



CHAT SERVER CREATION

By

P.CHITRA DEVI

Reg. No 71202702002

of

Kumaraguru College of Technology

A PROJECT REPORT

Submitted to the

FACULTY OF SCIENCE AND HUMANITIES

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF SCIENCE

IN

APPLIED SCIENCE-COMPUTER TECHNOLOGY

June, 2004



BONAFIDE CERTIFICATE

Certified that this project report titled

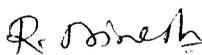
CHAT SERVER CREATION

IS A BONAFIDE WORK OF

Ms. P.CHITRA DEVI (Reg. No: 71202702002)

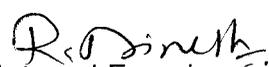
Who carried out the research under my supervision.

Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

for 
Project Guide (15-6-2004)


Head of the Department

The Candidate with University Register No. 71202702002 was examined by us in the project Viva-Voce examination held on 17-06-2004.


Internal Examiner
(17-6-2004)


External Examiner

TO WHOMSOEVER IT MAY CONCERN

It is to certify that Ms.ChitraDevi.P (Reg.No.71202702002) final year B.Sc [Computer Technology] , Kumaraguru College of Technology has successfully completed her project entitled" CHAT SERVER CREATION" using Asp.net , Vb.net and Sql Server 7.0. in our concern during the period from Nov 2003 to May 2004.

We wish her all the best for future endeavors.

For NewWalk Technologies Ltd.,



Authorized Signatory.

ABSTRACT

Chat Server is a collection of private computer networks within an organization with access to communication tools designed to enable easy information sharing within workgroups. The essential idea of an intranet is that it uses extended LAN technologies to facilitate communication between people and improve the knowledge base of an organization's employees. This project intranet chat server makes possible the connection, validation and administration of Client users using separated chat channels and Internet networks at same time.

The chat server has the characteristic features of sharing documents, Off-line messaging support, Users authentication by name, password and channel, Delivery of messages, notifications or web pages to other users, schedule meeting and share calendars with team members, have discussion regarding important issues in organization, send and receive emails from your intranet, send reoccurring email memos to ensure employee interaction, and multimedia.

In this project the administrator has the total control over the information of the chatting server. Administrator only has the authority to create clients account, and deny him if foul language is used during chatting. He has the control over the amount of connected users in chat server .The clients can log on into

different chatting categories like personal, official, information, meetings and group discussion. Public and private rooms are available for clients to chat efficiently.

The server authenticates the clients by the username and password and channel. A database is maintained to store all clients information like their names, address and division. Information about the client logged in and the time they logged out is also maintained as a report. Reports can be generated based on this, as they can be used for the customer traffic analysis and the advertisement can also be enhanced.

ACKNOWLEDGEMENT

I hereby take this opportunity to express my sincere gratitude to the management of Kumaraguru College Of Technology, Coimbatore for giving me an opportunity to study in this esteemed institution.

I deem it a great privilege to express my profuse thanks to our beloved principal **Dr. K. K. Padmanaban, PhD** for providing all facilities during the course of study.

I acknowledge with thanks to our Head Of The Department Of Computer Science and Engineering **Dr. S. Thangasamy, PhD** who has been a bastion of moral strength and a source of incessant encouragement to me.

I am greatly indebted to my esteemed project coordinator Mr. **Ranganathan Dinesh, M.S** who has guided me with his constructive and innovative ideas.

I express my sincere thanks to my beloved parents and friends and all other teaching and non-teaching staff members of Computer Science Department, who went out of their way and took pains to help us whenever needed.

LIST OF TABLES

	PAGE NO
• Table 4.1 - REGISTRATION	15
• Table 4.2 - MAPPING FILE FORMAT	19
• Table 4.3 - MAPPING DATA FORMAT	19
• Table 4.4 - MAP GROUP FILE FORMAT	21

LIST OF FIGURES

• Figure 4.1 - DATA FLOW DIAGRAM	14
----------------------------------	----

TABLE OF CONTENTS

ABSTRACT	iii
LIST OF TABLES	v
LIST OF FIGURES	v
Chapter 1 INTRODUCTION	1
1.1 ORGANIZATIONAL PROFILE	1
1.2 PROJECT OVERVIEW	2
Chapter 2 SYSTEM ANALYSIS	3
2.1 PROPOSED SYSTEM	3
Chapter 3 SYSTEM SPECIFICATION	4
3.1 USER SPECIFICATION	4
3.2 HARDWARE SPECIFICATION	5
3.3 SOFTWARE SPECIFICATION	6
3.4 ABOUT THE SOFTWARE	7

Chapter 4 SYSTEM DESIGN	11
4.1 MODULE DESIGN	11
4.2 DATAFLOW DIAGRAM	14
4.3 DATABASE DESIGN	15
4.4 INPUT DESIGN	16
4.5 OUTPUT DESIGN	17
4.6 FILE DESIGN	18
4.7 PRELIMINARY DESIGN	22
Chapter 5 SYSTEM TESTING	23
5.1 UNIT TESTING	24
5.2 INTEGRATED TESTING	24
5.3 VALIDATION TESTING	24
5.4 OUTPUT TESTING	25
5.5 SYSTEM TESTING	25
5.6 SPECIFICATION TESTING	25
Chapter 6 IMPLEMENTATION	27
Chapter 7 ENHANCEMENTS	30
CONCLUSIONS	31
APPENDICES	
REFERENCES	32

CHAPTER 1

INTRODUCTION

1.1 ORGANISATION PROFILE

New walk Technologies is a group of well experienced and highly technocrat's professionals in the field of information technology and business management. As a professionally managed conglomerate, developing a diversified range of products, New walk Technology is evolving into a global organization that benchmark world class practices.

We are committed to achieve a sustained growth in information technology services. Learning past experiences and adopting latest technologies innovations, we implement cost effective and value added solutions to our clients to suit their business requirements. Honesty, hard work and dedication to the cause are our hallmarks.

Redefining the norm of software products and development practices, New walk Technology is advancing at a fast pace thereby achieving higher growth rate every year. We assure to you the best quality and professional standards for each service that we provide. It has trained many professionals in the area of web designing and networking. The training division gives us an opportunity to meet variety of candidates with various skill sets. The training matches latest trends and make the students to compete in the IT Industry.

The services provided by them are Application development, Internet solutions, Website design, Webcontent development and integration, Web

1.2 PROJECT OVERVIEW

Chatting is one of the most popular activities on the Internet and involves people from all walks of life and just about all ages to coming together in areas where they can join in on in a variety of topics that they are interested in with people all around the world. Traditionally chat is text based but can also involve audio and video. In public chat areas people use "user names" or "nicknames" or sometimes represent themselves as graphical icons or "avatars" as they are known in the chat world.

While real time communication can be engaging and educationally useful, we strongly suggest it be used with students in a controlled environment using programs that enable direct links with specific parties rather than joining these public areas. In all cases where students are going to communicate on-line they should be aware of Netiquette and My Rules For Online Safety.

There are two main types of Internet chats. One is the web based chat room. These are graphical where people can choose images to represent themselves and offer elaborate virtual environments. They are not real time, messages will come up on your screen if you press the refresh or reload button. Some of the latest web based chats use server push technology or java but in this project we use ASP DOT NET, so that you don't have to press the refresh button but it still isn't as quick as the second type of chat called internet relay chat or IRC. This is not on the World Wide Web, but found directly on servers. It is a text-based chat (no pictures) and real time (all messages immediately come on screen when entered). Here In this project we use the advanced components such as the COM and it enables the faster chatting capability.

CHAPTER 2

SYSTEM ANALYSIS

2.1 PROPOSED SYSTEM

COM objects are roughly equivalent to normal classes, but COM defines how these objects interact with other programs at the binary level. binary, I mean compiled code, with all the methods and member variables the class already built into an object. This "binary encapsulation" allows you to treat each COM object as a "black box." You can call the black box and use functionality without any knowledge of the inner workings of the object implementation. In the Windows environment, these binary objects (COM objects) are packaged as either DLLs or executable programs. COM is backed by a series of utility functions that provide routines for instantiating COM objects, process communication, and so on.

COM was the first methodology to address object-oriented software reuse. COM has enjoyed great commercial success; many third-party software vendors provide COM objects to perform a wide range of tasks, from e-mail image processing. COM is also highly useful for creating components and business objects. Business objects are COM objects in the strict sense they are used to encapsulate business rules and logic. Typically these business objects are tied to database tables. The objects move around the data according to the business rules implemented in the COM object.

CHAPTER 3

SYSTEM SPECIFICATION

3.1 USER REQUIREMENTS

IPv6 is a new version of the Internet Protocol, designed as successor to IP version 4 (IPv4). IPv6 increases the IP address size from bits to 128 bits to support more levels of addressing hierarchy, provide greater number of addressable nodes, and to simplify auto-configuration addresses.

IPv6 is available in Windows Server 2003 and Windows XP (SP1) but it is not installed by default. IPv6 is supported by .NET Framework 1.1 and Visual Studio .NET 2003. Certain project types and features in Visual Studio require that specific software components, which may be listed as optional setup, must be installed before we use that feature or project. Some of the components must be installed on the development computer, while others be installed on a remote computer.

It is possible, in fact, to author a simple application that surfaces as XML conforming to the SOAP specification. It would also be relatively straightforward to build an application capable of receiving SOAP messages over HTTP and deriving meaningful value out of it. However, just as we work with frameworks such as ASP and ASP.NET to build Web applications, we would much rather use a framework for building Web Services.

3.2 HARDWARE REQUIREMENT

Enterprise Architect

Requirement

Processor
PC with a Pentium II- class Processor 450 MHZ.
Recommended: Pentium III- class, 600MHZ.

RAM
Windows 2000 Profession — 19 MB; Windows 2000 Server — 96 MB; Windows XP Professional & Windows Server 2003 — 192 MB.
Recommended: 128 MB for 2000 Professional, 256 MB for 2000 Server 160 MB for XP Home, 256 MB for XP Professional & Windows Server 2003

Available Hard Disk Space
900 MB on system drive,
4.1 GB installation drive.

Operating System
Windows@ 2000,
Windows XP, Windows Server 2003, or Windows NT 4.0.

Video
800 x 600, 256 colors
Recommended: High Color 16-bit.

Mouse
Microsoft Mouse or compatible pointing device.

3.3 SOFTWARE REQUIREMENT

The following table lists the components that must be installed on the various operating systems to perform the tasks specified.

Based on Requirement	Windows 2000	Windows XP Windows Server 2003 or later	Windows NT4 Windows 98 Windows XP
Develop ASP Web applications and XML Web services	Internet Information Services (IIS)	IIS	Not supported
Compile code related to Microsoft Windows Message Queuing (MSMQ)	Message Queuing Services	Message Queuing Services	Not supported
Debug code on remote computers	Visual Studio Remote Debugger	Visual Studio Remote Debugger	Visual Studio Remote Debugger
Use source code control to version stored procedures	Visual Studio 6.0 Stored Procedure Versioning Visual SourceSafe Microsoft SQL Server	Visual Studio 6.0 Stored Procedure Versioning Visual SourceSafe Microsoft SQL Server	Not supported

3.4 ABOUT THE SOFTWARE

Features of ASP.NET are

- **Easy Programming Model.**

ASP.NET makes building real world Web applications dramatically easier. ASP.NET server controls enable an HTML-like style of declarative programming that let you build great pages with far less code than with classic ASP. Displaying data, validating user input, and uploading files are all amazingly easy. Best of all, ASP.NET pages work in all browsers -- including Netscape, Opera, AOL, and Internet Explorer.

- **Flexible Language Options .**

ASP.NET lets you leverage your current programming language skills. Unlike classic ASP, which supports only interpreted VBScript and JScript, ASP.NET now supports more than 25 .NET languages (including built-in support for VB.NET, C#, and JScript.NET -- no tool required), giving you unprecedented flexibility in your choice of language.

- **Rich Class Framework.**

Application features that used to be hard to implement, or required a 3rd-party component, can now be added in just a few lines of code using the .NET Framework. The .NET Framework offers over 4500 classes that encapsulate rich functionality like XML, data access, file upload, regular expressions, image generation, performance monitoring and logging, transactions, message queuing, SMTP mail, and much more!

- **Compiled execution.**

ASP.NET is much faster than classic ASP, while preserving the "just hit save" update model of ASP. However, no explicit compile step is required! ASP.NET will automatically detect any changes, dynamically compile the files if needed, and store the compiled results to reuse for subsequent requests. Dynamic compilation ensures that your application is always up to date, and compiled execution makes it fast.

- **Rich output caching.**

ASP.NET output caching can dramatically improve the performance and scalability of your application. When output caching is enabled on a page, ASP.NET executes the page just once, and saves the result in memory in addition to sending it to the user. When another user requests the same page, ASP.NET serves the cached result from memory without re-executing the page. Output caching is configurable, and can be used to cache individual regions or an entire page. Output caching can dramatically improve the performance of data-driven pages by eliminating the need to query the database on every request.

- **Web-Farm Session State.**

ASP.NET session state lets you share session data user-specific state values across all machines in your Web farm. Now a user can hit different servers in the web farm over multiple requests and still have full access to her session. And since business components created with the .NET Framework are free-threaded, you no longer need to worry about thread affinity.

- **Memory Leak, DeadLock and Crash Protection.**

ASP.NET automatically detects and recovers from errors like deadlocks and memory leaks to ensure your application is always available to your users.

- **Dynamic update of running application.**

ASP.NET now lets you update compiled components without restarting the web server. In the past with classic COM components, the developer would have to restart the web server each time he deployed an update. With ASP.NET, you simply copy the component over the existing DLL -- ASP.NET will automatically detect the change and start using the new code.

- **Easy Migration Path.**

You don't have to migrate your existing applications to start using ASP.NET. ASP.NET runs on IIS side-by-side with classic ASP on Windows 2000 and Windows XP platforms. Your existing ASP applications continue to be processed by ASP.DLL, while new ASP.NET pages are processed by the new ASP.NET engine. You can migrate application by application, or single pages. And ASP.NET even lets you continue to use your existing classic COM business components.

- **XML Web Services.**

XML Web services allow applications to communicate and share data over the Internet, regardless of operating system or programming language. ASP.NET makes exposing and calling XML Web Services simple. Likewise, ASP.NET makes it incredibly easy to call XML Web Services from your application.

- **Mobile Web Device Support.**

ASP.NET Mobile Controls let you easily target cell phones, PDAs - over 80 mobile Web devices -- using ASP.NET. You write your application just once, and the mobile controls automatically generate WAP/WML, HTML, or iMode as required by the requesting device.

CHAPTER 4

SYSTEM DESIGN

4.1 MODULE DESIGN

In this project Chat server modular design reduces complexity, facilities change (a critical aspect of software maintainability), and results in easier implementation by encouraging parallel development of different parts of a system. Software with effective modularity is easier to develop because function may be compartmentalized and interfaces are simplified. Modularity is the single attribute of software that allows a program to be intellectually manageable. Monolithic software (i.e., a large program comprised of a single module) cannot be easily grasped by reader.

Thus in this project of "Chat Server" ASP DOT NET technology is used in which XML-like tags that encapsulate the logic that generates the content for the page. The application logic can reside in server-based resources (such as Component Object Model architecture) that the page accesses with these tags. Any and all formatting (HTML or XML and ASP programming tags are passed directly back to the response page. By separating the page logic from its design and display and supporting a reusable component-based design, ASP DOT NET technology makes it faster and easier than ever to build Web-based applications.

The five important criteria that enable us to evaluate a design method with respect to its ability to define an effective modular design are,

4.1.1 Modular Decomposability

If a design method provides a systematic mechanism for decomposing the problem into sub problems, it will reduce the complexity of the overall problem, thereby achieving an effective modular solution.

4.1.2 Modular Composability

If a design method enables existing design components to be assembled into a new system, it will yield a modular solution that does not reinvent the wheel.

4.1.3 Modular Understandability

If a module can be understood as a standalone unit, it will be easier to build and easier to change.

4.1.4 Modular Continuity

If small changes to the system requirements result in changes to individual modules, rather than system-wide changes, the impact of change-induced side effects will be minimized.

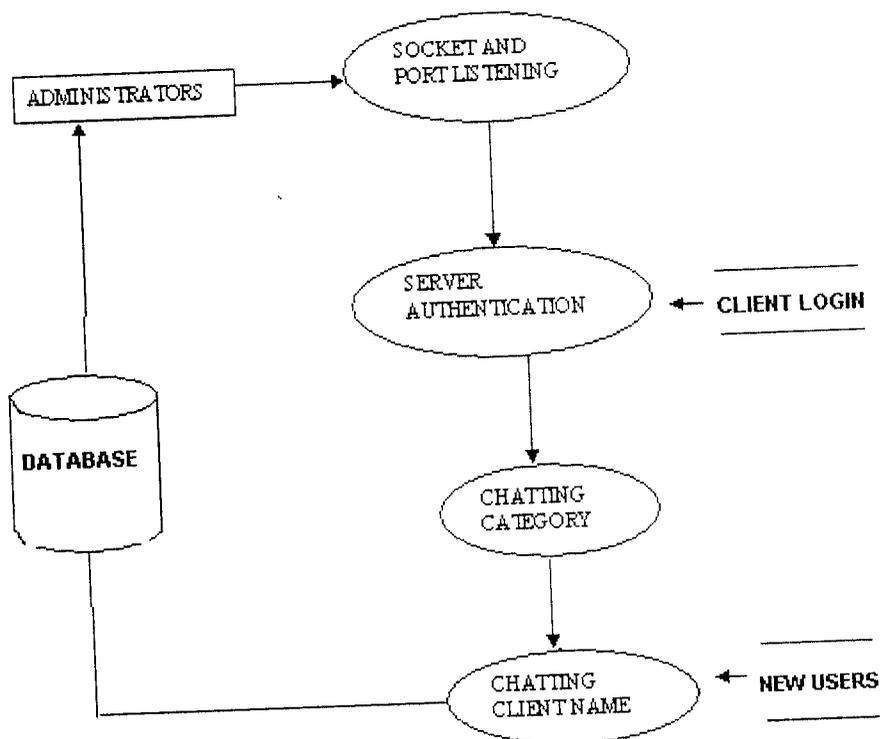
4.1.5 Modular Protection

If an aberrant condition occurs within a module and its effects are constrained within that module, the impact of error-induced side effects will be minimized.

Using the modular design concept in Chat Server the system has become really famous and the better we modularize the system the faster we can develop the system with each module performing some function to achieve the overall goal for the system.

4.2 DATA FLOW DIAGRAM

This is a tool used to show the flow of data in an information system. Data flow diagram is a network that describes the flow of data and the processes transforms data throughout the system. This network is constructed by using a set of standard symbols.



4.3 DATABASE DESIGN

Database Design is required to manage large pieces of information. The management of data involves both the definition and provision of mechanism for manipulation of data. In addition the database system must provide for the safety information handled despite system crashes or due to attempts at unauthorized access.

Normalization

Data in an unnormalized form consists of entry, which has repetitive occurrence. Unnormalized data is broken down and the repetitive items are eliminated. Only atomic values are contained in the tables. All the tables in our system confirm to this first normal form.

FIELD NAME	TYPE
First name	Char (15)
Last name	Char (15)
Gender	Char (10)
Age	Int (4)
Address	Char (10)
City	Char (10)
Country	Char (10)
Phone	Int (4)
Email	Char (15)
Total user	Int (4)

Table 4.1Registration

4.4 INPUT DESIGN

Input Design is the process of converting the user-originated inputs to a computer-based format. The design decisions for handling input specify how data are accepted for computer processing. Input design is a part of overall system design that needs careful attention and it includes specifying the means by which actions are to be taken. The collection of input data is regarded to be the most expensive part of the overall system design. Since the inputs have to be planned in such a manner so as to get the relevant information, extreme care is taken to obtain the pertinent information. If the data fed into the system is incorrect, then the processing as well as the outputs will magnify these errors. The arrangement of messages and comments in online conversations, as well as, the placement of data, headings, and titles on display screens or source documents, is also part of the input design. An improper design that leaves the screen blank will confuse a user about what action to be taken next. Online systems include a dialogue or conversation between the user and the system.

The input design consists of developing specifications and procedures for data preparation, those steps necessary to put transaction data into usable form for processing and data entry, the activity of putting the data into the computer for processing.

In this Project “Chat Server” thin-client multi-tiered applications are hard to write because they involve many lines of intricate code to handle transaction and state management, multithreading, resource pooling, and other complex low-level details. The component-based DOT Net architecture, which is organized into reusable components and the web server, provides underlying services in the form of a container for every component type. Because you do not have to develop these services yourself, you are free to concentrate on

4.5 OUTPUT DESIGN

Output design refers to the results and information that are generated by the system. Output is the main reason for developing the system and it is on the basis of the outputs generated that the usefulness of the application is evaluated.

The significant points that have to be considered for the output design are as follows:

- Determine what information to present.
- Decide whether to display, print, or voice the information and select the medium of output.
- Arrange the presentation of information in an acceptable format.
- Decide how to distribute the output to intended recipients.

The arrangement of information on a display, or a printed document is termed as a layout. The output design is specified on layout forms, sheets that describe the location characteristics and the format of the column headings and pagination. Output design phase of system is concerned with the convergence of information to the end user in user-friendly manner. In Web server, the output design is efficient, intelligible so that the system relationship with the end user is improved considerably and thereby, enhancing the decision-making process. The contents of the output are then defined in a detailed manner during the physical design of outputs. Certain data are edited in a desirable manner.

4.6 FILE DESIGN

Mapping File Format

In dot net we use file designs to easy access the file system in the project one such is mapping file format.

Name: map_XX_XX.xml (defined by mapspec in mapgroup file)

Location: mapdir (location defined by file in mapgroups file)

Format:

```
<Map>
  <Header>
  <Mapdata>
  <Objects>
</Map>
```

Header Format:

```
<Header>
  <name>
  <reset>
  <width>
  <height>
  <difficulty>
</Header>
```

only the name field is required.

Field Name	Description	Field Type	Default
name	Name of map	string	REQUIRED
reset	Time until map auto-resets	int	0
width	Width of map in pixels	int	inherited from mapgroup

height	Height of map in pixels	int	Inherited from mapgroup
difficulty	Difficulty level of this map	int	0

4.2 Mapping file format

Mapdata Format:

<Mapdata>

<elevfile>

<terrfile>

</Mapdata>

elevfile and terrfile are required fields.

Field Name	Description	Field Type
elevfile	Path to elevation PNG (8bit grayscale PNG)	string
terrfile	Path to terrain PNG (8bit grayscale PNG)	string

4.3 Mapping data format

Objects Format:

<Objects>

<Object arch="x">

<Object arch="n">

</Object>

The Objects section is completely optional.

An Object header, with a required arch property, defines each object. This arch property MUST match an archetype available to the server (in the archetypes.xml file). Inside the Object tag, you can then override any of the settings for that archetype to make this particular object unique. The only field

that cannot be overridden is the archetype field. Most commonly, the coord field is overridden, to place the object at a specific location on the map.

Mapgroups File Format

Name: mapgroups.xml

Location: mapdir

Format:

```
<Mapgroups>
  <file>
</Mapgroups>
```

Each file entry is the name of a map group description file.

Mapgroup File Format

Name: mapgroup.xml (can be anything, specified by mapgroups file)

Location: mapdir (can be anywhere, specified by mapgroups file)

Format:

```
<Mapgroup>
  <name>
  <nrofmaps>
  <width>
  <mapsize>
  <baseelev>
  <mapspec>
</Mapgroup>
```

Only name and mapspec are required fields.

Field Name	Description	Field Type	Default
name	Name of this mapgroup	string	REQUIRED
nrofmaps	Number of maps in this group	int	1

width	Width of mapgroup	int	1
mapsize	Size in pixels of sub-maps	int	256
baseelev	Base elevation in meters	double	0.0
mapspec	Map file name template, such as <code>_XX_XX</code> (width, height). Must contain two sets of X's	string	REQUIRED

4.4 Map group file format

4.7 PRELIMINARY DESIGN

Preliminary Design is the first phase of the design process. A Project Manager is assigned to the project and will coordinate a series of meetings with Users and the Design Team for information gathering. Users communicate specific needs/requirements and the Design Team will do field investigation regarding the layout of the existing areas in question including building systems and their impact on the project. The Design Team generates schemes based on the project information provided by Facilities Development. Schemes will be reviewed by all stakeholders and refined accordingly. This phase of the project defines the design parameters and lays out the overall scheme.

The agreed upon scope of each project is defined during the formulation process. The Design Team during the Preliminary Design phase to refine the scope with more detailed information from the site investigation and User input. Any changes to the agreed upon and authorized scope of the project must be approved by the Project Manager with all consequences for schedule and budget understood and accepted by the project funding source.

The agreed upon budget of each project is defined during the formulation process. The formulated budget to further refine it for each project once the project is authorized to begin. The authorized budget shall not be exceeded. Budget impacts will be monitored throughout the Preliminary Design phase and shared with all parties. As changes in the scope and schedule of a project will impact budget, an estimate will be prepared at the end of the PD phase to verify that the project remains within budget. The budget must be reconciled with scope prior to proceeding with the next phase of design.

CHAPTER 5

SYSTEM TESTING

System testing is an expensive but critical process that can take as much as 50% of the budget for program development. The common view of testing held by users is that it is performed to prove that there are no errors in a program. However, as indicated earlier, this is virtually impossible, since analysts cannot prove that software is free and clear of errors. Therefore, the most useful and practical approach is with the explicit intention of finding errors that is making the program fail. The tester, who may be the analyst, programmer, or specialist trained in software testing, is actually trying to make the program fail. A successful test, then, is one that finds an error. Analysts know that an effective testing program does not guarantee systems reliability. Therefore, reliability must be designed into the system.

System testing plays an important role where the implemented system is been verified. It should be tested with all sorts of data so that it works correctly and the result thus obtained is accurate. So, all the functional modules of the subsystem are tested individually. All combination of data is input into the system and the results are compared with original results got. Thus unit testing is done for all procedures of the system. The change was in messages when needed, which made the system more users - friendly.

5.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design. Four categories of tests are performed on each unit:

- Functional test
- Performance test
- Stress test
- Structured test

5.2 Integrated testing

Integrated testing takes place on code that constitutes a progressively more complex environment as the individual units are combined into larger functional components. More than one level of integrated testing may be needed to test all functional components of a software product.

5.3 Validation testing

System validation checks the quality of the software in both simulated and live environments. First the software goes through a phase (alpha testing) in which errors and failures based on simulated user requirements are verified and studied. The modified software is then subjected to phase two (beta testing) in the actual user's site or a live environment. After a scheduled time, failures and errors are documented and final correction and enhancements are made before the package is released for use. In this package, the validation test is performed with the different set of data in many ways and corrected the errors.



5.4 Output Testing

After performing the validation testing, the next step is output of the proposed system, since no system could not be useful if it does not produce the required output in the specific format. The outputs generated or displayed by the system under consideration are tested by asking the users about the required by them. Here, the output format is considered in two ways. One is on screen and another is printed format. The output format on the screen is found to be correct as the format was designed in the system design phase according to the used needs. Fro the hard copy also, the output comes out as specified requirements by the user. Hence, output testing has not resulted in any correction in the system.

5.5 System Testing

A system testing does not test the software per se but rather the integration of each module in the system. It also tests to find discrepancies between the system and its original objective, current specifications, and system documentation. The primary concern is the compatibility of individual modules. Systems testing must also verify that file sizes are adequate and that indices have been built properly. Sorting and re-indexing procedures assumed to be present in lower level modules must be tested at the systems level to see they in fact exist and achieve the result modules expect.

5.6 Specification Testing

To perform specification testing, the analyst examines the specifications stating what the program should do and how it should perform under various conditions. Then, test cases are developed for each condition or combination and submitted for processing. By examining the results, the

analyst can determine whether the program performs according to its specified requirements. This strategy treats the program, as it was a black box: the analyst does not look into the program to study the code and is not concerned about whether every instruction or path through the program is tested.

CHAPTER 6

IMPLEMENTATION

The first stage in the implementation phase is the implementation plan where the system is taken to the on line web site. The process follows the initial testing, which is done by the developer by giving as many data as possible to validate the system.

After this modification the results of the manual system with those of the computerized system is compared. If there is any difference, modifications are made as needed. If any modification is made, parallel running continued for some more time and the same process is continued till perfection is reached.

A set of users are selected, a general illustration of the working and operation is examined to this set of users. Then they are made to work in the system to identify the user difficulties that arise while working. As general help, for the operation and function, each button is provided with some meaningful names.

The implementation of online testing split up into six major components. They are

- Discovery Phase
- Configuration Phase
- Installation Phase
- Administration Phase

Training Phase

- Benchmarking Phase

Discovery Phase:

The first component is Discovery Phase. This phase is where we take the requirements we identified in our planning process and start setting up the system to capture the proper information and determine how it should be presented and/or reported, who should see what information, what the maintenance procedures will be, how to distribute technical updates, and more.

Configuration Phase:

The second component is Configuration Phase. This is where the database structure is tailored to our exact and initial needs. Fields are added, forms designed, reports customized etc.,

Installation Phase:

Third is the Installation Phase. This is the phase where we prepare our network for the new system-security, remote connectivity, hardware etc., this is also the phase where we install and configure the software on our server as well as convert and import our existing databases, if any, into the new system.

Administration Phase:

The fourth component is the Administration Phase where the system security is set up, menus are tailored to meet specific user needs, and views are created to capture relevant information.

Training Phase:

Component number five is Training Phase. The amount of information that should be included in the training course should be limited to the core competencies that are required to get the individual users productivity as quickly and as easy as possible. The number of days for training will vary depending on the experience of the users, perhaps one day for experienced users, two days for users who are not familiar with these applications, and three days for inexperienced Windows users. The training should not only include product features, but also how the product should be used within their organization, how management wants things to be entered and tracked, what probabilities to assign to forecasts based on levels of activities with the account, what codes needed to be entered in order to track particular activities, what methods are set up to synchronize from remote location and more.

Benchmarking Phase:

Phase six is the Benchmarking Phase, or the "How are we doing" phase. About sixty or ninety days after the initial rollout, we should measure our progress versus our original goals and objectives; this should be repeated after every phase of the rollout program, as well. Review the training, check whether was it effective? Is more needed? What were the project's successes and mistakes so far? What goals we have hit and which ones we have missed? Define what course of action needs to be taken from here, or what changes need to be made before proceeding.

CHAPTER 7

FUTURE ENHANCEMENTS

The .NET Framework leverages many existing Windows services to make it a more robust application environment. A particular technology that deserves attention is COM+ Component Services. These technologies were the predecessors to the .NET Framework. Security is of paramount importance, especially for applications intended to run on the Internet. In the past, programming security features into an Internet application was largely a manual effort. Often it consisted of custom security schemes that did not necessarily leverage the existing security infrastructure provided by the operating system. Besides being difficult to maintain, such security systems are typically costly to develop.

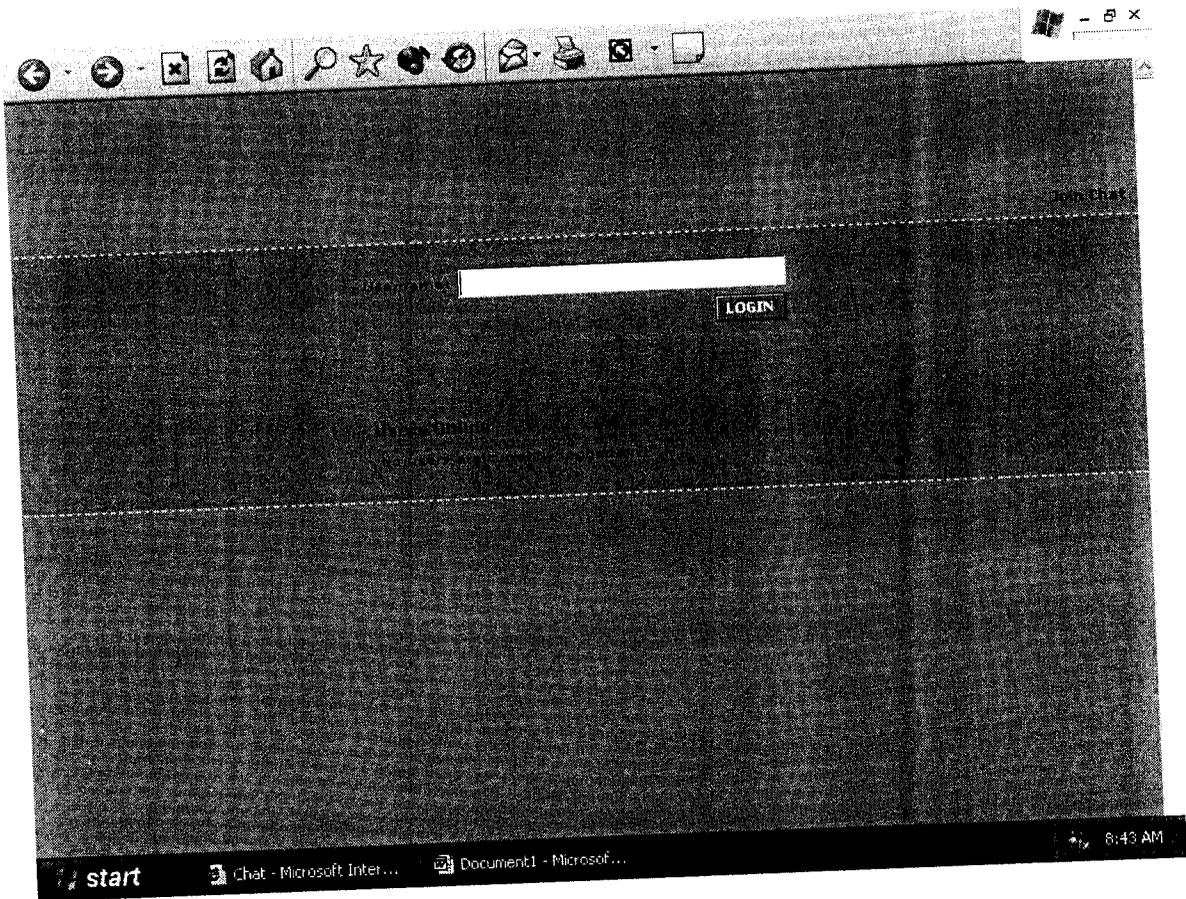
This role-based security is not only easy to implement (it can be done by the system administrator) but it also typically doesn't require the programmer to work on the components to implement any security code. When a call is made to a component running under COM+ Component Services, COM+ checks the user/group identity of the caller and compares it against the roles assigned to the component. Based on that comparison, the call is allowed or rejected. In future the ASP.net chat server when it uses the COM+ it can leverage the existing system.

CONCLUSION

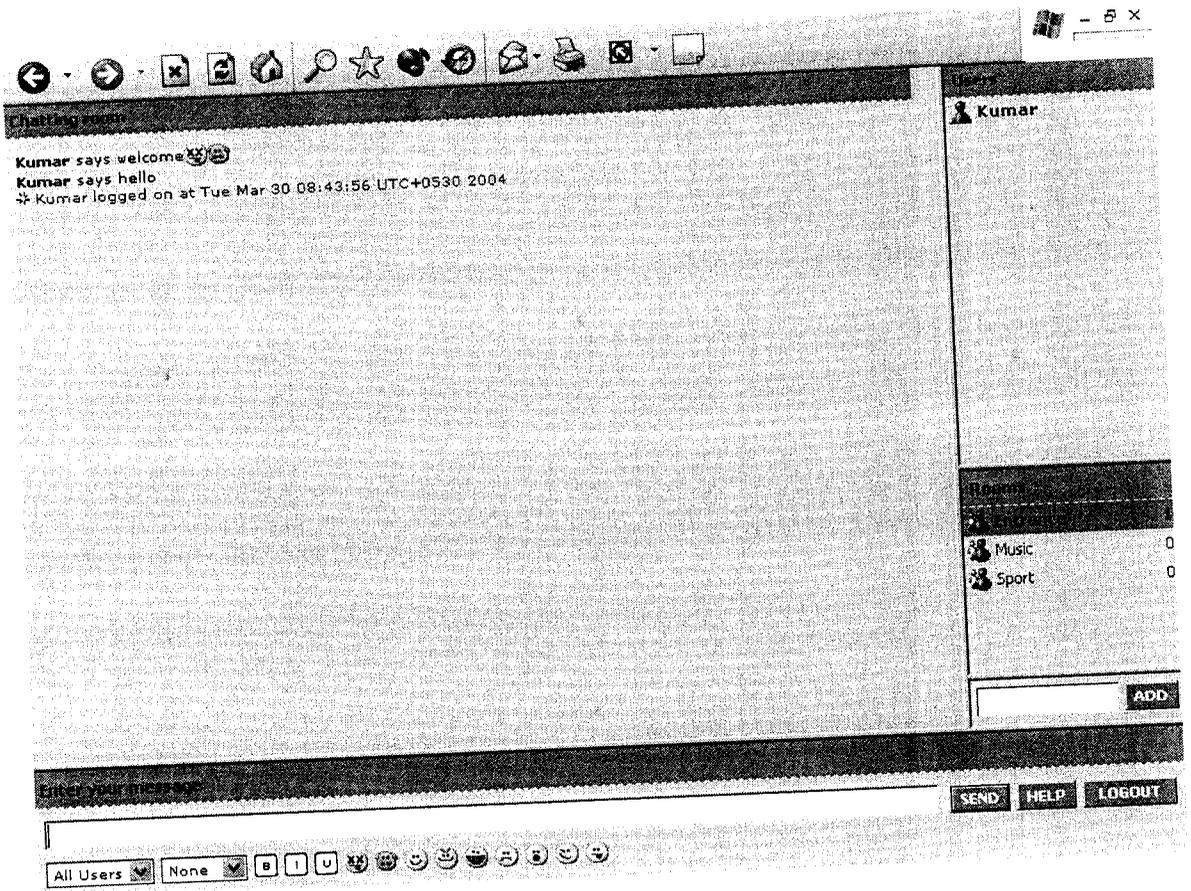
There can be many reasons to reroute incoming Web requests. For instance, sometimes it's necessary to redirect a browser to a page based on user criteria without passing long lists of parameters in the URL. In the past, the only way to intercept such page requests and send them elsewhere was with ISAPI. Now, in ASP.NET, the IHttpModule interface provides notification of server requests, and let's easily reroute them based on criteria other than browser type or version. Here the author in this project of chat server use the IHttpModule for interception and explains the use of ISAPI filters for in ASP.NET.

APPENDIX

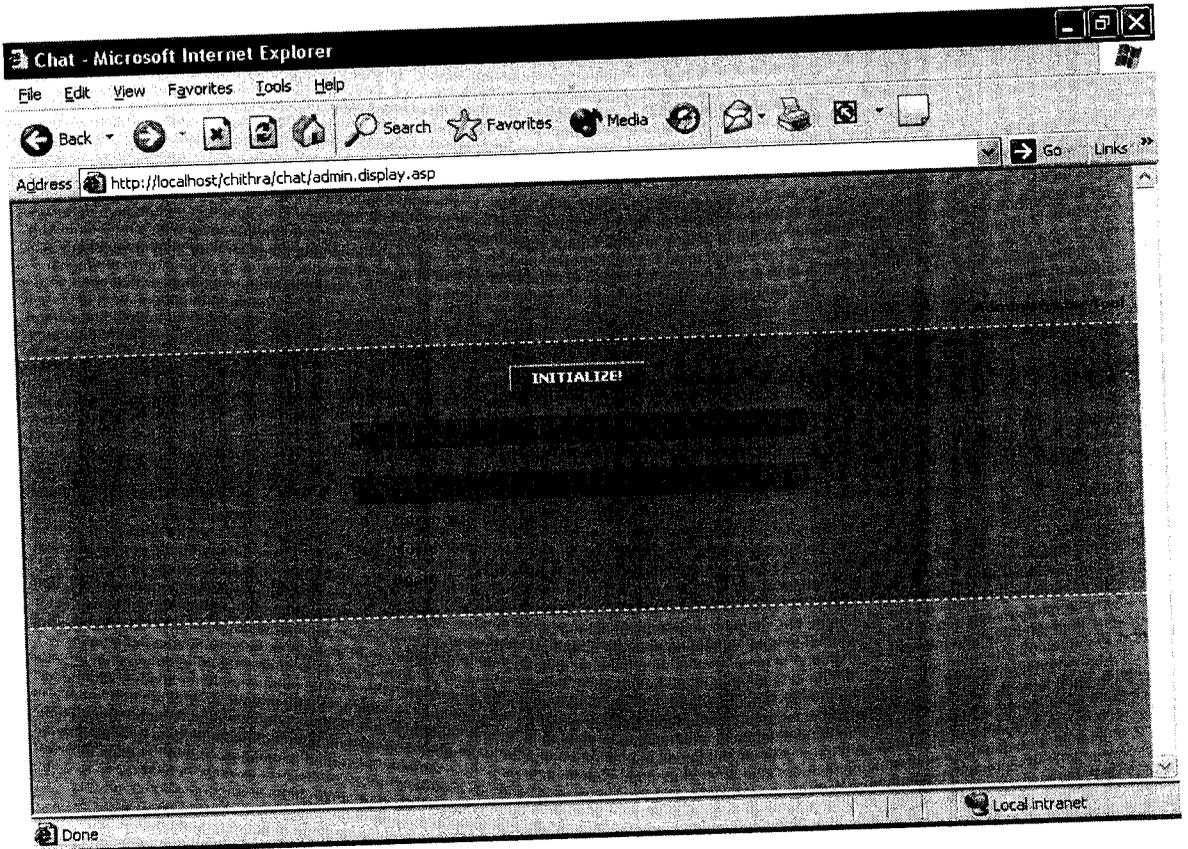
LOGIN FORM



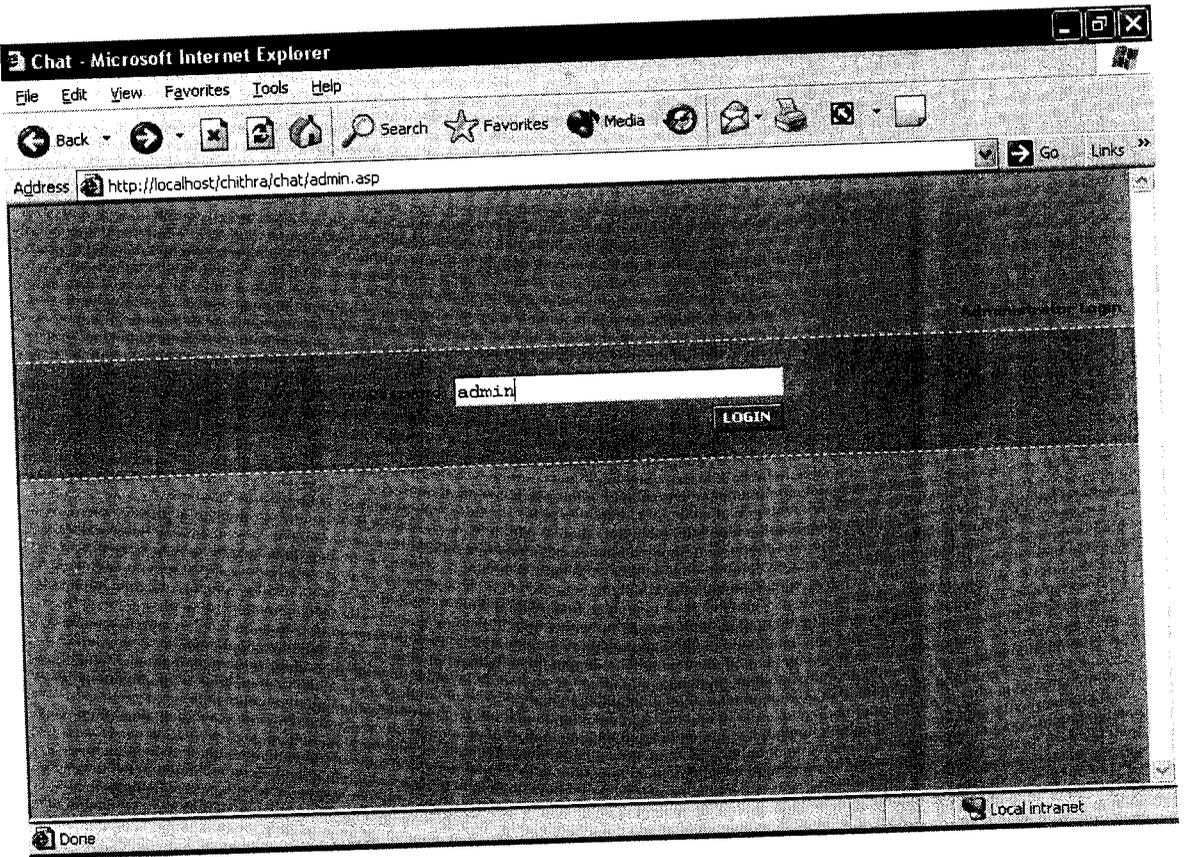
CHATTING ROOM:



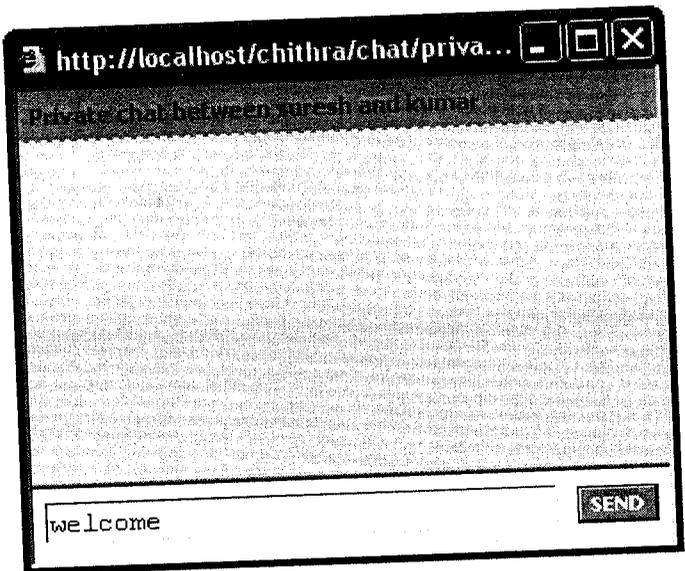
INITIAL FORM



ADMINISTRATOR FORM



PRIVATE CHAT ROOM



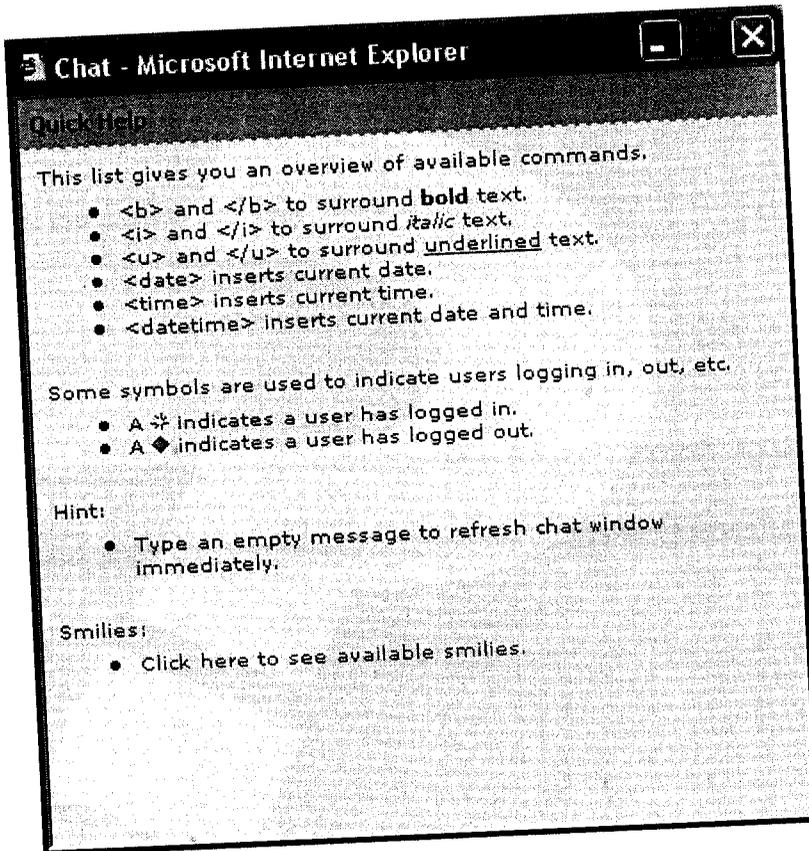
PROFILE FORM

The image shows a screenshot of a Microsoft chat application window titled "Chat - Profile for kumar - Microsof...". The window contains a profile form for a user named "kumar". The form lists the following information:

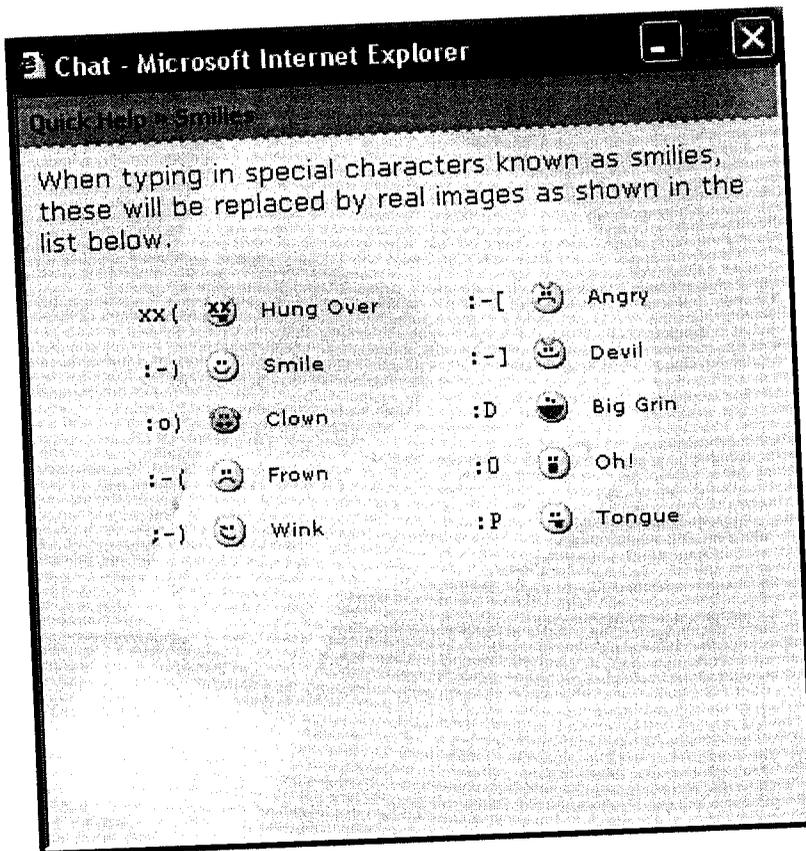
IP Address	127.0.0.1
Logged On	3/30/2004 9:54:06 AM
Last Action	3/30/2004 9:54:06 AM
Written Messages	0

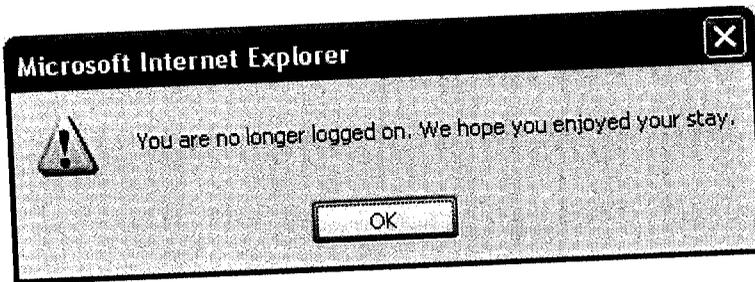
At the bottom right of the window, there is a "CLOSE" button.

FORM FOR QUICK HELP

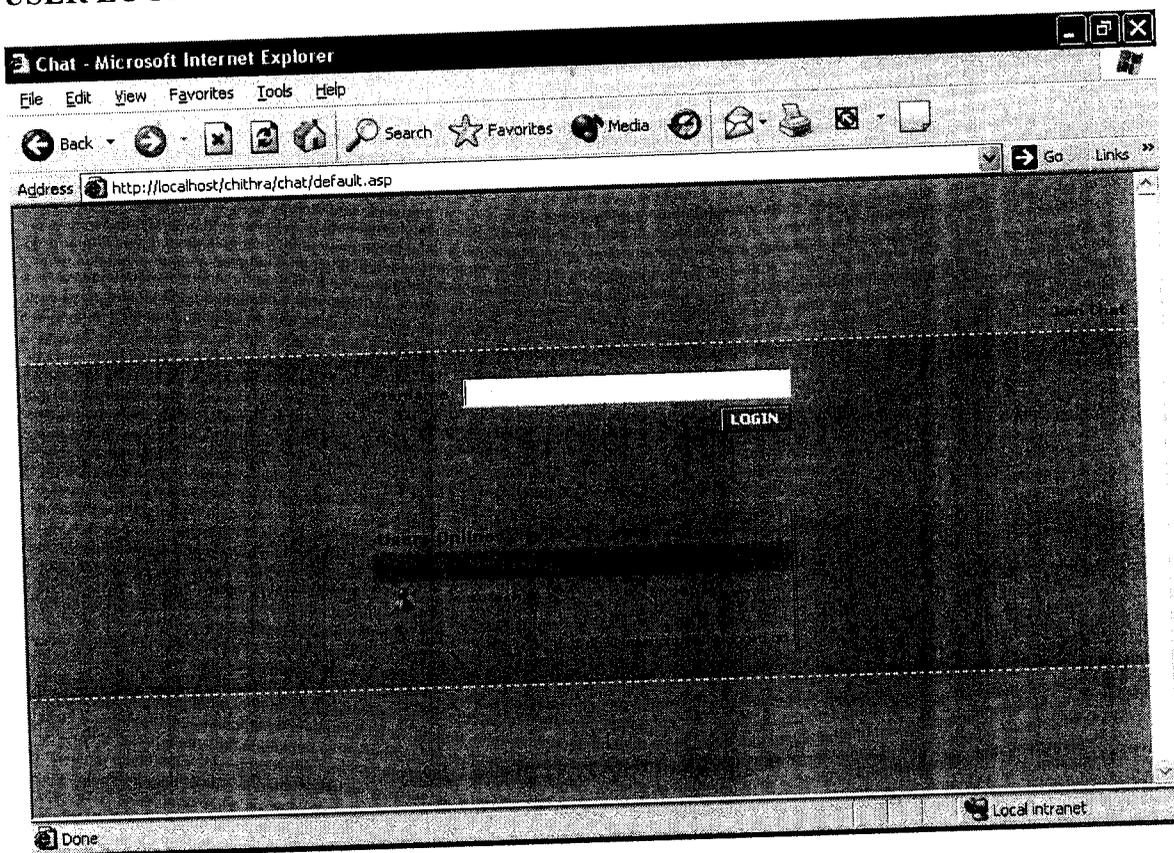


LIST OF SMILEYS



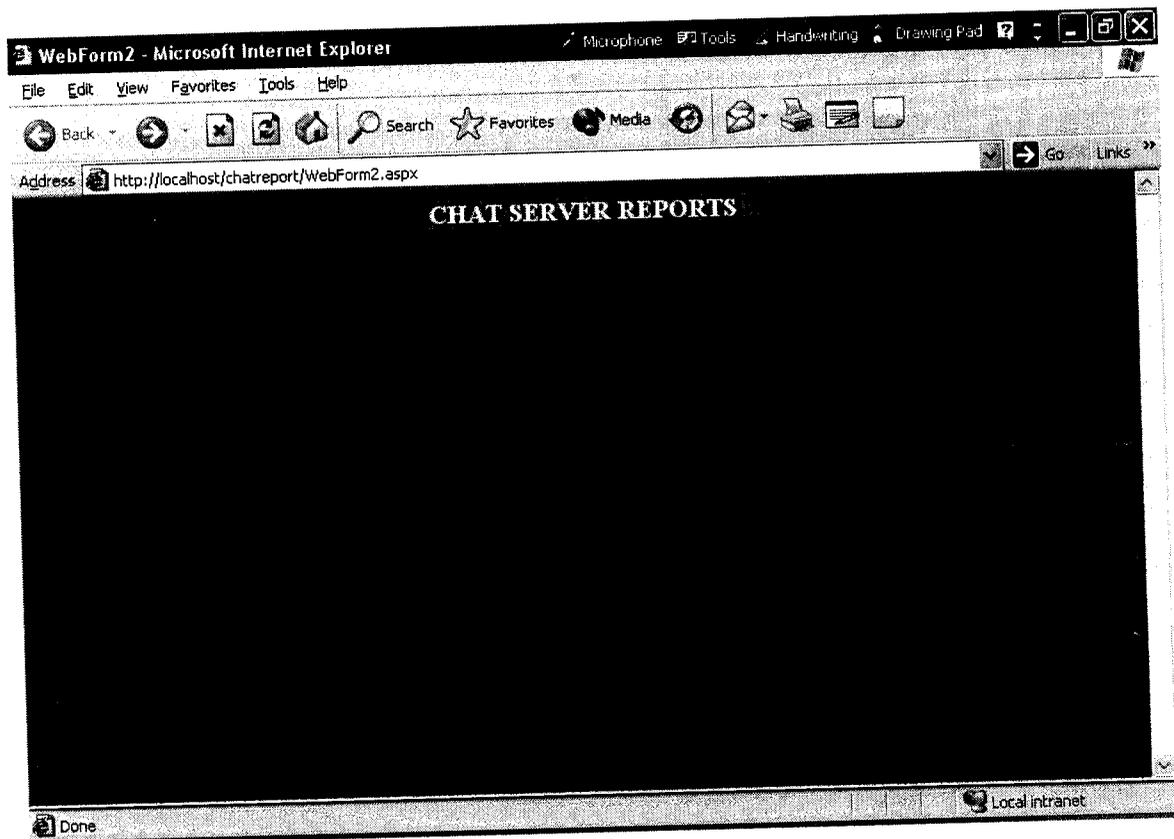
EXIT MESSAGE

USER LOGIN



REPORTS

CHAT SERVER REPORTS



DATE WISE REPORT

WebForm1 - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address <http://localhost/chatreport/WebForm1.aspx>

powered by crystal

Date Wise Report

<u>Registration date</u>	<u>first name</u>	<u>last name</u>	<u>gender</u>	<u>age</u>	<u>address</u>	<u>city</u>	<u>country</u>	<u>phone</u>	<u>email</u>	<u>Total users</u>
1/14/2004										
1/14/2004 12:00:00AM	Merlin	Joseph	female	17	NorthHamton	London	Engla	3432324	mar@yahoo.com	1
										1.00
11/1/2004										
11/1/2004 12:00:00AM	Kumar	babu	male	20	1st street, gandhi e puram	coimbatore	India	2389067	kumar@yahoo.com	1
										1.00
11/1/2004										
12/2/2004										

Done Local intranet

EMAIL LIST REPORT

5/13/2004

email	first_name	last_name	gender	ag	address	city	country	phone
kumar@yahoo.c kumar@yahoo.co	Kumar	babu	male	20	1st street, ghandhapur am	coimbatore	India	2389067
kumar@yahoo.c mar@yahoo.com	Merlin	Joseph	female	17	NorthHamton	London	England	34323243
mar@yahoo.com vig@yahoo.com	Vignes	mani	Male	30	Wlker street	USA	USA	55423343

Local intranet

COUNTRY WISE REPORT

cont - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail

Address http://localhost/chatreport/cont.aspx

1 of 1 100% powered by crystal

Country Wise Report

5/13/2004

<u>country</u>	<u>first_name</u>	<u>last_name</u>	<u>gender</u>	<u>age</u>	<u>address</u>	<u>city</u>	<u>phone</u>	<u>email</u>
<u>England</u> England	Merlin	Joseph	female	17	NorthHamton	London	34323243	mar@yahoo.com
<u>India</u> India	Kumar	babu	male	20	1st street,gandhipuram	coimbatore	2389067	kumar@yahoo.co
<u>USA</u> USA	Vignes	mani	Male	30	Wiker street	USA	55423343	vig@yahoo.com

Done Local Intranet

TOTAL NUMBER OF USERS

The screenshot shows a Microsoft Internet Explorer browser window. The address bar displays 'http://localhost/chatreport/tusers.aspx'. The page content includes a date '5/13/2004', a table with columns for 'Tot', 'first_name', 'address', 'city', 'country', 'phone', and 'email', and a summary section with 'Grand Total: 3.00'.

5/13/2004

<u>Tot</u>	<u>first_name</u>	<u>address</u>	<u>city</u>	<u>country</u>	<u>phone</u>	<u>email</u>
1	Kumar	1st	coimbatore	India	2389067	kumar@yahoo.com
1	Merlin	NorthHamton	London	England	34323243	mar@yahoo.com
1	Vignes	Wlker street	USA	USA	55423343	vig@yahoo.com
<u>3.00</u>	<u>1</u>					
<u>3.00</u>	<u>Grand Total:</u>					

Local intranet

7-1516

AGE WISE REPORT

WebForm3 - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Home Search Favorites Media

Address http://localhost/chatreport/age.aspx

1 of 1 100% powered by crystal

Age Wise Report

5/13/2004

<u>age</u>	<u>first_name</u>	<u>last_name</u>	<u>gender</u>	<u>address</u>	<u>city</u>	<u>country</u>	<u>phone</u>	<u>email</u>
17	17 Merlin	Joseph	female	NorthHamton	London	England	34323243	mar@yahoo.com
20	20 Kumar	babu	male	1st street,ghandhipur am	coimbatore	India	2389067	kumar@yahoo.co
30	30 Vignes	mani	Male	Wiker street	USA	USA	55423343	vig@yahoo.com

Done Local intranet

1516

GENERAL REPORT

gens - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media

Address: http://localhost/chatreport/gens.aspx

1 of 1 100% powered by crystal

General Report

5/13/2004

<u>first_name</u>	<u>last_name</u>	<u>gender</u>	<u>age</u>	<u>address</u>	<u>city</u>	<u>country</u>	<u>phone</u>	<u>email</u>
Kurnar	babu	male	20	1st street,ghandhipu ram	coimbatore	India	2389067	kumar@yahoo
Merlin	Joseph	female	17	NorthHamton	London	England	3432324	mar@yahoo.co
Vignes	mari	Male	30	Wlker street	USA	USA	5542334	vig@yahoo.co

Done Local Intranet

1516



REFERENCES

1. Harold Davis, Visual ASP .NET Programming.
2. Marco Bellinaso, Kevin Hoffman, ASP.NET Website Programming: Problem - Design – Solution.
3. Matthew Macdonald, Robert Standefer, ASP.NET: The Complete Reference.
4. G. Andrew Duthie, Microsoft ASP.NET Programming.
5. Dino Esposito, Programming Microsoft ASP.NET.

