



“Proyecto”

MONTSERRAT OLAN LOPEZ

MODELO ENTIDAD-RELACIÓN, INSTITUTO UNIVERSITARIO DE YUCATÁN

2303040768: REDES II

MAESTRA ING. PERLA ALEJANDRA LANDERO HEREDIA

19 DE OCTUBRE DEL 2025

UDP/TCP Cliente

Configuración

TCP

Host: localhost

Puerto: 3306

Desconectar

Conectado

Total: 0 mensajes

Chat UDP/TCP

? UDP/TCP
? Conéctate al servidor para comenzar a chatear
? Conectado al servidor
? Servidor: SERVIDOR: Bienvenido! Tu ID es #5

Limpiar historial

Enviar

UDP/TCP Cliente

Configuración

TCP

Host: localhost

Puerto: 3306

Desconectar

Chat UDP/TCP

? UDP/TCP

? Conéctate al servidor para comenzar a chatear

? Conectado al servidor

▣ Servidor: SERVIDOR: Bienvenido! Tu ID es #5

Limpiar historial

Enviar

UDP/TCP Cliente

Configuración

TCP

Host: localhost

Puerto: 3306

Desconectar

Chat UDP/TCP

? UDP/TCP

? Conéctate al servidor para comenzar a chatear

? Conectado al servidor

▣ Servidor: SERVIDOR: Bienvenido! Tu ID es #6

Limpiar historial

Enviar

UDP/TCP Cliente

Configuración

TCP

Host: localhost

Puerto: 3306

Desconectar

Chat UDP/TCP

? UDP/TCP
? Conéctate al servidor para comenzar a chatear
? Conectado al servidor
[?] Servidor: SERVIDOR: Bienvenido! Tu ID es #5
[?] Tú: MONTSERRAT
[?] Servidor: SERVIDOR: Mensaje difundido a 1 cliente(s)

Limpiar historial

Enviar

UDP/TCP Cliente

Configuración

TCP

Host: localhost

Puerto: 3306

Desconectar

Chat UDP/TCP

? UDP/TCP
? Conéctate al servidor para comenzar a chatear
? Conectado al servidor
[?] Servidor: SERVIDOR: Bienvenido! Tu ID es #6
[?] Servidor: Cliente #5: MONTSERRAT

Limpiar historial

Enviar

```

1  package cliente;
2
3  import java.io.*;
4  import java.net.*;
5  import javax.swing.*;
6  import java.awt.*;
7  import java.awt.event.*;
8
9  public class Cliente extends JFrame {
10
11     private JTextArea areaChat;
12     private JTextField campoMensaje, campoHost, campoPuerto;
13     private JButton btnEnviar, btnLimpiar, btnConectar;
14     private JComboBox<String> comboProtocolo;
15     private JLabel lblEstado, lblMensajesEnviados;
16     private JPanel panelEstadoConexion;
17
18     private DatagramSocket socketUDP;
19     private Socket socketTCP;
20     private boolean conectado = false;
21     private boolean escuchando = false;
22     private String protocoloActual = "TCP";
23     private String hostActual = "localhost";
24     private int puertoActual = 3306;
25     private int mensajesEnviados = 0;
26
27     public Cliente() {
28         configurarVentana();
29         crearComponentes();
30         configurarEventos();
31     }
32
33     private void configurarVentana() {
34         setTitle("UDP/TCP Cliente");
35         setSize(900, 650);
36         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
37         setLocationRelativeTo(null);
    
```

```

1  package cliente;
2  import java.io.*;
3  import java.net.*;
4  import java.util.*;
5  import java.util.concurrent.*;
6
7  /**
8   *
9   * @author monts
10  */
11  public class Servidor {
12
13
14      private static List<ManejadorClienteTCP> clientesTCP = new CopyOnWriteArrayList<>();
15      private static Map<String, ClienteUDP> clientesUDP = new ConcurrentHashMap<>();
16      private static int contadorClientes = 0;
17
18      public static void main(String[] args) {
19
20
21          servidorTCP();
22      }
23
24
25      public static void servidorTCP() {
26          final int PUERTO = 3306;
27
28          try {
29              ServerSocket servidor = new ServerSocket(PUERTO);
30              System.out.println(" ");
31              System.out.println(" Servidor TCP Multi-Cliente Iniciado ");
32              System.out.println(" Puerto: " + PUERTO + " ");
33              System.out.println(" \n");
34
35              while (true) {
36                  Socket socket = servidor.accept();
37                  contadorClientes++;

```

