Chat Application

A SUMMER TRAINING REPORT

Submitted by

PALAK GARG

Roll No: 1140873

In partial fulfilment of Summer Training for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER ENGINEERING

National Institute of Technology

Kurukshetra, Haryana

ACKNOWLEDGEMENT

If words are considered as a symbol of approval and token of appreciation, then let the words play the heralding role expressing my gratitude.

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. I am grateful to my teacher **Mr. Gurusharan Rawal** for the guidance, inspiration and constructive suggestions that helped me in the preparation and eventually completion of this project.

I would also like to thank my brother who gave me the direction of making this project and coming up with this idea whose constant guidance and motivation is always helpful for me.

Lastly, I would like to thank the almighty and my parents for their moral support and my friends with whom I shared my day-to-day experience and received lots of suggestions that improved my quality of work.

PALAK GARG

DECLARATION

I, Palak Garg, student of B.Tech. 5th Semester, studying at NIT Kurukshetra, hereby declare that the summer training report on "Chat Application named Ping Me" submitted to Computer Engineering Department, NIT Kurukshetra in partial fulfilment of Degree of Bachelor of Technology is the original work conducted by me. The information and data given in the report is authentic to the best of my knowledge. This summer training report is not being submitted to any other University for award of any other Degree, Diploma and Fellowship.

Palak Garg

1140873 (CO-5)

NIT KURUKSHETRA

Table of Contents

1. ABSTRACT

PING ME

PING ME is a chatting application where the users can chat in groups as well as persona,l increasing the capability of socket programming in JAVA. It stores the previous chats in the database which can be accessed using JDBC (JAVA Database Connectivity). The entire program code is written in Core JAVA. The project's GUI is made with the help of Swings framework in Core Java. The threading is used to update the available users at any moment which performs a check on the database after every 30 seconds which may lead to flickering of data sometimes. The chatting is made possible with the help of Socket programming in Core Java. The socket programming consists of two applications , namely the server and the client. The server starts its service on a port number given by the user and clients connect to the server on that port by getting the port address on the server table made in database. The entire chat is stored in chat tables created in database which stores the sender's name as well as the destination which can be a group or a person which helps to apply filter to show the chat on the chatting screen .

The proposed system was to combine the features of all chatting applications and prepare a chatting application which is a complete in itself and try to remove as many disadvantages of other chatting applications which enables any type of file transfer, voice etc.

The existing project has tried to combine as many features as I can but due to limitation of time , only some have been added. The remaining features will be added to make it the best chatting application.

2. INTRODUCTION

.2.1PROJECT DEFINITION

Chatting is a method of using technology to bring people and ideas "together" despite of geographical barriers. The technology has been available for years but the acceptance is quite recent.

My project is an example of chat application. It is made of two applications namely, Client Application, which runs on the user's PC and server application, which runs on any pc on the network. To start chatting client should get connected to server where they can practice two kinds of chatting in personal (between two users only) or in group (Amongst more than two people) and the last one is accomplished using JDBC (Java Database Connectivity).

2.2 PROJECT DESCRIPTION

HOME

It's the introductory page having options for user to sign up and sign in to the application.

SIGN UP

For the new user to register for accessing this application, here he/she shall provide the necessary details First Name, Surname, Date of Birth , Mobile Number and Gender and then provide an appropriate username and a password and then ultimately sign up after filling up all the details.

User can also reset the fields.

SIGN IN

Here user is required to provide his/her username and password to log in, if already registered. Or in case user has forgot his/her password then user is redirected to **FORGOT PASSWORD** page.

FORGOT PASSWORD

On this page user can enter his username and date of birth provided during the registration to reset the old password. User can also reset the fields.

If the entered details are correct then a new page will open up that is **NEW PASSWORD**.

NEW PASSWORD

Here the user's username will be displayed and option to set the new password only after confirming it once.

User can also reset the fields.

MAIN SCREEN

This is the first page that appears after signing in. It has a tabbed pane with following 3 tabs:

- ONLINE -- It shows the list of users which are available at the time when the user has signed in(just like Facebook).
- CHATS – It shows the names of groups of which user is a part. Clicking on any group name leads to the chatting screen of that group showing the previous chats as well.
- CONTACTS – It shows the list of all users who have registered on the PING ME chat application.

The menu bar of the page has two menu options:

- SIGN OUT-It signs out the user redirecting him/her back to the home page.
- CREATE GROUP-It gives an option to the user to create a group where he/she can add other members who have registered on the PING ME.

CREATE A GROUP

To create a group, this page appears which contains a text box where you can name your group and a list of all the users who have registered on the PING ME. Clicking on the users you want to add appears in other not editable textbox. At last there is a button , clicking on it creates your group and redirects you to main screen.

CHAT SCREEN

This is the window that appears while chatting . At top, there is a name showing the group name/ username with whom you are chatting. There is a not editable text showing your chats and an editable textbox to type your message and a send button to send your message.

3. TECHNICAL BACKGROUND

3.1 SOFTWARE REQUIREMENTS

Software can be defined as program which run on our computer. It acts as petrol in the vehicle. It provides a relationship between a human and a computer. It is very important to run software for functioning of a computer. Various softwares are required for development of this project, which are as follows:

- Operating System : Windows XP and above
- Languages : java
- Front End : Swings(Java)
- Back End : Oracle 10g XE

3.2 HARDWARE REQUIREMENTS

Here we require all those components which provide us the platform for the development if the project. The minimum hardware required for the development of this project is as follows:

- Processor : Intel Core2Duo/AMD A8
- RAM : 512 MB
- Hard disk : 20 GB
- Monitor : 14 inch
- Mouse : 2 Button scroll
- Keyboard : 108 keys

3.3 JAVA

Java is a small, simple, safe, object oriented, interpreted or dynamically optimized, byte coded, garbage collected, multithreaded programming language with a strongly typed exception-handling for writing distributed and dynamically extensible programs.

Java is an object oriented programming language. Java is a high-level, third generation language like C, FORTRAN, Small talk, Pearl and many others. Java can be used to write computer applications that make programs, play games, store data or do any of the thousands of other things computer software can do.

- It is simple and object oriented
- It helps to create user friendly interfaces.
- It supports multithreading.
- It is platform independent
- It can have connection with database.
- It supports internet programming

Java is a programming language originally developed by Sun Microsystems and released in 1995 as a core component of Sun's Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code which can run on any Java virtual machine (JVM) regardless of computer architecture. The Java language was created by James Gosling in June 1991 for use in a set top box project.

The Java platform is the name for a bundle of related programs, or platform, from Sun which allow for developing and running programs written in the Java programming language. The platform is not specific to any one processor or operating system, but rather an execution engine (called a virtual machine) and a compiler with a set of standard libraries which are implemented for various hardware and operating systems so that Java programs can run identically on all of them.

Different "editions" of the platform are available, including:

- Java ME (Micro Edition): It's a java platform designed for embedded systems, mobile devices are one kind of such devices. Java ME, designed by Sun Microsystems, acquired by Oracle, provides a robust, flexible environment for the applications.
- Java SE (Standard Edition): For general purpose use on desktop PCs, servers and similar devices.
- Java EE (Enterprise Edition): Java SE plus various APIs useful for multi-tier client-server enterprise applications.

The Java Platform consists of several programs, each of which provides a distinct portion of its overall capabilities. For example, the Java compiler converts Java source code into Java bytecode. The sophisticated Java Runtime Environment (JRE), complementing the JVM with a just-in-time (JIT) compiler, converts intermediate bytecode into native machine code on the fly. Also supplied are extensive libraries (pre-compiled into Java bytecode) containing

reusable code, as well as numerous ways for Java applications to be deployed, including being embedded in a web page as an applet. There are several other components, some available only in certain editions.

The essential components in the platform are the Java language compiler, the libraries, and the runtime environment in which Java intermediate byte code "executes" according to the rules laid out in the virtual machine specification.

SOCKET PROGRAMMING:

Sockets provide the communication mechanism between two computers using TCP. A client program creates a socket on its end of the communication and attempts to connect that socket to a server.

When the connection is made, the server creates a socket object on its end of the communication. The client and the server can now communicate by writing to and reading from the socket.

The java.net.Socket class represents a socket, and the java.net.ServerSocket class provides a mechanism for the server program to listen for clients and establish connections with them.

The following steps occur when establishing a TCP connection between two computers using sockets −

- The server instantiates a ServerSocket object, denoting which port number communication is to occur on.
- The server invokes the accept() method of the ServerSocket class. This method waits until a client connects to the server on the given port.
- After the server is waiting, a client instantiates a Socket object, specifying the server name and the port number to connect to.
- The constructor of the Socket class attempts to connect the client to the specified server and the port number. If communication is established, the client now has a Socket object capable of communicating with the server.
- On the server side, the accept() method returns a reference to a new socket on the server that is connected to the client's socket.

After the connections are established, communication can occur using I/O streams. Each socket has both an OutputStream and an InputStream. The client's OutputStream is connected to the server's InputStream, and the client's InputStream is connected to the server's OutputStream.

TCP is a two-way communication protocol, hence data can be sent across both streams at the same time.

JDBC

There are following six steps involved in building a JDBC application −

- Import the packages**:** Requires that you include the packages containing the JDBC classes needed for database programming. Most often, using *import java.sql.** will suffice.
- Register the JDBC driver**:** Requires that you initialize a driver so you can open a communication channel with the database.
- Open a connection: Requires using the *DriverManager.getConnection()* method to create a Connection object, which represents a physical connection with the database.
- Execute a query**:** Requires using an object of type Statement for building and submitting an SQL statement to the database.
- Extract data from result set**:** Requires that you use the appropriate *ResultSet.getXXX()* method to retrieve the data from the result set.
- Clean up the environment**:** Requires explicitly closing all database resources versus relying on the JVM's garbage collection.

MULTITHREADING:

Java is a *multi-threaded programming language* which means we can develop multi-threaded program using Java. A multi-threaded program contains two or more parts that can run concurrently and each part can handle a different task at the same time making optimal use of the available resources specially when your computer has multiple CPUs.

By definition, multitasking is when multiple processes share common processing resources such as a CPU. Multi-threading extends the idea of multitasking into applications where you can subdivide specific operations within a single application into individual threads. Each of the threads can run in parallel. The OS divides processing time not only among different applications, but also among each thread within an application.

Multi-threading enables you to write in a way where multiple activities can proceed concurrently in the same program.

3.4 DATABASE MANAGEMENT SYSTEM

A **database** is an organized collection of data. The data is typically organized to model aspects of reality in a way that supports processes requiring information. For example, modelling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

Database management systems (**DBMS**s) are computer software applications that interact with the user, other applications, and the database itself to capture and analyze data. A general-purpose DBMS is designed to allow the definition, creation, querying, update, and administration of databases. Well-known DBMSs include MySQL, PostgreSQL, Microsoft SQL Server, Oracle, SAP and IBM DB2. A database is not generally portable across different DBMSs, but different DBMSs can interoperate by using standards such as SQL and ODBC or JDBC to allow a single application to work with more than one DBMS. Database management systems are often classified according to the database model that they support; the most popular database systems since the 1980s have all supported the relational model as represented by the SQL language. Sometimes, a DBMS is loosely referred to as a "database".

Formally, "database" refers to the data themselves and supporting data structures. Databases are created to operate large quantities of information by inputting, storing, retrieving and managing that information. Databases are set up so that one set of software programs provides all users with access to all the data.

A "database management system" is a suite of computer software providing the interface between users and a database or databases. Because they are so closely related, the term "database" when used casually often refers to both a DBMS and the data it manipulates.

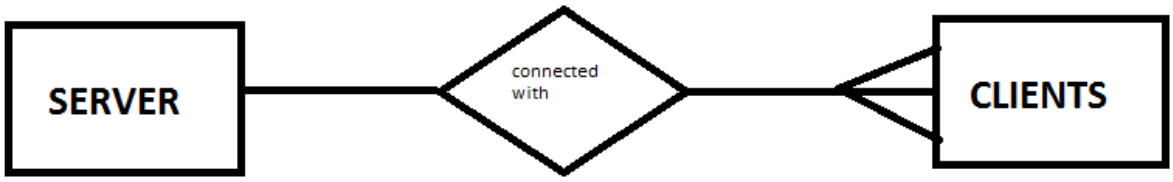
The interactions catered for by most existing DBMSs fall into four main groups:

- **Data definition** – Defining new data structures for a database, removing data structures from the database, modifying the structure of existing data.
- **Update** – Inserting, modifying, and deleting data.
- **Retrieval** – Obtaining information either for end-user queries and reports or for processing by applications.
- **Administration** – Registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control, and recovering information if the system fails.

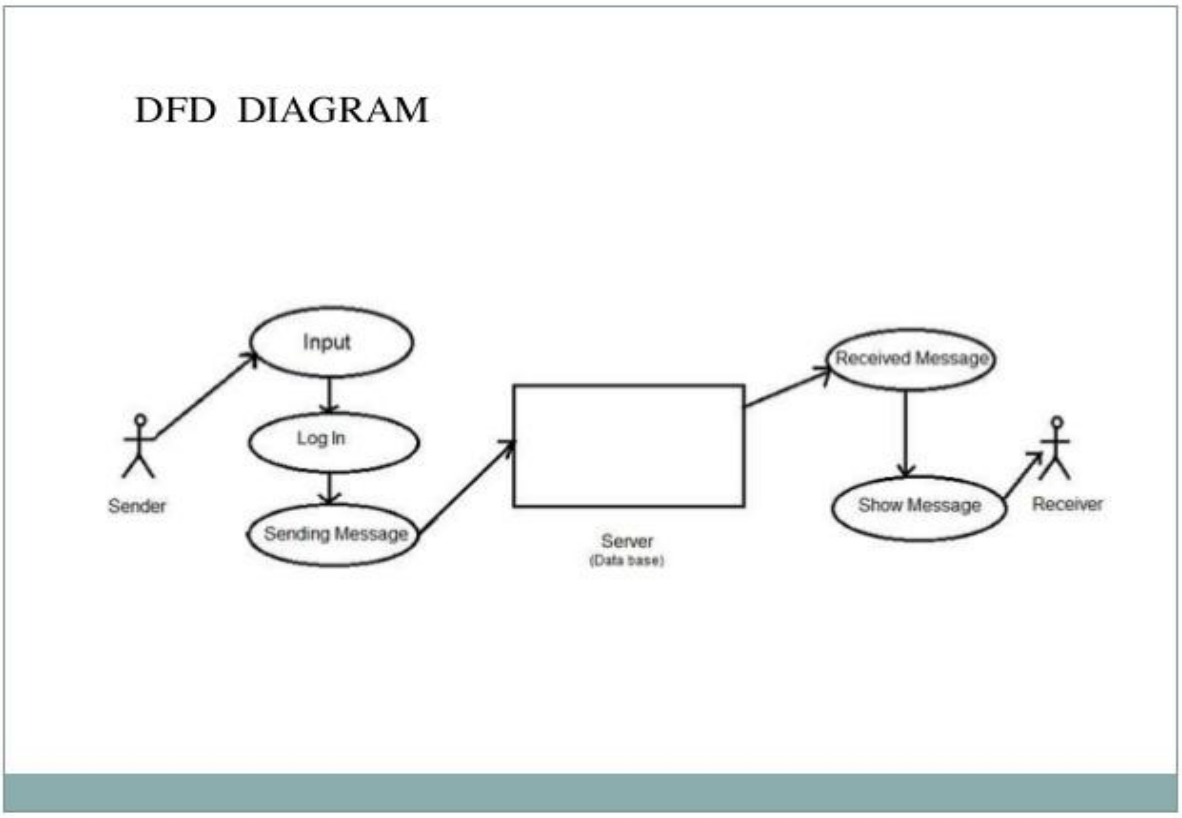
A DBMS is responsible for maintaining the integrity and security of stored data, and for recovering information if the system fails.

3. SYSTEM DESIGN

E-R DIAGRAM



DFD (DATA FLOW DIAGRAM)



5. SYSTEM ARCHITECTURE

8.1 THE SERVER

The server program begins by creating a new ServerSocket object to listen on a specific port. When running this server, choose a port that is not already dedicated to some other service.

The server program creates the ServerSocket object in a try-with-resources statement:

ServerSocket is a java.net class that provides a system-independent implementation of the server side of a client/server socket connection. The constructor for ServerSocket throws an exception if it can't listen on the specified port (for example, the port is already being used). In this case, the Server has no choice but to exit.

If the server successfully binds to its port, then the ServerSocket object is successfully created and the server continues to the next step—accepting a connection from a client by:

clientSocket = serverSocket.accept();

8.2 THE CLIENT

When you start the client program, the server should already be running and listening to the port, waiting for a client to request a connection. So, the first thing the client program does is to open a socket that is connected to the server running on the specified host name and port as the client in socket programming must know two information:

- IP Address of Server, and
- Port number.

8.3 THE DATABASE

PING ME uses the following tables for its working.

- Server
- Available
- Tblclients
- Grp
- Chat
- Gchat

SERVER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SERVER	PORT	NUMBER	-	20	0	-	✓	-	-
									1 - 1

This table stores the port number on which the server is available. All the clients need to access this port number in order to get services offered by the server.

AVAILABLE

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
AVAILABLE	CLIENTS	VARCHAR2	20	-	-	-	✓	-	-
									1 - 1

This table stores the usernames of all the clients who have logged at any moment. From this table , other users access the online users whe they sign in on their main screen.

TBLCLIENTS

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TBLCLIENTS	NAME	VARCHAR2	20	-	-	-	✓	-	-
	SURNAME	VARCHAR2	20	-	-	-	✓	-	-
	DOB	NUMBER	-	8	0	-	✓	-	-
	USERNAME	VARCHAR2	20	-	-	-	✓	-	-
	PASS	VARCHAR2	20	-	-	-	✓	-	-
	MOB	NUMBER	-	10	0	-	✓	-	-
	GENDER	VARCHAR2	10	-	-	-	✓	-	-
									1 - 7

The tblclients store the essential data(Name,Surname,DOB,username,password,mobile number,gender) of all the users registered on the PING ME.

GRP

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
GRP	GNAME	VARCHAR2	20	-	-	-	✓	-	-
	MEMBER	VARCHAR2	20	-	-	-	✓	-	-
									1 - 2

The group table stores the names of groups and their members .

CHAT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CHAT	SNAME	VARCHAR2	20	-	-	-	✓	-	-
	DNAME	VARCHAR2	20	-	-	-	✓	-	-
	CHAT	VARCHAR2	100	-	-	-	✓	-	-
									1 - 3

This table stores all the personal chats where one column has the sender's userid, other receiver's userid and the last one is chat.

GCHAT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
GCHAT	GNAME	VARCHAR2	20	-	-	-	✓	-	-
	SNAME	VARCHAR2	20	-	-	-	✓	-	-
	CHAT	VARCHAR2	100	-	-	-	✓	-	-
									1 - 3

This table stores the chatting done in various groups storing the chat as well as sender name and the group name as destination.

7. Conclusion

I have set up a client, a server and a database. The applications are done , except for the client which needs more enhancement but certain essential features have already been added. The standard functionality of the client, the server and the database is in place: users can register , log in, recover their forgotten passwords by first confirming authenticity by answering a question added in details while signing up, can see the available users at the time you logged in, the groups in which you are included, creating a group and adding members in it according to your choice, list of all users registered till date with the PING ME messenger, chat with other users either personally and to a single person or in a group, having the record of previous chats as well.

To sum it up:

- Client- enhancements needed
- Server- done
- Database- done

I am happy with my progress and project made till now. I think I have accomplished most of what I wanted to create. The project has a lot of potential for future development and enhancements.

8. FUTURE WORK

I will continue to work on this project. The most obvious future work will be:

- Adding and enhancing the client side.
    - Taking some question or other feature as a measure to recover the forgotten password.
    - Phone number authenticity can be added or e-mail authenticity while signing up.
    - Giving client the authority to leave the groups once added.
    - Clearing the previous chats in personal as well as in group chat.
    - Having the concept of group admin as in whats app where the group admin can remove or add other members and can also give this authority to other members in the group.
    - Giving suggestions of username while signing up.
    - Adding more details in signing up page.
    - Enabling image, file transfer via this messenger.
    - Checking password is strong enough or not.
    - Search the chat according to some particular work.
    - Setting your profile picture.
    - Enabling clients to change their profile picture.
- Polishing the GUI of applications.
- Adding security features.

9. REFERENCES

- JAVA Black Book

  by Steven Holzner et al

- JAVA Complete Reference

  by Herbert Schildt

- www.oracle.com

- https://en.wikipedia.org/wiki/**Java**_(software_platform)

- https://www.tutorialspoint.com

- http://www.w3schools.com/

Appendix: Screenshots

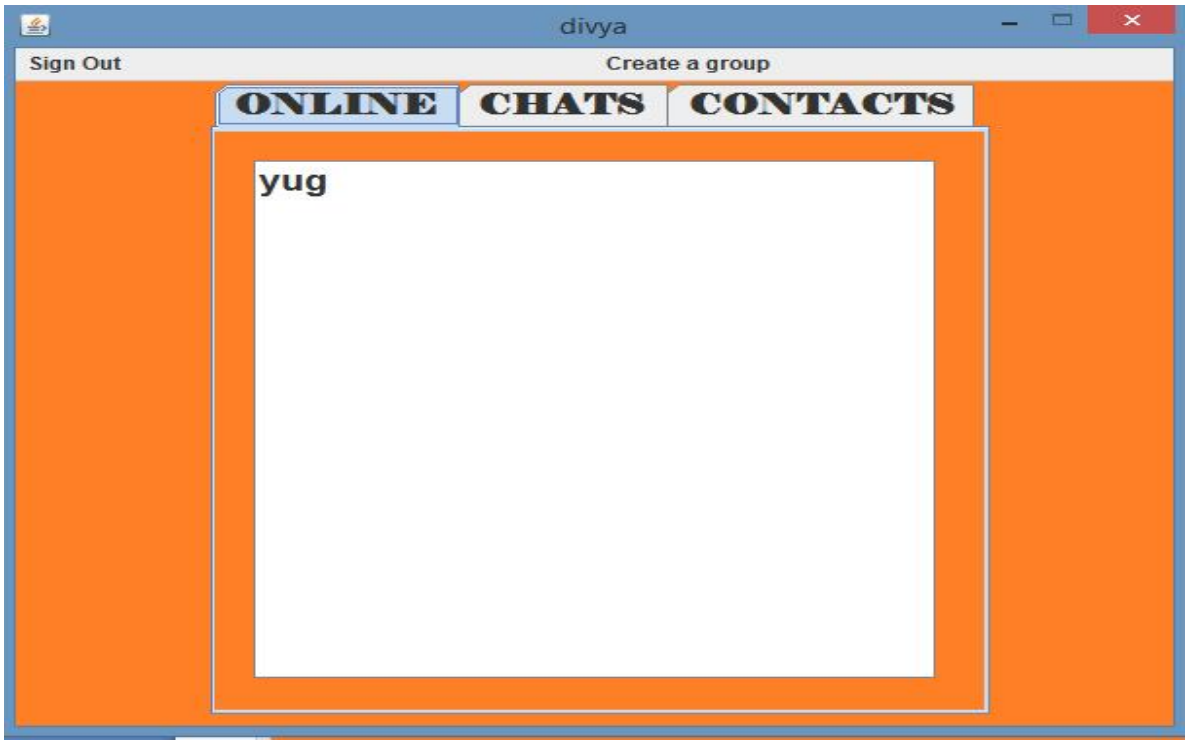
Screenshot 1: The home page of chat messenger

Screenshot 2: The Sign In Page

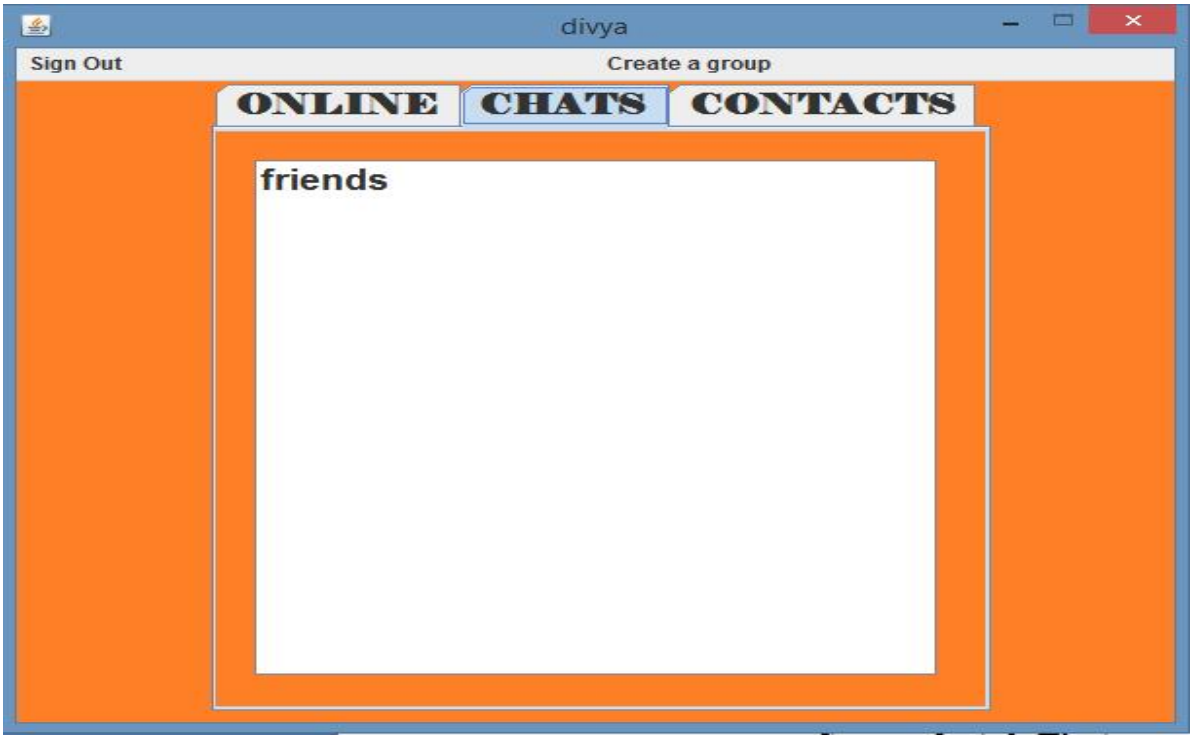
Screenshot 3: The Sign Up Page

Screenshot 4: DOB used to recover password (Forget Password Page)

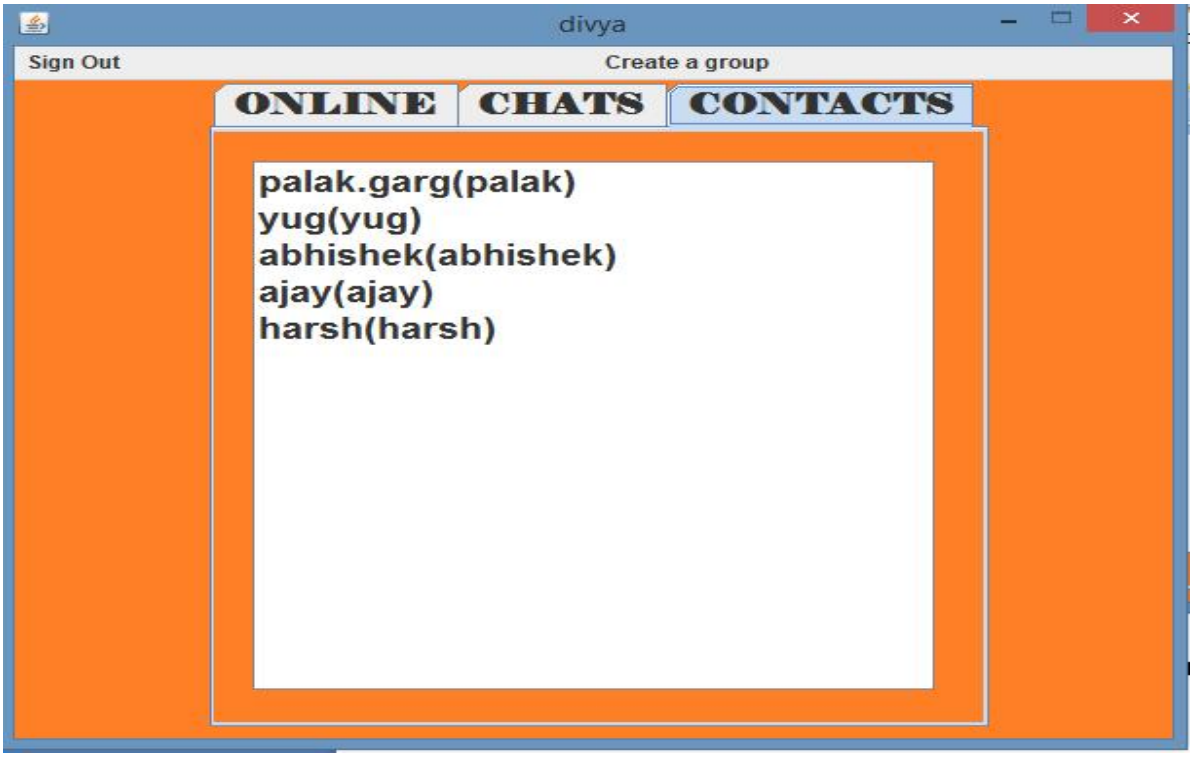
Screenshot 5: To set the new password after confirming DOB



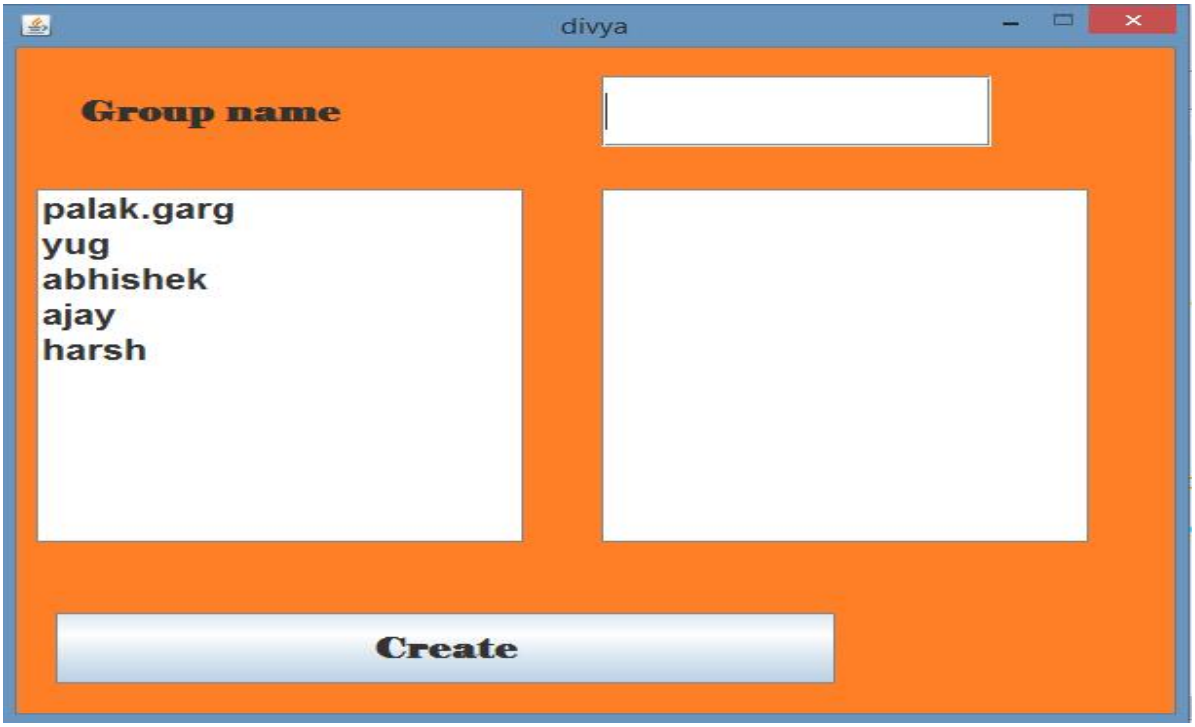
Screenshot 6: First page after logging in



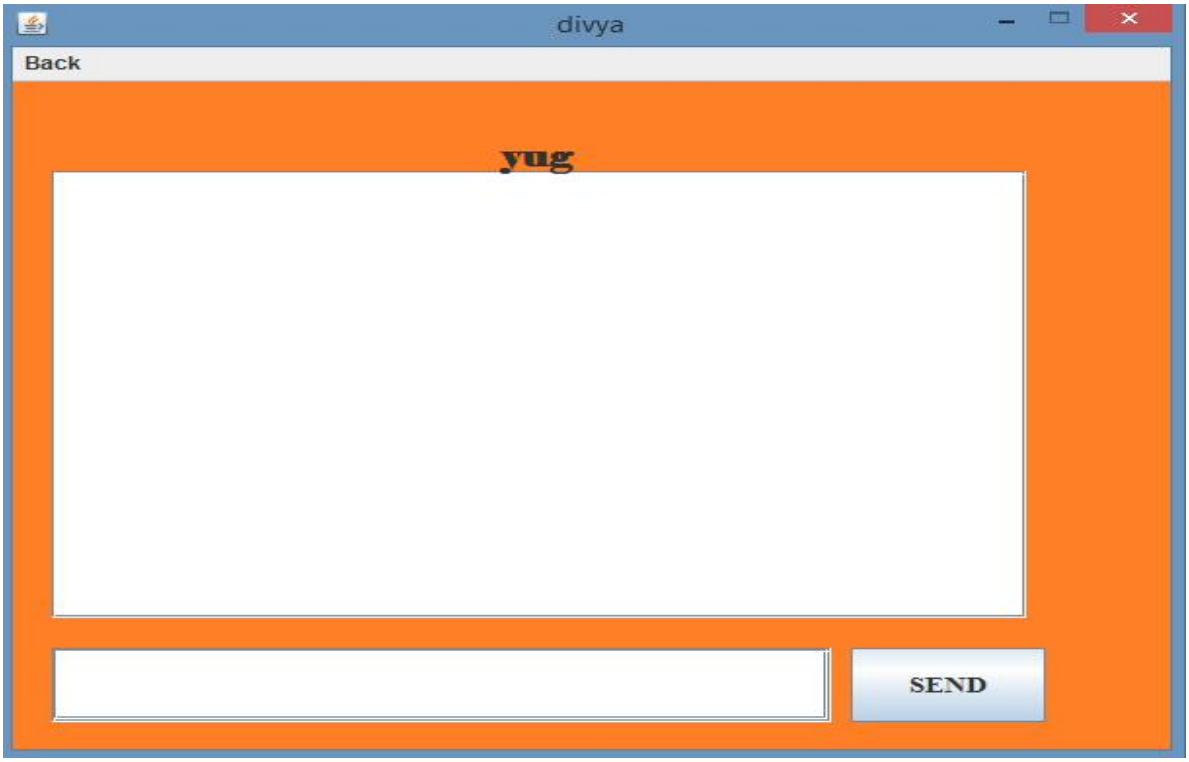
Screenshot 7: Chats tabbed pane showing your groups



Screenshot 8: Contacts tabbed pane listing all the usernames



Screenshot 9: Creating groups page



Screenshot 10: The chatting screen