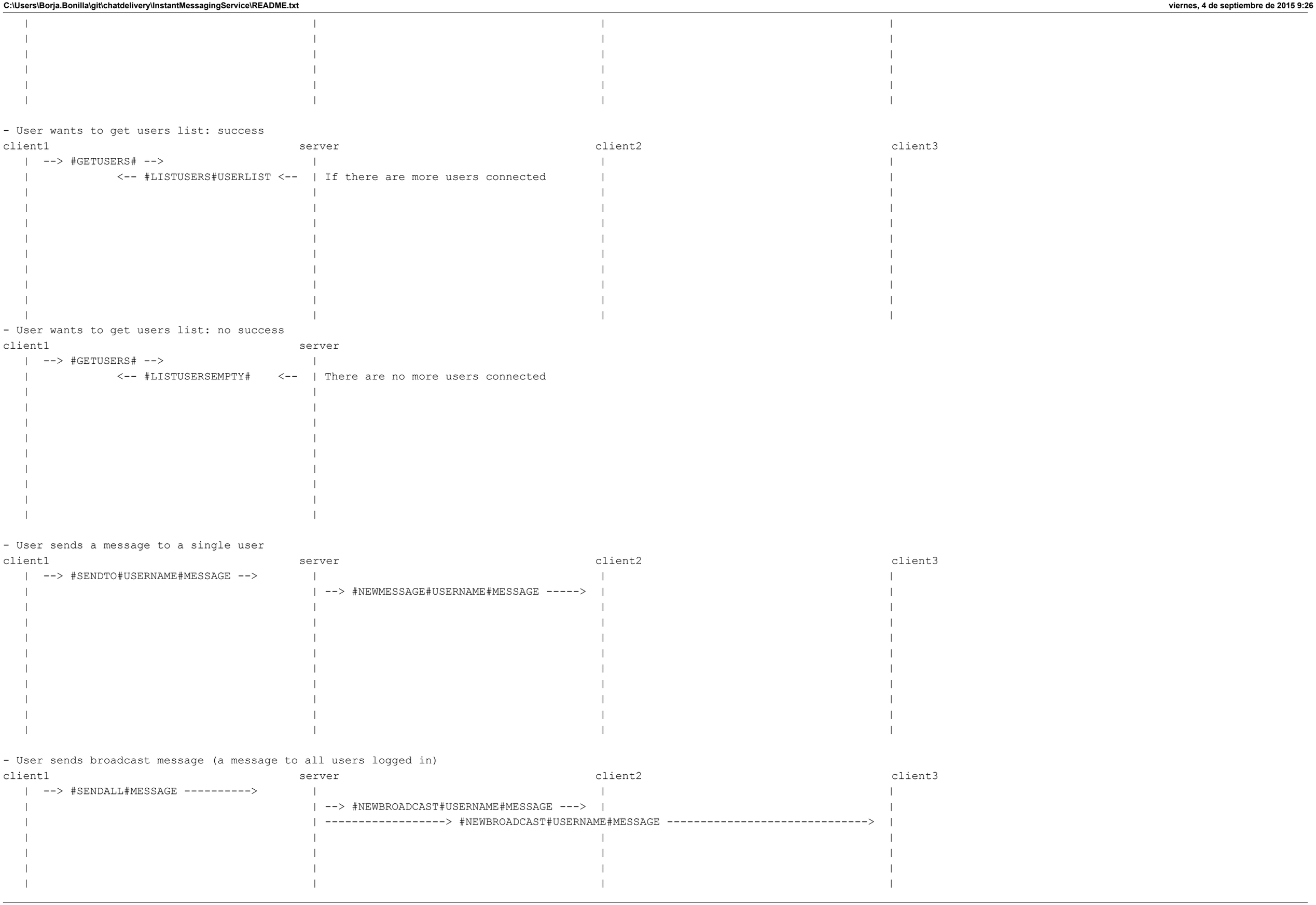
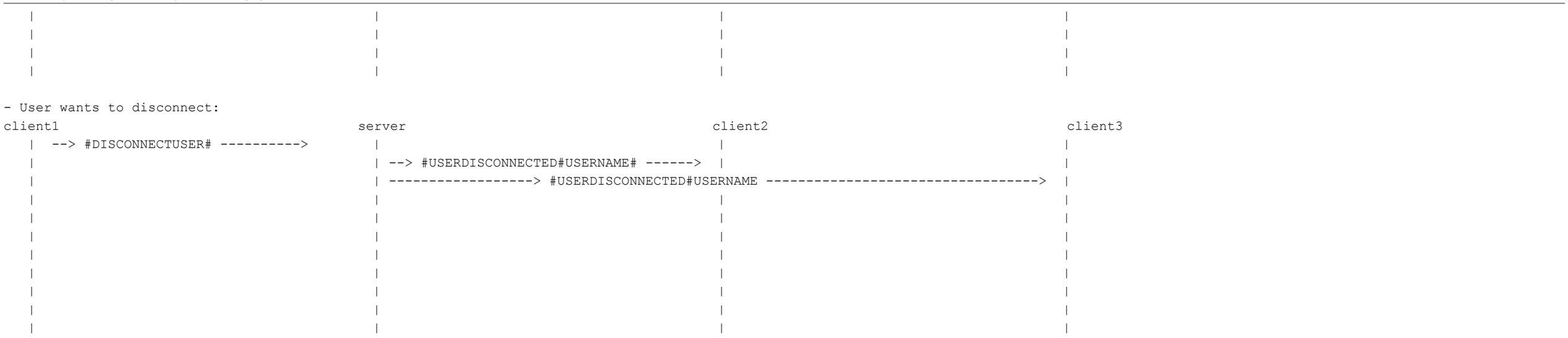
```
*****************************************************************************************************************
*                                                WELCOME TO CHAT                                                *
*                                        @author Borja Bonilla - borjabm@gmail.com                              *
*****************************************************************************************************************

*****************************************************************************************************************
*                                               Table of contents                                               *
*****************************************************************************************************************
* 1) Running server and client.                                                                                 *
* 2) To build and compile the code                                                                              *
* 3) The server used                                                                                            *
* 4) The networking protocol used                                                                               *
* 5) The chat protocol designed                                                                                 *
*****************************************************************************************************************

*****************************************************************************************************************
* 0) INTRODUCTION.                                                                                              *
*****************************************************************************************************************
Both a server and a client have been developed. To run them, please follow below instructions.
The client is a command line client: a prompt. Once it is executed, command line will display the instructions of how
to use it.

*****************************************************************************************************************
* 1) Running server and client.                                                                                 *
*****************************************************************************************************************
In the repository a runnable java file is delivered in order to directly run the project.
To run server and client go to the directory where the Speedy.jar file is located and run the following commands in the
given order:
 - To run the server: java -cp Speedy.jar com.borjabonilla.chat.SpeedyServer
 - To run the client, from a different command line window: java -cp Speedy.jar com.borjabonilla.chat.SpeedyClient

 So many clients as wanted can be executed. To terminate the client, please follow the instructions displayed in the
 command line after client execution. To terminate the server press Ctrl. + C.
*****************************************************************************************************************

*****************************************************************************************************************
* 2) To build and compile the code                                                                              *
*****************************************************************************************************************
The project is a java project done in Eclipse Version: Indigo Service Release 2 Build id: 20120216-1857. All the
external jars are included in the project.
To build it, first import the project. To do so:
- File/Import and choose "Existing Projects into Workspace"
- In the wizard, select the project location and click finish.
To build it from eclipse:
- Project/Build Project (if build automatically is not activated).
To run server from eclipe:
- Go to SpeedyServer.java in the package "com.borjabonilla.chat".
- Right click and select Run As/ Java application.
To run client from eclipe:
- Go to SpeedyClient.java in the package "com.borjabonilla.chat".
- Right click and select Run As/ Java application.
*****************************************************************************************************************

*****************************************************************************************************************
* 3) The server used                                                                                            *
*****************************************************************************************************************
To develop this project I used Jetty server. Documentation: http://www.eclipse.org/jetty/documentation/
```

```
**********************************************************************************************************************

**********************************************************************************************************************
* 4) The networking protocol used                                                                                    *
**********************************************************************************************************************
The protocol used is SPDY (Speedy). This protocol has been choosen in order to implement push notification from server
to the clients.
More information: http://dev.chromium.org/spdy


**********************************************************************************************************************

**********************************************************************************************************************
* 5) The chat protocol designed                                                                                      *
**********************************************************************************************************************
The protocol that the client and server will use to communicate with each other is based in the following headers:
    LOGIN("#LOGIN#"), // Request: Login
    LOGIN_OK("#LOGINOK#"), // Response: Login OK
    LOGIN_NOK("#LOGINNOK#"), // Response: Login NOK
    GET_ALL_USERS("#GETUSERS#"), // Request: Getusers. The client asks for all users.
    LIST_ALL_USERS("#LISTUSERS#"), // Response: Listusers. The server responds with a complete list of users.
    LIST_ALL_USERS_EMPTY("#LISTUSERSEMPTY#"), // Response: Listusersempty. There are no users.
    SEND_TO_USER("#SENDTO#"), // Request: Sendto. The client wants to send a message to a given user.
    USER_NOT_FOUND("#USERNOTFOUND#"), // Response: Usernotfound. The recipient of the message is not found or does not
                                      exist.
    NEW_MESSAGE("#NEWMESSAGE#"), // Response: Newmessage. The client receives a message from another user.
    SEND_TO_ALL("#SENDALL#"), // Request: SendAll.The client wants to send a message to all users.
    NEW_BROADCAST("#NEWBROADCAST#"), // Response: Newbroadcast. The client receives a broadcast message from
                                     // another user.
    NEW_USER_CONNECTED("#NEWUSER#"), // NewUserConnected. Clients will be notified when a new user joins.
    DISCONNECT_USER("#DISCONNECTUSER#"), // Disconnectuser. Client wants to disconnect.
    USER_DISCONNECTED("#USERDISCONNECTED#");// UserDisconnected. Clients will be notified when a logged-in user
                                            // disconnects.

                                            The message exchange pattern is as follows:
- New user wants to log in: success
client1                                     server                                      client2                                     client3
   |  --> #LOGIN#USERNAME -->                 |                                          |                                           |
   |             <-- #LOGINOK#USERNAME <--     | USERNAME is free                         |                                           |
   |                                          | --> #NEWUSER#USERNAME -----------------> |                                           |
   |                                          |      --> #NEWUSER#USERNAME ---------------------------------------------------------> |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |

- New user wants to log in: no success
client1                                     server                                      client2                                     client3
   |  --> #LOGIN#USERNAME -->                 |                                          |                                           |
   |             <-- #LOGINNOK#USERNAME <--    | USERNAME is occupied                     |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
   |                                          |                                          |                                           |
```


- User wants to get users list: success
client1
server
client2
client3
--> #GETUSERS# -->
<-- #LISTUSERS#USERLIST <--
If there are more users connected
- User wants to get users list: no success
client1
server
--> #GETUSERS# -->
<-- #LISTUSERSEMPTY# <--
There are no more users connected
- User sends a message to a single user
client1
server
client2
client3
--> #SENDTO#USERNAME#MESSAGE -->
--> #NEWMESSAGE#USERNAME#MESSAGE ----->
- User sends broadcast message (a message to all users logged in)
client1
server
client2
client3
--> #SENDALL#MESSAGE ---------->
--> #NEWBROADCAST#USERNAME#MESSAGE --->
------------------> #NEWBROADCAST#USERNAME#MESSAGE ------------------------------>


- User wants to disconnect:
client1
server
client2
client3
--> #DISCONNECTUSER# ---------->
--> #USERDISCONNECTED#USERNAME# ------>
------------------> #USERDISCONNECTED#USERNAME --------------------------------->