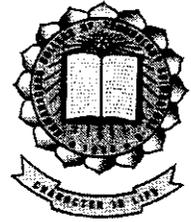


P-3131



MINING ONLINE USERS' ACCESS RECORDS FOR WEB BUSINESS INTELLIGENCE

A PROJECT REPORT

Submitted by

**VINILA C.V
VAISHNAVI R**

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

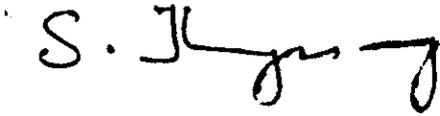
ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2010

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report “**MINING ONLINE USERS’ ACCESS RECORDS FOR WEB BUSINESS INTELLIGENCE**” is the bonafide work of “**Vinila C.V and Vaishnavi R**” who carried out the project under my supervision.



SIGNATURE

Dr. S. Thangasamy

HEAD OF THE

DEPARTMENT

Department of Computer

Science & Engg

Kumaraguru College of Technology,

Chinnavedampatti Post,

Coimbatore – 641606.



SIGNATURE

Mrs. V.S.Akshaya

SUPERVISOR

Senior Lecturer

Department of Computer

Science & Engg(PG)

Kumaraguru College of Technology,

Chinnavedampatti Post,

Coimbatore – 641606.

The candidates with University Register Nos. **71206104059** and **71206104053** were examined by us in the project viva-voce examination held on ...16/04/2019.....



DECLARATION

We hereby declare that the project entitled “**Mining online users’ access records for web business intelligence**” is a record of original work done by us and to the best of our knowledge, a similar work has not been submitted to Anna University or any Institutions, for fulfilment of the requirement of the course study.

The report is submitted in partial fulfilment of the requirement for the award of the Degree of Bachelor of Computer Science and Engineering of Anna University, Chennai.

Place: Coimbatore

Date: 16/04/2010



(C.V.Vinila)



(R.Vaishnavi)

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be but incomplete without the mention of the people who made this possible and whose constant guidance and encouragement crowns all efforts with success.

We are extremely grateful to **Dr.S.Ramachandran**, Principal, Kumaraguru College of Technology for having given us this opportunity to embark on this project.

We express our sincere and heartfelt thanks to **Dr.S.Thangasamy**, Dean, Department of Computer Science and Engineering, for his kind guidance and support.

We would like to express our sincere thanks to our project coordinator **Mrs. P. Devaki**, Assistant Professor, for her valuable guidance during the course of the project.

We would also like to thank our class advisor **Mrs. R. Kalaiselvi**, Senior Lecturer, for her constant support and guidance.

We would like to thank our guide **Mrs. V.S.Akshaya**, Senior Lecturer, Department of Computer Science and Engineering(PG), without whose motivation and guidance we would not have been able to embark on a project of this magnitude. We express our sincere thanks for her valuable guidance, benevolent attitude and constant encouragement.

We reciprocate the kindness shown to us by the staff members of our college, people at home and our beloved friends who have contributed in the form of ideas, constructive criticism and encouragements for the successful completion of the project.

ABSTRACT

The basic aim of this project is to create a user activity monitoring system based on user activities inside multiple websites. It also has an additional feature to it by monitoring user activity on windows applications.

This system creates a user access control tool, which will be installed on the websites. The tool requires the users to register and sign in to the website with their unique account details. Once a user is logged into the website a session is created and their activities are monitored. User may click on the various links and the resource pages inside the website. The information about their activity is stored inside the database. Administrators of the website will be able to view the report about user activity and their preferences on the website. This helps in gaining good business intelligence with less effort and limited resources.

System consists of a Hybrid Account Control tools inside a web server as a web service. This access control tools first checks the user state if the user already has a unique User ID to the websites or requires a new ID. If they are an existing user, an Access Session is automatically created for them inside the database and stores the information about their activity. User Activity includes the pages the user visits, navigation links the user clicks on etc. Monitored information gives information on the interest of

users and customer profiling can be done. In the case of a new user, a registration form with the user details are submitted with a unique User ID and password.

Finally a detailed user activity report is generated and given to the website administrator based on the user's activity session information. With the help of the report, website administrator will able to see the growth and business trend in their E-business application about their user activities.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO.
	ABSTRACT	vi
	LIST OF TABLES	x
	LIST OF FIGURES	xi
	LIST OF ABBREVIATIONS	xiii
	GLOSSARY	xiv
1	INTRODUCTION	1
	1.1 Definition of BI	1
	1.2 Need for BI	1
	1.3 Phases of BI	2
	1.4 Advantages of BI	4
	1.5 Applications	5
	1.6 Real time examples	5
2	OVERVIEW AND SCOPE OF THE PROJECT	6
	2.1 Overview of the project	6
	2.2 Scope of the project	6
3	LITERATURE REVIEW	7
4	FEASIBILITY STUDY	11
5	SYSTEM STUDY	13
	5.1 Existing System	13
	5.2 Proposed System	13
6	SYSTEM REQUIREMENTS	15
	6.1 Hardware Requirements	15
	6.2 Software Requirements	15

7	SYSTEM ANALYSIS	16
8	SOFTWARE OVERVIEW	17
	8.1 Introduction to .NET framework	17
	8.2 Front End	19
	8.3 Back End	22
9	DETAILS OF METHODOLOGY	
	EMPLOYED	24
	9.1 System architecture for proposed system	24
	9.2 Modules Involved	26
	9.3 Proposed System DFD	30
10	CONCLUSION	31
11	APPENDICES	32
	12.1 Appendix-1 SCREEN SHOTS	32
	12.2 Appendix-2 SCOURCE CODE	54
	12.3 Appendix-3 TABLE DESIGN	76
12	REFERENCE	78

LIST OF TABLES

TABLE NO	TITLE	PAGE NO
11.3.1	General User	76
11.3.2	Web Sessions	76
11.3.3	Websites	77

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
9.1	Proposed system architecture	24
9.3	Proposed system DFD	30
11.1.1	Desktop Application- signup form	32
11.1.2	Desktop Application- sign in form	33
11.1.3	Desktop Application- Home page	33
11.1.4	Desktop Application- Corresponding forms	34
11.1.5	Shopping website- sign in page	36
11.1.6	Shopping website- New user registration form	37
11.1.7	Shopping website- Home page	38
11.1.8	Shopping website- product page for bells	39
11.1.9	Online website for mining user activity	40
11.1.10	Online website- Home page	41
11.1.11	Test the service page	42
11.1.12	Web Intelligence	43
11.1.13	Top Countries	44

11.1.15	Customer Interest	46
11.1.16	Popular Products	47
11.1.17	Demographic and gender details	48
11.1.18	Chart-1	49
11.1.19	Chart-2	50
11.1.20	Gender Popularity	51
11.1.21	Sending Professional Emails	52
11.1.22	Email from Admin	53

LIST OF ABBREVIATION

BI	Business Intelligence
URL	Uniform Resource Locator
AC	Access Control
ETL	Extract, Transform and Load
VB	Visual Basic
CLR	Common Language Runtime
CTS	Common Type System
IL	Intermediate Language
MSIL	Microsoft Intermediate Language
ASP	Active Server Pages technology
HTML	Hyper Text Markup Language
XML	Extensible Markup Language
SQL	Structured Query Language

GLOSSARY

Assembly Language

Assembly languages are a type of low-level languages for programming computers.

Business Intelligence

BI represents the tools and systems that play a key role in the strategic planning process of the corporation. These systems allow a company to gather, store, access and analyze corporate data to aid in decision-making.

Data Analysis

Data Analysis is about estimating ongoing trends, integrate disparate information, validate and predict missing information or future trends.

Data mart

It is a department subset of a data warehouse.

Data Mining

It is also called Data Analysis. Data Analysis is about estimating ongoing trends, integrate disparate information, validate and predict missing information or future trends.

Data Sourcing

Data Sourcing is collecting information from multiple sources but in electronic form. Information can be text, pictures, photos, sound

Data Warehouse

A data warehouse collects information about subjects that span an entire organization, thus its scope is enterprise wide.

ETL

Extract, transform, and load (ETL) is a process in database usage and especially in data warehousing that involves: Extracting data from outside sources, Transforming it to fit operational needs (which can include quality levels), Loading it into the end target (database or data warehouse).

Event driven programming

Event-driven programming is a programming paradigm in which the flow of the program is determined by events-messages from other programs or threads.

Exception Handling

A type of error handling that allows control and information to be passed to an exception handler when an exception occurs. Under the OS/2 operating system, exceptions are generated by the system and handled by user code. In the C++ language, try and catch blocks and throw expressions are the constructs used to

Information Logistics

implement C++ exception handling.

Information Logistics is about getting the right information at the right time and right place.

Inheritance

It is a feature of OOP in which a class can inherit behavior and features from one or more super class.

Meta data

Metadata is data about data. It describes the structure, description and administration of data files.

Multidimensional databases

Multidimensional databases are frequently created using input from existing relational databases. It revolves around a central theme.

Multithreading

The ability of an operating system to execute different parts of a program, called threads, simultaneously.

If the system supports parallel processing, multiple processors may be used to execute the threads.

Object Oriented Programming

Object-oriented programming (OOP) is a programming paradigm that uses "objects" – data structures consisting of data fields and methods together with their interactions – to design applications and computer programs.

OLAP

OLAP is the analysis techniques with functionalities such as summarization, consolidation, and aggregation. It also provides the ability to view information in different angles.

OLTP

OLTP systems perform the major task on operational databases which is on-line transaction and query processing and a efficient tool for data management.

Ontology

Ontology is a formal representation of the knowledge by a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to describe the domain.

INTRODUCTION

INTRODUCTION

Most companies collect a large amount of data from their business operations. To keep track of that information, a business and would need to use a wide range of software programs, such as Excel, Access and different database applications for various departments throughout their organization. Using multiple software programs makes it difficult to retrieve information in a timely manner and to perform analysis of the data.

The term Business Intelligence (BI) represents the tools and systems that play a key role in the strategic planning process of the corporation. These systems allow a company to gather, store, access and analyze corporate data to aid in decision-making. Generally these systems will illustrate business intelligence in the areas of customer profiling, customer support market research, market segmentation, product profitability, statistical analysis, and inventory and distribution analysis.

1.1. Definition of Business Intelligence:

The term Business Intelligence (BI) represents the tools and systems that play a key role in the strategic planning process of the corporation. These systems allow a company to gather, store, access and analyze corporate data to aid in decision-making.

1.2. Need for Business Intelligence:

Business intelligence is today's tech priority for a reason: Information is growing at an exponential rate. As information grows, the need for organizations to manage it and make it actionable grows as well. Getting that information in a

timely manner and to the right people in the right places, throughout an organization is an important means to an enterprise success.

- The amount of information is growing rapidly.
- We've reached information overload
- The need for timely information is more pressing.
- Accessing unstructured information is difficult but increasingly necessary to decision making.
- To manage the avalanches of information, business intelligence tools are becoming more widespread.
- Timely business intelligence has become mission critical to many enterprises.

1.3. Phases of Business Intelligence:

Business Intelligence (BI) is a process of growing the competitive benefits of a company by intelligent applying of available data in decision making. By Business intelligence is often meant a software program for the assistance to the business management to increase profits. BI allows finding where the company loses money or where it is possible to earn more. So, expenses on BI should be comparable to the company plans to earn or save. There are five main phases of Business Intelligence:

- Data Sourcing
- Data Analysis
- Situation Awareness
- Risk Assessment

- Decision Support

1.3.1. Data Sourcing

Business Intelligence is aimed to collect information from multiple sources of data. The information can be extracted from: text documents (e.g. notebooks, emails, reports, etc.), pictures and photos, sounds, tables and schedules, web resources and URL lists. The main idea of data sourcing is to obtain the information in electronic form.

1.3.2. Data Analysis

Business Intelligence is aimed to apply benefits from collections of data. The process of data analysis is also called data mining. On this phase BI is about to estimate ongoing trends, integrate disparate information, validate and predict missing information or future trends.

1.3.3. Situation Awareness

Business Intelligence is aimed to filter off all irrelevant information, and then use the remaining information for the best business performance. The manager requires the relevant data (market forces, government policy, competitive advantages, etc.) in order to understand and make right decision.

1.3.4. Risk Assessment

Business Intelligence is about to reveal what probable actions might be taken or decisions made at different times. It is about to help you assess the current and future risk, cost or benefit of taking one action over another. On this phase BI is intended to conclude your best options or choices

1.3.5. Decision Support

The main idea is to apply business intelligence for comprehensive usage of the business information. It aims to warn the company of important events, such as market changes, defects, production decreasing, poor staff performance, so that the managers can take preventative measures. It provides the company with the basis for analyzing and making better business decisions to improve sales or customer satisfaction. To put it simply, decisions support represents the information the manager needs, when he needs it.

1.4. Advantages of Business Intelligence:

The advantages enjoyed by market leaders and made possible by business intelligence include:

- the high responsiveness of the company to the needs of its customers
- recognition of customer needs
- ability to act on market changes
- optimization of operations
- cost-effectiveness
- quality analysis as the basis for future projections
- the best possible utilization of resources

1.5.Applications:

- 1) Financial Analytics
- 2) Service Analytics
- 3) Marketing Analytics
- 4) Human Resources Analytics

1.6.Real Time Example:

- 92% of Americans are concerned (67% very concerned) about the misuse of their personal information on the Internet.
- FTC Report, May 2000
- 86% of executives don't know how many customers view their privacy policies.
- Forrester Report, November 2000

OVERVIEW AND SCOPE OF THE PROJECT

2. OVERVIEW AND SCOPE OF THE PROJECT

2.1. Overview of the project

This project provides a web service to obtain business intelligence on a website or a windows application by using user access records instead of web logs.

The web service requires users to register their profiles

Using the unique User ID and password user can gain the resources on the website or application.

This provides a common profile serving system. A common User ID and password can be used on websites using the web service tool.

2.2. Scope of the project

The project is undertaken is based upon internet applications and applications that run on Windows on specific domains.

The tool serves to provide customer profiling and also as intelligence tool. The feature of customer profiling is to reduce the number of times a user has to login for each website.

A common User Sign Up is provided to login into all websites using the tool. The intelligence tool mines user activity over the websites to provide an overview to improve the business of the website or application.

**LITERATURE
REVIEW**

3. LITERATURE REVIEW

3.1. Mining online users' access records for web business intelligence

Simon Fong and Serena Chan[1] proposed the mining of online users' access records to study the behavior of users on the website. Mining can be performed on a clean set of records by using user information instead of log files. Log files can rather be anonymous as using IP address one cannot identify the user. Using the information given by users during registration with a website, customer centric mining patterns can be found. Also the additional purpose of using user information is for e-personalization. Customers can be provided with e-newsletters for a better insight into using the resources on the website. The proposed system uses a Hybrid Access Control Architecture to integrate the data mining tool along with the website. The website requires certain changes on the links to capture the users' access on the various resource links in the website. The integration of the user information, the collection of users' access records and the mining agent is controlled by the Hybrid AC architecture. The mining agent discovers patterns for business intelligence. The patterns portrays the user study such as customer profiling, demographic classification and also the growth of the resources on the website by giving patterns of popular products and customer preferences over the website. The AC architecture also provides generalized information on the time of login, the user who has login, the purpose (links used by the user) of login. Thus customer trails are captured to provide reliable solutions for business. The reporting agent provides factors for improvement of business and supports in decision making. Hence Web Business Intelligence is provided for customers on the Internet.

3.2. A service oriented architecture for business intelligence

Liya Wu, Gilad Barash, Claudio Bartolini[3] have proposed a service oriented architecture which is an improvement to the traditional business intelligence architecture. In recent times, data is required to be collected from heterogeneous environments. In the traditional system, data marts are available for each data source but they are not integrated components. The SOA proposes to integrate reusable components for better data delivery. From the traditional architecture, the legacy systems are broken down to reusable components. The reusable components communicate using SOAP messaging protocols. The reusable components are flexible and its functionality is extensible. It delivers the truth version of information from data sources and in turn delivers insightful business solutions to the customers which are the data sources.

The integration of reported data includes historic, present and predictive data. The storing of all data may not prove to be very useful and thus cleansed data that are only required by the applications are stored. The data that is stored undergoes ETL (extract, transform and load) transformations. With data from different sources ETL transformations will complicate. This is the reason to use data only based on requirement. For a flexible and scalable ETL process from the legacy system each component is broken down and each consists of an ETL process. This also contributes to easy maintenance of the components. Further the BI modules are used for analytical purpose to deliver reports and business management solutions. Data marts communicate for the deliverance of multi-dimensional data and fast deliverance of data. A single web application is used to support all BI tools at the customer end. Solutions are easily transferred through data marts through the application. The SOA approach provides a system at simple

cost and low maintenance. Another advantage is SOA can integrate with IDE (integrates development environment) tools to simplify the complexity of SOA. In short, SOA makes possible seamless integration of technologies into a coherent business environment enabling simplified data delivery and low latency analytics.

3.3. Developing a framework for business intelligence systems integration based on ontology

Xu Xi and Xu Hongfeng[4] propose an integrated system using ontology for collecting information from various data sources. Information Logistics is about getting the right information at the right time and right place. To use information logistics we require the domain of interest in order to understand the need of information and the structure of information from different sources. For the shared understanding, we require ontology. Ontology models are applied to resource bases that share commonality. Using relation interfaces the system resources are reconfigured for integration. The collection of data is stored in data repositories. The repository consists of a master repository and several sub-work repositories. Master repository consists of security information, topological information and old version of objects. Sub-work repository consists of data models, meta data, business rules and packages and execution scenarios. Further data model patterns are used to reduce the complexity of communication between various data sources with different modelers. Design patterns are used in addition as reusable patterns for taking abstract data and for analysis patterns. These are the primitive features.

The ontology model shares commonality by using knowledge description and a common expression or a common rule structure. Ontology structure has six domains. They are component domain, implementation description domain,

information domain, environment domain, style domain and edition domain. The former four domains describe the content of a component, and the later two domains describe the relationship among components. The ontology is constructed for the base resources and all that is required is integration of service resource system in a protégé environment.

FEASIBILITY STUDY

4. FEASIBILITY

Feasibility study is a test of the system proposal according to the workability impact of the organization, ability to meet user's needs and effective use of resources. The feasibility study is to serve as a decision document: it must satisfy the following factors:

- User demonstrable needs
- Problem worth solving
- Method of solving problem.

4.1. Economical Feasibility:

Economic feasibility is the most frequently used method of evaluating the effectiveness of a candidate system. The procedure is to determine the savings and benefits from the candidate system and compare the costs. If the benefits outweigh the costs then it is decided to go ahead with the project. Otherwise, further justification or alternations in the proposed system will have to be made if it to have a chance of being approved. It is an on-going effort that improves the accuracy at each phase of the system life cycle.

In the economic feasibility study, the following points were found out:

- The automated system will cost bit as the initial expenditure.
- Maintenance also involves some investment in terms of money.

Once the computerized system is installed, it can cater to the needs of the customer and the business without much of manual work, which is more cost-effective for the management. In the sense, the automated system is economically feasible.

P-3131



4.2. Technical Feasibility:

Technical feasibility centers on the existing system and to what extent it can support the proposed system. It involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project will be judged not feasible. Now considering the proposed system, and company planning to implement

So the owner having realized the advantages, benefits and economic feasibility of the new system is ready to afford the extra expense that may arise for the satisfaction of all the hardware and software requirements.

4.3. Operational Feasibility:

People are inherently resistant to change and computers have been known to facilitate change. An estimate should be made how strong a reaction the general public is likely to have towards a new computer system. It is common knowledge that computer installations have a lot to do with the turnover transfer retaining and changes to employee job status. Therefore it is understandable that the introduction of the candidate system requires special effort to educate and train the staff on a new way of conducting business. But since ultimately the introduction of a new system will only reduce the staff's workload, staff's may have no objection to install a computerized system and of course will be eager to extend their co-operation.

4.4. Behavioral Feasibility:

Behavioral feasibility deals with how the developed software behaves in different scenarios when deployed. It is also a very important part in the different stages of software development.

SYSTEM STUDY

5. SYSTEM STUDY

5.1. Existing System

The existing system uses an intelligence tool where users are constrained to sign up only for a particular website. Users are expected to create an ID of their own in each of the website they visit. Thus for each website a separate database management system has to be considered. Also for each user in a website a separate session is to be created.

In some cases user credential are shared. This case occurs only if websites are partner websites i.e. a business tie up exists between both websites. Another possibility of user credentials being shared is when a website administrator posts more than one website on the Internet. As the websites is in the administrator's control databases can be shared.

5.2. Proposed System

The need for a new proposal arises due to the limitations in the existing system. Websites that implement this business intelligence tool can share user credentials. Users can sign up with a unique ID and password

The ID can be used to sign into any of the other websites which fall under this system. All websites share a common database. Also for every user a separate session is created. The session details can be shared between all websites. A main purpose that can be seen from the proposed system is that the customer interest can be monitored between the websites. This intelligence tool can help improve the strategy for the business of the website on the Internet.

An added advantage to the proposed system is desktop applications following the same system can also share the user credentials. Activity on the desktop application can also be stored on the database and with the help of the business intelligence system applications can improve to improve customer interest.

SYSTEM REQUIREMENTS

6. SYSTEM REQUIREMENTS

6.1. Hardware Requirements

Processor Name	: Pentium IV
Processor Speed	: 1.7 GHz
Memory (RAM)	: 256 MB
Hard Disk	: 10 GB
Floppy Drive	: 3 ½ “1.44 MB
Monitor	: Samsung Color Monitor
Keyboard	: 104 keys HCL Multimedia Keyboard
Mouse	: Logitech Optical Mouse

6.2. Software Requirements

Operating System	: Windows 2000 Onwards
Software Tools	: VB.Net, ADO.Net

SYSTEM ANALYSIS

7. SYSTEM ANALYSIS

The primary goal of this phase is to create a detailed Functional Specification defining the full set of system capabilities to be implemented, along with accomplishing data and process models illustrating the information to be managed and the process to be supported by the new system.

All analysis methods are related by a set of operational principles:

- The information domain of a problem must be represented and understood.
- The functions that the software is to perform must be defined.
- The behavior of the software must be defined.
- The model that depicts information function and behavior must be partitioned.
- The analysis process should move from essential information towards implementation detail.

By applying these principles we approach a problem systematically. The information domain is examined so that the functions may be understood completely. Partitioning is applied to reduce complexity.

SOFTWARE OVERVIEW

8. SOFTWARE OVERVIEW

8.1. Introduction to .Net Framework

The .Net Framework is a managed, type-safe environment for application development and execution. The framework manages all aspects of the execution of the program: it allocates memory for the storage of data and instruction, grants or denies the appropriate permissions to the application, initiates and manages application execution, and manages the reallocation of memory for resources that are no longer needed. The .NET Framework consists of two main components: the CLR and the .NET Framework class library.

8.1.1. Common Language Runtime (CLR)

CLR is described as the “execution engine” of .NET. It provides the environment within which programs run. The most important features are:

- Conversion from a low-level assembler-style language called Intermediate Language (IL), into code native to the platform being executed on
- Memory Management, notably including garbage collection.
- Checking and enforcing security restrictions on the running code.
- Loading and executing programs with version control and other such features.

The common language runtime can be thought of as the environment that manages code execution. It provides core services, such as code compilation, memory allocation, thread management, and garbage collection. Through the common type system (CTS), it enforces strict type safety, and it ensures that code is executed in a safe environment by enforcing code access security. The .NET

Framework class library provides a collection of useful and reusable types that are designed to integrate with the common language runtime. The types provided by the .NET Framework are object-oriented and fully extensible, and allow the user to seamlessly integrate the applications with the .NET Framework.

8.1.2. Languages and the .NET Framework

The .NET Framework is designed for cross-language compatibility. Simply put, it means that .NET components can interact with each other no matter what language they were originally Microsoft C++ or any other .NET language. The language interoperability extends to full object-oriented inheritance.

The level of cross-language compatibility is possible because of the common language run time. When a .NET application is compiled, it is converted from the language it was written in (Visual Basic .NET, any other .NET compliant language) to Microsoft Intermediate Language (MSIL or IL). It is a low-level language designed to be read and understood by the common language run time. Because all .NET executables and DLLs exist as intermediate language, they can freely interoperate.

The Common Language Specification defines the minimum standards that .NET language compilers must conform to, and thus ensures that any source code compiled by a .NET compiler can interoperate with the .NET Framework.

The CTS ensures type compatibility between .NET components. Because .NET applications are converted to IL prior to deployment and execution, all primitive data types are represented as .NET types. Thus, a Visual Basic Integer

represented in IL code as a System.Int32. Because both languages use a common and inter convertible type system, it is possible to transfer data between components and avoid time-consuming conversions or hard-to-find errors.

8.2.FRONT-END

8.2.1.VB.NET

Visual Basic .NET is a major component of Microsoft Visual Studio .NET suite. .NET is a Framework in which Windows applications may be developed and run. .NET must go back in time and follow the development of Windows and the advent of Windows programming. The .NET version of Visual Basic is a new improved version with more features and additions. After these new additions, VB qualifies to become a full object-oriented language such as C++.

VB.NET is the following version of VB 6.0. Microsoft .NET is a new programming and operating framework introduced by Microsoft. all .NET supported languages access a common .NET library to develop applications and share common tools to execute applications. Programming with Visual Basic using .NET is called VB.NET. VB.NET, the following version of VB 6.0 is an improved, stable, and full Object Oriented language. In VB 6.0 wasn't a true object-oriented language because there was no support for inheritance, overloading, and interfaces. VB.NET supports inheritance, overloading, and interfaces. Multithreading and Exception handling was two major weeks' areas of VB 6.0. In VB.NET, the user can develop multithreaded applications as the user can do in C++ and C# and it also supports structured exception handling.

Here is list of VB.NET features:-

- Object Oriented Programming language.
- Support of inheritance, overloading, interfaces, shared members and constructors.
- Supports all CLS features such as accessing and working with .NET classes, interaction with other .NET languages, Meta data support, common data types, and delegates.
- Multithreading support.
- Structured exception handling.

8.2.2.ASP.NET

ASP.NET is the latest version of Microsoft's Active Server Pages technology (ASP). ASP+ is the other name for ASP.NET. ASP+ is just an early name used by Microsoft when they developed ASP.Net.

ASP.NET provides a unified Web development model that includes the services necessary for developers to build enterprise-class Web applications. ASP.NET has been designed to work seamlessly with WYSIWYG (What You See Is What You Get) HTML editors and other programming tools, including Microsoft VisualStudio.NET. Not only does it make Web development easier, but it also provides all the benefits that these tools have to offer, including a GUI that developers can use to drop server controls onto a Web page and fully integrated debugging support.

New features in ASP.Net are,

- Programmable controls
- Event-driven programming
- XML-based components
- Higher scalability
- Increased performance – Compiled code
- Easier configuration and deployment

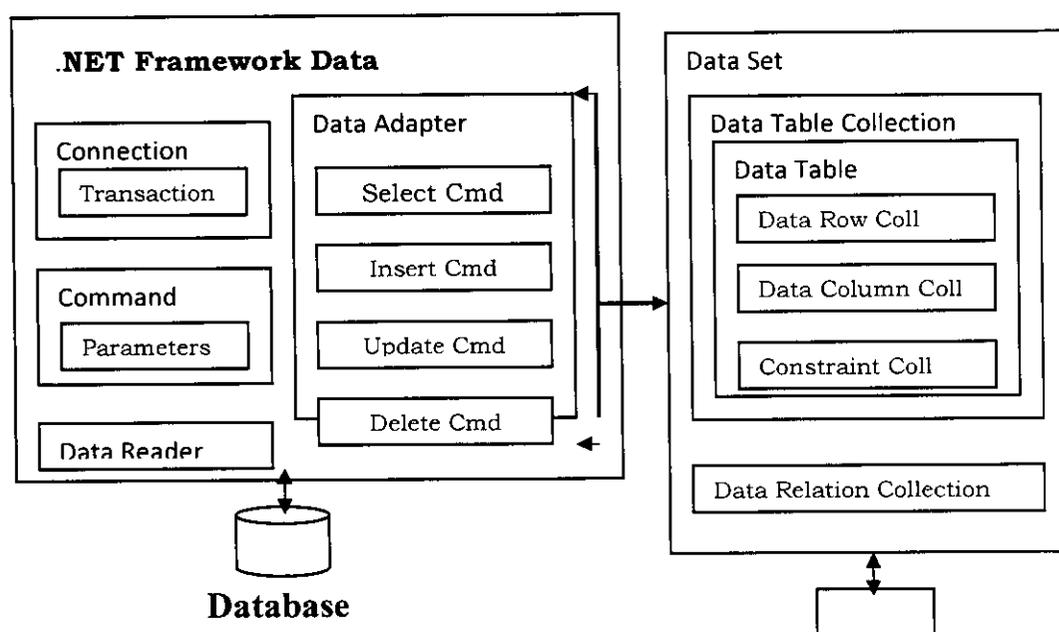
8.2.3. ADO.NET

ADO.NET is a set of classes that expose data access services to the .NET programmer. ADO.NET provides a rich set of components for creating distributed, data sharing applications. It is an integral part of the .NET Framework, providing access to relational data, XML, and application data. ADO.NET supports a variety of development needs, including the creation of front-end database clients and middle-tier business objects used by applications, tools, languages, or Internet browsers.

ADO.NET provides consistent access to data sources such as Microsoft SQL Server, as well as data sources exposed through OLE DB and XML. Data-sharing consumer applications can use ADO.NET to connect to these data sources and retrieve, manipulate, and update data.

ADO.NET cleanly factors data access from data manipulation into discrete components that can be used separately or in tandem. ADO.NET includes .NET Framework data providers for connecting to a database, executing commands, and

retrieving results. Those results are either processed directly, or placed in an ADO.NET Dataset object in order to be exposed to the user in an ad-hoc manner, combined with data from multiple sources, or remote between tiers. The ADO.NET Dataset object can also be used independently of a .NET Framework data provider to manage data local to the application or sourced from XML.



8.3. BACK-END

8.3.1. MS SQL Server 2000

Microsoft SQL Server 2000 extends the performance, reliability, quality, and ease-of-use of Microsoft SQL Server version 7.0. Microsoft SQL Server 2000 includes several new features that make it an excellent database platform for large-

scale online transactional processing (OLTP), data warehousing, and e-commerce applications.

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component.

The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services.

Visual Studio .NET ships with such languages as Visual Basic .NET, and Visual C++ with managed extensions as well as the JScript scripting language. The user can also write managed code for the .NET Framework in other languages. Third party compilers exist for FORTRAN .NET, COBOL .NET, Perl .Net, and a host of other languages. All of these languages share the same cross-language compatibility and inheritability. Thus the user can write code for the .NET Framework in the language of their choice, and it will be able to interact with code written for the .NET Framework in any other language.

**DETAILS OF
METHODOLOGY
EMPLOYED**

DETAILS OF THE METHODOLOGY EMPLOYED

9.1. Proposed system architecture

The following figure (number) shows the architectural model of the proposed system, which contains the different subsystems that are involved and their interrelationships.

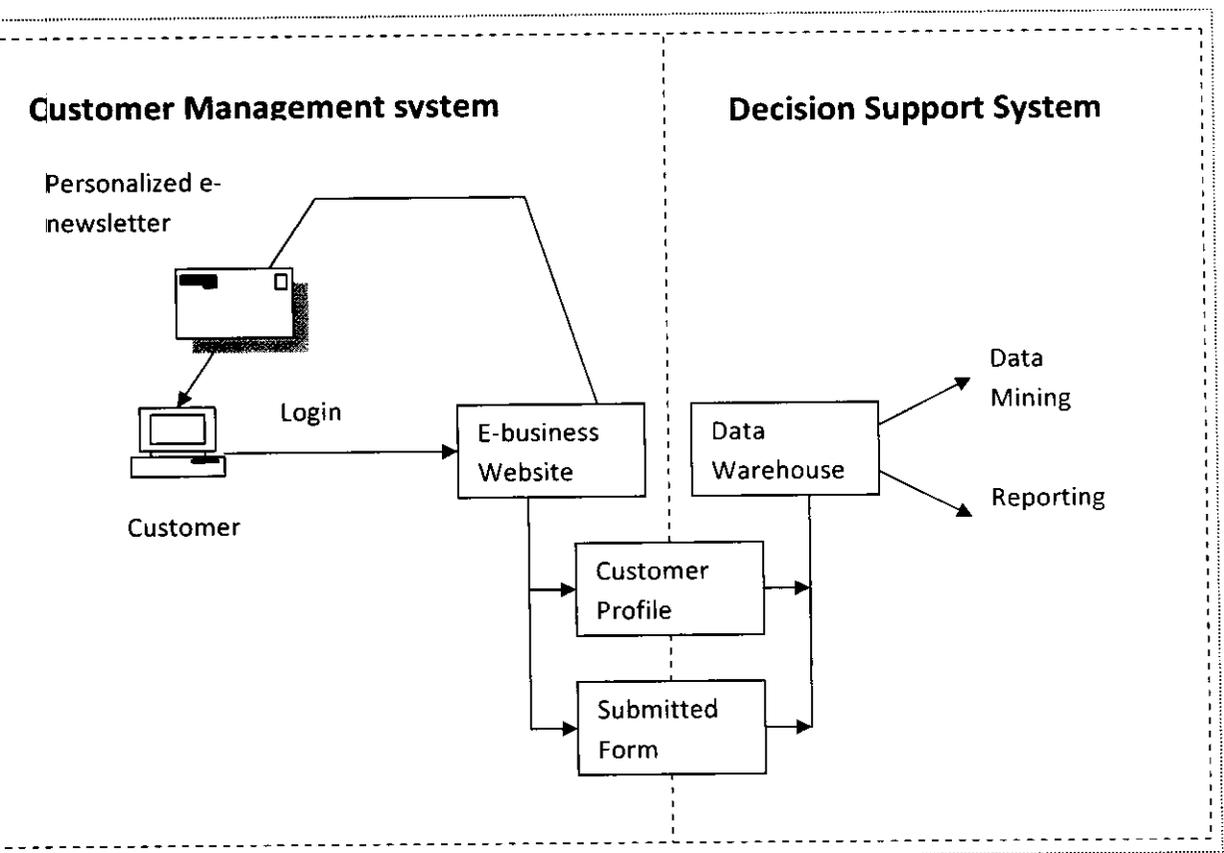


Figure 9.1: Proposed system architecture

9.1.1. Customer Management System

9.1.1.1. E-Business website/application:

The profiling service is offered for e-commerce websites. Users login into the website as soap clients to access the resources on the website. Through the website or application we monitor user activity on the website like navigation links, popular links etc.

9.1.1.2. Personalized E-Newsletter

With the help of intelligence reports, website administrators can send an automatic e-newsletter to all users. The e-newsletter can be edited by the administrator. This option does not require the effort to send a separate e-mail to all users. With this feature, all users will receive the e-newsletter for a personalized service.

9.1.1.3. Customer Profiling System

Users register in the websites in exchange for a password and a unique UserID. In the registration, users give in details that are asked in the form. These details are used for the customer profiling. Customers can be classified to produce intelligence results.

9.1.2. Decision Support System

9.1.2.1. Data Mining Agent

This agent monitors the activity of all users. Once all this activity is recorded, it is mined to observe patterns of user activity. The patterns can help to develop better strategies for the business of the website.

9.1.2.2. Reporting Agent

The reporting agent gives website administrators a structured form of pattern for better understanding of the user activity. The reports generated can show specific areas that require more improvement of business. Reports can be viewed as text or in figural form. The diagrammatic representation provides easier understanding of the growth of e-business.

9.2. MODULES INVOLVED

The following modules are implemented in the project

- Account Controller
- Website Admin

9.2.1. Account Controller:

This is a web service part which contains the code logic and data base to be accessed by other websites of similar type implementing the tool. Coded logic contains the following parts:

9.2.1.1. User Profile

This module is used to manage the user profile data onto the database. This assigns a unique account ID to each user along with a password. A new user can create this unique ID and password through the following sub-process.

This contains a registration form which gets user input and stores them into the database. Registration collects details such as Country, Age, and Gender etc that is used for the profiling service.

9.2.1.2. Activity Session

This module creates and manages the user activity details on the website or an application.

9.2.1.3. Session Information

This module stores the details about their session with their login detail includes time and type of login onto the website.

9.2.1.4. Page Resource access data

This module monitors and stores information about the pages visited by the user, the type of page, and frequency about the visit into their activity session data.

9.2.1.5. Link Navigation Information

This module monitors the information about the link navigation details of each user inside the website or within an application.

9.2.1.6. Activity Details

The information that is monitored through the above modules are stored and managed in the database.

9.2.2. Website Admin:

This module manages the details about the website or application using the access control tools and provides a detail report about the user activity to the administrator.

9.2.2.1. Website registration

This module is used to register the website. Only one administrator user registration is allowed for a particular website. Those administrators who are already registered can add more websites or applications under their account control.

9.2.2.2. Tool download

Once user registration is completed, the tool is now available for downloading. The tool can be used as an internet application for the websites or the tool can be added as a service onto an application.

9.2.2.3. Tool implementation

Tools are automatically configured for the website on which they are going to be installed. To implement the tool on a website, website administrators have to copy the generated content onto their web pages which they want to monitor.

9.2.2.4. Tool Management

This module is used to manage the mining tools. Administrator can add more tools, edit or delete their tools.

9.2.2.5. Tool Sharing

This module provides the tool sharing functionality to the administrator in order to implement the web service tool on their multiple websites.

9.2.2.6. Report Gaining

This module provides the detail report for the administrator about their user activity on their websites. Reports are also figural and are an intelligence tool to show the business trend on the website.

9.3. Proposed System DFD

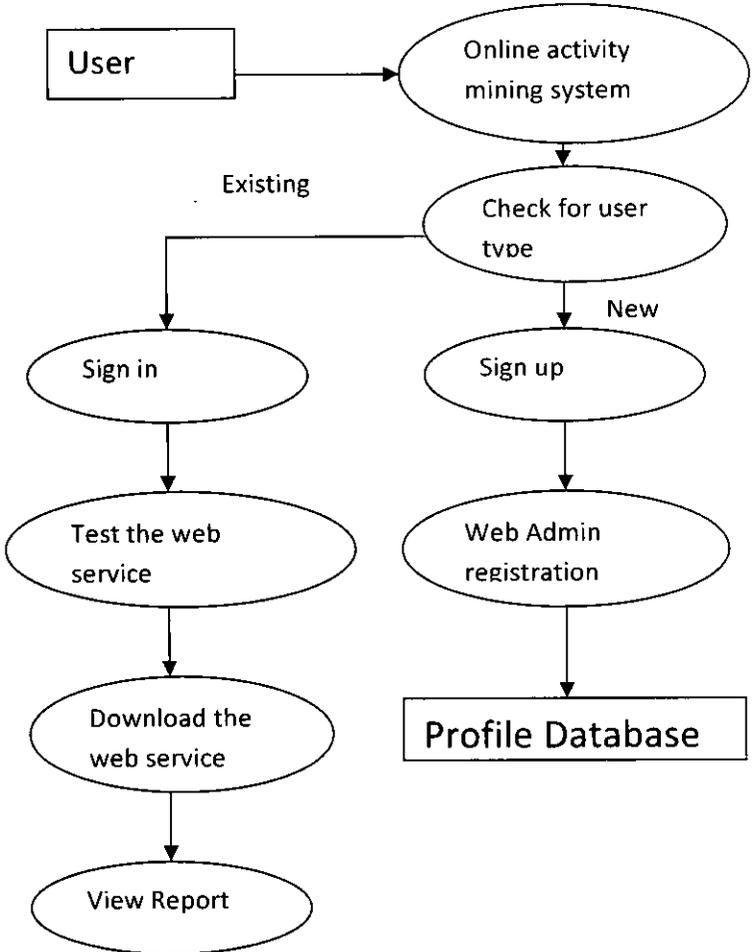


Figure 9.3: Proposed system DFD

CONCLUSION

10. CONCLUSION

The project completed can be used worldwide as it is easily available on the internet. Portability issues are alleviated through this. The system can be loaded into any windows operating system also making it usable by desktop applications. The project uses ASP.net that makes it more flexible, portable and more secure. Added it also support dynamic content forms.

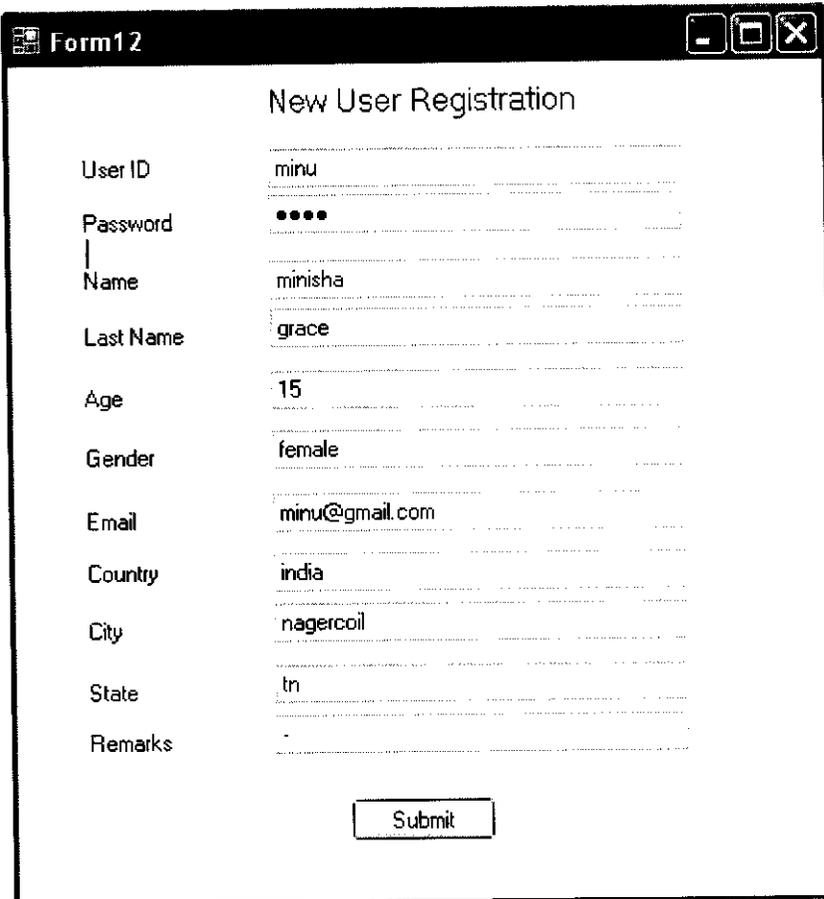
To usage and implementation of the tool is very user-friendly. The tool also reduces the overhead of creating user IDs for every website. With a single ID users can make use of the resources on various websites under this system. The addition of a application to this system also increase the scope to allow internet and desktop applications to use the system.

APPENDIX-1

12. APPENDIX

12.1. APPENDIX-1: SCREEN SHOTS

12.1.1. Desktop Application- signup form



The screenshot shows a desktop application window titled "Form12" with standard Windows window controls (minimize, maximize, close). The main content area is titled "New User Registration" and contains a form with the following fields and values:

Field	Value
User ID	minu
Password	••••
Name	minisha
Last Name	grace
Age	15
Gender	female
Email	minu@gmail.com
Country	india
City	nagercoil
State	tn
Remarks	-

At the bottom center of the form is a "Submit" button.

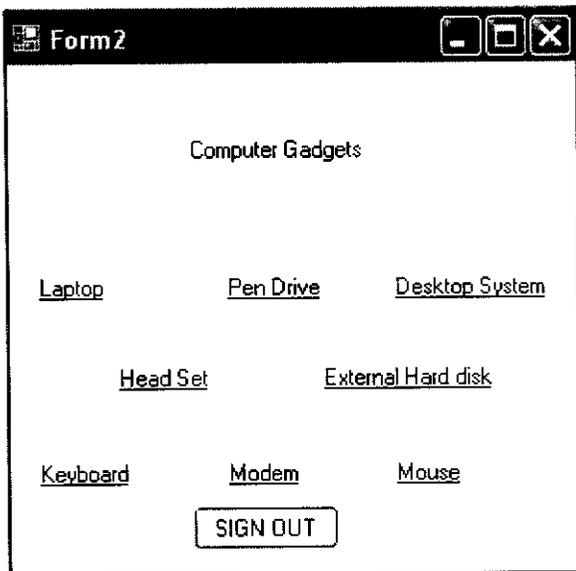
This figure shows the signup form for the desktop application where the user enters age, name, mail id, gender and demographic details.

12.1.2. Desktop Application- Sign in form

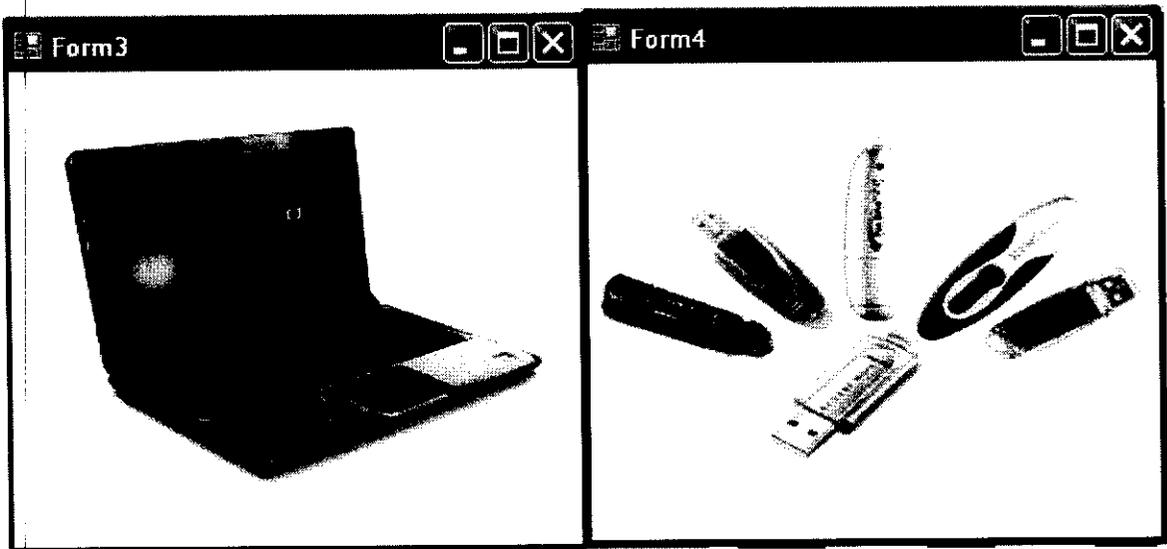


This figure shows the sign in form for the desktop application where the user enters the system using the user name and password.

12.1.3. Desktop Application- Home form

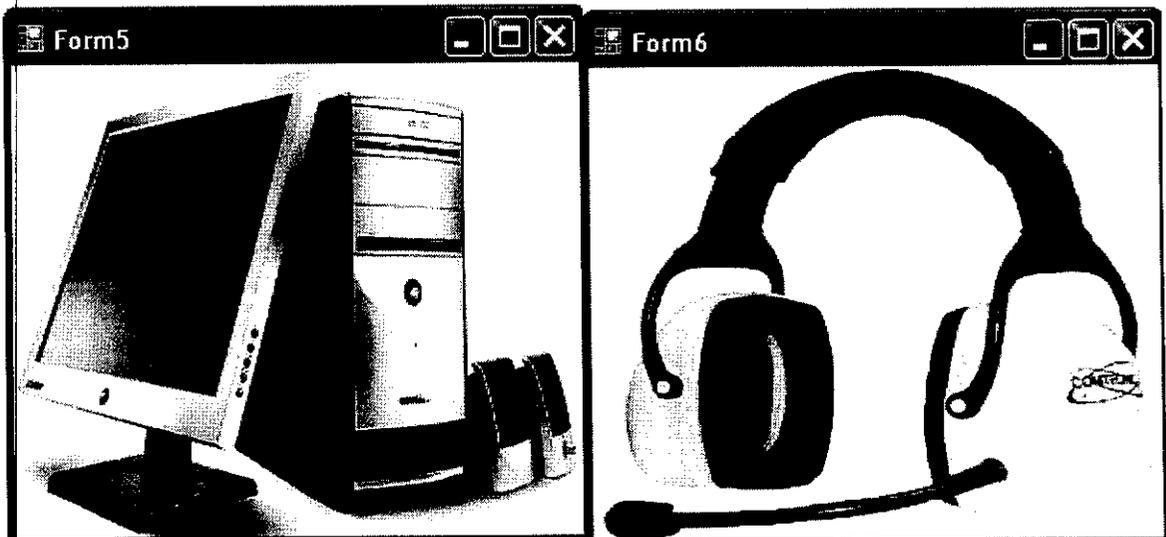


This figure shows the home form for the desktop application
**12.1.4. Desktop Application- The corresponding forms for the links In
home form**



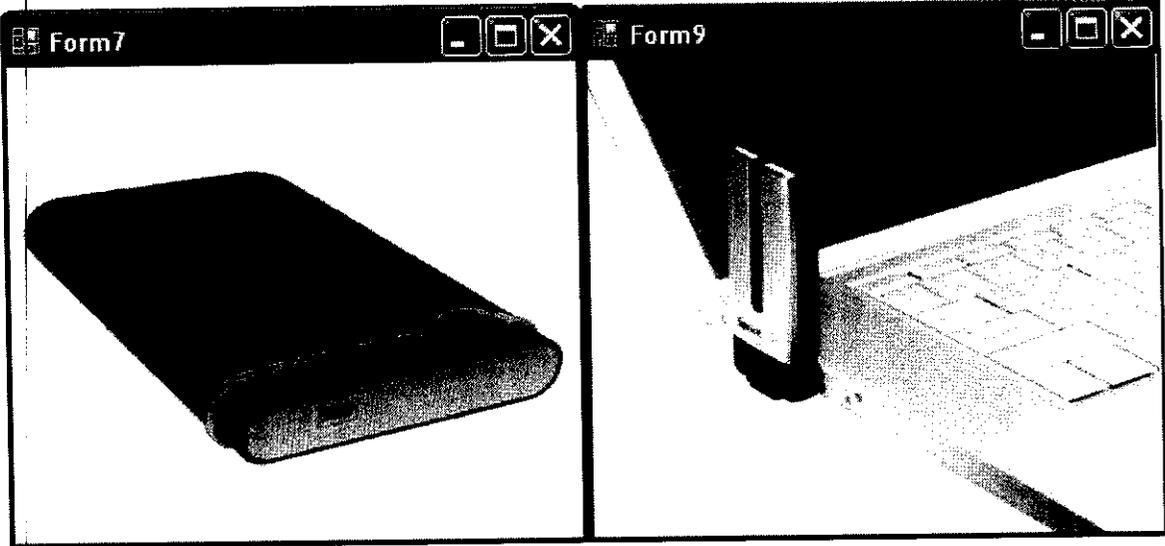
a. Laptop

b. Pen Drive



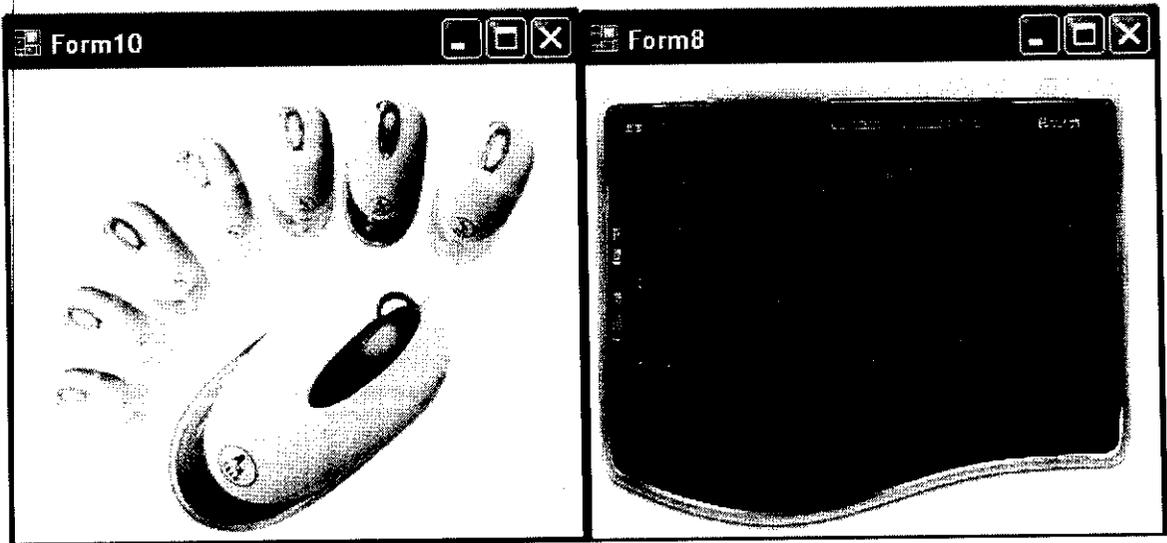
c. Desktop System

d. Headset



e. External Hard Disk

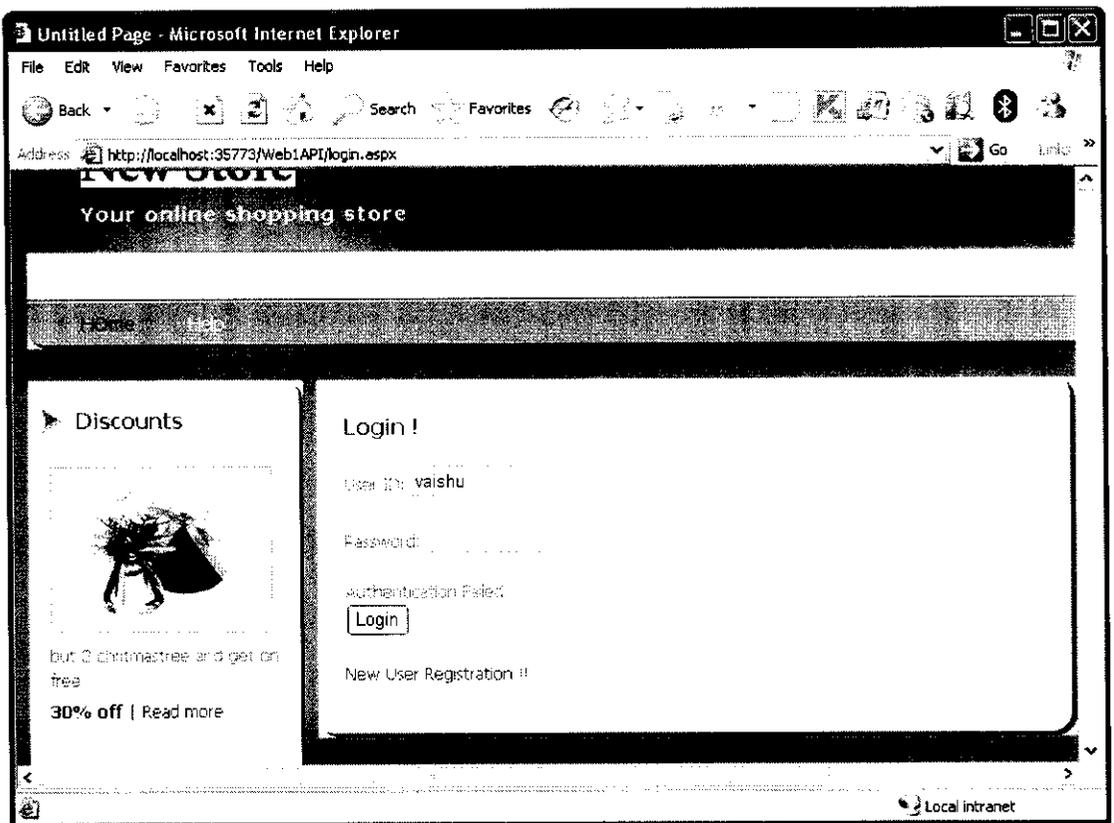
f. Modem



g. Mouse

h. Keyboard

12.1.5. Shopping website- Sign in page



This figure shows the sign in page for the shopping website where the user can enter the username and password to access the products.

12.1.6. Sopping website- New user registration form

Untitled Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Search Favorites

Address <http://localhost:35773/Web1API/Home.aspx> Go

Discounts



but 2 chitrnastree and get on free
30% off | Read more

New User Registration !

User ID:

Password:

Name:

Last Name:

Age:

Gender:
 Male
 Female

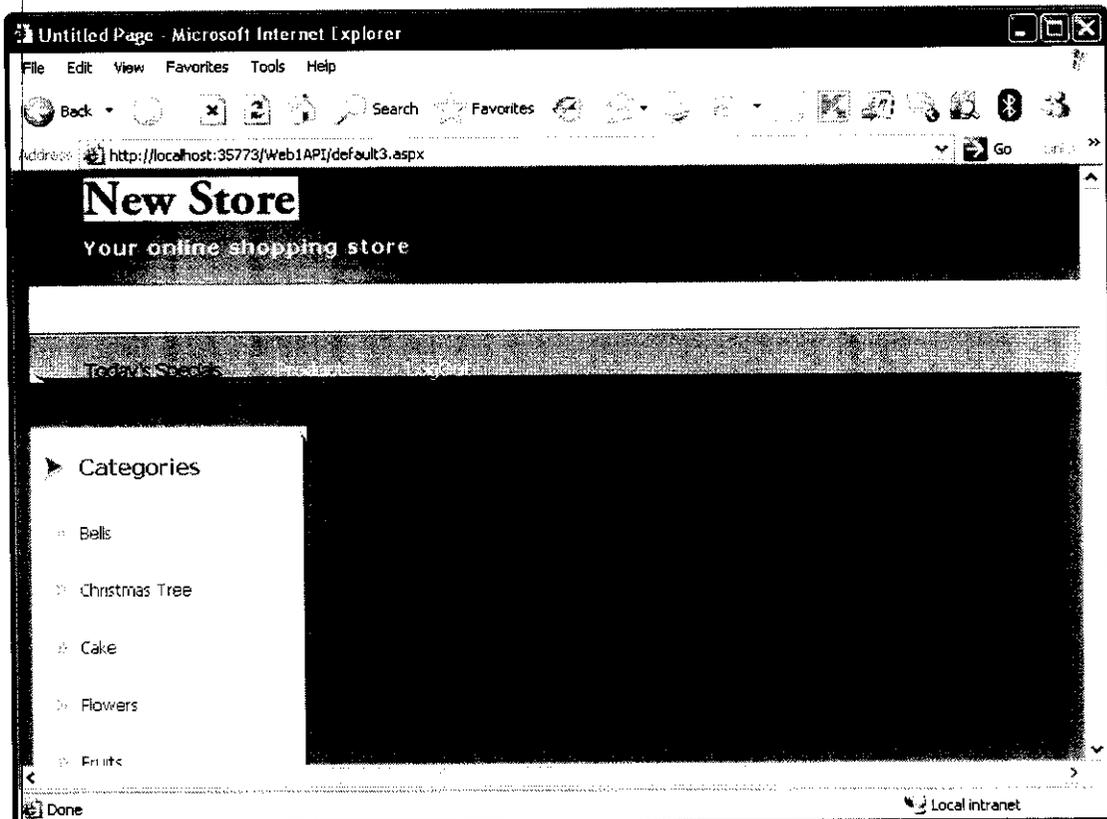
Email:

Country:

Done Local Intranet

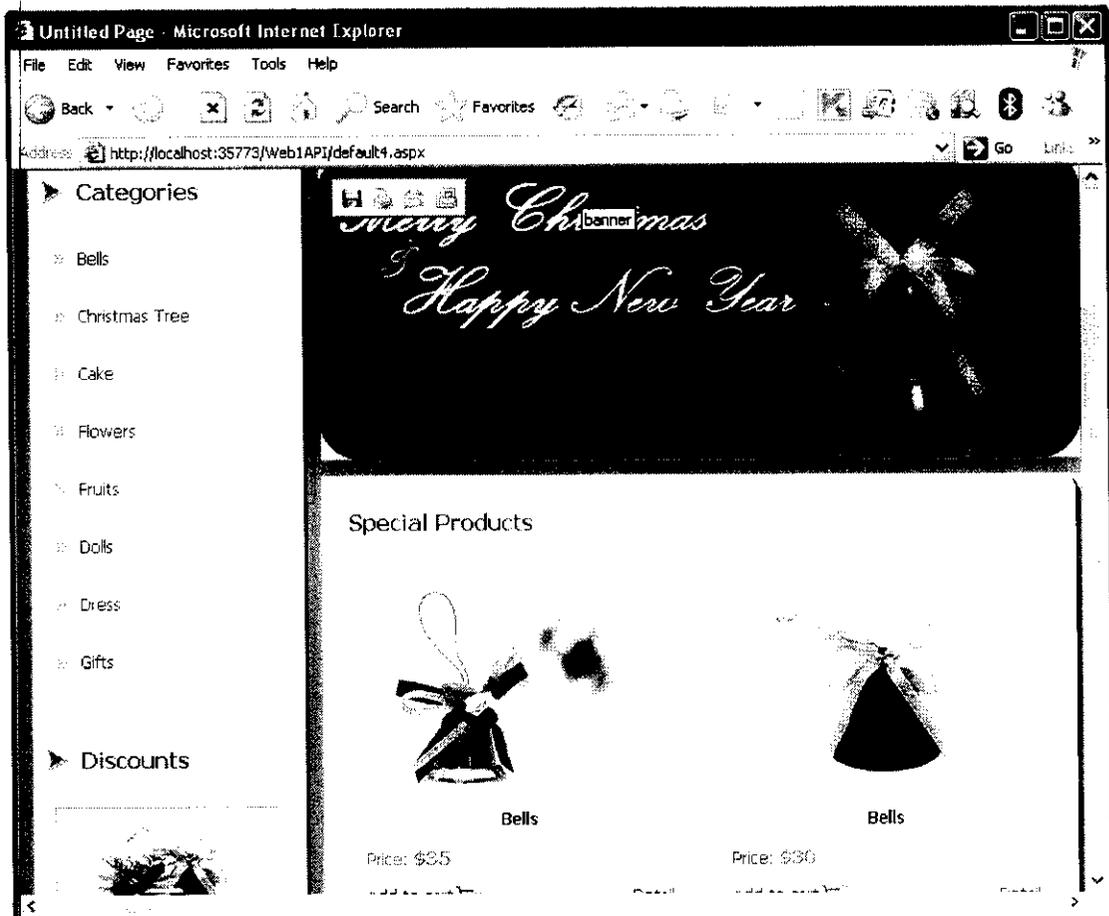
This figure shows the form that is used to sign up for the new users.

12.1.7. Shopping website- Home Page



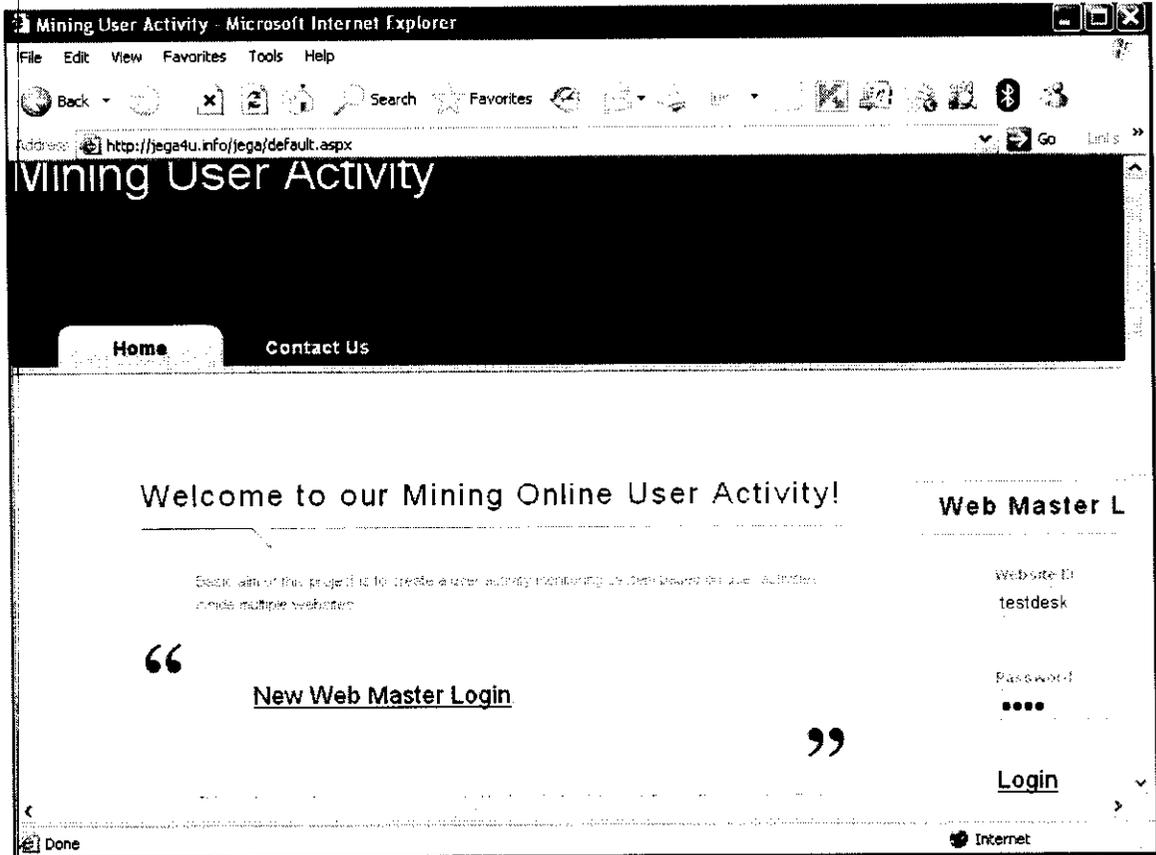
This figure shows the home page for the shopping website.

12.1.8. Shopping website- product page for bells



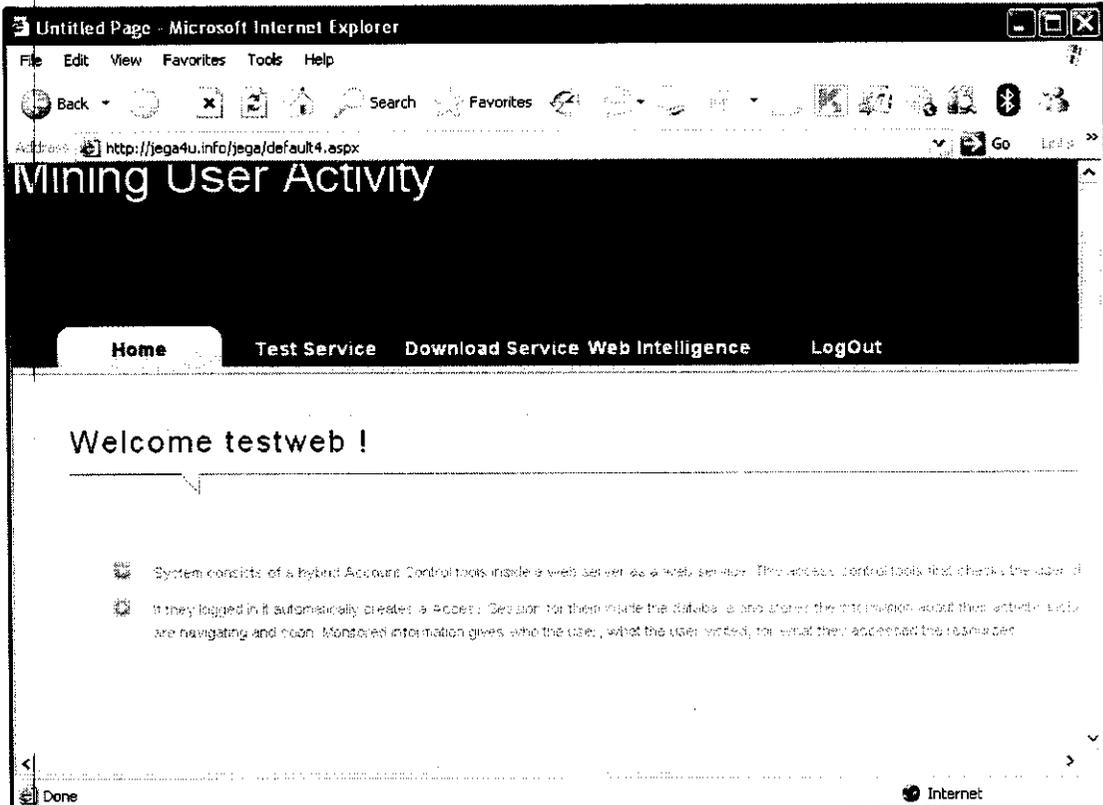
This figure shows the product page for the product bell.

12.1.9. Online website for mining user activity



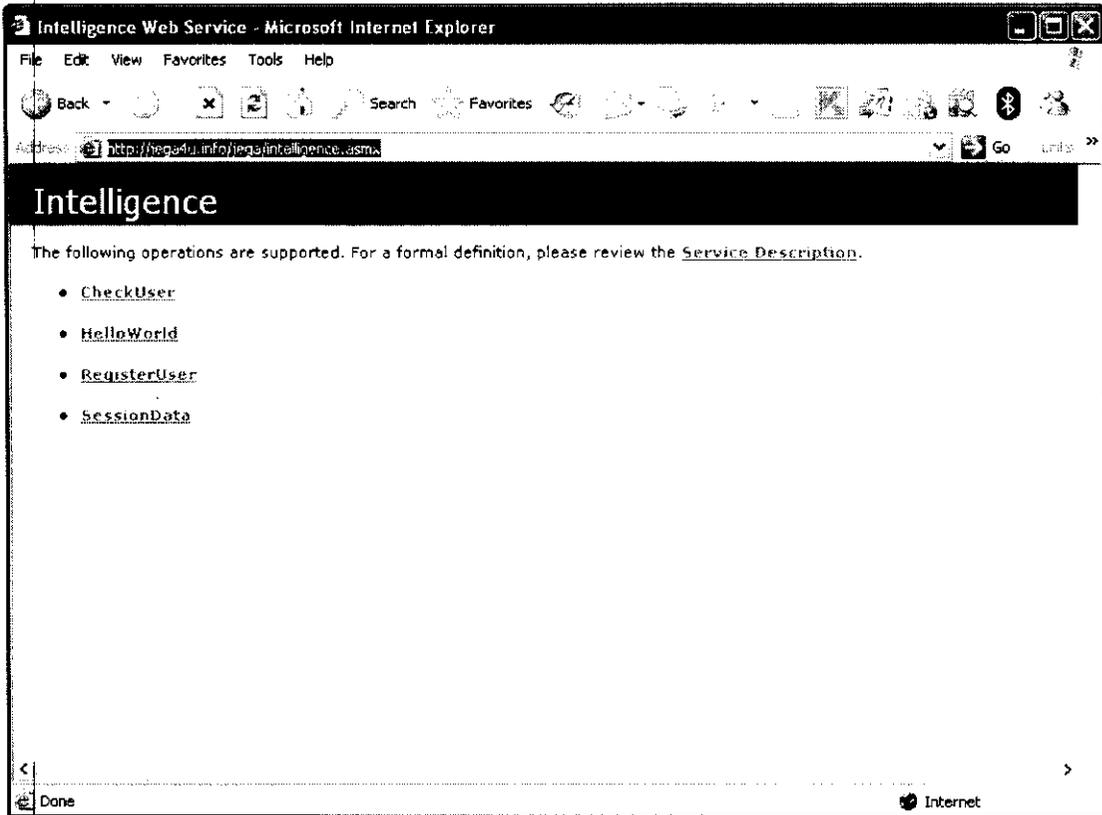
This figure shows the online existing website that deduces the user records from the website or the desktop application under this system and provides reports to the corresponding administrator.

12.1.10. Online website- Home page



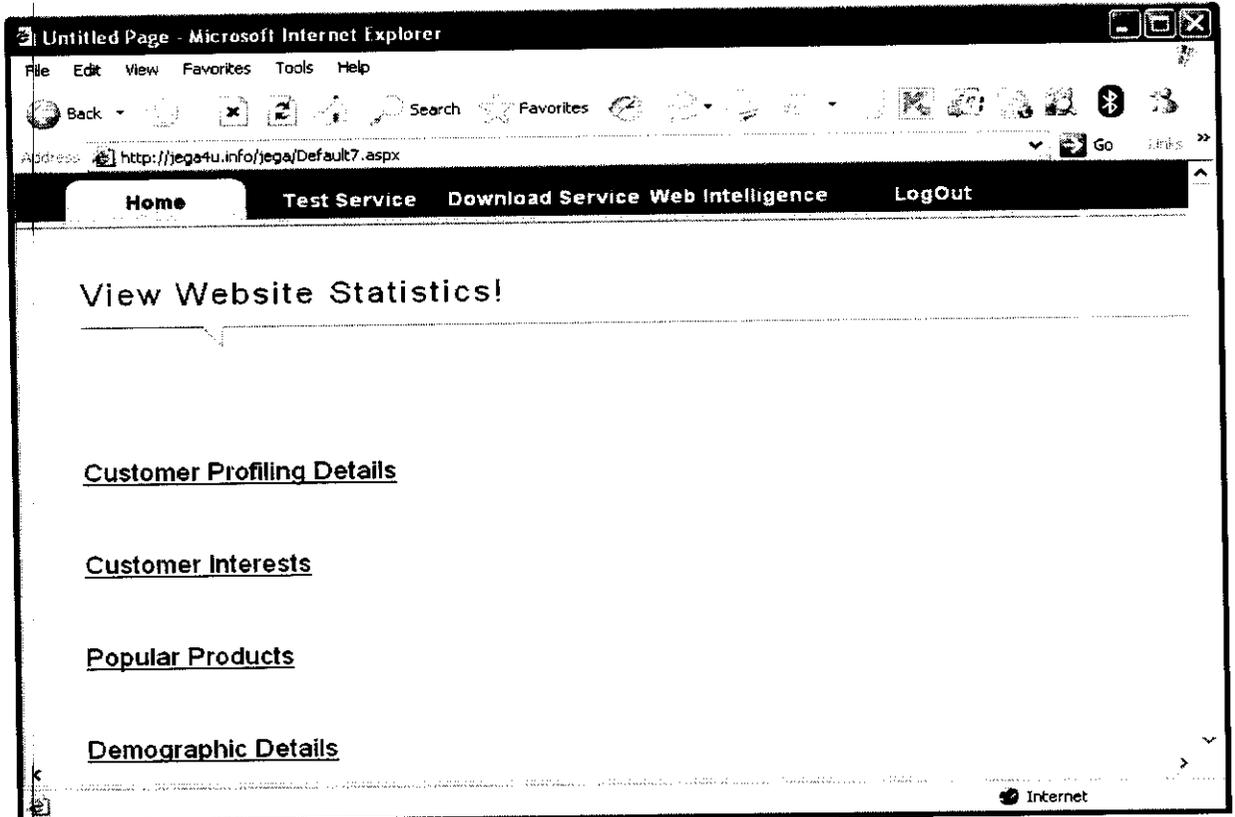
This figure shows the home page for the online website. The administrator of any website or application under this system can enter into this after login.

12.1.11. Test the service page



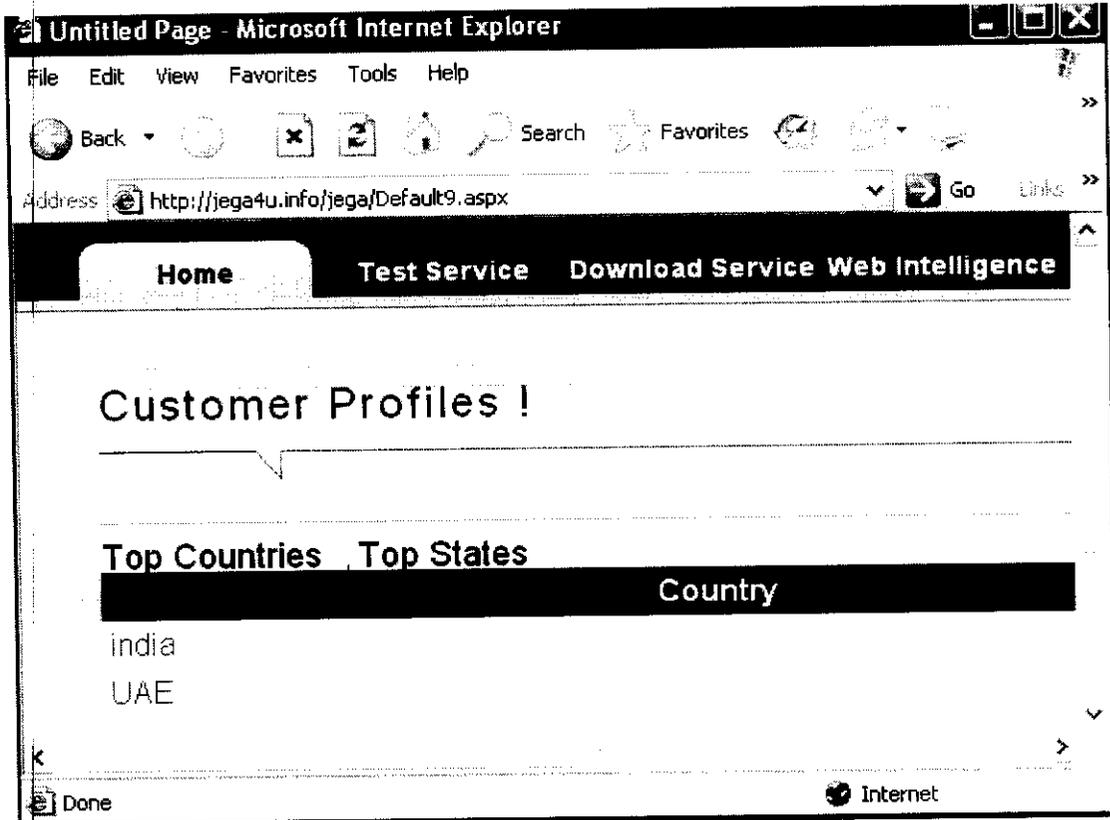
The service reference has to be added for this address to implement this tool in website

12.1.12. Web intelligence



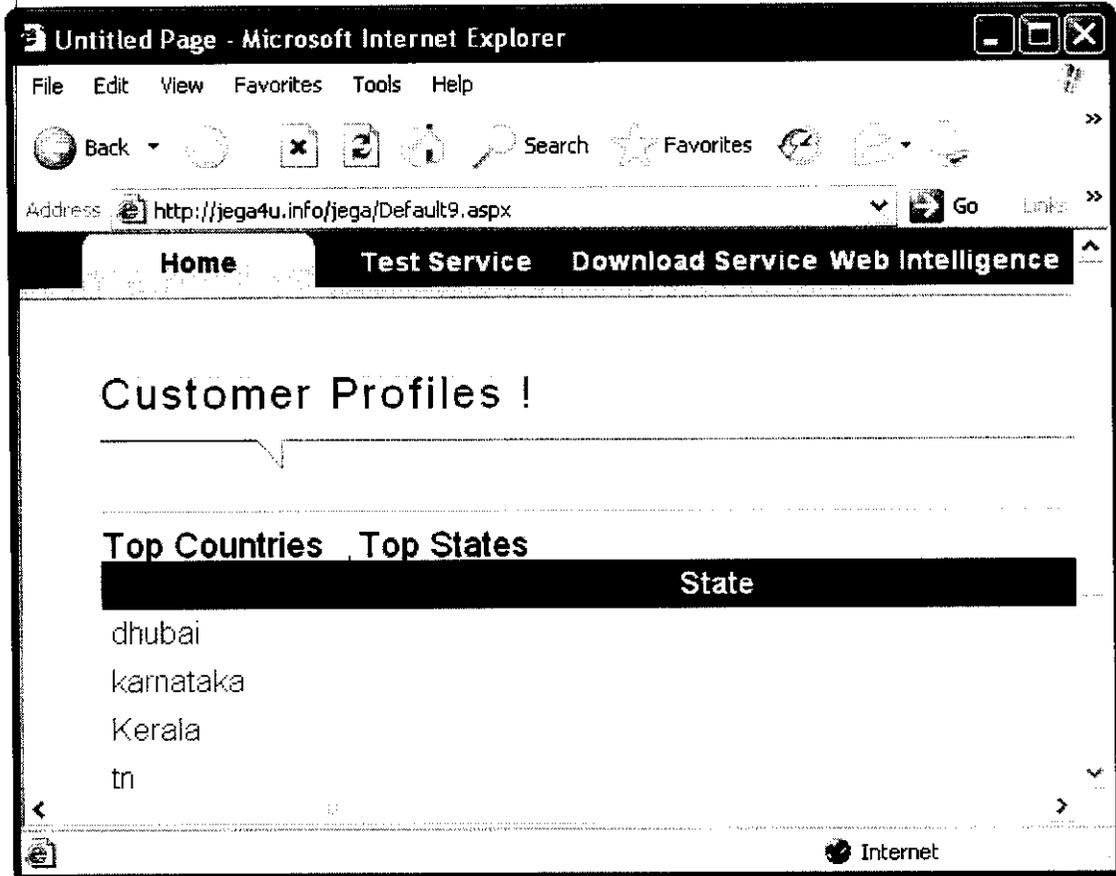
This page shows the available web intelligence details.

12.1.13. Top Countries



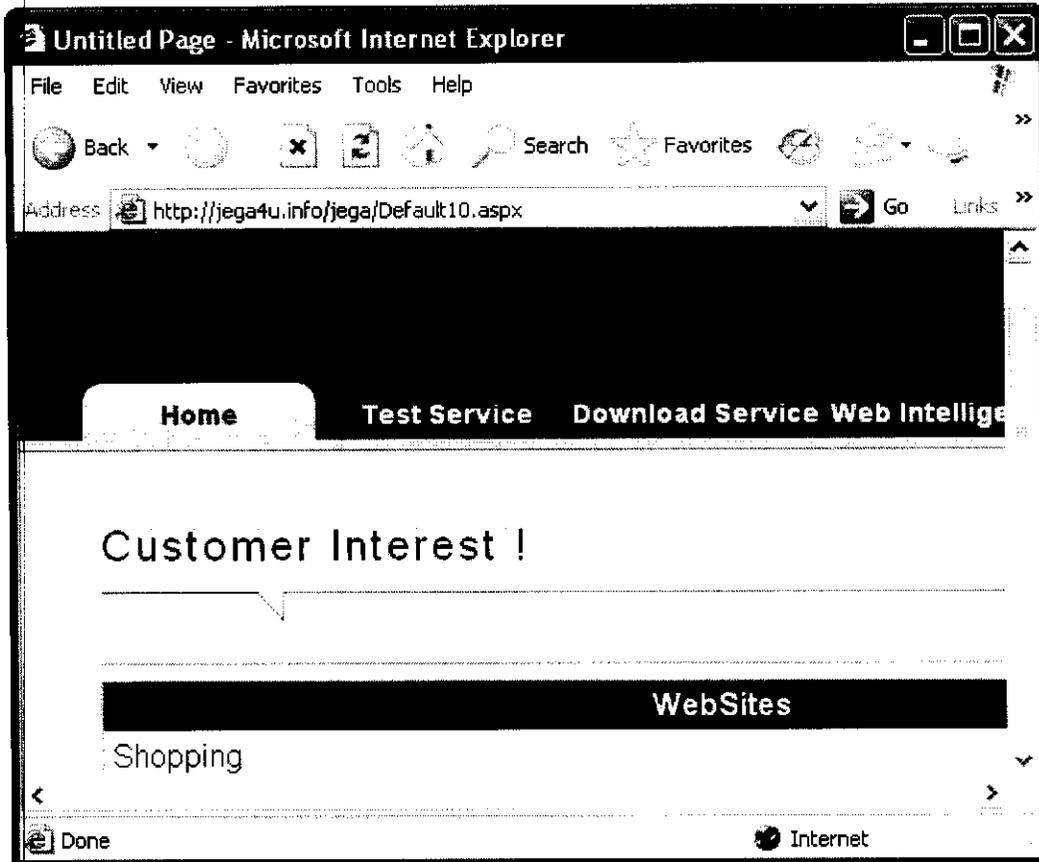
This figure shows top countries of the users.

12.1.14. Top States



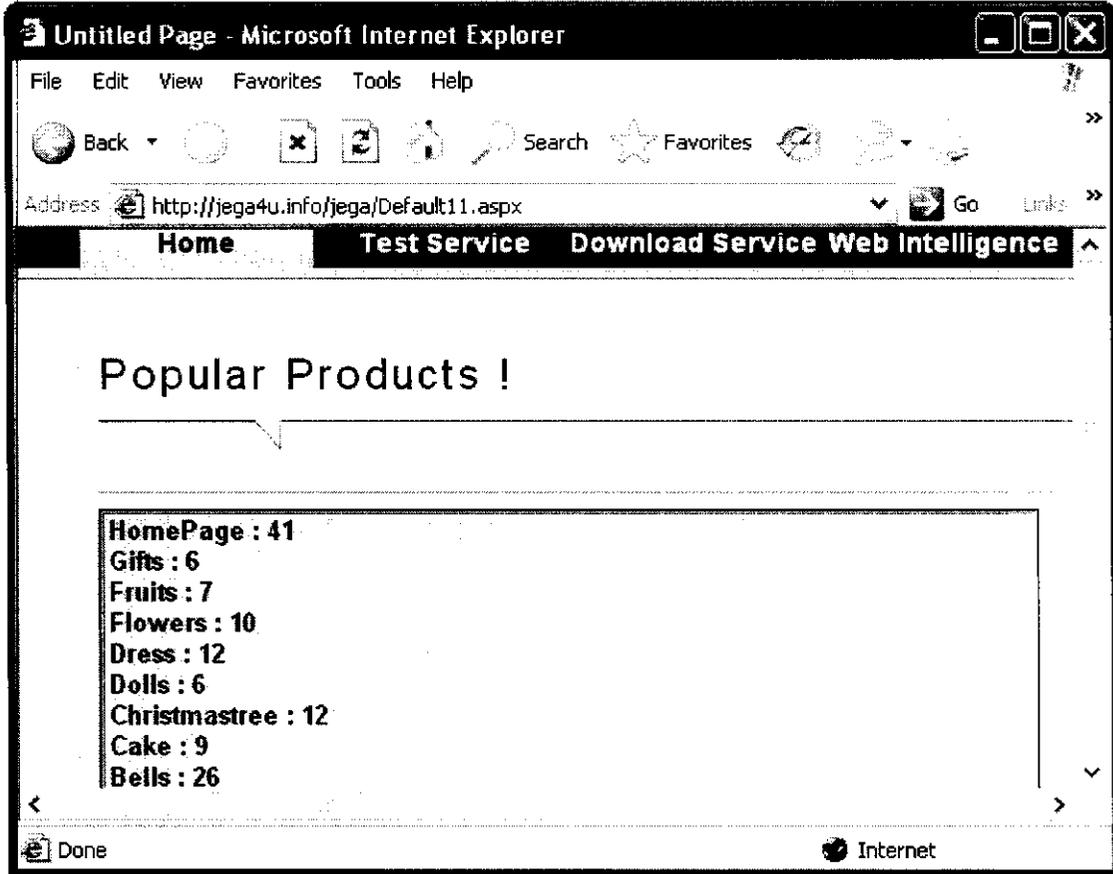
This figure shows the top states of the users.

12.1.15. Customer Interest



This figure shows the customer interest which application they prefer.

12.1.16. Popular products



This figure shows products their clicks or access.

12.1.17. Demographic and gender details

The screenshot shows a Microsoft Internet Explorer browser window displaying a table titled "Age Wise Details". The table contains 17 rows of user activity data. The columns are: SessionID, UserID, WebsiteID, LastLogin, VisitedArea, Country, State, WebType, Gender, and Age. The data shows users from India and UAE, with various activities like Login, Head Set, External Hard Disk, Modem, Mouse, and Keyboard usage.

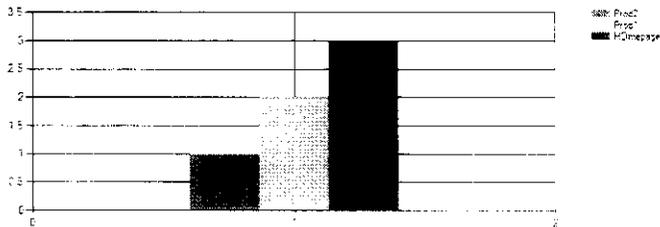
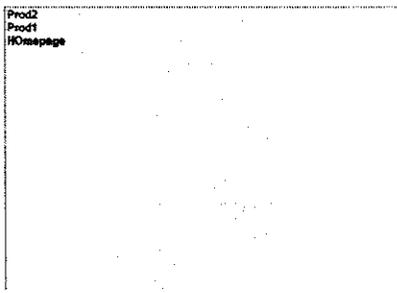
SessionID	UserID	WebsiteID	LastLogin	VisitedArea	Country	State	WebType	Gender	Age
testdesk97	vethasiromony	testdesk	4/2/2010	Login	india	tn	Desktop	male	5
testdesk98	vethasiromony	testdesk	4/2/2010	Head Set	india	tn	Desktop	male	5
testdesk156	vigin	testdesk	4/2/2010	Login	UAE	Dhubai	Desktop	male	2
testdesk157	vigin	testdesk	4/2/2010	External Hard Disk	UAE	Dhubai	Desktop	male	2
testdesk158	vigin	testdesk	4/2/2010	Modem	UAE	Dhubai	Desktop	male	2
testdesk145	pugali	testdesk	4/2/2010	Login	india	tn	Desktop	FeMale	2
testdesk146	pugali	testdesk	4/2/2010	Mouse	india	tn	Desktop	FeMale	2
testdesk147	pugali	testdesk	4/2/2010	Keyboard	india	tn	Desktop	FeMale	2
testdesk108	vigin	testdesk	4/2/2010	Login	UAE	Dhubai	Desktop	male	2
testdesk109	vigitha shajin	testdesk	4/2/2010	Login	UAE	dhubai	Desktop	female	2
testdesk110	vigitha shajin	testdesk	4/2/2010	Modem	UAE	dhubai	Desktop	female	2
testdesk148	chithra	testdesk	4/2/2010	Login	india	karnataka	Desktop	FeMale	2
testdesk149	chithra	testdesk	4/2/2010	Modem	india	karnataka	Desktop	FeMale	2
testdesk159	vigitha shajin	testdesk	4/2/2010	Login	UAE	dhubai	Desktop	female	2
testdesk160	vigitha shajin	testdesk	4/2/2010	Mouse	UAE	dhubai	Desktop	female	2
testdesk161	vigitha shajin	testdesk	4/2/2010	Keyboard	UAE	dhubai	Desktop	female	2

This figure shows the profile details, age, gender and demographic details along with their activities on website or desktop which comes under this system.

12.1.18. Chart-1

Home Test Service Download Service Web Intelligence LogOut

Popular Products !



Done

Local intranet | Protected Mode: Off

100%

This chart shows the comparison among the products.

12.1.19. Chart-2

Mining User Activity

Home

Test Service

Download Service Web Intelligence

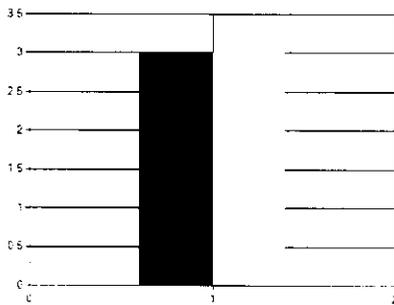
LogOut

Customer Age and Gender Details !

Gender Popularity :

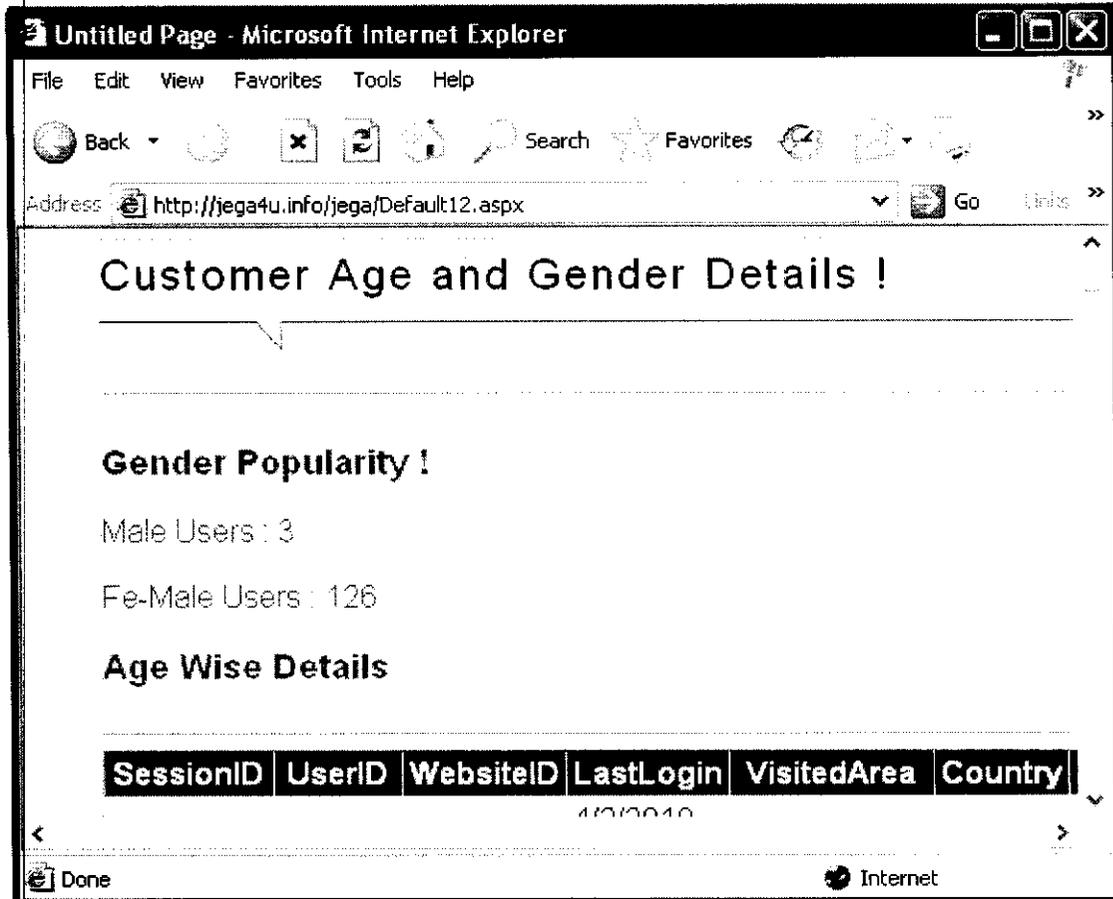
Male Users : 3

Age Wise Density :



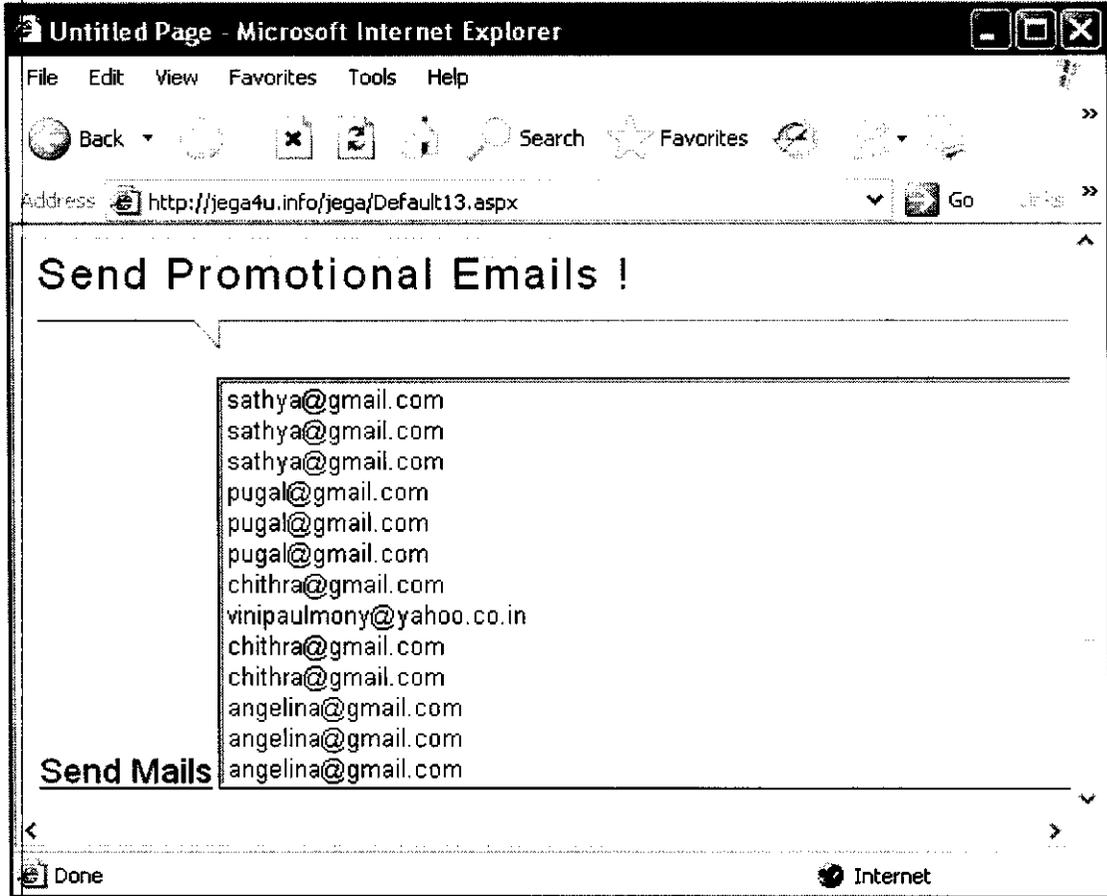
This chart shows the comparison between the male and female users.

12.1.20. Gender Popularity



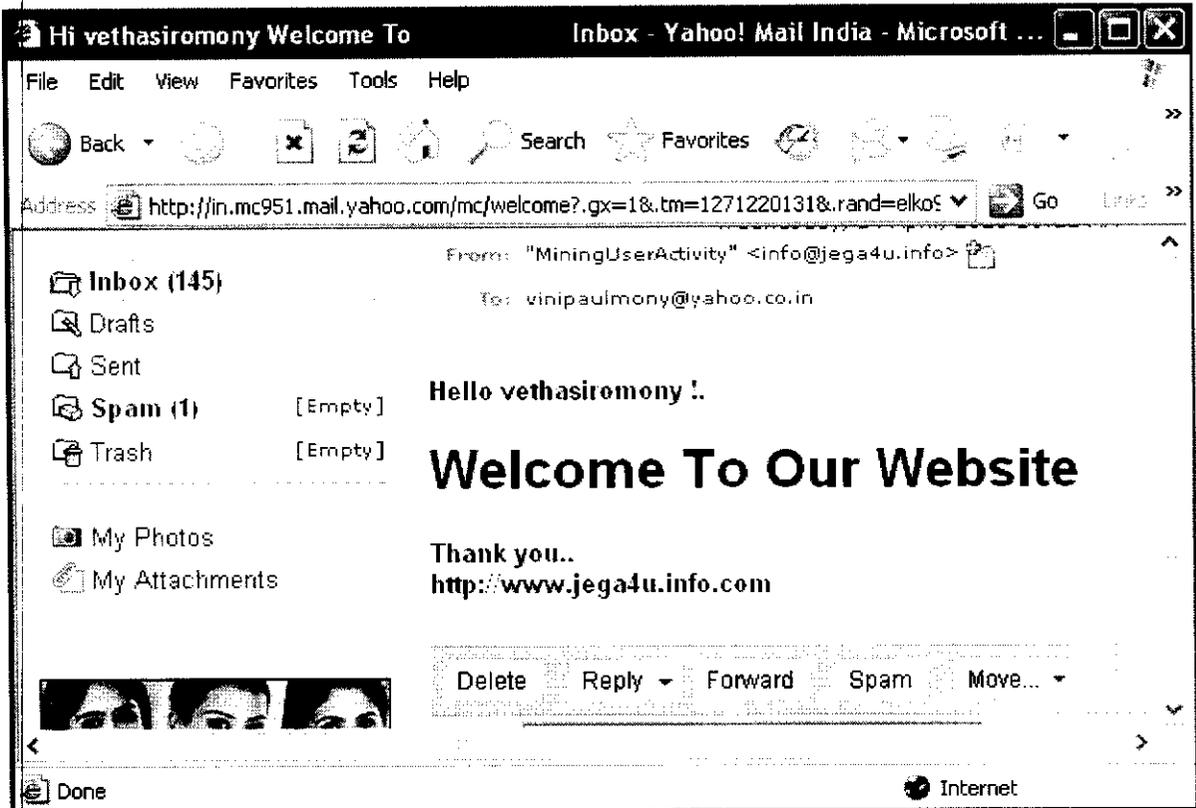
This figure shows the gender popularity among the users

12.1.21. Sending Promotional Emails



This figure shows the web page from which the email is sent by the admin.

12.1.22. Email from Admin



This figure shows the email received by the the user from the Admin.

APPENDIX-2

12.2. APPENDIX-2: SAMPLE CODE

Intelligence.cs of MiningUserLive

```
using System;
using System.Collections;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;

// <summary>
// Summary description for Intelligence
// </summary>
[WebService(Namespace = "http://jega4u.info/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
// To allow this Web Service to be called from script, using ASP.NET AJAX,
uncomment the following line.
// [System.Web.Script.Services.ScriptService]
public class Intelligence : System.Web.Services.WebService
{
    public Intelligence()
    {
        //Uncomment the following line if using designed components
        //InitializeComponent();
    }
    [WebMethod]
```

```

public string HelloWorld()
{
    return "Hello World";
}

[WebMethod]
public bool RegisterUser(string Userid, string name, string password, string
fullname, string email, string age, string gender, string country, string city, string
state, string address)
{
    try
    {
        GeneralUserDataContext Datas = new GeneralUserDataContext();
        GeneralUser data = new GeneralUser();
        data.Address = address;
        data.Age = age;
        data.City = city;
        data.Country = country;
        data.Email = email;
        data.FullName = fullname;
        data.Gender = gender;
        data.Name = name;
        data.Password = password;
        data.State = state;
        data.UserID = Userid;
        Datas.GeneralUsers.InsertOnSubmit(data);
        Datas.SubmitChanges();
    }
}

```

```

        return true;
    }
    catch (Exception ex)
    {
        return false;
    }
}

```

[WebMethod]

```

public bool SessionData(string userid, string websiteid, string logindate, string
visitedarea, string weftype)

```

```

{
    try
    {
        GeneralUserDataContext UData = new GeneralUserDataContext();
        var udata = from k in UData.GeneralUsers where k.UserID == userid
select k;
        if (udata.Count() >= 1)
        {
            foreach (var g in udata)
            {
                SessionDetailsDataContext Datas = new SessionDetailsDataContext();
                SessionDetailsDataContext SData = new
SessionDetailsDataContext();
                var sdat = from i in SData.WebSessions select i;
                int sid = sdat.Count() + 1;
                WebSession Data = new WebSession();

```

```

        Data.Age = g.Age;
        Data.Country = g.Country;
        Data.Gender = g.Gender;
        Data.LastLogin = logindate;
        Data.SessionID = websiteid + sid.ToString();
        Data.State = g.State;
        Data.UserID = userid;
        Data.VisitedArea = visitedarea;
        Data.WebsiteID = websiteid;
        Data.WebType = webtype;
        Datas.WebSessions.InsertOnSubmit(Data);
        Datas.SubmitChanges();
    }
}
return true;
}
catch (Exception ex)
{
    return false;
}
}
[WebMethod]
public string CheckUser(string userid, string pass)
{
    string returndata = "";
    GeneralUserDataContext UData = new GeneralUserDataContext();

```

```

var udata = from k in UData.GeneralUsers
            where k.UserID == userid && k.Password == pass
            select k;
if (udata.Count() >= 1)
{
    foreach (var t in udata)
    {
        returndata = t.UserID;
    }
}
else
{
    returndata = "false";
}
return returndata;
}
}

```

Default10.aspx.cs of MiningUserLive:

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;

```

```

using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
public partial class Default10 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        SessionDetailsDataContext Datas = new SessionDetailsDataContext();
        var data = from k in Datas.WebSessions
            where k.WebsiteID == Session["User"].ToString()
            select new { UserID = k.UserID, WebSites = k.WebType };
        GridView1.DataSource = data.ToList();
        GridView1.DataBind();
    }
}

```

Default11.aspx.cs of MiningUserLive:

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;

```

```

using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Web.UI.DataVisualization.Charting;
using System.Drawing;

public partial class Default11 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack)
        {
        }
        else
        {
            SessionDetailsDataContext Datas = new SessionDetailsDataContext ();
            var data = from k in Datas.WebSessions
                where k.WebsiteID == Session["User"].ToString()
                group k by k.VisitedArea into g
                orderby g.Key descending
                select new { prod = g.Key, Items = g };
            Legend D = new Legend("One");

            foreach (var t in data)

```

```

    {
        int i = 0;
        foreach (var d in t.Items)
        {
            i++;
        }
        ListBox1.Items.Add(t.prod);
        Series S2 = new Series();
        S2.Name = t.prod;
        DataPoint Dp1 = new DataPoint(1,i);
        S2.Points.Add(Dp1);
        S2.ChartArea = "ChartArea1";
        Chart1.Series.Add(S2);
    }
    Chart1.Legends.Add(D);
}
}
protected void ListBox1_SelectedIndexChanged(object sender, EventArgs e)
{
}
}

```

Default12.aspx.cs of MiningUserLive:

```

using System;
using System.Collections;
using System.Configuration;

```

```

using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Web.UI.DataVisualization.Charting;
using System.Drawing;

public partial class Default12 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        SessionDetailsDataContext Datas = new SessionDetailsDataContext();
        var mdata = from k in Datas.WebSessions
                    where k.WebsiteID == Session["User"].ToString() &&
                        k.Gender.ToLower() == "male"
                    select k;

        var fdata = from k in Datas.WebSessions
                    where k.WebsiteID == Session["User"].ToString() &&
                        k.Gender.ToLower() == "female"
                    select k;

        var data = from k in Datas.WebSessions

```

```

        where k.WebsiteID == Session["User"].ToString()
        select k;

double md = ((mdata.Count()));
double fd = ((fdata.Count()));
Label1.Text = "Male Users : " + md.ToString();
Label2.Text = "Fe-Male Users : " + fd.ToString();
Series S1 = new Series();
S1.Color = Color.Red;
S1.Name = "MaleUsers";
DataPoint Dp = new DataPoint(1, mdata.Count());
S1.Points.Add(Dp);
S1.ChartArea = "ChartArea1";
Series S2 = new Series();
S2.Color = Color.Yellow;
S2.Name = "FeMaleUsers";
DataPoint Dp1 = new DataPoint(1, fdata.Count());
S2.Points.Add(Dp1);
S2.ChartArea = "ChartArea1";
Chart1.Series.Add(S1);
Chart1.Series.Add(S2);
}
}

```

Default2.aspx.cs of MiningUserLive:

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
public partial class Default2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
        try
        {
            WebSiteDetailsDataContext Datas = new WebSiteDetailsDataContext();
            WebSite Data = new WebSite();
            Data.WebsiteID = TextBox1.Text;
            Data.Password = TextBox2.Text;
```

```

    Data.Name = TextBox3.Text;
    Data.WebType = TextBox4.Text;
    Data.Country = TextBox5.Text;
    Data.State = TextBox6.Text;
    Datas.WebSites.InsertOnSubmit(Data);
    Datas.SubmitChanges();
    Response.Redirect("default3.aspx");
}
catch (Exception ex)
{
}
}
}

```

Default7.aspx.cs of MiningUserLive:

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;

```

```
using System.Xml.Linq;
public partial class Default7 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
        Response.Redirect("Default9.aspx");
    }
    protected void LinkButton2_Click(object sender, EventArgs e)
    {
        Response.Redirect("Default10.aspx");
    }
    protected void LinkButton3_Click(object sender, EventArgs e)
    {
        Response.Redirect("Default11.aspx");
    }
    protected void LinkButton4_Click(object sender, EventArgs e)
    {
        Response.Redirect("Default12.aspx");
    }
}
```

Graphs.aspx.cs of MiningUserLive:

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Web.UI.DataVisualization.Charting;
using System.Drawing;

public partial class Graphs : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        SessionDetailsDataContext Datas = new SessionDetailsDataContext();
        var mdata = from k in Datas.WebSessions
                    where k.Gender.ToLower() == "male"
                    select k;
        var fdata = from k in Datas.WebSessions
```

```

        where k.WebsiteID == Session["User"].ToString() &&
k.Gender.ToLower() == "feMale"
        select k;
Series S1 = new Series();
S1.Color = Color.Red;
S1.Name = "MaleUsers";
DataPoint Dp = new DataPoint(1, mdata.Count());
S1.Points.Add(Dp);
S1.ChartArea = "ChartArea1";
Series S2 = new Series();
S2.Color = Color.Red;
S2.Name = "FeMaleUsers";
DataPoint Dp1 = new DataPoint(1, fdata.Count());
S2.Points.Add(Dp1);
S2.ChartArea = "ChartArea1";
Chart1.Series.Add(S1);
Chart1.Series.Add(S2);
}
}

```

Default9.aspx.cs of MiningUserLive:

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;

```

```

using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
public partial class Default9 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
        SessionDetailsDataContext Datas = new SessionDetailsDataContext();
        var data = from k in Datas.WebSessions
            where k.WebsiteID == Session["User"].ToString()
            select new { UserID = k.UserID, Country = k.Country };
        GridView1.DataSource = data.ToList();
        GridView1.DataBind();
    }
    protected void LinkButton2_Click(object sender, EventArgs e)
    {
        SessionDetailsDataContext Datas = new SessionDetailsDataContext();
        var data = from k in Datas.WebSessions
            where k.WebsiteID == Session["User"].ToString()

```

```
        select new { UserID = k.UserID, State = k.State };
```

```
        GridView1.DataSource = data.ToList();
```

```
        GridView1.DataBind();
```

```
    }
```

```
}
```

Home.aspx.cs of Web1API:

```
using System;
```

```
using System.Collections;
```

```
using System.Configuration;
```

```
using System.Data;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.Security;
```

```
using System.Web.UI;
```

```
using System.Web.UI.HtmlControls;
```

```
using System.Web.UI.WebControls;
```

```
using System.Web.UI.WebControls.WebParts;
```

```
using System.Xml.Linq;
```

```
namespace Web1API
```

```
{
```

```
    public partial class Home : System.Web.UI.Page
```

```
    {
```

```
        protected void Page_Load(object sender, EventArgs e)
```

```
        {
```

```
            if (Request.QueryString.Count > 0)
```

```
            {
```

```
        Label1.Text = Request.QueryString[0].ToString();
```

```
    }
```

```
    }
```

```
    }
```

```
}
```

Login.aspx.cs of Web1API:

```
using System;
```

```
using System.Collections;
```

```
using System.Configuration;
```

```
using System.Data;
```

```
using System.Linq;
```

```
using System.Web;
```

```
using System.Web.Security;
```

```
using System.Web.UI;
```

```
using System.Web.UI.HtmlControls;
```

```
using System.Web.UI.WebControls;
```

```
using System.Web.UI.WebControls.WebParts;
```

```
using System.Xml.Linq;
```

```
namespace Web1API
```

```
{
```

```
    public partial class Login : System.Web.UI.Page
```

```
    {
```

```
        protected void Page_Load(object sender, EventArgs e)
```

```
        {
```

```
        }
```

```
        protected void Button1_Click(object sender, EventArgs e)
```

```

{
    PassRef.WebService D = new Web1API.PassRef.WebService();
    if (D.GeneralLoginCredentialCheck(TextBox1.Text, TextBox2.Text))
    {
        D.CreateSession(TextBox1.Text, "NewStore",
DateTime.Now.ToString(), "1", "Shop", "Shop", "Shopping");
        Response.Redirect("Home.aspx?usr=" + TextBox1.Text + "");
    }
    else
    {
        Response.Write("Check Entries");
    }
}
}
}

```

Shopping.master.cs of Web1API:

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;

```

```

using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Xml.Linq;
public partial class Shopping : System.Web.UI.MasterPage
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
        MiningReference.Intelligence F1 = new MiningReference.Intelligence();
        F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Bells", "Shopping");
        Response.Redirect("default4.aspx");
    }
    protected void LinkButton2_Click(object sender, EventArgs e)
    {
        MiningReference.Intelligence F1 = new MiningReference.Intelligence();
        F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Christmastree", "Shopping");
        Response.Redirect("default5.aspx");
    }
    protected void LinkButton3_Click(object sender, EventArgs e)
    {
        MiningReference.Intelligence F1 = new MiningReference.Intelligence();

```

```

        F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Cake", "Shopping");
        Response.Redirect("default6.aspx");
    }
protected void LinkButton4_Click(object sender, EventArgs e)
{
    MiningReference.Intelligence F1 = new MiningReference.Intelligence();
    F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Flowers", "Shopping");
    Response.Redirect("default7.aspx");
}
protected void LinkButton5_Click(object sender, EventArgs e)
{
    MiningReference.Intelligence F1 = new MiningReference.Intelligence();
    F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Fruits", "Shopping");
    Response.Redirect("default8.aspx");
}
protected void LinkButton6_Click(object sender, EventArgs e)
{
    MiningReference.Intelligence F1 = new MiningReference.Intelligence();
    F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Dolls", "Shopping");
    Response.Redirect("default9.aspx");
}
protected void LinkButton7_Click(object sender, EventArgs e)

```

```

{
    MiningReference.Intelligence F1 = new MiningReference.Intelligence();
    F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Dress", "Shopping");
    Response.Redirect("default10.aspx");
}
protected void LinkButton8_Click(object sender, EventArgs e)
{
    MiningReference.Intelligence F1 = new MiningReference.Intelligence();
    F1.SessionData(Session["User"].ToString(), "testweb",
DateTime.Now.ToString(), "Gifts", "Shopping");
    Response.Redirect("default11.aspx");
}
protected void LinkButton9_Click(object sender, EventArgs e)
{
    Session.Abandon();
    Response.Redirect("login.aspx");
}
}

```

APPENDIX-3

12.3. APPENDIX-3: TABLE DESIGN

12.3.1. General Users

UserID	varchar(50)
Name	varchar(50)
Password	varchar(50)
FullName	varchar(50)
Email	varchar(50)
Age	varchar(50)
Gender	varchar(50)
Country	varchar(50)
City	varchar(50)
State	varchar(50)
Address	varchar(150)

12.3.2. Web sessions

SessionID	varchar(50)
UserID	varchar(50)
WebsiteID	varchar(50)
LastLogin	varchar(50)

VisitedArea	varchar(50)
Country	varchar(50)
State	varchar(50)
Webtype	varchar(50)
Gender	varchar(50)
Age	varchar(50)

12.3.3. WebSites

WebsiteID	varchar(50)
Password	varchar(50)
Name	varchar(50)
WebType	varchar(50)
Country	varchar(50)
State	varchar(50)

REFERENCES

13. REFERENCE

- [1] Simon Fong and Serena Chan (2002), "Mining Online Users' Access Records for Web Business Intelligence" in Proceedings, IEEE International Conference ICDM 2002.
- [2] S.Fong, A..Sun, K.Wong, "Price Watcher Agent for E-Commerce", Int, Conference on Intelligent Agent Technology, World Scientific, pp.294-300,2001.
- [3] Liya Wu, Gilad Barash, Claudio Bartolini (2007), "A Service Oriented Architecture for Business Intelligence", in Proc IEEE International Conference on Service Oriented Computing and Applications.
- [4] Xu Xi and Xu Hongfeng (2009), Developing a framework for business intelligence systems integration based on ontology,IEEE International Conference on Networking and Digital Society.
- [5] Prieto-Diaz, R.; A faceted approach to building ontologies , Information Reuse and Integration, 2003.IRI 2003. IEEE International Conference on, 27-29 Oct. 2003, 458 – 465.
- [6] GU Xin-jian and QI Guo-ning, (1996) (1997) Global integrated manufacturing and Great Group Engineering(1)(2) .In Factory construction and design, No.6 pp.28–32, No.1 pp.30–32.
- [7] Liu, D., Wang, ZH., Xie, QS. and Gu, XJ. (2003) Internet-based great group technology. In Third international conference on electronic commerce engineering-digital enterprises and nontraditional industrialization. pp.423–425.

- [8] D. C. Hay. Data Model Patterns - Conventions of Thought Dorset House Publishing, 1996.
- [9] Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation 27 March 2000, <http://www.w3c.org/TR>.
- [10] Gao ji. Software intelligence technology based on knowledge Hangzhou: Zhejiang university publisher, 2000.
- [11] P.Chan, "A Non-invasive Learning Approach to Building Web User Profiles", KDD-99 Workshop on Web Usage Analyses and User Profiling, pp.7-12,1999.