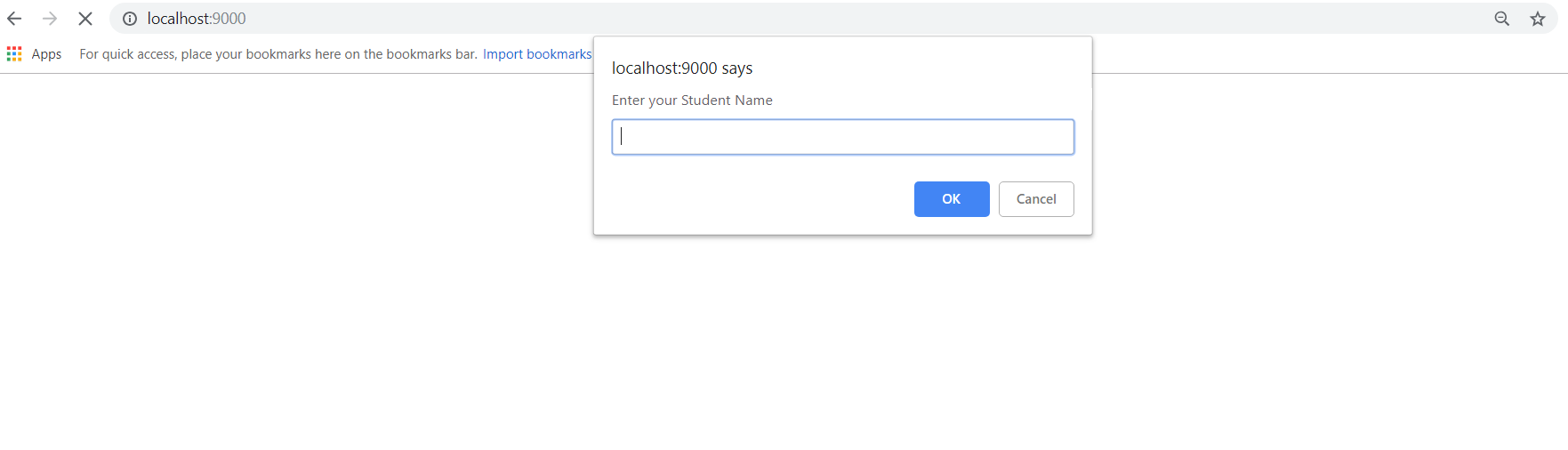
**SOFT352 Client-side Web Scripting**

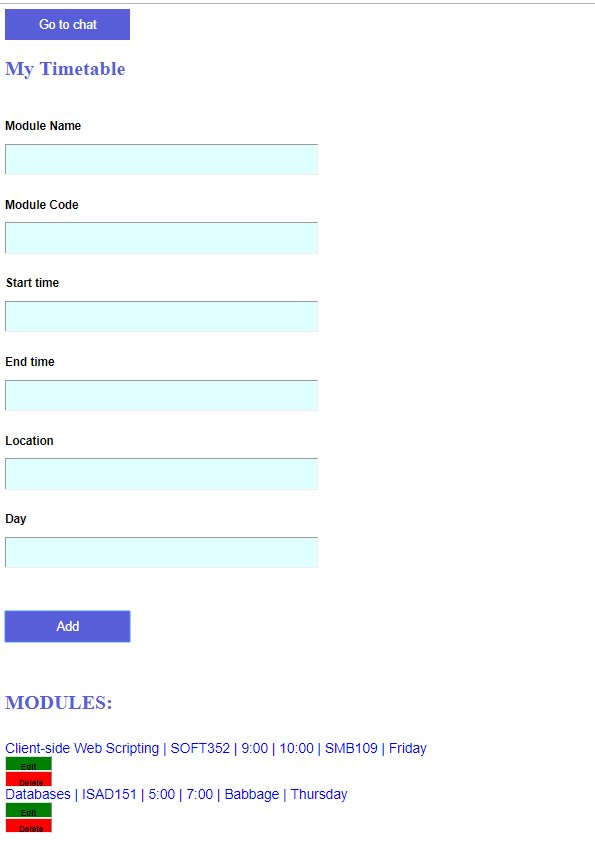
**Student Timetable and Chatroom**

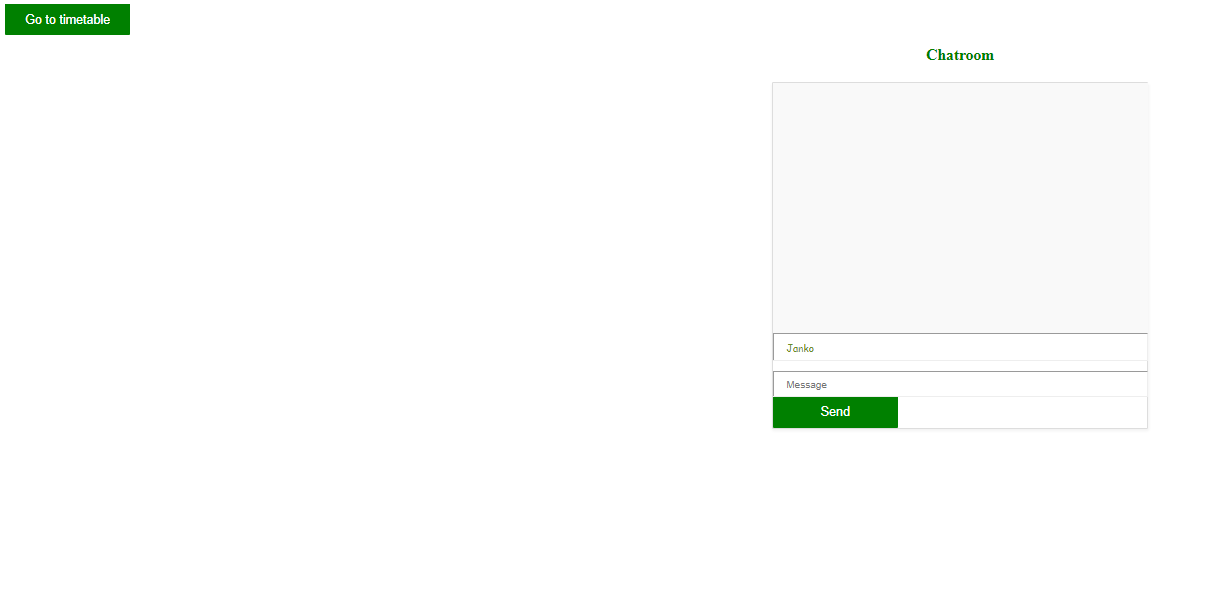
**Functionality:**

To begin with, my application is creating a timetable and creates a chatroom where people can chat with each other. The way that the users interact with my application is by entering their module name, module code, start time, end time location and day that is generating into a timetable that creates a task on their modules list. After adding the modules, the user can delete and edit in case they made a mistake. The user can also enter in a chatroom and communicate with other people. The technologies that I used to build the app are JavaScript, jQuery. Also, I used CSS to make the design of the application, html to structure my Web pages. Regarding the chatroom I used the library called Socket.IO in node.js which helped me make a connection between the server and client and send messages.

***Enter Student name:***



***Create timetable:***

***Chatroom:***

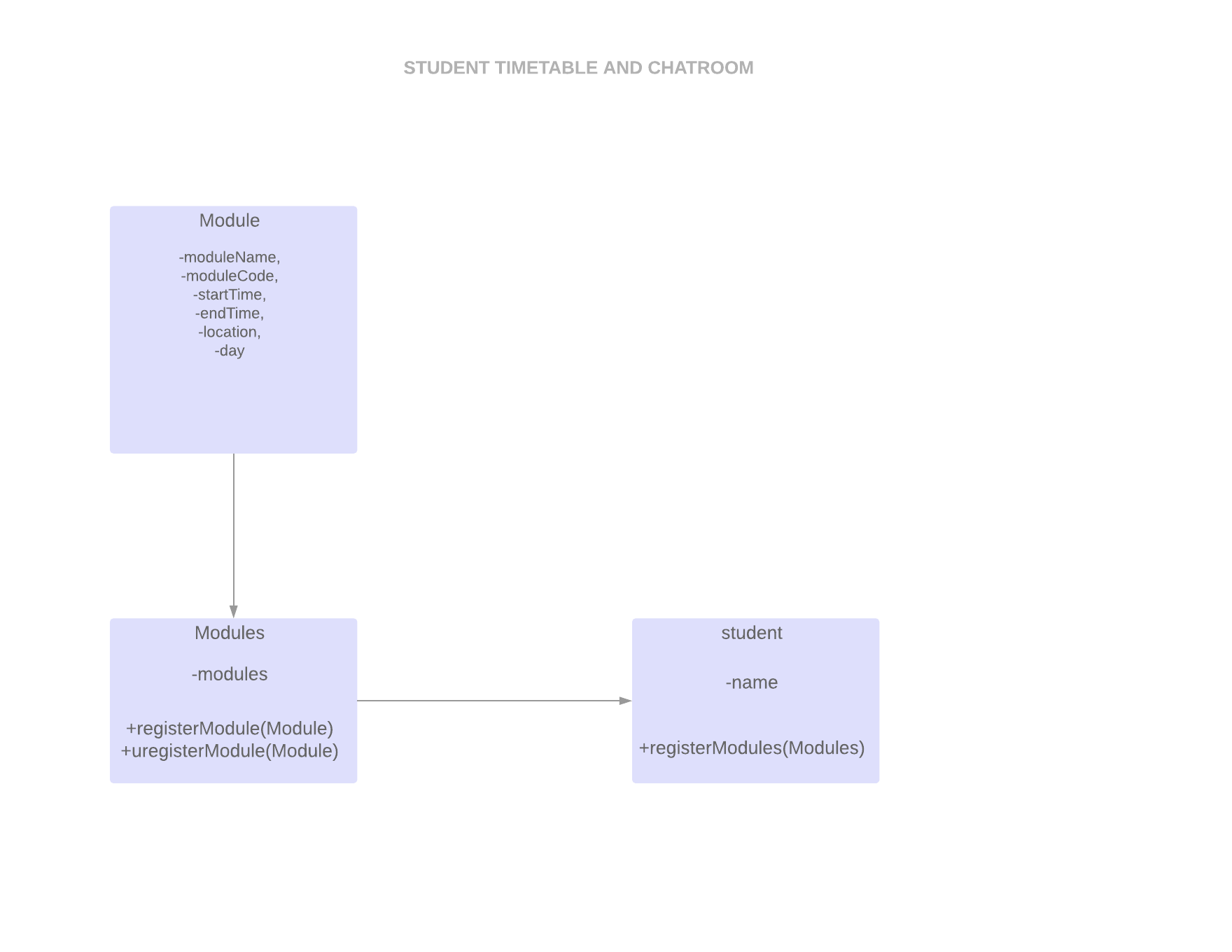
**Requirements:**

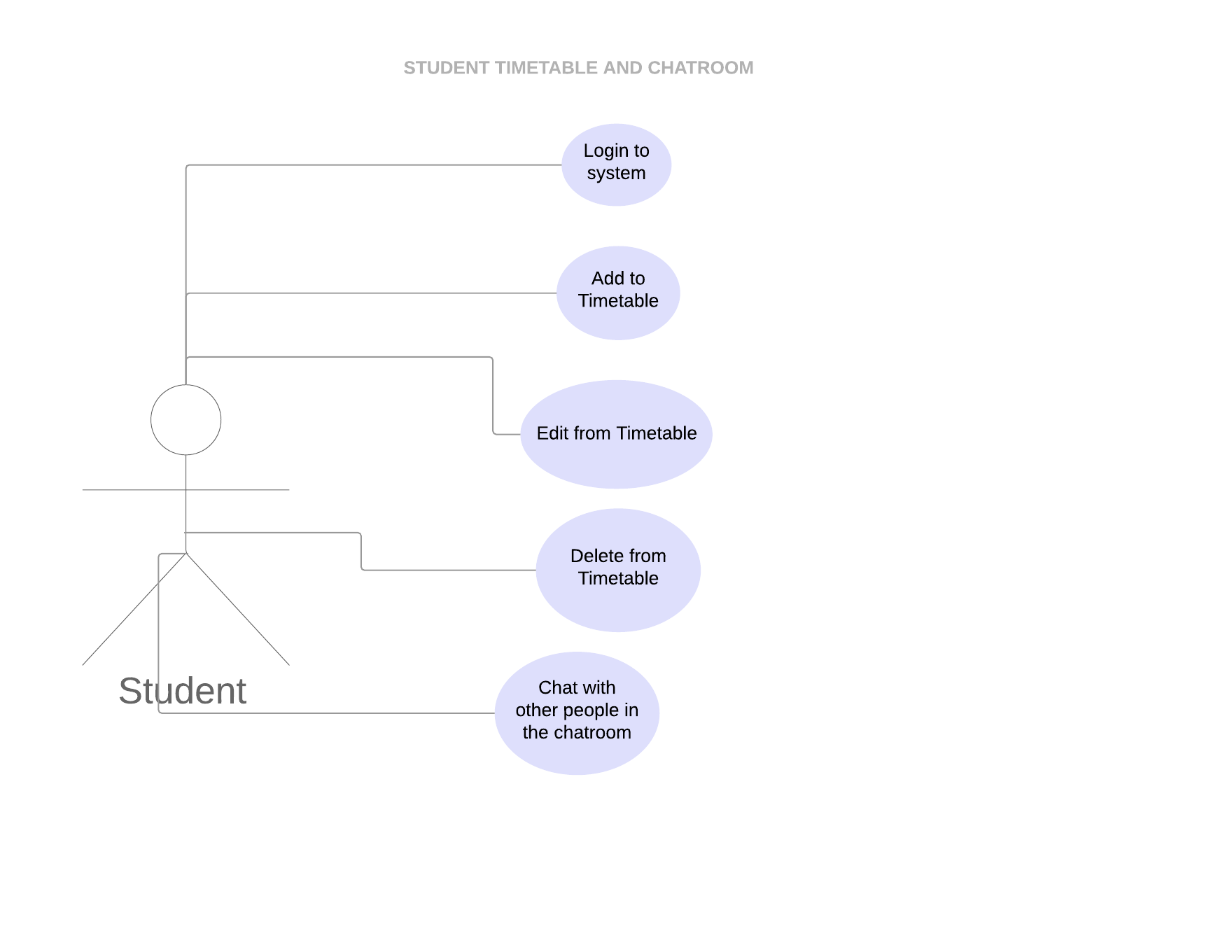
The aim of my application is to help students create their own timetable to plan their time for different modules they want, and to communicate with their colleagues and discuss different subjects of their interest. The feature that was added, is for the user to be able to delete the module tasks that they want anymore and to edit in case they inserted it in the list and it needs to be changed. I used QUnit testing to see if my module class is created correctly.

**Design:**

My web Application has both client-side and server-side functions. The processes that I have on my server-side is that the server is getting requests from the clients that want to send data to each other that are connected to the server, so the server’s job is to open the sockets and emit the data which in my case are messages, to all the other clients that are connected to him. Regarding the messaging the client-side is firstly making a connection to the server and afterwards it is sending the data to the server that wants to emit. I am creating three classes which are module, modules and student where the module class inherits into modules and the modules class into student to see the registered modules of the specific student that enters the web application. Also, when the student enters the chat room and then return to enter something more in the timetable, the data that he entered before are not getting lost. However, if the user is in the web application the previous chat that the user had is not getting lost.

***User Class diagram:***



***User case Diagram:***

**Performance:**

In order to make my system have no delays when it comes to the server sending the data to the clients because I implemented it in a way where the socket connection with the server happens as soon as the application opens. In order to make my website more efficient I used Ajax where it is makes moving from one page to another faster.

**Development Process:**

When I started building the project I needed a lot of time creating and recreating the classes in order to make sure that everything is going to work fine, and that I will have a nice structured system. After finishing with creating classes I started implementing the code and creating my application using JavaScript, jQuery and CSS. Then I created my server in Node.js using Socket.IO that was going to help me create my chatroom, by getting connection between the clients and the server.

**Persona Reflection:**

All in all, beginning with my web application my main goal was to create an application where the students could track their modules by creating them by them self and to create a webchat where they will be able to communicate with their colleagues. The things that I would like to improve is the visual appearance as It will make it more user friendly and easier to use. Also, I would improve the webchat where I could put the times when the messages where sent and informing and also the user to get notifications when a new message appears in the chatroom while creating the timetable. The lesson I would like to bring to my next project is to research more, and find the most recent libraries, frameworks etc. in order the project to be and look up to date as possible.