

## Test Project

The project was developed using VS 2013, so you will need to open it using the same. The database has four tables to exhibit EF-6 codes and n-tier architecture. A back up of the database has been provided. Database was developed using SQL 2008 R2.

Within the project folder and the solution, there is a RestSharpSerializer project that deserializes the given json string into a c# object.

To view the chat working,

1. Please restore the backed up sql table into your local sql database server.
2. Unzip the rar file content into a local folder.
3. Use VS 2013 to open the solution. (So that the demo can use IIS Express)
4. Then navigate to the "Clients" folder of the TestChat project.
5. There is a single file jqChatPage.html. Right click on it and view it in a browser. This will start up the IIS express to host the chat hub.
6. Open up couple of more browser sessions and copy / paste the url from the first one in each session.
7. Then type different names from each session and click on Logon.
8. This will initiate the chat session.
9. The logged user pane will automatically be populated with incoming users name as they log in.
10. All messages, users and room status are saved in the database.

There is no doubt that this application can be refined much further, but this should be enough to exhibit my technical expertise especially considering the fact that this is my first SignalR application. No codes have been copied from anywhere, as directed.