

# Index

<b>Sr. No.</b>	<b>Contents</b>	<b>Page No.</b>
<b>1</b>	<b>Annexure I– Micro Project Proposal</b>	<b>02-04</b>
	1.Aims/Benefits of the Micro-Project	02
	2. Course Outcome Addressed	02
	3.Proposed Methodology	02
	4. Action Plan	03
	5. Resources Required	04
	6. Name of student with roll no	04
<b>2</b>	<b>Annexure II – Micro Project Report</b>	<b>5-15</b>
	1.Rationale	5
	2.Aims/Benefits of the Micro-Project	5
	3.Course Outcome Achieve	5
	4. Literature Review	5-6
	5.Actual Methodology Followed	06-14
	6.Actual Resources Used	15
	7. Skill developed / Learning out of this Micro-Project	15
	8. Applications of this Micro-Project	15

## Micro Project Proposal.

### CHATTING APPLICATION.

#### 1. Aims/Benefits of the Micro-Project:

1. To study how to develop Java Messenger using java frame.
2. To learn concepts of developing java application using java language.
3. To understand how Java Messenger is also useful to chatting, and study its works.

#### 2. Course Outcome Addressed:

1. Develop programs using GUI Framework (AWT and Swing).
2. Handle events of AWT and Swings components.
3. Develop programs to handle events in Java Programming.

#### 3. Proposed Methodology:

The Java Messenger application to build is a console application that is launched from the command line.

- i. **Server:-** To communicate with the client, the server uses two type of sockets.
- ii. **Server Socket :** this class is used by the server to declare a Server Socket object which the server needs to listen to connection requests from the clients
- iii. **Socket :** this class is used by the server to declare a Socket object, which the server uses to send and recieve data from the client.
- iv. **Client :** To communicate with the server, the client uses one single socket which's an object from Socket class, to send and recieve data from server => the instantiating of Socket class in server and client is different even though it is the same class but it differs from client to the server.

#### 4. Action Plan:

Sr. No.	Details of Activity	Planned Start date	Planned Finish date	Name of Responsible Team Members
1	Search the topic	25/08/2022 03:30 PM- 05:30 PM	01/09/2022 03:30 PM- 05:30 PM	More Ganesh Vishwanath
2	Search the information	08/09/2022 03:30 PM- 05:30 PM	15/09/2022 03:30 PM- 05:30 PM	
3	Find the course outcomes.	22/09/2022 03:30 PM- 05:30 PM	29/09/2022 03:30 PM- 05:30 PM	
4	Find the literature review	06/10/2022 03:30 PM- 05:30 PM	13/10/2022 03:30 PM- 05:30 PM	
5	Implementing the code.	20/10/2022 03:30 PM- 05:30 PM	27/10/2022 03:30 PM- 05:30 PM	
6	Debugging the code.	03/11/2022 03:30 PM- 05:30 PM	10/11/2022 03:30 PM- 05:30 PM	
7	Making Index and Certificate of project	17/11/2022 03:30 PM- 05:30 PM	24/11/2022 03:30 PM- 05:30 PM	
8	Finalizing Project with its report	28/11/2022 03:30 PM- 05:30 PM	08/12/2022 03:30 PM- 05:30 PM	

### 5. Resources Required:

Sr. No.	Name of resource / material	Specification	Quantity	Remarks
1	Computer	WINDOWS 11,2GB RAM, 160GB HDD	1	
2	Operating System	WINDOWS 11	1	
3	Idea for compiling.	CMD		
4	Browser	Chrome	1	

### 6. Names of Team Member with Roll No. :-

Sr. No.	Enrollment No.	Name of Team Member	Roll No.
1	2010950104	More Ganesh Vishwanath	27

**Mr. Sugare D.D.**

**Name and Signature of the Teacher**

## Micro-Project Report

### **CHATTING APPLICATION.**

#### **1. Rationale:**

Java technology is widely used for web applications development. Based on the object oriented concepts and core Java concepts, this course will equip the students with the required knowledge and skill of

object oriented programming approach needed for the development of robust, powerful web applications. Through this course students will get hands-on experience on GUI Technologies viz. AWT and Swings, event handling mechanisms and network programming. The course also gives coverage to various web applications aspects like Database Interaction, server side components and servlets.

#### **2. Aims/Benefits of the Micro-Project:**

1. To study how to develop media player using java frame.
2. To learn concepts of developing java application using java language.
3. To understand how Java Messenger is also useful to listeners, and study its works.

#### **3. Course Outcomes Achieved:**

1. Develop programs to handle events in Java Programming.
2. Develop programs using GUI Framework (AWT and Swing).
3. Handle events of AWT and Swings component.

#### **4. Literature Review:**

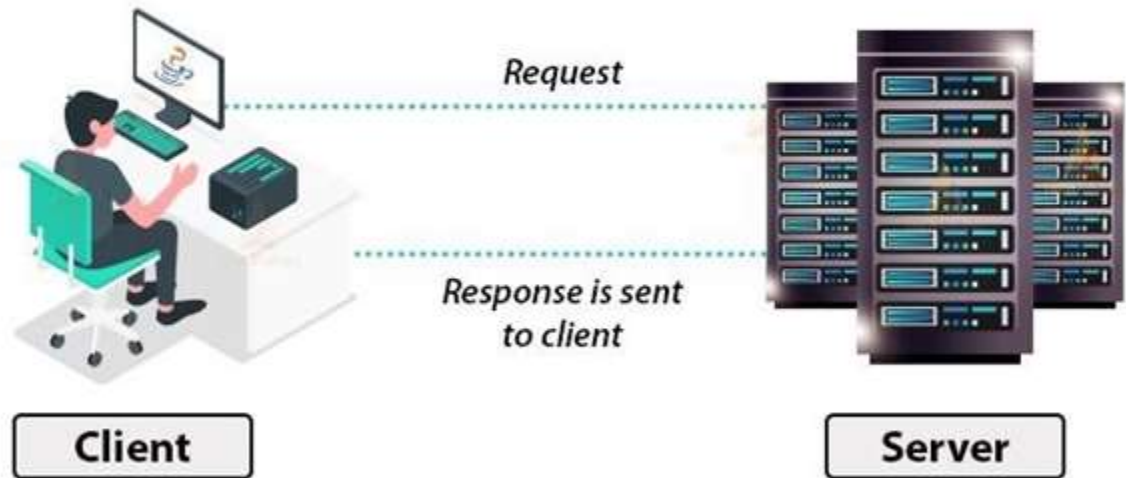
Instant Messaging has become an important means of communication between people around the globe, providing an alternative to telephone and email conversations. The number of users using Instant Messenger products has been increasing over recent years.

Messenger services provide a perfect environment for private and personal chat. People of all ages log into messenger service or chat rooms to spend time chatting with known and unknown

persons.

The aim of the project is to develop a client server java chat application with security in mind.

## Java Socket Programming Process



### 5.Actual Methodologies Followed:

🌈 Source code:-

#### Client:-

```
import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import java.text.*;
import java.net.*;
import java.io.*;
public class Client implements ActionListener {

    JTextField text;
    static JPanel a1;
    static Box vertical = Box.createVerticalBox();

    static JFrame f = new JFrame();
    static DataOutputStream dout;
    Client() {
        f.setLayout(null);
        JPanel p1 = new JPanel();
```

```

p1.setBackground(new Color(7, 94, 84));
p1.setBounds(0, 0, 450, 70);
p1.setLayout(null);
f.add(p1);
ImageIcon i1 = new ImageIcon(ClassLoader.getResource("3.png"));
Image i2 = i1.getImage().getScaledInstance(25, 25, Image.SCALE_DEFAULT);
ImageIcon i3 = new ImageIcon(i2);
JLabel back = new JLabel(i3);
back.setBounds(5, 20, 25, 25);
p1.add(back);

back.addMouseListener(new MouseAdapter() {
public void mouseClicked(MouseEvent ae) {
System.exit(0);
}
});

ImageIcon i4 = new ImageIcon(ClassLoader.getResource("2.png"));
Image i5 = i4.getImage().getScaledInstance(50, 50, Image.SCALE_DEFAULT);
ImageIcon i6 = new ImageIcon(i5);
JLabel profile = new JLabel(i6);
profile.setBounds(40, 10, 50, 50);
p1.add(profile);

ImageIcon i7 = new ImageIcon(ClassLoader.getResource("video.png"));
Image i8 = i7.getImage().getScaledInstance(30, 30, Image.SCALE_DEFAULT);
ImageIcon i9 = new ImageIcon(i8);
JLabel video = new JLabel(i9);
video.setBounds(300, 20, 30, 30);
p1.add(video);

ImageIcon i10 = new ImageIcon(ClassLoader.getResource("phone.png"));
Image i11 = i10.getImage().getScaledInstance(35, 30, Image.SCALE_DEFAULT);
ImageIcon i12 = new ImageIcon(i11);
JLabel phone = new JLabel(i12);
phone.setBounds(360, 20, 35, 30);
p1.add(phone);

ImageIcon i13 = new ImageIcon(ClassLoader.getResource("3icon.png"));
Image i14 = i13.getImage().getScaledInstance(10, 25, Image.SCALE_DEFAULT);
ImageIcon i15 = new ImageIcon(i14);
JLabel morevert = new JLabel(i15);
morevert.setBounds(420, 20, 10, 25);
p1.add(morevert);

JLabel name = new JLabel("Bunty");

```

```

name.setBounds(110, 15, 100, 18);
name.setForeground(Color.WHITE);
name.setFont(new Font("SAN_SERIF", Font.BOLD, 18));
p1.add(name);

JLabel status = new JLabel("Active Now");
status.setBounds(110, 35, 100, 18);
status.setForeground(Color.WHITE);
status.setFont(new Font("SAN_SERIF", Font.BOLD, 14));
p1.add(status);

a1 = new JPanel();
a1.setBounds(5, 75, 440, 570);
f.add(a1);

text = new JTextField();
text.setBounds(5, 655, 310, 40);
text.setFont(new Font("SAN_SERIF", Font.PLAIN, 16));
f.add(text);

JButton send = new JButton("Send");
send.setBounds(320, 655, 123, 40);
send.setBackground(new Color(7, 94, 84));
send.setForeground(Color.WHITE);
send.addActionListener(this);
send.setFont(new Font("SAN_SERIF", Font.PLAIN, 16));
f.add(send);

f.setSize(450, 700);
f.setLocation(800, 50);
f.setUndecorated(true);
f.getContentPane().setBackground(Color.WHITE);

f.setVisible(true);
}

public void actionPerformed(ActionEvent ae) {

    try {

        String out = text.getText();
        JPanel p2 = formatLabel(out);
        a1.setLayout(new BorderLayout());
        JPanel right = new JPanel(new BorderLayout());
        right.add(p2, BorderLayout.LINE_END);
        vertical.add(right);
    }
}

```



```

vertical.add(Box.createVerticalStrut(15));
a1.add(vertical, BorderLayout.PAGE_START);
dout.writeUTF(out);
text.setText("");
f.repaint();
f.invalidate();
f.validate();
}

catch (Exception e)
{
    System.out.println(e);
    e.printStackTrace();
}
}

public static JPanel formatLabel(String out) {
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel, BoxLayout.Y_AXIS));

    JLabel output = new JLabel("<html><p style=\"width: 150px\">" + out + "</p></html>");
    output.setFont(new Font("Tahoma", Font.PLAIN, 16));
    output.setBackground(new Color(37, 211, 102));
    output.setOpaque(true);
    output.setBorder(new EmptyBorder(15, 15, 15, 50));

    panel.add(output);

    Calendar cal = Calendar.getInstance();
    SimpleDateFormat sdf = new SimpleDateFormat("HH:mm");

    JLabel time = new JLabel();
    time.setText(sdf.format(cal.getTime()));

    panel.add(time);
    return panel;
}

public static void main(String[] args) {
    new Client();

    try {

        Socket s = new Socket("127.0.0.1", 6001);
        DataInputStream din = new DataInputStream(s.getInputStream());
        dout = new DataOutputStream(s.getOutputStream());
    }
}

```

```

while(true) {
a1.setLayout(new BorderLayout());
String msg = din.readUTF();
JPanel panel = formatLabel(msg);
JPanel left = new JPanel(new BorderLayout());
left.add(panel, BorderLayout.LINE_START);
vertical.add(left);
vertical.add(Box.createVerticalStrut(15));
a1.add(vertical, BorderLayout.PAGE_START);
f.validate();
}
}
catch (Exception e)
{
System.out.println(e);
e.printStackTrace();
}
}
}

```

### **Server:-**

```

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import java.text.*;
import java.net.*;
import java.io.*;
public class Server implements ActionListener {

JTextField text;
JPanel a1;
static Box vertical = Box.createVerticalBox();
static JFrame f = new JFrame();
static DataOutputStream dout;

Server() {

f.setLayout(null);

JPanel p1 = new JPanel();
p1.setBackground(new Color(7, 94, 84));
p1.setBounds(0, 0, 450, 70);
p1.setLayout(null);

```

```

f.add(p1);

ImageIcon i1 = new ImageIcon(ClassLoader.getResource("3.png"));
Image i2 = i1.getImage().getScaledInstance(25, 25, Image.SCALE_DEFAULT);
ImageIcon i3 = new ImageIcon(i2);
JLabel back = new JLabel(i3);
back.setBounds(5, 20, 25, 25);
p1.add(back);

back.addMouseListener(new MouseAdapter() {
public void mouseClicked(MouseEvent ae) {
System.exit(0);
}
});

ImageIcon i4 = new ImageIcon(ClassLoader.getResource("1.png"));
Image i5 = i4.getImage().getScaledInstance(50, 50, Image.SCALE_DEFAULT);
ImageIcon i6 = new ImageIcon(i5);
JLabel profile = new JLabel(i6);
profile.setBounds(40, 10, 50, 50);
p1.add(profile);

ImageIcon i7 = new ImageIcon(ClassLoader.getResource("video.png"));
Image i8 = i7.getImage().getScaledInstance(30, 30, Image.SCALE_DEFAULT);
ImageIcon i9 = new ImageIcon(i8);
JLabel video = new JLabel(i9);
video.setBounds(300, 20, 30, 30);
p1.add(video);

ImageIcon i10 = new ImageIcon(ClassLoader.getResource("phone.png"));
Image i11 = i10.getImage().getScaledInstance(35, 30, Image.SCALE_DEFAULT);
ImageIcon i12 = new ImageIcon(i11);
JLabel phone = new JLabel(i12);
phone.setBounds(360, 20, 35, 30);
p1.add(phone);

ImageIcon i13 = new ImageIcon(ClassLoader.getResource("3icon.png"));
Image i14 = i13.getImage().getScaledInstance(10, 25, Image.SCALE_DEFAULT);
ImageIcon i15 = new ImageIcon(i14);
JLabel morevert = new JLabel(i15);
morevert.setBounds(420, 20, 10, 25);
p1.add(morevert);

JLabel name = new JLabel("Gaitonde");
name.setBounds(110, 15, 100, 18);
name.setForeground(Color.WHITE);

```

```

name.setFont(new Font("SAN_SERIF", Font.BOLD, 18));
p1.add(name);

JLabel status = new JLabel("Active Now");
status.setBounds(110, 35, 100, 18);
status.setForeground(Color.WHITE);
status.setFont(new Font("SAN_SERIF", Font.BOLD, 14));
p1.add(status);

a1 = new JPanel();
a1.setBounds(5, 75, 440, 570);
f.add(a1);

text = new JTextField();
text.setBounds(5, 655, 310, 40);
text.setFont(new Font("SAN_SERIF", Font.PLAIN, 16));
f.add(text);

JButton send = new JButton("Send");
send.setBounds(320, 655, 123, 40);
send.setBackground(new Color(7, 94, 84));
send.setForeground(Color.WHITE);
send.addActionListener(this);
send.setFont(new Font("SAN_SERIF", Font.PLAIN, 16));
f.add(send);

f.setSize(450, 700);
f.setLocation(200, 50);
f.setUndecorated(true);
f.getContentPane().setBackground(Color.WHITE);

f.setVisible(true);
}

public void actionPerformed(ActionEvent ae) {
    try {
        String out = text.getText();
        JPanel p2 = formatLabel(out);
        a1.setLayout(new BorderLayout());
        JPanel right = new JPanel(new BorderLayout());
        right.add(p2, BorderLayout.LINE_END);
        vertical.add(right);
        vertical.add(Box.createVerticalStrut(15));
        a1.add(vertical, BorderLayout.PAGE_START);
        dout.writeUTF(out);
        text.setText(" ");
    }
}

```

```

f.repaint();

} catch (Exception e) {
e.printStackTrace();
}
}

public static JPanel formatLabel(String out) {
JPanel panel = new JPanel();
panel.setLayout(new BoxLayout(panel, BoxLayout.Y_AXIS));

JLabel output = new JLabel("<html><p style=\"width: 150px\">" + out + "</p></html>");
output.setFont(new Font("Tahoma", Font.PLAIN, 16));
output.setBackground(new Color(37, 211, 102));
output.setOpaque(true);
output.setBorder(new EmptyBorder(15, 15, 15, 50));

panel.add(output);

Calendar cal = Calendar.getInstance();
SimpleDateFormat sdf = new SimpleDateFormat("HH:mm");

JLabel time = new JLabel();
time.setText(sdf.format(cal.getTime()));

panel.add(time);

return panel;
}

public static void main(String[] args) {
new Server();

try {
ServerSocket skt = new ServerSocket(6001);
while(true) {
Socket s = skt.accept();
DataInputStream din = new DataInputStream(s.getInputStream());
dout = new DataOutputStream(s.getOutputStream());

while(true) {
String msg = din.readUTF();
JPanel panel = formatLabel(msg);

JPanel left = new JPanel(new BorderLayout());
left.add(panel, BorderLayout.LINE_START);

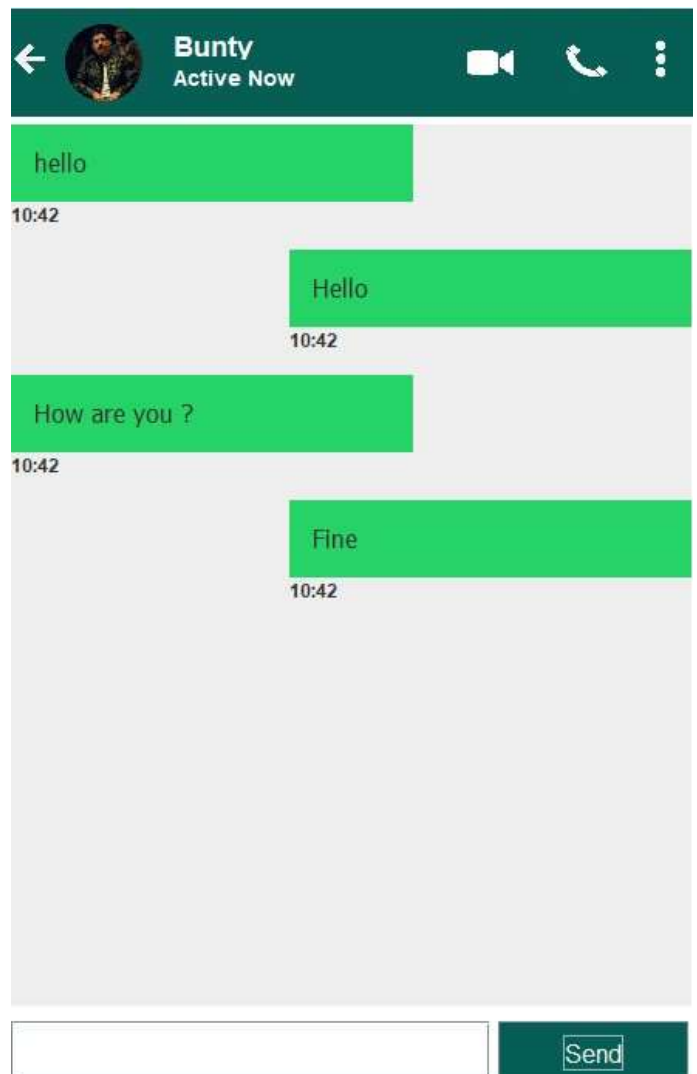
```

```

vertical.add(left);
f.validate();
}
}
}
catch (Exception e) {
System.out.println(e);
e.printStackTrace();
}
}

```

## **OUTPUT: -**



## 6. Actual Resources Used:

Sr. No.	Name of resource / material	Specification	Quantity	Remarks
1	Computer	WINDOWS 11,8GB RAM, 1 TB HDD	1	
2	Operating System	WINDOWS 11	1	
3	Compiler/ Idea	CMD	1	
4	Browser	Chrome	1	

## 7. Skill developed / Learning out of this Micro-Project:

There are so many thing that we learn from this project of

- We learn from this project is, how to program in Java.
- We learnt to develop the Application using frames in java.
- We learn the use Swing and awt.
- We discovered new things while researching the project.

## 8. Applications of this Micro-Project:

1. User Interface and App Functionality. Your Java Messenger user interface (UI) and overall functionality determine how successful your app will be.
2. Graphics Performance.
3. User Features.
4. Open-Source Security Liabilities.

\*\*\*\*\*