawlers. There is a URL server that sends lists of URLs to be fetched to the crawlers. The web pages that are fetched are then sent to the store server. The store server then compresses and stores the web pages into a repository. Every web pages has an associated id number called a docID which is assigned whenever a new URL is parsed out of a web page. The indexing function is performed by the indexer and the sorter. The indexer performs a number of functions. It reads the repository, uncompresses the documents, and parses them. Each document is converted into a set of word occurrence called hits. The hits record the word, position in document, an approximation of font size, and capitalization. The indexer distributes these hits into a set of “barrels”, creating a partially sorted forward index. The indexer performs another important function. It parses out all the links in every web page and stores important information to determine where each link points from and to, and the text of the link. The URL resolver reads the anchors file and converts relative URLs into absolute URLs and in turn into docID. It puts the anchor text into the forward index, associated with the docIDs. The links database is used to compute PageRanks for all the documents. The sorter takes the barrels, which are sorted by docID, and resorts them by worded to generate the inverted index. This is done in place so that little temporary space is neede for this operation. The sorter also produces a list of wordIDs and offsets into the inverted index. A program called Dump Lexicon takes this list together with the lexicon produced by the indexer and generates a new lexicon to be used by the searcher. The searcher is run by a web server and uses the lexicon built by Dump Lexicon together with the inverted index and the PageRanks to answer queries.

Chapter 3: Tactics and Methods

SEO techniques are classified into two broad categories:

- Techniques that search engines recommend as part of good design referred to as White Hat SEO, and
- Techniques that search engines do not approve and attempt to minimize the effect of referred to as Black Hat or spamdexing.

3.1. White Hat SEO

- If it conforms to the search engine's guidelines.
- If it does not involves any deception.
- It ensures that the content a search engine indexes and subsequently ranks is the same content a user will see.
- It ensures that a Web Page content should have been created for the users and not just for the search engines.
- It ensures the good quality of the web pages
- It ensures the useful content available on the web pages.

3.2. Black Hat Or Spamdexing

- Try to improve rankings that are disapproved of by the search engines and/or involve deception.
- Redirecting users from a page that is built for search engines to one that is more human friendly.
- Redirecting users to a page that was different from the page the search engine ranked.
- Serving one version of a page to search engine spiders/bots and another version to human visitors. This is called **Cloaking** SEO tactic.
- Using Hidden or invisible text or with the page background color, using a tiny font size or hiding them within the HTML code such as "no frame" sections.
- Repeating keywords in the Meta tags, and using keywords that are unrelated to the site's content. This is called **Meta tag stuffing**.
- Calculated placement of keywords within a page to raise the keyword count, variety, and density of the page. This is called **Keyword stuffing** .
- Creating low-quality web pages that contain very little content but are instead stuffed with very similar key words and phrases. These pages are called **Doorway or Gateway Pages**
- Mirror web sites by hosting multiple web sites all with conceptually similar content but using different URLs.
- Creating a rogue copy of a popular web site which shows contents similar to the original to a web crawler, but redirects web surfers to unrelated or malicious web sites. This is called **Page hijacking**.

Chapter 4:- About Websites:-

Website: A **website** is a set of related web pages containing content such as text, images, video,audio, etc. A website is hosted on at least one web server, accessible via a network such as the Internet through an Internet address known as a Uniform Resource Locator (Domain name).

Websites can be classified majorly:

1. **As per their Functionality**
2. **As per their Purpose**

1. **Based on functionality:**

a) **Static website: A static website is one in which content does'nt changes or can not be changed without changing the source code.**

Features:

Content is static. i.e. the information does not change over time .
Easier to develop.
Can be created in no time.
Uses html & CSS (content styling sheets).

b) **dynamic website: A dynamic is one which generally comes with a CMS and content changes time to time and even a end user can upload or download some content.**

Features:

Content is dynamic. i.e. the information does change over time.
Takes longer time to develop.
Require a lot of coding.
Uses various languages like PHP, ASP.net & scripts.

c) **Responsive website: Responsive Web Design** (**RWD**) is an approach to web design in which a site is crafted to provide an optimal viewing experience—easy reading and navigation with a minimum of resizing, panning, and scrolling—across a wide range of devices (from desktop computer monitors to mobile phones)

2. **Based on purpose:**

a) **Personal websites:**
- **Resume sites**
- **Artwork sites**
- **Hobby sites**

b) **Informational websites:**
- **Blogs**
- **Directories**
- **Question and Answers**
- **Forums**
- **Wikis**
- **Portals**

c) **E-commerce websites:**
Electronic commerce websites, commonly known as **e-commerce websites**, are associated with buying and selling of products or services over internet

d) **Social networking websites:** A site where users could communicate with one another and share media, such as pictures, videos, music, blogs, etc. with other users. These may include games and web applications.

e) **Business/corporate websites:** A **corporate website** or **corporate site** is an website operated by a business or other private enterprise such as a charity or nonprofit foundation. Corporate sites differ from electronic commerce, portal, or sites in that they provide information to the public about the company rather than transacting or providing other services.

Keywords Planning & Research:
Keywords are words that are used to search information in Search Engines that refers to a specific topic. Keywords can vary like Singular Keywords, Plural keywords, Strong & Weak keywords, Global & Local keywords, Misspelled keywords, Seasonal keywords and Long tail keywords.

(1) **Singular Keywords:**

Form of a word that is used to search a single piece of product or niche information. Singular Keywords had to be used according to the need and generally Keyword formation should be smart.

For example, a singular keyword can be something like - book and can't be like - Cricket shoe. People will search for shoes and not shoe.

(2) Plural keywords:

Form of a word that is used to search more than one specific object or range of information. Compared to the above example Plural keywords can be used like - Cricket shoes and it is meaningful and the simple difference between the singular and the plural is an “s” or “es” at the end of the word.

Now a days, Google and other major search engines are smart enough to find the relationship between plural and singular. Google also index and values singular keywords for some of the targeted plural keywords based on the people search behavior. That is, users who search for the plural keywords may still get benefit out of web pages that focus the singular keywords.

(3) Strong Keywords:

Strong keywords have high volume of search and it takes up from single word to even some Geo specific keywords. This depends entirely on the user volume those who are looking for the information. Promoting web page for strong keywords is not so easy as the competition would be much high and as well will have large number of results.

Here few examples for Strong Keywords:

* Apple ipad
* Toyota

(4) Weak Keywords:

Weak Keywords are low volume keywords but it can produce a decent reference for business enquiries. Normally more than two or more words search keyword comes under weak keywords criteria. For creating week keywords, you can try to add your location to strong keywords and measure the search volume using a keyword tool.

This will work for many industries and you can obtain a decent traffic for start and gradually you can focus strong keywords. There are even week keywords that are not location specific since those keywords might have less competition and search results based on industries you target.

(5) Global Keywords:

Global Keywords are not location specific and most of the keywords are targeted for Dot com search engines and the importance of site listing for Global keywords are not restricted to regional specific search engines.

(6) Local Keywords:

Local Keywords are location specific and the keywords can be targeted for regional search engines. Normally local keywords are specific to your languages and targeted country.

Example for local keywords

* Web design London
* SEO New York

(7) Misspelled Keywords:

Misspelled Keywords are formed when huge volume of search engine users have problem with spellings and make typo mistakes while searching for information. And also some spell different and it depends on country. In US the spell is 'color' and in Australia it is written as 'Colour'

One good example is 'accommodation' word. Some type as 'accomodation' and 'accomadation'

Keyword research: Keyword research is a process that tells you, what all people are searching on internet. How these searches are are being performed with respect to region, numbers, match types, behaviour & trends.

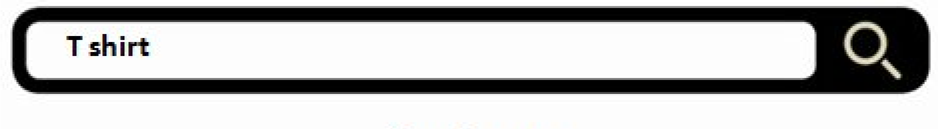
While searching for keywords we take care of:

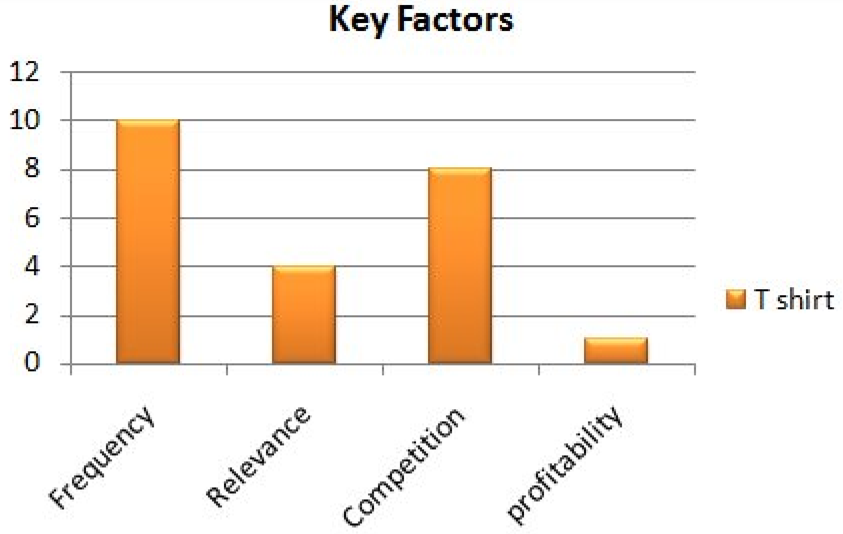
- Frequency
- Relevance
- Competition
- Profitability

Let’s take a case:

Suppose, we want to sell t shirt online so what can be our keyword

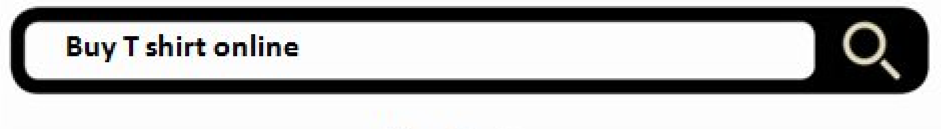
a) T shirt : So here is how is key factors will be for term T shirt.

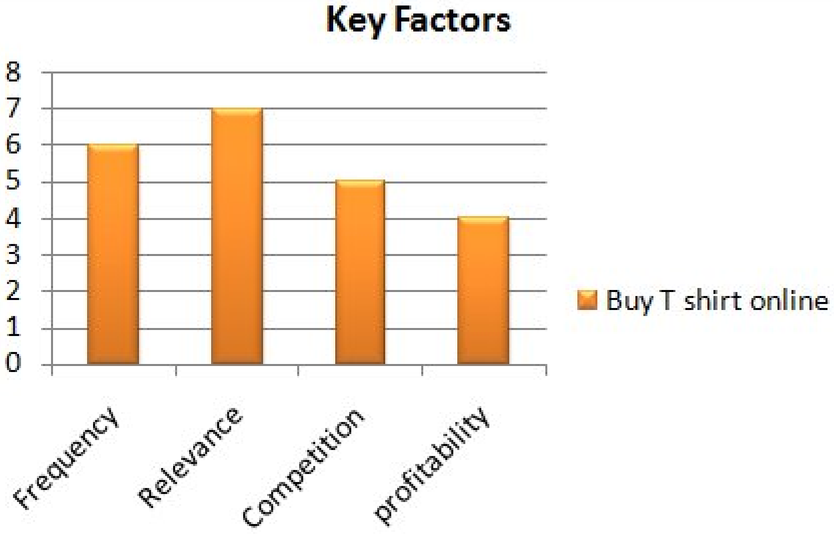




The term “T shirt” will have great search volume i.e. It will be searched a lot but when we take a look at it’s relevance and profitability it will be very less as it’s more of a generic term and competing results in search engine will also be high for it. So what could be a better keyword.

b) Buy T shirts Online:

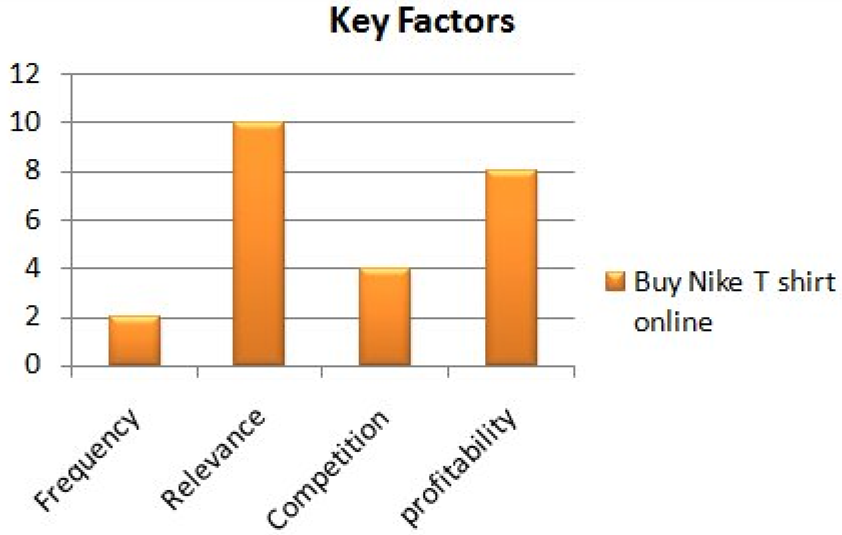




The term “Buy T shirt” will have a greater relevance and a higher chance of conversion although frequency of search volume would have been reduced and competiton would still be a bit stiff. So is their a possibility of further refining? Yes.

c) **Buy Nike T shirt Online:** Bingo,This could be something that could directly lead to a sale online. .

Buy Nike T shirt online



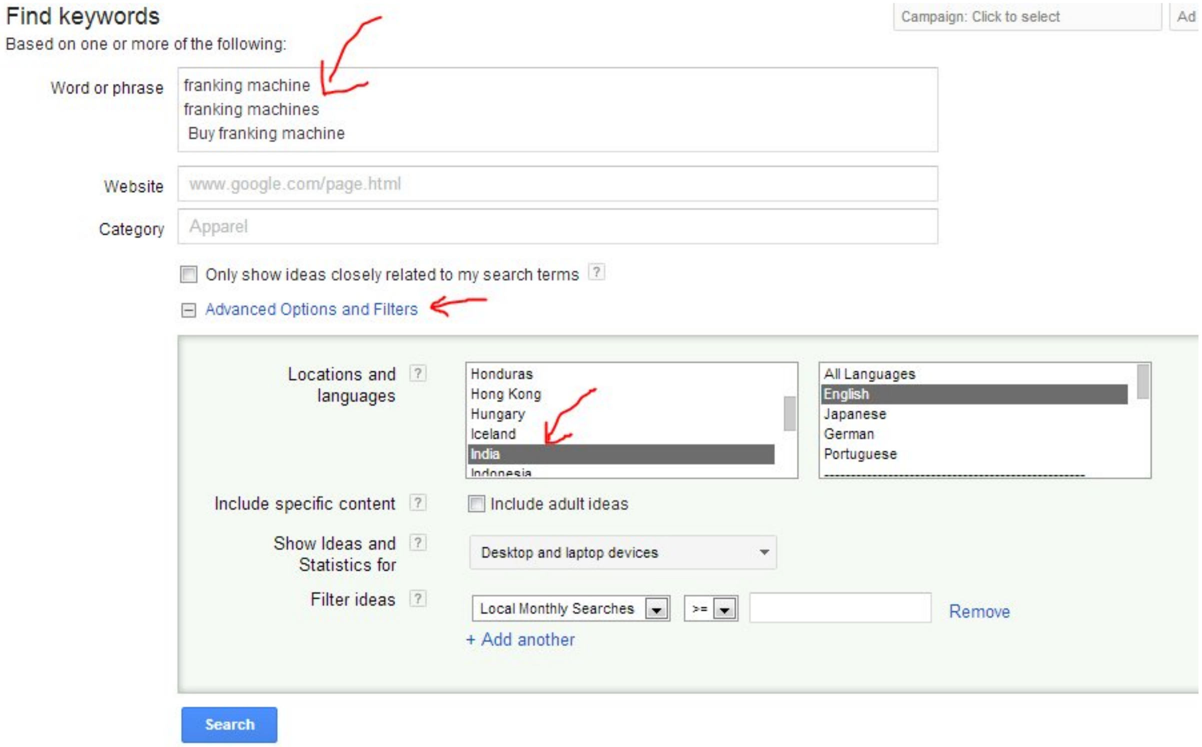
So all in all, there is no secret strategy for keyword research as every business have different needs. But, the best way that it can work for you will be to get a thinking cap of your prospect and think what you might have searched for… if you would have been at his place.

Keyword Selection: So how to select your keywords?
To select your keywords you can use :

1. **Google Keyword tool (Free)**
2. **Wordtracker (Paid)**

Let’s understand keyword tool:
This will help you to broaden up your options with suggestions on typical keywords.
Also it will provide you Advanced options & filters for:

1. Location
2. Languages
3. Devices
4. Match types
5. Keyword search trends over the entire year.



Find keywords

Based on one or more of the following:

Word or phrase: franking machine
franking machines
Buy franking machine

Website: www.google.com/page.html

Category: Apparel

□ Only show ideas closely related to my search terms

Advanced Options and Filters

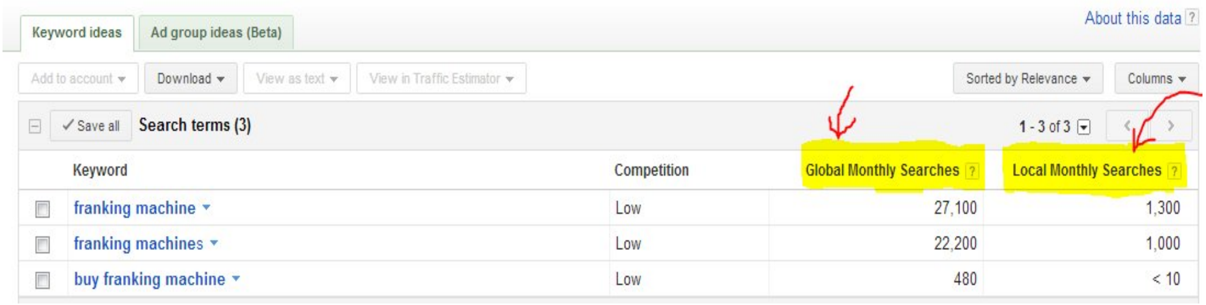
Locations and languages: India; English

Include specific content: □ Include adult ideas

Show Ideas and Statistics for: Desktop and laptop devices

Filter ideas: Local Monthly Searches >=

So here is your result for your search terms franking machine, franking machines & buy franking machine.

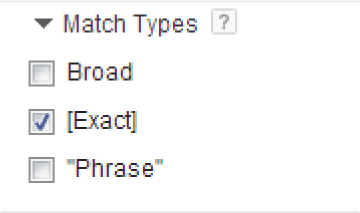


Search terms (3)

Keyword	Competition	Global Monthly Searches	Local Monthly Searches
franking machine	Low	27,100	1,300
franking machines	Low	22,200	1,000
buy franking machine	Low	480	< 10

Global monthly searches tell you how many times this word is search globally and local monthly searches will tell you how many times this word is searched in "India" which you already selected as a location.

But does that mean there are1300 search for franking machine monthly in India. Probably not. Why? Because by default keyword tool gives you **broad match** to get exact results you need to select **Exact Match. (Column will be available in top left of keyword tool)**



Match Types ?

□ Broad

☑ [Exact]

□ "Phrase"

Confused, OK lets understand match types first.

Broad match: The default matching option, broad match means that your ad may show if a search term contains your keyword terms in any order, and possibly along with other terms. Your ads can also show for singular or plural forms, synonyms, stemmings (such as *floor* and *flooring*), related searches, and other relevant variations. Sticking with the broad match default is a great choice if you don't want to spend a lot of time building your keyword lists and want to capture the highest possible volume of ad traffic.

Example

Broad match keyword	Searches might be for
tennis shoes	buy tennis shoes best shoes for tennis tennis shoe laces running shoes tennis sneakers

Phrase match: With phrase match, your ad can show when someone searches for your exact keyword, or your exact keyword with additional words before or after it. We'll also show your ad when someone searches for close variants of that exact keyword, or with additional words before or after it. Close variants include misspellings, singular and plural forms, acronyms, stemmings (such as *floor* and *flooring*), abbreviations, and accents. Using phrase match can help you reach more customers, while still giving you more precise targeting. In other words, your keywords are less likely to show ads to customers searching for terms that aren't related to your product or service.

To use a phrase match keyword, simply surround the words you want matched with quotation marks. Since we'll automatically show your ads for close variants in your new and existing campaigns, there's no need to separately add variants of your keyword.

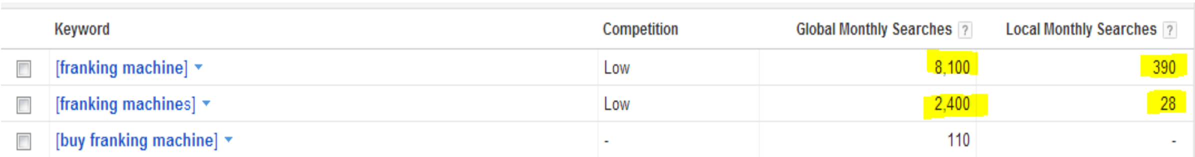
Example

Phrase match keyword	Searches might be for	Searches won't include
"tennis shoes"	red tennis shoes red tenis shoes	shoes for tennis tennis sneakers

Exact match: With exact match, your ads can appear when someone searches for your exact keyword, without any other terms in the search. We'll also show your ad when someone searches for close variants of that specific keyword. Close variants include misspellings, singular and plural forms, acronyms, stemming (such as floor and flooring), abbreviations, and accents. The difference between exact match and phrase match is that if someone enters additional

words before or after the keyword, your ad won't show. Using exact match means that your keywords are targeted more precisely than broad match or phrase match.
To use an exact match keyword, simply surround the words you want matched with brackets. Since we'll automatically show your ads for close variants in your new and existing campaigns, there's no need to separately add other variants of your keyword.
Example

Exact match keyword	Searches might be for	Searches won't include
[tennis shoes]	tennis shoes tenis shoes	red tennis shoes buy tennis shoes



	Keyword	Competition	Global Monthly Searches ?	Local Monthly Searches ?
□	[franking machine] ▾	Low	8,100	390
□	[franking machines] ▾	Low	2,400	28
□	[buy franking machine] ▾	-	110	-

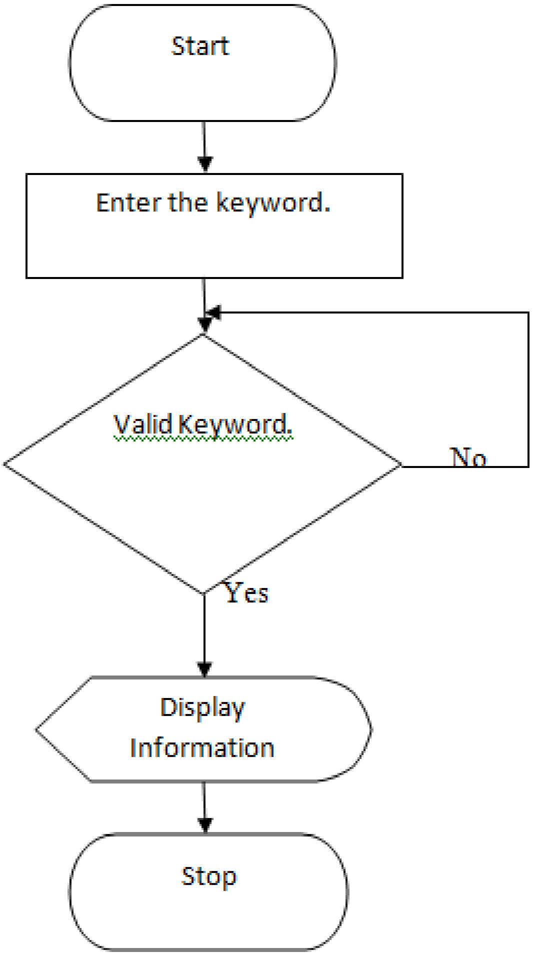
Fig: Updated result after exact match check box enabled.

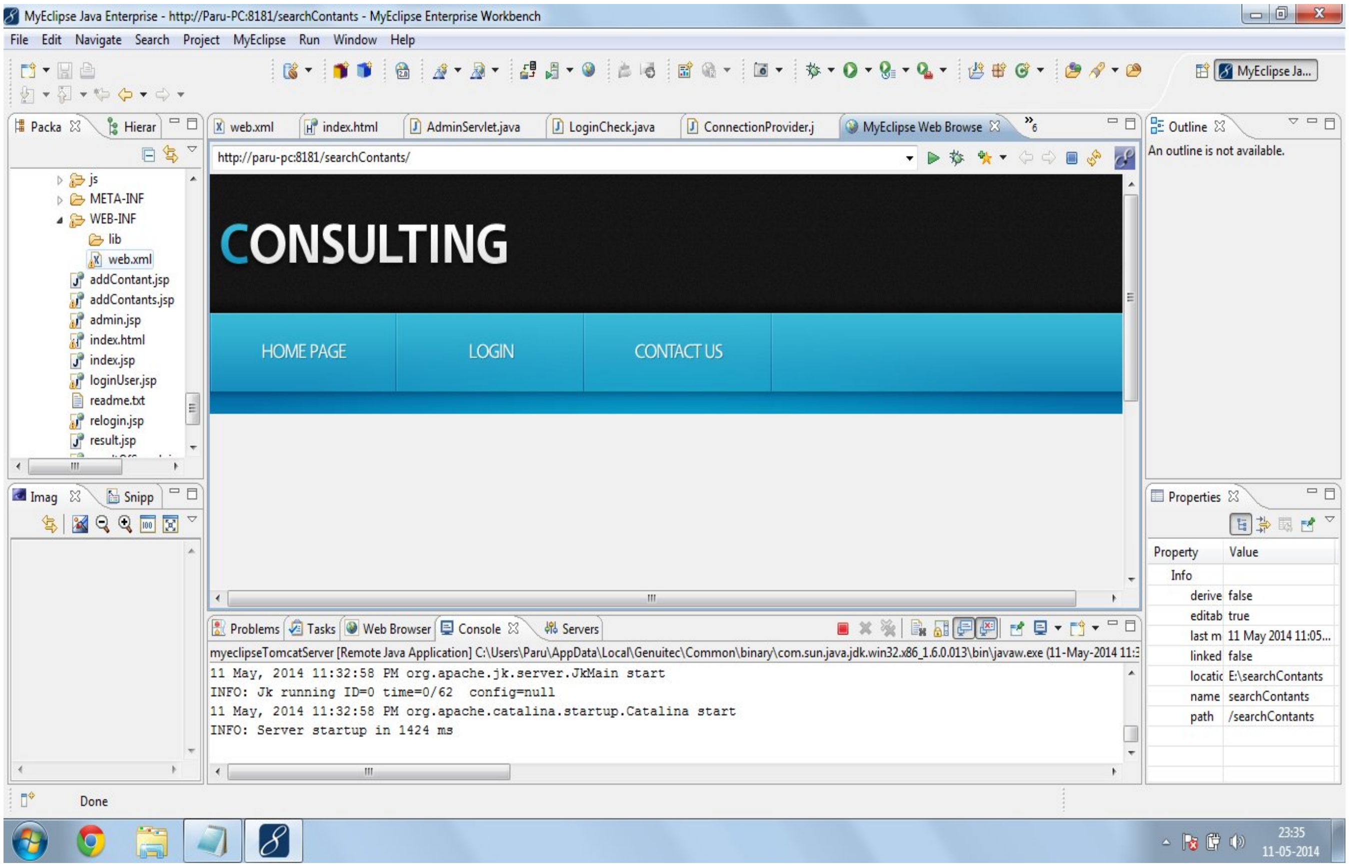
Also note there is a option of "including" or "excluding" a particular term in your keyword suggestions that can come in very handy to further refine search. Suppose, you only want to search for keywords that have your Brand Name or want to exclude a particular set of adjectives for. Eg."cheap" or "second hand" if you only sell brand new luxury cars from the possible keyword suggestions.

So while selecting a keywords for SEO we should generally consider exact matches to check relevant search volume and start creating our "keyword mix" or "keyword cluster" that has entire set of permutations and combinations on main keywords we want to rank for. Suggestions that Google provide can really be handy when planning for keywords.

Chapter 5:- Flowcharts:-

1. Search Engine:-



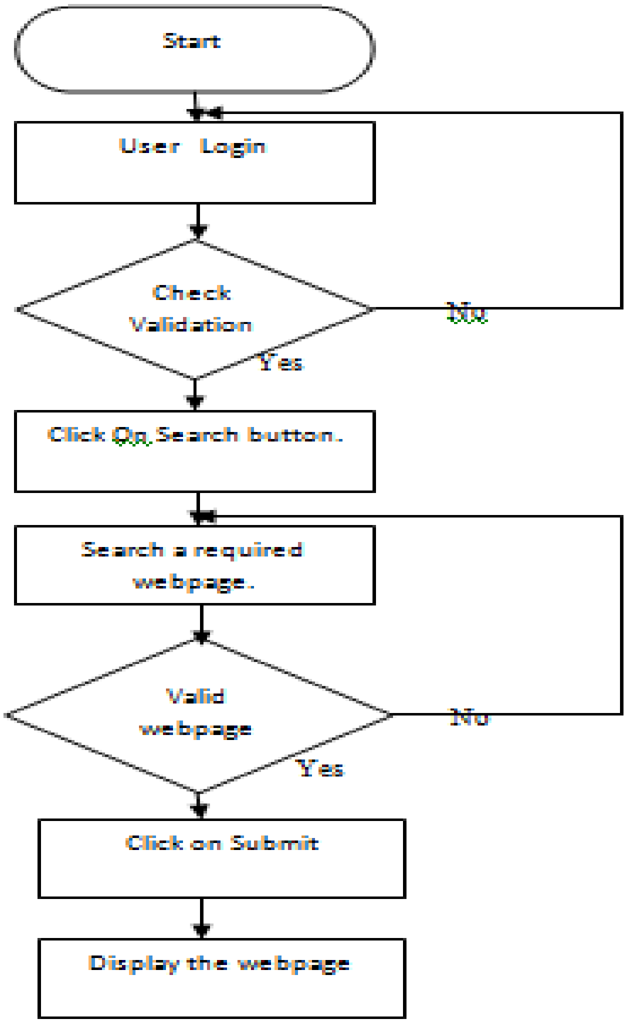


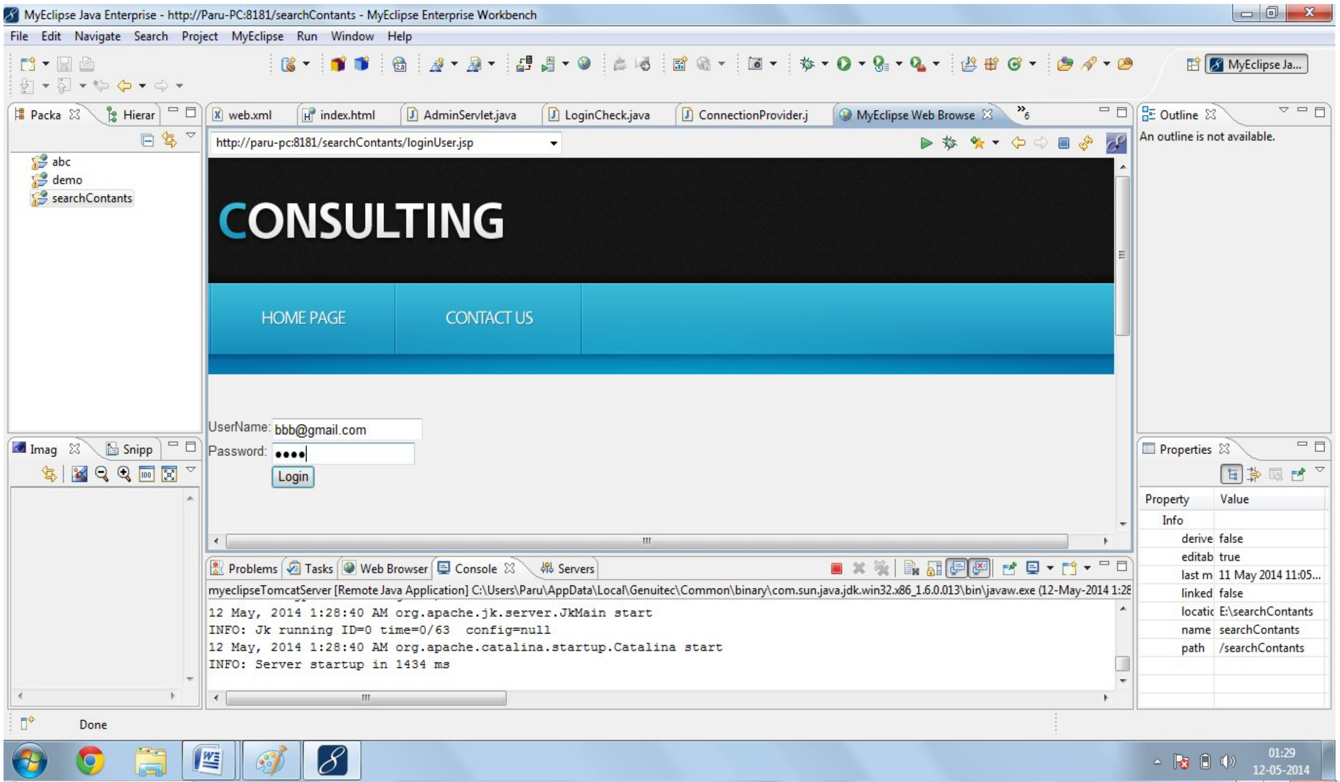
http://paru-pc:8181/searchContants/

```
myeclipseTomcatServer [Remote Java Application] C:\Users\Paru\AppData\Local\Genuitec\Common\binary\com.sun.java.jdk.win32.x86_1.6.0.013\bin\javaw.exe (11-May-2014 11:3
11 May, 2014 11:32:58 PM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/62  config=null
11 May, 2014 11:32:58 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 1424 ms
```

Property	Value
Info	
derive	false
editab	true
last m	11 May 2014 11:05...
linked	false
locatic	E:\searchContants
name	searchContants
path	/searchContants

2. User login:-


Start
User Login
Check Validation
No
Yes
Click On Search button.
Search a required webpage.
Valid webpage
No
Yes
Click on Submit
Display the webpage



MyEclipse Java Enterprise - http://Paru-PC:8181/searchContants - MyEclipse Enterprise Workbench

http://paru-pc:8181/searchContants/loginUser.jsp

UserName: bbb@gmail.com

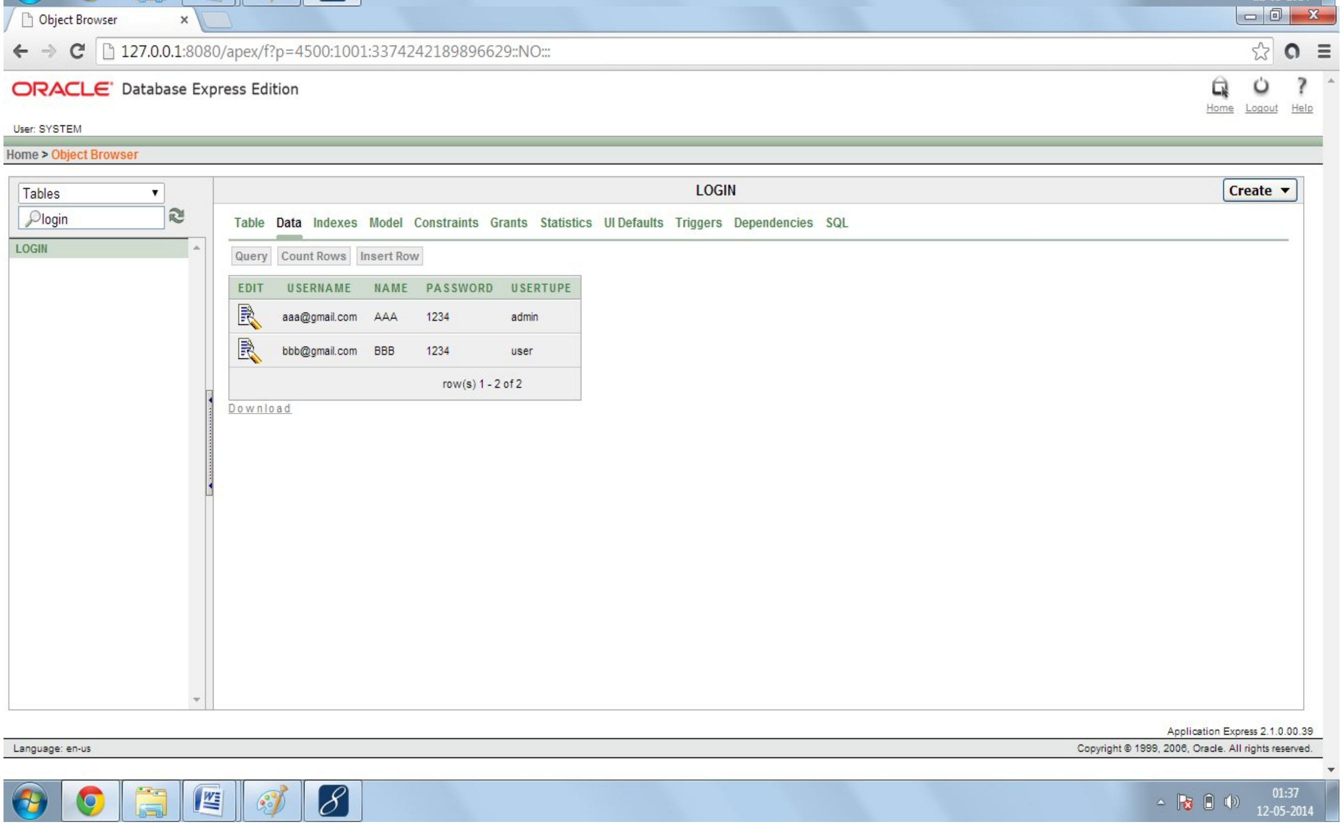
Password: ••••

An outline is not available.

```
myeclipseTomcatServer [Remote Java Application] C:\Users\Paru\AppData\Local\Genuitec\Common\binary\com.sun.java.jdk.win32.x86_1.6.0.013\bin\javaw.exe (12-May-2014 1:28
12 May, 2014 1:28:40 AM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/63  config=null
12 May, 2014 1:28:40 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 1434 ms
```

Property	Value
Info	
derive	false
editab	true
last m	11 May 2014 11:05...
linked	false
locatic	E:\searchContants
name	searchContants
path	/searchContants

01:29
12-05-2014



Object Browser
ORACLE Database Express Edition
Home
Logout
Help
Tables
LOGIN
Create
Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL
Query
Count Rows
Insert Row
Download

127.0.0.1:8080/apex/f?p=4500:1001:3374242189896629::NO:::

User: SYSTEM

Home > Object Browser

login

EDIT	USERNAME	NAME	PASSWORD	USERTYPE
	aaa@gmail.com	AAA	1234	admin
	bbb@gmail.com	BBB	1234	user

row(s) 1 - 2 of 2

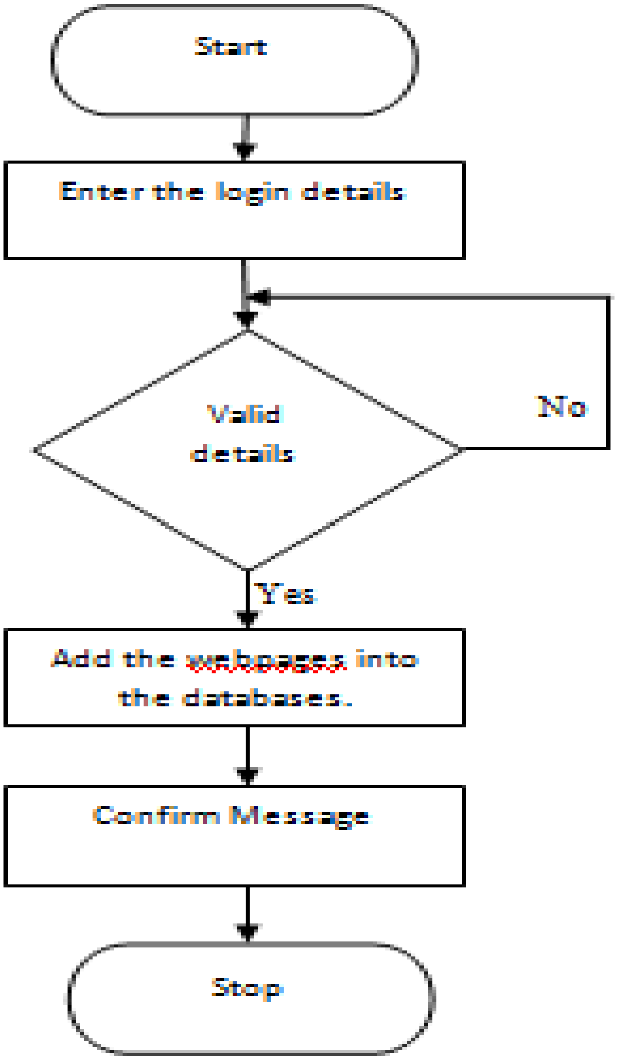
Application Express 2.1.0.00.39

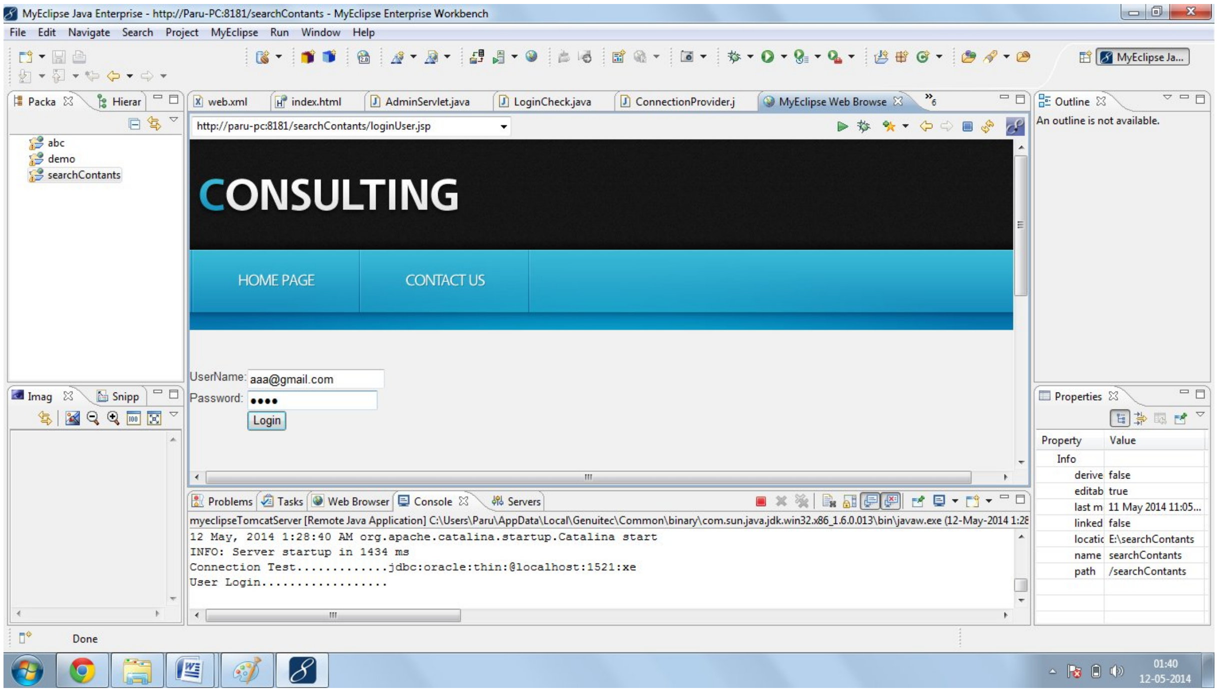
Language: en-us

Copyright © 1999, 2006, Oracle. All rights reserved.

01:37
12-05-2014

3. Admin Login:-



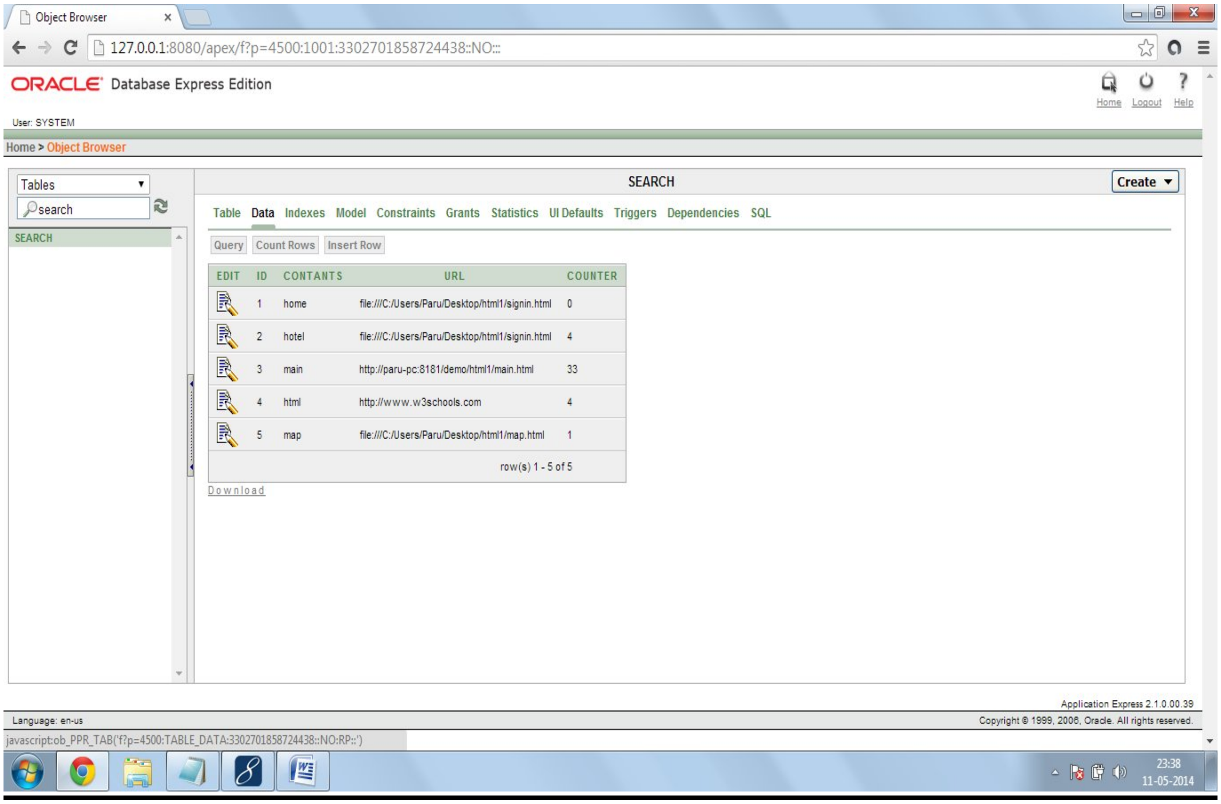

MyEclipse Java Enterprise - http://Paru-PC:8181/searchContants - MyEclipse Enterprise Workbench
File Edit Navigate Search Project MyEclipse Run Window Help
abc
demo
searchContants
web.xml
index.html
AdminServlet.java
LoginCheck.java
ConnectionProvider.j
MyEclipse Web Browse
http://paru-pc:8181/searchContants/loginUser.jsp
CONSULTING
HOME PAGE
CONTACT US
Login
Outline
An outline is not available.
Properties
Problems Tasks Web Browser Console Servers
Done

UserName: aaa@gmail.com
Password: ●●●●

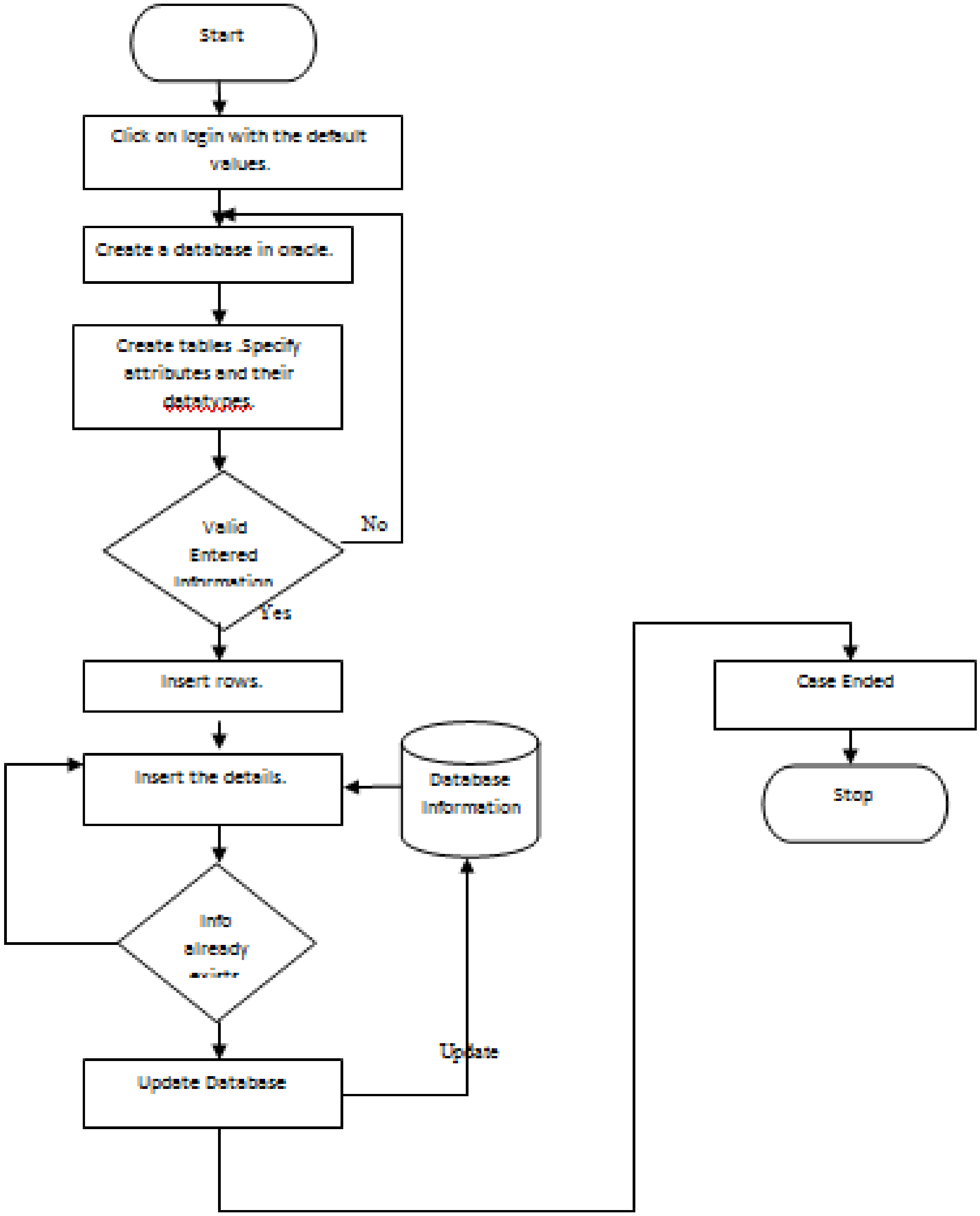
Property	Value
Info	
derive	false
editab	true
last m	11 May 2014 11:05...
linked	false
locatic	E:\searchContants
name	searchContants
path	/searchContants

```
myeclipseTomcatServer [Remote Java Application] C:\Users\Paru\AppData\Local\Genuitec\Common\binary\com.sun.java.jdk.win32.x86_1.6.0.013\bin\javaw.exe (12-May-2014 1:28
12 May, 2014 1:28:40 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 1434 ms
Connection Test.............jdbc:oracle:thin:@localhost:1521:xe
User Login..................
```

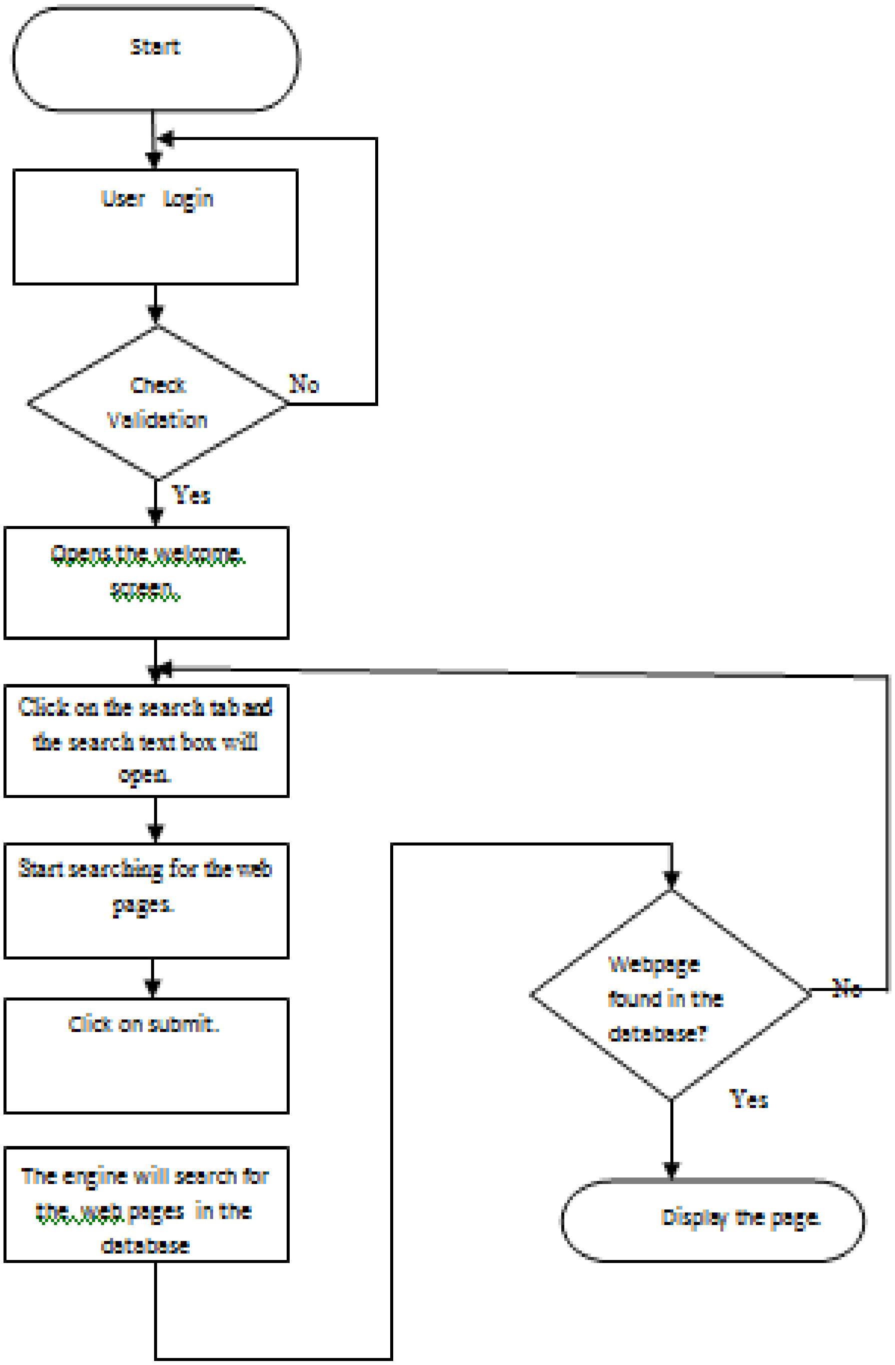
4.**Add webpage:-**

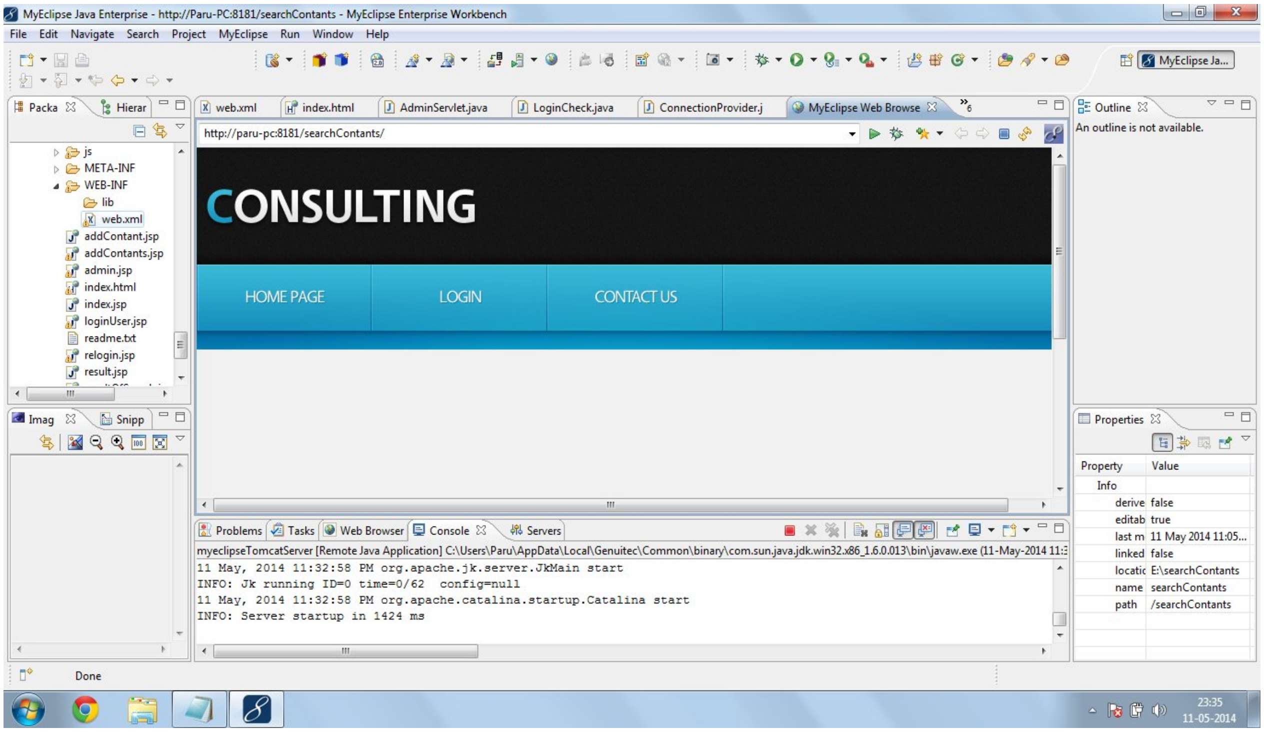

Object Browser
127.0.0.1:8080/apex/f?p=4500:1001:3302701858724438::NO:::
ORACLE Database Express Edition
Home Logout Help
User: SYSTEM
Home > Object Browser
Tables
search
SEARCH
SEARCH
Create
Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL
Query Count Rows Insert Row
Download
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
javascript:ob_PPR_TAB('f?p=4500:TABLE_DATA:3302701858724438::NO:RP::')

EDIT	ID	CONTANTS	URL	COUNTER
✎	1	home	file:///C:/Users/Paru/Desktop/html1/signin.html	0
✎	2	hotel	file:///C:/Users/Paru/Desktop/html1/signin.html	4
✎	3	main	http://paru-pc:8181/demo/html1/main.html	33
✎	4	html	http://www.w3schools.com	4
✎	5	map	file:///C:/Users/Paru/Desktop/html1/map.html	1
			row(s) 1 - 5 of 5	


Start
Click on login with the default values.
Create a database in oracle.
Create tables .Specify attributes and their datatypes.
Valid Entered Information
No
Yes
Insert rows.
Insert the details.
Database Information
Info already exists
Update Database
Update
Case Ended
Stop

5.Abstract Overview:-


Start
User Login
Check Validation
No
Yes
Opens the welcome screen.
Click on the search tab and the search text box will open.
Start searching for the web pages.
Click on submit.
The engine will search for the web pages in the database
Webpage found in the database?
No
Yes
Display the page.


MyEclipse Java Enterprise - http://Paru-PC:8181/searchContants - MyEclipse Enterprise Workbench
File Edit Navigate Search Project MyEclipse Run Window Help
web.xml
index.html
AdminServlet.java
LoginCheck.java
ConnectionProvider.j
MyEclipse Web Browse
http://paru-pc:8181/searchContants/
CONSULTING
HOME PAGE
LOGIN
CONTACT US
Outline
An outline is not available.
Properties
Problems
Tasks
Web Browser
Console
Servers
Done
23:35
11-05-2014

```
myeclipseTomcatServer [Remote Java Application] C:\Users\Paru\AppData\Local\Genuitec\Common\binary\com.sun.java.jdk.win32.x86_1.6.0.013\bin\javaw.exe (11-May-2014 11:3
11 May, 2014 11:32:58 PM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/62  config=null
11 May, 2014 11:32:58 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 1424 ms
```

Property	Value
Info	
derive	false
editab	true
last m	11 May 2014 11:05...
linked	false
locatic	E:\searchContants
name	searchContants
path	/searchContants

Chapter 6: Software Engineering Aspects of Project

Software Engineering is the application of a systematic, disciplined, quantifiable approach to the design, development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software. It is inter-disciplinary in nature. It has emerged as a discipline very recently. Due to rapid growth of knowledge in this field, software professionals and academicians felt the need to have a consistent view of software engineering worldwide. To achieve this objective the IEEE Computer Society's Professional Practices Committee has published a guide to Software Engineering Body of Knowledge (SWEBOK) in 2004. The guide is based on a generally accepted portion of the Body of Knowledge. The material that is recognized as being within this discipline is organized into ten knowledge areas as follows:

- **Software requirements:** The elicitation, analysis, specification, and validation of requirements for software.
- **Software design:** The process of defining the architecture, components, interfaces, and other characteristics of a system or component. It is also defined as the result of that process.

- **Software construction:** The detailed creation of working, meaningful software through a combination of coding, verification, unit testing, integration testing, and debugging.
- **Software testing:** The dynamic verification of the behavior of a program on a finite set of test cases, suitably selected from the usually infinite executions domain, against the expected behavior.
- **Software maintenance:** The totality of activities required to provide cost-effective support to software.
- **Software configuration management:** The identification of the configuration of a system at distinct points in time for the purpose of systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the system life cycle.
- **Software engineering management:** The application of management activities—planning, coordinating, measuring, monitoring, controlling, and reporting—to ensure that the development and maintenance of software is systematic, disciplined, and quantified.
- **Software engineering process:** The definition, implementation, assessment, measurement, management, change, and improvement of the software life cycle process itself.
- **Software engineering tools and methods**: The computer-based tools that are intended to assist the software life cycle processes, see Computer Aided Software Engineering, and the methods which impose structure on the software engineering activity with the goal of making the activity systematic and ultimately more likely to be successful.
- **Software quality**: The degree to which a set of inherent characteristics fulfills requirements.

Software engineering is concerned with both technical as well as managerial aspects of software development. There are four important core aspects of software development. These four aspects are: (1) Product, (2) Process, (3) People and (4) Project.

- **Product:** Software, as a product, has to perform certain specific functions required by users (customers). Determination of correct functional requirements and features of software to be produced is a very critical activity of software development. For this, various stakeholders and users of software are identified to elicit information for determining functional specification of software. Sometimes requirements of one class of users may conflict with those of another. Finalization of functional specification is often a balancing act of satisfying requirements of different stakeholders within cost and time constraints.

- **Process:** It refers to methodologies to be followed for developing the software. It is the framework for establishment of a comprehensive plan and strategies for software development. The process specifies the policies, procedures, tools and techniques to be used for software development. A number of models are available. CMM (Capability Maturity Model) is a widely used standard model for software development process.

- **People:** Software development requires creativity and knowledge work. It is often difficult to specify the quantitative and qualitative measures of this work. Since fast technological developments are taking place in the field of computer science, updating of knowledge is a part of computer professionals' job. A number of people having diverse expertise are required to work together for developing any software. Since work of one individual affects the work of others, quality software is mostly developed through teamwork. Hence, developing motivation, morale and teamwork among people and upgrading their professional expertise are important aspects of software engineering.

- **Project:** As stated earlier, there is a requirement for great amounts of effort, time and money to design and develop any commercial software. A number of interrelated activities have to be performed in a planned schedule for completing the software within time constraints. Hence, development of any software can be considered as a project. Thus, project management aspects such as planning and monitoring of activities, schedule, resources and expenditure are important for software development.

DIAGRAMS:-

6.1 Use Case Diagram:-

A use case diagram is a behaviorial diagram, which aims to present a graphical overview of the functionalities provided by the system. It consists of a set of actions (use cases) that the concerned system can perform, one or more actors, and dependencies among them. They are useful for presentations to managements and/or project stakeholders, but for actual development you will find that use cases provide significantly more value because they describe “the meet” of the actual requirements.
Use case diagrams depicts:-

Use Cases: A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

Actor: An actor is a person, organization or an external system that plays a role in one or more interactions with your system. Actors are drawn as stick figures.

Associations: A use case describes a specific functionality that the system provides to its users. The functionality is triggered by actor. Actors are connected to use cases through binary associations. The association indicates that the actor and use case communicates through message passing. An actor must be associated with atleast one use case. Similarly, a given use case must be associated with atleast one actor. However, when a use case is associated with multiple actors, it might not be clear who triggers the functionality. No association among the actors are shown.

Graphical Representation

An actor is represented by a stick figure and name of the actor is written below it. A use case is depicted by an ellipse and label of the use case is written inside it. The subject could be shown by drawing a rectangle. Label for the system could be put inside it either at the top or near the bottom. Use cases are enclosed inside the rectangle and actors are drawn outside the rectangle.

Use Case Relationships

Three types of relationships exist among use cases:

- Include relationship
- Extend relationship
- Use case generalization

Include Relationship

Include relationships are used to depict similar behaviour that are shared by multiple use cases without replicating the common behaviour in each of those use cases.

Extend Relationship

A use case extends a base use case to obtain the properties and behaviour of the base use case as well as add some new features. This is often the case when the extended use case could not be modified to accomodate additional functionalities, or there are multiple use cases having behaviour and propertiessimilar to the base use case.

Generalization Relationship

Generalization relationship exists between a base use case and a derived use case when the derived use case specializes some functionalities it has inherited from the base use case.

Use Case Diagram of Search Engine

- User
- Query Processor
- Indexer
- Crawler

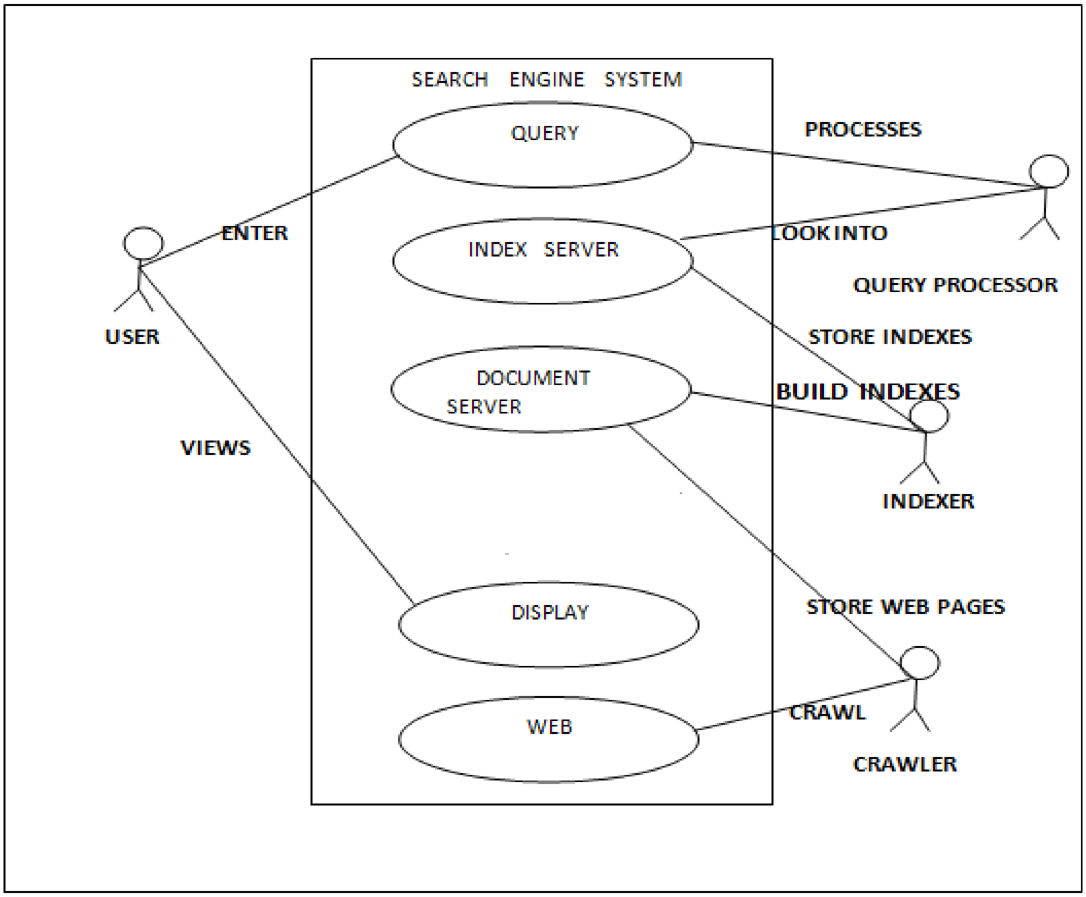


Fig 2: Use Case Diagram

Use Case Number	1
Use Case Name	Search Engine Optimization
Precondition(s)	Required softwares installed
Successful Post Condition	The highest ranked page is displayed first
Actors	**User , Admin**
Priority	High
Related Use Cases	1. Create Database 2. Create table 3. Create a web site 4. Create search engine 5. Create login 6. Increment the counter 7. View results
	Basic Flow **Step No.** | **Steps**

Fig 3:- Use case description of SEO

6.2. Data Flow Diagram

A **data flow diagram** (**DFD**) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFD’s can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

It is common practice to draw the context-level data flow diagram first, which shows the interaction between the system and external agents which act as data sources and data sinks. This helps to create an accurate drawing on the context diagram. The system's interactions with the outside world are modelled purely in terms of data flows across the system boundary. The context diagram shows the entire system as a single process, and gives no clues as to its internal organization.

This context-level DFD is next "exploded", to produce a Level 1 DFD that shows some of the detail of the system being modeled. The Level 1 DFD shows how the system is divided into sub-systems (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job, and shows the flow of data between the various parts of the system.

FUNCTION

FILE DATABASE

INPUT/OUTPUT

FLOW

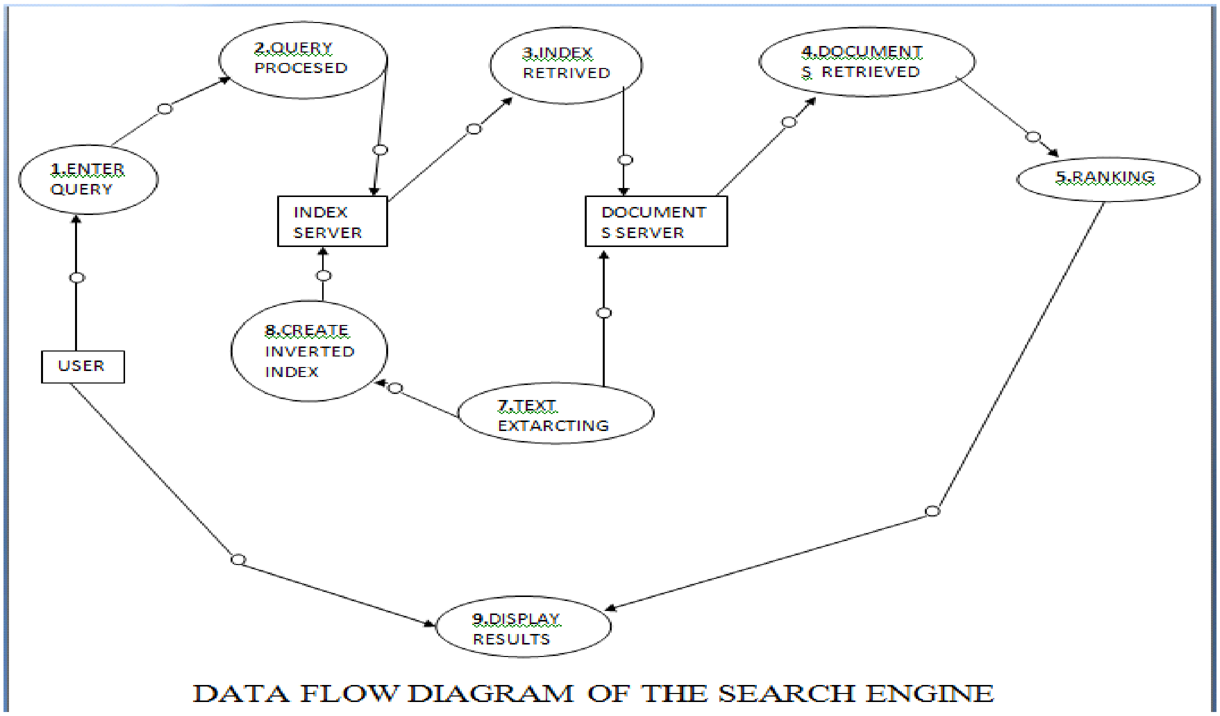


Fig 4: Data Flow Diagram

Chapter 7: Software Development Life Cycle

A software Development process ,also known as Software development life-cycle (SDLC),is a structure imposed on the development of the software product. Similarly terms include software life cycle and software process. It is often considered a subset of systems development life cycle. There are several models for such processes ,each describing approaches to a variety of tasks or activities that take place during the process. Some people consider a life- cycle model or more general term and a software development processor more specific term. For example, there are many specific software development processes that' fit' the spiral life cycle model.ISO/IEC 12207 is an international standard for software life cycle processes . It aims to be the standard that defines all the tasks required for developing and maintaining software.

7.1. Software Development Models

Several models exist to streamline the development process. Each one has its pros and cons, and it's up to the development team to adopt the most appropriate one for the project. Sometimes a combination of the models may be more suitable.

7.1.1. Waterfall Model

Waterfall model shows a process, where developers are to follow these phases in order.

- Requirements specification (requirements analysis)
- Software design
- Implementation and Integration
- Testing or Validation
- Deployment (or Installation)
- Maintenance

In a strict waterfall model, after each phase is finished, it proceeds to the next one. Reviews may occur before moving to the next phase which allows for the possibility of changes (which may involve the formal change control process). Reviews may also be employed to ensure that the phase is indeed complete; the phase completion criteria are often referred to as “gate” that the project must pass through to move to the next phase. Waterfall discourages revisiting and revising any prior phase since it’s complete. This “inflexibility” in a pure waterfall model has been a source for criticism by supporters of other more flexible models.

7.1.2. Spiral Model

The key characteristic of spiral model is risk management at regular stages in the development cycle. In 1988, Barry Boehm published a formal software system development “spiral model”, which combines some key aspect of the waterfall model and rapid prototyping methodologies, but provided emphasis in a key area many felt has been neglected by other methodologies: deliberate iterative risk analysis, particularly suited to large- scale complex systems.

The spiral is visualized as a process passing through some number of iterations, with the four quadrant diagram representative of the following activities:

- **Formulate plans to:** identify software targets, selected to implement the program , clarify the project development restrictions.
- **Risk analysis:** an analytical assessment of selected programs, to consider hoe to identify and eliminate risk.
- **Implementation of the project:** the implementation of software development and verification.

Risk-driven spiral model, emphasizing the conditions of options and constrains in order to support software reuse, software quality can help as a special goal of integration into the project development. However, the spiral model has some restrictive conditions, as follows:

- The spiral model emphasizes risk analysis, and thus requires customers to accept this analysis and act on it. This requires both trust in the developer as well as the willingness to spend more to fix the issues, which is the reason why this model is often used for large- scale internal software development.
- If the implementation of risk analysis will greatly affect the profits of the project, the spiral model should not be used.
- Software developers have to actively look for possible risks, and analyze accurately for the spiral model to work.

The first stage is to formulate a plan to achieve the objectives with these constraints, and then strive to find and remove all potential risks through careful analysis and, if necessary, by constructing a prototype. If some risks cannot be ruled out, the customer has to decide whether to terminate project or to ignore the risks and continue anyways. Finally, the results are evaluated and the design of the next phase begins.

7.1.3. Iterative and Incremental Development

Iterative development prescribes the construction of initially small but ever-larger portions of a software project to help all those involved to uncover important issues early before problems or faulty assumptions can lead to disaster.

7.1.4. Incremental Prototype Model

The prototyping model is systems development method(SDM) in which a prototype(an early approximation of a final system or a product) is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed. This model works best in scenarios where not all of the project requirements are known in detail ahead of time. It is an iterative, trail-and –error process that takes place between the developers and users.

The incremental approach can be likened to “building blocks”; incrementing each time a new component is added or integrated, based on an overall design solution. When all the components are in place, the solution is complete.

An advantage of this model is that the client and/or end users have the opportunity to test the developed components and their functionality. They also have oppotunitites to provide feedback while other components are still in development, and can thus influence the outcome of further development.

Chapter 8: Source Code

- AdminServlet.java:-

```
package process;


import javax.servlet.http.*;
import javax.servlet.*;


import dao.DaoImpl;


import model.Contants;


import java.io.*;


public class AdminServlet extends HttpServlet
{
	public void doGet(HttpServletRequest request,HttpServletResponse 
response)throws IOException,ServletException
	{
		String cont = request.getParameter("contantName");
```

```
                String url = request.getParameter("url");


                Contants con = new Contants();

                con.setContants(cont);

                con.setUrl(url);


                DaoImpl dao = new DaoImpl();

                dao.save(con);


                RequestDispatcher rd = request.getRequestDispatcher("addContants.jsp");

                rd.forward(request, response);

        }

}
```

- **Login Servlet:-**

```
package process;


import javax.servlet.http.*;

import javax.servlet.*;


import dao.LoginCheck;

import model.User;


import java.io.*;
```

```
public class LoginServlet extends HttpServlet
{
    public void doPost(HttpServletRequest
request,HttpServletResponse response)throws
IOException,ServletException
    {
        RequestDispatcher rd = null;
        User user = new User();

        user.setUserName(request.getParameter("userName"));
        user.setPassword(request.getParameter("password"));

        LoginCheck login = new LoginCheck();
        user = login.findUser(user);

        HttpSession session = request.getSession();
        session.setAttribute("user",user);



        if(user.getName()!=null && user.getName()!="")
        {
            if(user.getUserType().equalsIgnoreCase("admin"))
            {
                System.out.println("Admin
Login.................");
                rd =
request.getRequestDispatcher("admin.jsp");
```

```
            }

            else

            {

                System.out.println("User Login.................");

                 rd = 
request.getRequestDispatcher("user.jsp");

            }

        }

        else

        {

            System.out.println("Re-Login.................");


    System.out.println("UserType....................."+user.getUserType());

            rd = request.getRequestDispatcher("relogin.jsp");

        }


        rd.forward(request, response);

    }

}
```

- **Logout Servlet:-**

```
package process;


import javax.servlet.http.*;

import javax.servlet.*;

import java.io.*;
```

```
public class LogoutServlet extends HttpServlet
{
        public void doGet(HttpServletRequest
request,HttpServletResponse response)throws
IOException,ServletException
        {
                HttpSession session = request.getSession();
                session.invalidate();

                RequestDispatcher rd =
request.getRequestDispatcher("index.html");
                rd.forward(request, response);
        }
}
```

- **Search Servlet:-**

```
package process;

import javax.servlet.http.*;
import javax.servlet.*;

import model.Contants;
import dao.DaoImpl;

import java.io.*;
import java.util.*;
```

```
public class SearchServlet extends HttpServlet

{

	public void doGet(HttpServletRequest
request,HttpServletResponse response)throws
IOException,ServletException

	{

		String contants = request.getParameter("searchContant");



		System.out.println("SearchServlet..........................");

		DaoImpl dao = new DaoImpl();

		List list = dao.findAll(contants);



		HttpSession session = request.getSession();

		session.setAttribute("list",list);


		RequestDispatcher rd =
request.getRequestDispatcher("resultOfSearch.jsp");

		//RequestDispatcher rd =
request.getRequestDispatcher("result.jsp");

		rd.forward(request,response);


	}

}
```

- **Update Counter:-**

```
package process;


import javax.servlet.http.*;
import javax.servlet.*;


import dao.DaoImpl;


import java.io.*;


public class UpdateCounterServlet extends HttpServlet
{
    public void doGet(HttpServletRequest
request,HttpServletResponse response)throws
IOException,ServletException
    {
        System.out.println("UpdateCounterServlet Is
Invoked...............");



        String href = request.getParameter("a");
        int id = Integer.parseInt(request.getParameter("c"));


        System.out.println("HREF..."+href);
        System.out.println("id..."+id);
```

```
                DaoImpl dao = new DaoImpl();

                boolean flag = dao.updateCounter(id);


                //RequestDispatcher rd =
        request.getRequestDispatcher("http://paru-
        pc:8181/demo/html1/main.html");

                //rd.forward(request,response);


                System.out.println("forwarding........................"+href);

                response.sendRedirect(href);

            }

        }
```

- **User.Java:-**

```
package model;

public class User
{
	String userName,name,password,userType;

	public String getUserName() {
		return userName;
	}

	public void setUserName(String userName) {
		this.userName = userName;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}
```

```
    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getUserType() {
        return userType;
    }

    public void setUserType(String userType) {
        this.userType = userType;
    }


}
```

- **Contants.java:-**

```
package model;

public class Contants
{
    int id,count;
    String contants,url;
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public int getCount() {
        return count;
    }
    public void setCount(int count) {
        this.count = count;
    }
    public String getContants() {
        return contants;
    }
    public void setContants(String contants) {
        this.contants = contants;
```

```
	}
	public String getUrl() {
		return url;
	}
	public void setUrl(String url) {
		this.url = url;
	}


}
```

- **Deolmpl.java:-**

```
package dao;

import java.util.*;
import java.sql.*;

import model.Contants;
import connection.ConnectionProvider;

public class DaoImpl
{
	public List findAll(String contants)
	{
		String sql = "select * from search where contants like 
'%"+contants+"%' order by counter desc";

		List list = new ArrayList();
		Contants cont = null;
		try
		{
			Connection conn = 
ConnectionProvider.getConnection();
			PreparedStatement stmt = 
conn.prepareStatement(sql);
			ResultSet rset = stmt.executeQuery();
			while(rset.next())
			{
				cont = new Contants();
				cont.setId(rset.getInt(1));
				cont.setContants(rset.getString(2));
				cont.setUrl(rset.getString(3));
```

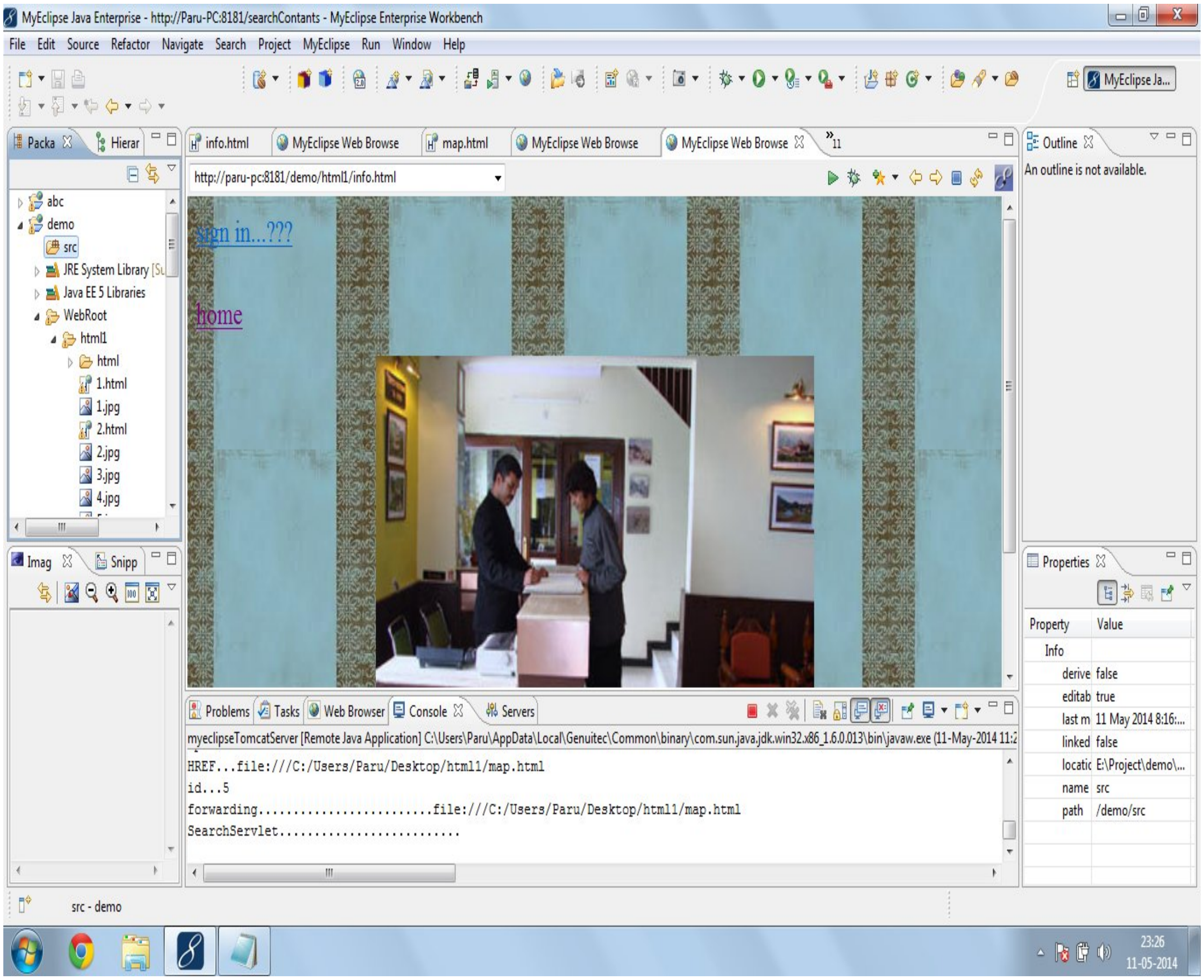
```
                cont.setCount(rset.getInt(4));
                list.add(cont);
            }
        }
        catch(Exception e)
        {
            System.out.println("Exception In DaoImpl method
findAll().........."+e);
        }

        return list;
    }

    public void save(Contants cont)
    {
        String sql = "insert into search values(?,?,?,?)";
        try
        {
            Connection conn =
ConnectionProvider.getConnection();
            Statement stmt = conn.createStatement();
            ResultSet rset = stmt.executeQuery("select max(id)
from search");
            rset.next();
            int id = rset.getInt(1);

            PreparedStatement pstmt =
conn.prepareStatement(sql);
            pstmt.setInt(1,id);
            pstmt.setString(2,cont.getContants());
            pstmt.setString(3,cont.getUrl());
            pstmt.setInt(4,0);
            int rows=pstmt.executeUpdate();
            System.out.println("Total Rows Effected With
Insert: "+rows);
        }
        catch(Exception e)
        {
            System.out.println("Exception in save method in
class DaoImpl........"+e);
        }
    }

    public boolean updateCounter(int id)
    {
        boolean flag=false;
```

```
                String sql = "update search set counter=counter+1 where
id=?";
                try
                {
                        Connection conn =
ConnectionProvider.getConnection();
                        PreparedStatement pstmt =
conn.prepareStatement(sql);
                        pstmt.setInt(1, id);
                        flag = pstmt.execute();
                }
                catch(Exception e)
                {
                        System.out.println("Exception In UpdateCounter In
Class DaoImpl...."+e);
                }
                return flag;
        }
}
```

- **LoginCheck.java:-**

```
package dao;



import model.User;

import java.sql.*;



import connection.ConnectionProvider;



public class LoginCheck

{

        public User findUser(User user)

        {

                String sql = "select * from login where userName=? and password=?";
        try
        {
            Connection conn = ConnectionProvider.getConnection();
            PreparedStatement pstmt = conn.prepareStatement(sql);
            pstmt.setString(1,user.getUserName());
            pstmt.setString(2,user.getPassword());
            ResultSet rset = pstmt.executeQuery();
            if(rset.next())
            {
                user.setName(rset.getString(2));
                user.setUserType(rset.getString(4));
            }
        }
        catch(Exception e)
        {
            System.out.println("Exception In FindUser of class LoginCheck: 
"+e);
        }
        return user;
    }
}
```

Result:-

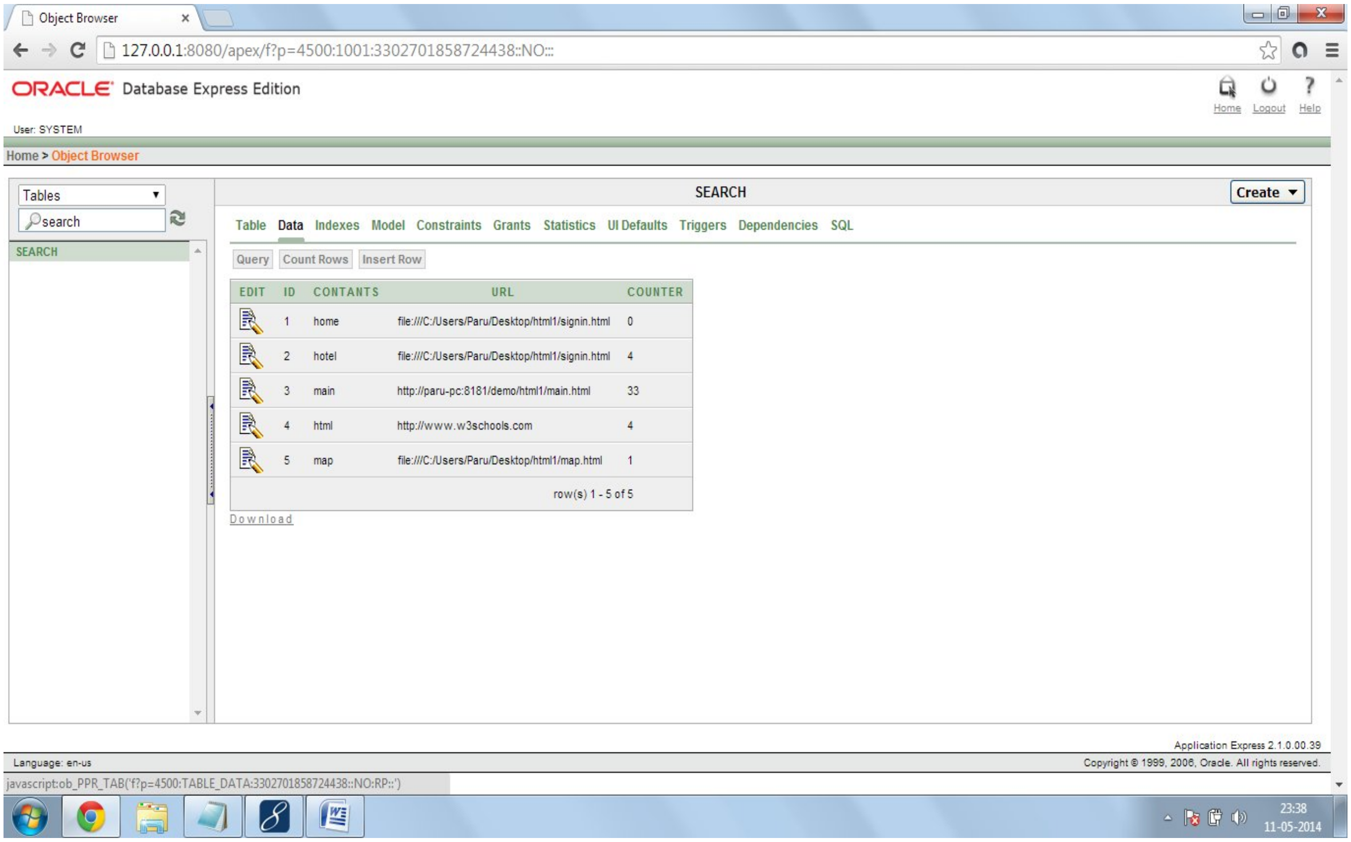


http://paru-pc:8181/demo/html1/info.html

myeclipseTomcatServer [Remote Java Application] C:\Users\Paru\AppData\Local\Genuitec\Common\binary\com.sun.java.jdk.win32.x86_1.6.0.013\bin\javaw.exe (11-May-2014 11:2

```
HREF...file:///C:/Users/Paru/Desktop/html1/map.html
id...5
forwarding........................file:///C:/Users/Paru/Desktop/html1/map.html
SearchServlet.........................
```

Property	Value
Info	
derive	false
editab	true
last m	11 May 2014 8:16:...
linked	false
locatic	E:\Project\demo\...
name	src
path	/demo/src



EDIT	ID	CONTANTS	URL	COUNTER
✎	1	home	file:///C:/Users/Paru/Desktop/html1/signin.html	0
✎	2	hotel	file:///C:/Users/Paru/Desktop/html1/signin.html	4
✎	3	main	http://paru-pc:8181/demo/html1/main.html	33
✎	4	html	http://www.w3schools.com	4
✎	5	map	file:///C:/Users/Paru/Desktop/html1/map.html	1

row(s) 1 - 5 of 5

javascript:ob_PPR_TAB('f?p=4500:TABLE_DATA:3302701858724438::NO:RP::')

Work Done till last semester:-

Implementation of a search engine using singular value decomposition.

REFERENCES

[Cohen, 1998] W. Cohen. A web-based information system that reasons with structured collections of text. In *Agent '98*, 1998.

http://searchengineland.com/
Ultimate guide to search engine optimization.[Aaron Wall's seo book]

[Kaelbling *et al.*, 1996] L. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *Journal of Arti_cial Intelligence Research*, 4:237-285, 1996.

[Rennie and McCallum, 1999] Jason Rennie and Andrew McCallum. Using reinforcement learning to spider theWeb efficiently. In *ICML-99*, 1999.

http://www.submitexpress.com/

Search-Engine-Optimization-Starter guide pdf