WEBSITE OPTIMIZATION USING WCM AND WUM

A PROJECT REPORT

Submitted by

NISHMIJA.B.H	71202205026
RAMYA.M	71202205035

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2006

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"WEBSITE OPTIMIZATION USING WCM AND WUM"** is the bonafide work of "NISHMIJA.B.H. and M.RAMYA" who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr.G.GOPALSAMY	Mr.V.VIJILESH
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	Senior Lecturer
Information Technology Kumaraguru College of Technology Coimbatore-641006.	Information Technology Kumaraguru College of Technology Coimbatore-641006.

The candidates with University Register Nos. **71202205026, 71202205035** were examined by us in the project viva-voce examination held on 26.04.2006

INTERNAL EXAMINER

26/4/06
EXTERNAL EXAMINER

DECLARATION

We,

NISHMIJA.B.H	**71202205026**
RAMYA.M	**71202205035**

Declare that the project entitled **"WEBSITE OPTIMIZATION USING WCM AND WUM"**, submitted in partial fulfillment to Anna University as the project work of Bachelor Of Technology (Information Technology) Degree, is a record of original work done by us under the supervision and the guidance of **Mr.V.VIJILESH, M.E.,** Senior Lecturer, Department of Information Technology, Kumaraguru College of Technology, Coimbatore.

Place: Coimbatore

Date: 21-04-2006

Nishmija.

[Nishmija.B.H]

M. Ramya

[Ramya.M]

Project Guided by

[Mr.V.Vijilesh, M.E.]

Head of the Department

[Dr.G.Gopalsamy, Ph.D.]

ACKNOWLEDGEMENT

We are extremely grateful to **Dr.K.K.Padmanabhan**, B.Sc. (Engg.)., M.Tech., Ph.D., Principal, Kumaraguru College of Technology for having given us a golden opportunity to embark on this project.

We are deeply obliged to **Dr.G.Gopalsamy**, Ph.D., Head of the Department of Information Technology for his valuable guidance and useful suggestions during the course of this project.

We also extend our heartfelt thanks to our project co-ordinator **Mr.K.R.Baskaran**, M.S., Assistant Professor, Department of Information Technology for providing us the support which really helped us make this project a success.

We are indebted to our project guide **Mr.V.Vijilesh**, M.E., Department of Information Technology for his helpful guidance and valuable support given to us throughout this project.

We thank the teaching and non-teaching staff of our department for providing us the technical support during our project.

We also thank our friends and family who helped us to complete this project successfully.

ABSTRACT

This project 'WEBSITE OPTIMIZATION USING WCM AND WUM' optimizes the structure and content of a website according to visitor preferences.

The new trend in the sales market is towards virtualization of business. Almost all companies in local and global market have started using WWW as a new channel of world wide coverage and easy business. The struggle to retain old customers and attract new ones is being the decisive factor for the very existence of any company.

So improving the website of the company to suit user preferences is a very critical issue to survive in today's competitive world. The key issue is that most content of the website is not organized properly. This hinders the quick finding of the product or service the users need.

This results in waste of time to the visitor which leads to loss of valuable customers to the company. So the objective of our project is to understand the behavior of the visitors in a web site and customize the website according to their preferences. Our motivation is to show relevant information to visitors and capture their attention.

The visitors' preferences in a website can be understood from the time they spend in each page, the content of the page and their navigation sequence.An analysis of these parameters allows us to identify the most catchy pages and navigation path traversed by majority of the visitors.

In order to prove the methodology's effectiveness, it was applied in ONLINE PET STORE, a fictional web site, showing the benefits of the described approach.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

J2EE	Java 2 Enterprise Edition
WCM	Web Content Mining
WUM	Web Usage Mining
UML	Unified Modeling Language
TF-IDF	Term Frequency-Inverse Document Frequency
WWW	World Wide Web
HTTP	HyperText Transfer Protocol
API	Application Programming Interface
EIS	Enterprise Information System
JDBC	Java DataBase Connectivity
EJB	Enterprise Java Beans
JSP	Java Server Pages
XML	eXtended Markup Language

1. INTRODUCTION

The main objective of this project is to optimize the structure and content of a website to suit the majority of the buyers' preferences and interests. This attracts more customers to the site which will result in a profit to the organization. We take into account the time spent by the visitor in each page of the website, the navigation sequence and the content of the pages he/she visited. We use web mining techniques to extract these parameters and analyze them.

1.1 THE EXISTING SYSTEM AND ITS LIMITATIONS

The existing web sites try to understand their visitors' preferences by asking them to give direct feedback using feedback forms and questionnaires. But most of the visitors are not usually inclined to fill up such forms to express their interest. So the webmasters have to resort to other indirect techniques to acquire knowledge about visitor behavior.

Today's web visitors expect the web site to give all their required information in just a few page clicks. To provide such sophistication, the web masters should have a complete picture of the users varied interests and organize the website accordingly.

The existing analyzer tools just take into account the usage information of the site which is available from the web server log files. They do not pay attention to the contents of the page in which the user is interested.

1.2THE PROPOSED SYSTEM AND ITS ADVANTAGES

Our system uses web mining techniques such as content mining and usage mining to understand the visitors' interest. The usage mining depends on web log files which give information about the pages visited by the user, the time spent in the pages and other useful information. The content mining depends on the content of the pages i.e. the information available in the pages that the users spent their most time. We combine these two differential mining techniques in a unified manner to identify the pages, the content and the sequence appealing to most visitors. The web master can make use of this information to customize the website in an interesting manner. Our system effectively contributes to the market basket analysis which is very important for today's commercial shopping websites.

Our system doesn't depend on the direct feedback from the users. We directly extract information from the log files and the web site itself using mining techniques [2]. So the users are not bothered with cumbersome feedback forms or questionnaires to express their interests.

We use mining techniques to extract sequence clusters from user sessions to identify the sequence of pages visited by users. A clustering algorithm is applied to this data to identify the most navigated sequence [3]. From this we are able to gauge the visitors' varied interests.

Unlike the existing log analyzer tools which provide all statistical information just based on the web log files, we take into account the content of the pages visited by the user. This helps in a more detailed and interesting customization than the former.

2. SYSTEM REQUIREMENTS ANALYSIS

The purpose of system requirements analysis is to obtain a detailed and thorough understanding of the business need. Then it is broken down into discrete requirements, which are then clearly defined, reviewed and agreed upon with the customers and decision makers. During system requirements analysis, the framework for the application is developed, providing the foundation for all future design and development efforts.

2.1 PRODUCT DEFINITION

Our intention is to develop a product that will help web masters to cater their website to satisfy most of their customer needs. The visitor behavior in a website is understood using various parameters and the website is optimized accordingly [1].Our system helps to a great extent to webmasters who design and run shopping cart websites.

2.2 PROJECT PLAN

In the analysis phase, we learnt about various kinds of application servers and log formats. From this study, the most suitable language for implementing this project was found to be Java, making use of Java's various features like multi platform support, swing support and the like. We selected the Sun Java Application Server to host the sample website ONLINE PET STORE to provide easy log customization and several advanced features and easy GUI support.

We also learnt about various clustering algorithms to cluster unsupervised data sets like visitor behavior in a website. We found k-means clustering algorithm to be an efficient yet simple algorithm to plot the clusters as well as to identify the most populated cluster.

From the analysis phase, the design phase commences in which the various modules functionalities are identified. The complete system's flow of control and data are identified and depicted in the form of diagrams.

Next the implementation phase is taken care of in which the coding of the project is done. Each module is coded separately and finally integrated to form the entire application.

Various users were allowed to access the website on a trial basis and their visits were recorded in the log files. These log files were collected for a period of one month and integrated together to be used as an input to the application.

The web mining techniques and the log file preprocessing were all implemented in Java and the k-means clustering algorithm was implemented in MATLAB. The software was chosen because of its ease in handling graphics applications and efficient command line interface.

In the testing phase, each module is tested thoroughly and finally integrated modules are tested together to ensure the correct working of the entire application. Testing is also done to ensure this application satisfied the specified requirements and set criteria. It is checked whether the product will meet most of the webmasters requirements in easy log analysis.

The scheduled plan for completion of the project is shown in Table 2.1.

WORK	DURATION
• Feasibility Analysis • Abstract Preparation • Requirements Gathering • Selecting sample website	One week
• Collecting required s/w and servers • Installing the server and setting log formats	Two weeks
• Coding –preprocessing of the content and log files	One week
• Coding-finding similarity between various user sessions	One week
• Studying various clustering algorithms • Analyzing the merits of k-means clustering algorithm	Two weeks
• Implementing k means algorithm to find clusters in user sessions	One week
• Coding the time hit program and the content similarity program	One week
• Implementation of the system	Two weeks
• Testing of the system	One week
• Optimization of the website based on the results derived	One week

Table 2.1 PROJECT PLAN

2.3 SOFTWARE REQUIREMENTS SPECIFICATION

2.3.1 PURPOSE

The purpose of this document is to describe the optimization of the structure and content of a website to fulfill the needs of most of the customers. This application is intended to serve the webmasters of e-shopping websites.

2.3.2 SCOPE

This document is the only one that describes the requirements of the system. It is meant for the use by the developers and will be the basis for validating the final delivered system. Any changes made to the requirements in the future will have to go through a formal change approval process. The developer is responsible for asking for clarifications, where necessary, and will not make any alterations without the permission of the client.

2.3.3 DEFINITIONS

Customer:

A person or organization, internal or external to the producing organization, who takes financial responsibility for the system.

In a large system, this may not be the end user. The customer is the ultimate recipient of the developed product and its artifacts.

User:

A person who will use the system that is developed.

Analyst:

The analyst details the specification of the systems functionality by describing the requirements aspect under the supporting software requirements.

2.3.4 GENERAL DESCRIPTION

2.3.4.1 PRODUCT OVERVIEW

This project is to optimize the structure and content of a website according to the visitors' interests and their preferences. This makes the website to attract more customers to the site which helps organization to gain profit. We use web mining techniques to understand their interest.

2.3.4.2 USER CHARACTERISTICS

The main users of this system are webmasters of shopping websites. The webmasters are required to have the moderate technical expertise in handling the system.

2.3.4.3 GENERAL CONSTRAINTS

The application has been programmed to run in the windows platform but can be migrated to other platforms if the need arises. The server however is a multi-platform application.

2.3.5 SPECIFIC REQUIREMENTS

2.3.5.1 INPUTS AND OUTPUTS

The inputs to the system are the web site files itself which encompass the content of the site. The input also includes the log files which are stored in the application server.

The output of the system is the top pages in which the users were most interested in, the top navigation sequences through which most users traversed by and the content in which most users were interested in.

2.3.5.2 FUNCTIONAL REQUIREMENTS

1. The log file should be in the required format to be accessible to the application
2. The server should be able to record all the pages visited and the timestamps associated with the entries accurately.
3. The application should be run on the system where the web application server is running.

2.3.5.3 HARDWARE REQUIREMENTS

Processor	PENTIUM IV
RAM	128 MB
Hard Disk Capacity	20 GB

2.3.5.4 SOFTWARE REQUIREMENTS

Operating System	WINDOWS 2000/XP/98
Language	Java
Package	JDK 1.5, JWSDP 2.0
Server	Sun Application Server 8
Tool	MATLAB 6.x

2.3.5.5 PERFORMANCE CONSTRAINTS

The system should be able to process a huge amount of logs within a reasonable amount of time and produce the results in an easily understandable graphical format. It should also point out the correct pages on which optimization can be done.

2.3.5.6 SOFTWARE CONSTRAINTS

The application runs on Windows platform. The server requires Java Web services developer pack and jdk1.5 to be installed and running in the server system. The system should not be hosting any other web server application.

3. SYSTEM STUDY

3.1 AN INTRODUCTION TO JAVA

Java is an object oriented programming language with a built in application programming interface that can handle graphics in user interfaces. It can be used to create applications or applets. Java is a general purpose programming language with a number of features that make the language well suited for use on the World Wide Web [7]. Small java applications called Java applets can be downloaded from a web server and can be run on your computer by a Java compatible web browser. These features have made Java the first application language of the World Wide Web.

Properties of Java

- Compiled and interpreted
- Platform independent and portable
- Robust and secure
- Multithreading and interactive
- Dynamic and extensible

Its rapid ascension and wide acceptance can be traced to its design and programming features, particularly in its promise that we can write a program once and run it anywhere. Because of these promising attributes, Java can be considered as the perfect language for programming web related applications such as mining techniques.

3.2 MATLAB – A CURTAIN RAISER

MATLAB is a high-level language and interactive environment that enables you to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and FORTRAN [5]. MATLAB can be used in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational biology. Add-on toolboxes (collections of special-purpose MATLAB functions, available separately) extend the MATLAB environment to solve particular classes of problems.

MATLAB provides a number of features for documenting and sharing our work. MATLAB code can be integrated with other languages and applications and be distributed.

Key Features

- High-level language for technical computing
- Development environment for managing code, files, and data
- Interactive tools for iterative exploration, design, and problem solving
- Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration
- 2-D and 3-D graphics functions for visualizing data
- Tools for building custom graphical user interfaces
- Functions for integrating MATLAB based algorithms with external applications and languages, such as C, C++, Fortran, Java, COM, and Microsoft Excel

3.3 JAVA SERVER PAGES

Java Server Pages (JSP) technology provides an easy way to create dynamic web pages and simplify the task of building web applications that work with a wide variety of web servers, application servers, browsers and development tools.

Java Server Pages technology allows web developers and designers to easily develop and maintain dynamic web pages that leverage existing business systems [8]. As part of the Java technology family, JSP enables rapid development of web-based applications that are platform-independent. JSP separates user interfaces from content generation, enabling designers to change the overall page layout without altering the underlying dynamic content.

In its basic form, a JSP page is simply an HTML web page that contains additional bits of code that execute application logic to generate dynamic content. This application logic may involve JavaBeans, JDBC objects, Enterprise Java Beans (EJB), and Remote Method Invocation (RMI) objects, all of which can be easily accessed from a JSP page.

For example, a JSP page may contain HTML code that displays static text and graphics, as well as a method call to a JDBC object that accesses a database; when the page is displayed in a user's browser, it will contain both the static HTML content and dynamic information retrieved from the database.

At first glance, a JSP page looks similar to an HTML (or XML) page--both contain text encapsulated by tags, which are defined between <angle brackets>. While HTML tags are processed by a user's web browser to display the page, JSP tags are used by the web server to generate dynamic content. These JSP tags can define individual operations, such as making a method call to a JavaBean, or can include blocks of standard Java code (known as *scriptlets*) that are executed when the page is accessed.

Advantages of JSP

Even if you're already content writing servlets for web applications, there are plenty advantages to using JSP:

- JSP pages easily combine static templates, including HTML or XML fragments, with code that generates dynamic content.
- JSP pages are compiled dynamically into servlets when requested, so page authors can easily make updates to presentation code. JSP pages can also be precompiled if desired.
- JSP tags for invoking JavaBeans components manage these components completely, shielding the page author from the complexity of application logic.
- Developers can offer customized JSP tag libraries that page author's access using an XML-like syntax.
- Web authors can change and edit the fixed template portions of pages without affecting the application logic. Similarly, developers can make logic changes at the component level without editing the individual pages that use the logic.

3.4 WEB SERVER ARCHITECTURE

An application server is a component-based product that resides in the middle-tier of a server centric architecture. It provides middleware services for security and state maintenance, along with data access and persistence.

In many usages, the application server combines or works with a Web (Hypertext Transfer Protocol) server and is called a *Web application server*. The Web browser supports an easy-to-create HTML-based front-end for the user.

The Web server provides several different ways to forward a request to an application server and to forward back a modified or new Web page to the user. These approaches include the Common Gateway Interface (CGI), FastCGI, Microsoft's Active Server Page, and the Java Server Page. In some cases, the Web application servers also support request "brokering" interfaces such as CORBA Internet Inter-ORB Protocol (IIOP).

Java application servers are based on the Java™ 2 Platform, Enterprise Edition (J2EE™). J2EE uses a multi-tier distributed model [4]. This model generally includes a Client Tier, a Middle Tier, and an EIS Tier.

The Client Tier can be one or more applications or browsers. The J2EE Platform is in the Middle Tier and consists of a Web Server and an EJB ServerThe Enterprise Information System (EIS) tier has the existing applications, files, and databases.For the storage of business data, the J2EE platform requires a database that is accessible through the JDBC, SQLJ, or JDO API. The database may be accessible from web components, enterprise beans, and application client components. The three tier architecture of a web server is shown in the figure 3.1.

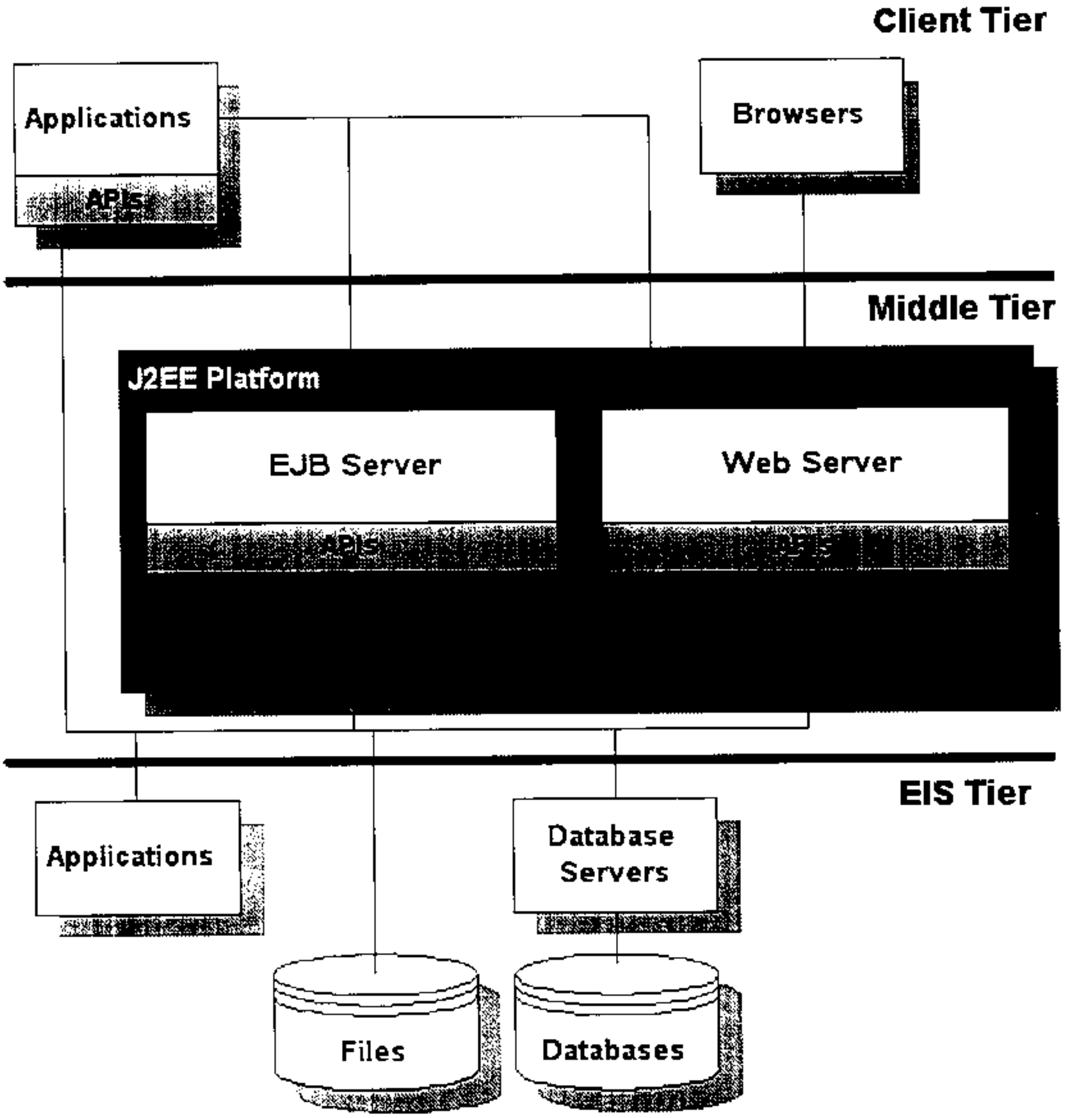


Fig 3.1 THREE TIER ARCHITECTURE

3.5 WEB MINING

Web mining can be broadly defined as the automated discovery and analysis of useful information from the web documents and services using data mining techniques. It is a huge, interdisciplinary and very dynamic scientific area, converging from several research communities such as database, information retrieval, and artificial intelligence especially from machine learning and natural language processing. This area is so broad today partly due to the interests of various research communities, the tremendous growth of information sources available on the Web and the recent interest in e-commerce [1].Web is a collection of interrelated files stored on one more web servers.

Web data is composed of

- Web content – Text, images and records.
- Web structure – Hyperlinks and tags.
- Web usage – http logs, Application server logs, referrer logs and agent logs.

Web mining is used to understand customer behavior, evaluate the effectiveness of a particular Web site, and help quantify the success of a marketing campaign [2]. The information gathered through Web mining is evaluated (sometimes with the aid of software graphing applications) by using traditional data mining parameters such as clustering and classification, association, and examination of sequential patterns.

3.5.1 WEB USAGE MINING

It is the discovery of meaningful patterns from data generated by client server transactions on one or more web localities. The sources for web usage mining are automatically stored data in server access logs, referrer logs, agent logs, client side cookies and user profiles. Using such sources, for example, it is possible to determine the number of accesses to the server and the individual files within the organization's Web space, the times or time intervals of visits, and domain names and the URLs of users of the Web server [2].The server logs are preprocessed to identify valid user sessions from which valuable information like page hits and time spent.

An example of a web server log can be given as,

```
212.209.212.66 - [29/Jul/2001:00:35:33 -0500] "GET /Data-mining.htm HTTP/1.1" 200 11631 "http://internetmarketingengine.com/" "Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0)"
```

212.209.212.66 - IP Address (or XX if the IP address has been resolved)

29/Jul/2001:00:35:33 - Date and Time of the entry

-0500 - Time difference to Greenwich Mean time (Universal Time). This log file entry was created when the web server was on US Central Summer time

GET - Action

Data-mining.htm HTTP/1.1 - Object - i.e. retrieve the page data-mining.htm

200 - result (Result 200 means the task has been completed)

11631 - size of object, in bytes

http://internetmarketingengine.com/ - Referring URL (i.e. this particular page was accessed from the home page of the Internet Marketing Engine)

Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0) - Browser / version and platform - i.e. this person was using Microsoft Internet Explorer 5.5 and the Windows 2000 operating system.

The preprocessing scenario of web usage mining can be represented pictorially as shown below.

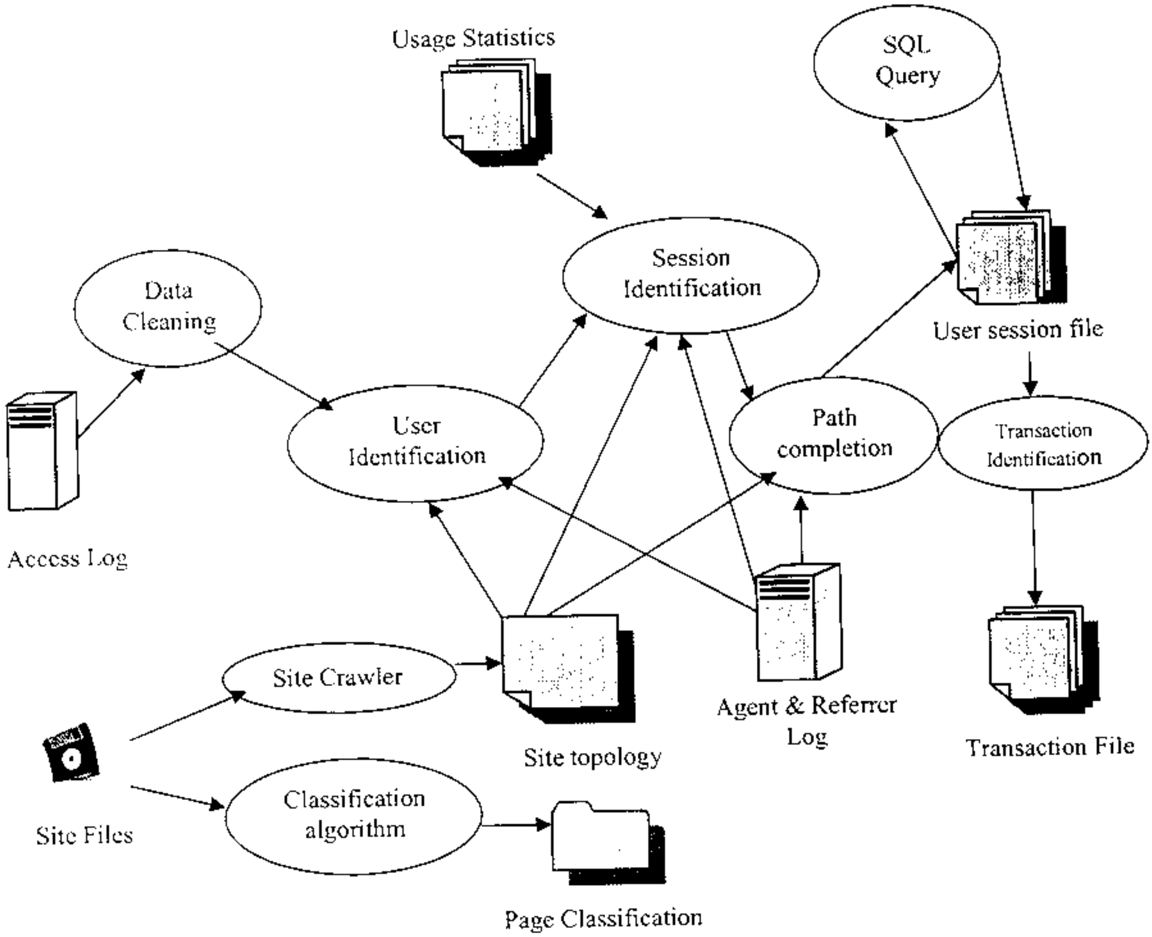


Fig. 3.2 Web Usage Mining Preprocessing Steps

3.5.2 WEB CONTENT MINING

Web Content Mining uses the ideas and principles of data mining and knowledge discovery to screen more specific data. The use of the Web as a provider of information is unfortunately more complex than working with static databases. Due to its very dynamic nature and its vast number of documents, there is a need for new solutions that are not depending on accessing the complete data on the outset. Another important aspect is the presentation of query results. Due to its enormous size, a web query can retrieve thousands of resulting webpages. Thus meaningful methods for presenting these large results are necessary to help a user to select the most interesting content.

It is the process of extracting useful information from the contents of the web documents. Content data corresponds to the collection of facts a web page was designed to convey to the users. It may consist of text, images, audio, video or structured records such as lists and tables. The content of the web should be pre processed before applying any mining techniques [1]. The preprocessing steps include

- Extract text from HTML
- Perform stemming
- Remove stop words
- Calculate collection wide word frequencies
- Calculate per document frequencies

An example of preprocessing the content of a HTML web page is shown below. Here the HTML tags of the page are first extracted. Then the tags and stop words are removed from the page and the weights of each word are calculated using TF-IDF weighting [1]. TF-IDF is the ratio of the times a word occurs in an entire page to the number of documents in which the word appears in the whole website.

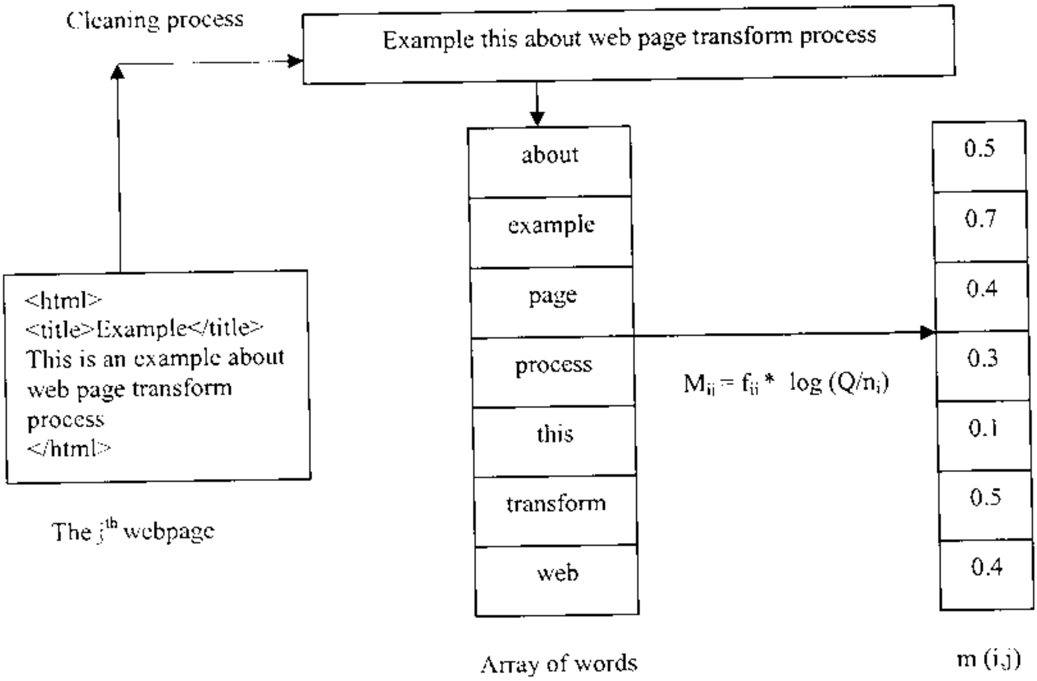


Fig 3.3 Web content processing

3.6 k-means clustering algorithm

k-means clustering is the algorithm chosen for clustering the navigation sequence data of various user sessions [3]. After clustering the data, we proceed to find the most populated cluster. This cluster gives the navigation sequence through which most of the visitors have traveled.

k-means clustering is an algorithm to classify or to group your objects based on attributes/features into k number of groups. k is a positive integer number. Thus the purpose of k-mean clustering is to classify the data.

It is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done.

At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The k-means algorithm can be run multiple times to reduce this effect.The algorithm is composed of the following steps:

1. Place k points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the k centroids.
4. Repeat steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Table 3.2 k-means clustering algorithm

3.7 INTRODUCTION TO WEBSITE OPTIMIZATION

In order to discover hidden knowledge about the visitor behavior in the web site, we used WCM and WUM techniques with clustering algorithms. It aims to combine the philosophies behind WCM and WUM through providing a complemented vision of both techniques [1]. Using WUM is possible to understand the visitor browsing behavior, but is not directly which content is interesting for the visitor. It is possible to review using WCM, specify specifically sub area related to the semantics of the Web.

Our approach is the definition of a measurement that allows to compare behaviors between two visitor sessions, through the analysis of visitor preferences, i.e., at first time what pages visited and soon, what contains were more interesting for him.

The final idea is to use the measurement in a cluster algorithm in order to find group of visitor sessions with closed behavior and using this information, make prediction about the preferences of the future web site's visitors.

4. DESIGN DOCUMENTS

4.1 INTRODUCTION

A software design is representation of the system of the real world in a format that can be easily understood by both the developers and the users. The diagrams drawn in software design help easy communication between the developers of the various modules of the system and with the users of the system.

4.2 PROCESS DESIGN

The design of a process in the object oriented paradigm can be represented using various UML diagrams like class diagram, sequence diagram, activity diagram and collaboration diagram.

Class diagrams are the backbone of almost every object oriented method, including UML. They describe the static structure of a system. Classes represent an abstraction of entities with common characteristics. Associations represent the relationships between classes. The various kinds of associations that can exist between classes are generalization, specialization, composition, aggregation, etc.

A class diagram is a pictorial representation of the detailed system design. Design experts who understand the rules of modeling and designing systems design the system's class diagrams. The structure of a system is represented using class diagrams. Class diagrams are referenced time and again by the developers while implementing the system.

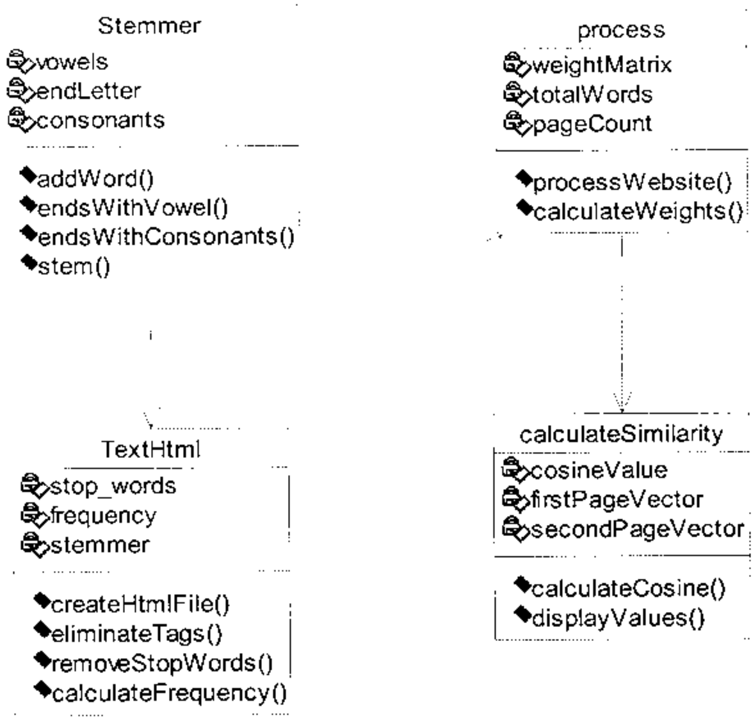


Fig.4.1 Class Diagram for Web Content Mining

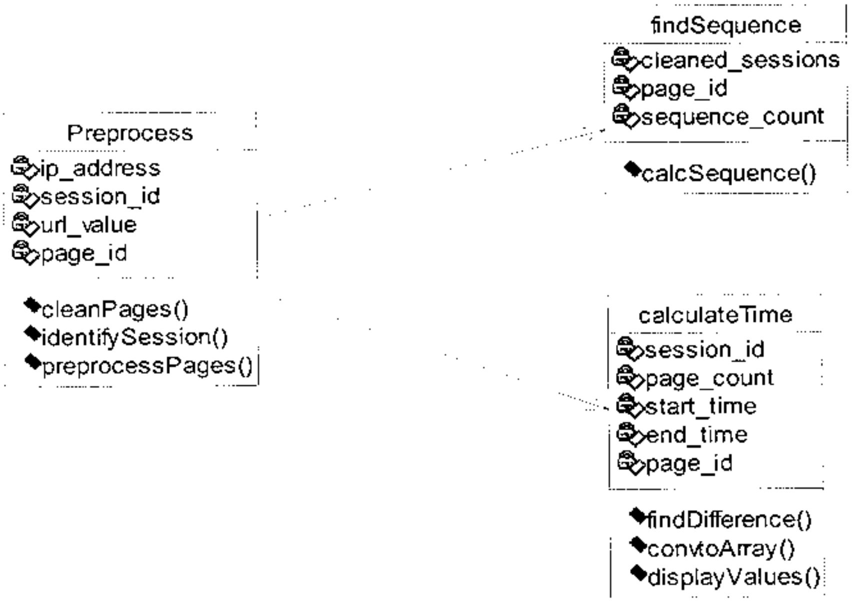


Fig 4.2 Class Diagram for Web Usage Mining

5. SOFTWARE IMPLEMENTATION MODEL

5.1 MODULE DESCRIPTION

The proposed system consists of the following modules

I. Java Modules

1. Content Processing
2. Log files Processing
3. Sequence Parameter Calculation
4. Similarity Parameter Calculation
5. Time Parameter Calculation
6. MATLAB Preprocessing

II. MATLAB Modules

1. Identification of most similar pages
2. Identification of most interesting pages
3. k-means implementation

5.2 IMPLEMENTATION OF MINING PROCESSES

5.2.1 CONTENT PROCESSING

In this module,all the pages' source code are extracted, the tags in the files removed, the stopwords eliminated and the frequency of each word in the website is calculated using the formula below.

$$\mathbf{M_{ij} = f_{ij} * \log (Q/n_i)}$$

where M_{ij} is the weight of word i in page j

f_{ij} is the frequency of the word i in the j th page

Q is the number of pages in the website

n_i is the number of documents the word i appears in the website

5.2.2 LOG FILES PROCESSING

This module is to clean the server logs in order to remove irrelevant information and to identify unique users. The web log registers contain information about the browsing behavior of the web site visitors, in particular the page navigation sequence and the spent time in each page visited in chronological order. It consists of following submodules:

MODULE	DESCRIPTION
Data Cleaning	Remove log entries of graphic files, irrelevant user requests and implicit entries.
User Identification	To identify unique users categorized by IP address
Session Identification	Segregate the IP address into sessions based on session IDs
Path Completion	Complete the navigation paths based on site topology and cache information

Table 5.1 Sub modules in log file processing

5.2.3 SEQUENCE PARAMETER CALCULATION

This module takes as input the log file which is arranged sessionwise. Each session entry's page is considered as one entry in the navigation sequence. The sequences are validated with the site's topology considering cache consistency and back button tracking.

5.2.4 SIMILARITY PARAMETER CALCULATION

This module is to find the extent of similarity that exists between two pages of the website. More the words same in the two pages, the higher the similarity number. It may vary between 0 and 1. The similarity between the two weight vectors of the two pages is calculated as the cosine between them.

$$dP(WP^i, WP^j) = \frac{\sum_{k=1}^{R} wp^i_k \, wp^j_k}{\sqrt{\sum_{k=1}^{R} (wp^i_k)^2} \sqrt{\sum_{k=1}^{R} (wp^j_k)^2}}$$

Where WP^i,WP^j – Page vectors of page i an j

R- total number of words in the website

wp^i-weight of word i in page vector WP^i

5.2.5 TIME PARAMETER CALCULATION

This phase takes care of calculating the time difference between two valid log entries. This difference is assumed to be the time spent in the first page.(It is expressed in milliseconds). The session time out in this case is assumed to be 30 minutes.

5.2.6 MATLAB PREPROCESSING

This acts as an interface between the MATLAB and Java coding. Here the sequence, time and similarity parameters are written into Excel files which are directly passed as input to the MATLAB programs. The time entries and the navigation sequences are converted into arrays for easy processing. While processing, the minimum number of pages is considered 3 and the maximum is set to 6 pages.

5.3 IMPLEMENTATION OF RESULT RETRIEVAL

5.3.1 IDENTIFICATION OF MOST SIMILAR PAGES

In this program, the pages and the similarity measures are plotted in a three dimensional graph. From the graph, the most similar pages are identified which have the highest similarity measures.

5.3.2 IDENTIFICATION OF MOST INTERESTING PAGES

This program takes care of identifying the catchiest pages to majority visitors. This is asserted by the time spent in those pages. The pages and the times spent in each page are plotted in a two dimensional graph. From this, a time span of 30 seconds to 1 minute is plotted in a separate child graph. In that, the pages that get the highest number of hits are considered to be the most interesting pages.

5.3.3 k-means IMPLEMENTATION

This program proceeds to identify the navigation path traversed by most of the visitors. This is done by plotting the sequences of all the visitors and applying k-means algorithm to organize them into clusters. The clusters formed should pertain to their centroid values. From the clustered data, the most populated cluster is identified. This is taken to be the winning cluster and the centroid of this cluster is the winner navigation sequence.

6. PRODUCT TESTING

The system testing deals with the process of testing the system as a whole. This is done after the integration process. The entire system is tested by traversing each module from top to bottom. The verification and validation process are being carried out. The errors that occur at the testing phase are eliminated and a well functioning system is developed.

6.1 UNIT TESTING

It focuses verification effort on the smallest unit of software design, the module. It is also known as module testing. The modules are tested separately. The testing is carried out usually during programming stage itself.

Each and every module is tested separately to check if its intended functionality is met. Some unit testing tasks performed are listed below.

- Check whether the server is running in the correct mode and the connectivity between the clients and the server.
- Check whether the log file is in the required format
- Verify whether all the log file entries are cleaned properly and sent to the corresponding module
- Check whether the time format in the log file is in the required format of

  DD/MM/YYYY HH:MM:SS

- Check whether the time and sequence files are presented correctly to the MATLAB interface.
- Check whether the silhouette value of the k-means clusters are within the acceptable range.

6.2 VALIDATION TESTING

Validation is a process of finding out if the product being built is right, i.e. whatever the software product is being developed, and it should do what the user expects it to do. The software product should functionally do what it is supposed to, it should satisfy all the functional requirements set by the user. Validation is done during or at the end of the development process in order to determine whether the product satisfies specified requirements.

After the validation test has been conducted, one of the two possible conditions exists:

- The functions and the performance characteristics confirm to the specification and are accepted.
- Deviation from the specification is uncovered and the deficiency list is created.

6.3 OUTPUT TESTING

After performing the validation testing, the next step is the output testing of the proposed system since no system is useful if it does not produce the required output in the specific format. The outputs generated and displayed by the system under consideration are tested by the users about the formats required by them. The output formats tested are as follows:

- Verify whether the page numbers assigned and the actual pages in the website match with each other.
- Verify whether the session entries are grouped together and displayed in the correct format.
- Ensure whether the actual and the modified webpages open in separate browsers for the sake of clarity.

6.4 INTEGRATION TESTING

It is the testing performed to detect errors on interconnection between modules. Here, all the modules pertaining to the log and content processing in Java and the cluster formation and identification in MATLAB are integrated to form the entire application and tested to ensure that they work in synchronization and without interference from each other.

6.5 SYSTEM TESTING

The system is tested against the software requirements specification to see if all the requirements are met and if the system performs as per the client's expectations. The system is tested as a whole to check for its functionality. Non functional requirements like performance considerations and platform support are checked as a whole.

7. FUTURE ENHANCEMENTS

As this project is completely done in java, it provides greater flexibility and reusability. Any kind of change can be made to the project without any major change in the underlying functionality. The classes are defined clearly with the necessary access parameters so that they can be modified easily and any additional classes can be added in case of any additional functionality required in the future.

The following modules can be optionally added to the system when the need arises.

- The presented methodology can be improved by introducing advanced variables derived from visitor sessions.
- Any advanced clustering algorithm like self organizing maps can be used instead of the present k-means algorithm.

8. CONCLUSION

We proposed a way to study the visitor behavior in a Web site, based on web content and usage mining. The result is the definition of a new similarity measure based on three characteristics derived from the visitor sessions: the sequence of visited pages, their content and the time spent in each one of them. Using this similarity in k-means clustering algorithm, we found clusters from visitor sessions, which allow us to study the user behavior in the web. The similarity introduced, can be very useful to increase the knowledge about the visitor behavior in the web.

Using this knowledge, we proceeded to optimize the website by adding catchy advertisements to the interesting pages, inserting additional links to link similar pages and many other similar techniques.

9. APPENDIX

9.1 SAMPLE SOURCE CODE

CONTENT PROCESSING

TextHtml.java

```
import java.io.*;
import java.net.*;
import java.text.*;
import java.util.*;
import javax.swing.*;
import javax.swing.text.*;
import javax.swing.text.html.*;
import javax.swing.text.html.HTML.Tag;
import javax.swing.text.html.parser.*;


public class TextHtml extends HTMLEditorKit.ParserCallback
{

   private PrintWriter pw;
   private List stopw=new ArrayList();
   private List searchw=new ArrayList();
   private ListIterator searchitr,stopitr;
   private boolean insideStyleTags;
```

```
public TextHtml(){}

                //Opens a new file to print the output
public TextHtml(File outputFile) throws IOException
{
  pw = new PrintWriter(new BufferedWriter(new
                                  FileWriter(outputFile)),false
                             );
}
                //Opens an input stream to read the source code of the
                //HTML page and write it to a new file
public void createHtmlFile(String s)throws IOException
{
    PrintWriter outhtml =new PrintWriter(new FileWriter(new
                                        File("htmlfile.txt")));
    URL u = new URL(s);
    InputStream is = u.openStream();
    DataInputStream dis = new DataInputStream(new
                                    BufferedInputStream(is));
    BufferedReader br = new BufferedReader(new
                                    InputStreamReader(dis));
    while ((s = br.readLine()) != null)
    outhtml.println(s);
    outhtml.close();
 }
public void handleEndTag(Tag t, int pos)
```

```
{
    if (t == HTML.Tag.STYLE)
            insideStyleTags = false;
}
public void handleStartTag(Tag t, MutableAttributeSet a, int pos)
{
    if (t == HTML.Tag.STYLE)
            insideStyleTags = true;
}
            //Remove tags from the html file using html tag parser.
            //The built  in method is overridden here
public void handleText(char[] data, int pos)
{
    if (insideStyleTags)
            return;
    pw.println(data);
}

public void flush()
{
    pw.close();
}
```

```
                 //This method removes stopwords like punctuation
                 //marks and pronouns, prepositons etc… which are
                 //fetched from a file ‘stoplist.txt’
public void removeStopWords(BufferedReader sbr,BufferedReader
                   fbr)throws IOException
{
    PrintWriter outres =new PrintWriter(new FileWriter(new
                                        File("stopresult.txt")));
    String line,word;
    searchw.clear();
    stopw.clear();
    BreakIterator bi = BreakIterator.getWordInstance();
     while ((line = sbr.readLine()) != null)
   {
     bi.setText (line);
     int start = bi.first();
     for (int end = bi.next(); end != BreakIterator.DONE; start = end,
                                                      end = bi.next())
      {
           word =line.substring (start, end).trim();
            if (word.length() > 0)
                  stopw.add(word);
      }
   }
   sbr.close();
   bi = BreakIterator.getWordInstance();
   while ((line = fbr.readLine()) != null)
```

```
  {
     bi.setText (line);
     int start = bi.first();
     for (int end = bi.next(); end != BreakIterator.DONE; start = end,
                                                            end = bi.next())
     {
        word =line.substring (start, end).trim().toLowerCase();
          if (word.length() > 0)
             searchw.add(word);
     }
  }
    searchw.removeAll(stopw);
    searchitr=searchw.listIterator();
    while(searchitr.hasNext())
            outres.println((String)searchitr.next());
    outres.close();
    searchw.clear();
    sbr.close();
    fbr.close();
}
                    //This method is written to calculate the frequency i.e.
                    //the number of times each word appears in the HTML
                    //page selected.
public Map calcFrequency()throws IOException
{
        BufferedReader br=new BufferedReader(new
        FileReader("result.txt"));
```

```
String line,word;
searchw.clear();
BreakIterator bi = BreakIterator.getWordInstance();
 while ((line = br.readLine()) != null)
 {
                 bi.setText (line);
                 int start = bi.first();
          for(int end = bi.next(); end != BreakIterator.DONE;  start =
          end, end = bi.next())
          {
                  word =line.substring (start, end).trim();
                 if (word.length() > 0)
                          searchw.add(word);
          }
}
  br.close();
  Map map=new HashMap();
  int token_length=searchw.size();
  int m=-1;
  int freq[]=new int[token_length];
  for(int i=0;i<token_length;i++)
          freq[i]=0;
   ListIterator lit=searchw.listIterator();
   ArrayList ref=new ArrayList();
   while(lit.hasNext())
   {
          String a=(String)lit.next();
```

```
            if(ref.contains(a)==false)
                ref.add(a);
        }

        ListIterator refitr=ref.listIterator();
        while(refitr.hasNext())
        {
            m = m + 1;
            word=(String)refitr.next();
            freq[m]= Collections.frequency(searchw,word);
            map.put(word,freq[m]);
        }
        searchw.clear();
        ref.clear();
        return map;

    }




        /*This is the final integrated method which accepts an URL as an
input, reads the source code of the page, parses the HTML tags, removes
stopwords, stems the words and calculates the frequency of each word.*/

 public Map processPage(String s) throws IOException
    {
        Stemmer st=new Stemmer();
        createHtmlFile(s);
        ParserDelegator parser = new ParserDelegator();
        TextHtml app = new TextHtml(new File("plaintext.txt"));
        parser.parse(new FileReader("htmlfile.txt"), app, true);
         app.flush();
        BufferedReader sbr = new BufferedReader (new FileReader
                                                              ("stoplist.txt"));
        BufferedReader fbr = new BufferedReader (new FileReader
                                                              ("plaintext.txt"));
        removeStopWords(sbr,fbr);
                        //Stemming algorithm to stem each word to its root is
                        //called here
        st.stemming("stopresult.txt");
        Map m=calcFrequency();
        return m;
    }
}
```

Process.java

```
import java.io.*;
import java.util.*;
import java.util.Map.Entry;
public class process
{
 public int pagecnt;
 public int total_words;
                    //This method processes the entire website's pages
                    //and calculates the frequency of each word in the
                    //whole website
 public double[][] processWebSite()throws IOException
 {
            BufferedReader br=new BufferedReader(new
     FileReader("pages.txt"));
     TextHtmlTest obj=new TextHtmlTest();
     Set all_words=new HashSet();
     pagecnt=0;
     int i=0,counter=0;
     String line;
     while((line=br.readLine())!=null)
            pagecnt=pagecnt+1;
     Map m[]=new Map[pagecnt];
     Map total_map=new HashMap();
     br.close();
     br=new BufferedReader(new FileReader("pages.txt"));
```

```
while((line=br.readLine())!=null)
{
 m[i]=new HashMap();
 m[i]=obj.processPage(line);
 i=i+1;
}
for(i=0;i<pagecnt;i++)
 all_words.addAll(m[i].keySet());
Iterator a=all_words.iterator();
while(a.hasNext())
{
        String word=(String)a.next();
        counter=0;
        for(i=0;i<pagecnt;i++)
        {
         if(m[i].containsKey(word))
                counter=counter+1;
        }
        total_map.put(word,counter);
}
total_words=total_map.size();
                        /*A two dimensional matrix is declared to hold
                        the weights of all words. The matrix has the
                        total number the words in the website as rows
                        and the total  number of pages in the website as
                        columns*/
double weightmatrix[][]=new double[total_words][pagecnt];
```

```
     Iterator itr=total_map.entrySet().iterator();
     i=0;
     int j=0;
     while(itr.hasNext())
     {
      Map.Entry e=(Map.Entry)itr.next();
      j=0;
      while(j<pagecnt)
      {
       if(m[j].containsKey(e.getKey()))
       {
        Integer value=(Integer)m[j].get(e.getKey());
        Integer value1=(Integer)e.getValue();
                    //The weight of each word is calculated here.
        weightmatrix[i][j]=value*(Math.log(pagecnt/value1));
       }
       else
        weightmatrix[i][j]=0;
       j=j+1;
      }
      i=i+1;
    }
     return weightmatrix;
  }
}
```

SIMILARITY PARAMETER CALCULATION

calculateSimilarity.java

```
import java.io.*;
public class calculateSimilarity
{

            //This program is written to calculate the level of similarity
            //between  two pages of the website. The weights of each
            //word in the two pages are extracted from the weight matrix
            //and compared using dot product of the two vectors.
      public static void main(String args[])throws IOException
      {
            int i,j;
            int page1,page2;
            process pro=new process();
            double weight[][]=pro.processWebSite();
            double sump1=0,sump2=0,prod=0;
            double sim=0;
            int p1,p2;
            System.out.println("Page1\tPage2\tSimilarity");
            for(p1=0;p1<50;p1++)
            {
                  for(p2=p1+1;p2<50;p2++)
                  {
                        prod=0;
                        sim=0;
                        sump1=0;
                        sump2=0;
                        for(i=0;i<weight.length;i++)
                        {
                        sump1=sump1+(weight[i][p1]*weight[i][p1]);
                        sump2=sump2+(weight[i][p2]*weight[i][p2]);
                        sim=sim+(weight[i][p1]*weight[i][p2]);
                        }
                        prod=Math.sqrt(sump1)*Math.sqrt(sump2);
                        //The cosine formula is implemented here
                        // cos A. B= A . B/|A||B|
                        sim=sim/prod;
                        page1=p1+1;
                        page2=p2+1;

        System.out.println(page1+"\t"+page2+"\t"+sim);
                    }
            }
        }
}
```

MATLAB PROGRAMS

k-means PROGRAM TO CLUSTER SEQUENCES

Sequ.m

```
load sequ.txt
binseq=dec2bin(sequ')-'0';
res=reshape(binseq',30,71)';
[ci,cm]=kMeans(res,6,'dist','Hamming');
[s,h]=silhouette(res,ci,'hamming');
[aa,i]=sort(s);
m=mean(s);
```

sixdimensions.m

```
load sequ.txt
%[ci,cm]=kmeans(sequ,5,'dist','cityblock');
%[s,h]=silhouette(sequ,ci,'sqeuclidean');
ptsymb = {'bs','r^','md','go','m+','k.'};
for i = 1:6
  clust = find(ci==i);
  plot3(sequ(clust,1),sequ(clust,2),sequ(clust,3),ptsymb{i},sequ(clust,4),
                                        sequu(clust,5),sequ(clust,6),ptsymb{i});
   hold on
end
   plot3(cm(:,1),cm(:,2),cm(:,3),'ko',cm(:,4),cm(:,5),cm(:,6),'ko'); hold on
   plot3(cm(:,1),cm(:,2),cm(:,3),'kx',cm(:,4),cm(:,5),cm(:,6),'kx');
```

PROGRAM TO CALCULATE MOST INTERESTING PAGES USING TIME PARAMETER

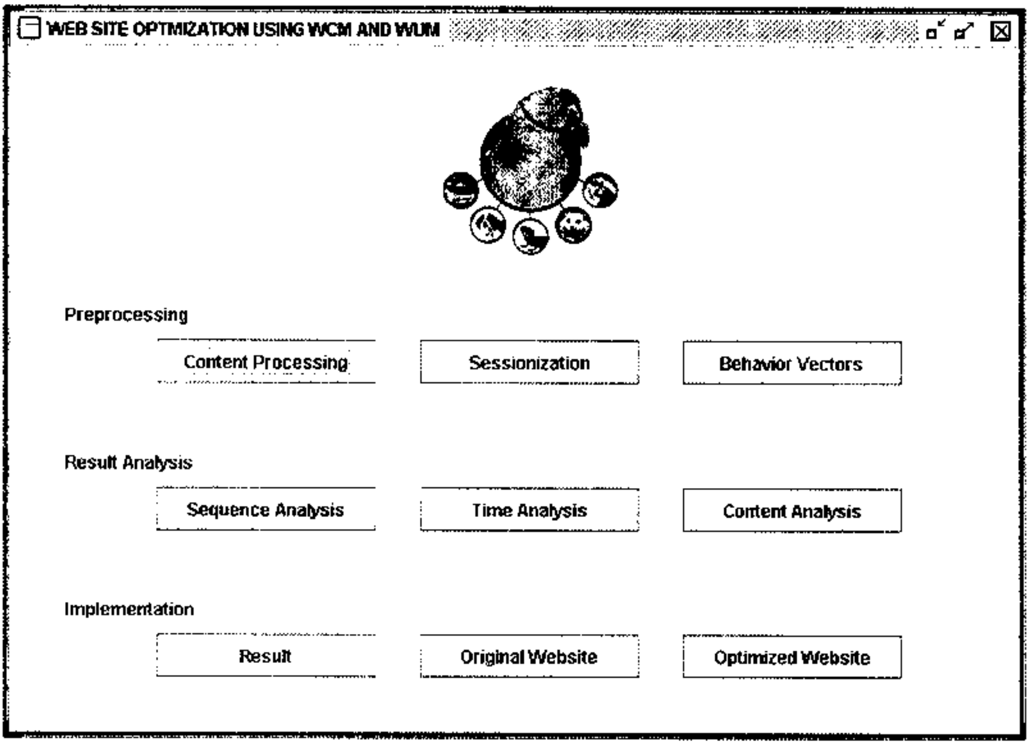
Timecal.m

```
load timefile.txt;
sortarray=sortrows(timefile,2);
timelimit=sortarray((sortarray(:,2)>=30000)&(sortarray(:,2)<=60000),:);
pages=timelimit(:,1);
uni=unique(pages);
hits=histc(pages,uni);
[a,i]=sort(hits);
bb=uni(i);
cc=bb(end:-1:end-2);
result=[cc,a(end:-1:end-2)];hold off;
subplot(1,2,1);
plot(timefile(:,2),timefile(:,1),'b*');
xlabel('Time(msec)');
ylabel('Pages');
hold on;title('TIME SPENT ON EACH PAGE');
subplot(1,2,2);
r=plot(uni,hits,'r*');
hold on;
title('PAGE HITS BETWEEN 30 SECS AND 1 MINUTE');
ko=plot(result(:,1),result(:,2),'ko');legend(ko,'Top pages');
xlabel('Pages');
ylabel('Hits');
```

9.2 SAMPLE OUTPUT

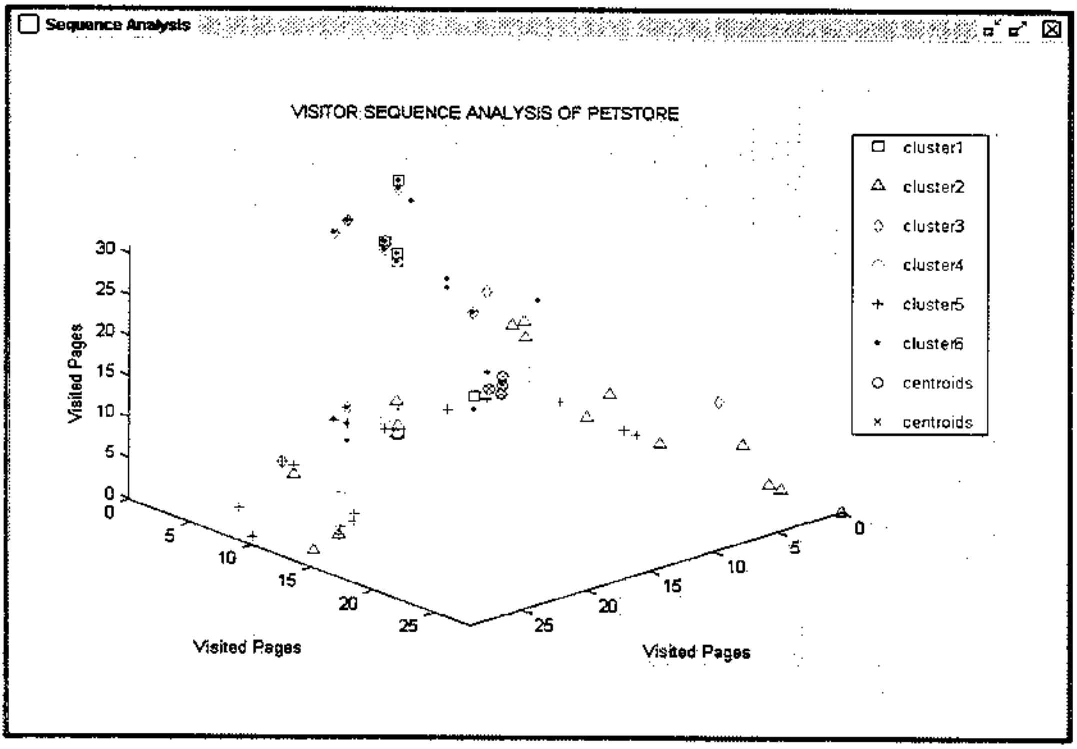
MAIN SCREEN

This screen shows the main page of our system. It contains options to view the entire project module by module.



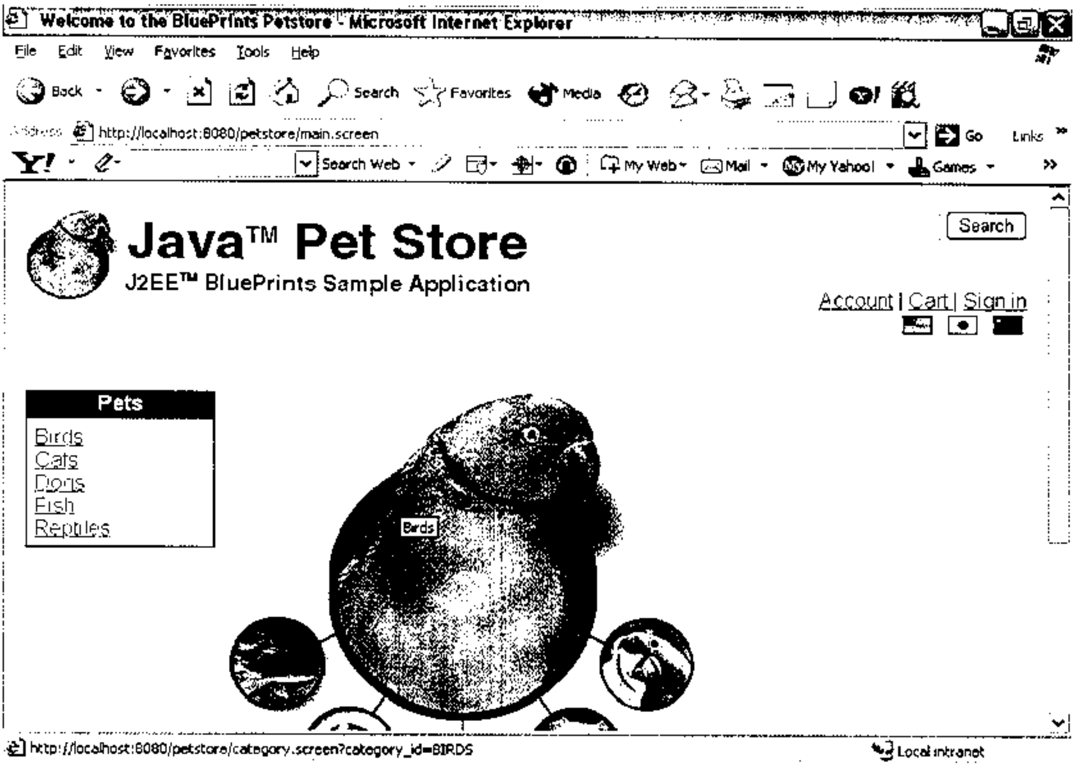
SEQUENCE ANALYSIS USING k-means

The k-means algorithm is implemented here using MATLAB to identify the most traversed sequence by organizing the sequences into clusters.



WEB SITE – HOME PAGE

This is the home page of the sample website under consideration. This is one of the sample websites from java.sun.com.



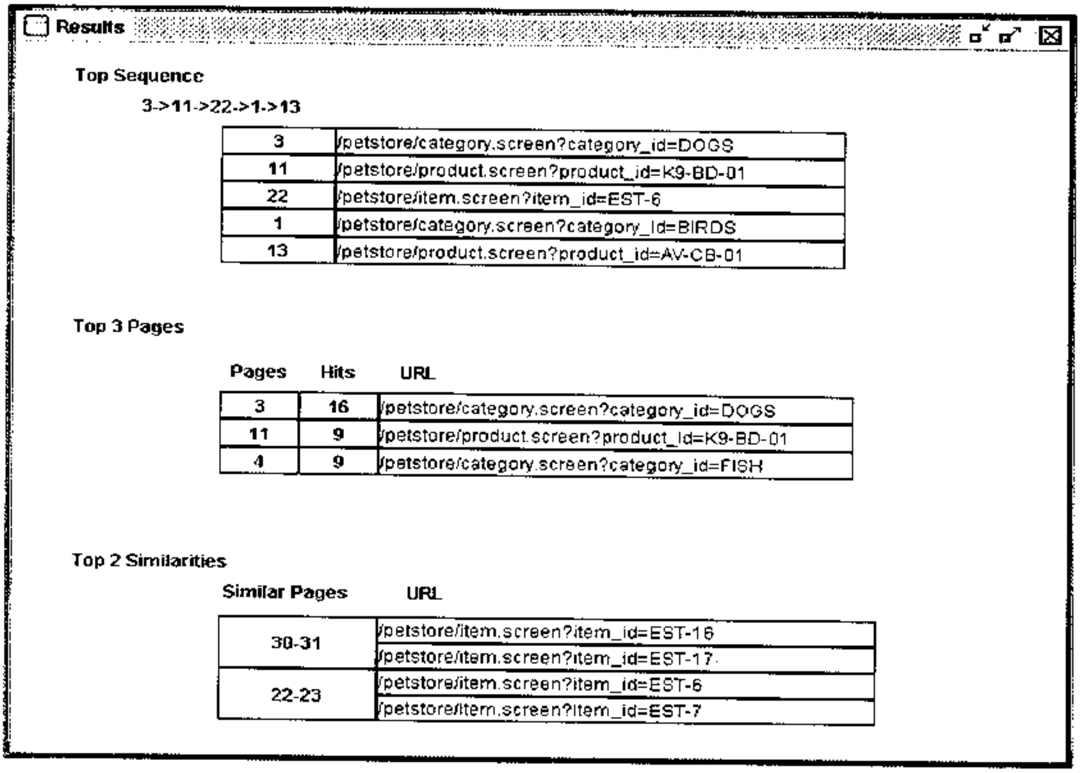
Welcome to the BluePrints Petstore - Microsoft Internet Explorer

Address http://localhost:8080/petstore/main.screen

http://localhost:8080/petstore/category.screen?category_id=BIRDS

RESULT SCREEN

This is the results screen that shows the results derived from all the three kinds of analysis performed. These are the pages to be optimized.



Top Sequence

3->11->22->1->13

3	/petstore/category.screen?category_id=DOGS
11	/petstore/product.screen?product_id=K9-BD-01
22	/petstore/item.screen?item_id=EST-6
1	/petstore/category.screen?category_Id=BIRDS
13	/petstore/product.screen?product_id=AV-CB-01

Top 3 Pages

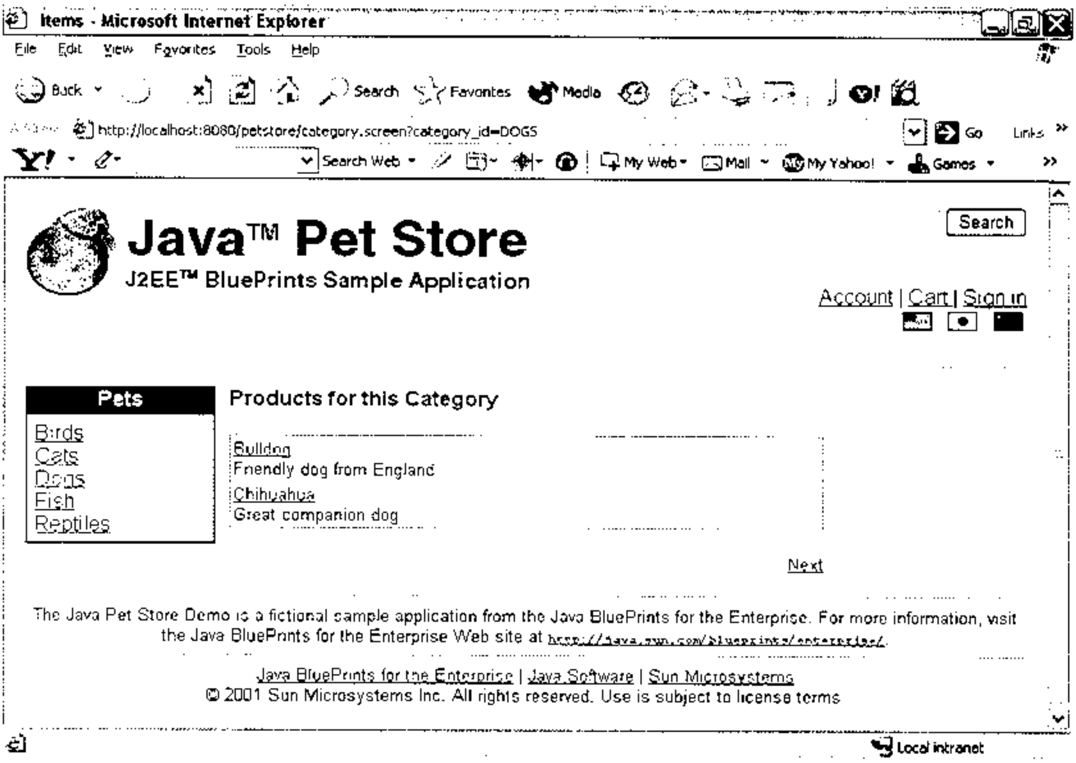
Pages	Hits	URL
3	16	/petstore/category.screen?category_id=DOGS
11	9	/petstore/product.screen?product_Id=K9-BD-01
4	9	/petstore/category.screen?category_id=FISH

Top 2 Similarities

Similar Pages	URL
30-31	/petstore/item.screen?item_id=EST-16
	/petstore/item.screen?item_id=EST-17
22-23	/petstore/item.screen?item_id=EST-6
	/petstore/Item.screen?Item_id=EST-7

WEB PAGE BEFORE OPTIMIZATION

A sample page of the website before optimization is shown here.


Items - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Media
http://localhost:8080/petstore/category.screen?category_id=DOGS
Go
Links
Search Web
My Web
Mail
My Yahoo!
Games
Local intranet

Search

Java™ Pet Store

J2EE™ BluePrints Sample Application

Account | Cart | Sign in

Pets

Birds
Cats
Dogs
Fish
Reptiles

Products for this Category

Bulldog
Friendly dog from England

Chihuahua
Great companion dog

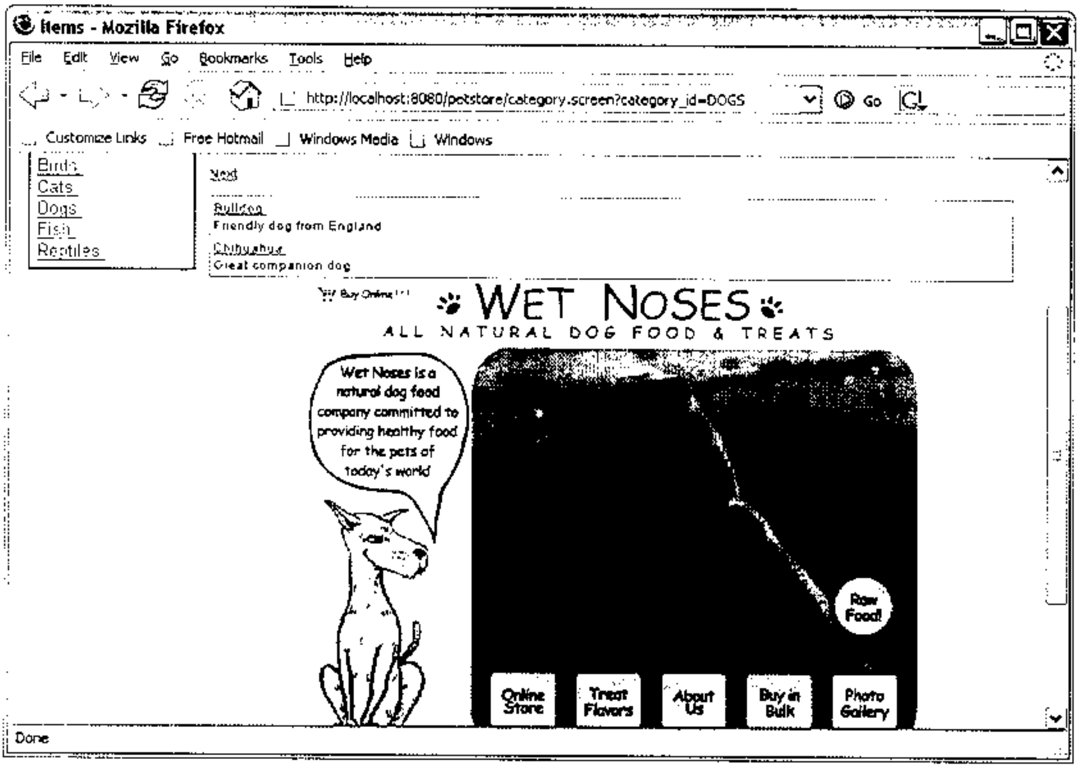
Next

The Java Pet Store Demo is a fictional sample application from the Java BluePrints for the Enterprise. For more information, visit the Java BluePrints for the Enterprise Web site at http://java.sun.com/blueprints/enterprise/.

Java BluePrints for the Enterprise | Java Software | Sun Microsystems
© 2001 Sun Microsystems Inc. All rights reserved. Use is subject to license terms

WEB PAGE AFTER OPTIMIZATION

This is the same page that has been optimized. This page has been identified as one of the most interesting pages from our time analysis.



Items - Mozilla Firefox

http://localhost:8080/petstore/category.screen?category_id=DOGS

Birds
Cats
Dogs
Fish
Reptiles

Next

Bulldog
Friendly dog from England

Chihuahua
Great companion dog

Done

File Edit View Go Bookmarks Tools Help
Go
Customize Links
Free Hotmail
Windows Media
Windows
Buy Online!!!
WET NoSES
ALL NATURAL DOG FOOD & TREATS
Wet Noses is a natural dog food company committed to providing healthy food for the pets of today's world
Raw Food!
Online Store
Treat Flavors
About Us
Buy in Bulk
Photo Gallery

10. REFERENCES

[1] J.D. Velásquez, H. Yasuda and T. Aoki, Combining the web content and usage mining to understand the visitor behavior in a web site, Proc. 3th IEEE Int. Conf. on Data Mining, 669-672, Melbourne, Florida, USA, November, 2003.

[2] R. Cooley, B. Mobasher, J. Srivastava. "Data Preparation for Mining World Wide Web Browsing Patterns", Journal of Knowlegde and Information Systems Vol. 1, pages 5-32, 1999.

[3] k-means clustering tutorial
Website: http://people.revoledu.com/kardi/tutorial/kMean/index.html

[4] Java Forums
Website: http://forum.java.sun.com/index.jspa

[5] MATLAB developer forums
Website: http://newsreader.mathworks.com/WebX?14@@/comp.soft-sys.matlab

[6] Tutorial on Java Collections Framework
Website:
http://java.sun.com/developer/onlineTraining/collections/Collection.html

[7] Herbert Schildt, Java 2 Complete Reference, Fifth Edition, Tata McGraw-Hill

[8] Java Server Pages Tutorial
Website: http://www.apl.jhu.edu/~hall/java/Servlet-Tutorial/