



## Pràctica 2

XARXES TRONCALS

MÀSTER EN ENGINYERIA INFORMÀTICA

*Meriem Abjil Bajja*

27 de gener de 2019

# Índex

<b>1</b>	<b>Introducció</b>	<b>3</b>
<b>2</b>	<b>Entorn de treball</b>	<b>4</b>
<b>3</b>	<b>Creació del Chat amb el Socket IO</b>	<b>5</b>
<b>4</b>	<b>Resultats</b>	<b>17</b>
<b>5</b>	<b>Lliurament</b>	<b>19</b>
<b>6</b>	<b>Referències bibliogràfiques</b>	<b>20</b>

# Índex de figures

1	Inici de l'aplicació . . . . .	17
2	El <i>nickname</i> ja existeix . . . . .	17
3	El <i>nickname</i> no existeix . . . . .	18
4	La comunicació entre els diferents usuaris . . . . .	18

# 1 Introducció

S'ha de realitzar una aplicació de xat, Client-Servidor (CS), basada en el navegador, amb **Node.JS** i *socket.io*

- el client de xat s'ha de fer amb *HTML*, *JS* i *socket.io*
- el servidor web (on s'allotja la pàgina web amb el client de xat) s'ha de fer amb **Node.JS** (i *Express*)
- el servidor de xat s'ha de fer amb **Node.JS** i *socket.io*
- s'ha de fer un sol script, que fa alhora de servidor web i de servidor de xat
- cal que s'estudii aquest codi i afegiu-ri 2 o més millores d'entre aquestes:
  1. que el servidor no reenvii el missatge al client que l'envia sinó que això ho faci el mateix client
  2. **que hi hagi el nickname de l'usuari al davant de cada línia**
  3. **que quan un usuari entri/surti del xat es difongui a tots els usuaris un missatge informant-ho**
  4. que hi hagi "sales" de xat

Les millores aplicades són la 2 i la 3.

Realitzar un estudi dels paquets intercanviats entre clients i el servidor en una sessió concreta:

- amb *Wireshark* es captura els paquets de tota la sessió
- s'analitza la 1a connexió TCP entre un client web i el servidor web per obtenir la pàgina *HTML+JS* amb el client de xat, el diàleg *HTTP* transportat per la connexió, etc.
- s'analitza la 2a connexió TCP entre un client de xat i el servidor de xat, el diàleg "*socket.io*" transportat per la connexió, etc. \*

## 2 Entorn de treball

Cal el següent:

- una 1a màquina amb el servidor web i el servidor de xat
- una 2a i 3a (i potser més) màquines amb un navegador \*
- les tres (o més) màquines unides per una xarxa (si hi ha un *router NAT* entre les màquines, poseu a la xarxa interna la 2a i la 3a, o si al revés, “obriu-hi” el #port del servidor)

Es proposa el següent entorn:

- la 1a màquina que sigui una màquina virtual (VM) de **VirtualBox** amb el S.O. **Ubuntu Server**
- la 2a i 3a màquines que siguin la màquina “real” on corre la virtual i un mòbil, o bé la màquina “real” i un altre ordinador, o bé dos ordinadors, o bé dos mòbils, etc.

Segon l’entorn, es configurarà diferent:

- p.e., a la xarxa domèstica de casa vostra (*Ethernet* i *Wifi*): escolliu el mode de xarxa *bridged* de la VM (així la VM i la màquina real es troben a la mateixa xarxa IP; heu de fer la configuració de xarxa IP de l’Ubuntu d’acord amb això); la 2a i 3a màquines (p.e., la real i un mòbil) a la xarxa domèstica amb *Ethernet* o Wi-Fi
- p.e., a la xarxa de l’aula de pràctiques/Internet (*Ethernet*): escolliu el mode de xarxa *bridged* de la VM, com abans; la 2a i 3a màquines (p.e., la real i un altre PC, o dos PCs) a la mateixa xarxa amb *Ethernet*

### 3 Creació del Chat amb el Socket IO

Escriure una aplicació de xat ha estat tradicionalment molt difícil. Sockets ha estat tradicionalment la solució al voltant de la qual s'articulen els sistemes de xat en temps real, proporcionant un canal de comunicació bidireccional entre un client i un servidor.

Això significa que el servidor pot enviar missatges als clients. Cada vegada que escriu un missatge de xat, la idea és que el servidor l'obtingui i l'empeny a tots els altres clients connectats.

Primer anem a crear un fitxer `package.json` que descriu el nostre projecte.

```
1 {  
2   "name": "nodechat",  
3   "version": "0.1.0",  
4   "description": "Chat Socket IO",  
5   "main": "index.js",  
6   "repository": "",  
7   "author": "Meriem Abjil Bajja",  
8   "dependencies": {  
9     "express": "~3.2.4",  
10    "socket.io": "~0.9.14"  
11  }  
12 }
```

Ara que s'ha descrit el projecte, podem crear un arxiu `index.js` que configurarà la nostra aplicació.

```
1 /var express = require('express'),  
2   app = express(),  
3   server = require('http').createServer(app),  
4   io = require("socket.io").listen(server),  
5   nicknames = {};  
6  
7 //S'escoltarà pel port 8000  
8 server.listen(8000);  
9
```

```

10 //Al carregar l'arrel ('/'), redirecciona l'enllaç del directori + l'index.html
11 app.get('/', function(req, res) {
12     res.sendFile(__dirname + '/index.html');
13 });
14
15 //Sempre que s'utilitzi l'arrel ('/'), carrega el contingut del directori actual
16 app.use('/', express.static(__dirname));
17
18 //S'activa la funció cada vegada que un usuari s'ha connectat.
19 //La funció rep per parametre el socket de la connexió
20 io.sockets.on('connection', function(socket) {
21     //Quan rep un missatge "send message", envia a tots els sockets el missatge
        i el nickname del socket
22     socket.on('send message', function(data) {
23         io.sockets.emit('new message', {msg: data, nick: socket.nickname});
24     });
25
26     //Quan hi ha un nou client, li assigna un nickname al socket i incrementa el
        nombre de nicknames que hi ha.
27     socket.on('new user', function(data, callback) {
28         if (data in nicknames) {
29             callback(false);
30         } else {
31             callback(true);
32             socket.nickname = data;
33             nicknames[socket.nickname] = 1;
34             updateNickNames();
35         }
36     });
37
38     //Quan un usuari es desconnecta, s'elimina el nickname dels usuaris
39     socket.on('disconnect', function(data) {
40         if(!socket.nickname) return;
41         delete nicknames[socket.nickname];
42         nicknameDisconnect(socket.nickname);
43         updateNickNames();

```

```

44     });
45
46     //Envia a tots els usuaris que el nickname "nicknameDisconnect" s'ha
        disconnectat
47     function nicknameDisconnect(nicknameDisconnect) {
48         io.sockets.emit('usernameDisconnect', {nick: nicknameDisconnect});
49     }
50
51     //Actualitza els nicknames dels usuaris
52     function updateNickNames() {
53         io.sockets.emit('usernames', nicknames);
54     }
55 });

```

Això es tradueix en el següent:

- Express inicialitza l'aplicació per ser un controlador de funció que podeu subministrar a un servidor HTTP (tal com es veu a la línia 2).
- Definim un controlador de ruta / que es crida quan arribem a la nostra pàgina web.
- Definim el missatges de client / servidor.
- Realitzem l'escolta del servidor http al port 8000.

Crearem un fitxer **index.html** i el servirem. El fitxer **index.html** és el següent:

### Codi HTML de la pàgina web

```

1 <html>
2   <head>
3     <title>Chat Socket IO</title>
4
5     <script src="https://code.jquery.com/jquery.js"></script>
6
7     <!-- Estil general -->
8     <link rel="stylesheet" href="css/estil_general.css">

```



```

9      <!-- Estil form missatges -->
10     <link rel="stylesheet" href="css/estil_form_messages.css">
11     <!-- Estil form nickname -->
12     <link rel="stylesheet" href="css/estil_form_nickname.css">
13     <!-- Latest compiled and minified CSS -->
14     <link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.0.3/
        css/bootstrap.min.css">
15     <!-- Optional theme -->
16     <link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.0.3/
        css/bootstrap-theme.min.css">
17     <!-- Latest compiled and minified JavaScript -->
18     <script src="//netdna.bootstrapcdn.com/bootstrap/3.0.3/js/bootstrap.min.
        js"></script>
19
20 </head>
21 <body>
22
23     <div class="container">
24         <hr id="linia_top_nick" />
25         <!-- Primera pantalla: Escriure el nickname -->
26         <div class="input-group" id="nickWrap">
27             <input id="nickname" class="form-control input-lg" type="text"
                placeholder="Escriu el teu nickname">
28             <span class="input-group-btn">
29                 <button class="btn btn-lg btn-primary" type="submit" id="
                    setNick">OK</button>
30             </span>
31         </div>
32         <!-- Si el nickname ja està utilitzat, t'apareix una alerta diguent
                que el nickname ja està utilitzat -->
33         <div class="alert fade in alert-danger alert-dismissible" data-
            dismiss="alert" id="login-error" style="display:none;">
34             <button type="button" class="close" id="closeAlert">x</button>
35             Aquest nickname ja està utilitzat. Prova-ho amb un altre.
36         </div>
37         <!-- Fi de l'error -->

```

```

38     <!-- Fi primera pantalla -->
39     <hr id="linia_bottom_nick"/>
40
41     <!-- Sala de chat -->
42     <div id="contentWrap" class="row" style="height: 70%;">
43         <div class="col-md-2"></div>
44         <div id="chatWrap" class="col-md-6">
45             <!-- Els missatges del chat -->
46             <div id="panel_messages" class="panel panel-success">
47                 <div class="panel-heading">Chat</div>
48                 <div id="chat" class="panel-body">
49                     <ul id="messages"></ul>
50                 </div>
51             </div>
52
53             <!-- Formulari per escriure un missatge -->
54             <div>
55                 <form id="send-message" class="input-group" style="text-align: center; margin: 5px auto;">
56                     <input id="message" class="form-control input-lg" type="text">
57                     <span class="input-group-btn">
58                         <button id="send_button" class="btn btn-lg btn-primary" type="submit">Envia</button>
59                     </span>
60                 </form>
61             </div>
62         </div>
63
64
65     <!-- Els usuaris actius -->
66     <div id="usersWrap" class="col-md-2">
67         <div class="panel panel-info" style="margin-top: 10px;">
68             <div class="panel-heading" style="text-align: center; -webkit-box-shadow: 0 1px 2px rgba(0,0,0,0.05); box-shadow: 0 1px 2px rgba(0,0,0,0.05);">Usuaris</div>

```

```

69         <div id="users" class="panel-body"></div>
70     </div>
71 </div>
72 </div>
73
74 </div>
75 <script src="/socket.io/socket.io.js"></script>
76 <script src="./js/enviament_missatges.js"></script>
77 </body>
78 </html>

```

## Codi Javascript de la pàgina web

```

1  /jQuery(function($) {
2      //Creem un socket
3      var socket = io.connect();
4      //Formulari per enviar un missatge
5      var $messageForm = $('#send-message');
6      //El missatge a enviar
7      var $messageBox = $('#message');
8      //El panel de chat
9      var $chat = $('#chat');
10     //Els missatges del chat
11     var $messages_chat = $('#messages');
12     //El botó per el qual s'envia el missatge
13     var $buttonSend = $('#send');
14
15     //Formulari del nickname
16     var $nickForm = $('#setNick');
17     //El nickname a enviar
18     var $nickBox = $('#nickname');
19     //La llista d'usuaris que hi ha
20     var $users = $('#users');
21     //Creu de l'alerta de l'error
22     var $closeAlert = $('#closeAlert');
23

```

```

24
25
26 $nickForm.click(function(e) {
27     e.preventDefault();
28     console.log("nickform");
29     //El socket emet el missatge de "new user" on envia el nickname i espera
        un valor retornat
30     //Si el valor és true, llavors amaga el formulari del nickname i mostra
        la sala de chat
31     //Si el valor és fals, llavors mostra el missatge d'error: "Aquest
        nickname ja està utilitzat. Prova-ho amb un altre."
32     socket.emit('new user', $nickBox.val(), function(data) {
33         console.log("new user");
34         if(data) {
35             console.log("new nickname is OK");
36             $('#nickWrap').hide();
37             $('#linia_top_nick').hide();
38             $('#linia_bottom_nick').hide();
39             $('#contentWrap').show();
40         } else {
41             console.log("new nickname is WRONG");
42             $('#login-error').show();
43         }
44     });
45     //Eliminem el valor del nickname
46     $nickBox.val('');
47 });
48
49 //Quan cliquem la creu de l'alerta "Aquest nickname ja està utilitzat. Prova-
        ho amb un altre.", s'amagarà l'error
50 $closeAlert.click(function(e) {
51     $('#login-error').hide();
52 });
53
54 //Quan enviem un missatge, s'envia pel socket "send message" amb el valor del
        missatge.

```

```

55     $messageForm.submit(function(e) {
56         e.preventDefault();
57         console.log("messageForm submit");
58         if($messageBox.val() != '') {
59             console.log("send message" + $messageBox.val());
60             socket.emit('send message', $messageBox.val());
61         }
62         //Eliminem el valor del missatge
63         $messageBox.val('');
64     });
65
66     //Quan reps un missatge "new message", s'afageix en el chat, el nickname i el
        missatge
67     socket.on('new message', function(data) {
68         console.log('<b>' + data.nick + "</b> " + data.msg + "<br/>");
69         $messages_chat.append('<li><b>' + data.nick + "</b> " + data.msg + "</li>");
70     });
71
72     //Quan reps un missatge "new message", s'afageix en el chat, el nickname i el
        missatge
73     socket.on('usernameDisconnect', function(data) {
74         console.log('Username Disconnect ' + data.nick);
75         $messages_chat.append("<li>L'usuari <b>" + data.nick + "</b> s'ha desconnectat
            </li>");
76     });
77
78     //Quan reps el missatge "usernames", s'insereixen tots els nicknames dels
        usuaris en el div dels usuaris
79     socket.on('usernames', function(data) {
80         console.log("usernames");
81         var html = '';
82         for (var username in data) {
83             html += "<p>" + username + "</p>";
84         }
85         $users.html(html);
86     });

```

```
87
88 });
```

## Estil general de la pàgina web

```
1 #contentWrap{
2     display: none;
3 }
4 #chatWrap{
5     float: left;
6     background: rgba(19, 35, 47, 0.9);
7     max-width: 600px;
8     margin: 40px auto;
9     border-radius: 4px;
10    box-shadow: 0 4px 10px 4px rgba(19, 35, 47, 0.3);
11    position: relative;
12 }
13 #chat{
14     position: relative;
15     width: 100%;
16     float: left;
17     height: 100%;
18     overflow: auto;
19     margin: 10px 0px;
20     padding: 0px;
21 }
22
23 #panel_messages {
24     margin-top: 10px;
25 }
```

## Estil del formulari del nickname de la pàgina web

```
1 #linia_top_nick , #linia_bottom_nick {
2     margin-top: 100px;
3     border: 1px solid #d6e9c6;
4     -webkit-box-shadow: inset 0 1px 0 rgba(255,255,255,0.15), 0 1px 1px rgba
```

```

        (0,0,0,0.075);
5     box-shadow: inset 0 1px 0 rgba(255,255,255,0.15), 0 1px 1px rgba
        (0,0,0,0.075);
6 }
7
8 #nickname {
9     padding: 10px 17px;
10    background: white;
11    color: #3c763d;
12    font-size: 16px;
13    height: auto;
14    border: 1px solid #d6e9c6;
15    -webkit-box-shadow: inset 0 1px 0 rgba(255,255,255,0.15), 0 1px 1px rgba
        (0,0,0,0.075);
16    box-shadow: inset 0 1px 0 rgba(255,255,255,0.15), 0 1px 1px rgba
        (0,0,0,0.075);
17 }
18
19 #setNick {
20     padding: 10px 17px;
21     background: #dff0d8;
22     border: 1px solid #d6e9c6;
23     color: #3c763d;
24     font-size: 16px;
25 }
26
27 #nickWrap {
28     width: 330px;
29     text-align: center;
30     margin: 100px auto;
31 }
32 #usersWrap {
33     background: rgba(19, 35, 47, 0.9);
34     margin: 40px auto;
35     border-radius: 4px;
36     box-shadow: 0 4px 10px 4px rgba(19, 35, 47, 0.3);

```

```

37     position: relative;
38     margin-left: 15px;
39 }
40
41 #users {
42     position: relative;
43     width: 100%;
44     float: left;
45     overflow: auto;
46     margin: 10px 0px;
47     color: #31708f;
48     padding: 0px;
49 }
50
51 #users p {
52     margin: 0 10px 10px 10px;
53     background: aliceblue;
54     padding: 10px;
55     border-radius: 30px;
56     -webkit-box-shadow: 0 1px 2px rgba(0,0,0,0.05);
57     box-shadow: 0 1px 2px rgba(0,0,0,0.05);
58     text-align: center;
59 }

```

## Estil del formulari del chat de la pàgina web

```

1 #messages
2 {
3     list-style-type: none;
4     padding: 0;
5 }
6
7 #messages li
8 {
9     padding: 10px;
10    border-radius: 15px;

```



```
11     margin-bottom: 5px;
12     color: #3C761E;
13     margin-right: 4px;
14 }
15
16 #messages li:nth-child(odd)
17 {
18     background: #eee;
19 }
20
21 #messages li:nth-child(even)
22 {
23     background: #dce3e2;
24 }
25 }
26
27 #message {
28     padding: 10px 17px;
29     background: white;
30     color: #3c763d;
31     font-size: 15px;
32     height: auto;
33 }
34
35 #send_button {
36     padding: 10px 17px;
37     background: #d9f0d9;
38     border: 1px solid #d6e9c6;
39     color: #3c763d;
40     font-size: 15px;
41 }
```

## 4 Resultats

A l'inici de l'aplicació, et demana un **nickname**, veure *Figura 1*.

---

---

Figura 1: Inici de l'aplicació

Si no existeix el **nickname**, mostra un **error** i et torna a demanar un **nickname**, veure *Figura 2*.

---

Aquest nickname ja està utilitzat. Prova-ho amb un altre.

---

Figura 2: El *nickname* ja existeix

Si existeix el **nickname**, entraràs al **chat**, veure *Figura 3*.

---

---

Figura 3: El *nickname* no existeix

Una vegada ha iniciat el **chat**, hi ha la comunicació entre els diferents usuaris, veure *Figura 4*. A la dreta hi ha el **chat** i a l'esquerra els diferents **usuaris**.

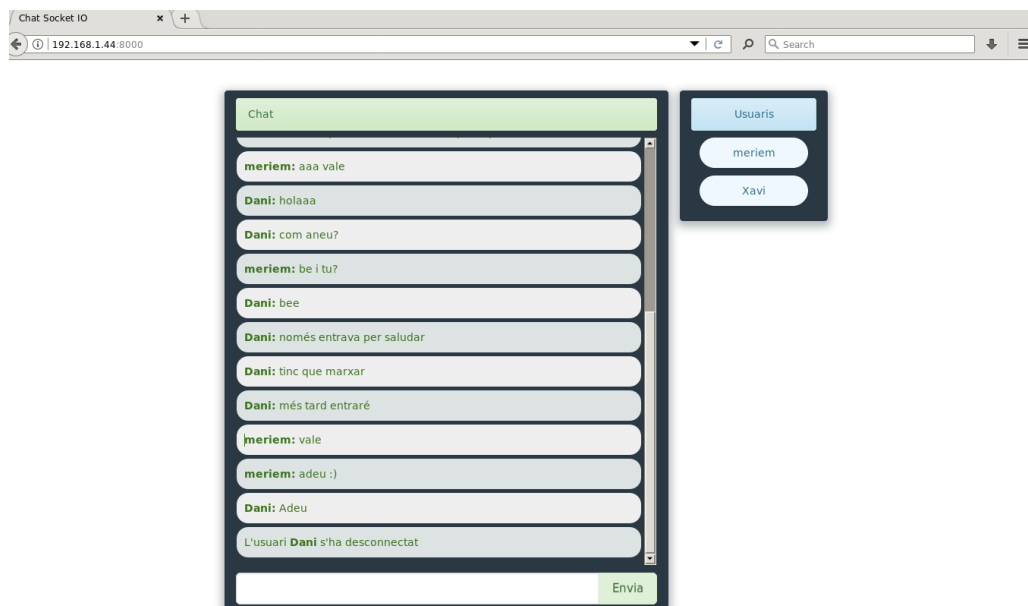


Figura 4: La comunicació entre els diferents usuaris

## 5 Lliurament

El video de la **demo**, el pots trobar en el següent enllaç: [Demo](#)

L'enllaç de la **presentació del treball**, el pots trobar en el següent enllaç: [Presentacio](#)

L'**estudi dels paquets** intercanviats entre el client i el servidor, el pots trobar en el següent enllaç: [El cas d'estudi](#)

## 6 Referències bibliogràfiques

### Sobre Sockets

- NPM. *socket.io*. [online]. Disponible a: <https://www.npmjs.com/package/socket.io> (27 de gener de 2019).
- SOCKET.IO. *Socket.IO*. [online]. Disponible a: <https://socket.io/> (27 de gener de 2019).
- SOCKET.IO. *Socket.IO - Chat*. [online]. Disponible a: <https://socket.io/get-started/chat/> (27 de gener de 2019).

### Sobre Apache

- *Welcome! - The Apache HTTP Server Project*. [online]. Disponible a: <http://httpd.apache.org/> (27 de gener de 2019).
- *HTTPD - Apache2 Web Server*. [online].  
Disponible a: <https://help.ubuntu.com/lts/serverguide/httpd.html> (27 de gener de 2019).

### Sobre Node.js

- *Node.JS*. [online]. Disponible a: <https://nodejs.org/es/> (27 de gener de 2019).
- *Create HTTP Web Server in Node.js*. [online]. Disponible a: <https://www.guru99.com/node-js-create-server-get-data.html> (27 de gener de 2019).
- NODEJS.ORG. *Node.js v11.8.0 Documentation*. [online].  
Disponible a: <https://nodejs.org/api/synopsis.html> (27 de gener de 2019).

### Sobre Express

- *Escritura de middleware para su uso en aplicaciones Express*. [online]. Disponible a: <https://expressjs.com/es/guide/writing-middleware.html> (27 de gener de 2019).

- W3SCHOOLS.COM. *Node.js. Introduction.* [online]. Disponible a: [https://www.w3schools.com/nodejs/nodejs\\_intro.asp](https://www.w3schools.com/nodejs/nodejs_intro.asp) (27 de gener de 2019).
- EXPRESSJS.COM. *Express - Node.js web application framework.* [online]. Disponible a: <http://expressjs.com/> (27 de gener de 2019).
- EXPRESSJS.COM. *Express "Hello World" example.* [online]. Disponible a: <http://expressjs.com/en/starter/hello-world.html> (27 de gener de 2019).
- MDN WEB DOCS. *Express Web Framework (Node.js/JavaScript).* [online]. Disponible a: [https://developer.mozilla.org/es/docs/Learn/Server-side/Express\\_Nodejs](https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs) (27 de gener de 2019).

## Sobre VirtualBox

- VIRTUALBOX.ORG. *Oracle VM VirtualBox.* [online]. Disponible a: <https://www.virtualbox.org/> (27 de gener de 2019).
- VIRTUALBOX.ORG. *Oracle VM VirtualBox.* [online].  
Disponible a: <https://www.virtualbox.org/manual/UserManual.html> (27 de gener de 2019).

## Sobre Configuració de xarxa IP a Ubuntu

- FÀBREGA, L. *Configuració de xarxa IP d'una estació (manualment i automàticament via DHCP) i d'un router via encaminament estàtic a GNU/Linux Ubuntu*, curs 2017/18, UdG, 2018.