



**Jaypee University of Information Technology
Solan (H.P.)**

LEARNING RESOURCE CENTER

Acc. Num. **SP04051** Call Num:

General Guidelines:

- ◆ Library books should be used with great care.
- ◆ Tearing, folding, cutting of library books or making any marks on them is not permitted and shall lead to disciplinary action.
- ◆ Any defect noticed at the time of borrowing books must be brought to the library staff immediately. Otherwise the borrower may be required to replace the book by a new copy.
- ◆ The loss of LRC book(s) must be immediately brought to the notice of the Librarian in writing.

Learning Resource Centre-JUIT



SP04051

INTELLIGENT MULTIMEDIA ADVERTISEMENT PLATFORM FOR WEB BASED NETWORKS

By

ANKIT BHARDWAJ

(041245)

LAKSHYA SHRIVASTAVA

(041247)

SHOBHIT MAHAJAN

(041255)

SUMIT SACHDEVA

(041243)



MAY-2008

**Submitted in partial fulfillment of the Degree of Bachelor of
Technology**

Project Guide: Mr. Vipin Arora

**DEPARTMENT OF COMPUTER SCIENCE
JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
WAKNAGHAT**

CERTIFICATE

This is to certify that the work entitled, "Intelligent Multimedia Advertisement platform for web based networks " submitted by Ankit Bhardwaj, Lakshya Shrivastava, Shobhit Mahajan and Sumit Sachdeva for the award of degree of Bachelor of Technology in Computer Science of "Jaypee University of Information Technology" has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

First and foremost, we thank our Guide, Mr. Vipin Arora, who in spite of his various engagements, always found time for us, and without whose assistance, the entire project would never have borne fruits. Actually, rank will not reflect on the amount of help that we have got from our Guide.

We also would like to thank Mrs. Sunita Chandra, who rendered us time to time during the

Mr. Vipin Arora
(Lecturer)

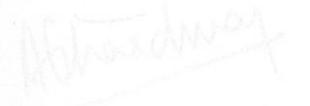
Department Of Computer Science

Dr. S.P. Gharera
(Head of Department)

Department Of Computer Science

valuable suggestions from time to time.

We thank the Department of Computer Science, JUIT, for providing us the necessary facilities to work on our project.


ANKIT BHARDWAJ
041331


LAKSHYA SHRIVASTAVA
041331


SHOBHIT MAHAJAN
041331


SUMIT SACHDEVA
041331

ACKNOWLEDGMENT

Though this project batch consists of only four members, it would be very wrong on our part to call this project a result of the efforts of just four individuals. In fact, we would like to call this a collective effort, as we had received assistance from various quarters during the development of our project. We take this opportunity to thank each and everyone who have helped us.

First and foremost, we thank our Guide, Mr. Vipin Arora, who in spite of his various engagements, always found time for us, and without whose assistance, the entire project would never have borne fruits. Actually, words will not reflect on the amount of help that we have got from our Guide.

We also would like to thank Mr. Satish Chandra, who assisted us time to time during the progress of our work..

I am also thankful to Mr. Rohit Sharma, web maintainer (JUIT) for providing us with valuable suggestions from time to time

We thank the Department of Computer Science, JUIT, for providing us the necessary facilities to work on our project.



(ANKIT BHARDWAJ)

(041245)



(LAKSHYA SHRIVASTAVA)

(041247)



(SHOBHIT MAHAJAN)

(041255)



(SUMIT SACHDEVA)

(041243)

TABLE OF CONTENTS

Chapter No.	Page No.
1. List of Figures	1
2. Abstract	2
3. Introduction	
3.1 . Introduction	3
3.2. Types of Advertising Modes	6
3.3. Advertisement Revenue Models	9
4. Problem Statement	
4.1 . Problem Statement	11
5. Approach	
5.1. Approach	13
6. Technology Used	
6.1. Technology Used	14
7. Design Specification	
1.1. ERD, DFDS, GANTT CHARTS	15
8. Modular Decomposition of the project	
8.1. Advertiser interface Module	20
8.2. User Interface Module	20
8.3. Search And Matching Module	20
9. Software Requirements	21
10. Methodology for advertisement Matching	22
11. Response System	26
11.1. Procedure for long term Interests	27
11.2. Procedure for short term Interests	29
12. Snap Shots and GUI Details	31
13. Conclusions and Future Work	91
14. References	92
15. Bibliography	93

LIST OF FIGURES

I. Fig1 Web Application Model	14
II. Fig2 ER Diagram	15
III. Fig 3 LEVEL 0-DFD of the Project	16
IV. Fig 4 LEVEL 1-DFD of the Project	17
V. Fig 5 Gantt Chart	18
VI. Fig6 Gantt Chart	19
VII. Methodology Diagram	
Fig 7 Block Diagram	27
Fig 8 Cubical View	28
VIII. Fig 9 Response System Diagram	26

ABSTRACT

This project aims to make an intelligent advertisement platform which will run on web based networks. Our system can be used by multimedia content distributors on internet for procuring advertisements particular to a user on their group. Furthermore we are providing advertisers a platform for promotions of their brands by means of broadcast and targeted advertising i.e. their advertisements will be distributed to only those users who could be their potential customers and that appeal to their area of interest.

networks and e-mail marketing, including e-mail spam.

A major result of online advertising is information and content that is not limited by geography or time. The emerging uses of interactive advertising presents fresh challenges for advertisers who have hitherto adopted an interruptive strategy.

Online video directories for brands are a good example of interactive advertising. These directories complement television advertising and allow the viewer to view the commercials of a number of brands. If the advertiser has opted for a response feature, the viewer may then choose to visit the brand's website, or interact with the advertiser through other touch points such as email, chat or phone. Response to brand communication is instantaneous, and conversion to business is very high. This is because in contrast to interruptive forms of interactive advertising, the viewer has actually chosen to see the commercial.

INTRODUCTION

An advertising network, or ad network is a company that connects web sites that want to run advertisements, advertising banners who want to run advertisements. Increasingly Ad networks are companies that pay software developers as well as take their money for

Online advertising is a form of advertising that uses the Internet and World Wide Web in order to deliver marketing messages and attract customers. Examples of online advertising include contextual ads on search engine results pages, banner ads, advertising networks and e-mail marketing, including e-mail spam.

A major result of online advertising is information and content that is not limited by geography or time. The emerging area of interactive advertising presents fresh challenges for advertisers who have hitherto adopted an interruptive strategy.

Online video directories for brands are a good example of interactive advertising. These directories complement television advertising and allow the viewer to view the commercials of a number of brands. If the advertiser has opted for a response feature, the viewer may then choose to visit the brand's website, or interact with the advertiser through other touch points such as email, chat or phone. Response to brand communication is instantaneous, and conversion to business is very high. This is because in contrast to conventional forms of interruptive advertising, the viewer has actually chosen to see the commercial.

represents the advertising banner.

Large publishers often sell only their remnant inventory through ad networks. Typical numbers range from 10% to 60% of total inventory being remnant and sold through advertising networks.

Smaller publishers often sell all of their inventory through ad networks. One type of ad network, known as a blind network, is such that advertisers place ads, but do not know the exact places where their ads are being placed.

ADVERTISING NETWORK

An advertising network or ad network is a company that connects web sites that want to host advertisements with advertisers who want to run advertisements. Increasingly Ad networks are companies that pay software developers as well as web sites money for allowing their ads to be shown when people use their software or visit their sites.

Ad networks serve advertising on your website and share advertiser revenue for qualified clicks each time your site's visitors click on ads. An advertising network (also called an online advertising network or ad network) is a collection of (often unrelated) online advertising inventory. When it is clear that the environment involved is the Internet, companies who run or administrate such networks are also called Advertising Agents or simply Agents.

Online advertising inventory comes in many different forms. This inventory can be found on websites, in RSS feeds, on blogs, in instant messaging applications, in ad ware, in e-mails, and on other sources. Some examples of advertising inventory include: banner ads, rich media, text links, and e-mails. (This is not an exhaustive list.)

3) Targeted Networks: Sometimes called "next generation" or "2.0" ad networks, these An advertiser can buy a run of network package, or a run of category package within the network. The advertising network serves advertisements from its ad server, which responds to a site once a page is called. A snippet of code is called from the ad server that represents the advertising banner.

Large publishers often sell only their remnant inventory through ad networks. Typical numbers range from 10% to 60% of total inventory being remnant and sold through advertising networks.

Smaller publishers often sell all of their inventory through ad networks. One type of ad network, known as a blind network, is such that advertisers place ads, but do not know the exact places where their ads are being placed.

In most cases, ad networks deliver their content through the use of a central ad server. Large ad networks include a mixture of search engines, media companies, and technology vendors.

There are three types of online advertising networks: 1) Representative Networks, 2) Blind Networks and 3) Targeted Networks.

1) Representative (or Rep) Networks: They represent the publications in their portfolio, with full transparency for the advertiser about where their ads will run. They typically promote high quality traffic at market prices and are heavily used by brand marketers. The economic model is generally revenue share.

2) Blind Networks: These companies offer low pricing to direct marketers in exchange for those marketers relinquishing control over where their ads will run. Blind networks achieve their low pricing through large bulk buys of typically remnant inventory combined with campaign optimization and ad targeting technology. The financial model is arbitrage.

3) Targeted Networks: Sometimes called “next generation” or “2.0” ad networks, these focus on specific targeting technologies such as behavioral or contextual. Targeted networks specialize in using consumer click stream data to enhance the value of the inventory they purchase.

There are two types of advertising networks: first-tier and second-tier networks. First-tier advertising networks have a large number of their own advertisers and publishers, they have high quality traffic, and they serve ads and traffic to second-tier networks. Examples of first-tier networks include the major search engines. Second-tier advertising networks may have some of their own advertisers and publishers, but their main source of revenue comes from syndicating ads from other advertising networks.

TYPES OF ADVERTISING MODES

CONTEXTUAL ADVERTISING

Many advertising networks display graphical or text-only ads that correspond to the keywords of an Internet search or to the content of the page on which the ad is shown. These ads are believed to have a greater chance of attracting a user, because they tend to share a similar context as the user's search query. For example, a search query for "flowers" might return an advertisement for a florist's website.

Another newer technique is embedding keyword hyperlinks in an article which are sponsored by an advertiser. When a user follows the link, they are sent to a sponsor's website

AFFILIATE MARKETING

Affiliate marketing is a form of online advertising where advertisers place campaigns with a potentially large number of small (and large) publishers, whom are only paid media fees when traffic to the advertiser is garnered, and usually upon a specific measurable campaign result (a form, a sale, a sign-up, etc). Today, this is usually accomplished through contracting with an affiliate network or CPA network, such as Performics, Hydra Network, Motive Interactive, Commission Junction/Be Free, Link Share, Primeq, Consort Media or Azoogel.

Affiliate marketing was an invention by CDNow.com in 1994 and was excelled by Amazon.com when it launched its Affiliate Program, called Associate Program in 1996. The online retailer used its program to generate low cost brand exposure and provided at the same time small websites a way to earn some supplemental income.

BEHAVIORAL TARGETING

Behavioral targeting or behavioral targeting is a technique used by online publishers and advertisers to increase the effectiveness of their campaigns.

Behavioral targeting uses information collected on an individual's web-browsing behavior, such as the pages they have visited or the searches they have made, to select which advertisements to display to that individual. Practitioners believe this helps them deliver their online advertisements to the users who are most likely to be influenced by them.

Behavioral marketing can be used on its own or in conjunction with other forms of targeting based on factors like geography, demographics or the surrounding content.

Examples of behavioral targeting in advertising targeting systems include: Ad LINK 360, Boomerang, Double-Click and under Loop.

These targeting systems allow two main ways of using the technology:

- Firstly they allow advertisers \ site owners to display different creative to different people;
- Secondly they allow publishers to sell inventory against specific segments of their audiences to advertisers.

ONSITE BEHAVIORAL TARGETING

Behavioral targeting techniques may also be applied to content within retail or other e-commerce website as a technique for increasing the relevance of product offers and promotions on a visitor by visitor basis. Again, behavioral data can be combined with demographic and past purchase history in order to produce a greater degree of granularity in the targeting.

Self-learning onsite behavioral targeting systems will monitor visitor response to site content and learn what is most likely to generate a desired conversion event. Some good content for each behavioral trait or pattern is often established using numerous

simultaneous multivariate tests. Onsite behavioral targeting requires relatively high level of traffic before statistical confidence levels can be reached regarding the probability of a particular offer generating a conversion from a user with a set behavioral profile. Some providers have been able to do so by leveraging its large user base, such as Yahoo!. Some providers use a rules based approach, allowing administrators to set the content and offers shown to those with particular traits.

Examples of onsite behavioral targeting systems include: Kefta, Maxi miser, Net mining and Touch Clarity. Unique for Net mining is that they add an interaction layer on top of their solution so you can get in touch with your web visitor at the right time and with the right interaction. Yahoo! Inc. has been offering onsite behavioral targeting for many years as well.

NETWORK BEHAVIORAL TARGETING

Advertising Networks use behavioral targeting in a different way to individual sites. Since they serve many adverts across many different sites, they are able to build up a picture of the likely demographic makeup of internet users. An example would be a user seen on football sites, business sites and male fashion sites. A reasonable guess would be to assume the user is male. Demographic analyses of individual sites provided either internally (user surveys) or externally (COM score \ net ratings) allow the networks to sell audiences rather than sites. Although advertising networks used to sell this product, this was based on picking the sites where the audiences were. Behavioral targeting allows them to be slightly more specific about this.

This service is offered by (among others): Ad link, Ad Viva, Almond Net, Blue Lithium, Burst, Nebu Ad , Phorm and Revenue Science.

ADVERTISEMENT REVENUE MODELS

PAY PER CLICK

Pay per click (PPC) is an advertising model used on search engines, advertising networks, and content websites/blogs , where advertisers only pay when a user actually clicks on an ad to visit the advertiser's website. Advertisers bid on keywords they predict their target market will use as search terms when they are looking for a product or service. When a user types a keyword query matching the advertiser's keyword list, or views a page with relevant content, the advertiser's ad may be shown. These ads are called a "Sponsored link" or "sponsored ads" and appear next to or above the "natural" or organic results on search engine results pages or anywhere a webmaster / blogger chooses on a content page.

Pay per click ads may also appear on content network websites. In this case, ad networks such as Google Ad Sense and Yahoo! Publisher Network attempt to provide ads that are relevant to the content of the page where they appear, and no search function is involved. While many companies exist in this space, Google Ad Words, Yahoo! Search Marketing, and Microsoft ad Center are the largest network operators as of 2007. Minimum prices per click, often referred to as Costs Per Click (CPC), vary depending on the search engine, with some as low at \$0.01. Very popular search terms can cost much more on popular engines. Arguably this advertising model may be open to abuse through click fraud, although Google and other search engines have implemented automated systems to guard against this.

Ads are typically automatically placed when a visitor loads a web page, most commonly initiated via javascript, audio ads normally cannot be stopped once they start. Audio ads vary in length, and a visitor [Place Your Ad Here](#) can add an ad per slot to any specific web page.

COST PER IMPRESSION

Cost Per Impression is a phrase often used in online advertising and marketing related to web traffic. It is used for measuring the worth and cost of a specific e-marketing campaign. This technique is applied with web banners, text links, e-mail spam, and opt-in e-mail advertising, although opt-in e-mail advertising is more commonly charged on a Cost per Action (CPA) basis.

The Cost per Impression is often abbreviated to CPI

This type of advertising arrangement closely resembles Television and Print Advertising Methods for speculating the cost of an Advertisement. Often, industry agreed approximates are used. With Television the Nielsen Ratings are used and Print is based on the circulation a publication has.

For Online Advertising, the numbers of views can be a lot more precise. When a user requests a Web Page, the originating server creates a log entry. Also, a third party tracker can be placed in the web page to verify how many accesses that page had.

PAY PER PLAY

Pay per play (PPP) is an online advertising method that plays an audio advertisement on websites. The term "pay per play" comes from advertisers paying for each audio ad played. Also, the web page playing the audio ad is normally paid for each ad they serve. Ads are typically automatically played when a visitor loads a web page. Most commonly initiated via JavaScript, audio ads normally cannot be stopped once they start. Audio ads vary in length, and a website visitor will usually hear only one advertisement per visit to any specific web page.

There is an advertiser who is planning on an nationwide launch but in select cities and to a select audience say 20-30. He would upload his advertisement on the network and select the profile (age group, gender, marital status, job profile, geography)

PROBLEM STATEMENT

The Internet has brought many unique benefits to marketing including low costs in distributing information and media to a global audience. The interactive nature of Internet media, both in terms of instant response, and in eliciting response at all, are both unique qualities of Internet marketing.

Currently there is no network based service for multimedia advertisement as such and most of the advertisement is textual and image based. The models that are there today for advertising are either static or content based. We are focusing on building a user specific model of advertising.

Working

Scenario 1

With advancement of technology and the large addition in the number of Internet user there is a transition happening of internet and other conventional broadcasting media such as television, radio, print media etc. There is a housewife who is using internet so the advertisement she would view would be related to her profile.

Ex. A new departmental store in her locality or an event such as an end season sale

Scenario 2

There is an advertiser who is planning on a nationwide launch but in select cities and to a select audience say 20-30. He would upload his advertisement on the network and select the profile (age group, gender, marital status, job profile, geography)

Scenario 3

There is an auto insurance dealer who wants to target people who have bought a new vehicle in his particular geography. So he would upload his advertisement on the network and can specify that show it only to people in his geography who had a month ago had responded to car finance or auto sales advertisement, assuming that by now he must have bought his vehicle and now the user can be his potential customer for insurance.

We would be collecting the data elements that are needed to generate a profile of a user in form of input cases being input by the user and this will be continuously updated.

Advantages

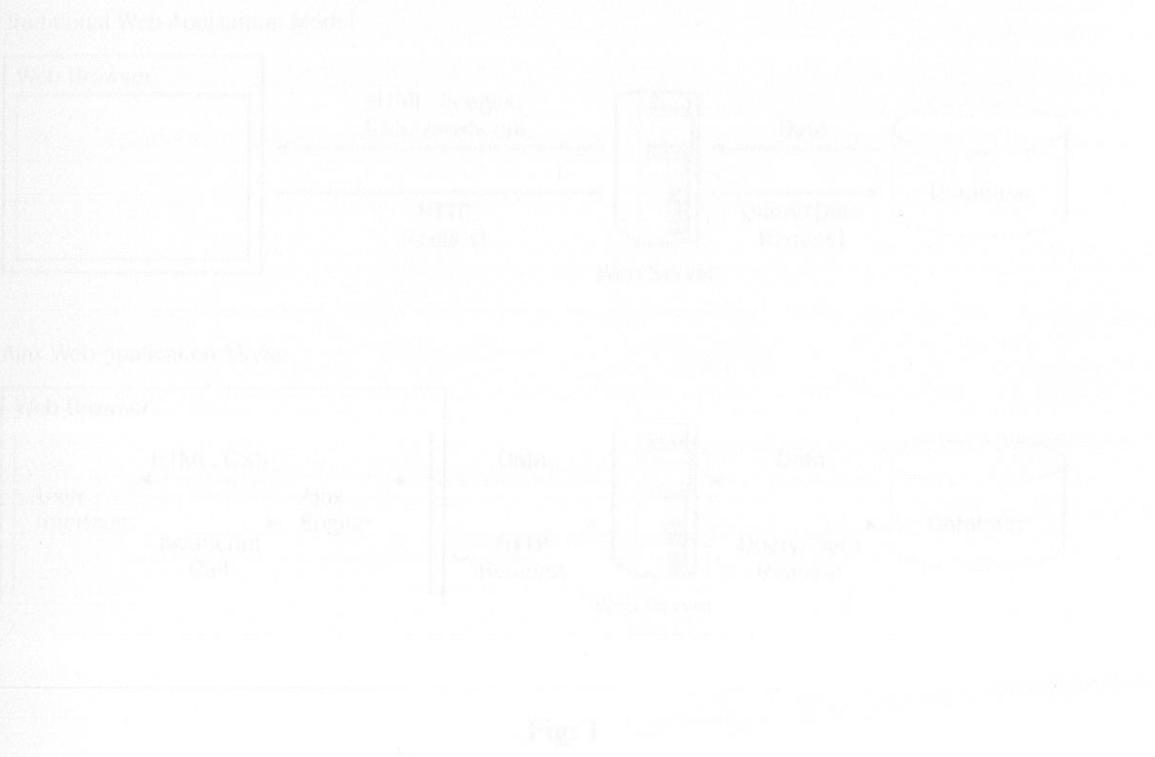
- Multimedia and rich content.
- Give opportunities to for advertising on every scale.
- Behavioral patter of the users of track the suitable costumers.
- Reaches new areas such as mobile advertising and its spin off. (user ads)

With the advancement of technology and the large addition in the number of internet user there is an integration happening of internet and other conventional broadcasting media such as Radio and Television. Internet is increasing becoming the favorite medium for multimedia content delivery.

APPROACH

We have developed an algorithm for searching associated tags with an advertisement that fits with the data tags of the user for which a particular advertisement is being searched from the database of advertisement that will perfectly suite the profile of the user, i.e. the user profile fits the description of the prospective client profile uploaded by the advertiser of that particular advertisement.

We would be collecting the data elements that are needed to generate a profile of a user in form of input cases being input by the user and this will be continuously updated according to the response pattern of the user.



Ajax (Asynchronous JavaScript and XML) is a web development technique used for creating interactive web applications. The intent is to make web pages feel more responsive by exchanging small amounts of data with the server behind the scenes, so that the entire web page does not have to be reloaded each time the user makes a change. This is intended to increase the web page's interactivity, speed, fluidity, and intuitiveness.

TECHNOLOGY USED

ASP.NET – The platform would be developed on the asp.net which is a popular software for web based application development.

SQL server 2000 – All the databases would be hosted on the database server having the requisite tables.

AJAX – The main technology being used is “Ajax” that would be used to display the advertisement at the user end.

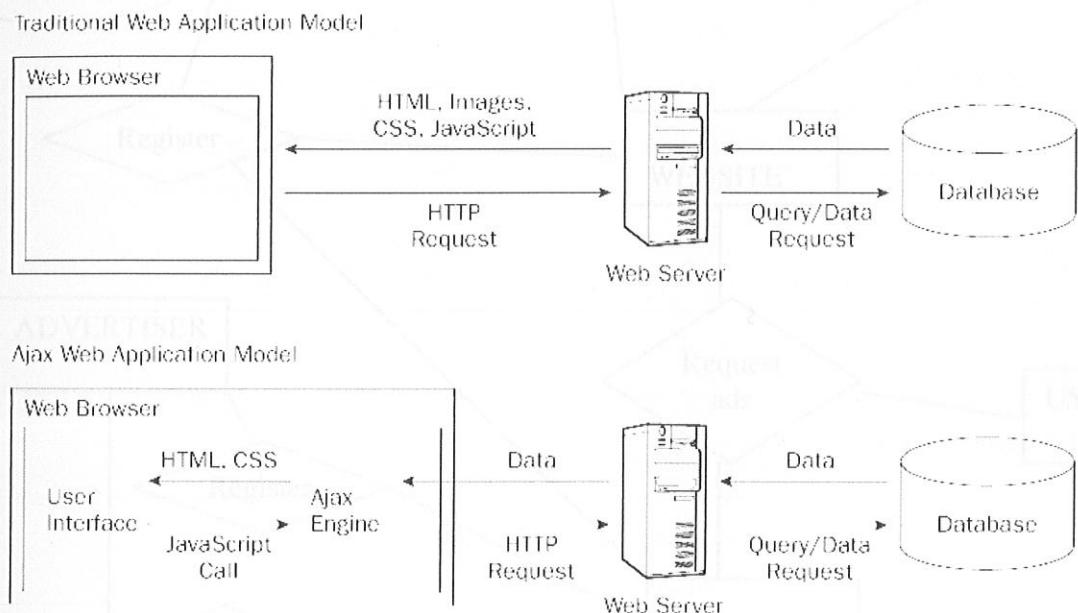


Fig: 1

Ajax (Asynchronous JavaScript and XML), is a web development technique used for creating interactive web applications. The intent is to make web pages feel more responsive by exchanging small amounts of data with the server behind the scenes, so that the entire web page does not have to be reloaded each time the user requests a change. This is intended to increase the web page's interactivity, speed, functionality, and usability.

ENTITY RELATIONSHIP DIAGRAM

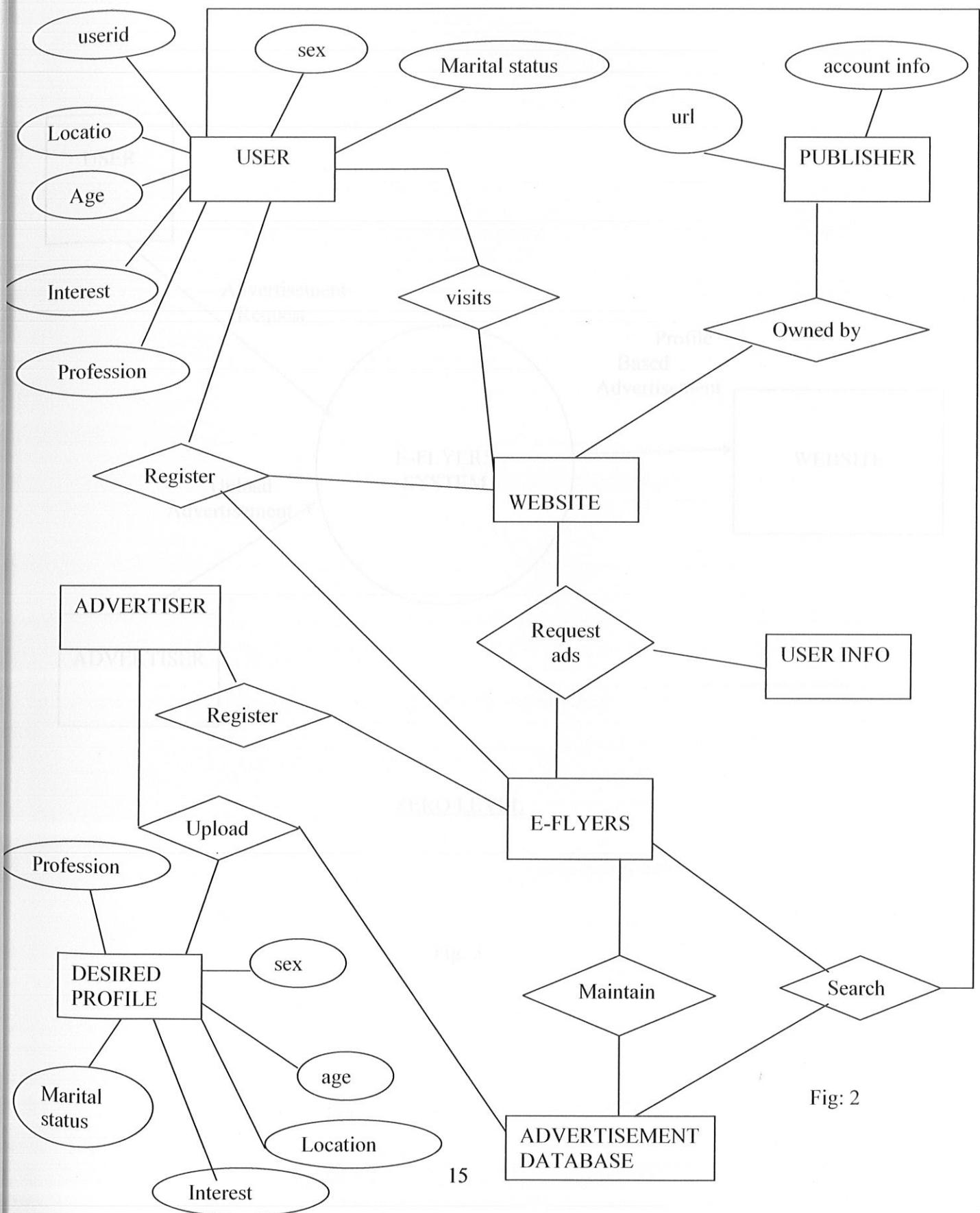


Fig: 2

DATA FLOW DIAGRAM

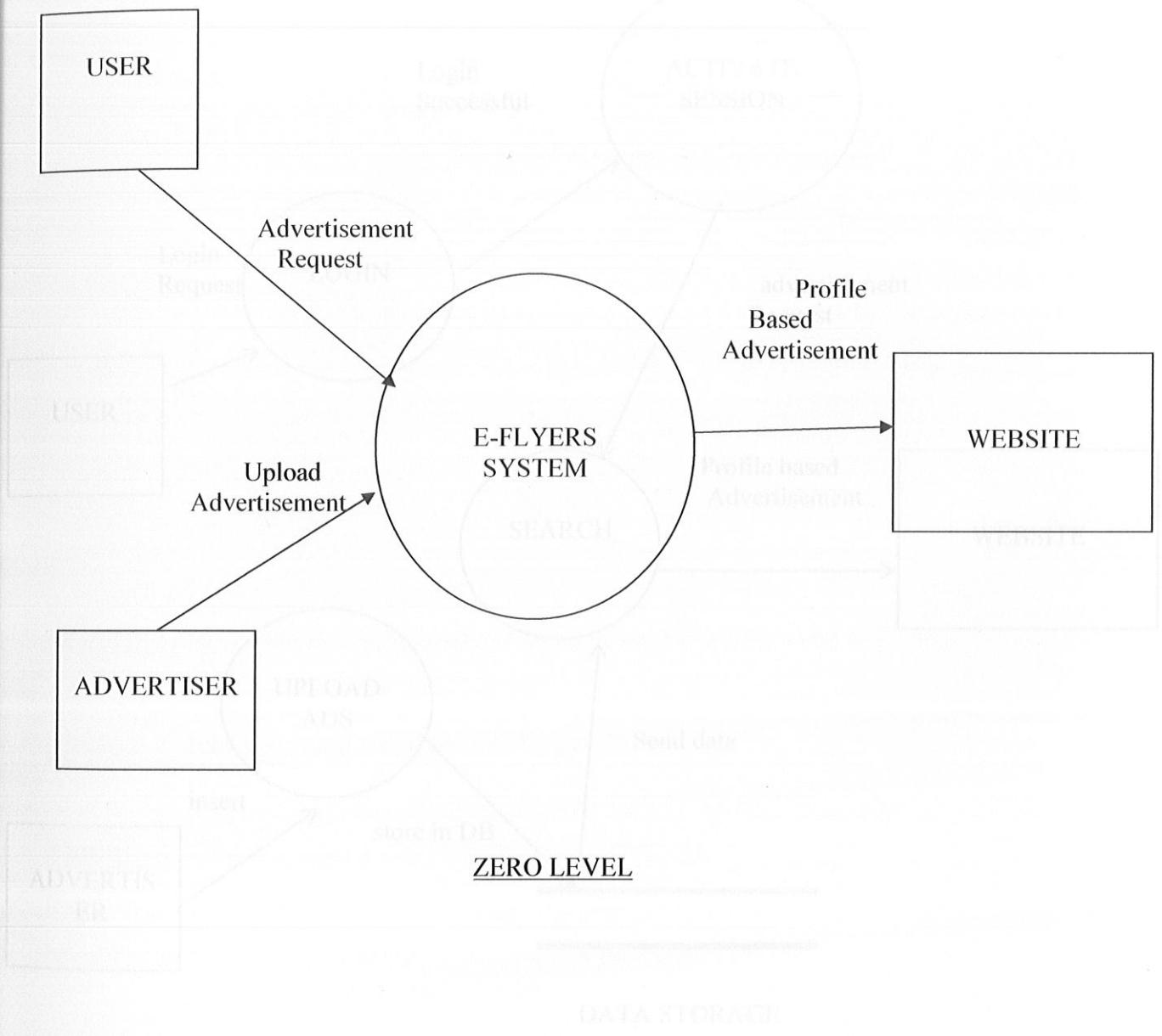


Fig: 3

DATA FLOW DIAGRAM

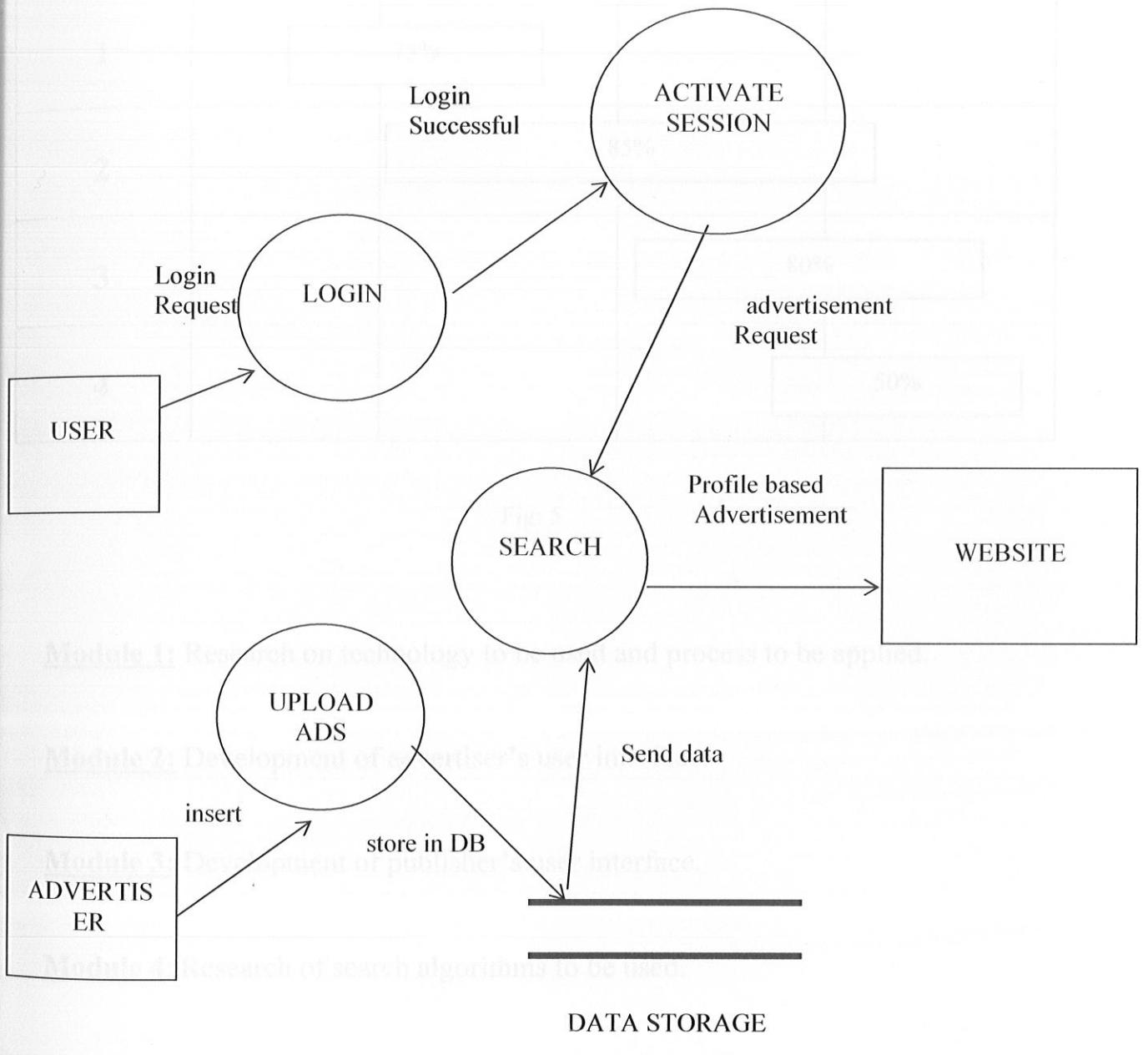


Fig: 4

GANTT CHART

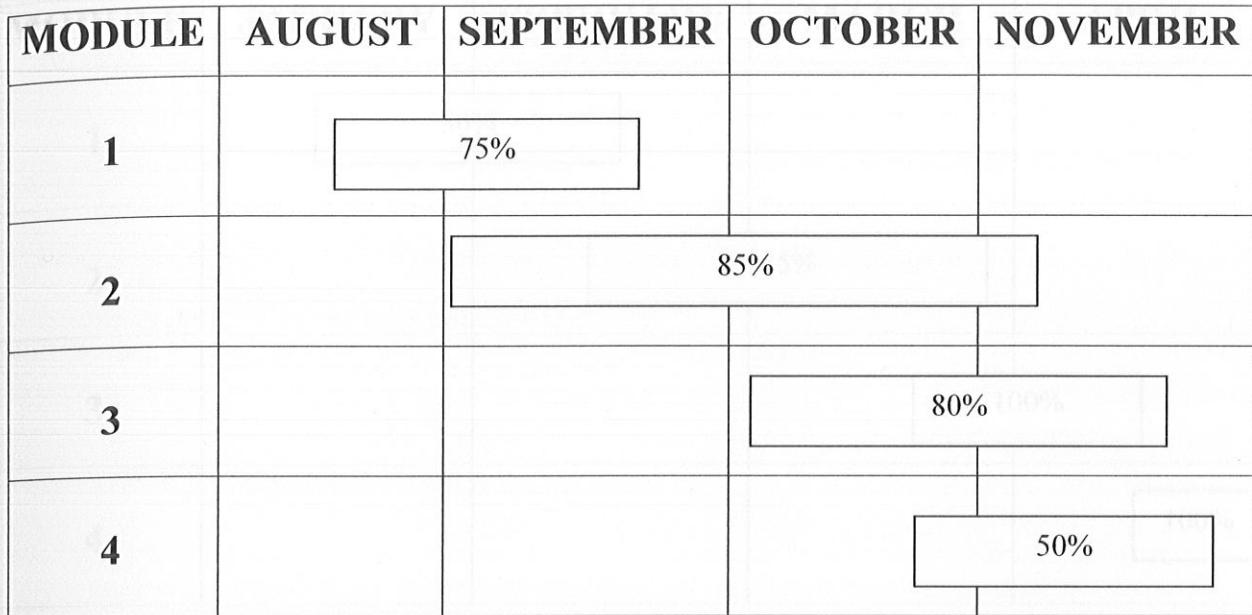


Fig: 5

Module 1: Research on technology to be used and process to be applied.

Module 2: Development of advertiser's user interface.

Module 3: Development of publisher's user interface.

Module 4: Research of search algorithms to be used.

GANTT CHART

MODULE	JANUARY	FEBRUARY	MARCH	APRIL
1		50%		
2		35%		
3			100%	
4				100%

Through the publisher, it would be collecting the data elements that are needed to generate a profile of a user in form of input Fig: 6 being input by the user and this will be continuously updated according to the response pattern of the user.

Module 1: Development of matching algorithm.

Module 2: Development of advertiser's and Publisher's interface.

algorithms to determine the best method to be used to match the desired user's profile.

Module 3: Deployment and integration of modules.

Module 4: Testing of application.

MODULE DECOMPOSITION

ADVERTISER INTERFACE MODULE

In this module we developed an interface for the advertiser to upload his advertisements on the network and select the desired user's profile for his targeted customers. It would be collecting the data elements that are needed to generate a profile of a user in form of input cases being input by the advertiser .

USER INTERFACE MODULE

In this module we developed an interface for the User to register with the network through the publisher. It would be collecting the data elements that are needed to generate a profile of a user in form of input cases being input by the user and this will be continuously updated according to the response pattern of the user

SEARCH AND MATCHING MODULE

In this module we have undertaken extensive research in the field of search and matching algorithms to determine the best method to be used to match the desired user's profile given by the advertiser. We have also generated a response system for the user to track the behavior profile of the user to generate and continually update his interest elements.

SOFTWARE REQUIREMENTS

For efficient functioning our software requirements are:-

1. Internet Explorer 5.0 or above.
2. Macromedia flash plug-in should be installed.
3. Video and sound drivers should be installed.
4. Sufficient internet bandwidth.

TARGET MODE

Target Mode is helpful for those organizations that have a specific audience to interact with. It defines elements such as age group, interests, gender, and other demographic factors who can be effectively used by small businesses who have geographic reach. It can also be used by specialized pushes who want to reach their specific audience.



METHODOLOGY FOR ADVERTISEMENT MATCHING

The advertisers can upload advertisement on the platform in two modes, namely:-

- Broadcast Mode
- Target Mode

BROADCAST MODE

This mode is useful for large organizations who want to use this platform for marketing and brand building. Advertisements put up under broadcast mode would be available in specified geographies without constraints of interests, gender, profession etc.

When a user logs in to the system it first authenticates the identity of the user. After **TARGET MODE** only, the system extracts information like geography, age, interests, gender, marital status etc from the database.

Target Mode is helpful for those organizations that have a specific audience in mind with defined elements such as geography, interests, gender, and profession. This mode can be effectively used by small merchants who have geographical constraints for business or either by specialized business who want to reach their specific audience.

Then the algorithm would extract all the merchant ids that have either entered in geography,

- Add
- Add a rule
- Add a rule to a campaign
- Add a rule to a campaign

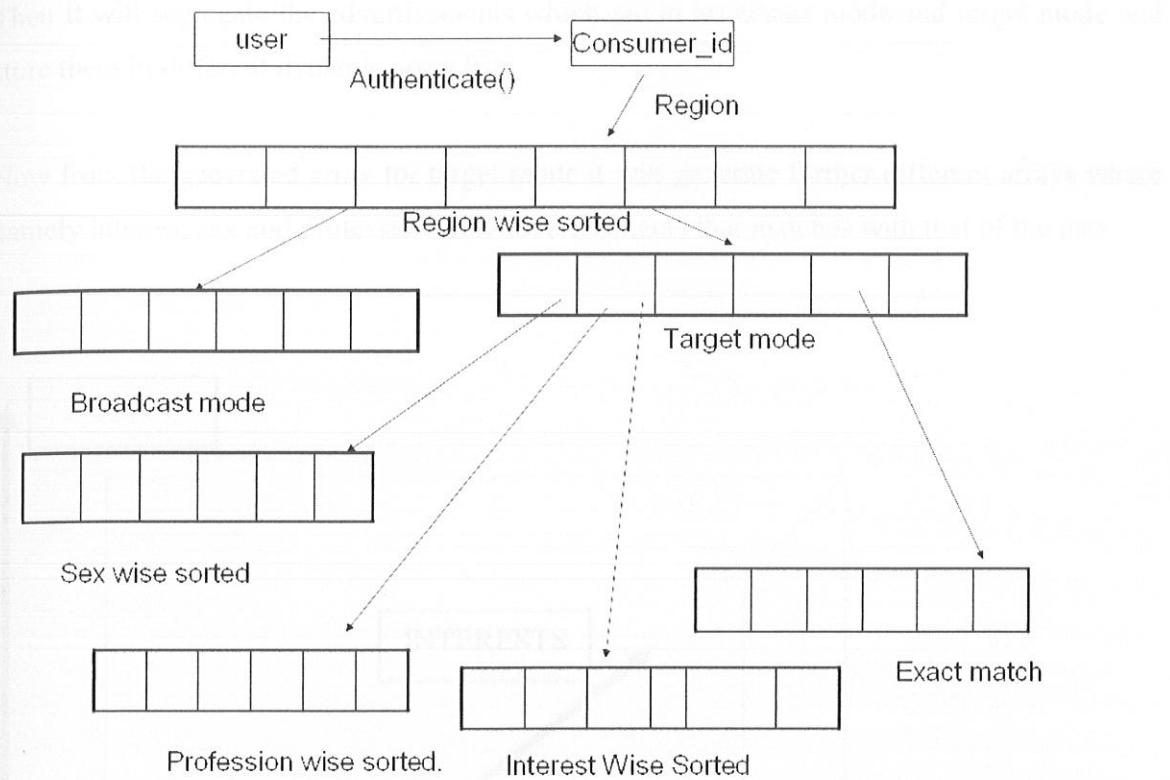


Fig: 7

When a user logs in to the system it firsts authenticates the identity of the user. After confirming the identity, the system extracts information like geography, age, interests, gender, marital status etc from the database.

Then the algorithm search for advertisements that are specified to the geography of the user.

For ex. There is a male who lives in shimla, Himachal Pradesh, India, Asia.

Then the algorithm would extract all the transaction ids that have either entered in geography-

- Asia
- Asia + India
- Asia + India + Himachal Pradesh
- Asia + India + Himachal Pradesh + Shimla

Then it will segregate the advertisements which are in broadcast mode and target mode and store them in different dynamic array lists.

Now from the generated array for target mode it will generate further different arrays where namely interest, sex and profession with advertisements that matches with that of the user.

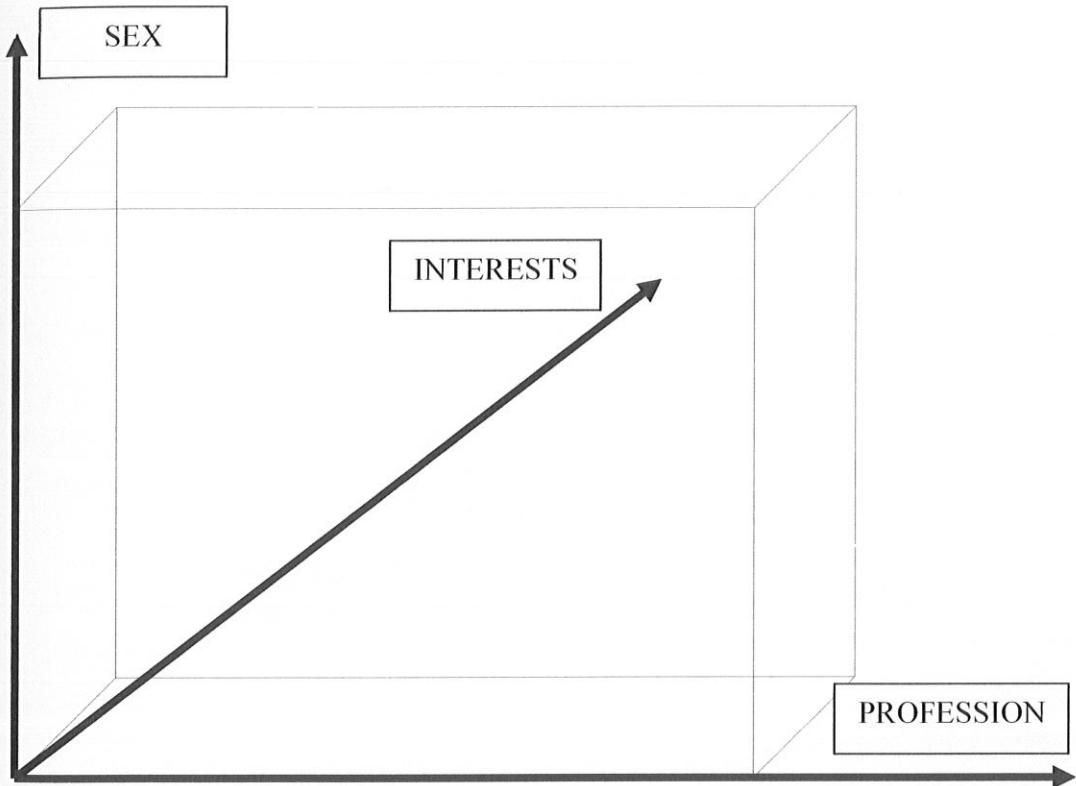


Fig: 8

Now for advertisements that have exact match to specific profile of the user we will find the common entities in different lists generated of interests, profession, sex and store them in an array starting with best match.

We will randomize the arrays generated namely broadcast array, sex sorted array, profession sorted array, interest sorted array and select some advertisements from the ex. 5 advertisements from each list and store it in another array. And similarly top 5 advertisements are selected from the array list for exact match and copied into the main array list.

Now this main array list is randomized and copied into the session variables of the user.

Every time a user will open a new webpage while continuing in the session a new advertisement from the list would be fetched from the database and streamed to the webpage which the user is viewing.

on the video player and on the background shows our application by itself of interest

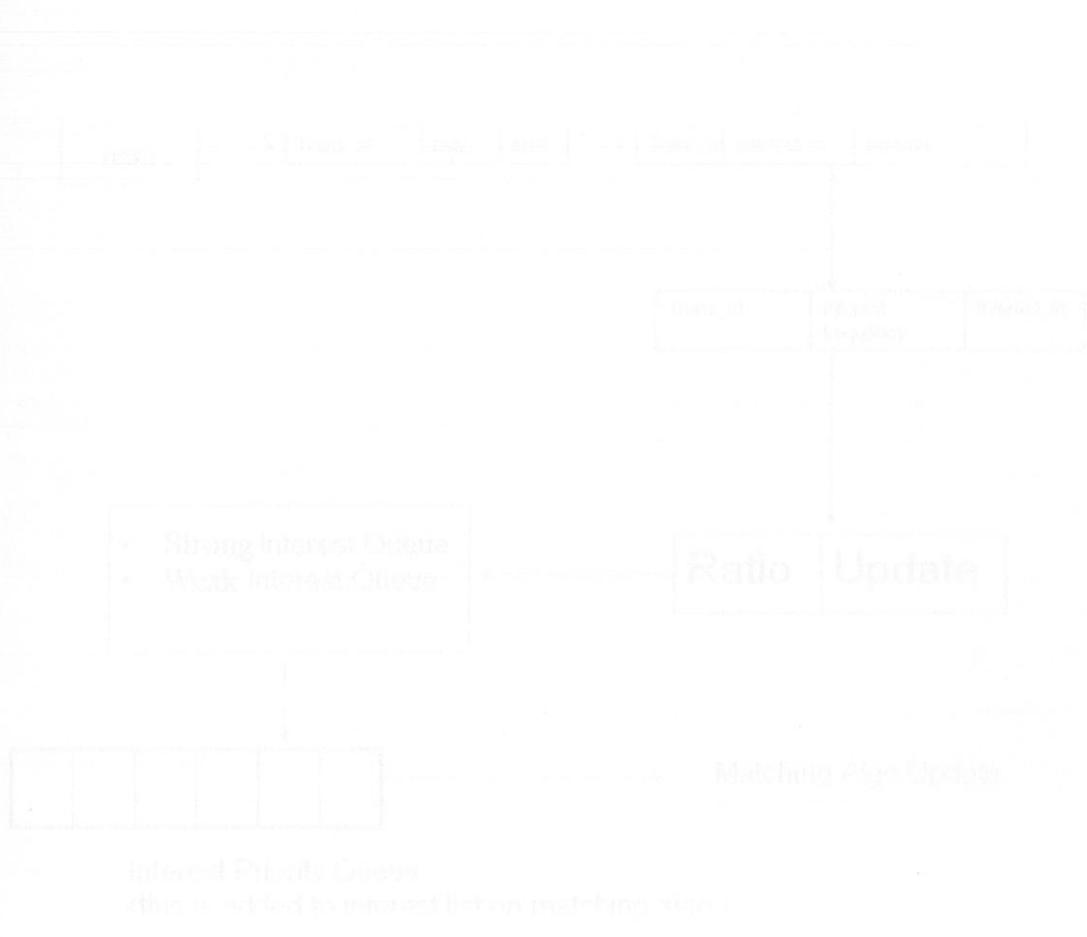


Fig. 6

The above representation shows that how the whole process will gathered the information regarding user's long term as Strong and short term as Weak Interests.

Finally we can say that the user will get the whole ads according to his profile to the profile of that user by using different methodologies and putting these ads in his session.

RESPONSE SYSTEM

As soon as user click on the player or response any advertisement the following data would be inserted in the transaction.

In this system we basically track the behavior of user so as to maintain his short term as well as long term interest. We are tracking his behavior according to the number of clicks by him on the video player and on the basis of these clicks we concluding his area of interests.

-transactional identity of advertisement

-publisherid (identity of publisher)

-date&time (information of click)

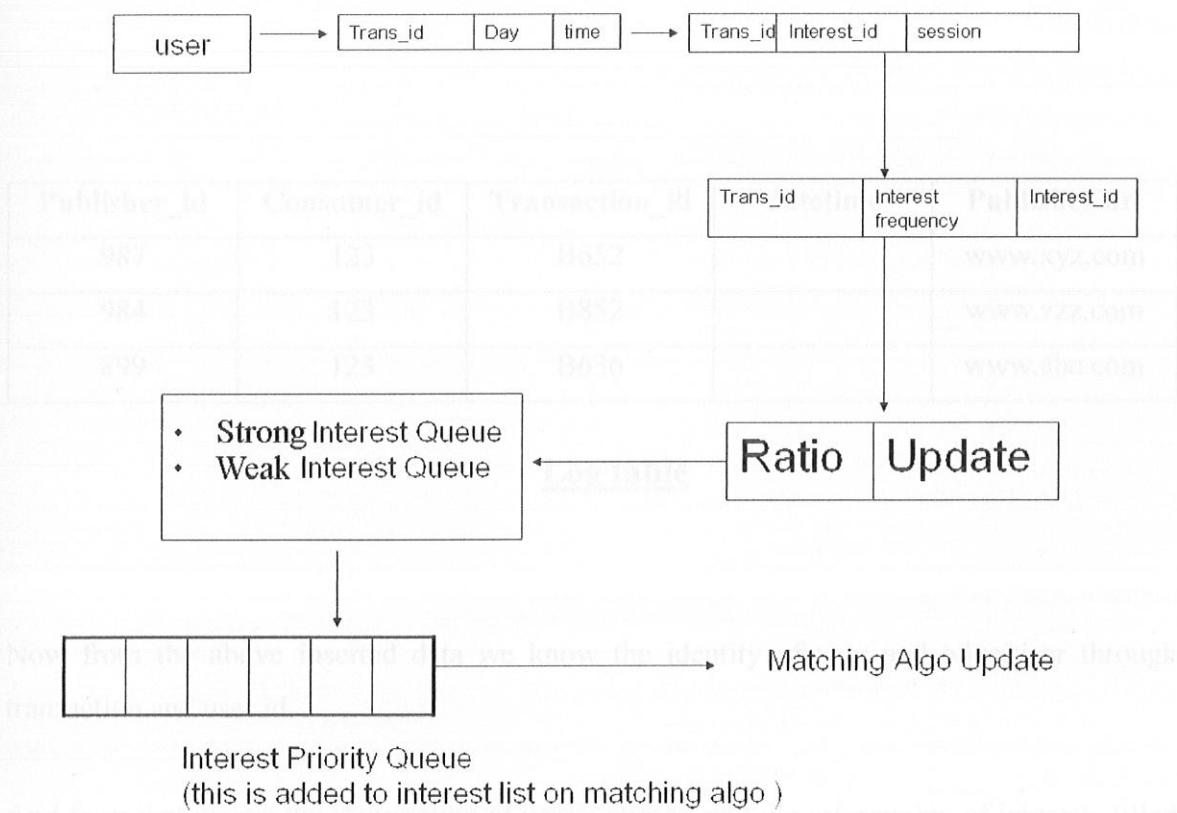


Fig: 9

The above representation shows that how the whole process will gathered the information regarding user's long term as strong and short term as weak interests.

Finally we creating a session for the user and gathering the whole advertisements according to the profile of that user by using different methodology and putting these ads in his session.

PROCEDURE FOR LONG TERM INTEREST

As soon as user click on the player or response any advertisement the following data would be inserted in the log table:

-consumerid (identity of user)

-transactionid (identity of advertisement)

-publisherid (identity of publisher)

-date&time (information of click) is added to the user interest table so that we know that the user is now interested in such interest of advertisement.

Publisher_id	Consumer_id	Transaction_id	datetime	Publisher url
987	123	B652		www.xyz.com
984	123	B852		www.zzz.com
899	123	B636		www.abc.com

Log table

Now from the above inserted data we know the identity of user and advertiser through transaction and user id.

And from that we get the information of user's interest and the information of interests filled by the advertiser for that particular advertisement.

Then our system will check the interest of that user with the advertiser's filled interest and if they match to each other, the number of clicks for that interest will increment to one.

Consumer_id	Interest_id	Clicks
123	5	3
123	7	2
123	9	5
123	4	8

User interest table

Otherwise a new interest would be added to the user interest table so that we know that the user is now interested in such interest of advertisement.

Consumer_id	Interest_id	Clicks
123	5	3
123	7	2
123	9	5
123	8	1
123	3	1
123	1	1
123	4	8

User interest table

Then on the basis of high frequency of clicks we would select the first five interest of that particular user and maintain that information in database and these information can be change again and again as the user will respond to different types of advertisements.

Consumer_id	Int 1	Int 2	Int 3	Int 4	Int 5
123	4	9	5	7	8

User behaviour table

PROCEDURE FOR SHORT TERM INTEREST

Similar to Long term interest we map the Short term interest of the user i.e. the interest user had responded in the last two weeks.

- consumerid (identity of user)
- transactionid (identity of advertisement)
- publisherid (identity of publisher)
- date&time (information of click)

Publisher_id	Consumer_id	Transaction_id	datetime	Publisher url
987	123	B652		www.xyz.com
984	123	B852		www.zzz.com
899	123	B636		www.abc.com

Log table

When the user login we will select those advertisements from the log table to which the user had responded in the last two weeks and on the basis of that we will populate this table as shown below

Consumer_id	Interest_id	Clicks
123	5	3
123	7	2
123	9	5
123	8	1
123	3	1
123	1	1
123	4	8

Short term user interest table

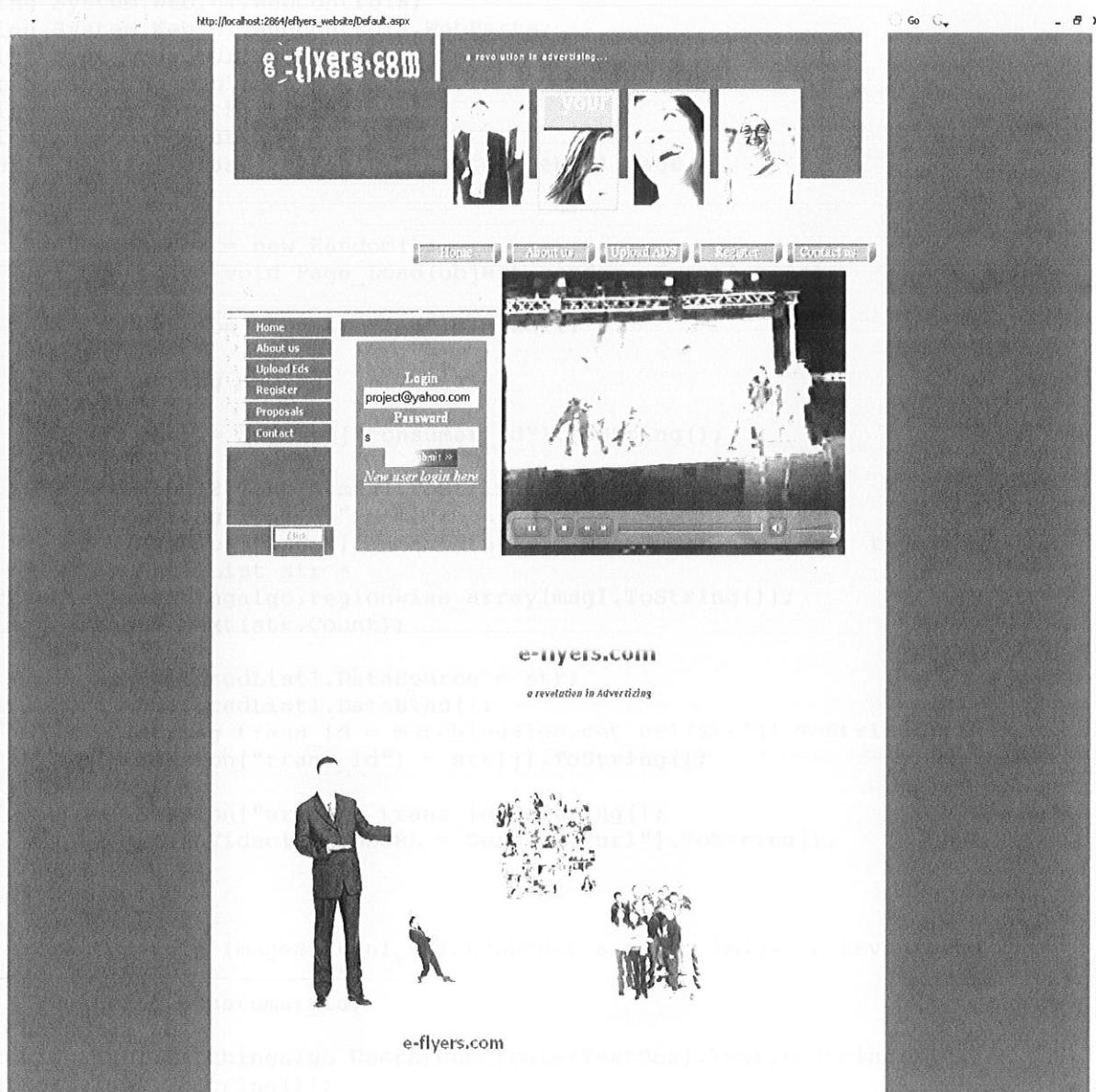
Then similarly on the basis of high frequency of clicks as in long term interest we would select the first five interest of that particular user and maintain that information in database and these information can be change again and again as the user will respond to different types of advertisements.

Consumer_id	Int 1	Int 2	Int 3	Int 4	Int 5
123	4	9	5	7	8

Short term user behaviour table

GRAPHIC USER INTERFACE

Homepage



Default.aspx

Default.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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 matchingalgo1;
using System.Data.SqlClient;
using response_system;
public partial class _Default : System.Web.UI.Page
{
    int j,i;
    Random random = new Random();
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["consumer_id"] != null)
        {
            string msg1;

            msg1 = Session["consumer_id"].ToString();

            Label2.Text = msg1.ToString();
            string type = "email";
            Session["user"] = matchingalgo.know userinfo(msg1, type);
            ArrayList str =
                (ArrayList)matchingalgo.regionwise_array(msg1.ToString());
            j = random.Next(str.Count);

            BulletedList1.DataSource = str;
            BulletedList1.DataBind();
            string trans_id = matchingalgo.get_url(str[j].ToString());
            Session["trans_id"] = str[j].ToString();

            Session["url"] = trans_id.ToString();
            FlashVideo1.VideoURL = Session["url"].ToString();
        }
    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        string constumer_id;
        constumer_id =
            matchingalgo1.matchingalgo.UserAthenticate(TextBox1.Text.ToString(),
            TextBox2.Text.ToString());
        Session["consumer_id"] = constumer_id.ToString();
        if (constumer_id != "1")
        {
            Label1.Text = constumer_id.ToString();
            string type = "email";
            Session["user"] = matchingalgo.know userinfo(constumer_id, type);
            ArrayList str =
                (ArrayList)matchingalgo.regionwise_array(constumer_id.ToString());
            string trans_id;
            j = random.Next(str.Count);
            if (str.Count != 0)
```

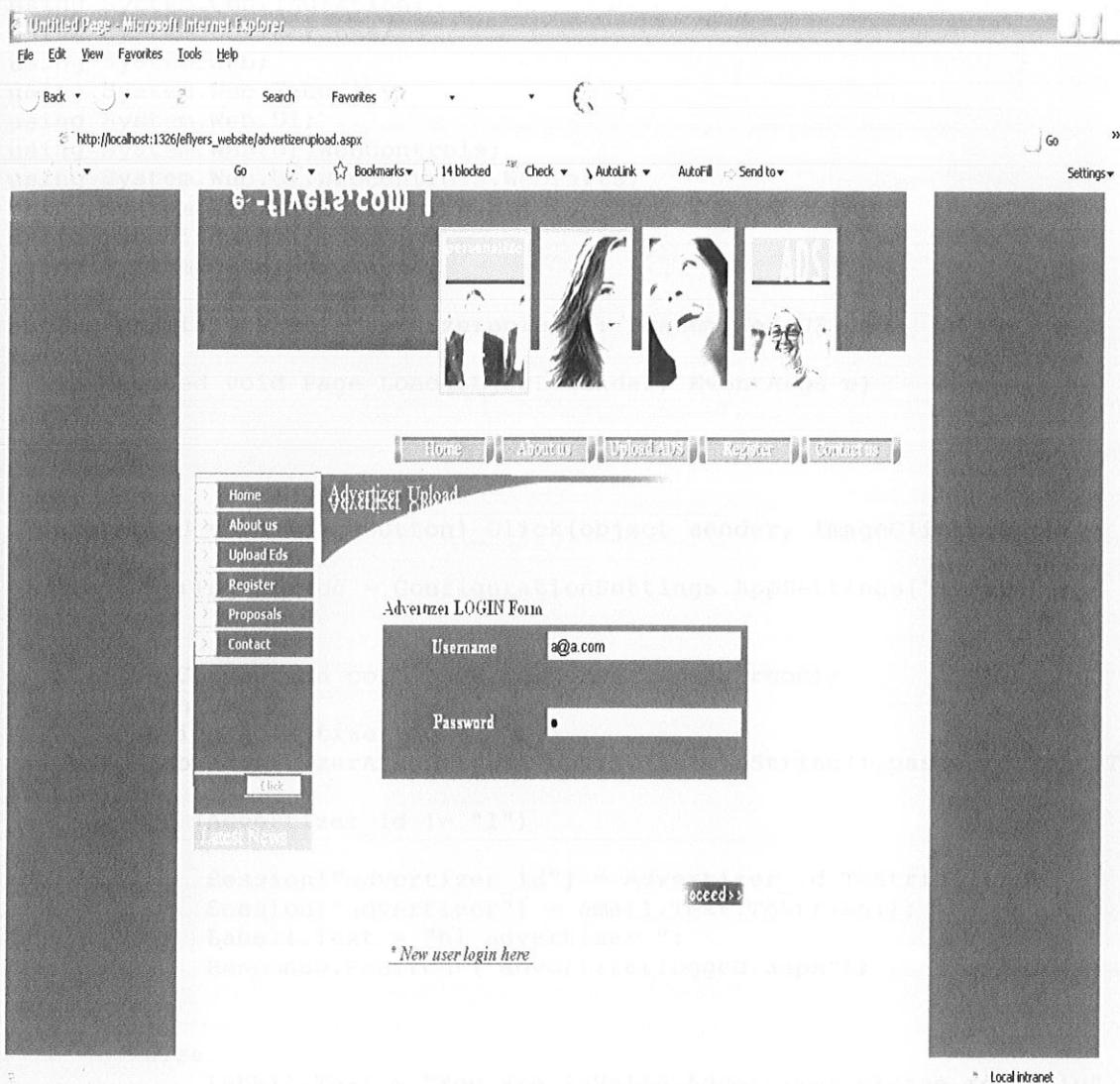
```
        {
            trans_id = matchingalgo.get_url(str[j].ToString());
            Session["url"] = trans_id.ToString();
            FlashVideo1.VideoURL = Session["url"].ToString();
            Session["login"] = "no";
        }
        else
        {
            FlashVideo1.VideoURL = "~/uploaded/school chale hum.flv";
        }

        BulletedList1.DataSource = str;
        BulletedList1.DataBind();
    }
    else
        Label2.Text = "You are InValid User";
}

protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.RemoveAll();
    Response.Redirect("Default.aspx");
}
protected void LinkButton2_Click(object sender, EventArgs e)
{
string consumer_id=Session["consumer_id"].ToString();
    string trans_id=Session["trans_id"].ToString();

response_system.response_sytem.insert_userlog(consumer_id,trans_id);
}
}
```

ADVERTISER LOGIN



Advertiser upload.aspx

Advertiser upload.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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 matchingalgo;
using System.Data.SqlClient;

public partial class advertizerupload : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];

        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        String Advertizer_id =
matchingalgo.advertizerAthuncate(email.Text.ToString(),password.Text.ToString());
        if (Advertizer_id != "1")
        {
            Session["advertiser_id"] = Advertizer_id.ToString();
            Session["advertiser"] = email.Text.ToString();
            Label1.Text = "hi advertiser ";
            Response.Redirect("advertiserlogged.aspx");
        }
        else
            Label1.Text = "You are InValid Advertiser please re-login";
    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
    }
}
```

NEW ADVERTISER MODULE

http://localhost:2864/eflyers_website/advertiserupload.aspx

Advertiser Upload

Advertiser Personal Information Form

Email: Treated as
UserName: lakshya.shrivastava@gmail.com

Choose Password: *****

Confirm Password:

Company Shop Name:

Designation:

Company's Description:

Other Relevant Information

Advertiser ID: Email: Password: Temporyfield: Proceed >>

New advertiser.aspx

New advertiser.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;

public partial class advertiserupload : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");

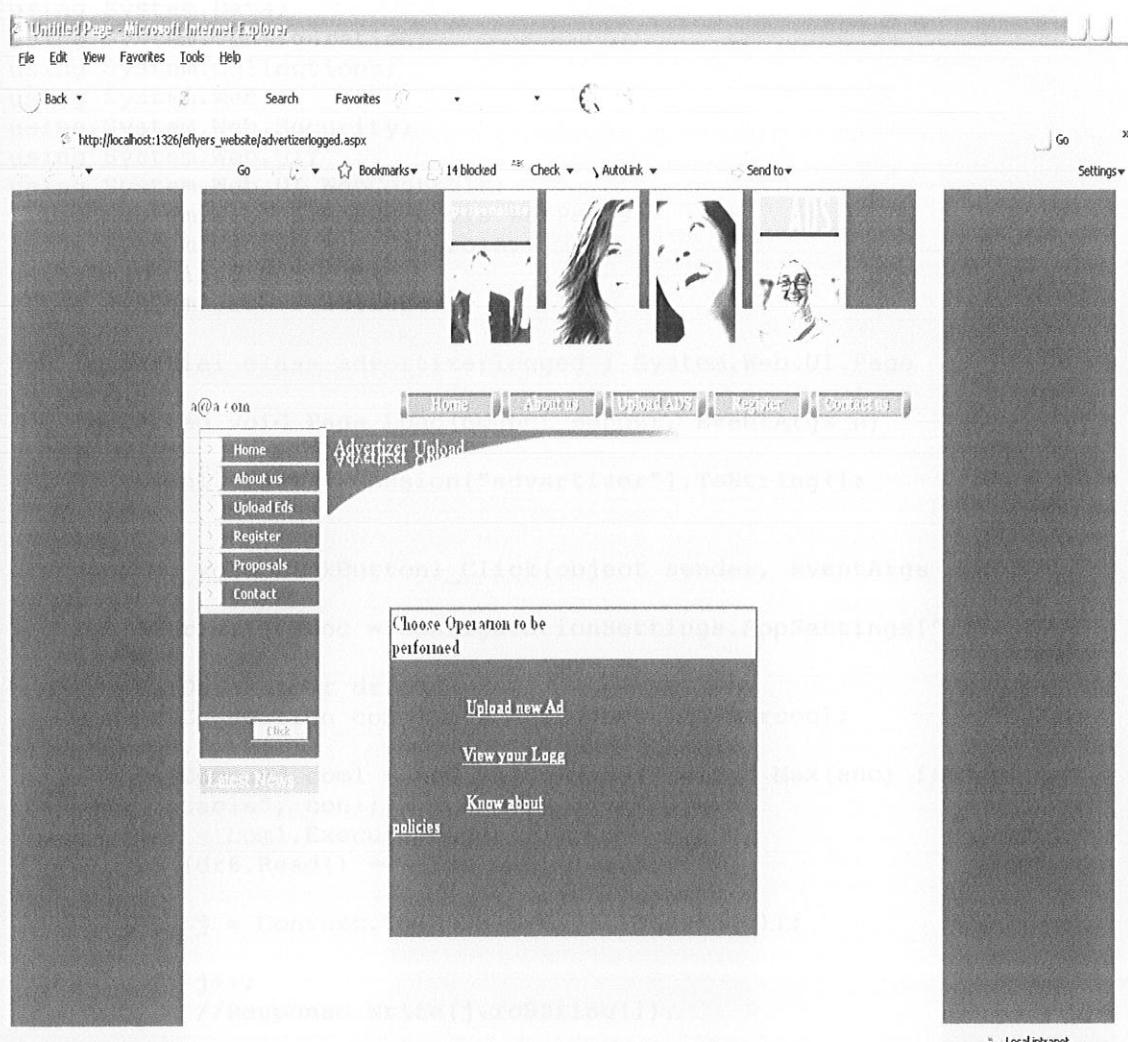
    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();
    }

    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        int user_id = email.Text.GetHashCode();

        Session["advertiser_id"] = user_id.ToString();
        Session["advertiser"] = email.Text.ToString();
        SqlCommand com = new SqlCommand("insert into
advertiser(user_id,email,password,compname,desig,compdesc,info) values('"
+ Session["advertiser_id"] + "','" + email.Text + "','" + password.Text +
"', '" + compname.Text + "','" + desig.Text + "','" + compdesc.Text + "'",
'" + info.Text + "')", con);
        com.ExecuteNonQuery();
        user_id++;
        Session["trans_id"] = user_id.ToString();
        Response.Redirect("portfoli_Geo.aspx");
        con.Close();

    }
}
```

OPTIONS FOR ADVERTISER



Advertiselogged.aspx

Advertiselogged.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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 matchingalgol;
using System.Data.SqlClient;

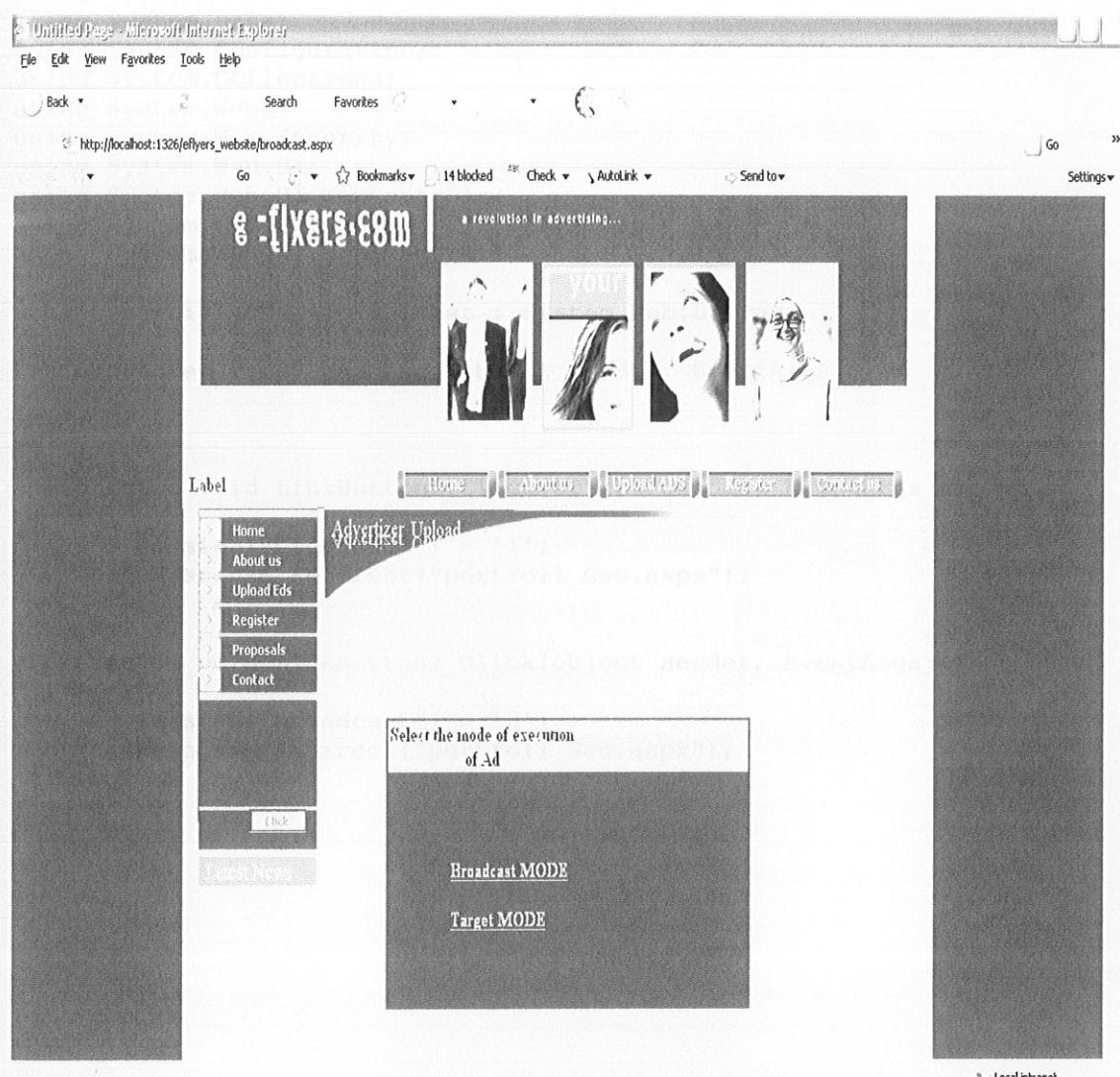
public partial class advertizerlogged : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = Session["advertiser"].ToString();

    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        int j;
        SqlDataReader dr6;
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        SqlCommand com1 = new SqlCommand("select Max(sno) from transactiontable", con);
        dr6 = com1.ExecuteReader();
        if (dr6.Read() == true)
        {
            j = Convert.ToInt32(dr6[0].ToString());

            j++;
            //Response.Write(j.ToString());

            Session["trans_id"] = j.ToString(); ;
            Response.Redirect("broadcast.aspx");
        }
        con.Close();
    }
}
```

MODES FOR UPLOAD



Broadcast.aspx

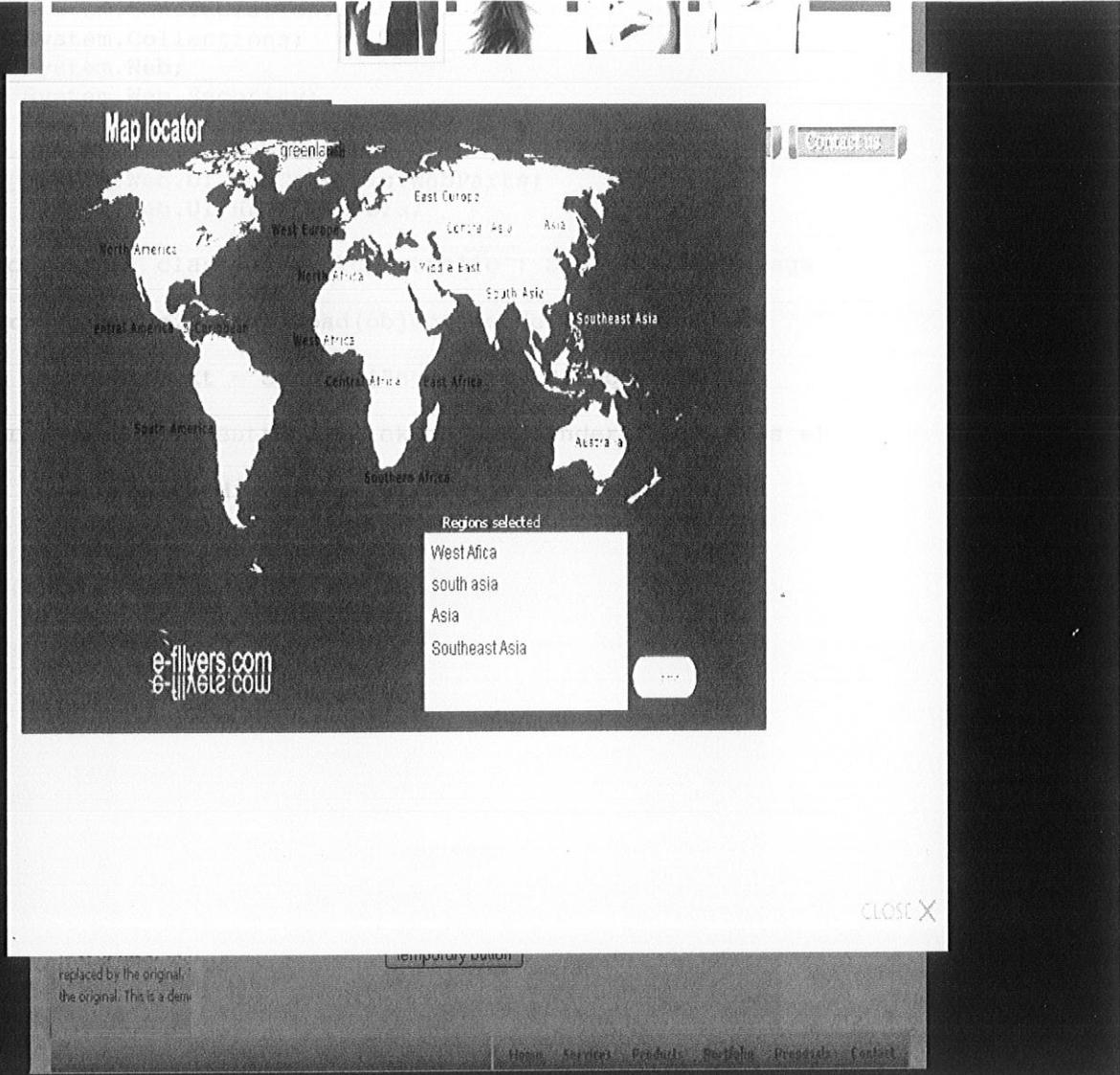
Broadcast.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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;

public partial class broadcast : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void LinkButton1_Click(object sender, EventArgs e)
    {
        Session["broadcast"] = '1';
        Response.Redirect("portfoli_Geo.aspx");
    }
    protected void LinkButton2_Click(object sender, EventArgs e)
    {
        Session["broadcast"] = '0';
        Response.Redirect("portfoli_Geo.aspx");
    }
}
```

GEOGRAPHIC LOCATIONS MODULE



[Portfoliogeo.aspx](#)

Portfoliogeo.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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;

public partial class advbuildportfolio : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = Session["advertizer"].ToString();
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        Response.Redirect("selectRegion.aspx");
    }
}
```

REGIONS

http://localhost:2664/eflyers_website/selectRegion.aspx

Latest Headlines

Label

Home | About us | Upload ADS | Register | Proposals | Contact

Makita Portfolio

Geographic Location

You have selected following Regions.

Region	region	region_id
<input type="checkbox"/> Asia	Asia	1
<input type="checkbox"/> Americ	Americ	2
<input type="checkbox"/> chin	chin	3
<input type="checkbox"/> anica	anica	4
<input type="checkbox"/> gfh	gfh	5
<input type="checkbox"/> jkghkj	jkghkj	6

Proceed >>

you have selected
countries : States : city

flyers_website/countryform.aspx?region_id=1

Selectregion.aspx

Selectregion.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;

public partial class selectRegion : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");
    protected void Page_Load(object sender, EventArgs e)
    {
        //lk.Attributes.Add("onclick",
"window.open('countryform.aspx?region_id=' + e.Item.Cell[x].Text, null, 'heigh
t=250, width=250, status= client window, resizable= no, scrollbars=no,
toolbar=no, left=400, menubar=no '');");
        con.Open();
        //Label1.Text = Session["advertizer"].ToString();
        fillgrid();
        ArrayList myList = new ArrayList();
        ArrayList ds = (ArrayList)Session["countries"];

        BulletedList1.DataSource = ds;
        BulletedList1.DataBind();
        ArrayList myList1 = new ArrayList();
        ArrayList ds1 = (ArrayList)Session["state"];

        BulletedList2.DataSource = ds1;
        BulletedList2.DataBind();
        ArrayList myList2 = new ArrayList();
        ArrayList ds2 = (ArrayList)Session["city"];

        BulletedList3.DataSource = ds2;
        BulletedList3.DataBind();
    }
    public void fillgrid()
    {

        SqlDataAdapter da;
        string str = "select * from regiontable";
        DataSet ds = new DataSet();
        da = new SqlDataAdapter(str, con);
        da.Fill(ds);
        DataGrid1.DataSource = ds;
        DataGrid1.DataBind();
    }
}
```

```
protected void DataGrid1_SelectedIndexChanged1(object sender,
EventArgs e)
{
}

protected void ImageButton1_Click(object sender, ImageClickEventArgs
e)
{
    ArrayList MyArray = new ArrayList();
    for (int i = 0; i < DataGrid1.Rows.Count; i++)
    {
        GridViewRow row = DataGrid1.Rows[i];
        bool isChecked =
((CheckBox)row.FindControl("chkSelect")).Checked;

        if (isChecked)
        {
            Response.Write(DataGrid1.Rows[i].Cells[1].Text);
            MyArray.Add(DataGrid1.Rows[i].Cells[1].Text);

        }
    }
    Session["region"] = MyArray;
    Response.Redirect("selectedLocation.aspx");
}
}
```

Country

india [go detailed](#)

pakistan [go detailed](#)

[Submit >>](#)

[Proceed >>](#)

you have selected

region countries :

♦ india

♦ pakistan

States city

♦ madhya pradesh

♦ uttar pradesh

city

♦ delhi

♦ mumbai

♦ chennai

♦ hyderabad

♦ Bangalore

Countryform.aspx

State

madhya pradesh [go detailed](#)

uttar pradesh [go detailed](#)

[Submit >>](#)

[Proceed >>](#)

you have selected

countries

♦ india

♦ pakistan

States

♦ madhya pradesh

♦ uttar pradesh

city

Stateform.aspx

Country Information

City

bhopal go detailed

the capital city of madhya pradesh
is bhopal. it is located in central

india. it is the second largest city in

central india after jaipur.

it is the administrative center of

madhya pradesh state and

the district of bhopal.

it is also the headquarter of

the bhopal division of the

central railway zone of indian

railways. it is the third largest

city in madhya pradesh state.

it is the second largest city in

central india after jaipur.

it is the administrative center of

madhya pradesh state and

the district of bhopal.

it is also the headquarter of

the bhopal division of the

central railway zone of indian

railways. it is the third largest

city in madhya pradesh state.

it is the second largest city in

central india after jaipur.

it is the administrative center of

madhya pradesh state and

the district of bhopal.

it is also the headquarter of

the bhopal division of the

central railway zone of indian

railways. it is the third largest

city in madhya pradesh state.

it is the second largest city in

central india after jaipur.

it is the administrative center of

madhya pradesh state and

the district of bhopal.

it is also the headquarter of

the bhopal division of the

central railway zone of indian

railways. it is the third largest

city in madhya pradesh state.

it is the second largest city in

central india after jaipur.

it is the administrative center of

madhya pradesh state and

the district of bhopal.

it is also the headquarter of

the bhopal division of the

central railway zone of indian

railways. it is the third largest

city in madhya pradesh state.

it is the second largest city in

central india after jaipur.

it is the administrative center of

madhya pradesh state and

the district of bhopal.

[submit >>]

[cancel >>]

you have selected
countries

- india
- pakistan

: States

- madhya pradesh
- uttar pradesh

CITY

- bhopal

Cityform.aspx

EventArea: 01

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

Countryform.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;
using System.Text;

public partial class countryform : System.Web.UI.Page
{
    SqlDataAdapter myda;
    DataSet ds;
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");
    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();
    }

    protected void CheckBoxList1_SelectedIndexChanged(object sender,
EventArgs e)
    {
    }

    protected void ImageButton1_Click(object sender, ImageClickEventArgs
e)
    {
        StringBuilder str = new StringBuilder();
        ArrayList MyArray = new ArrayList();
        // Select the checkboxes from the GridView control
        for (int i = 0; i < GridView1.Rows.Count; i++)
        {
            GridViewRow row = GridView1.Rows[i];
            bool isChecked =
((CheckBox)row.FindControl("chkSelect")).Checked;

            if (isChecked)
            {
                //SqlCommand com56 = new SqlCommand("insert into
temptable(country) values('" + GridView1.Rows[i].Cells[1].Text + "')",
con);
                //com56.ExecuteNonQuery();
                MyArray.Add(GridView1.Rows[i].Cells[1].Text);
                // str.Append(GridView1.Rows[i].Cells[1].Text);
            }
        }
        Session["countries"] = MyArray;
        String[] intMyArrayLocal = (String[])Session["contries"];

        ArrayList myList = new ArrayList();
    }
}
```

```

ArrayList ds = (ArrayList)Session["countries"];

BulletedList1.DataSource = ds;
BulletedList1.DataBind();
ArrayList myList1 = new ArrayList();
ArrayList ds1 = (ArrayList)Session["state"];

BulletedList2.DataSource = ds1;
BulletedList2.DataBind();
ArrayList myList2 = new ArrayList();
ArrayList ds2 = (ArrayList)Session["city"];

BulletedList3.DataSource = ds2;
BulletedList3.DataBind();
// string sr=Request.QueryString["region_id"].ToString();
// Response.Redirect("countryform.aspx?region_id="+sr);

}

protected void SqlDataSource1_Selecting(object sender,
SqlDataSourceSelectingEventArgs e)
{
    // TODO: Add your code here
}

protected void GridView1_SelectedIndexChanged1(object sender,
EventArgs e)
{
    ArrayList myList1 = new ArrayList();
    ArrayList ds = (ArrayList)Session["countries"];
}

protected void ImageButton2_Click(object sender, ImageClickEventArgs
e)
{
    Response.Redirect("selectRegion.aspx");
}

protected void GridView2_SelectedIndexChanged(object sender, EventArgs
e)
{
    ArrayList myList2 = new ArrayList();
    ArrayList ds2 = (ArrayList)Session["city"];
}

protected void BulletedList3_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void BulletedList2_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void BulletedList1_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void SqlDataSource1_Selecting(object sender,
SqlDataSourceSelectingEventArgs e)
{
    // TODO: Add your code here
}

protected void GridView1_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void GridView2_SelectedIndexChanged(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void ImageButton2_Click(object sender, ImageClickEventArgs
e)
{
    // TODO: Add your code here
}

protected void BulletedList3_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void BulletedList2_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

protected void BulletedList1_SelectedIndexChanged1(object sender, EventArgs
e)
{
    // TODO: Add your code here
}

```

Stateform.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;
using System.Text;

public partial class Stateform : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");
    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();

        ArrayList myList = new ArrayList();
        ArrayList ds = (ArrayList)Session["countries"];

        BulletedList1.DataSource = ds;
        BulletedList1.DataBind();
        ArrayList myList1 = new ArrayList();
        ArrayList ds1 = (ArrayList)Session["state"];

        BulletedList2.DataSource = ds1;
        BulletedList2.DataBind();
        ArrayList myList2 = new ArrayList();
        ArrayList ds2 = (ArrayList)Session["city"];

        BulletedList3.DataSource = ds2;
        BulletedList3.DataBind();
    }
    protected void GridView1_SelectedIndexChanged(object sender, EventArgs
e)
    {
    }

    protected void SqlDataSource1_Selecting(object sender,
SqlDataSourceSelectingEventArgs e)
    {

    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs
e)
    {
        StringBuilder str = new StringBuilder();
        ArrayList MyArray = new ArrayList();
        // Select the checkboxes from the GridView control
        for (int i = 0; i < GridView1.Rows.Count; i++)
        {
            GridViewRow row = GridView1.Rows[i];
            bool isChecked = ((CheckBox)row.FindControl("chkSelect")).Checked;
    }
```

```

        if (isChecked)
        {
            MyArray.Add(GridView1.Rows[i].Cells[1].Text);
            // str.Append(GridView1.Rows[i].Cells[1].Text);
        }
    }
    Session["state"] = MyArray;
    string sr = Request.QueryString["country_id"].ToString();
    Response.Redirect("Stateform.aspx?country_id=" + sr);
}
protected void GridView1_SelectedIndexChanged1(object sender,
EventArgs e)
{
}

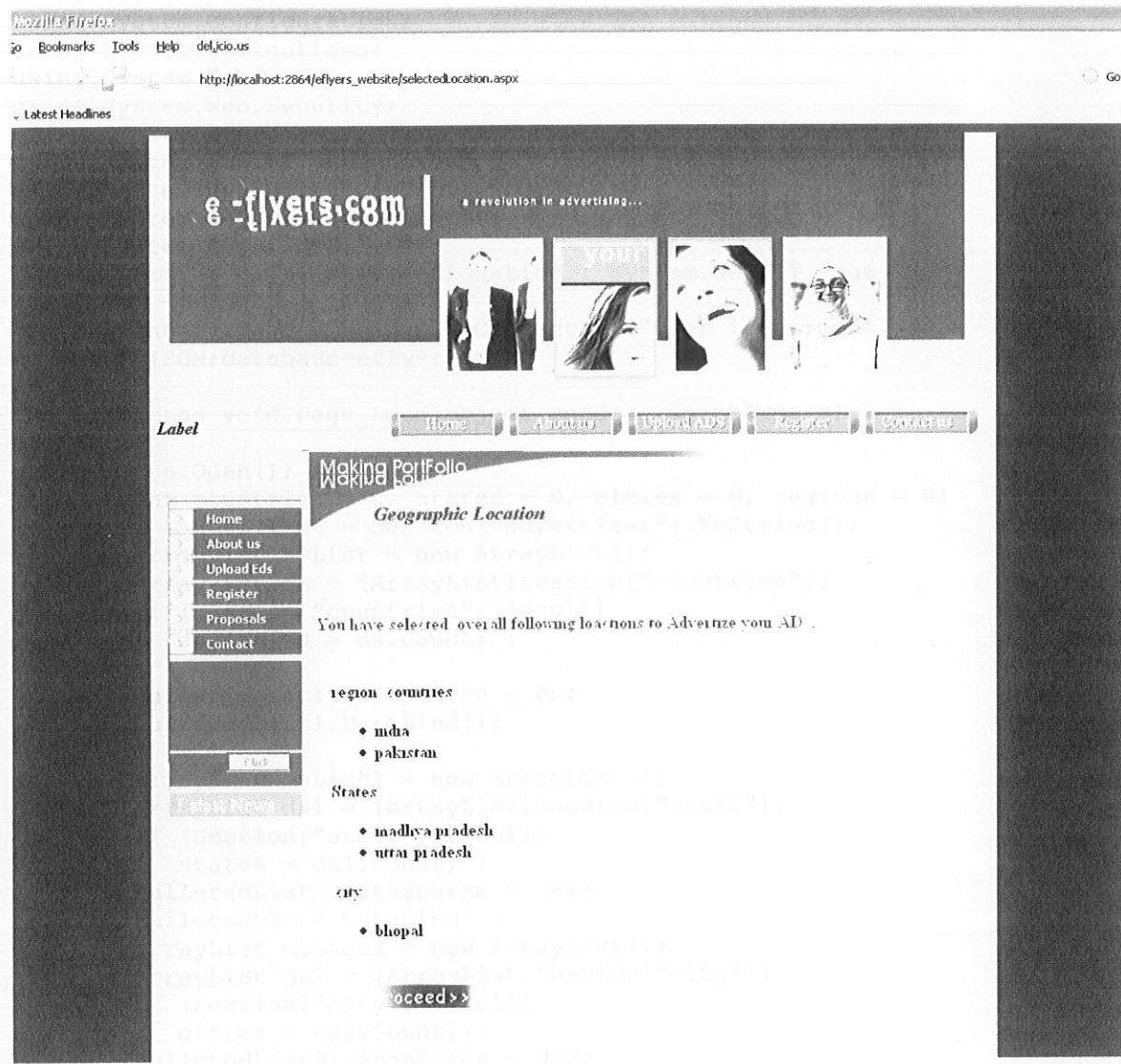
protected void ImageButton2_Click(object sender, ImageClickEventArgs
e)
{
    Response.Redirect("selectRegion.aspx");
}
}

```

Cityform.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;
using System.Text;
public partial class cityform : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");
    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();
        ArrayList myList = new ArrayList();
        ArrayList ds = (ArrayList)Session["countries"];
        BulletedList1.DataSource = ds;
        BulletedList1.DataBind();
        ArrayList myList1 = new ArrayList();
        ArrayList ds1 = (ArrayList)Session["state"];
        BulletedList2.DataSource = ds1;
        BulletedList2.DataBind();
        ArrayList myList2 = new ArrayList();
        ArrayList ds2 = (ArrayList)Session["city"];
        BulletedList3.DataSource = ds2;
        BulletedList3.DataBind();
    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        StringBuilder str = new StringBuilder();
        ArrayList MyArray = new ArrayList();
        // Select the checkboxes from the GridView control
        for (int i = 0; i < GridView1.Rows.Count; i++)
        {
            GridViewRow row = GridView1.Rows[i];
            bool isChecked =
((CheckBox)row.FindControl("chkSelect")).Checked;
            if (isChecked)
            {
                MyArray.Add(GridView1.Rows[i].Cells[1].Text);
            }
        }
        Session["city"] = MyArray;
        string sr = Request.QueryString["state_id"].ToString();
        Response.Redirect("cityform.aspx?state_id=" + sr);
    }
    protected void GridView1_SelectedIndexChanged1(object sender, EventArgs e)
    {
    }
    protected void SqlDataSource1_Selecting(object sender,
SqlDataSourceSelectingEventArgs e)
    {
    }
    protected void ImageButton2_Click(object sender, ImageClickEventArgs e)
    {
        Response.Redirect("selectRegion.aspx");
    }
}
```

SELECTED LOCATION



Selectedlocation.aspx

Selectedlocation.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;
public partial class selectedLocation : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");

    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();
        int countries = 0, states = 0, cities = 0, regions = 0;
        // Label1.Text = Session["advertiser"].ToString();
        ArrayList myList = new ArrayList();
        ArrayList ds = (ArrayList)Session["countries"];
        if (Session["countries"] != null)
        { countries = ds.Count; }

        BulletedList1.DataSource = ds;
        BulletedList1.DataBind();

        ArrayList myList1 = new ArrayList();
        ArrayList ds1 = (ArrayList)Session["state"];
        if (Session["state"] != null)
        { states = ds1.Count; }
        BulletedList2.DataSource = ds1;
        BulletedList2.DataBind();
        ArrayList myList2 = new ArrayList();
        ArrayList ds2 = (ArrayList)Session["city"];
        if (Session["city"] != null)
        { cities = ds2.Count; }
        BulletedList3.DataSource = ds2;
        BulletedList3.DataBind();

        ArrayList myList6 = new ArrayList();
        ArrayList ds5 = (ArrayList)Session["region"];
        if (Session["region"] != null)
        { regions = ds5.Count; }
        BulletedList5.DataSource = ds5;
        BulletedList5.DataBind();
        int[] myArray = { countries, states, cities, regions };
        Session["total"] = getMax(myArray);
    }
    private int getMax(int[] myArray)
    {
        int massimo = 0;
        foreach (int valore in myArray)
            if (valore > massimo)
                massimo = valore;
        return massimo;
    }
}
```

```

protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    int countries = 0, states = 0, cities = 0, regions = 0;
    // Label1.Text = Session["advertisor"].ToString();
    ArrayList myList = new ArrayList();
    ArrayList ds = (ArrayList)Session["countries"];
    if (Session["countries"] != null)
    { countries = ds.Count; }
    BulletedList1.DataSource = ds;
    BulletedList1.DataBind();
    ArrayList myList1 = new ArrayList();
    ArrayList ds1 = (ArrayList)Session["state"];
    if (Session["state"] != null)
    { states = ds1.Count; }
    BulletedList2.DataSource = ds1;
    BulletedList2.DataBind();
    ArrayList myList2 = new ArrayList();
    ArrayList ds2 = (ArrayList)Session["city"];
    if (Session["city"] != null)
    { cities = ds2.Count; }
    BulletedList3.DataSource = ds2;
    BulletedList3.DataBind();
    ArrayList myList6 = new ArrayList();
    ArrayList ds5 = (ArrayList)Session["region"];
    if (Session["region"] != null)
    { regions = ds5.Count; }
    BulletedList5.DataSource = ds5;
    BulletedList5.DataBind();
    int[] myArray = { countries, states, cities, regions };
    for (int i = 0; i < regions; i++)
    {
        SqlCommand com = new SqlCommand("insert into
trans_region(region,trans_id) values('" + ds5[i].ToString() + "','" +
Session["trans_id"].ToString() + "')", con);
        com.ExecuteNonQuery();
    }
    for (int j = 0; j < countries; j++)
    {
        SqlCommand com = new SqlCommand("insert into
trans_country(country,trans_id) values('" + ds[j].ToString() + "','" +
Session["trans_id"].ToString() + "')", con);
        com.ExecuteNonQuery();
    }
    for (int k = 0; k < states; k++)
    {
        SqlCommand com = new SqlCommand("insert into
trans_state(state,trans_id) values('" + ds1[k].ToString() + "','" +
Session["trans_id"].ToString() + "')", con);
        com.ExecuteNonQuery();
    }
    for (int l = 0; l < cities; l++)
    {
        SqlCommand com = new SqlCommand("insert into
trans_city(city,trans_id) values('" + ds2[l].ToString() + "','" +
Session["trans_id"].ToString() + "')", con);
        com.ExecuteNonQuery();
    }
    Response.Redirect("visitorPortfolio.aspx");
}
}

```

DESIRED PROFILE SELECTION (for target mode)

You can choose desired profile of audiences viewing your AD

Sex	Male
Age Range	10-18
Professions	
choose	
other	
doc	
♦ Engineer	
♦ Lawyer	
Interest	
choose	
other	
Hobbies	Add
view more	
Active Tags	
doctor medicine	
(used for searching your AD)	
Other Relevant Information	
Proceed >>	

http://localhost:2864 - Untitled Page ...

Engineer Lawyer Businessman
 Doctor

Done

[Visitorportfolio.aspx](#)

Visitorportfolio.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;

public partial class visitorPortfolio : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");
    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();
        string scriptString = "<script language=JavaScript> "
+"window.opener.document.forms(0).submit(); </script>";
        LinkButton1.Attributes.Add("onclick",
"window.open('professionlist.aspx',null,'height=250, width=350,status=
client window, resizable= no, scrollbars=no,
toolbar=no, left=400,menubar=no ');");
        if
(!Page.ClientScript.IsClientScriptBlockRegistered(scriptString))
        {
            Page.ClientScript.RegisterClientScriptBlock(this.GetType(),
"script", scriptString);
        }
        LinkButton2.Attributes.Add("onclick",
"window.open('interestlist.aspx',null,'height=250, width=350,status=
client window, resizable= no, scrollbars=no,
toolbar=no, left=400,menubar=no ');");
        ArrayList myList1 = new ArrayList();
        ArrayList ds1 = (ArrayList)Session["professions"];

        BulletedList1.DataSource = ds1;
        BulletedList1.DataBind();

        ArrayList myList2 = new ArrayList();
        ArrayList ds2 = (ArrayList)Session["Interest"];

        BulletedList2.DataSource = ds2;
        BulletedList2.DataBind();
    }
    protected void ImageButton1_Click(object sender, ImageClickEventArgs
e)
    {
        ArrayList myList1 = new ArrayList();
        ArrayList ds1 = (ArrayList)Session["professions"];

        BulletedList1.DataSource = ds1;
        BulletedList1.DataBind();

        ArrayList myList2 = new ArrayList();
        ArrayList ds2 = (ArrayList)Session["Interest"];
```

```

        BulletedList2.DataSource = ds2;
        BulletedList2.DataBind();
        string[] wrd = tag.Text.Split(' ');
        for (int i = 0; i < wrd.Length; i++)
        {
            SqlCommand com56 = new SqlCommand("insert into tagtable(tag)
values('" + wrd[i] + "')", con);
            com56.ExecuteNonQuery();

        }

        {
            SqlCommand com = new SqlCommand("insert into
transactiontable(trans_id,age,sex,info) values('" +
Session["trans_id"].ToString() + "','" + age.Text.ToString() + "','" +
sexrange.Text.ToString() + "','" + info.Text.ToString() + "')", con);
            com.ExecuteNonQuery();
        }

        Response.Redirect("MultipleFileUploadDemo.aspx");
    }
protected void LinkButton1_Click(object sender, EventArgs e)
{
    //Response.Redirect("professionlist.aspx");
}
protected void LinkButton2_Click(object sender, EventArgs e)
{
    //Response.Redirect("interestlist.aspx");
}
protected void BulletedList1_Click(object sender,
BulletedListEventArgs e)
{
}

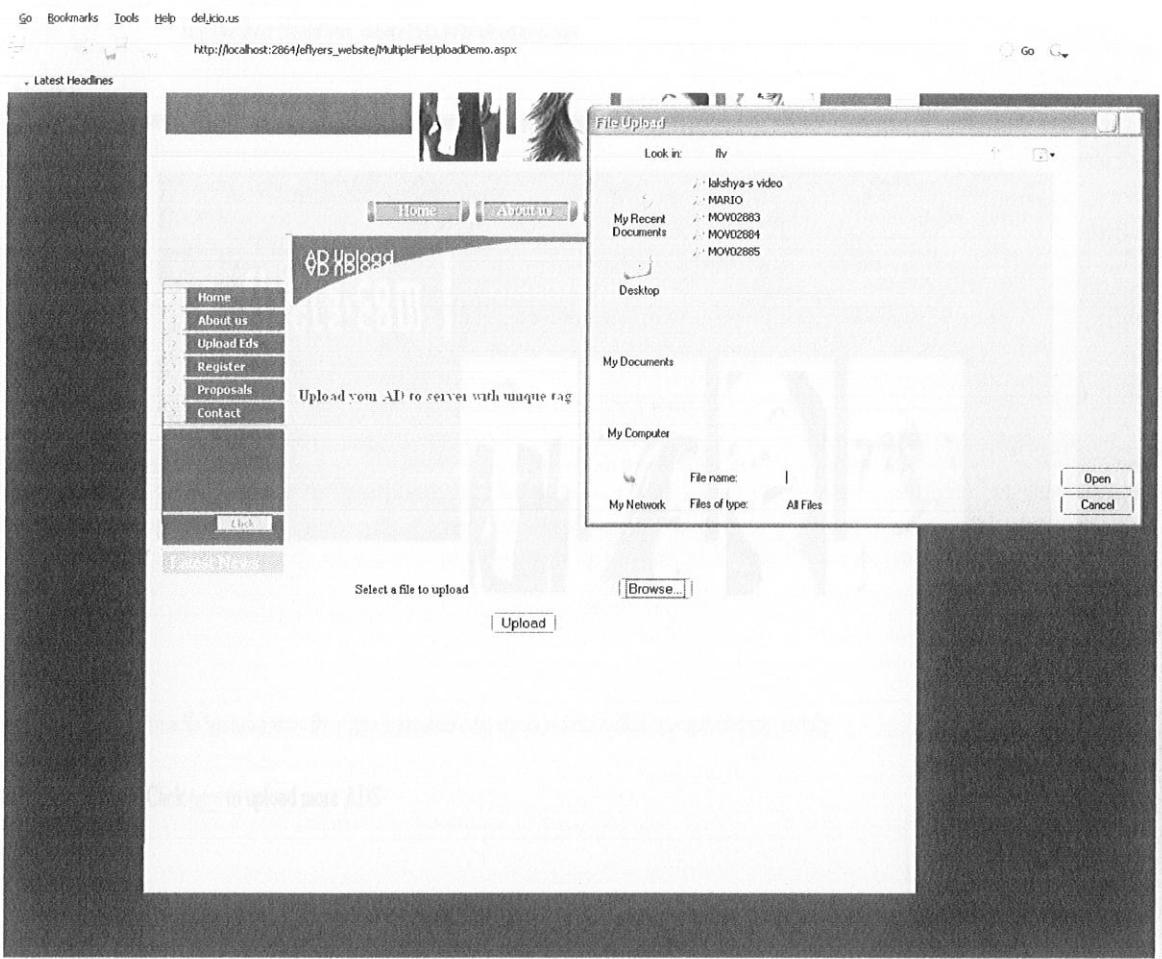
protected void TextBox1_TextChanged(object sender, EventArgs e)
{
}

protected void BulletedList2_Click(object sender,
BulletedListEventArgs e)
{
}

protected void TextBox4_TextChanged(object sender, EventArgs e)
{
}
}

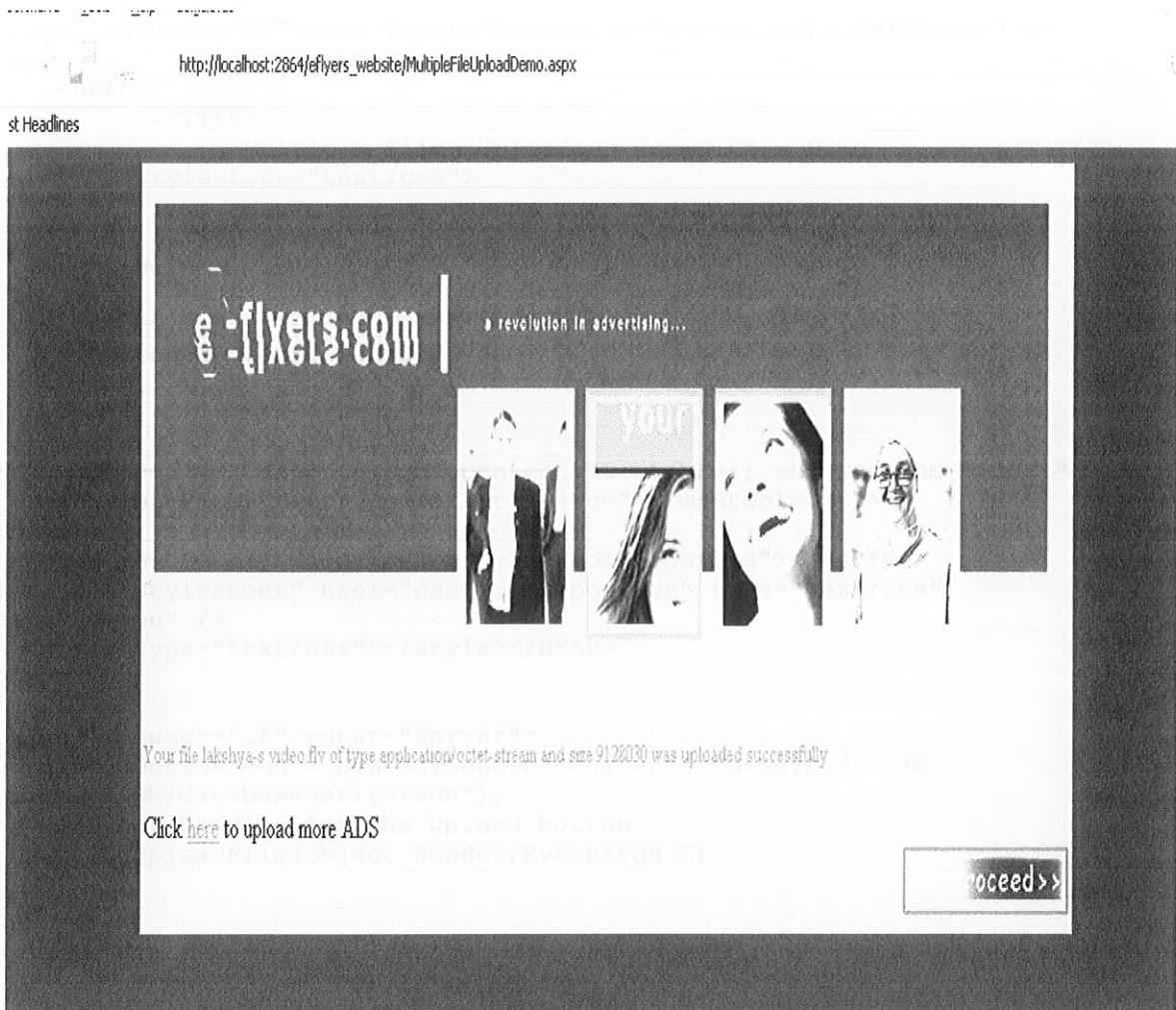
```

UPLOAD ADVERTISEMENT



[Multiplefileuploaddemo.aspx](#)

UPLOADED



Uploaded.aspx

Multipleupload.aspx.cs

```
<%@Page Language="C#" %><%@ Import Namespace="System.Data.SqlClient" %>
<HTML>
    <HEAD>
        <TITLE>
            Multiple Files Uploading Using C# - Demo
        </TITLE>
        <style type="text/css">
        .style1 {color: #003333;
            font-weight: bold;
        }
        body {
            background-color: #003333;
        }
    -->
        </style>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
        <script type="text/javascript" src="js/mootools.js">
    </script>
    <script type="text/javascript" src="js/moodalbox.js"></script>
    <link rel="stylesheet" href="css/moodalbox.css" type="text/css"
    media="screen" />
        <style type="text/css"></style></HEAD>
<BODY>

<script language="C#" runat="Server">
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
    source=PRISON;database=eflyersdb");
    //Event handler for the upload button
    void UploadFile(object Sender, EventArgs E)
    {

        con.Open();
        int IntLoop=0;
        Response.Write("<table width=\\"760\\" height=\\"400\\"
border=\\"0\\" align=\\"center\\" bgcolor=\\"#FFFFFF\\"><tr><td><object
classid=\\"clsid:D27CDB6E-AE6D-11cf-96B8-444553540000\\""
codebase=\\"http://download.macromedia.com/pub/shockwave/cabs/flash/swflash
.cab#version=7,0,19,0\\" width=\\"757\\" height=\\"250\\"><param name=\\"movie\\"
value=\\"../eflyers_website/upperface.swf\\" /><param name=\\"quality\\"
value=\\"high\\" /><embed src=\\"../eflyers_website/upperface.swf\\""
quality=\\"high\\""
pluginspage=\\"http://www.macromedia.com/go/getflashplayer\\""
type=\\"application/x-shockwave-flash\\" width=\\"757\\""
height=\\"250\\"></embed></object></td></tr><tr><td>");
        for(IntLoop=0;IntLoop<Request.Files.Count;IntLoop++)
        {
            if (Request.Files[IntLoop] !=null) //Checking for valid
file
            {
                // Since the FileName gives the entire path we use
                Substring function to rip of the filename.
                string StrFileName
                =Request.Files[IntLoop].FileName.Substring(Request.Files[IntLoop].FileName
                .LastIndexOf("\\") + 1) ;
                string StrFileType =
                Request.Files[IntLoop].ContentType ;
                int IntFileSize =Request.Files[IntLoop].ContentLength;
```

```

        //Checking for the file length. If length is 0
then file is not uploaded.
        if (StrFileType != "application/octet-stream")
            Response.Write(" <font color='Red' size='2'>Uploading
of file " + StrFileName + " failed due to wrong filetype </font><br>");
        else
        {
            if (IntFileSize <= 0)
                Response.Write(" <font color='Red'
size='2'>Uploading of file " + StrFileName + " failed. </font><br>");
            else
            {
                //Saving the file to the web server

Request.Files[IntLoop].SaveAs(Server.MapPath("./uploaded/" +
StrFileName));
                Session["filepath"] = "~/uploaded/" +
StrFileName.ToString();
                //Response.Write(Session["trans_id"]);
                //string ff = "258276238";

                SqlCommand com56 = new SqlCommand("UPDATE
transactiontable SET fileurl = '" + Session["filepath"].ToString() + "'"
WHERE trans_id = '" + Session["trans_id"].ToString() + "'", con);
                com56.ExecuteNonQuery();
                Response.Write("<font color='green' size='2'>Your
file " + StrFileName + " of type " + StrFileType + " and size " +
IntFileSize.ToString() + " was uploaded successfully.</font><br>");
            }
        }
    Response.Write("<br>Click <a href='MultipleFileUploadDemo.aspx'>
here</a> to upload more ADS <a href=\"PaymentOption.aspx\"><img
src=\"images/proceed.jpg\" width=\"131\" height=\"31\" align=\"right\" />
");
    Response.Write("</td> </tr></table>");
}
</script>
<%if(!Page.IsPostBack)
{
%>
<form id="FrmFileUploadDemo" name="FrmFileUploadDemo" method="post"
enctype="multipart/form-data" runat="server">
<TABLE align="center" bgcolor="lightyellow" cellspacing="5">
    <TR><TD>
<font size="2">Select a file to upload</font>
    <input type="file" id="File1" name="File1" runat="server">
    </TD>
    <TR>  <TR>

    <TD align="center"><asp:button value="Upload" Text="Upload"
runat="server" id="CmdUpload" onClick="UploadFile" />
    </TD>
    </TR>
</TABLE>
</form></td>
<td>&ampnbsp</td>
</tr>
</table></td>

```

ADVERTIZER VIEW LOG

Trans_id	Date	Clicks	Amount Remaining
15438	26/7/06	47	390.00
15324	27/9/05	56	453.00
16754	27/9/05	56	453.00
17525	27/9/05	56	453.00

Click on Transaction ids to get further information & make updation

NFO

Welcome - **advertiser@gmail.com**

Geographical location

Region **asia ,africa**

Country **India ,pakistan ,china**

State **M.P.,U.P.**

City **Vidisha ,Lucknow**

And desired Visitor profile

Sex **Male**

Age **18-26**

Mstatus **Unmarried**

clicks **76**

update

Advertiserviewlog.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;

public partial class advertiserlog : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        //int j;
        SqlDataReader dr6;
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        //Response.Write(Session["advertiser_id"]);

        string str = "select trans_id from transactiontable where
user_id=" + Session["advertiser_id"].ToString() + """;
        DataSet ds = new DataSet();
        SqlDataAdapter da = new SqlDataAdapter(str, con);
        da.Fill(ds);

        hyperlink.DataSource = ds;
        hyperlink.DataBind();
    }
}
```

Info.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;

public partial class info : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        //Response.Write();
        Label1.Text = Session["advertiser"].ToString();
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        string trans_id = Request.QueryString["trans_id"];
        Response.Write(trans_id);
        ArrayList region= new ArrayList();
        ArrayList country = new ArrayList();
        ArrayList city= new ArrayList();
        ArrayList state = new ArrayList();
        ArrayList sex = new ArrayList();
        ArrayList age = new ArrayList();
        ArrayList MStatus = new ArrayList();
        ArrayList broadcast = new ArrayList();
        SqlCommand Cmd = new SqlCommand("select distinct (region) from
trans_region where trans_id = '" + trans_id.ToString() + "'", con);
        SqlDataReader reader;
        reader = Cmd.ExecuteReader();
        while(reader.Read())
        {
            region.Add(reader["region"].ToString());
        }

        BulletedList1.DataSource = region;
        BulletedList1.DataBind();
        reader.Close();
        SqlCommand Cmdd = new SqlCommand("select distinct(country)
from trans_country where trans_id = '" + trans_id.ToString() + "'", con);

        SqlDataReader reader1;
        reader1 = Cmdd.ExecuteReader();
        while(reader1.Read())
        {
            country.Add(reader1["country"].ToString());
        }

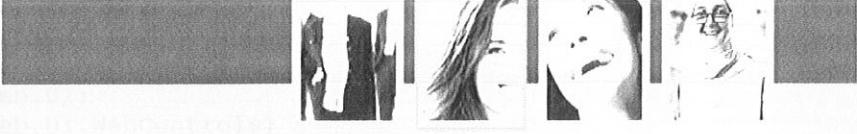
        BulletedList2.DataSource = country;
        BulletedList2.DataBind();
        reader1.Close();
    }
}
```

```
SqlCommand Cmddd = new SqlCommand("select distinct(state) from  
trans_state where trans_id = '" + trans_id.ToString() + "'", con);  
SqlDataReader reader2;  
reader2 = Cmddd.ExecuteReader();  
while (reader2.Read())  
{  
    state.Add(reader2["state"].ToString());}
```

USER MODULE

http://localhost:2864/eflyers_website/register.aspx

Latest Headlines



Home | About Us | Upload EDS | Register | Contact

Registration Form

Home
About us
Upload EDS
Register
Proposals
Contact

Email ID : Treated as your Username : sumit@yahoo.com

Password : *

Confirm Password : *

Sex : Male

Marital Status : Married

DOB : Dec 3 2007 December 2007
S M T W T F S
1
2 4 5 6 7 8
9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31

Country : india

State : bhopal

City : bhopal

Profession : Engineer

Any Specific Area Of Interest

Area of Interest

Hobbies

Read

Submit >

[Register.aspx](#)

Register.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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.Data.SqlClient;

public partial class register : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("user id=sa;pwd= ;data
source=PRISON;database=eflyersdb");
    protected void Page_Load(object sender, EventArgs e)
    {
        con.Open();
    }
    protected void RadioButtonList1_SelectedIndexChanged(object sender,
EventArgs e)
    {
    }

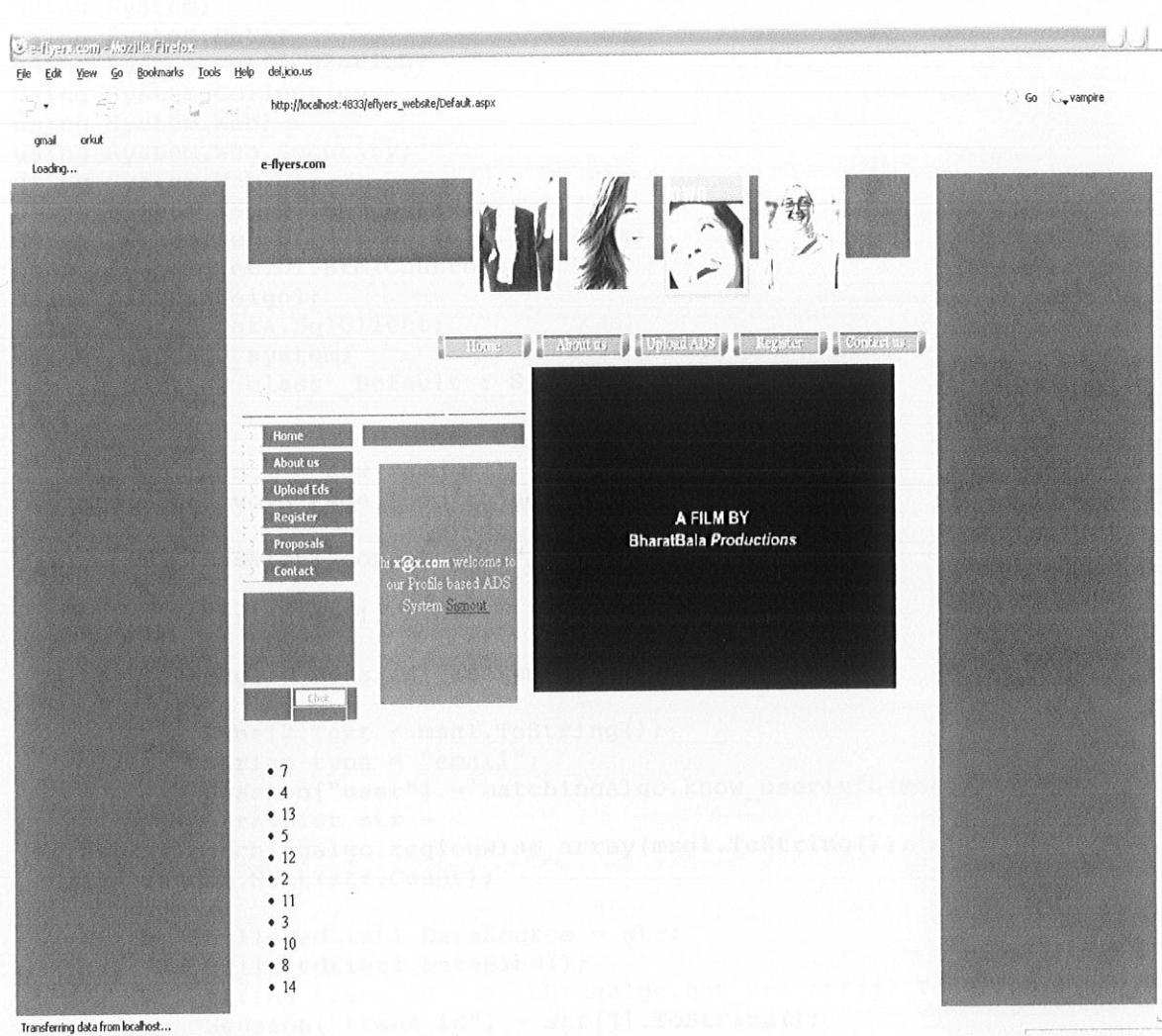
    protected void TextBox4_TextChanged(object sender, EventArgs e)
    {
    }

    protected void ImageButton1_Click(object sender, ImageClickEventArgs
e)
    {
        //Response.Write(sex.SelectedValue);

        SqlCommand com = new SqlCommand("insert into
usertable(email,passwd,conpasswd,sex,Mstatus,country,state,city,profession
) values('" + email.Text + "','" + passwd.Text + "','" + conpasswd.Text +
"', '" + sex.SelectedValue + "','" + Mstatus.SelectedValue + "','" + +
country.SelectedValue + "','" + state.Text + "','" + city.Text + "','" +
profession.SelectedValue + "')", con);

        com.ExecuteNonQuery();
        Response.Write("data inserted");
    }
    protected void country_SelectedIndexChanged(object sender, EventArgs e)
    {}
```

LIST OF TRANSACTION ID'S FOR USER THAT'S MATCH WITH HIS PROFILE



Default.aspx

Default.aspx.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
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 matchingalgo1;
using System.Data.SqlClient;
using response_system;
public partial class _Default : System.Web.UI.Page
{
    int j,i;
    Random random = new Random();
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["consumer_id"] != null)
        {
            string msg1;
            msg1 = Session["consumer_id"].ToString();
            Label2.Text = msg1.ToString();
            string type = "email";
            Session["user"] = matchingalgo.know userinfo(msg1, type);
            ArrayList str =
                (ArrayList)matchingalgo.regionwise_array(msg1.ToString());
            j = random.Next(str.Count);

            BulletedList1.DataSource = str;
            BulletedList1.DataBind();
            string trans_id = matchingalgo.get_url(str[j].ToString());
            Session["trans_id"] = str[j].ToString();

            Session["url"] = trans_id.ToString();
            FlashVideo1.VideoURL = Session["url"].ToString();
        }
    }

    protected void TextBox1_TextChanged(object sender, EventArgs e)
    {
    }

    protected void Flash_Click(object sender, EventArgs e)
    {
    }

    protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
    {
        string consumer_id;
        consumer_id =
            matchingalgo1.matchingalgo.UserAthunicate(TextBox1.Text.ToString(),
            TextBox2.Text.ToString());
    }
}
```

```

Session["consumer_id"] = constumer_id.ToString();
if (constumer_id != "1")
{
    //else
    //  msg1= Session["consumer_id"].ToString();

    Label1.Text = constumer_id.ToString();
    string type = "email";
    Session["user"] = matchingalgo.know userinfo(constumer_id,
type);
    ArrayList str =
(ArrayList)matchingalgo.regionwise_array(constumer_id.ToString());
    string trans_id;
    j = random.Next(str.Count);
    if (str.Count != 0)
    {
        trans_id = matchingalgo.get_url(str[j].ToString());
    }

    Session["url"] = trans_id.ToString();

    FlashVideo1.VideoURL = Session["url"].ToString();
    Session["login"] = "no";
    }
    else
    {
        FlashVideo1.VideoURL = "~/uploaded/school chale hum.flv";
    }

    BulletedList1.DataSource = str;
    BulletedList1.DataBind();
}
else
    Label2.Text = "You are InValid User";
}

protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.RemoveAll();
    Response.Redirect("Default.aspx");
}
protected void LinkButton2_Click(object sender, EventArgs e)
{
string consumer_id=Session["consumer_id"].ToString();
    string trans_id=Session["trans_id"].ToString();

response_system.response_system.insert_userlog(consumer_id,trans_id);
}
}

```

Matchingalgo.cs

```
using System;
using System.Data;
using System.Text;
using System.Configuration;
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.Data.SqlClient;
using System.Data.OleDb;
using System.Collections;
/// Summary description for Class1
namespace matchingalgo1
{
    public class matchingalgo
    {
        public matchingalgo()
        {
        }

        Random random = new Random();
        int j, i;

        public static void CreateMessageAlert(System.Web.UI.Page aspxPage,
string strMessage, string strKey)
        {
            string strScript = "<script language=JavaScript>alert('" +
strMessage + "')</script>";
            if ((! (aspxPage.IsStartupScriptRegistered(strKey))))
            {
                aspxPage.RegisterStartupScript(strKey, strScript);
            }
        }
        //user authentication
        public static string UserAthenticate(string email, string
password)
        {
            string SQLString = "SELECT * from usertable";
            string email1, password1, consumer_id1;
            string strcoc =
ConfigurationSettings.AppSettings["strcon"];
            string msg;

            SqlConnection conn1 = new SqlConnection(strcoc) ;
            SqlCommand Cmd = new SqlCommand(SQLString,conn1);
            bool check = false;
            SqlDataReader reader;

            try
            {
                conn1.Open();
                reader = Cmd.ExecuteReader();
            }
            catch { }
        }
    }
}
```

```

        while(reader.Read())
        {
            if((String.Compare((email.ToString()),(reader["email"].ToString())==0)&&
            (String.Compare((password.ToString()),(reader["passwd"].ToString()))==0))
            {
                check=true;
                email1= reader["email"].ToString();
                password1 =
reader["passwd"].ToString();
                consumer_id1=reader["consumer_id"].ToString();
                //return consumer_id1.ToString();
                break;
            }
        }

        if (check == true)
        {
            msg = reader["consumer_id"].ToString();
            // msg = "hai authorized";
            return msg;
        }

//Response.Redirect("~/Admin/AdministrationArea.aspx");
        else
        {
            msg = "1";
            return msg;
        }

    }
    catch(Exception err)
    {
        msg = err.Message.ToString() ;
    return msg;
    }
    finally
    {
        conn1.Close();
    }
}
//another authentication
public static int UserAuthunicate1(string email, string password)
{
    string SQLString = "SELECT * from usertable";
    string email1, password1, consumer_id1;
    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    int msg;

    SqlConnection conn1 = new SqlConnection(strcoc);
    SqlCommand Cmd = new SqlCommand(SQLString, conn1);
    bool check = false;
}

```

```

SqlDataReader reader;

try
{
    conn1.Open();
    reader = Cmd.ExecuteReader();

    while (reader.Read())
    {
        if ((String.Compare((email.ToString()),
(reader["email"].ToString()) == 0) )
        {
            check = true;
            break;
        }

    }

    if (check == true)
    {
        msg = 0;
        // msg = "hai authorized";
        return msg;
    }
}

//Response.Redirect("~/Admin/AdministrationArea.aspx");
else
{
    msg = 1;
    return msg;
}

}

catch (Exception err)
{
    msg = 2;
    return msg;
}
finally
{
    conn1.Close();
}
}

//regionwise array
public static ArrayList regionwise_array(string consumer_id)
{
    ArrayList geographicalarray =
geographical_array(consumer_id);
    ArrayList broadcastarray =
broadcast_array(geographicalarray,consumer_id);
    ArrayList targetarray = target_array(geographicalarray,
consumer_id);
}

```

```

        ArrayList interestarray =
interest_array(targetarray, consumer_id);
        ArrayList professionarray = profession_array(targetarray,
consumer_id);
        ArrayList sexarray = sex_array(targetarray, consumer_id);
        ArrayList Mstatusarray = Mstatus_array(targetarray,
consumer_id);
        ArrayList final = join_array(broadcastarray,
interestarray, professionarray, sexarray, Mstatusarray);
        return final;
    }

    // broadcastarray
    public static ArrayList broadcast_array(ArrayList
geographical_array, string consumer_id)
{
    //return regionwise_array;

    ArrayList broadcast_array = new ArrayList();

    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    SqlConnection con = new SqlConnection(strcoc);

    con.Open();

    foreach (string transaction_id in geographical_array)
    {
        char l='1';
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select trans_id from
transactiontable where trans_id='" + transaction_id + "' and broadcast
=''" + l + "' ", con);

        dr6 = com.ExecuteReader();

        while (dr6.Read())
        {
            broadcast_array.Add(dr6[0]);
        }
        dr6.Close();
    }

    return broadcast_array;
}

}

```

```

// targetarray
public static ArrayList target_array(ArrayList geographical_array,
string consumer_id)
{
}

ArrayList target_array = new ArrayList();

string strcoc = ConfigurationSettings.AppSettings["strcon"];
SqlConnection con = new SqlConnection(strcoc);

char l = '0';

con.Open();

foreach (string transaction_id in geographical_array)
{
    SqlDataReader dr6;
    SqlCommand com = new SqlCommand("select trans_id from
transactiontable where trans_id=" + transaction_id + " and broadcast
= '" + l + "' ", con);

    dr6 = com.ExecuteReader();

    while (dr6.Read())
    {
        target_array.Add(dr6[0]);
    }

    dr6.Close();
}
return target_array;

con.Close();
}

}

//join and sort final arry
//,ArrayList target_array
public static ArrayList join_array(ArrayList broadcastarray,
ArrayList interestarray, ArrayList professionarray, ArrayList sexarray,
ArrayList Mstatusarray)
{
ArrayList final_array= new ArrayList();
    final_array.InsertRange(final_array.Count, broadcastarray);
    final_array.InsertRange(final_array.Count, interestarray);
    final_array.InsertRange(final_array.Count, professionarray);
    final_array.InsertRange(final_array.Count, sexarray);
    final_array.InsertRange(final_array.Count, Mstatusarray);

    final_array.Sort();
    return final_array;
}

```

```

        }
    //get url
    public static string get_url(string trans_id)
    {
        string msg;
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select * from
transactiontable where trans_id='" + trans_id.ToString() + "' ", con);
        string url;
        con.Open();
        dr6 = com.ExecuteReader();
        if (dr6.Read() == true)
        {
            url = dr6["fileurl"].ToString();
            return url.ToString();
        }
        else
        {
            msg = "no url";
            return msg;
        }

        dr6.Close();
        con.Close();
    }
}

//geographical array
public static ArrayList geographical_array(string consumer_id)
{
    ArrayList geographical_array1 = new ArrayList();
    string type = "state", msg;
    string state = know_userinfo(consumer_id, type);
    type = "city";
    string city = know_userinfo(consumer_id, type);
    type = "country";
    string country1 = know_userinfo(consumer_id, type);
    string region = know_region(consumer_id);

    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    SqlConnection con = new SqlConnection(strcoc);

    SqlDataReader dr6;

    con.Open();

    SqlCommand com = new SqlCommand("select distinct
transactiontable.trans_id from transactiontable,
trans_region,trans_country,trans_city,trans_state where
((trans_region.region=''" + region.ToString() + "' or
trans_region.region='null') and ( trans_country.country=''" +
country1.ToString() + "' or trans_country.country ='null') and
(trans_city.city=''" + city.ToString() + "' or trans_city.city ='null') and

```

```

        ( trans_state.state='"' + state.ToString() + "' or
trans_state.state='null') ) and
(transactiontable.trans_id=trans_region.trans_id or
transactiontable.trans_id=trans_country.trans_id or
transactiontable.trans_id=trans_state.trans_id or
transactiontable.trans_id=trans_city.trans_id)", con);

        dr6 = com.ExecuteReader();

        if (dr6.Read())
        {
            while (dr6.Read())
            {
                geographical_array1.Add(dr6[0]);
            }

            //}
            return geographical_array1;
        }
        else
        {
            geographical_array1.Add("no broad cast   ");
            return geographical_array1;
        }

        dr6.Close();
    //}
    con.Close();
}

}

}

//interest wise array
public static ArrayList interest_array(ArrayList targetarray,
string consumer_id)
{
    ArrayList interest = know_userinterest(consumer_id);
    ArrayList interestwise_array = new ArrayList();

    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    SqlConnection con = new SqlConnection(strcoc);

    con.Open();
    foreach (string interest_id in interest)
        foreach (string transaction_id in targetarray)
    {
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select trans_id
from trans_intereststable where interest_id ='" + interest_id.ToString() +
"' and trans_id=''" + transaction_id + "' ", con);
}

```

```

        dr6 = com.ExecuteReader();

        while (dr6.Read())
        {
            interestwise_array.Add(dr6[0]);
        }
        dr6.Close();
    }

    return interestwise_array;
}

con.Close();

}

public static ArrayList profession_array(ArrayList targetarray,
string consumer_id)
{
}

string type = "profession";
string profession = know_userinfo(consumer_id, type);
ArrayList professionwise_array = new ArrayList();

string strcoc = ConfigurationSettings.AppSettings["strcon"];
SqlConnection con = new SqlConnection(strcoc);

con.Open();

foreach (string transaction_id in targetarray)
{
    SqlDataReader dr6;
    SqlCommand com = new SqlCommand("select
trans_professiontable.trans_id from profession , trans_professiontable
where trans_professiontable.trans_id=' " + transaction_id + "' and
profession.profession ='" + profession.ToString() + "' and
trans_professiontable.profession_id=profession.pro_id ", con);

    dr6 = com.ExecuteReader();

    //if (dr6.Read())
    //{
    //while (dr6.Read())
    //{
        professionwise_array.Add(dr6[0]);
    }
    dr6.Close();
}

return professionwise_array;
}

con.Close();

```

```

}

//sexsrray

    public static ArrayList sex_array(ArrayList targetarray, string
consumer_id)
{
    //return regionwise_array;
    string type = "sex"; ;
    string sex = know_userinfo(consumer_id, type);
    ArrayList sex_array = new ArrayList();

    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    SqlConnection con = new SqlConnection(strcoc);

    con.Open();

    foreach (string transaction_id in targetarray)
    {
        //regionwise_array.Add(transaction_id);

        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select trans_id from
transactiontable where trans_id='"
+ transaction_id + "' and sex ='"
+ sex.ToString() + "'", con);

        dr6 = com.ExecuteReader();

        while (dr6.Read())
        {
            sex_array.Add(dr6[0]);
        }
        dr6.Close();
    }

    return sex_array;
}

con.Close();
}

```



```

        string countryl = know userinfo(consumer_id, type);
        string msg;
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select regiontable.region
from regiontable,countrytable where countrytable.country='"
+ countryl.ToString() + "' and countrytable.region_id=regiontable.region_id
", con);
        string consumer_region;
        con.Open();
        dr6 = com.ExecuteReader();
        if (dr6.Read() == true)
        {
            consumer_region = dr6[0].ToString();
            return consumer_region.ToString();
        }
        else
        {
            msg = "no region";
            return msg;
        }
        dr6.Close();
        con.Close();
    }

//get user info
    public static string know userinfo(string consumer_id, string
type)
    {
        string msg;
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select * from usertable where
consumer_id='"
+ consumer_id.ToString() + "' ", con);
        string consumer_country;
        con.Open();
        dr6 = com.ExecuteReader();
        if (dr6.Read() == true)
        {
            consumer_country = dr6[type].ToString();
            return consumer_country.ToString();
        }
        else
        {
            msg = "no country";
            return msg;
        }
        dr6.Close();
        con.Close();
    }

//get user interest
    private static ArrayList know userinterest(string consumer_id)
    {
        ArrayList interest = new ArrayList();
        string msg;
        string strcoc = ConfigurationSettings.AppSettings["strcon"];

```

```

        SqlConnection con = new SqlConnection(strcoc);
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select * from
userintereststable where consumer_id=' " + consumer_id.ToString() + "'",
con);
        string consumer_country;
        con.Open();
        dr6 = com.ExecuteReader();
        while (dr6.Read())
        {
            interest.Add(dr6["interest_id"]);
        }
        return interest;
    }

    //insert integer value in temp database
    public static string videocounter(string consumer_id)
    {
        string j="lakshya";
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        return j;
        con.Close();
    }
    public static void deletevideocounter(string consumer_id)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        SqlCommand com56 = new SqlCommand("delete from videocounter
where consumer_id=' " + consumer_id.ToString() + "' ", con);
        com56.ExecuteNonQuery();

        con.Close();
    }
    public static string showvideocounter(string consumer_id)
    {
        string msg;
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        SqlDataReader dr6;

        SqlCommand com = new SqlCommand("select inte from videocounter
", con);

        dr6 = com.ExecuteReader();
        if (dr6.Read() == true)
        {

```

```

        msg = dr6[0].ToString();
        return msg.ToString();

    }
    else
    {
        msg = "koi id nahi hai ";
        return msg.ToString();
    }
    dr6.Close();
    con.Close();
}
//advertiser authentication
public static string advertiserAthenticate(string email, string password)
{
    string SQLString = "SELECT * from advertiser";
    string email1, password1, advertiser_id1;
    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    string msg;
    SqlConnection conn1 = new SqlConnection(strcoc);
    SqlCommand Cmd = new SqlCommand(SQLString, conn1);
    bool check = false;
    SqlDataReader reader;
    try
    {
        conn1.Open();
        reader = Cmd.ExecuteReader();
        while (reader.Read())
        {
            if ((String.Compare((email.ToString()),
(reader["email"].ToString())) == 0) &&
(String.Compare((password.ToString()), (reader["password"].ToString())) == 0))
            {
                check = true;
                break;
            }
        }
        if (check == true)
        {
            msg = reader["user_id"].ToString();
            return msg;
        }
        else
        {
            msg = "1";
            return msg;
        }
    }
    catch (Exception err)
    {
        msg = err.Message.ToString();
        return msg;
    }
    finally
    {
        conn1.Close();
    }
}
}

```

Response_system.cs

```
using System;
using System.Data;
using System.Text;
using System.Configuration;
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.Data.SqlClient;
using System.Data.OleDb;
using System.Collections;
/// Summary description for response_system
namespace response_system
{
    public class response_system
    {
        public response_system()
        {
            //
            // TODO: Add constructor logic here
            //
        }
        public static void insert_userlog(string consumer_id, string trans_id)
        {
            string strcoc = ConfigurationSettings.AppSettings["strcon"];
            SqlConnection con = new SqlConnection(strcoc);
            con.Open();
            SqlCommand com = new SqlCommand("insert into user_logtable(consumer_id,trans_id) values('" + consumer_id.ToString() +
            "','" + trans_id.ToString() + "')", con);
            com.ExecuteNonQuery();

            ArrayList trans_interest = new ArrayList();
            SqlDataReader dr6;

            SqlCommand comm = new SqlCommand("select * from trans_interesttable where trans_id ='" + trans_id.ToString() + "' ", con);

            dr6 = comm.ExecuteReader();
            while (dr6.Read())
            {
                if(dr6["interest_id"]!=null)
                    trans_interest.Add(dr6["interest_id"]);
            }
            updateuserinterest(trans_interest, consumer_id);
            topfiveinterest(consumer_id);

            dr6.Close();
        }
    }
}
```

```

        con.Close();
    }
    public static void updateuserinterest(ArrayList trans_interest,
string consumer_id)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        bool check = false;
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select * from
userinteresttable where consumer_id='"+consumer_id.ToString()+"' ", con);
        int click ;
        dr6 = com.ExecuteReader();
        foreach (string interestid in trans_interest)
        {
            while (dr6.Read())
            {

                if
((String.Compare((interestid.ToString()), (dr6["interest_id"].ToString())))
== 0))

                    {
                        check = true;
                        break;
                    }
                else
                {
                    click =
Convert.ToInt32(dr6["click"].ToString());
                    click++;
                    insert_click(interestid, consumer_id,
click);
                }
            }
            else
            {
                insert_interest(interestid, consumer_id);
            }
        }
        dr6.Close();
        con.Close();
    }
    public static void insert_click(string interest_id,string
consumer_id,int click)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();

```

```

        SqlCommand com = new SqlCommand("update userinteresttable set
click = '" + click + "' where consumer_id='" + consumer_id.ToString() + "
'" + interest_id + "' + interest_id.ToString() + "'", con);
        com.ExecuteNonQuery();
        con.Close();
    }
    public static void insert_interest(string interest_id, string
consumer_id)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        int click=1;
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        SqlCommand com = new SqlCommand("insert into
userinteresttable(consumer_id,interest_id,click) values('" +
consumer_id.ToString() + "','" + interest_id.ToString() + "','" + click +
"') ", con);
        com.ExecuteNonQuery();
        con.Close();
    }
    public static void topfiveinterest(string consumer_id)
    {
        string strcoc =
ConfigurationSettings.AppSettings["strcon"];
        ArrayList fiveinterest = new ArrayList();
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        SqlDataReader dr6;
        SqlCommand com = new SqlCommand("select distinct
interest_id,click from userinteresttable where consumer_id='" +
consumer_id.ToString() + "' order by click desc ", con);
        dr6 = com.ExecuteReader();
        while (dr6.Read())
        {
            fiveinterest.Add(dr6["interest_id"]);
        }
        update_userbehaviortable(fiveinterest, consumer_id);
        con.Close();
    }
    public static void update_userbehaviortable(ArrayList
fiveinterest, string consumer_id)
    {
        string strcoc = ConfigurationSettings.AppSettings["strcon"];
        int click = 1;
        SqlConnection con = new SqlConnection(strcoc);
        con.Open();
        SqlCommand com = new SqlCommand("update userbehaviortable set
int1 = '" + fiveinterest[0] + "',int2=''" + fiveinterest[1] + "',int3=''" +
fiveinterest[2] + "',int4=''" + fiveinterest[3] + "',int5=''" +
fiveinterest[4] + "' where consumer_id='" + consumer_id.ToString() + "'",
con);
        com.ExecuteNonQuery();
        con.Close();
    }
}

```

```

//short term interest update
public static ArrayList short_term_interest(string consumer_id)
{
    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    string msg;
    SqlConnection con = new SqlConnection(strcoc);
    con.Open();
    ArrayList trans_id1 = new ArrayList();
    DateTime theTime = System.DateTime.Now;

    // Create a time span object for 2weeks
    System.TimeSpan timespan = new TimeSpan(14, 0, 0, 0);

    // Subtract the span from the time
    theTime = theTime.Subtract(timespan);
    SqlDataReader dr2;
    SqlCommand cov = new SqlCommand("select trans_id from
user_logtable where consumer_id ='" + consumer_id.ToString() + "' ", con);
    //string consumer_country;

    dr2 = cov.ExecuteReader();
    while (dr2.Read())
    {
        trans_id1.Add(dr2["trans_id"]);
    }
    ArrayList trans_interest = new ArrayList();

    SqlDataReader dr6;
    foreach (string trans_id in trans_id1)
    {
        SqlCommand comm = new SqlCommand("select * from
trans_interesttable where trans_id ='" + trans_id.ToString() + "' ", con);
        //string consumer_country;
        dr6 = comm.ExecuteReader();
        while (dr6.Read())
        {
            if (dr6["interest_id"] != null)
                trans_interest.Add(dr6["interest_id"]);
        }
        updateuserinterest1(trans_interest, consumer_id);
        topfiveinterest1(consumer_id);
    }

    dr6.Close();dr2.Close();
    con.Close();
}
}

public static void updateuserinterest(ArrayList trans_interest,
string consumer_id)
{
    string strcoc = ConfigurationSettings.AppSettings["strcon"];
    SqlConnection con = new SqlConnection(strcoc);
    con.Open();
    bool check = false;
    SqlDataReader dr6;
    SqlCommand com = new SqlCommand("select * from
userinteresttable where consumer_id='"+consumer_id.ToString()+"' ", con);
    int click ;
    dr6 = com.ExecuteReader();
    foreach (string interestid in trans_interest)
    {
}

```

```

        while (dr6.Read())
        {
            if
((String.Compare((interestid.ToString()), (dr6["interest_id"].ToString())))
== 0))

            {
                check = true;
                break;
            }
        }

        if (check == true)
        {
            click =
Convert.ToInt32(dr6["click"].ToString());
//click = dr6["click"];
            click++;
            insert_click(interestid, consumer_id, click);

        }
        else
        {
            insert_interest(interestid, consumer_id);
        }
    }
    dr6.Close();
    con.Close();
}
}

public static void topfiveinterest(string consumer_id)
{
    string strcoc =
ConfigurationSettings.AppSettings["strcon"];
    ArrayList fiveinterest = new ArrayList();
    SqlConnection con = new SqlConnection(strcoc);
    con.Open();
    SqlDataReader dr6;
    SqlCommand com = new SqlCommand("select distinct
interest_id,click from userinteresttable where consumer_id='"
+ consumer_id.ToString() + "' order by click desc ", con);
    dr6 = com.ExecuteReader();
    while (dr6.Read())
    {
        fiveinterest.Add(dr6["interest_id"]);
    }
    update_userbehaviortable(fiveinterest, consumer_id);
    con.Close();
}
}

```

CONCLUSION AND FUTURE WORK

- Understanding the information search process within a tourism domain-specific system.
- We have developed a unique system which is beneficial for User, Advertiser and Publisher. This system provides an opportunity to even small merchants to use internet as a medium to reach their targeted customers. We have currently developed the prototype. For the same and we plan to continue this work to build an efficient system in all respects.

(Anil Kumar, Naveen, Jason Rennie, Christie Heyneke)

We plan a platform synchronous with the web platform for publishing advertisements on the mobile. Similar to the web based system here also the profile of the user would be created and in addition to that his geographical location would be gathered, so that the advertisements of the establishments near his geographical location could be published. This system is currently not in use and it would be a new feature in advertising.

- Profile-Based Object Matching For Information Integration
(Antal Duau, Yoonkyung Lee, Jewel Han, University of Illinois)

REFERENCES

- Understanding the information search process within a tourism domain-specific search engine.
(Nicole Mitsche)
- A Machine Learning Approach to Building Domain-Specific Search Engines
Andrew McCallum
(Kamal Nigam, Jason Rennie, Kristie Seymore)
- A Framework For Domain-Specific Search Engine: Design Pattern Perspective
(Junlin Zhang, Weirnin Qu, Lin Du, Yufang Sun)
- Interestingness of association rules in data mining: Issues relevant to e-commerce
(Rajesh Natarajan and B. Shekhar)
- Profile-Based Object Matching For Information Integration
(AnHai Doan, Yoonkyong Lee, Jiawei Han, University of Illinois)

BIBLIOGRAPHY

- Black Book ASP.NET
(By Belmarks Solution Content Team)
- Data Mining – Concept and Techniques
(Jiawei Han, Micheline Kamber)
- Fundamentals Of Database Systems
(Ramez Elmasri, Shamkant B. Navathe)