Real-time Chat Application with Node.js

Traditional approach

Web applications running inside a web browser are designed in such a way that the client (browser) initiates communication with the server. For instance, the browser:

1. Indicates to the server that it is interested in accessing certain service (say getting stock quotes)
2. Further tells the server what stock symbols it is interested in (say IBM or MSFT)
3. Receives the stock details from the server (such as price or percentage change)
4. Recursively polls the server for information such as updated prices

So, there is a piece of code (such as JavaScript) running in the browser constantly, polling for further data from the server.

Several improvements were made to streamline this approach with the introduction of AJAX, GWT, etc. But the basic idea is still the same—a client making requests to which the server responds.

New Approach

A different approach to address such requirements is to let the server push any data to the clients rather than having the clients “pulling” it. A few frameworks emerged lately to support this approach, one of which is WebSockets.

You can use Node.js and Socket IO to develop this project.

Requirements

You can develop a chat application with the following:

1. Allowing users to pick a name before they start chatting
2. Allowing the user to send a message to the whole room or a specific user
3. Developing UI using RWD Library (Bootstrap)
4. Using Moment API to capture message time
5. Allowing the user to use multiple rooms to join friends
6. Notifying people if a new user joins the room
7. Allowing the user to leave the chat room anytime