

ABSTRACT

Name of the Project	Online Library
Vision	Online Library is a system, which maintains books in the server in any format. It allows storing books, documents, audio and video files. User can Search and open documents by giving a simple query.
Deliverables / Functional Specifications	<p>Online Library Management System is a system, which maintains books, documents, audio/video files. To use this system user must be a member in this. Then user allowed storing books, documents, A/V files. Then user can search by giving a simple query. It is very difficult search books manually. Online Library makes easy.</p> <p>Online Library uses user internal feedback from user to improve search process in every searching. It also uses indexing to locate documents very fast in searching which relate to query. Main aim of online Library system is providing highly relevant documents in very short period after giving query. It is just like Search Engine. It uses Vector space Model and Probabilistic Model to find similarity coefficient, weights.</p>
User Interface Requirements	Browser based
Database	Centralized

Requirements	
Integration Requirements	Web based Interface
Preferred Technologies	J2EE Technologies i.e. JSP, Servlets, jdbc Frontend: - Html, JavaScript, Java Server pages. Backend: - Oracle 10g.
Hardware Requirement	Pentium 4 processor with 256 MB RAM, 40 GB Hard disk

INTRODUCTION

INTRODUCTION

Online Library Management System is a system, which maintains books, documents, audio/video files. To use this system user must be a member in this. Then user allowed storing books, documents, A/V files. Then user can search by giving a simple query. It is very difficult search books manually. Online Library makes easy.

Online Library uses user internal feedback from user to improve search process in every searching. It also uses indexing to locate documents very fast in searching which relate to query. Main aim of online Library system is providing highly relevant documents in very short period after giving query. It is just like Search Engine. It uses Vector space Model and Probabilistic Model to find similarity coefficient, weights.

Existing System

Early days Libraries are managed manually. It required lot of time to record or to retrieve the details. The employees who have to record the details must perform their job very carefully. Even a small mistake would create many problems. Security of information is very less. Report generations of all the information is very tough task.

Maintenance of Library catalogue and arrangement of the books to the catalogue is very complex task. In addition to its maintenance of member details, issue dates and return dates etc. manually is a complex task.

All the operations must be performed in perfect manner for the maintenance of the library without any degradation, which may finally result in the failure of the entire system.

Benefits of automation

Automation is procedure of converting a traditional system in to a computer organization. To overcome the defects of the existing system automation was introduced by the computerization of organization we get many benefits.

The main objectives of undertaking this project are:

- The students will register them through Online
- Individually each member will have his account through which he can access the information he needs.
- Books, documents, A/V files are stored in server.
- User will enter simple query in text box and click on the Search button.
- Server takes the request and processes it. Returns hyper links of documents with their names, ranks weights and description.
- If the user clicks on name of the document then it will open with respective program and rank of document will increase for that query.(for example .pdf files with Adobe Reader)
- User can upload documents, books and a/v files.
- Time consuming is low, gives accurate results, reliability can be improved with the help of security.

Proposed System:

To solve the inconveniences as mentioned above, an **Online Library** proposed.

PROCESS LOGIC:

1. The user inputs data (e.g.: fills out an HTML form and clicks the submit button.)
2. The client (Browser) sends the data to the web server in a standard format (i.e., the GET method or the POST method).
3. The web server launches the program specified by the user and feeds it the input (form) data.
4. The program (e.g.: servlet) processes the form data and produces another HTML page.
5. The web server sends back the HTML page to the browser.

The web browser then displays the response page.

ANALYSIS

Analysis is the process of understanding the existing system by gathering and interpreting the facts, diagnosing the problems. It is not just to determining the how best to solve the manual system problems, it should also work for the system observes the feasibility of system then design, coding phases will be executed. Analysis phase delivers requirements specification .The system specification serves as an interface between the designer and developer as well as between developers and users. This describes the external behavior of the software without bothering about the internal implementation. Specification must be carefully checked for suitability, omission, inconsistencies and ambiguities.

Problem analysis is performed to getting a clear understanding of the needs of the clients and the users and what exactly desired form the software. Analysis leads to the actual specification. During the process of analysis, a massive amount of information is collected in the form of interviews, questionnaires, and information from documentation, and so forth. The major problem during analysis is resolving how to organize the information from documentation, and so forth. So the information can be effectively evaluated for completeness and consistency.

REQUIREMENT SPECIFICATION

FUNCTIONAL MODEL / USE CASE MODEL

IDENTIFYING ACTOR:

1. **Administrator:** He should register new Users, Delete Users etc. He can upload new documents, Documents, and A/V files.
2. **User:** He should register in site in order to access books. He can search for books. He can upload new books, documents, and A/V files into server.

IDENTIFYING SCENARIOS:

A Scenario is a scene, which explains a particular situation in more visualized model. A scenario consists of actors, scenes, and flow of events.

In our project, we have the following scenarios:

Scenario for Searching of EBooks, Document or A/V file:

Scene Name	Searching for books.
Actors	User/Administrator
Flow of events	User Opens home page of site. It will display text box and Search button. User simply enter query into text box, clicks on Search button. Browser sends request to the server. Server search documents for query and sends hyperlinks to documents (with weight, ranks, and a small description about document) in order of relevance to the query.

Scenario a for Opening EBook, Document or A/V file:

Scene Name	Opening Book, Document, or A/V file.
Actors	User / Administrator
Flow of events	User clicks on hyperlink of Document by reading description about document that had displayed below hyperlink. Then browser sends a request to the server again. The request contains name of document which user wants to open. The server will increase rank for document with respect to query into order to increase the relevance. Then server sends the document file in the form of response to browser. Now browser will open the document by using related application program on client system. (For example if server sent .mp3 file then browser opens it by using audio player, which installed on client's computer. Of course, Browser needs permeation of System Administrator to access application program.)

Scenario for Uploading a EBook, Document or A/V file into Server:

Scene Name	Uploading
Actors	Administrator / User
Flow of events	<p>First User clicks on hyperlink of "Upload document". Then browser sends request to upload file. Server return a webpage to browser it contains text box and browse button.</p> <p>By clicking on browse button a file window is displayed which allows to user to select a document from clients computer. After selecting document users click on upload button. Server checks for its extension. If document is.exe file server will not allow uploading file for the purpose of security. Then server insert name of document into documents table with unique document id. Then server creates index for document and insert into indexes table. Server returns a webpage, which contains confirmation and document id.</p>

Scenario for Modifying Users:

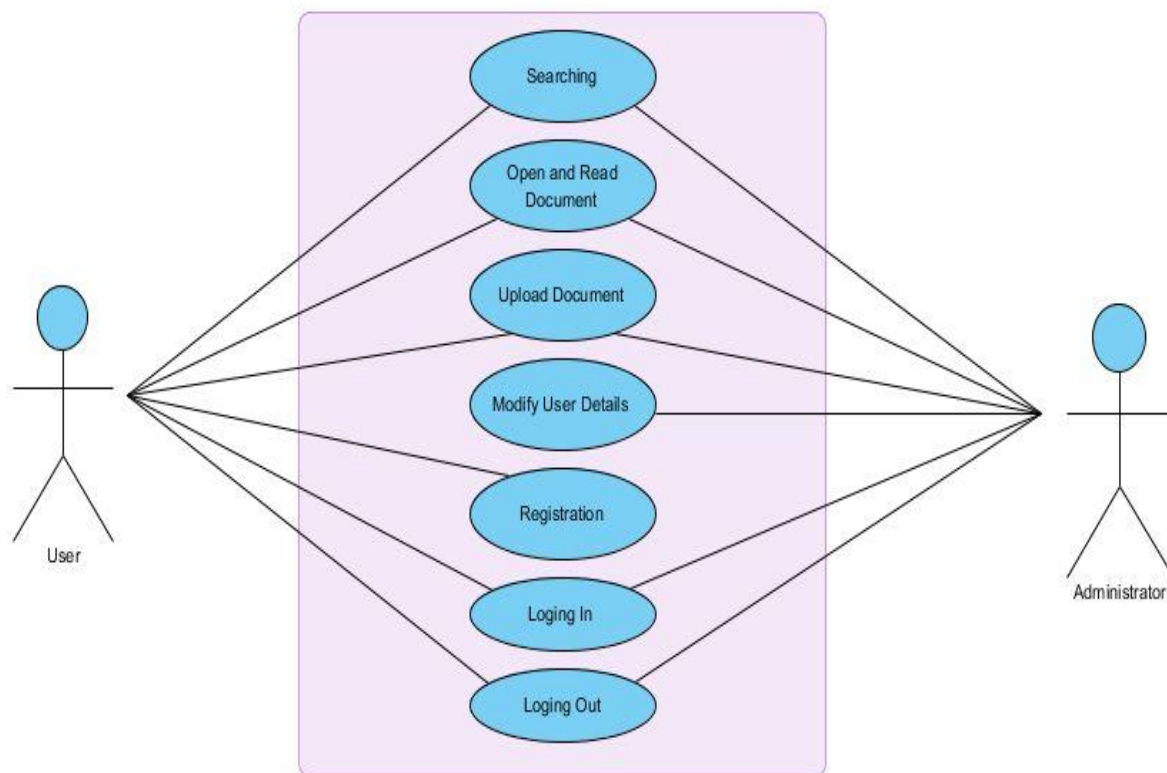
Scene Name	Modifying Users
Actors	Administrator
Flow of events	If login user is admin then homepage contains a special Hyperlink that “modify user.” If admin clicks on that hyperlink he will navigate to another webpage which contains details of users. In that admin can modify user details admin can delete users who violated terms of site.

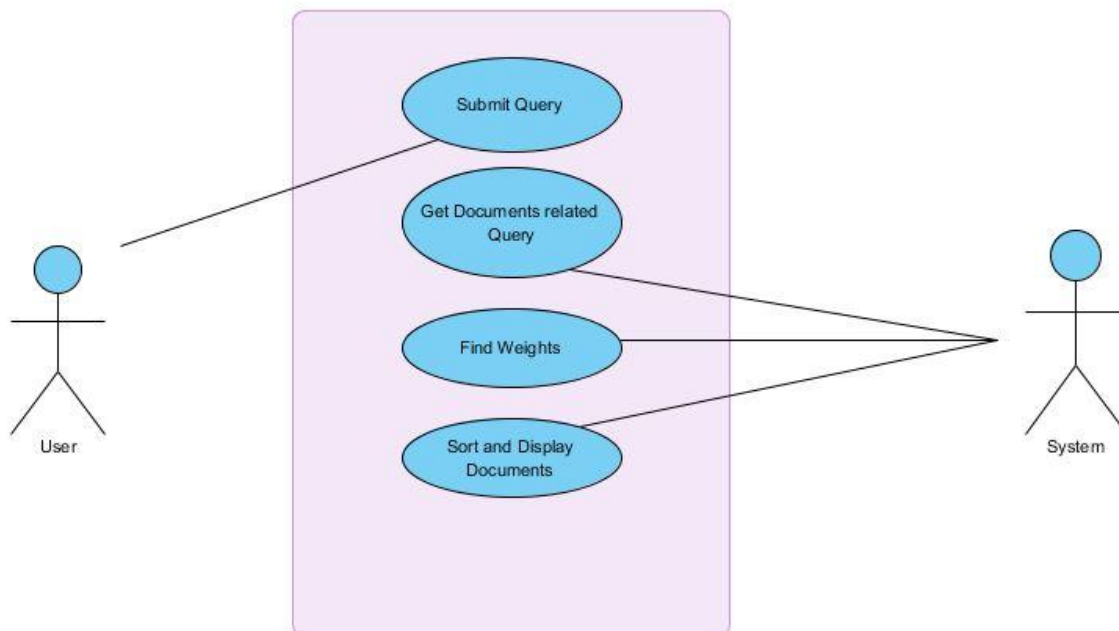
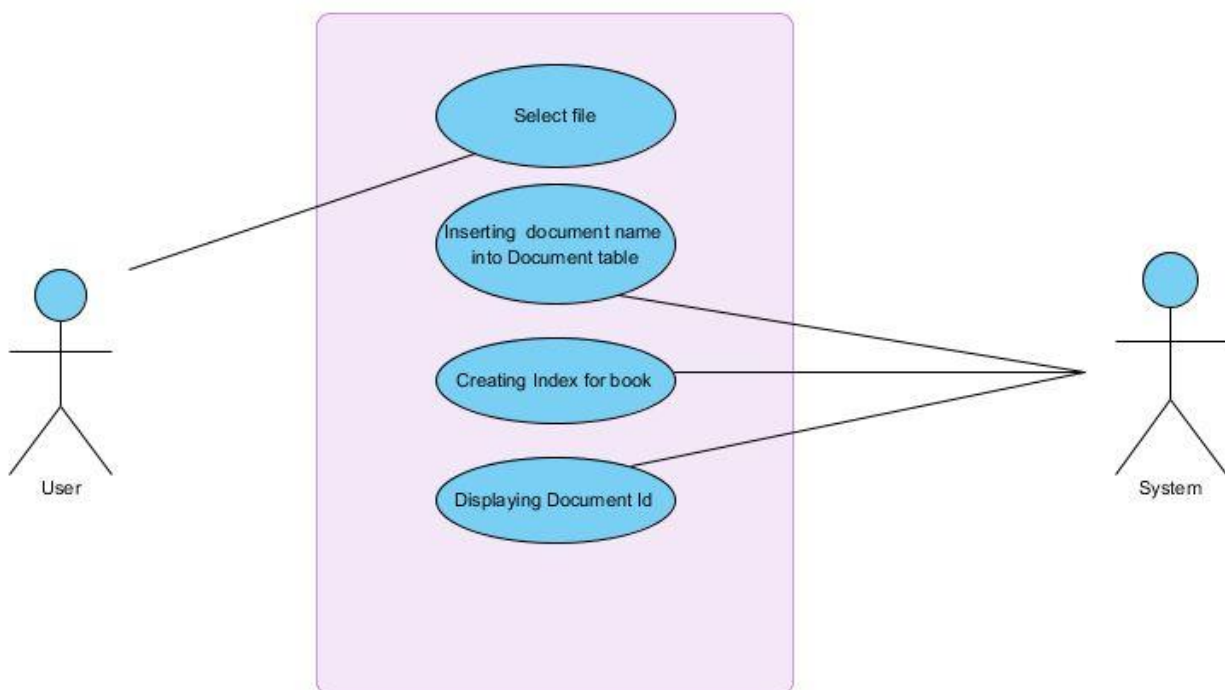
IDENTIFYING USE CASES:

A use case is a description of systems behaviour from a user's standpoint. It is a tried and true technique for gathering systems requirement from a user's point of view.

USE CASE DIAGRAMS

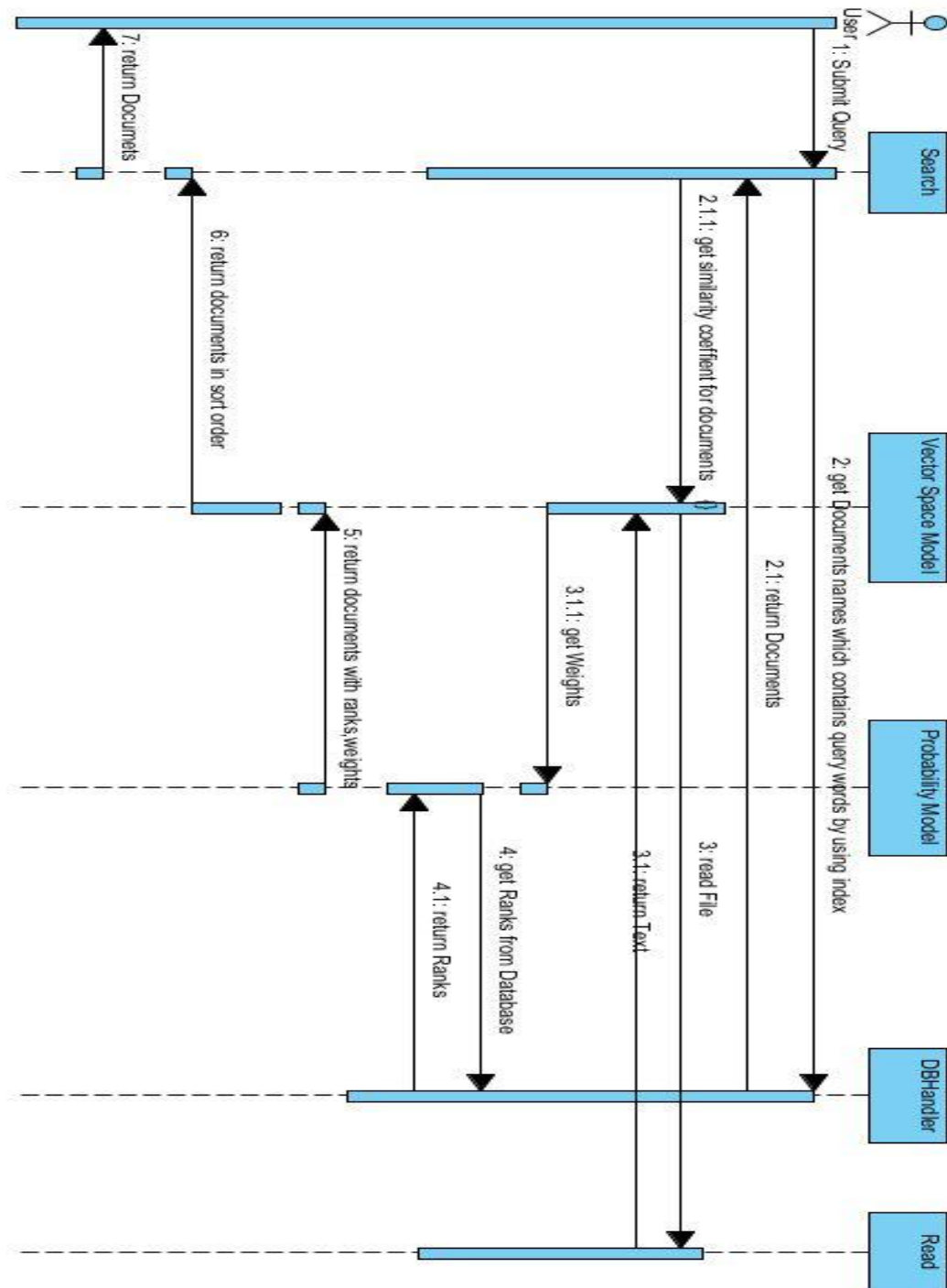
Use case diagram for entire Online Library System:



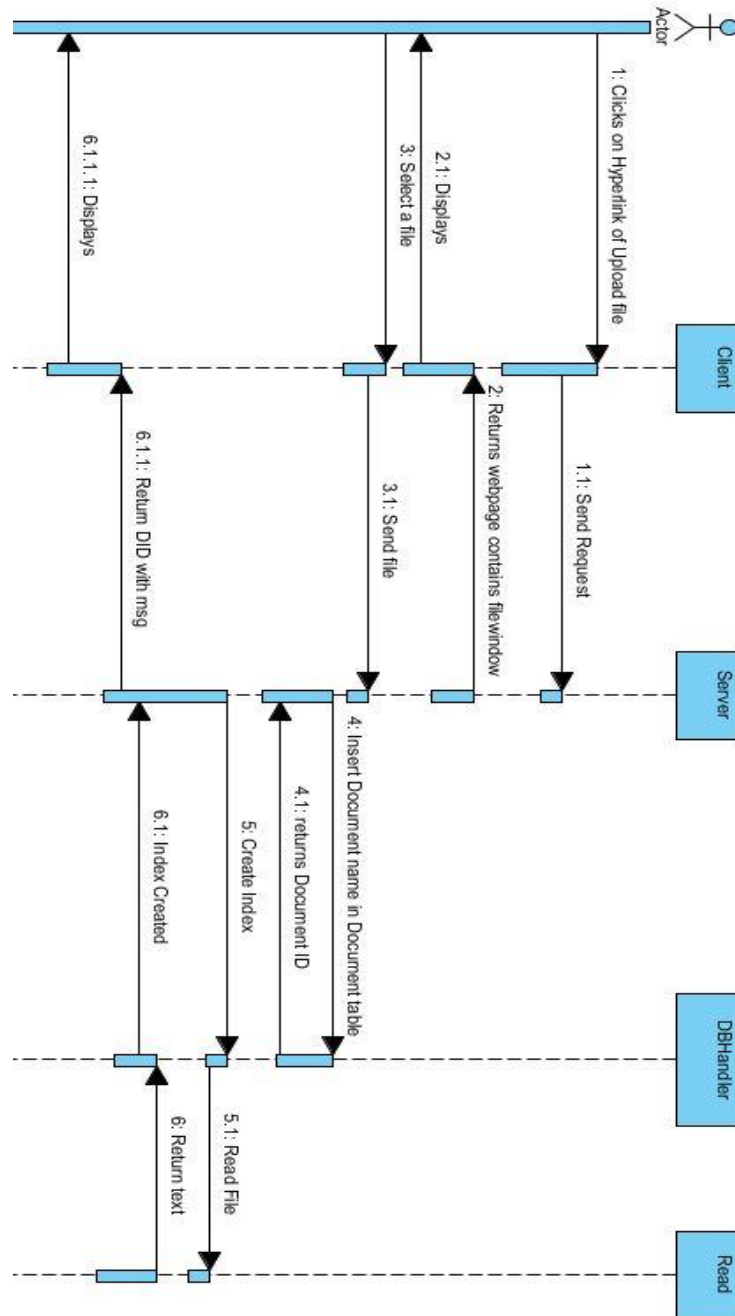
Use Case Diagram for Searching:**Use case Diagram for Uploading file:**

SEQUENCE DIAGRAMS

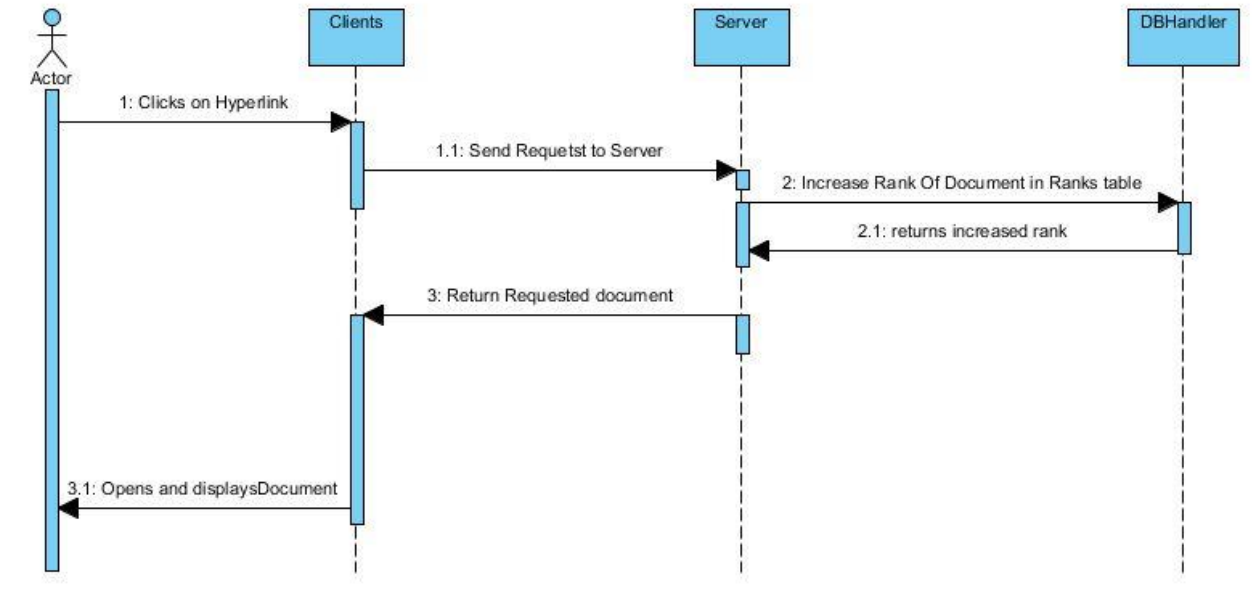
Sequence Diagram for Searching:



Sequence Diagram for uploading file:

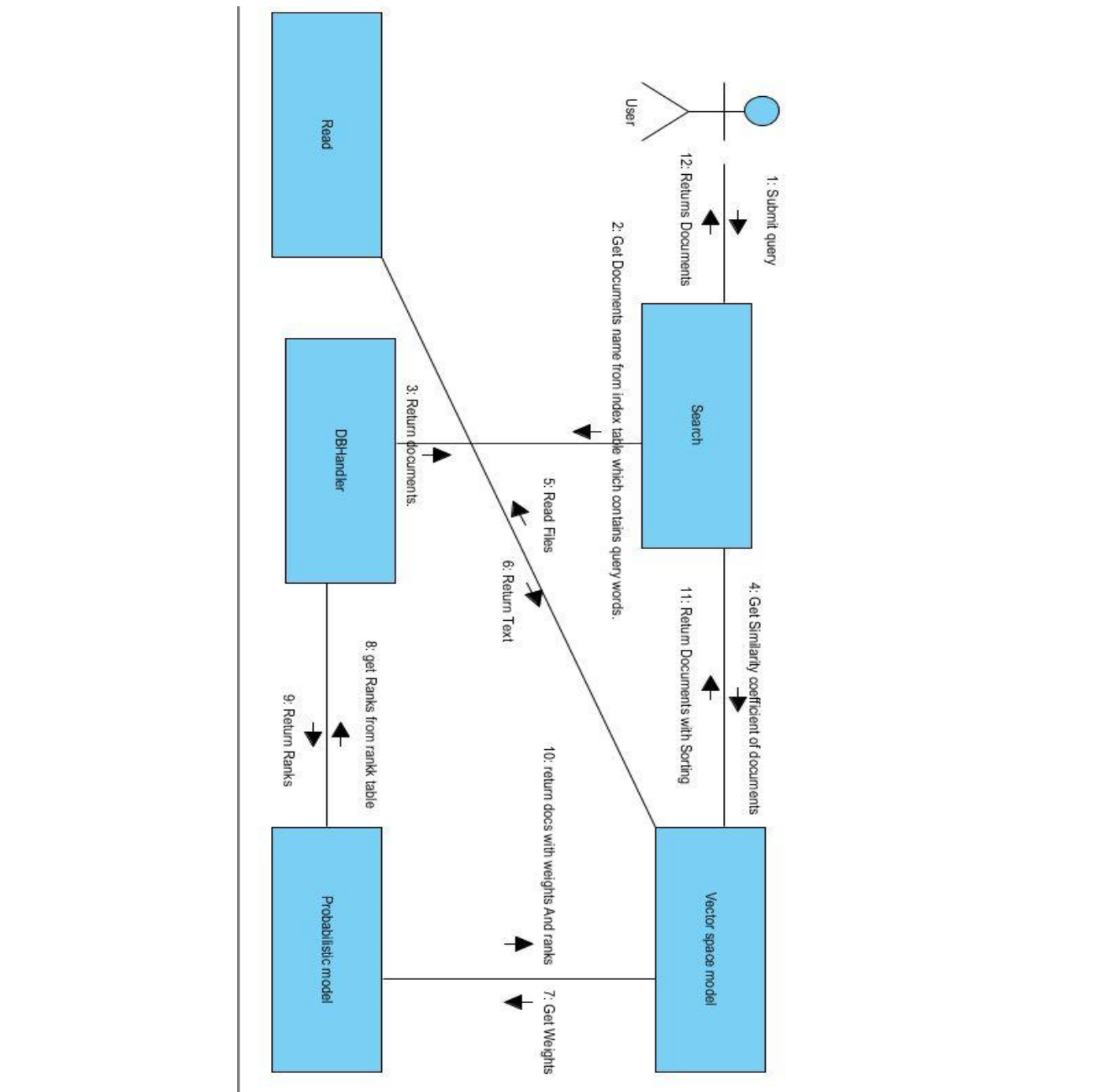


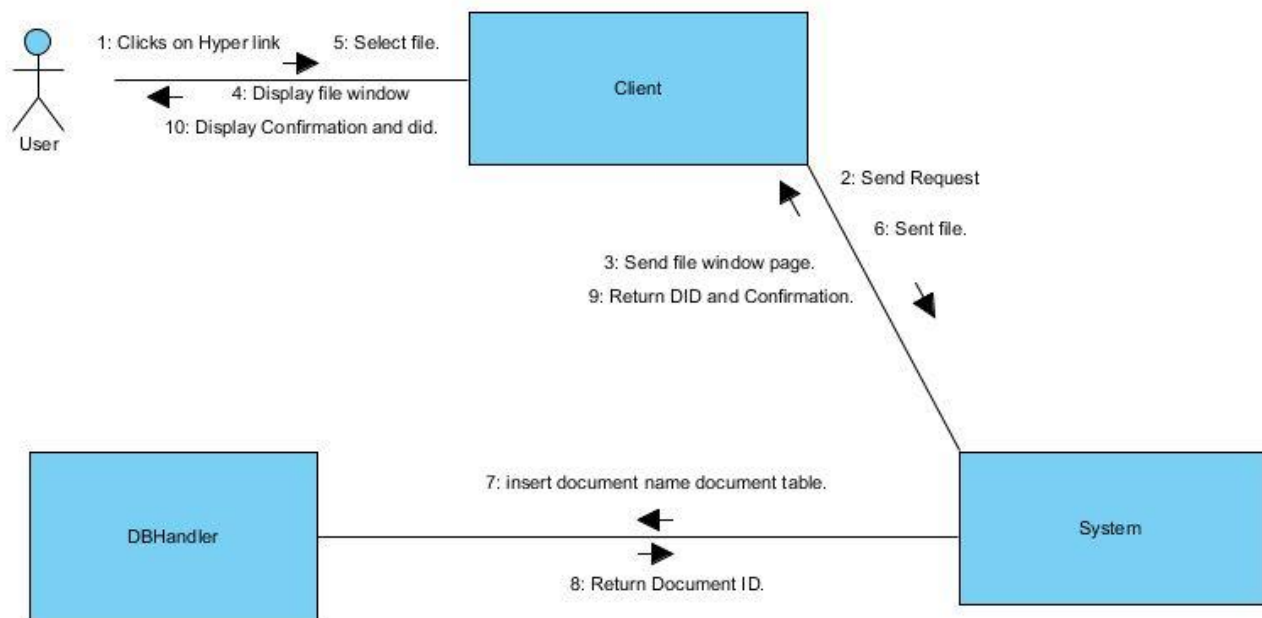
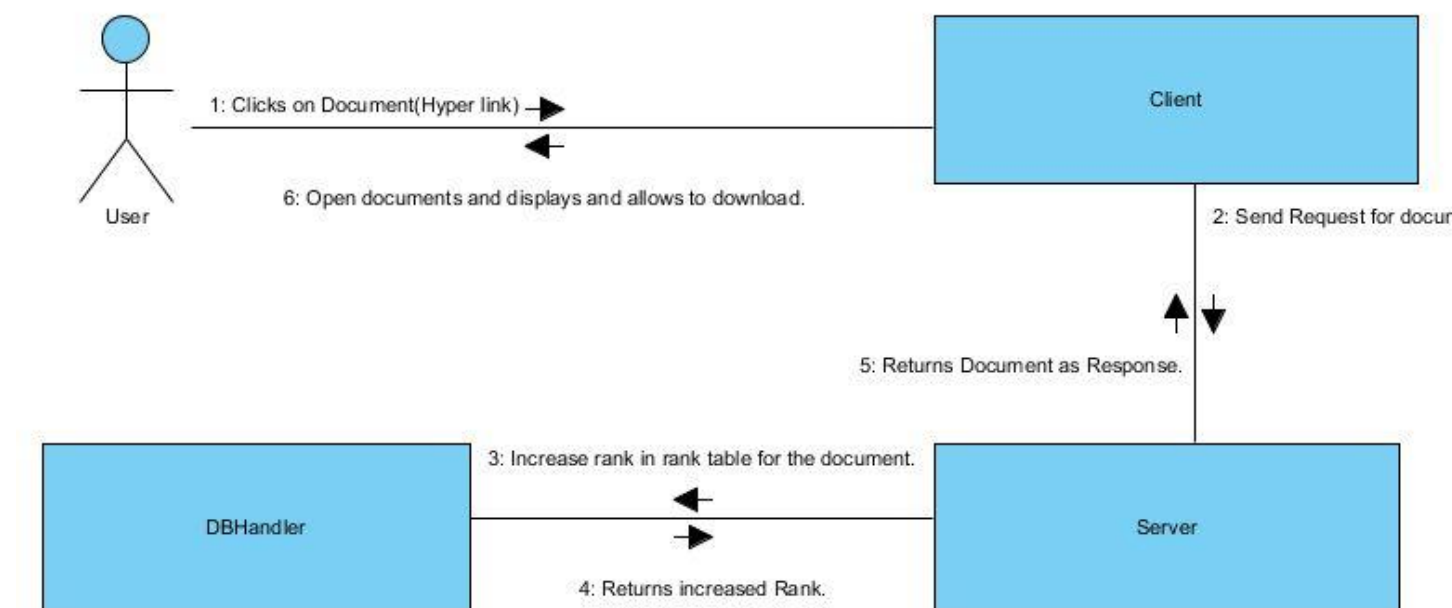
Sequence Diagram for Opening file:



COLLABORATION DIAGRAMS

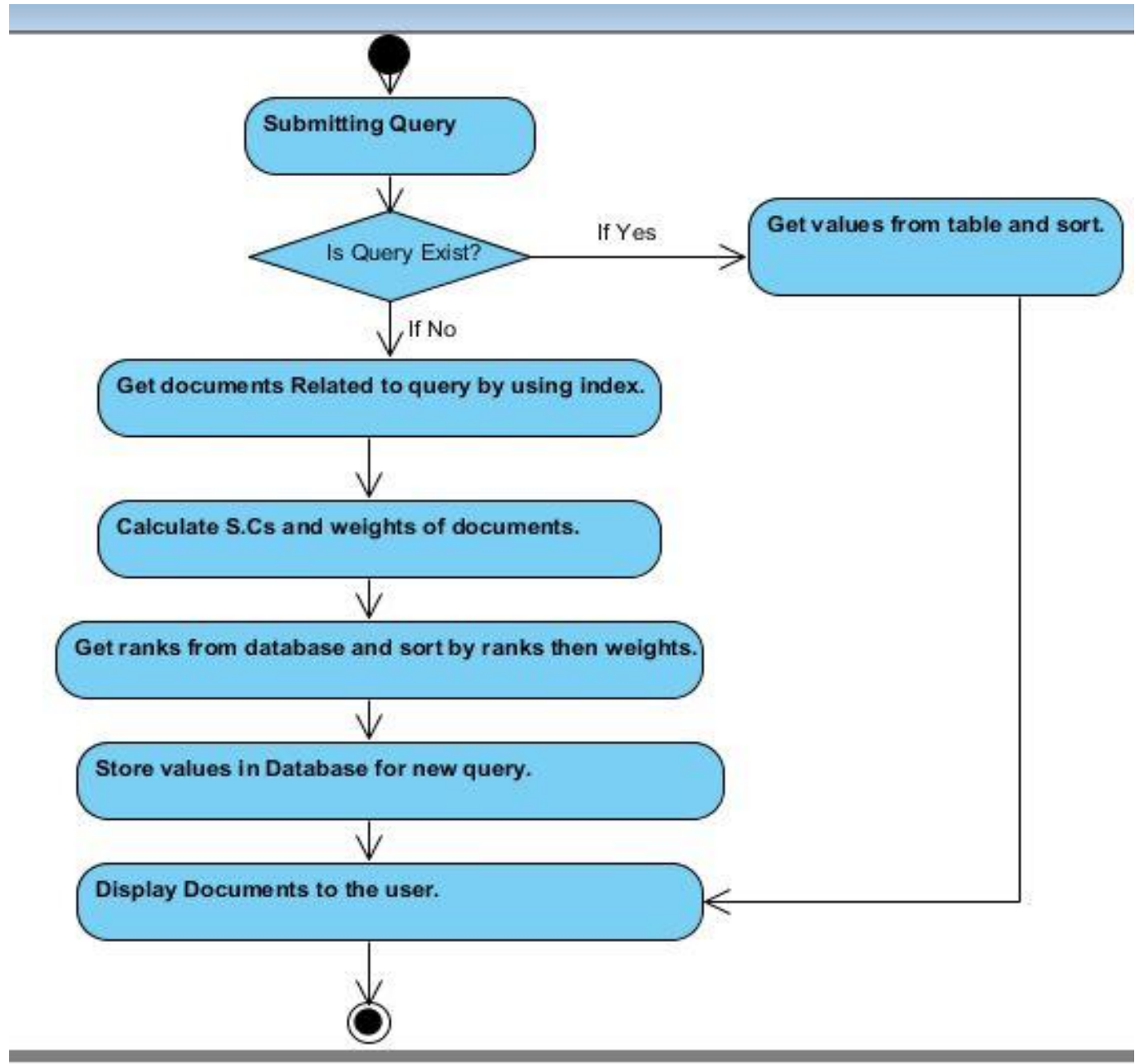
Collaboration Diagram for Searching:

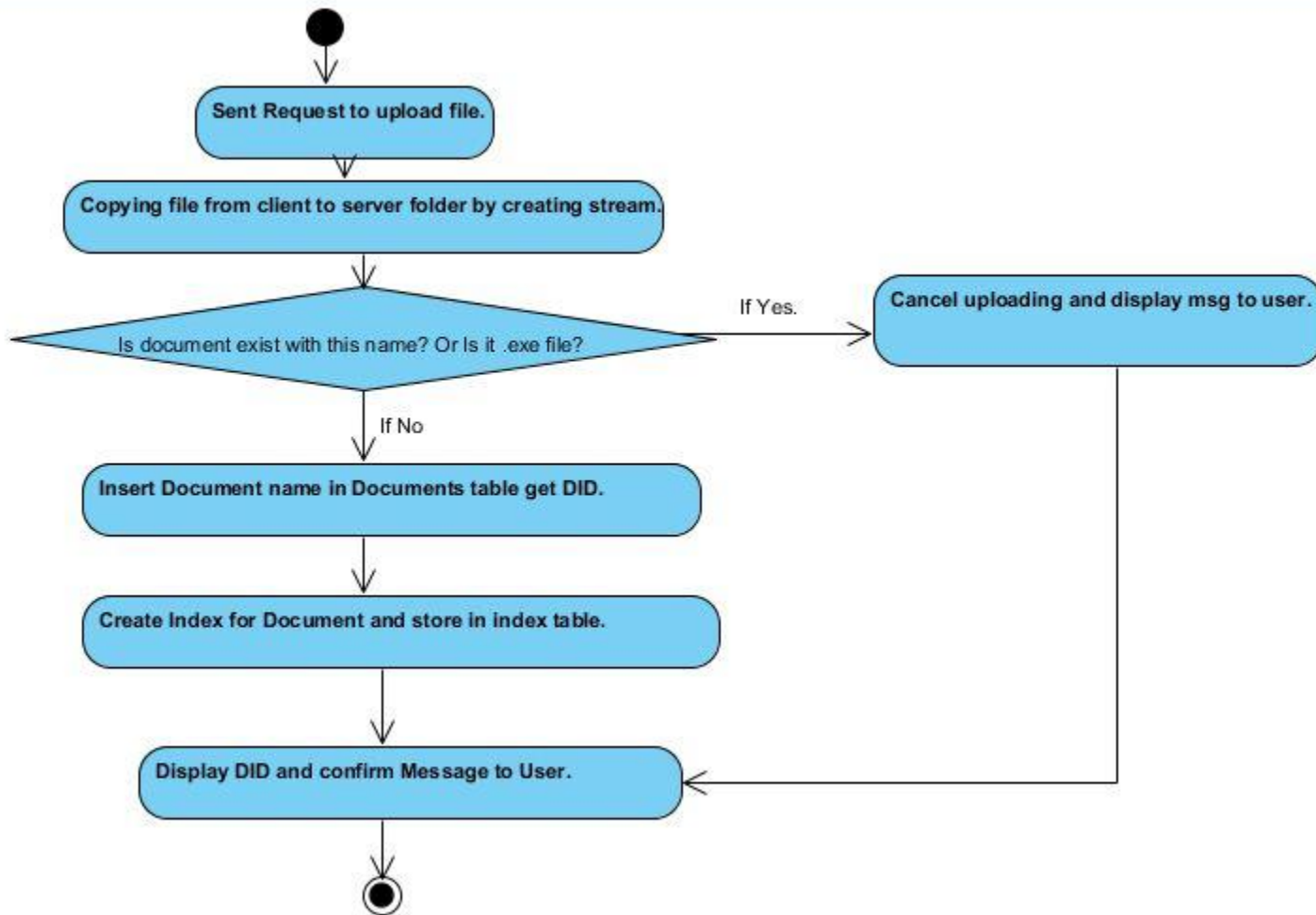


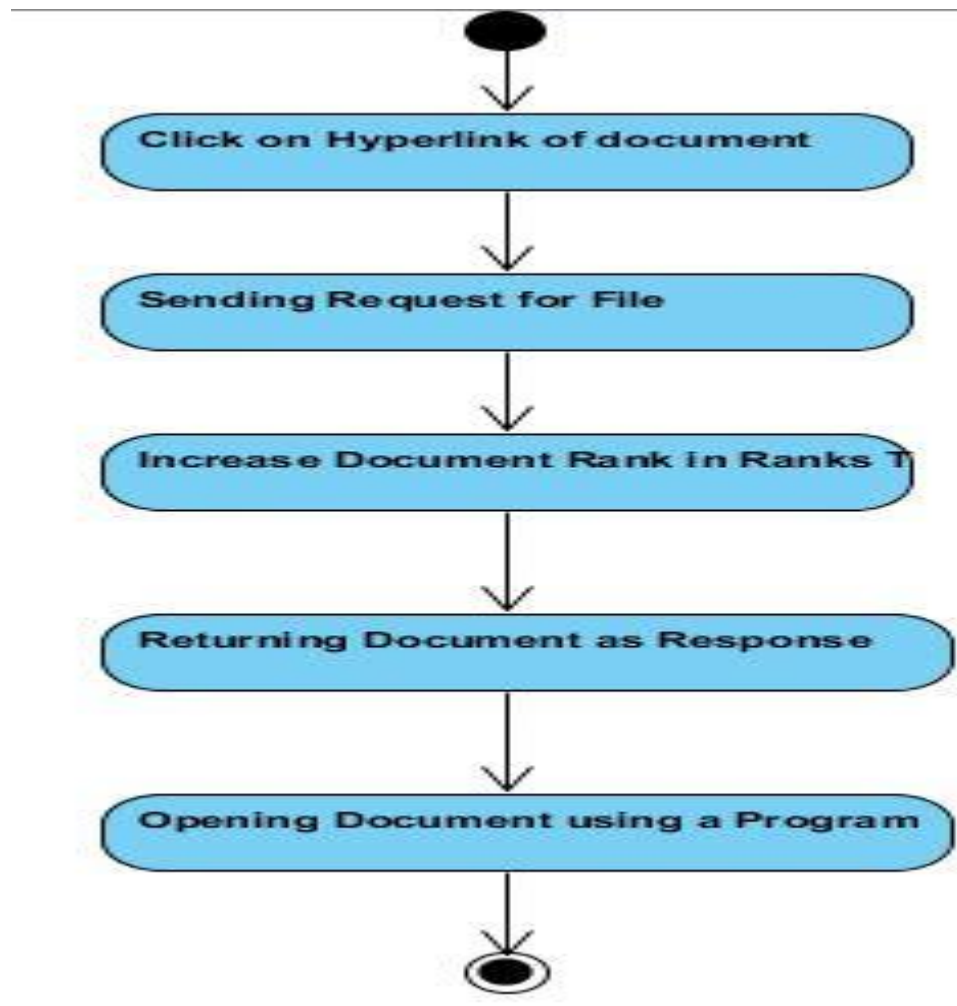
Collaboration Diagram for uploading file:**Collaboration Diagram for Opening file:**

ACTIVITY DIAGRAMS

Activity Diagram for Searching:

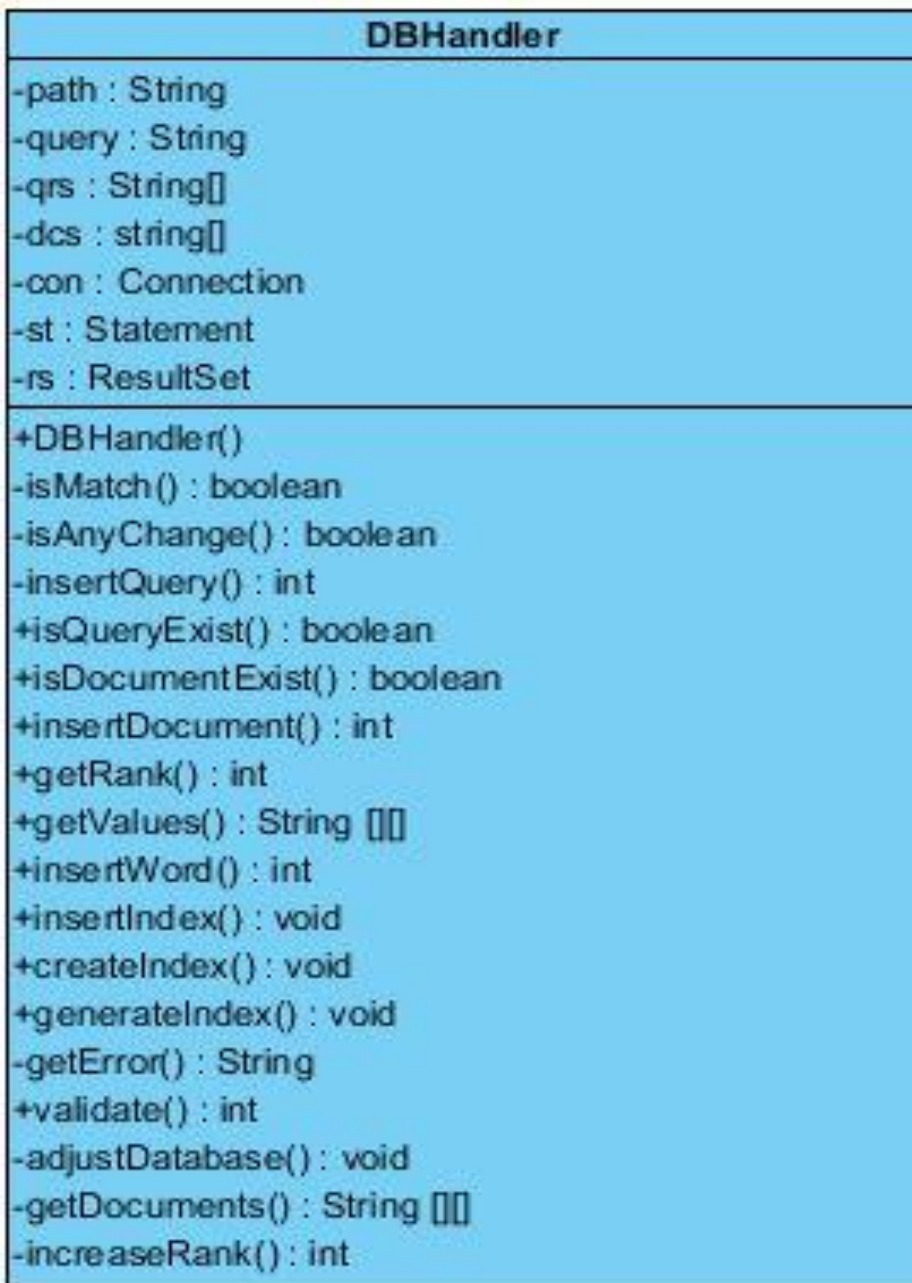


Activity Diagram for Uploading file:

Activity Diagram for Opening file:

CLASS DIAGRAMS

DBHandler Class: DBHandler is responsible for performing all operations which are need interaction with Database. It perform all operation in database like inserting document, creating index, inserting query, mating database consistency , creating tables, recovering database when database crashed. It highly simplifies maintains of database. Adminstrator need not worry about database at any time.



VectroSpaceModel Class: VectorSpaceModel is class it responsible for calculating similarity coefficients of documents with respect to query. After calculating similarity coefficient it passes documents with similarity coefficients to probabilistic model class, which will finds weights using similarity coefficients.



ProbabilisticModel Class: Probabilisticmodel class responsible for calculating weights for documents using similarity coefficients. After calculating weights, it will get ranks for documents from rank table. And return documents to Search class.

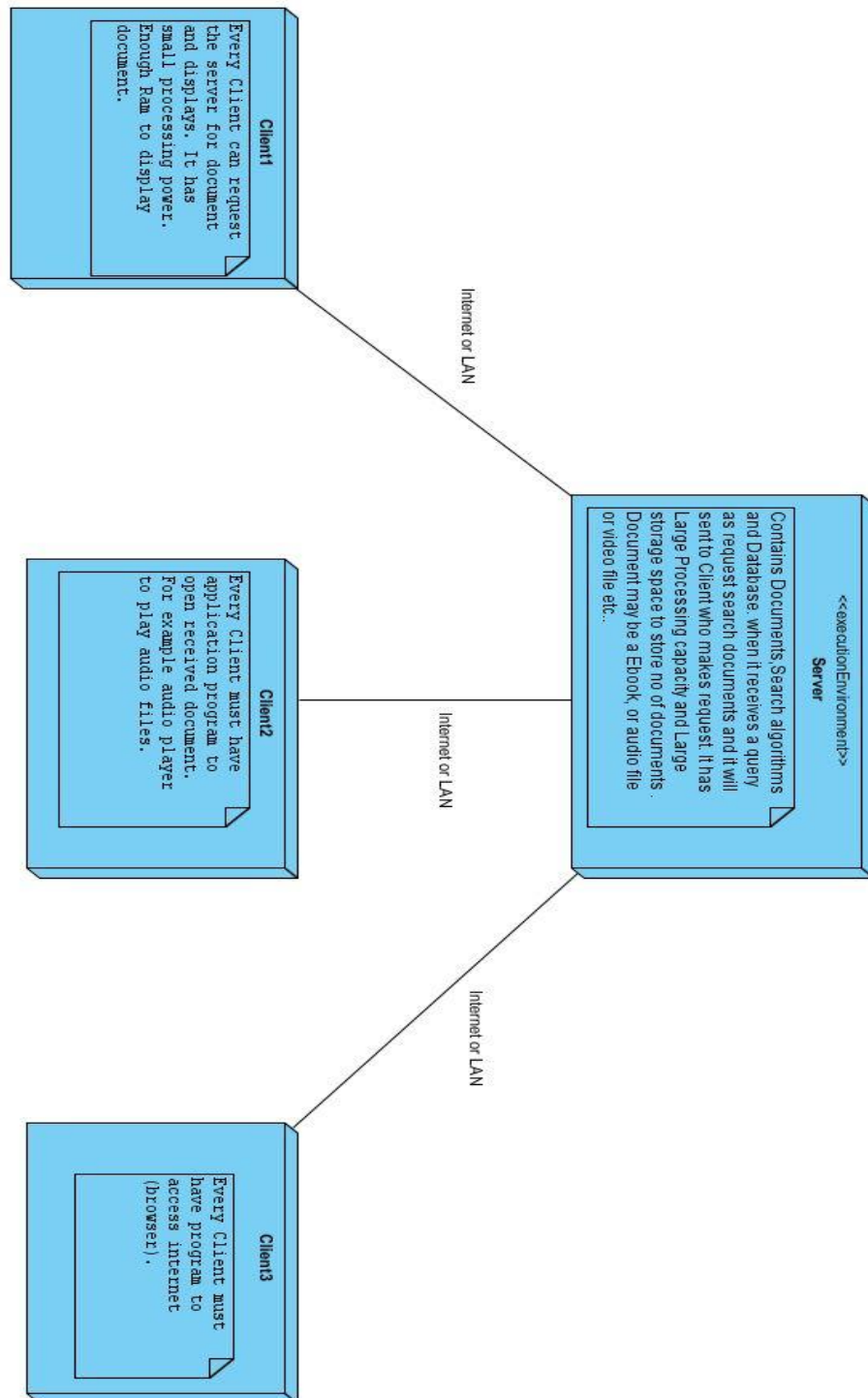
Probabilistic Model
-OD : DBHandler -query : String[] -documents : String[] -fp : file[] -NoofDocuments : int -NoOfRelavancedocs : int -Wiegths : double[] -values : double[]
-sort() : double [] -W1() : double -W2() : double -W3() : double -W4() : double +start() : String [] -ntf() : int -rtf() : int

Read Class: Read class for responsible for reading text from text files. It will read text from different format of text files. For example .doc, .pdf, .xls, .txt, .html etc. It read reads text and clean text and returns as string. If the file is not text file then it will return name and extension of file as string.

Read
-file : File -filename : String
+readFile() : String -readTxt() : String -readPdf() : String -readHtml() : String -readXls() : String -readDoc() : string -cleanText() : String -readCVS() : String

Search Class: Search Class acts as driver class. It does not perform any tasks. It makes all class working together.

DEPLOYMENT DIAGRAM



DESIGN

SOFTWARE STRUCTURE:

Software is fundamental characteristic of computer software. Software structure is to decompose the complex groups of module into sub modules i.e., Process, Menus, Inputs and Reports.

The most general form of the software structure is the network. The structure is the network. The structure inside a complex processing node might consist of concurrent processes executing in parallel and communication through some combination of shared variables.

DATABASE DESIGN:

Planning the Database:

The most important consideration in designing the database is how the information will be used.

- Business activities that will use the database to perform.
- Business rules that apply to these activities
- Data wanted to maintain in the database

The main objectives of designing a database are

- Data integration
- Data integrity
- Data independence

DATA INTEGRATION:

In a database, information from several files is co-ordinate, accessed and operand upon as though it is single file.

Logically, the information is centralized, physical, the data may be located in different devices connected through data communication facilities.

Design Methodology:

Design is concerned with identifying software components, specifying relationships among components, specifying software structure and providing a blue print for the implementation phase.

Design consists of three types:

1. Architectural Design
2. Detail Design
3. External Design

Architectural Design:

Architectural Design involves identifying the software components, decoupling and decomposing them into processing modules and conceptual data structure and specifying relationships among the components.

Detailed Design:

Detailed design is concerned with the details of how to package the processing modules and how to implement the processing algorithms, data structure and interconnection among modules and data structure.

GUI based design:

In order to have a better understanding over the system design, it is appropriate to know the actual designing in terms of the development platform and the way in which the GUI are designed to satisfy the requirements of the user. The analysis of the design aspects of this package is proposed External design. External design of software involves conceiving, planning and specifying the external observable characteristics

of a software product. This includes reports and display formats. External design begins analysis phase it continues into the design phase.

DATA INTEGRITY:

Data integrity means storing all the data in single place and allow each application to access it. This approach results in more consistent, on update being sufficient to achieve a new record status for all the applications, which use it. This leads to less data redundancy, data items need not be duplicated, a reduction in the direct access storage requirement.

DATA INDEPENDENCE:

Data independence is the insulation of application programs from changing aspects of physical data organization. This objective seeks to allow changes in the content and organization of physical data without reprogramming of applications and to allow modifications to application programs without the reorganizing the physical data.

Normalization:

Data structuring is refined through a process called normalization. Normalization is a formal process of developing data structures in a manner that eliminates redundancy and promotes integrity. It is a step-by-step decomposition of complex records into simple records to reduce redundancy, inconsistencies and remove anomalies.

There are several normal forms to be followed in normalization process. The most important and widely used are:

- First Normal Form
- Second Normal Form
- Third Normal Form
-

First Normal Form:

A table is said to be in first normal form if the intersection of any column and row contains only value.

Method:

This is identifying a suitable identifier from the pool of normalized data. Remove any item that repeat within a single value of this key to another relation bringing with them the identifier key to form part of new composite key in the relation.

Second Normal Form:

For a table to be in the second normal form it should also be in the first normal form and the values in every column are functionally dependent on the complete primary key.

Method:

Examine every column and section whether its value depend on the whole of the compound key or just some parts of it. Remove those that depend only on part of the key to a new table with that part as the primary key.

Third Normal Form:

For a table to be in the third normal form it should be in the second normal form and the values in every non-key column are not transitively dependent on the primary key.

Method:

Examine every non-key column with every other non-key column. If the value of the non-key column depends on the other non-key column then remove the columns to separate table.

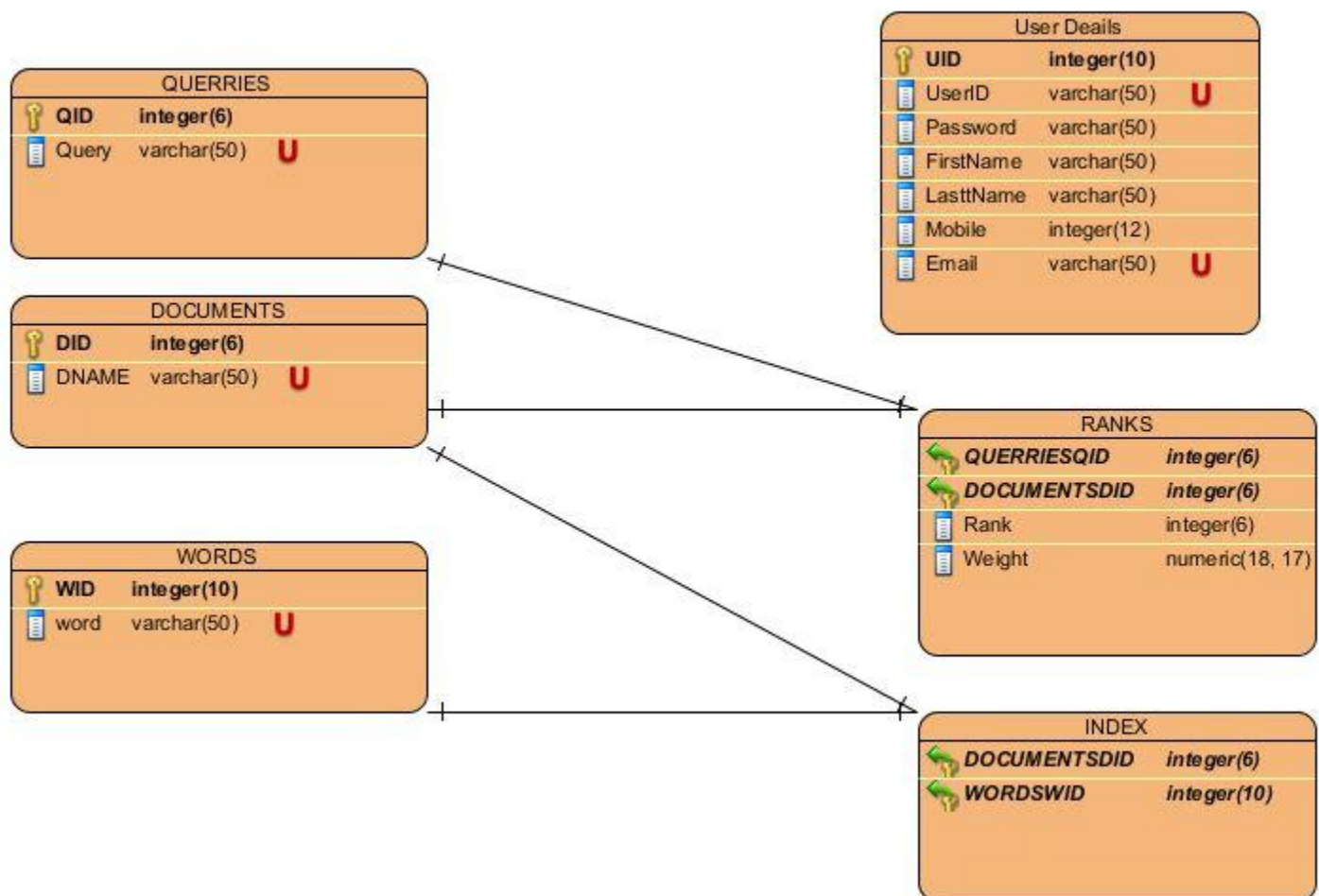
De Normalization:

The normalization process helps in reducing to a great extent but sometimes when information is required from more than one table, at a fast rate, it is desirable to have some degree of redundancy in table. Their deliberate introduction of redundancy for a highly improved performance is referred to as "De Normalization".

DATABASE TABLES:

1. **QUARRIES:** The QUARRIES table can store queries entered by user with unique ID (QID). It stores unique queries irrespective of order of words and case of letters.
2. **DOCUMENTS:** The DOCUMENTS table can hold name of documents, which are in server with unique ID for every document.
3. **WORDS:** The WORDS table can store words, which are in documents. This table plays key role in creating index for documents. This table contains a unique ID and words column. Unique words can be stored in this table irrespective of case of letters.
4. **RANKS:** The RANKS table can hold rank and weight for each document with respect to query. This table contains four columns. QID and DID are refer QID of QUERRIES and DID of DOCUMENTS table respectively.
5. **INDEX:** The INDEX table helps to search engine while searching for documents, which are relate to query given by user. Simply it makes fast Searching. It contains two columns WID, DID refer WID of WORDS, DID of DOCUMENTS tables.
6. **USERDETAILS:** The USERDETAILS can hold details of users. Username, user-id, password etc.

ENTITY RELATIONSHIP DIAGRAMS



ABOUT THE SOFTWARE

INTRODUCTION:

HTML stands for hyper text makeup language. It is a language used to create hypertext documents that have hyperlinks embedded in them. You can build web pages. It is only a formatting language and not a programming language. Hyperlinks are underlined or emphasized words or locations in a screen that leads to other documents, W W W is a global, Interactive, dynamic, cross Platform, graphical hypertext information system. The idea behind hypertext is that instead of reading text in rigid linear structure you can easily jump from one point to another. You can Navigate through the information based on your interest and preferences.

HYPERMEDIA

HTML pages with audio and video files linked to them are called Hypermedia. HTML is platform independent.

HTML IS PLATFORM INDEPENDENT:

If you can access internet, you can access WWW, Irrespective of your operating system and the operating system to the web server. All you require to view and unload the HTML files, which are on the WWW, are a browser and internet connection.

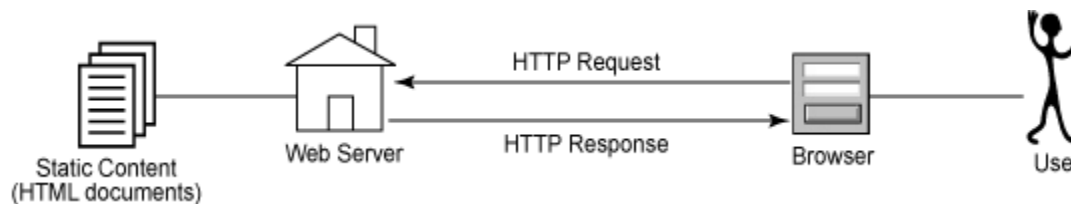
WEB IS DISTRIBUTED:

The information on the WWW is distributed through out the world. You can access it with a few mouse clicks. You do not have to store it on your machine. Information on the Web is dynamically updateable. As the information is at the site where it is published, the people who publish it can change it any time.

HTML is a language for describing structured documents, HTML describes the structures of documents - lists, headings, paragraphs Etc. Elements of web documents are labeled through the usage of HTML tags, It is the tags that describe the documents. Anything that is not a tag part of the Document itself.

The Static Web

When the World Wide Web was born, it consisted of static Web sites. Web sites consisted of static documents (mostly HTML pages) that were accessible through Web servers. Users used Web browsers to access these Web sites. A Web browser communicated with a Web server over the Hyper Text Transfer Protocol (HTTP). Using an HTTP request, the browser communicated with the Web server and asked to get access to a certain document. The Web server managed a set of documents stored on a file system. When asked for a document, it would retrieve it and return it within an HTTP response:



HTML DOES NOT DESCRIBE PAGE LAYOUT:

World for windows or lotus improve for example, have different styles for Headings, font, Size, and identification. HTML does not have all these. Based on the plat form , exact placement or appearance of any element will change. There may be or may not be fonts installed. By separating the structure of the document and its appearance, a

program, that reads and understands HTML can make formatting decisions

based on capabilities of the individual platform . In addition to providing the

networking functions to retrieve documents, Web browsers are also HTML formatters.

They parse and format documents and display them on the screen. Different rowers

show data differently.

ADVANTAGES :

A HTML document is small and hence easy to send over the net. It is small because it does not include format information. HTML documents are cross platform compatible and device independent. You only need a HTML reliable browser to view them. Font names, locations Etc.. are required. Currently the standard fully supported is HTML 2.0 & HTML 3.0 is in use. It supports. Centered and right aligned text.

Tables .

Math equations Text and image alignment

JAVA

INTRODUCTION

Java is an intercepted language. Though it bears a close resemblance to C++ , it is different from it in many ways. It is a smaller, portable, purely object oriented Language that eliminates many of the sources of bugs and complexities that are common with C Or C++ .

JAVA FEATURES:

Java is simple, Object Oriented, Intercepted, Robust, Secure Architecture neutral, Portable, has high performance, multi threaded and dynamics. It is easier To use Java because it's syntax is similar to C and C++, more so when it eliminates components of C that causes bugs and memory leaks and hence a lot less debugging, Java provides a powerful Set of pre-tested libraries that give us the ability to use advanced features even with few lines of Code.

OBJECT ORIENTED

Java is an Object oriented programming Language that uses software objects called CLASSESS and it is based on reusable, extensible code. This implies we can use Java's classes, which are sets of variables and methods, as

templates to create other Classes with added functionality without the need to write the code from Scratch. The Application can be made small and easy to develop if we plan the class hierarchy well.

Robust:

Java is Robust because, the language removes the use of pointers and The Java Runtime System manages the memory for us. Java gives us automatic bounds checking for arrays, so that they cannot reside in an address space which is not allocated for them. Automatic memory management is taken care of by the Garbage Collector.

Interpreted:

Java is interpreted; hence the development cycle is much faster. Java needs to compile for a single, virtual machine and then the code can run on any machine that has JVM ported to it.

Secure:

Java is secure, so user can download Java programs from anywhere. Java provides extensible compile time checking followed by a multi layered level of runtime Checking.

Architecture Neutral:

Java is architecture neutral, so user applications are portable across multiple platforms. Java's Applications are written and compiled into Byte Code for JVM, Which emulates an actual hardware chip. The Java Interpreter installed at the client, so applications need not be written for various platforms separately converts byte Code to machine code. Java further ensures that the applications are the same on every platform by strictly defining the sizes of the basic data types and their behaviour.

Dynamic:

Java is Dynamic., so the applications are adapted to changing environments, Java's architecture allows the user to dynamically load classes at runtime from any where on the network, which means that the user can add functionality to applications by simply linking in new classes.

JDBC OVERVIEW

What is JDBC?

JDBC is a Java TM API for executing SQL statements. It consists of a set of classes and interfaces written in the JAVA programming language that makes it easy to send SQL statements to virtually any relational database. In other words, with the JDBC API, it isn't necessary to write one program to access a Sybase database, another program to access an Oracle database, another program to access an Informix database, and so on. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. And, with a program written in the JAVA programming language, one doesn't have to worry about writing different programs to run on different platforms. The combination of JAVA and JDBC lets a programmer write it once and run it anywhere.

JAVA, being robust, secure, easy to use, easy to understand, and automatically downloadable on a network, is an excellent language basis for database applications. What is needed is a way for JAVA applications to talk to variety of different databases. JDBC is the mechanism for doing this. JDBC extends what you can do in JAVA. For example, with JAVA and the JDBC API, it is possible to publish a web page containing an applet that uses information obtained from a remote database. Or an enterprise can use JDBC to connect all its employees (Even if they are using a conglomeration of Windows, Macintosh and Unix machines) to one or more internal databases via. An Internet. With more and more programmers using the JAVA programming language, the need for easy database access from

JAVA is continuing to grow. MIS managers like the combination of JAVA and JDBC because it makes disseminating information easy and economical. Business can continue to use their installed databases and access information easily even if it is stored on different database management systems. Development time for new applications is short. Installation and version controls are greatly simplified. A programmer can write an application or an update once, put it on the server and everybody has access to the latest version and for business selling information services, JAVA and JDBC offers better way of getting out information updates to

external customers. Various ways to use JDBC are discussed in.

What Does JDBC Do?

Simply put, JDBC makes it possible to do three things:

Establishes connection to databases.

Send SQL statements.

Process the results.

JDBC is a Low-level API and a base for Higher-Level API.

JDBC is a "Low-level" interface, which means that it is used to invoke SQL commands directly. It works very well in this capacity and is easier to use than other database connectivity APIs, but it was designed also to be a base upon which to build higher-level interfaces and tools. A higher-level interface is "User-friendly", using a more understandable or more convenient API that is translated behind the scenes into a Low level interface such as JDBC.

So why not just use ODBC from Java?

The answer is that you can use ODBC from JA V A, but this is best done with the help of JDBC in the form of the JDBC-ODBC Bridge. The question now becomes

"Why do you need JDBC?" There are several answers to this question:

1. ODBC is not appropriate for direct use from Java because it uses a C interface.
2. Calls from Java to native C code have a number of drawbacks in the security,
 1. implementation, robustness, and automatic probability of applications.

A literal translation of the ODBC API into a Java API would not be desirable. For example, Java has no pointers, and ODBC makes copious use of them, including the notoriously error-prone generic pointer "void *".

You can think of JDBC as ODBC translated into an object-oriented interface that is natural for Java programmers.

2. ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries. JDBC, on the other

hand, was designed for a wide range of programmers and keeps simple things simple.

3. A Java API like JDBC is needed in order to enable an "all-Java" solution. When ODBC is used, the driver manager and drivers must be manually installed on every client machine. When the JDBC driver is written completely in JAVA, however, JDBC code is automatically installed, portable, and secure on all Java platforms from network computers to mainframes. In summary, the JDBC API is natural JAVA interface is to the basic SQL abstractions and concepts. It builds an jdbc rather than starting from scratch, so programmers familiar with jdbc will find it very easy to learn JDBC. JDBC retains the basic design features of ODBC; in fact, both interfaces are based on the x/Open SQL CU(call level interface).The big difference is that JDBC builds on and reinforces the style and virtues of Java, and of course, it is easy to use.

ADVANTAGES.

Until now, the middle tier has typically been written in languages such as C or C++, which offer fast performance. However, with introduction of optimizing compilers, translating Java byte code into efficient machine-specific code, it is becoming practical to implement the middle tier in Java. This is a big plus, making it possible to take advantage of Java's robustness, multi-threading and security features.

SQL Conformance

Structured Query Language(SQL) is the standard language for accessing relational databases. One area of difficulty is that almost most DBMSs(Database Management Systems) gives a standard form of SQL for basic functionality, they do not conform to the more recently defined standard SQL syntax or semantics for more advanced functionality. For example, not all databases support stored procedures or outer joins, and those that do are not consistent with each other. It is hoped that the portion of SQL that is truly standard will expand to include more functionality. In the mean time, however, the JDBC API must support SQL as it is. One way the JDBC API deals with the problem is to allow any query string to be passed through to an underlying DBMS driver. This means that an application is free to use as much SQL functionality as is desired, but it runs the risk of receiving an error on some DBMS. In fact an application query need not even be an SQL, or it may be a specialized derivation of SQL designed for specific DBMS(for document or image queries). For example, a second way JDBC deals with the problems of SQL conformance is to provide ODBC style escape clauses. Which are discussed in SQL Escape Syntax in Statement objects. This escape syntax allows a programmer to use SQL

functionality item within a JDBC program. The ODBC API is a natural choice for Java developers because it offers easy database access for Java applications and applets. Because JDBC brings together Java and databases, the remainder of the topic gives a brief overview of each.

JDBC DRIVERS

The JDBC drivers that we are aware of at this time fit into one of four categories.

1. JDBC-ODBC bridge plus ODBC Driver:

The JavaSoft bridge product provides JDBC access via ODBC drivers. Note that ODBC binary code and in many cases database client code must be loaded on each client machine that uses this driver. As a result, this kind of driver is more appropriate on a corporate network where client installations are not a major problem, or for applications server code written in Java in a three-tier architecture.

2. Native-API partly - JAVA Driver:

This kind of driver converts JDBC calls into calls on the client api for Oracle, sybase, Informix, DB2, or other DBMS. Note that, like the bridge driver this style of driver requires that some binary code be loaded on each client machine.

3. JDBC-Net all-- JAVA Driver:

This driver translates JDBC calls into a DBMS independent net protocol, which is then translated to a DBMS protocol by a server. This net server middleware is able to connect its all-Java clients to many different databases. The specific protocol used depends on the vendor. In general, this is the most flexible JDBC alternative. It is likely that all vendors of this solution will provide products suitable for internet use. In order for these products to also support internet access, they must handle the additional requirements for security, access through fire walls, etc., that the web imposes. Several vendors are adding JDBC drivers to their existing database middleware products.

4. Native - Protocol all-Java Driver:

This kind of driver converts JDBC calls into the network protocol used by dbms's directly. This allows a direct call from the client machine to the dbms server and is a practical solution for internet access. Since many of

these protocols are proprietary, the database vendors themselves will be the primary source. Several database vendors have these in progress. Eventually, we expect that categories 3 and 4 will be the preferred way to access databases from JDBC. Driver categories 1 and 2 are in term solutions where direct all-Java drivers are not yet available. Category 4 is in some sense the ideal; however, there are many cases where category 3 may be preferable: e.g., where a thin dbms in dependent client is desired, or if a dbms-independent protocol is standardized and implemented directly by many dbms vendors.

SERVLETS

Introduction:

The Java web server is JavaSoft's own web Server. The Java web server is just a part of a larger framework, intended to provide you not just with a web server, but also with tools. To build customized network servers for any Internet or Intranet client/server system. Servlets are to a web server, how applets are to the browser.

About Servlets:

Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions, platform-specific APIs, and incomplete interfaces.

Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server-side what applets are to the client-side - object byte codes that can be dynamically loaded off the net. They differ from applets in that they are faceless objects (without graphics or a GUI component). They serve as platform independent, dynamically loadable, pluggable helper byte code objects on

the server side that can be used to dynamically extend server-side functionality.

For example, an HTTP Servlets can be used to generate dynamic HTML content. When you use Servlets to do dynamic content you get the following advantages:

- They're faster and cleaner than CGI scripts
- They use a standard API (the Servlets API)
- They provide all the advantages of Java (run on a variety of servers without needing to be rewritten).

Attractiveness of Servlets:

There are many features of Servlets that make them easy and attractive to use. These include:

- Easily configured using the GUI-based Admin tool
- Can be loaded and invoked from a local disk or remotely across the network.
- Can be linked together, or chained, so that one Servlets can call another Servlets, or several Servlets in sequence.
- Can be called dynamically from within HTML pages, using server-side include tags.
- Are secure - even when downloading across the network, the Servlets security model and Servlets sandbox protect your system from unfriendly behavior.

Advantages of Servlet API

One of the great advantages of the Servlet API is protocol independence. It assumes nothing about:

- The protocol being used to transmit on the net

- How it is loaded
- The server environment it will be running in
- These qualities are important, because it allows the Servlet API to be embedded in many different kinds of servers. There are other advantages to the Servlet API as well. These include:
 - It's extensible - you can inherit all your functionality from the base classes made available to you.
 - it's simple, small, and easy to use.

Features of Servlets:

- Servlets are persistent. Servlet are loaded only by the web server and can maintain services between requests.
- Servlets are fast. Since Servlets only need to be loaded once, they offer much better performance over their CGI counterparts.
- Servlets are platform independent.
- Servlets are extensible. Java is a robust, object-oriented programming language, which easily can be extended to suit your needs
- Servlets are secure.
- Servlets can be used with a variety of clients.

Loading Servlets:

Servlets can be loaded from three places

From a directory that is on the CLASSPATH. The CLASSPATH of the JavaWebServer includes service root/classes/ which is where the system classes reside.

From the <SERVICE_ROOT /Servlets/ directory. This is **not** in the server's classpath. A class loader is used to create Servlets from this directory. New Servlets can be added - existing Servlets can be recompiled and the server will notice these changes.

From a remote location. For this a code base like `http: // nine.eng / classes / foo /` is required in addition to the Servlets class name. Refer to the admin GUI docs on Servlet section to see how to set this up.

Loading Remote Servlets:

Remote Servlets can be loaded by:

1. Configuring the Admin Tool to setup automatic loading of remote Servlets
2. Setting up server side include tags in .shtml files
3. Defining a filter chain configuration

I n v o k i n g S e r v l e t s

A Servlet invoker is a Servlet that invokes the "service" method on a named Servlet. If the Servlet is not loaded in the server, then the invoker first loads the Servlet (either from local disk or from the network) and then invokes the "service" method. Also like applets, local Servlets in the server can be identified by just the class name. In other words, if a Servlet name is not absolute, it is treated as local.

A client can invoke Servlets in the following ways:

- The client can ask for a document that is served by the Servlet.
- The client (browser) can invoke the Servlet directly using a URL, once it has been mapped using the Servlet Aliases section of the admin GUI.
- The Servlet can be invoked through server side include tags.
- The Servlet can be invoked by placing it in the Servlets/ direct

TESTING

Test Plan:

Testing the newly developed or modified systems is one of the most important activities in the system development methodology. The goal of testing is to verify the logical and physical operation of the design blocks to determine that they operate as intended, or testing the system we planned and followed the strategies given below.

Unit testing:

Unit or module testing is the process of testing the individual components (subprograms or procedures) of a program. The purpose is to discover discrepancies between the modules interface specification and its actual behavior.

In our system each module namely

- (i) Community admin module
- (ii) Community user module
- (iii) Super Admin module must be tested independently for validation.

Integration Testing:

Integration testing is the process of combining and testing multiple components together. The primary objective of integration testing is to discover errors in the interfaces between the components. In our system each of the modules mentioned above, are tested for checking the integration between them, after each of them are tested individually.

System Testing:

Validation testing provides the final assurance that software meets all functional, behavioural and performance requirement. The software once validated must be combined with other system elements. System testing verifies that as elements combine properly and that overall system function and performance is achieved.

Integration and System Testing:

Integration Testing:

In this the different modules of a system are integrated using an integration plan. The integration plan specifies the steps and the order in which modules are combined to realize the full system. After each integration step, the partially integrated system is tested. The primary objective of integration testing is to test the module interface.

An important factor that guides the integration plan is the module dependency graph. The module dependency graph denotes the order in which different modules call each other. A structure chart is a form of a module dependency graph. Thus, by examining the structure chart the integration plan can be developed based on any of the following approaches:

- Big-bang approach.
- Top-down approach.
- Bottom-up approach.
- Mixed approach

Bottom-up Integration Testing:

In this approach, each subsystem is tested separately and then the full system is tested. A subsystem might consist of many modules which communicate among each other through well-defined interfaces. The primary purpose of testing each subsystem is to test the interfaces among various modules making up the subsystem. Both control and data interface are tested. A principal advantage of bottom-up integration testing is that several disjoint subsystems can be tested simultaneously. A disadvantage of bottom-up testing is the complexity that occurs when the system is made up of large number of small subsystems.

SYSTEM TESTING:

Once we are satisfied, that all the modules work well in themselves and there are no problems, we do in to how the system will work or perform once all the modules are put together. The main objective is to find discrepancies between the system and its original objective, current specifications, and system documentation. Analysts try to find moulds that have been designed with different specifications, which could cause incompatibility.

At this stage the system is used experimentally to ensure that all the requirements of the user are fulfilled. At this point of the testing takes place at different levels to ensure that the system is free from failure. Testing is mostly performed by persons who have never worked with the system before, so that the feedback we get is free from bias.

Testing is vital to success of the system. System testing makes a logical assumption that whether all parts of the system are correct. Initially the system was given to the user for entry validation was provided at each and every stage. So that the user is not allowed to enter unrelated data. The training is given to user about how to make an entry.

While implementing the system it was observed that the user was initially resisting the change, however the system being the need of the hour and user friendly, the fear was overcome. Entering live data of the past months records was little tedious, prior to the actual day to day transactions.

The best test made on the system was whether it produces the correct outputs. All the outputs were checked out and were found to be correct. Feed-back sessions were conducted and the suggested changes given by the user were made before the acceptance test. Finally the system is being accepted and made to run with live data.

System tests are designed to validate a fully developed system with a view to assuring that it meets its requirements. There are three main kinds of system testing:

- Alpha Testing.
- Beta Testing.
- Acceptance Testing.

Alpha Testing: This refers to the system testing that is carried out by the test team with the organization. Beta Testing: This refers to the system testing that is performed by a select group of friendly customers.

Acceptance Testing: This refers to the system testing that is performed by the customer to determine whether or not to accept the delivery of the system. Documentation is a method of communication. A satisfactory documentation of the system should be objective, factual and complete.

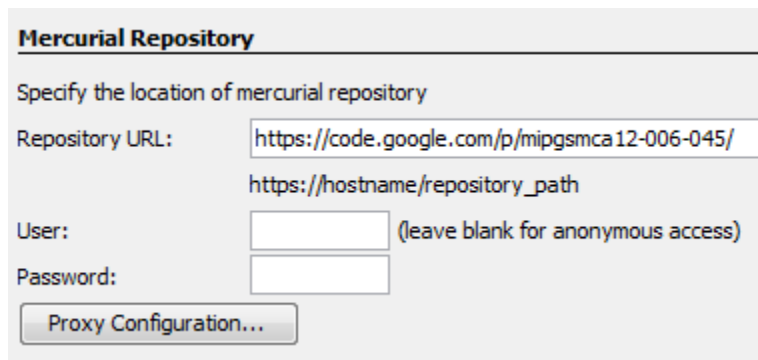
Thus it's adequacy is not determined by format, length, volume or complexity. In documentation, there are no uniform standards that are applicable to all system projects. Documentation is essential to the development, implementation and operation of any system. Documentation is necessary as it helps in maintaining the system and also acts as a reference for the user. Embedding Comments in the executable portion of the code did proper documentation of each module. To enhance the readability of the comments, indentation, parenthesis, blank lines and spaces, proper lineation of the loops were used around the block of comments. Care was also taken to use descriptive names of tables, fields, modules, forms etc. The proper use of indentation, parenthesis, blank lines and spaces were also ensured during coding to enhance the readability of the code.

Installation and running

The project read-only copy is available form

<https://SATHISH.M99@code.google.com/p/mipgsmca12-108-126--online-library/>

1).open netbeans and goto → team→mercurial→clone other.



then click next.

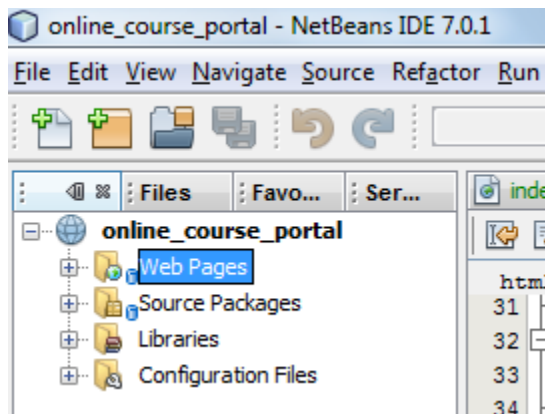
2) follow the instructions click on finish.

after completion of cloning ; click 'open project' button in conformation box.

online course portal project will be opened in netbeans

3).download all jar files from <http://code.google.com/p/mipgsmca12-108-126/downloads/list>

add these jar files to libraries of ONLINE LIBRARY project. Add tomcat server to project.



4) Install Oracle 10G Database software and create user as onlinelibrary and with password onlinelibrary123. Steps for creating user

→Goto command prompt

→type command

sqlplus

→it will ask for userid and password

Enter username as "system"

Enter password as "manager"

→Then it will connect database then type the following commands

create user onlinelibrary identified by onlinelibrary123;

grant all privileges to onlinelibrary;

5) download install Apache tomcat server from the internet.

6) Goto netbeans and run the project.(by default the administrator userid and password is 'admin' and 'admin').. the home page of the project will be opened.

Mercurial Software

Version Control:

Version control is the process of managing multiple versions of a piece of information. In its simplest form, this is something that many people do by hand: every time you modify a file, save it under a new name that contains a number, each one higher than the number of the preceding version.

There are a number of reasons why you or your team might want to use an automated version control tool for a project:

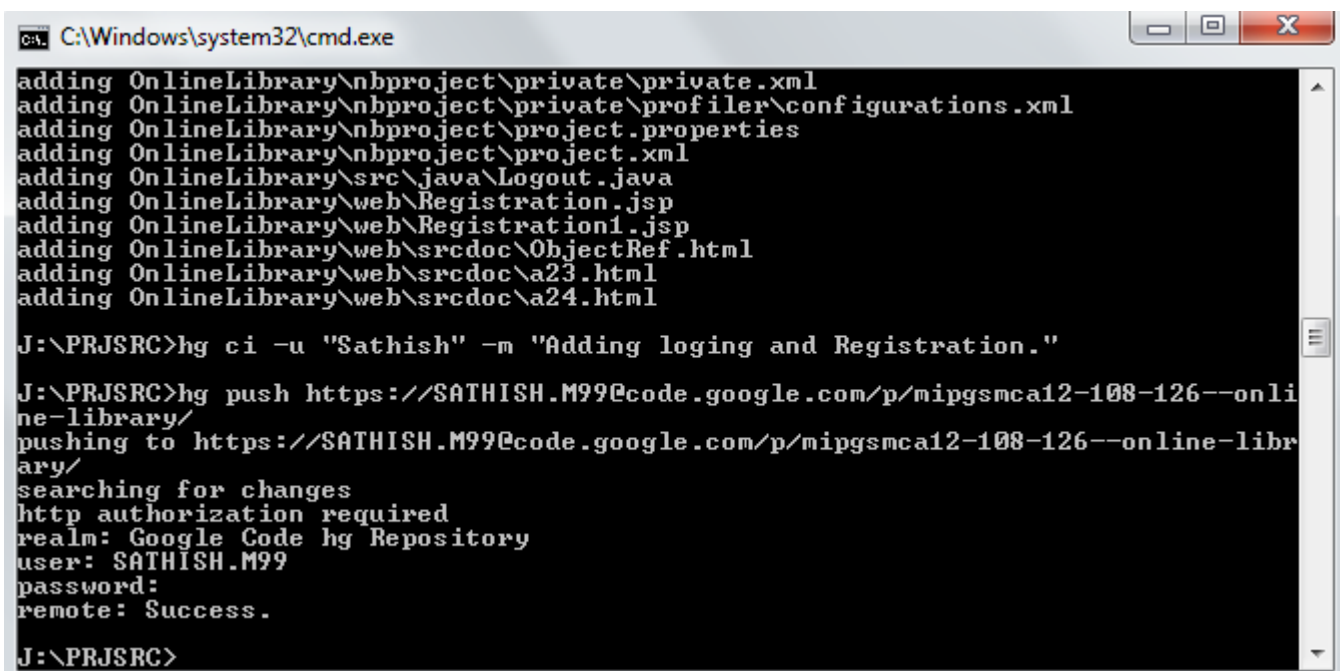
- It will track the history and evolution of your project, so you don't have to. For every change, you'll have a log of *who* made it; *why* they made it; *when* they made it; and *what* the change was.
- When you're working with other people, revision control software makes it easier for you to collaborate. For example, when people more or less simultaneously make potentially incompatible changes, the software will help you to identify and resolve those conflicts.
- It can help you to recover from mistakes. If you make a change that later turns out to be in error, you can revert to an earlier version of one or more files. In fact, a *really* good revision control tool will even help you to efficiently figure out exactly when a problem was introduced.
- It will help you to work simultaneously on, and manage the drift between, multiple versions of your project.

Google project hosting using Mercurial and net beans:

(to do the following steps u need to install mercurial2.x version on your system.)

- 1) To create a project on Google login into <http://code.google.com> with your Google account.
- 2) Click on create new project fill all the required fields (project name, title, summery, License, labels etc,. In addition, select Mercurial as version controller.
Click on create project.
- 3) Now you can see your project on my favourites tab (top right).
- 4) Click on your project. Go to source and find the URL smiting like.
hg clone "https://.....@code.google.com/p/your project name"
- 5).as discussed previous in installation and running section you can get the source code from Google to your PC. You can also use command prompt using hg command initially go to any folder (say D:\workshop) type 'hg init'.

for my project i can use the following command



```
C:\Windows\system32\cmd.exe

adding OnlineLibrary\nbproject\private\private.xml
adding OnlineLibrary\nbproject\private\profiler\configurations.xml
adding OnlineLibrary\nbproject\project.properties
adding OnlineLibrary\nbproject\project.xml
adding OnlineLibrary\src\java\Logout.java
adding OnlineLibrary\web\Registration.jsp
adding OnlineLibrary\web\Registration1.jsp
adding OnlineLibrary\web\srcdoc\ObjectRef.html
adding OnlineLibrary\web\srcdoc\a23.html
adding OnlineLibrary\web\srcdoc\a24.html

J:\PRJSRC>hg ci -u "Sathish" -m "Adding logging and Registration."

J:\PRJSRC>hg push https://SATHISH.M99@code.google.com/p/mipgsmca12-108-126--online-library/
pushing to https://SATHISH.M99@code.google.com/p/mipgsmca12-108-126--online-library/
searching for changes
http authorization required
realm: Google Code hg Repository
user: SATHISH.M99
password:
remote: Success.

J:\PRJSRC>
```

it creates mipgsmca12-108-126 folder in D:\workshop\test. In that you will get the code.

To Upload the code to Google

If you change add or delete your code in the test directory. And if you want to upload this code to your Google code account.

- 1) hg status(shows changes done, if any).
- 2) hg add.
- 3) hg commit -u "username" -m "message about update".
- 4) hg push <your project URL>
then you will be asked to enter your Google code password.
- 5) If password is correctly provided it shows "remote success".
your code is uploaded.

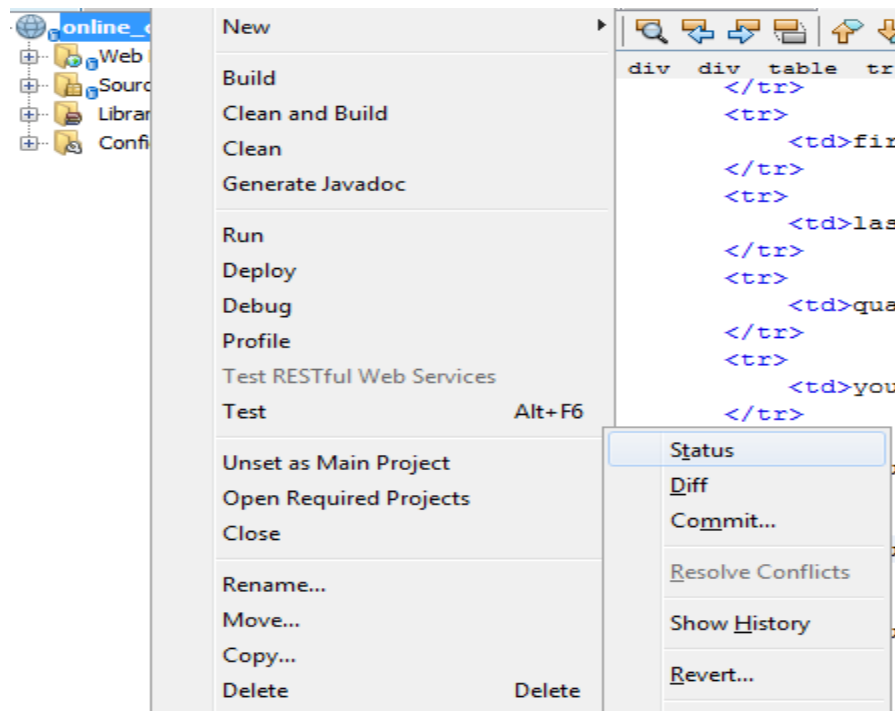
To Upload using netbeans

After cloning the project from Google if make any changes to project and if you want to

Save the code in Google then.

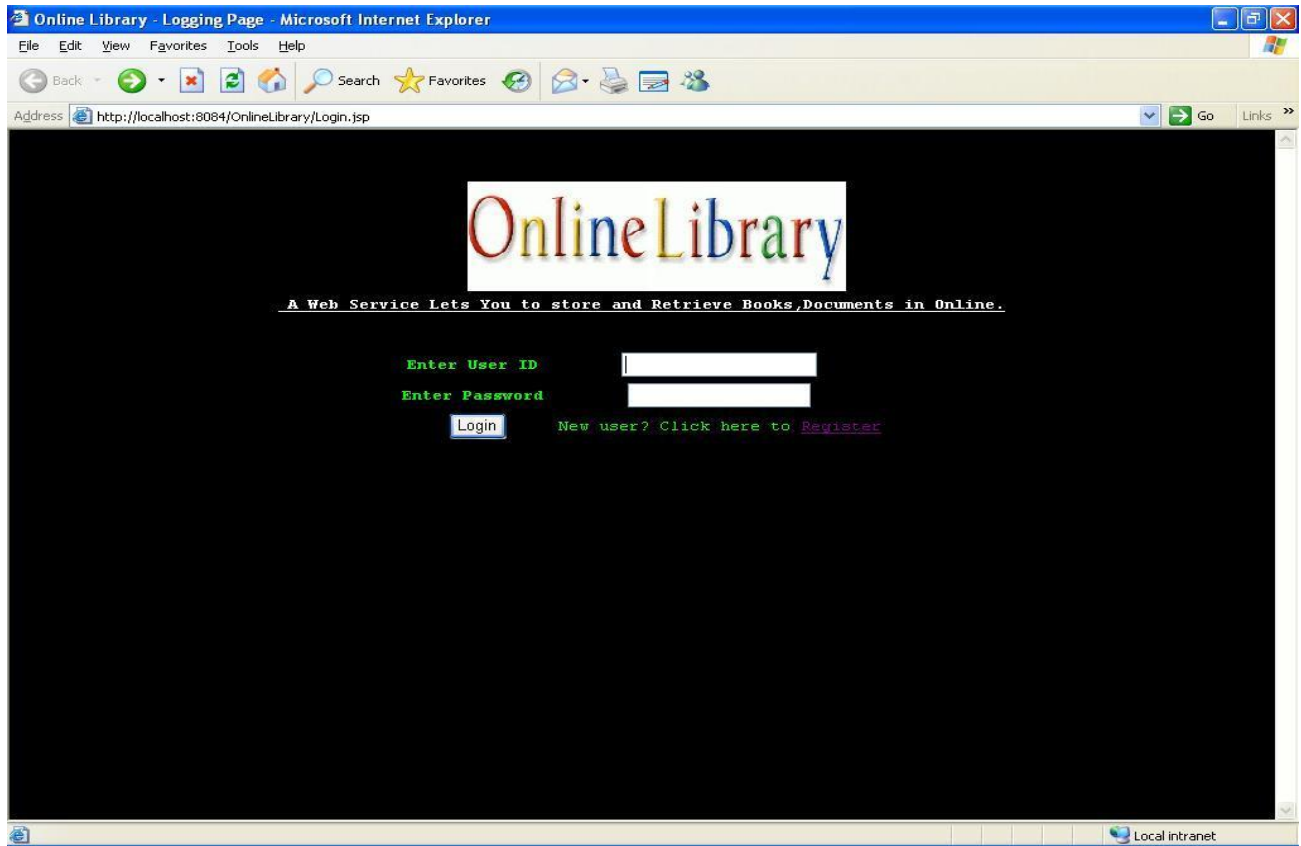
- 1).right click on the project. →Mercurial→Status (shows changes, if any).
- 2).to commit click on commit.

3).to upload click on share→push to default.

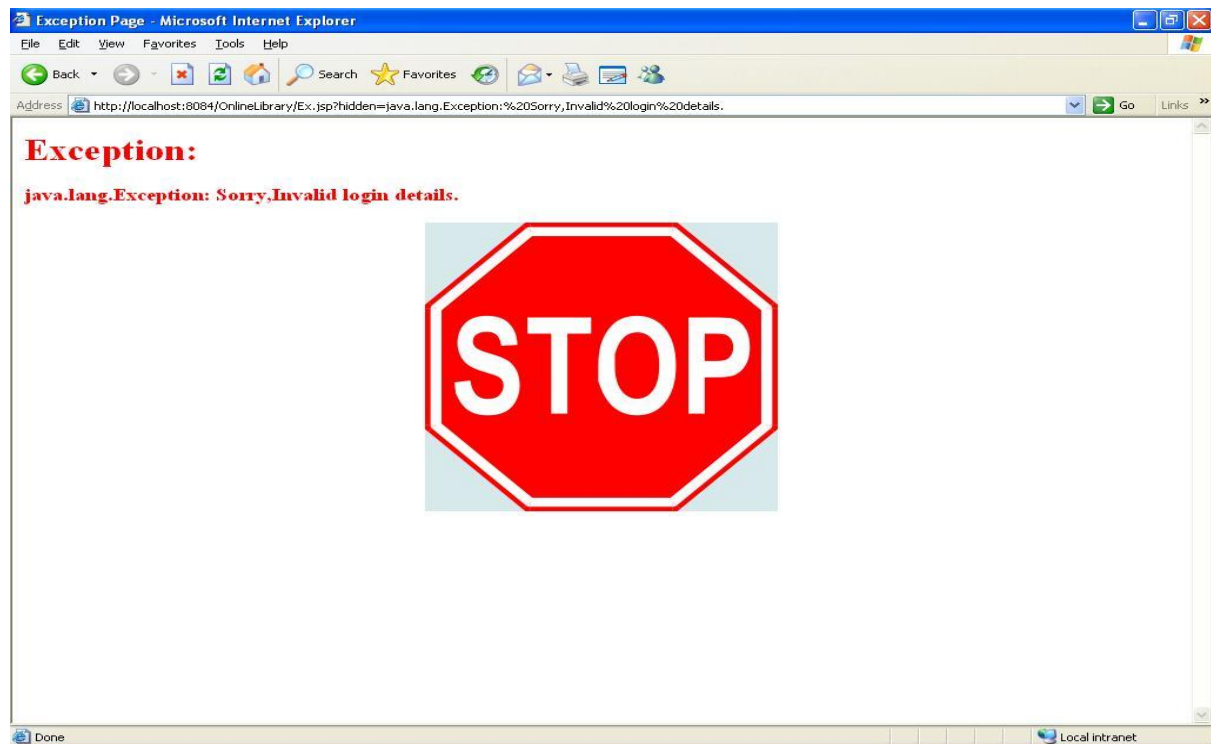


OUT PUT SCREEN SHOTS

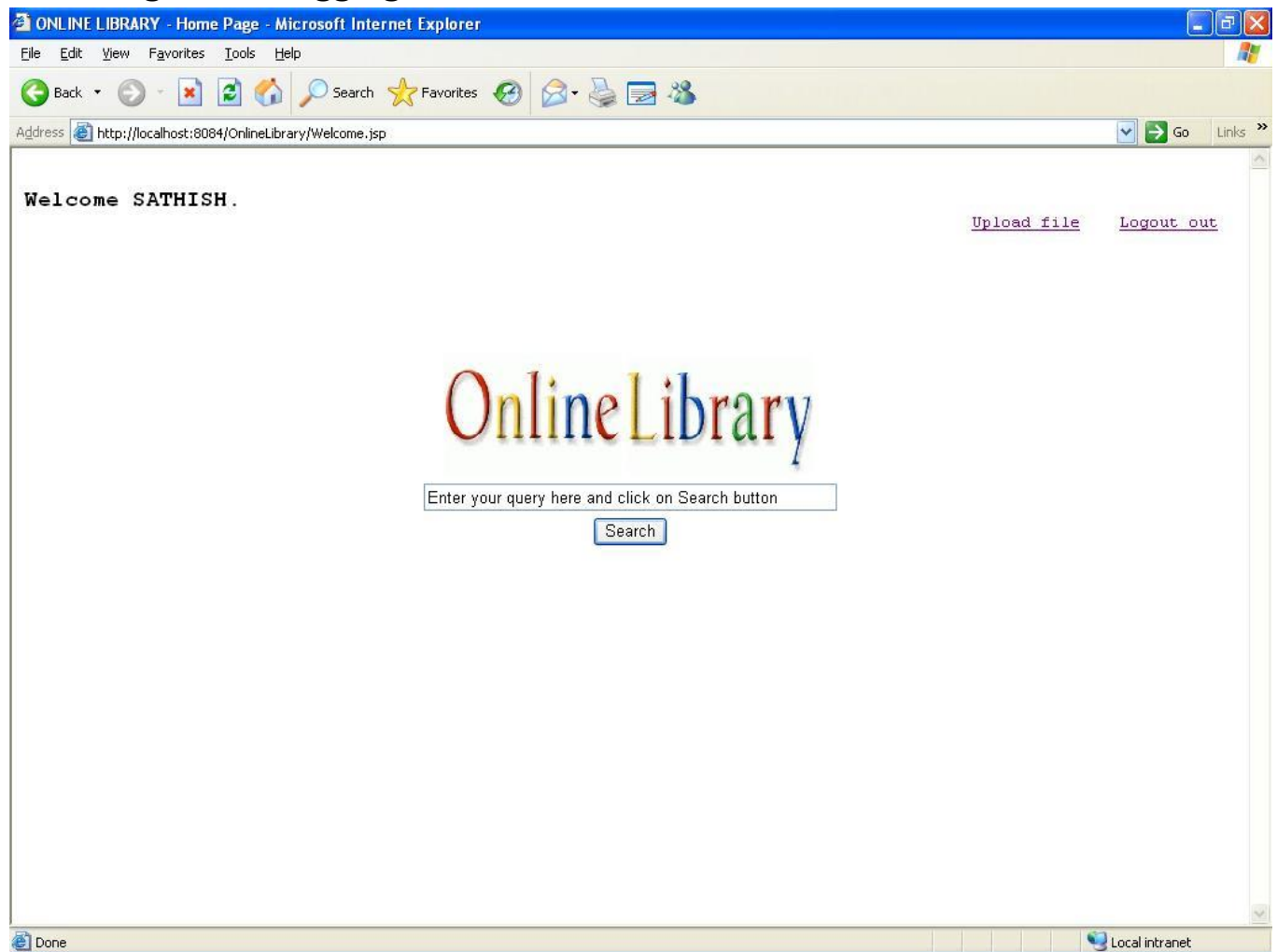
1) Logging Page.

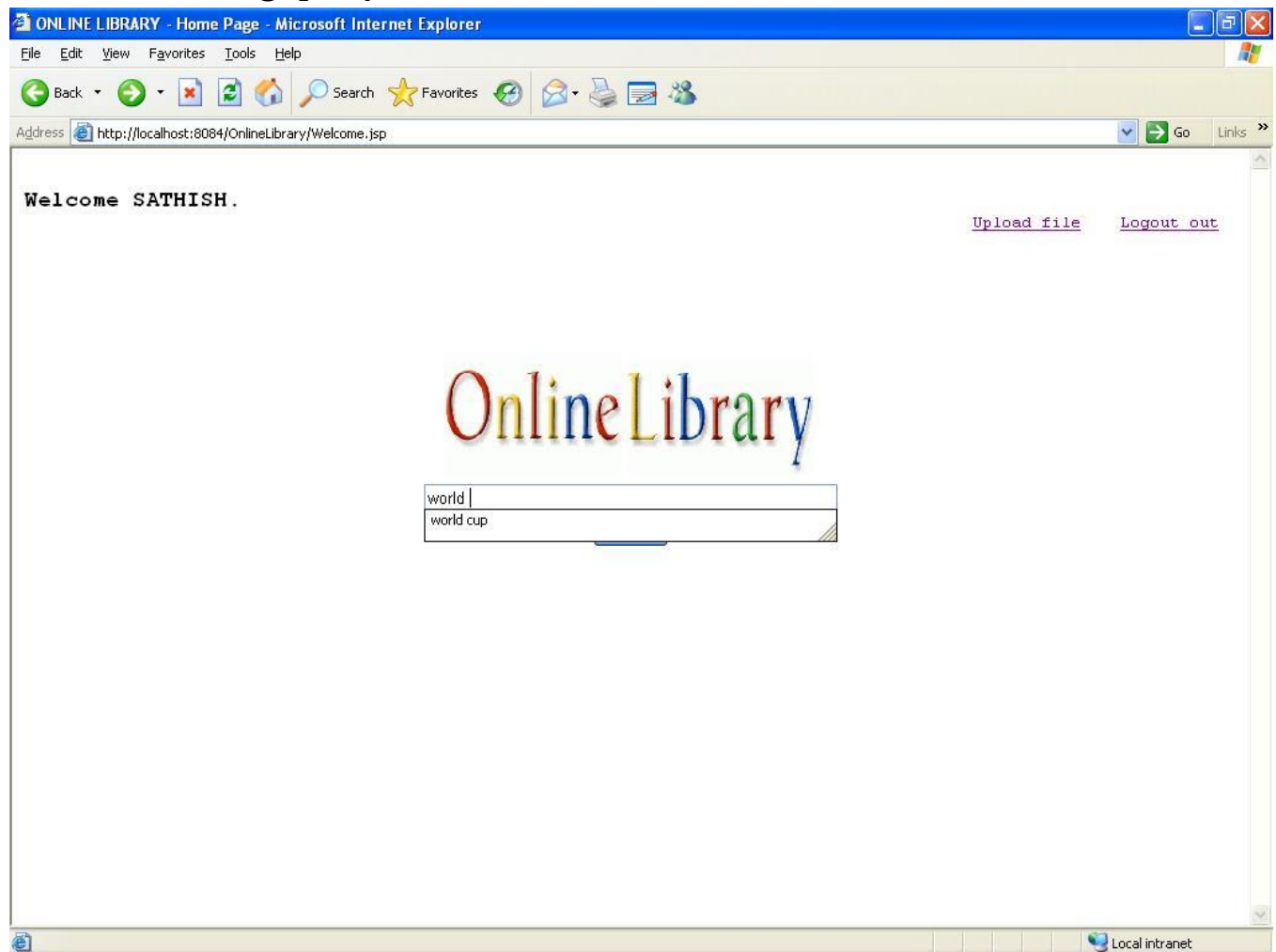


Logging Error

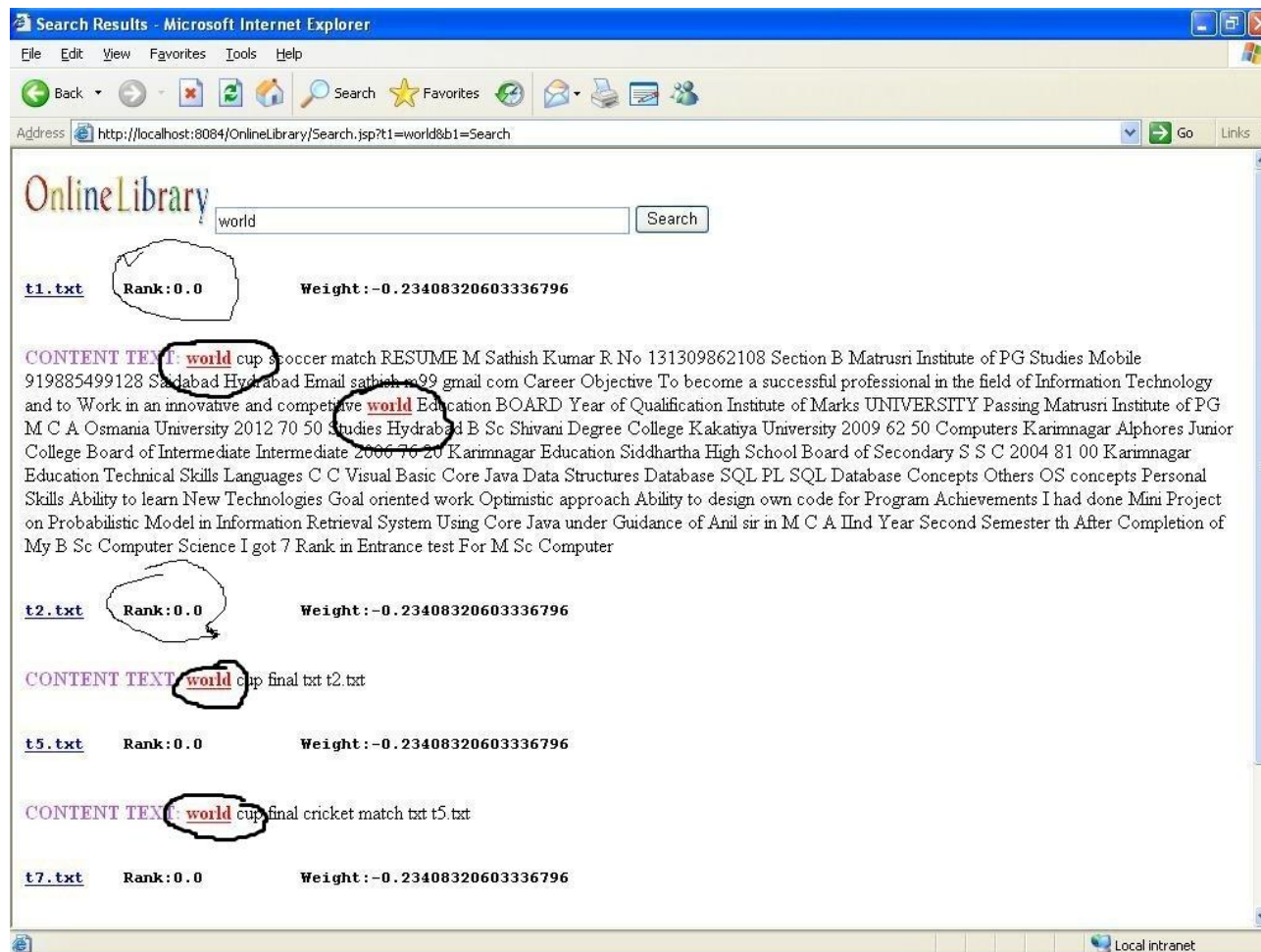


Home Page after Logging.

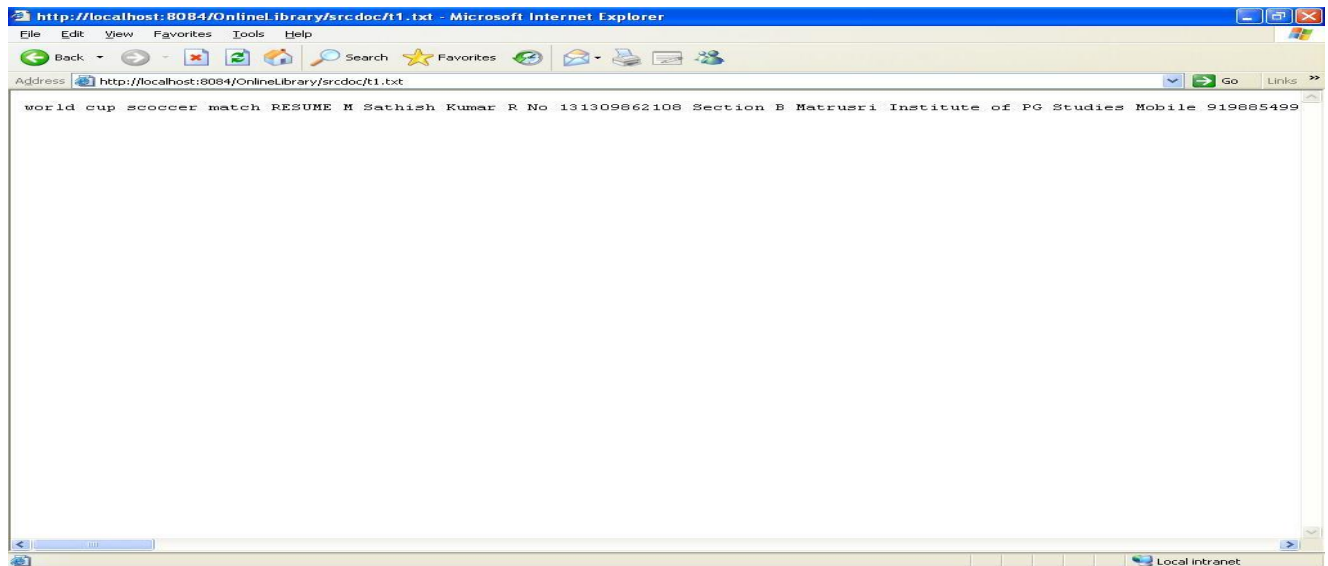


While submitting query:

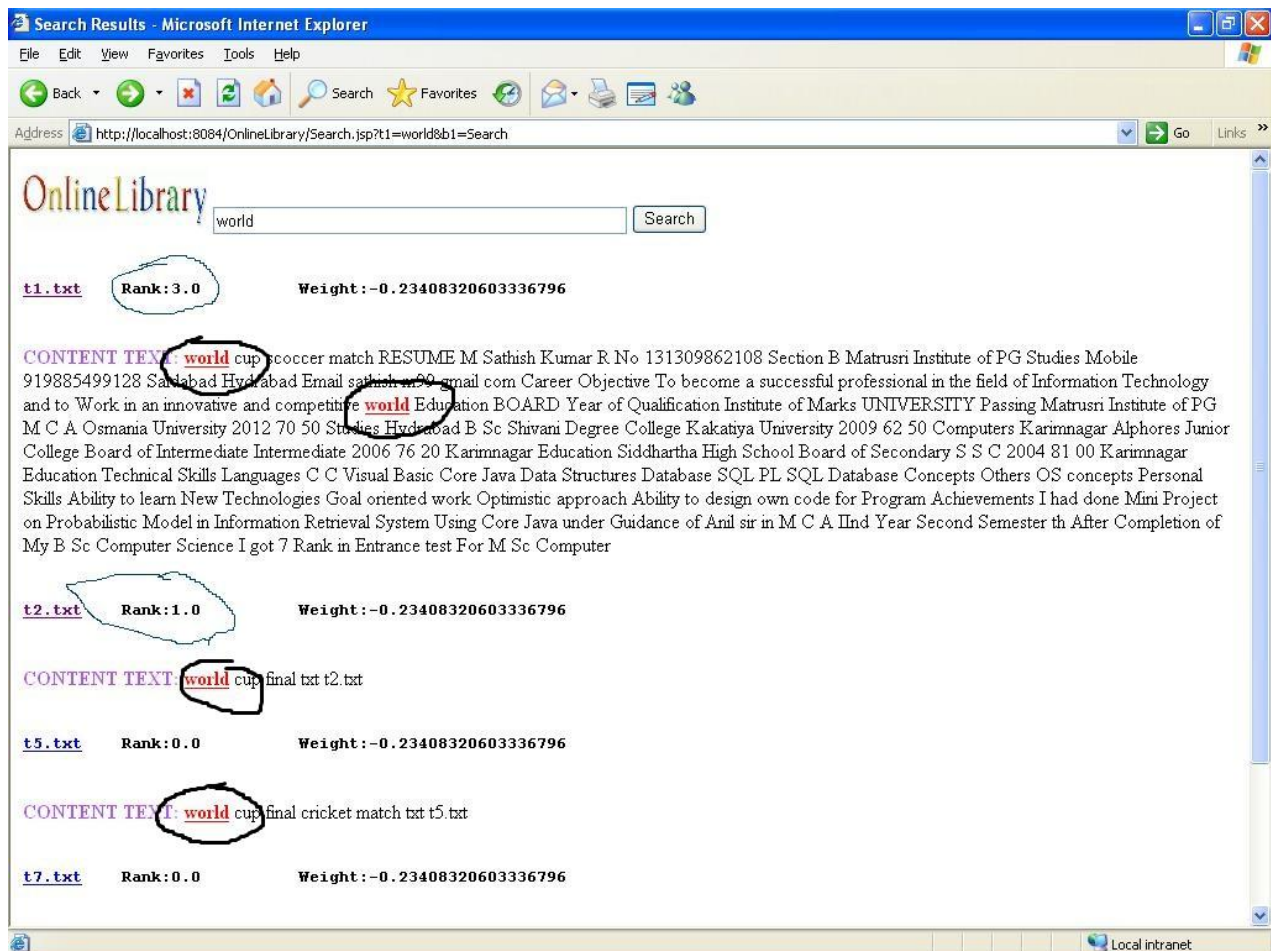
After searching Displaying results with ranks and weights with highlighting key term (world):



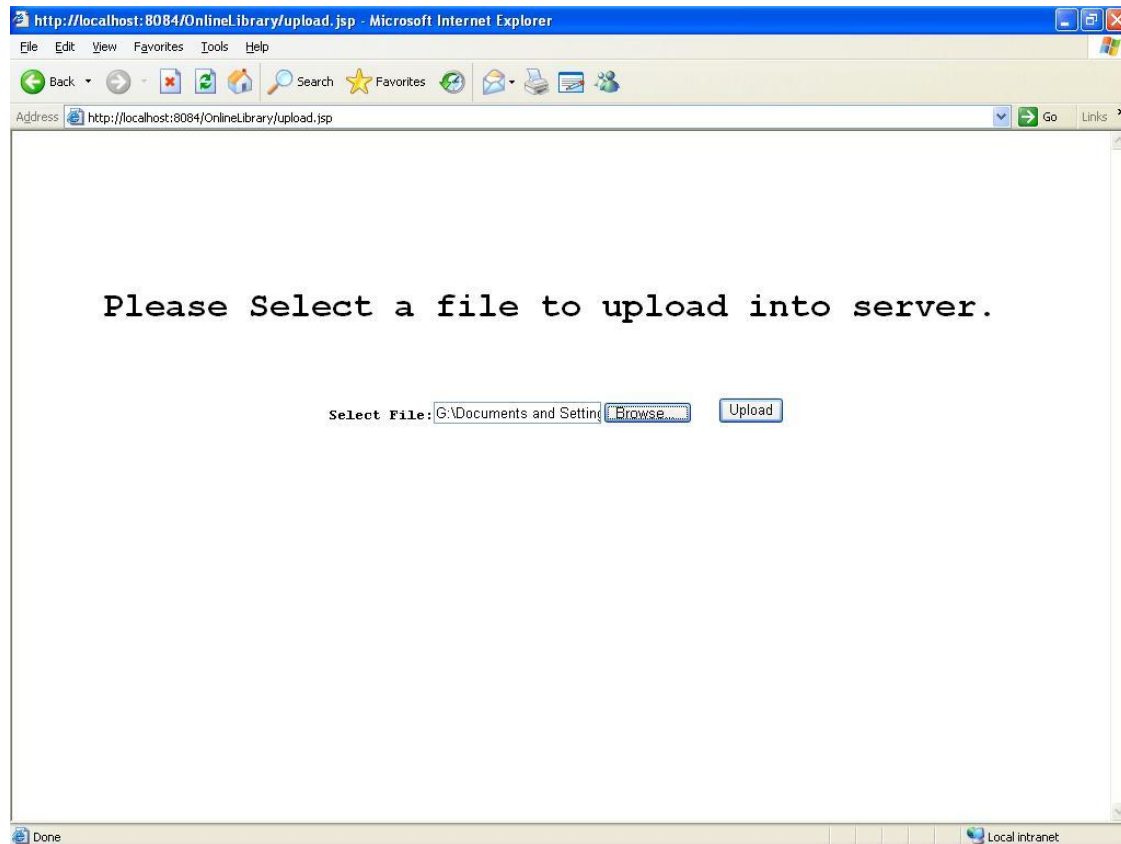
After Opening Document by clicking on Hyper link:



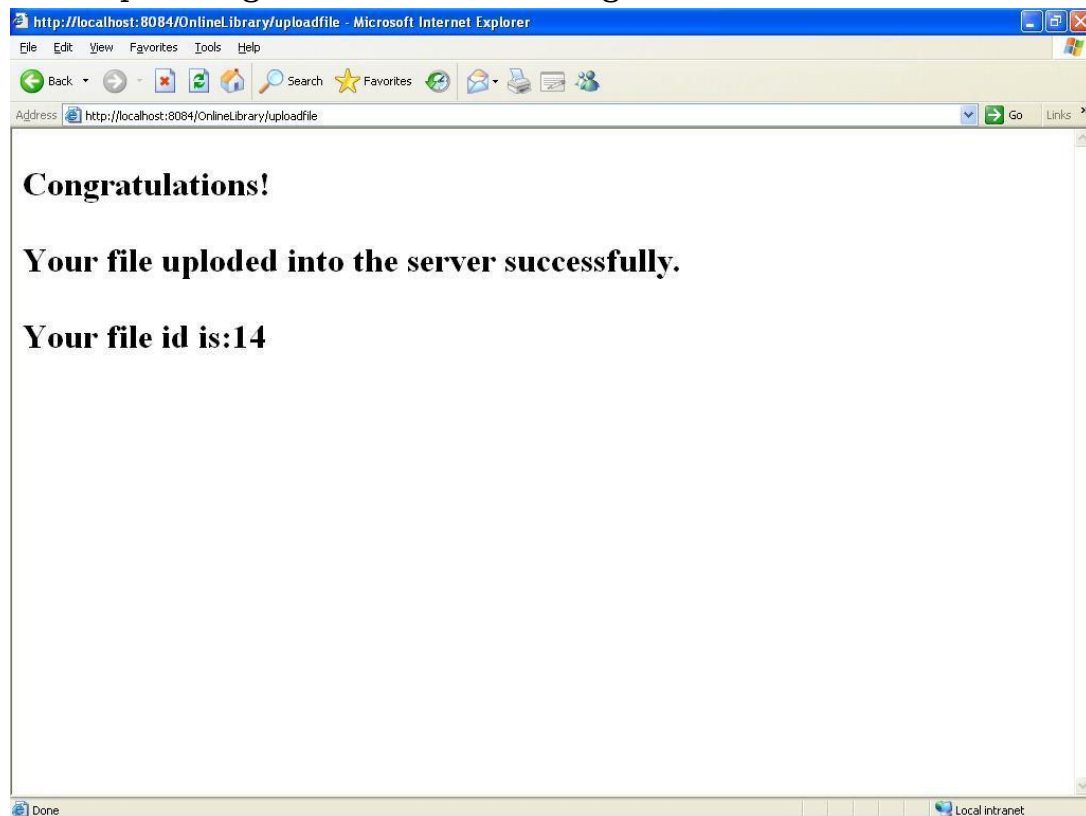
On next Search engine increase rank of document, which opened previously.



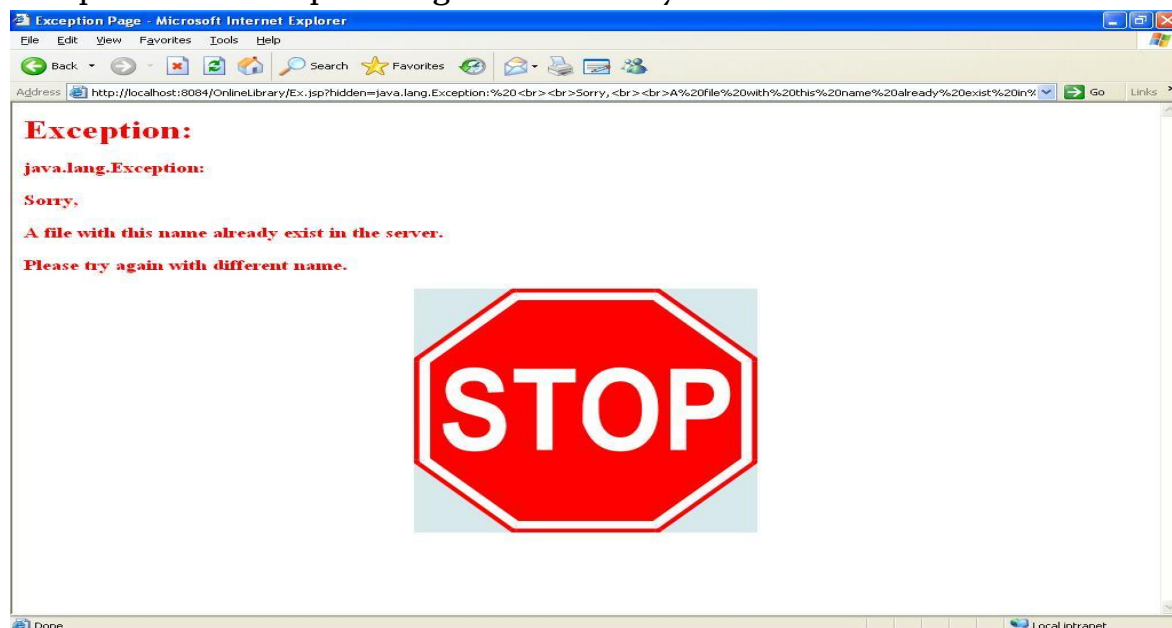
While Uploading file.



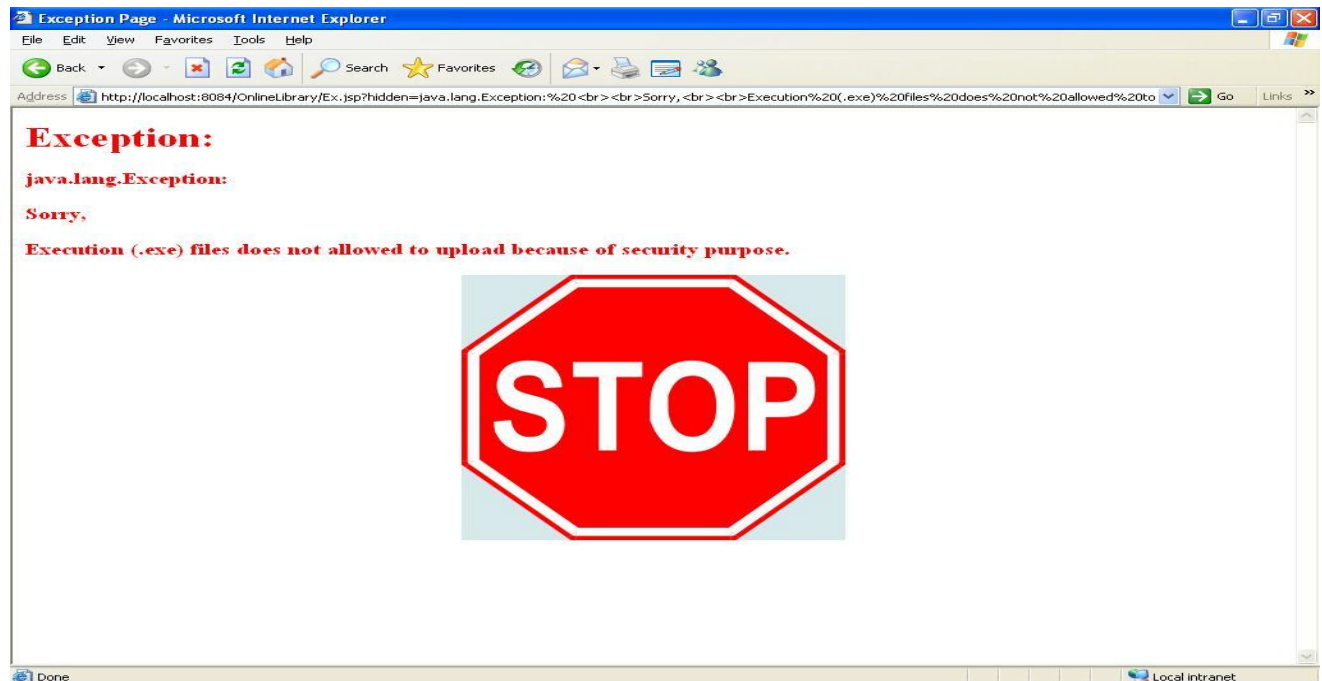
After uploading conformation message.



Exceptions while uploading if file already exist in the server.



If we try upload to upload .exe file!



Registration Page.

A screenshot of a Microsoft Internet Explorer window titled "Registration Page - Microsoft Internet Explorer". The address bar shows "http://localhost:8084/OnlineLibrary/Registration.jsp". The main content area displays the heading "Please Fill The Registration Form". Below the heading is a registration form with the following fields and values: "Enter Your Name" (SATHISH2), "Enter Your User ID" (sa1), "Enter Password" (sq), "Re-Enter Password" (sq), "Enter Your Email Address" (sathish.m99@gmail.com), "Enter Your Mobile Number" (9885499128), and "Select Your Gender" (Male). A "Register" button is located below the form. The status bar at the bottom shows "Done" and "Local intranet".

Registration Page Errors.

The first screenshot shows a web browser window titled "Registration For Online Library - Microsoft Internet Explorer". The address bar shows a URL with a long string of parameters. The page content is titled "Please Fill The Registration Form". The form fields are as follows:

Enter Your Name	SATHISH2
Enter Your User ID	sa1
Enter Password	sq
Re-Enter Password	sq
Enter Your Email Address	sathish.m99@gmail.com
Enter Your Mobile Number	9865499128
Select Your Gender	Male

Below the form, there is a "Register" button and an error message: "Invalid Name.".

The second screenshot shows the same web browser window. The form fields are as follows:

Enter Your Name	SATHISH
Enter Your User ID	1234
Enter Password	sq
Re-Enter Password	sq
Enter Your Email Address	sathish.m99@gmail.com
Enter Your Mobile Number	9865499128
Select Your Gender	Male

Below the form, there is a "Register" button and an error message: "User id already exist.".

Registration For Online Library - Microsoft Internet Explorer

Address: http://localhost:8084/OnlineLibrary/Registration1.jsp?t1=SATHISH&t2=sathish7&t3=sq&t4=sq1&t5=sathish.m99@gmail.com&t6=9885499128&s1=Male&b1=Register

Please Fill The Registration Form

Enter Your Name: SATHISH
Enter Your User ID: sathish7
Enter Password: sq
Re-Enter Password: sq1
Enter Your Email Address: sathish.m99@gmail.com
Enter Your Mobile Number: 9885499128
Select Your Gender: Male

Password do not match.

Registration For Online Library - Microsoft Internet Explorer

Address: http://localhost:8084/OnlineLibrary/Registration1.jsp?t1=SATHISH&t2=sathish7&t3=sq&t4=sq&t5=sathish.m99@mail.com&t6=9885499128&s1=Male&b1=Register

Please Fill The Registration Form

Enter Your Name: SATHISH
Enter Your User ID: sathish7
Enter Password: sq
Re-Enter Password: sq
Enter Your Email Address: sathish.m99@mail.com
Enter Your Mobile Number: 9885499128
Select Your Gender: Male

Invalid Electronic mail

Registration For Online Library - Microsoft Internet Explorer

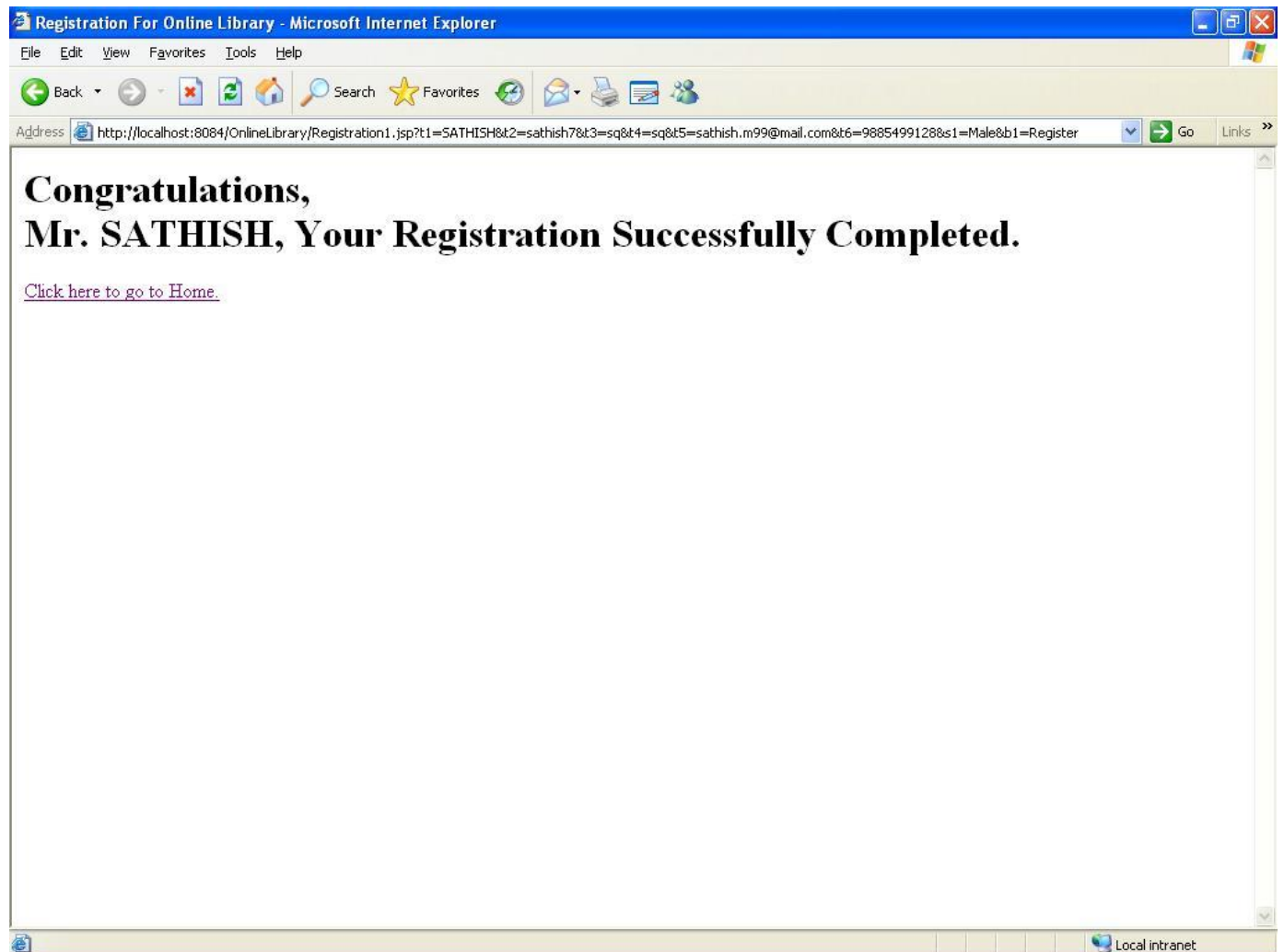
Address: http://localhost:8084/OnlineLibrary/Registration1.jsp?t1=SATHISH&t2=sathish7&t3=sq&t4=sq&t5=sathish.m99@mail.com&t6=9885499128&s1=Male&b1=Register

Please Fill The Registration Form

Enter Your Name: SATHISH
Enter Your User ID: sathish7
Enter Password: sq
Re-Enter Password: sq
Enter Your Email Address: sathish.m99@mail.com
Enter Your Mobile Number: 988549912
Select Your Gender: Male

Invalid mobile no.

Confirmation Message After successful Registration.



CONCLUSION

The application is designed using Java Server Pages, Java Beans, and Oracle 10G database. In order to ensure the quality of software, all software engineering concepts, including test cases had implemented.

This has been developed by considering all the needs of the Examination System and by thorough interaction with the users of the system. The refined version facilitates to capture photographs and provides better communication services through file uploads and news.

The functionality of this system can be extended by providing services like status reports and statistical data maintenance, proper interface designs to OMR and OCR readers, Remuneration bill processing.

The main aim of project is providing highly relevant result in short period of time. Speed of searching of documents and relevance of documents can be increase by improving algorithm. Speed and relevance depends on algorithms.

SCOPE

The system provides a number of services, which enhance the automated process, and provides better service management web enabled applications. In addition, provides the user with an interface that is intuitive and easy to use and employs most user-friendly features.

The scope of project is not having end. Scope is improving searching speed and relevance. We can add advance searching and custom searching to this project.

Bibliography

- 1) Information Retrieval System.
Elective of M.C.A II year II sem(for Searching Algorithms)
- 2) The Programming Language of oracle - Ivan Bayross.
For Database concepts.