

Soluzione Esercizio 2

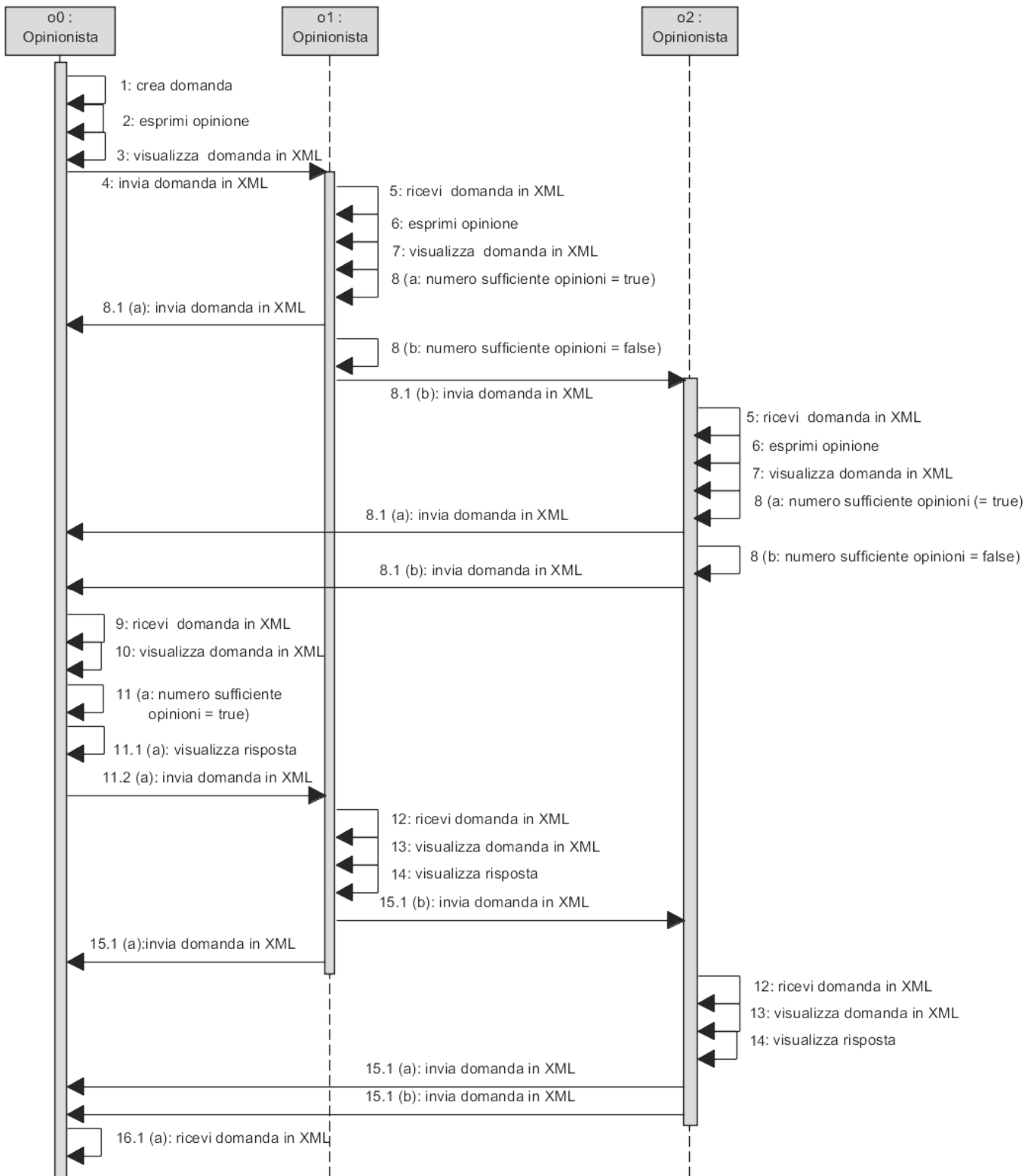


Fig.5 - Sequenza di passi che le istanze di *Opinionista* devono svolgere in alcuni dei casi di test. Per brevità non viene rappresentata la classe *Domanda*.

```

1 // Domanda.java
2 import com.thoughtworks.xstream.*;
3
4 public class Domanda {
5     private String testo;
6     private double sommaOpinioni;
7     private int numeroOpinioni;
8     private int numeroSufficienteOpinioni;
9     private int portaOpinionistaInterrogante;
10
11     public Domanda(String t, int suffOp, int portaOpInterr) {
12         testo = t;
13         sommaOpinioni = 0.0;
14         numeroOpinioni = 0;
15         numeroSufficienteOpinioni = suffOp;
16         portaOpinionistaInterrogante = portaOpInterr;
17     }
18
19     public Domanda(String xml) {
20         Domanda d = (Domanda)(new XStream()).fromXML(xml);
21         testo = d.testo;
22         sommaOpinioni = d.sommaOpinioni;
23         numeroOpinioni = d.numeroOpinioni;
24         numeroSufficienteOpinioni = d.numeroSufficienteOpinioni;
25         portaOpinionistaInterrogante = d.portaOpinionistaInterrogante;
26     }
27
28     public boolean sufficientiOpinioni() {
29         return numeroOpinioni >= numeroSufficienteOpinioni;
30     }
31
32     public String toString() {
33         return (new XStream()).toXML(this);
34     }
35
36     public double elaboraRisposta () {
37         return sommaOpinioni/numeroOpinioni;
38     }
39
40     public void esprimiOpinione (double op) {
41         sommaOpinioni += op;
42         numeroOpinioni++;
43     }
44
45     public int portaOpinionistaInterrogante () {
46         return portaOpinionistaInterrogante;
47     }
48 }
49

```

```

1 // Opinionista.java
2 import java.net.*;
3 import java.io.*;
4 import java.sql.*;
5
6 public class Opinionista {
7     private static int portaRicezione, portaInvio;
8
9     public static void invia(int porta, Domanda d) {
10         try (Socket sock = new Socket("localhost", porta);

```

```

11     DataOutputStream dos =
12         new DataOutputStream(sock.getOutputStream());
13     } { dos.writeUTF(d.toString());
14     } catch (IOException e) { e.printStackTrace();}
15     visualizzaSuLog("invia", d.toString());
16 }
17
18 public static Domanda ricevi() {
19     Domanda d = null;
20     try {
21         ServerSocket servsock = new ServerSocket(portaRicezione);
22         Socket sock = servsock.accept();
23         DataInputStream dis =
24             new DataInputStream(sock.getInputStream());
25     } {
26         d = new Domanda(dis.readUTF());
27         visualizzaSuLog("ricevi", d.toString());
28     } catch (IOException e) { e.printStackTrace();}
29     return d;
30 }
31
32 public static void visualizzaSuLog(String operazione, String risultato) {
33     try {
34         Connection co = DriverManager.getConnection("jdbc:mysql://localhost:3306/opinionista",
35 "root", "");
36         PreparedStatement ps = co.prepareStatement("INSERT INTO log VALUES (?, ?, ?)");
37     } {
38         ps.setInt(1, portaRicezione); ps.setString(2, operazione); ps.setString(3, risultato);
39         ps.executeUpdate();
40     } catch (SQLException e) { System.err.println(e.getMessage());}
41 }
42
43 public static void main(String[] args) {
44     portaRicezione = Integer.parseInt(args[0]);
45     portaInvio = Integer.parseInt(args[1]);
46     Domanda d;
47     if (args.length > 3) { // e' interrogante
48         visualizzaSuLog("inizio", "*****");
49         d = new Domanda(args[3], Integer.parseInt(args[4]), portaRicezione); //1
50         d.esprimiOpinione(Double.parseDouble(args[2])); //2
51         visualizzaSuLog("esprimiOpinione", d.toString()); //3
52         if (d.sufficientiOpinioni()) //11(a)
53             visualizzaSuLog("elaboraRisposta", Double.toString(d.elaboraRisposta())); //11.1(a)
54         else {
55             invia(portaInvio, d); //4
56             d = ricevi(); //9,10
57             if (d.sufficientiOpinioni()) { //11(a)
58                 visualizzaSuLog("elaboraRisposta", Double.toString(d.elaboraRisposta()));
59                 //11.1(a)
60                 invia(portaInvio, d); //11.2(a)
61                 ricevi(); //16.1(a)
62             }
63         }
64         visualizzaSuLog("fine", "*****");
65     }
66     else { //non è interrogante
67         d = ricevi(); //5
68         d.esprimiOpinione(Double.parseDouble(args[2])); //6
69         visualizzaSuLog("esprimiOpinione", d.toString()); //7

```

```

69     if (d.sufficientiOpinioni()) { //8(a)
70         invia(d.portaOpinionistaInterrogante(), d); //8.1(a)
71         d = ricevi(); //12,13
72         visualizzaSuLog("elaboraRisposta", Double.toString(d.elaboraRisposta())); //14
73         invia(d.portaOpinionistaInterrogante(), d); //15.1(a)
74     } else { //8(b)
75         invia(portaInvio, d); //8.1(b)
76         d = ricevi(); //12, 13
77         visualizzaSuLog("elaboraRisposta", Double.toString(d.elaboraRisposta())); //14
78         invia(portaInvio, d); //15.1(b)
79     }
80 }
81 }
82 }
83

```

Log (formato contratto)

idprocesso operazione risultato (somOp, numOp, numSuf,porInt)

8080	inizio.....	*****	8082	ricevi.....	115.0, 2, 4, 8080
8080	esprimiOpinione	55.0, 1, 2, 8080	8082	esprimiOpinione	165.0, 3, 4, 8080
8080	invia.....	55.0, 1, 2, 8080	8082	invia.....	165.0, 3, 4, 8080
8081	ricevi.....	55.0, 1, 2, 8080	8080	ricevi.....	165.0, 3, 4, 8080
8081	esprimiOpinione	115.0, 2, 2, 8080	8080	fine.....	*****
8081	invia.....	115.0, 2, 2, 8080	8080	inizio.....	*****
8080	ricevi.....	115.0, 2, 2, 8080	8080	esprimiOpinione	55.0, 1, 1, 8080
8080	elaboraRisposta	57.5	8080	elaboraRisposta	55.0
8080	invia.....	115.0, 2, 2, 8080	8080	fine.....	*****
8081	ricevi.....	115.0, 2, 2, 8080			
8081	elaboraRisposta	57.5			
8081	invia.....	115.0, 2, 2, 8080			
8080	ricevi.....	115.0, 2, 2, 8080			
8080	fine.....	*****			
8080	inizio.....	*****			
8080	esprimiOpinione	55.0, 1, 3, 8080			
8080	invia.....	55.0, 1, 3, 8080			
8081	ricevi.....	55.0, 1, 3, 8080			
8081	esprimiOpinione	115.0, 2, 3, 8080			
8081	invia.....	115.0, 2, 3, 8080			
8082	ricevi.....	115.0, 2, 3, 8080			
8082	esprimiOpinione	165.0, 3, 3, 8080			
8082	invia.....	165.0, 3, 3, 8080			
8080	ricevi.....	165.0, 3, 3, 8080			
8080	elaboraRisposta	55.0			
8080	invia.....	165.0, 3, 3, 8080			
8081	ricevi.....	165.0, 3, 3, 8080			
8081	elaboraRisposta	55.0			
8081	invia.....	165.0, 3, 3, 8080			
8082	ricevi.....	165.0, 3, 3, 8080			
8082	elaboraRisposta	55.0			
8082	invia.....	165.0, 3, 3, 8080			
8080	ricevi.....	165.0, 3, 3, 8080			
8080	fine.....	*****			
8080	inizio.....	*****			
8080	esprimiOpinione	55.0, 1, 4, 8080			
8080	invia.....	55.0, 1, 4, 8080			
8081	ricevi.....	55.0, 1, 4, 8080			
8081	esprimiOpinione	115.0, 2, 4, 8080			
8081	invia.....	115.0, 2, 4, 8080			