**A Project report on**

**“INTRANET CHATTING”**

****

**Submitted to**

**Rajiv Gandhi Prodhyogiki Vishwavidhyalaya, Bhopal**

**Towards partial fulfillment for**

**the Degree of Bachelor of Engineering in**

**Computer Science and Engineering**

**2011-12**

**GUIDED BY:** **SUBMITTED BY:**

**Mr. Brajesh Chaturvedi Mrs. Bhavna Kushwah (0822CS091011)**

**(Lecturer) Mrs. Mamta Bhamre (0822CS091037)**

**Mr. Vaibhav Namdev (0822CS091082)**

**Department of Computer Science and Engineering**

**Swami Vivekanand College of Engineering, Indore**

**www.svceindore.ac.in**

**CERTIFICATE**



This is to certify that **Mrs. Bhavna Kushwah, Mrs. Mamta Bhamre and Mr. Vaibhav Namdev,** students of Third year (VI SEM) **Computer Science and Engineering** have completed their minor project entitled **“INTRANET CHATTING”** towards the partial fulfillment of the degree in **Bachelor of Engineering** in Computer Science and Engineering awarded by **Rajeev Gandhi Technical University, Bhopal** for the academic year 2013.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Internal Examiner** **External Examiner**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Director**

**ACKNOWLEDGEMENT**

First, we would like to express our heartfelt gratitude to our project guide **Mr. Brajesh Chaturvedi** for guiding us throughout the project. We are highly indebted to them for their invaluable guidance and every-ready support, which was necessary for the successful completion of the project in stipulates time. Their deep knowledge of computer engineering and information technology field made us realize that theoretical knowledge always helps to develop efficient operation industrial software, which are blend of all core subjects of the field. Working under their guidance has been a fruitful experience, which will be very valuable for us, when we enter the corporate world.

We would like to give a warm expression of thanks to **Dr. A.G.Ambekar**, Principal S.V.C.E., Indore for providing the facilities and academic environment for our project work.

We also thank our respected Head of Department, **Mr.** for his valuable guidance and encouragement. We also thank all the staff members for their encouragement and support throughout this project and all those who have embedded us with technical knowledge of computer technology.

We sincerely thank to all our friends and well -wishers for directly or indirectly helping us during the course of the work.

Thank You

Ms. Bhavna Kushwah (0822CS091011)

Ms. Mamta Bhamre (0822CS091037)

Mr. Vaibhav Namdev (0822CS091082)

**ABSTRACT**

This Project Entitled as ‘INTRANET CHATTING’ is used basically for chatting purpose with the remote clients or users on Internet or local networks. Here in this project a java client / server combination is used to chat with remote users. When a Client wants to chat with a user on a remote host, he sends a request to the Server with a identification name like chat-id, the server responds to the request by identifying the client-id which is already registered in the server domain and when matched his request is granted and the client can begin to chat with the remote users present on the internet or local network.

The benefit of using “INTRANET CHATTING” over other chatting tools is that, with the help of java, the programmer can create applet applications which can be use the internet as a server. Applets are machine independent and so java programs can run on any computer on the internet.

The program's premier feature is its WHITE BOARD drawing utility. You can draw freehand, do circles, squares, lines, text, or paste image files to the canvas.  This is ideal when users want to "sketch" concepts for one another. This feature of “INTRANET CHATTING” can be a boon for the technical people who want to share their ideas of concepts in the pictorial form. Users can interact publicly with everyone else on the server, or they can chat/draw privately using java Chat's "whisper mode". Users can create and manage chat rooms, which can be either "public" or "private".

**INDEX**

**Title** **Page No.**

1. **Introduction** 
   1. **Problem Definition**
   2. **Objectives** 
      1. **Scope**

**1.4 Platform Specification**

* + 1. **Hardware**
    2. **Software**
    3. **Implementation Language**
    4. **Tools Used**

**2. Literature Survey**

* 1. **Work done by others**

**2.2 Benefits**

* 1. **Proposed solution**

**3. Feasibility Study**

* 1. **Technical Feasibility**
  2. **Economical Feasibility**

**4. Software Engineering Approach**

* 1. **Requirement Analysis**
  2. **Analysis & Design**

* + 1. **Data Flow Diagrams**
  1. **Diagrams**

**4.3.1 Use case Diagrams**

**4.3.2 Class Diagrams**

**4.3.2 Sequence Diagrams**

**4.3.2 Activity Diagrams**

* + 1. **Collaboration Diagrams**

**5. Implementation Phase**

**5.1 Code Explanation**

**5.2 Output Screens**

**5.3 Testing**

**5.3.1 Testing Objectives**

**5.3.1.1 Testing Methods and Strategies**

**5.3.1.2 Test Case**

* 1. **Conclusion and Discussion** 
     1. **Limitation of the Project**
     2. **Difficulties Encountered**
     3. **Future Enhancement Suggestion**

**7.** **Bibliography and Reference**

**7.1 Web and Book References**

**7.2 Other Documentation and Resources**

**Chapter 1**

**INTRODUCTION**

**INTRODUCTION**

**1.1 PROBLEM DEFINATION**

The program is called Chat, when you are Chatting, everything you type will instantly be transmitted around the world to other users that might be watching their terminals at the time. They can then type something and respond to your messages, and vice versa. Chatting is based on a client-server model. Clients are programs that connect to a server; a server is a program that transports data, (messages), from a user client to another. There are clients running on many different systems that allow you to connect to a Chat server. Currently most of the current Chat applications are text based and few are capable of transferring tiny images, but there lacks a Chat system for the technical people who can chat as well as they can represent their ideas in the form of drawing the pictures online. Many vendors distribute even these technologies separately but to obtain these features at one system will be the haunting task.

So what should we do and how should we solve the problem that even the technical people are benefited by this chatting technology. Through current chatting technologies we are able to send only text based messages to people and tiny images, but this type of chatting is not helping the technical people to work efficiently when the question of sending big pictures like say business plans to the customers that is the business clients to approve of the plan or the client project, imagine big business plans and deals getting approved through chatting and large business projects started, how beneficial it will be to the technical people as well the client on the other side.

The tool we are here going to design in this project is the White Board utility embedded in the Chat Application. This tool is not generally present in the chat application we are using. Our purpose is to design such tool in chat application that can share free hand drawing and other shapes on runtime.

These are our major tasks we are going to include in our project. And the goal of this is to provide simplicity and providing additional facilities to the user and specially Engineers, Doctors and Architects.

**1.2 OBJECTIVES**

This Project Entitled as “INTRANET CHATTING‟ is used basically for chatting purpose with the remote clients or users on Internet or local networks. Here in this project a java client / server combination is used to chat with remote users. When a Client wants to chat with a user on a remote host, he sends a request to the Server with a identification name like chat-id, the server responds to the request by identifying the client-id which is already registered in the server domain and when matched his request is granted and the client can begin to chat with the remote users present on the internet or local network.

The power of Internet is such that it integrates together LANs located across diverse software and hardware forms into a single large communication network that spans the globe. The client needs to have a client software such as Netscape Navigator or Internet Explorer to retrieve information as well as chat on the www. WWW is referred to as Netsurfing. And it supports chatting, text, video and audio.

The benefit of using “INTRANET CHATTING” over other chatting tools is that, with the help of java, the programmer can create applet applications which can be use the internet as a server. Applets are machine independent and so java programs can run on any computer on the internet. The term client/server is used in the context of networking, what it actually means. It is important to understand the terms client/server because the INTRANET CHATTTING project is supported completely by client/server model. A server is anything that has some resource that can be shared. There are compute servers, which provide computing power, web servers, which store web pages. A client is simply any other entity that wants to gain access to a particular server. The interaction between client\server is like an interaction between an electrical socket and a lamp. The server is a permanently available resource while client is free to “unplug” after it has been served.

The program's premier feature is its whiteboard drawing utility. You can draw freehand, do circles, squares, lines, text, or paste image files to the canvas. This is ideal when users want to "sketch" concepts for one another. Users can interact publicly with everyone else on the server, or they can chat/draw privately using java Chat's "whisper mode". Users can create and manage chat rooms, which can be either "public" or "private". The server will also store and manage answering machine-style messages for users who aren't online, and instant messages can be sent to those who are. Additionally, users who aren't watching their screens can be paged with sound.

**1.3 SCOPE OF THE PROJECT**

WWW is called the World Wide Web. WWW supports many kinds of text, pictures, video and audio. WWW resources through a web browser which basically a program that runs on the internet.

There are two kinds of browsers:

1. Text only browsers.
2. Graphical browsers.

Graphical browser like Netscape Navigator and Internet Explorer are popular. These browsers provide your Inline images, fonts & document layouts. When you access a WWW server, the document is transfer to your computer and then the connection is terminated.

The World Wide Web is a network of information, accessible via an easy-to-use interface. The information is often presented in hypertext or multimedia and provided by the servers located around the world. The usability of the Web depends largely on the performance of these servers.

This application is a Java client/server combination, which can be used to chat over the internet or local networks. With these features and with the advent of WWW, Web browsers and the “INTRANET CHATTING”, Internet has become the media of applications.

We can use “INTRANET CHATTING” for following activities:

1. To exchange information and converse with friends and family.
2. To participate in group discussion through public news bulletin board.
3. For Entertainment.
4. Leisure activities.
5. Access business while at home.
6. Communicate and collaborate through pictures and images.
7. At any given point of time, up-to-date information is provided.

**1.4 PLATFORM SPECIFICATION**

**1.4.1 Hardware Specification**

* CPU : Pentium 3
* Processor Speed : 2 GHz
* Coprocessor : Built In
* Total Ram : 128 Mb
* Hard Disk : 40 GB
* Keyboard : 101 keys of above

**1.4.2 Software Specification**

* Front End : Java Swing
* Operating System : Windows XP
* Java Development Toolkit 1.2 and above.
  + 1. **Implementation Language**
* Java

**1.4.4 Tools Used**

* NetBeans IDE 6.9 (For java application)
* IBM Rational Rose (For UML)

**Chapter 2**

**Literature Survey**

**LITERATURE SURVEY**

* 1. **WORK DONE BY OTHERS**

It is limited to only two clients. The existing Chat Server System is only meant for transfer of messages from one client to the other. But the messages cannot be sent to all i.e. group of persons or clients. The work done by others are:

* It provides GUI for chatting.
* It is limited to only two clients.
* It provides list of all clients online to server.
* It provides emotions, smiley, logs.

Having all this they does not provided with any whiteboard utility. The work done by others are listed above. Now we have to work on multiclient chatting i.e. group chatting and whiteboard premium feature.

**2.2 BENEFITS**

The proposed system should have the following benefits:

* The Chat Server and Clients Interface should be as simple as possible so that they can be configured even by a naïve user.
* Server can maintain a list of Clients and list of the clients who are currently online.
* Server should be able to create facility for one to one communication and multiple user communication at once.
* Users can able to share their ideas by drawing.
* A user can able to save the chatting information if he feels conversation is important.

**2.3 PROBLEM SOLUTION**

To solve the inconveniences as mentioned above, an “**INTRANET CHATTING**” is proposed. The proposed system’s premier feature is its **whiteboard** drawing utility. You can draw freehand, do circles, squares, lines, text, or paste image files to the canvas.  This is ideal when users want to "sketch" concepts for one another. This feature of “INTRANET CHATTING” can be a boon for the technical people who want to share their ideas or concepts in the pictorial form. This system includes the facilities of traditional chat servers and clients like providing a window for each Other user, Whisper chat, multiple chat rooms etc. With the help of the ‘WHITE BOARD’ drawing utility now the technical people can carry out their tasks easily and can share their big picture plans regarding their business to the clients, exchange ideas and concepts and many more things, basically exchange as well as share the information along with the using the drawing utility even long conversations can be made between two users which may be important business meetings or deals to be sanctioned and all this is carried out with the support of applets with the help of image based web menu images can be transferred.

**CHAPTER 3**

**FEASIBILITY STUDY**

**FEASIBILITY STUDY**

**3.1 TECHNICAL FEASIBLITY**

It is related with the technical specifications of the projects. It focuses on the three major questions: Hardware & software available in market. H/W & S/W required potential of the hardware.

Our project is technically feasible. That is the hardware and software for making this project is easily available in the market.

* Hardware required for this project is specified in the hardware specifications which is feasible.
* Software required to make this is NetBeans IDE 6.0.1, Java toolkit, IBM Rational Rose. This is available to us. Thus its software requirement is also fulfilled.

Thus, this is Technically feasible project. Technical feasibility centers on the existing computer system (hardware, software, etc.) and to what extent it can support the proposed addition. For example, if the current computer is operating at 80% capacity-an arbitrary ceiling- then running another app. could overload the system or require additional hardware.

**3.2 ECONOMICAL FEASIBLITY**

It deals with the cost benefit analysis. It includes the expenses like H/W, S/W, installation cost, furniture, fixture, wages, salaries etc. Economic analysis is most frequently used method for evaluating the effectiveness of candidate system. More commonly known as cost and benefit analysis.

After the analysis of cost, this project is economically is also feasible. Now we can move on further development of this project.

**CHAPTER 4**

**SOFTWARE ENGINEERING APPROACH**

**SOFTWARE ENGINEERING APPROACH**

**4.1 REQUIREMENT ANALYSIS**

**4.1.1 Software Engineering Paradigm Applied**

The system uses linear sequential model. The linear sequential model or the waterfall model is a systematic and sequential approach to software development that begins at the system level and progresses through following phases:

In this phase of our project, the requirements of the system are analyzed which enables to understand the nature of the software to be built. The information domain, required, function, behavior, performance and interfaces are studied.

**Design:**

In this phase we had to decide the data structure, software architecture, interface and algorithmic details of the project. Then this design is translated into coding in the next step.

**Coding:**

In coding phase, we can implements the language packages containing various classes and instruction to develop the project.

**Testing:**

In this phase, new components that is which are not the reusable ones and interfaces of the system had to be tested.

**Advantage and disadvantages**

Reasons for use:

There is no need for the feedback each and every time.

All requirements are well understood.

The requirements are explicitly known before the development of the project so we have used the linear sequential model for this project.

**4.1.2 Requirement analysis**

Requirements engineering provides the appropriate mechanism for understanding what, the customer wants, analyzing needs, accesing feasibility and negotiating reasonable solutions. The requirement engg process can be described in six distinct steps:-

**Requirements elicitation:** It consists of what the objective for the product is and finally how the product is to used. Requirement elicitation of three main problems:

1. Problem of scope
2. Problem of understanding
3. Problem of volatility

**Requirements analysis and negotiation:** Once requirement have been gather the work product noted earlier from the basis for requirements analysis categorizes requirements and organize them into related subsets, explores each requirement in relationship to others, examines requirements for consistency, omission and ambiguity and ranks requirement based on the need of the viewer.

**Requirement specification:** Requirement specification in the project means what are the various requirements which are necessary which are necessary to complete a project.

It consists of the following:

Hardware and software used

Technology

Team structure

Database

**System modeling:** In system modeling first a model is made based on all the analysis we have done. After that implementation is done.

**Requirement validation:** It examine the specification to ensure that all the system requirement have been stated unambiguously, the inconsistencies omission and errors have been detected and corrected, and that the work products conform to the standard established for the process, the project, and the produce.

**Requirement management:** Requirement management is a set activities that the help the project team to identify, control, and track requirements and changes to requirements at any time as the project proceeds, many of the activities are identical to the software configuration management techniques.

**4.1.2.1 Requirement Specification**

**Normal requirements**

These are the basic requirement that the project must provide. Normal requirement are the objective of goals that are stated for the product of system during meetings with the customer. If requirements are fulfilled, the customer is satisfied.

**Expected requirements**

* User friendly interface
* Portable
* Small in size
* Less costly
* Easy maintenance
* Correctness
* Efficient

**4.1.2.2 Functional Requirements**

The functional requirements describe the interaction between the system and the environment, for our project the functional requirements may be follows:-

Human Requirements

The human requirements for the software refer to the number of the person required for developing and maintaining the product. However, our team for the project development consists of three members. The names of team members are

1. Ms. Bhavna Kushwah (Project Manager)
2. Ms. Mamta Bhamre (Project Leader)
3. Mr. Vaibhav Namdev (Project Engineer)

Other than the member our project guide and software Engineering Approach guide Mr.Brajesh Chaturvedi has been a great help at various stages of project development life cycle.

**4.1.2.3 Non-Functional Requirements**

The non-functional requirements describe a restriction on the system that limits out choice for developing a solution to a problem.

The non-functional requirements in our project are:

* Time

The project should be completed within the stipulated time period.

* Cost

The cost involved in making the project in the less.

**General Types of Requirement**

Physical Environment

a) The equipment for the project is located at the college Swami Vivekanand College of Engineering and at the home of the developers.

b) There are no environmental limitations such as temperatures, etc.

* Documentation

Documentation in the system must be such that if some of new developers wish to enhance the system, they are unable to comprehend the code that has been previously.

The documentation should be present in both the soft copy and in terms of printed matter.

The documentation explains the architecture of the system, how connection is established and how communication done.

* Resources

The resources needed to build and maintain the system include the platform, on which it will be developed, JDK 1.5, the Window Operating System, computer system to develop and test the project.

The developers must be well acquainted with the intricacies of JDK 1.5 and must be able to program using sockets and awt.

The computer system must have the necessary power supply requirements and must also have a device such as constant voltage stabilizers to stabilize the power supply to computer to avoid damage to the systems during fluctuations.

The project is required to be completed with time span 1-1/2 month.

**4.2 ANALYSIS AND DESIGN**

**Project Analysis**

**1) Study of the System**

This application can be mainly divided into two modules:

1. Server

2. Client

This project “INTRANET CHATTING” is mainly depended on client/server model. The client requests the server and server responses by granting the clients request.

The proposed system should provide both of the above features along with the followed ones:

**Server:** The server should be able to perform the following features:

The first and foremost problem is to find the server. We should identify the program in the server which processes the client’s request.

Administrator Client who will be acting as a super user.

**Client:** The client should be able to perform the following features:

Should be able to send message to anybody in the room with clients unique chat name created in the server for chatting purpose.

Should be provided with the drawing tools like free hand, rectangle, oval, line and also sending text message over the room.

In all the network applications, we find two sort program where the first i.e., server sends the information and the second i.e., client receives the information.

**2) Input and Output:**

The user has to provide to which server it has to connect. In this, the server name to be provided is local host.

The user has to provide the username and the password for proceeding with the server for chatting purpose.

**4.2.1 Data Flow Diagrams**

****

* Client requests service to server and Server grants the request through a response.

****

* Client has to first register himself in the server to begin chatting. Server encrypts the PSWD and client is registered, welcome message prompted by server.





* Client can send offline messages to other clients, server stores and forwards the messages when other users log on.



* Client logs out of chat, server notifies & updates all other users by a message.



* Server shuts down by sending message and closes connection by giving a prompt.

**4.3 DIAGRAMS**

**4.3.1 Use case Diagrams**

System

****



System

System

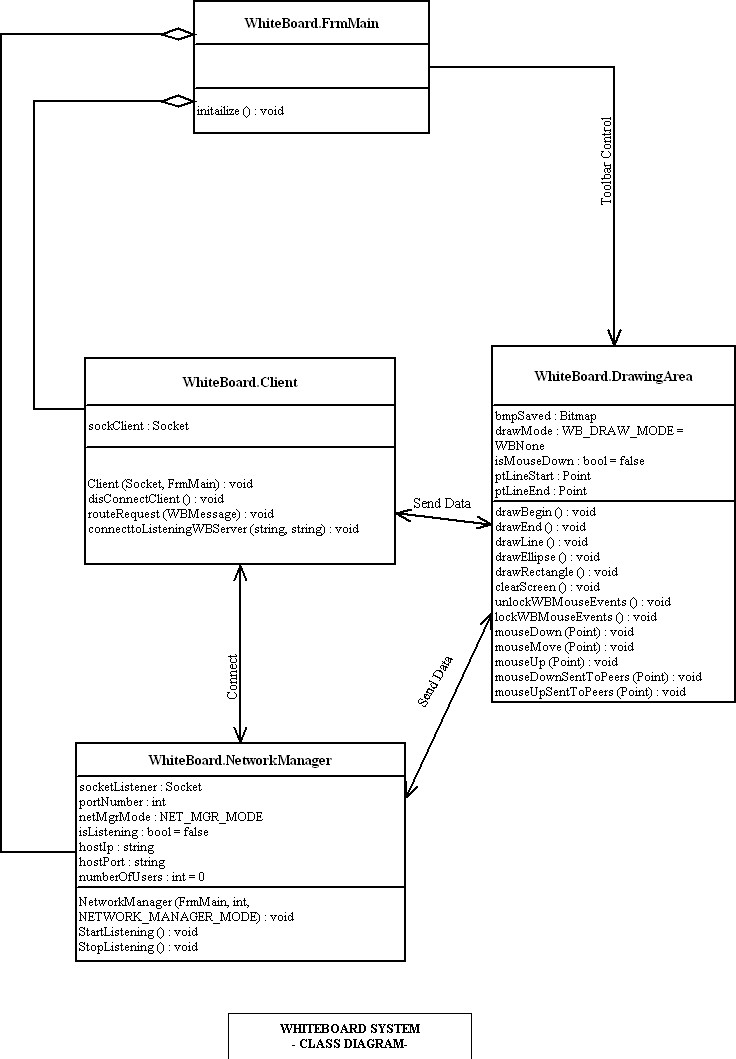


System



**4.3.2 Class Diagram**





**4.3.3 Sequence Diagram**

|

|

|

|

|

|

|

[validMssg]

mssgType := getMssgType()

aServerSocket

\* write()

aClientSocket

**X**

aSendMessageWindow

new

new

|

|

|

|

|

|

|

|

|

|

|

|

|

|

aLogEntry

\* recieve()

aBroadcastMssg

|

|

|

|

|

|

|

client

Message()

new

**4.3.4 Activity Diagram**

logout

kill

kill

\* listenToChatter()

sendToServer()

mssgRequest()

\* listenToServer()

postToClient()

recieve()

|

|

|

|

|

|

|

|

|

|

|

|

|

|

aChatClientSender

new

aClientListener

new

aChatClient

**X**



Act use case model

**4.3.5 Collaboration Diagram**

1.1.3: new

1.1.2 getMssgType()

1.1.1 \* recieve()

: LogEntry

:BroadcastMssg

:ServerSocket

1.1 \* write()

:ClientSocket

1 :clientMessage()

:SendMessageWindow

1.1.4: [validMssg]:new

**CHAPTER 5**

**IMPLEMENTATION PHASE**

**5.1 CODE EXPLANATION**

The INTRANET CHATTING application is developed used awt (Abstract Window Toolkit).

The **java.awt** package is much useful for creating user interfaces and for painting graphics and images. A user interface object such as a button or a scrollbar is called, in AWT terminology, a component. The Component class is the root of all AWT components. Some components fire events when a user interacts with the components. A container is a component that can obtain components and other containers. A container can also have a layout manager that controls the visual placement of components in the container.

The **java.awt** package implements different interfaces like **Layout Manager**, which defines the interface for classes that know how to layout Containers.

**Paint** interface defines how color patterns can be generated for Graphics2D operations. A class implementing the Paint interface is added to the Graphics2D context in order to define the color pattern used by the draw and fill methods.

The **java.net package** provides the classes for implementing networking applications. Using the socket classes, one can communicate with any server on the Internet or implement their own Internet server. A number of classes are provided to make it convenient to user Universal Resource Locators (URLs) to retrieve data on the Internet.

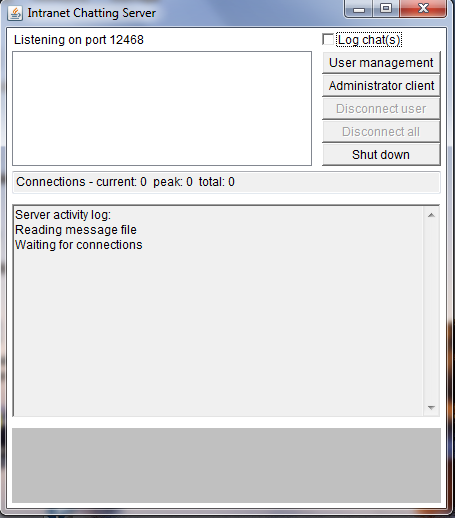
**Intranet Chatting Documentation**

The java.net package implements different interfaces like **DatagramSocketImplFactor*y*** for implementing data gram socket implementations. Classes DatagramSocket to create actual socket implementation use it.

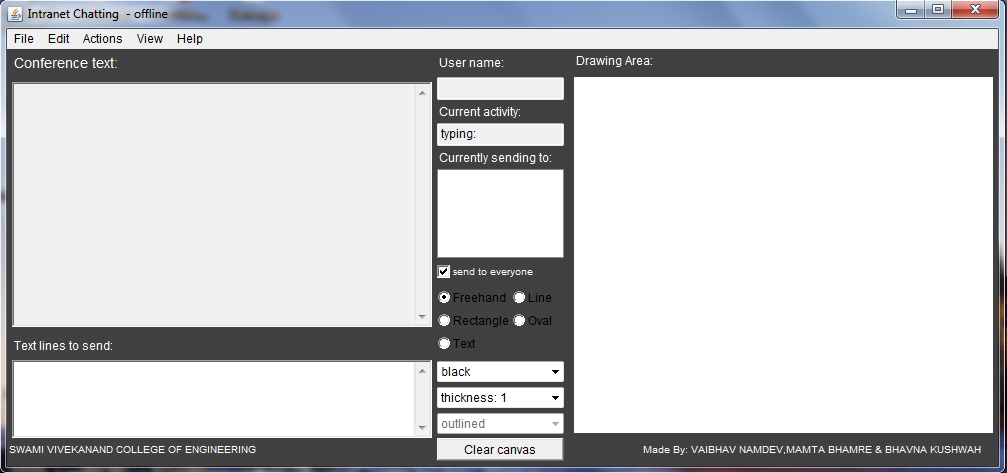
**SocketImplFacotry** interface defines a factory for Socket implementations. It is used by the classes socket and ServerSocket to create actual socket implementations.

**SocketOptions** interface of methods to get/set socket options. Is implemented by SocketImpl and DatagramSocketImpl.

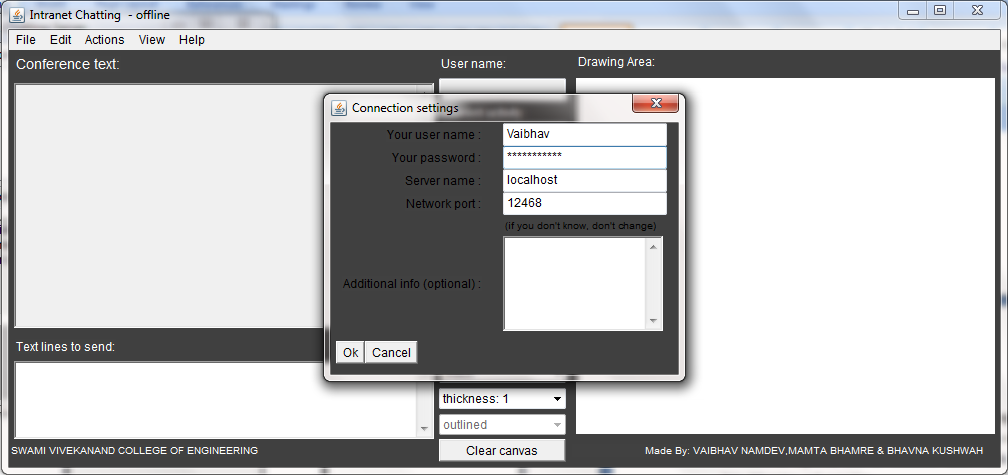
**5.2 OUTPUT SCREENS AND VALIDATION**

****

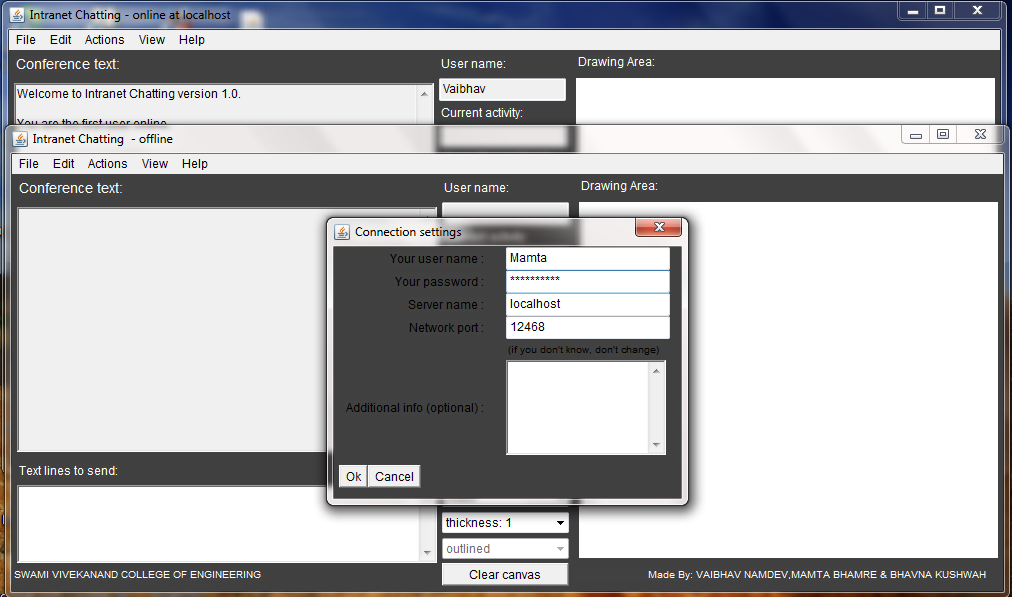
Server started



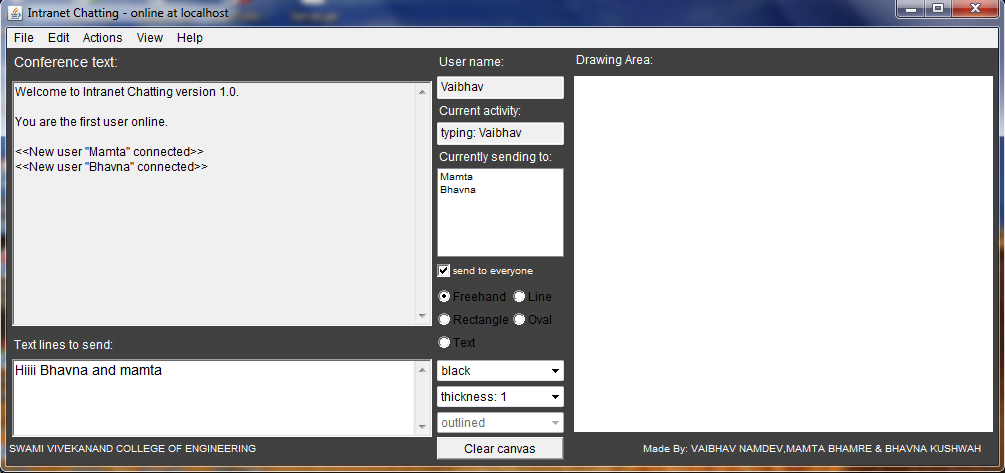
Client started in another window



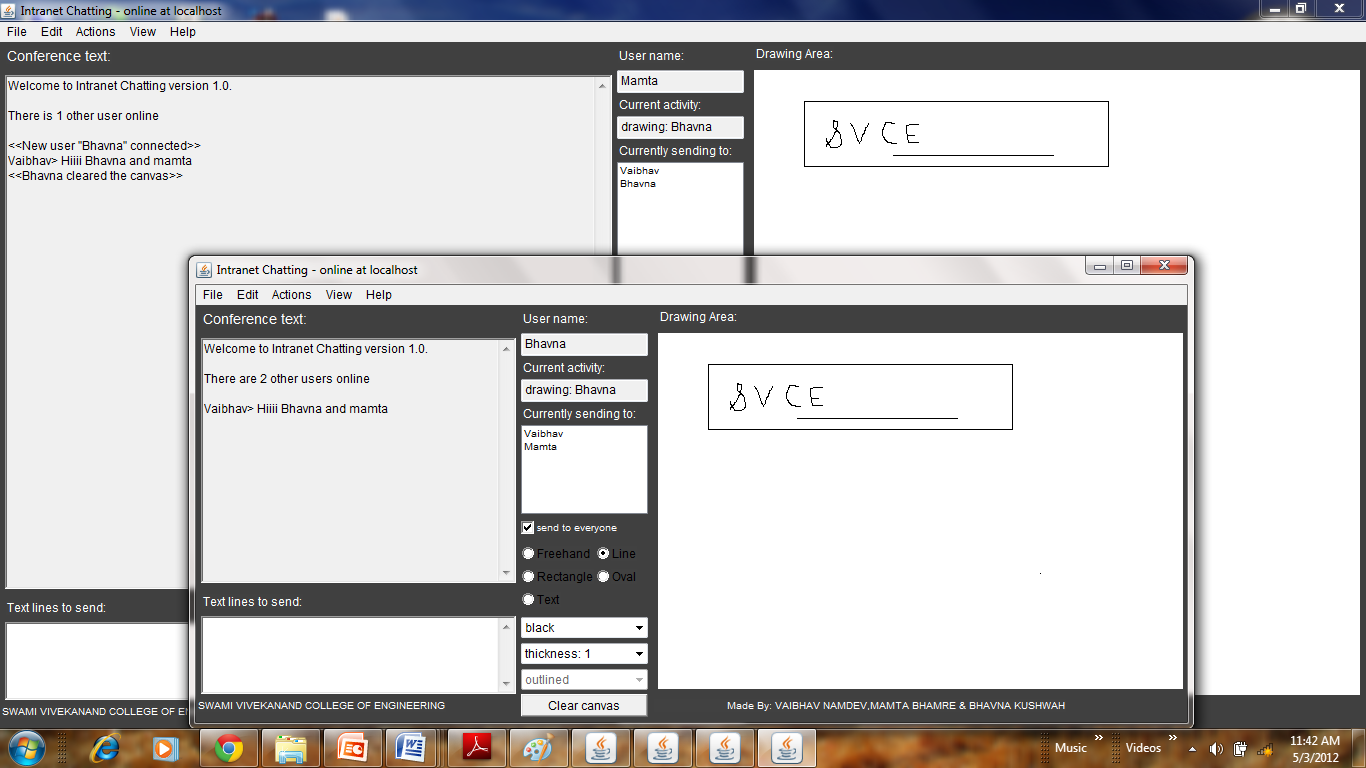
Client login to server



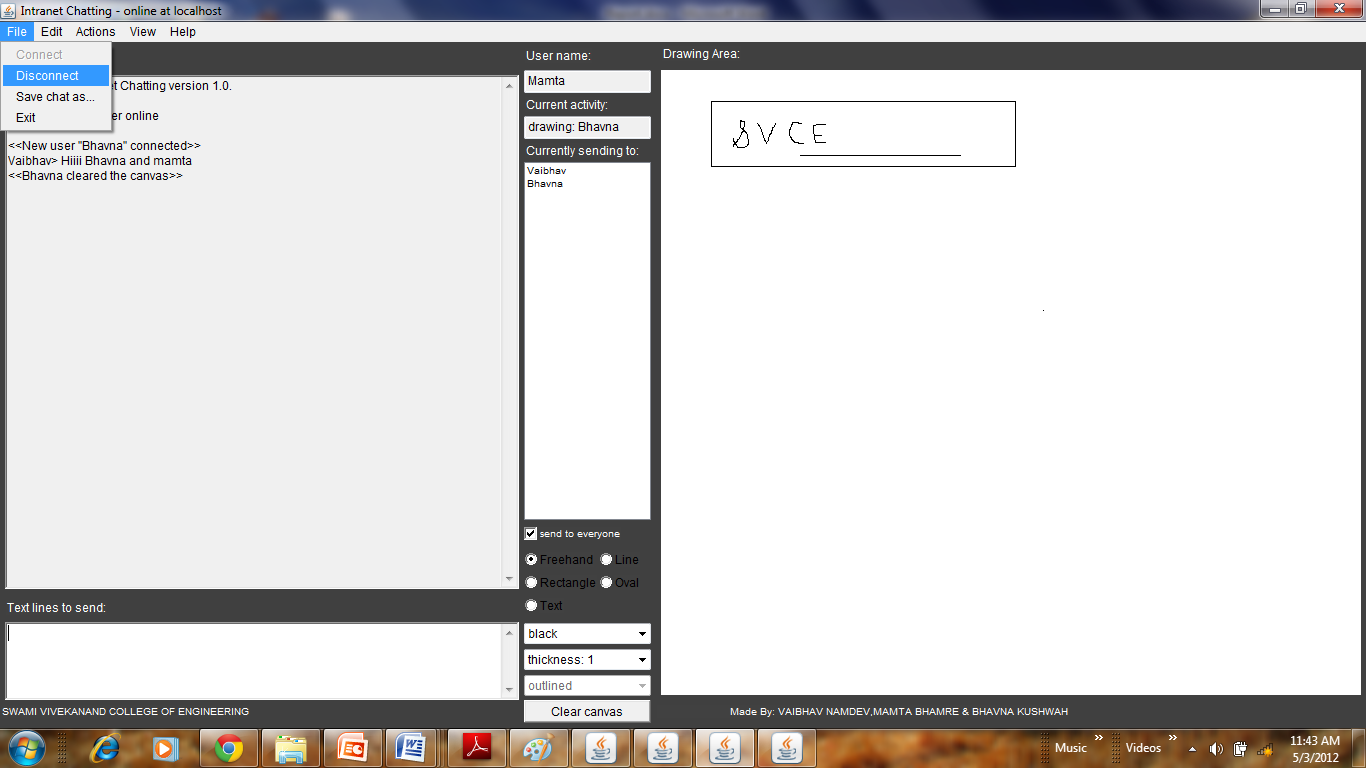
Another client login to server

****

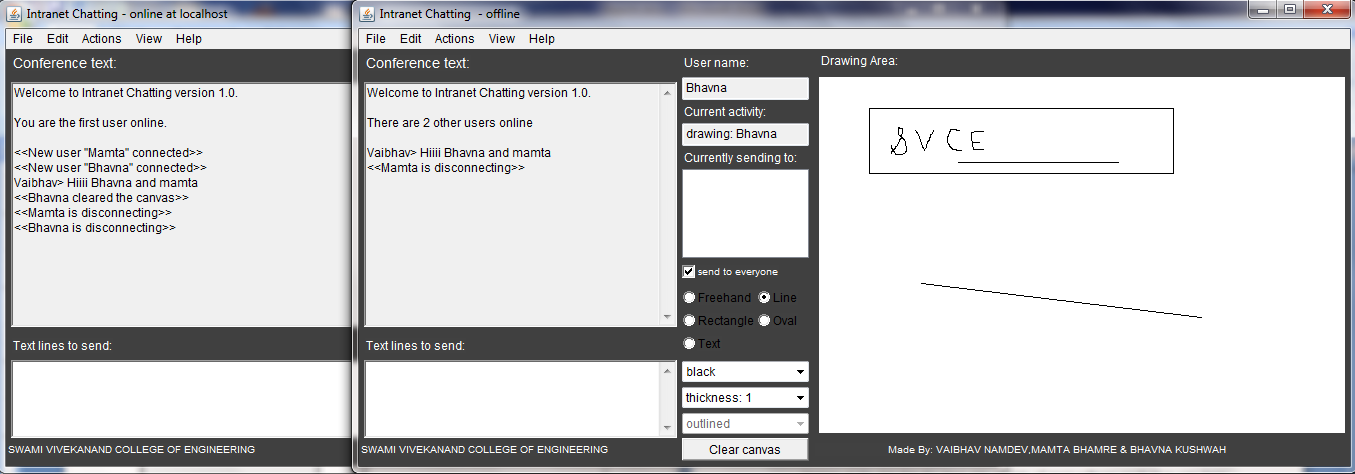
User 1 “Vaibhav” typing text



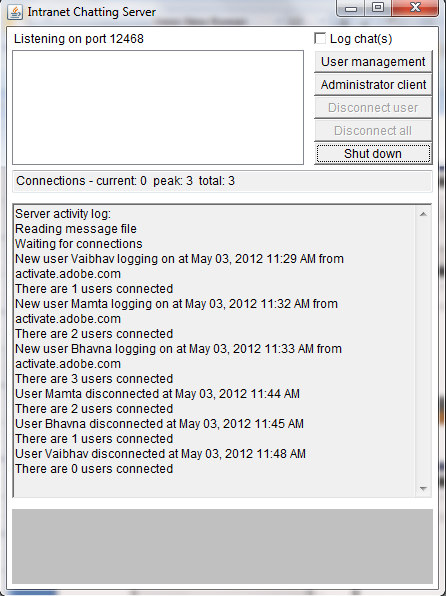
Sending picture



User requesting server to disconnect himself



Both the user get disconnected



Users disconnected from the server successfully

**5.3 TESTING**

Testing a program consists of providing the program with a set of test inputs (or test cases) and observing if the program behaves as expected. If the program fails to behave as expected, then the conditions under which failure occurs are noted for later debugging and correction.

Some commonly used terms associated with testing are:

• **Failure:** This is a manifestation of an error (or defect or bug). But, the mere presence of an error may not necessarily lead to a failure.

• **Test case:** This is the triplet [I,S,O], where I is the data input to the system, S is the state of the system at which the data is input, and O is the expected output of the system.

• **Test suite:** This is the set of all test cases with which a given software product is to be tested.

**5.3.1 TESTING OBJECTIVES**

The aim of the testing process is to identify all defects existing in a software product. However for most practical systems, even after satisfactorily carrying out the testing phase, it is not possible to guarantee that the software is error free. This is because of the fact that the input data domain of most software products is very large. It is not practical to test the software exhaustively with respect to each value that the input data may assume. Even with this practical limitation of the testing process, the importance of testing should not be underestimated. It must be remembered that testing does expose many defects existing in a software product. Thus testing provides a practical way of reducing defects in a system and increasing the users’ confidence in a developed system.

**5.3.1.1 TESTING METHODS AND STRATEGIES**

**Black box testing**

In black box testing the structure of the program is not considered. Test cases are decided solely on the basis of the requirements or specifications of the program or module, and the internals of the module or the program are not considered for selection of test classes. In this section, we will present some techniques for generating test cases for black-box testing. White-box testing is discussed in the next section.

In black-box testing, the tester only knows the inputs that can be given to the system and what output the system should give. In other words, the basis for deciding test cases in functional testing is the requirements or specifications of the system or module. This form of testing obvious functional or behavioral testing.

The most obvious functional testing procedure is exhaustive testing, which as we have stated, is empirical. One criterion for generating test causes is to generate them randomly. This strategy has little chance of resulting in a set of test cases that is close to optimal (i.e. that detects the maximum errors with minimum test cases). Hence, we need some other criterion or rule for selecting test cases. There are no formal rules for designing test cases for functional testing. In fact, there are no precise criteria for selecting test cases. However, there have been found to be very successful in detecting errors. Here we mention some of these techniques.

**White box testing**

In the previous section we discussed black-box testing, which is concerned with the function that the tested program is purposed to perform and does not deal with the internal structure of the program responsible for actually implementing that function. Those black-box testing is concerned with functionality rather than implementation of the program. White-box testing, on the other hand is concerned with testing the implementation of the program. The intent of this testing is not to exercise all the different input or output conditions ( although that may be a by-product) but to exercise the different programming structures and date structures used in the program. White-box testing is also called structure testing, and we will use the two terms interchangeably. To test the structure of a program, structural testing aims to achieve test cases that will force the desired coverage of different structures. Various criteria have been proposed for this. Unlike the criteria for functional testing which are frequently imprecise, the criteria for structural testing are generally quite precise as they are based on program structures, which are formal and precise.

**5.3.1.2 TEST CASE**

**Compiling Test**

It was a good idea to do our stress testing early on, because it gave us time to fix some of the unexpected deadlocks and stability problems that only occurred when components were exposed to very high transaction volumes.

**Execution Test**

This program was successfully loaded and executed. Because of good programming there were no execution errors. The complete performance of the project “INTRANET CHATTING” was good.

**Output Test**

The successful output screens are placed in the output screens section above with brief explanation about each screen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.no. | Test Case | Error | Result | Remark |
| 1 | Connection to server | No | Success | No response if load to server |
| 2 | Text sent | No | Success | - |
| 3 | Whiteboard Drawing | No | Success | Small problem in multiclient |
| 4 | Send to all | No | Success | - |
| 5 | Picture from file | No | Success | - |
| 6 | Clear canvas | No | Success | Cleared at all client ends. |
| 7 | Server Shut down | No | Success | - |
| 8 | Connection details | Yes | - | Not maintained |
| 9 | Help text file | Yes | - | Text file not opening |
| 10 | Log Chat at server | No | Success | Not always |
| 11 | Connection to various PC | No | Success | IP configuration |
| 12 | Wi-Fi Connectivity | - | - | - |
| 13 | Icon | Yes | - | Not loaded |

**CHAPTER 6**

**CONCLUSION AND DISCUSSION**

**CONCLUSION AND DISCUSSION**

**6.1 LIMITATIONS OF PROJECT**

1. It is only used for intranet chatting and not internet.
2. It is limited to certain area else its performance will be decreased.
3. It’s limited up to certain clients only for handling more load server computer should be good configuration.
4. Multiple client requesting server at the exact same time can put the processes in deadlock and server will shut down or connection will be failed.
5. This application does not provide any sort of authentication for user.

**6.2 DIFFICULTIES ENCOUNTERED**

1. Connection of multiple clients to the server.
2. Maintaining log of the chat and the log of the users’ session.
3. Creating whiteboard and send drawing to other client.

**6.3 FUTURE ENHANCEMENT SUGGESTION**

1. Offline messages: User can send messages to friends even when they are offline.
2. File transferring and sharing: User can transfer one or more files to other users. A file can also be shared between two or more users.
3. We thought to add the facility of chat rooms.
4. Profile Database: The profile information of every user can be stored at the server.
5. Login Timeout: This feature allows the user to be logged in only for a specific time. After this time span ends, the user is automatically logged out.
6. This project can be enhanced by implementing different protocols and can be made more useful for varied clients according to the requirements of the client, it can also possible in future that each client in this globe has his own customized “INTRANET CHATTING”.
7. It can be enhanced in the field of voice chatting. Using VoIP protocol
8. It can be enhanced in the field of Video Conferencing.

**CHAPTER 7**

**BIBLIOGRAPHY AND REFERENCES**

**BIBLIOGRAPHY AND REFERENCES**

* 1. **WEB AND BOOK REFERENCE**
  2. **OTHER DOCUMENTATION AND RESOURCES**