

# Web Development Reflective Report

## Introduction

This report aims to communicate my experience and thought process while developing a personal web portfolio site, a blog and a chat application using a library called socket.io.

## Planning

My first approach to this project was to use a tool called [Draw.io](#) to develop a wireframe of the website I planned to build. This was very good practice as this served as a building block for directing the remainder of my design architecture.

## Functionality

The next step I took was to analyze the sample code that was provided. The sample code provided a minimalistic but functional chat application. I aimed to understand the function of each line of code in the Clients.js and Index.js to comprehend how the chat communication channel worked from client to server and broadcasted to other users. I realized that a library called [Socket.io](#) was being used. I visited the documentation and consulted several YouTube Videos to understand how to build my implantation of this chat application from scratch. This was a time-consuming task but was a very valuable process because of the knowledge I gained.

Socket.io works using web sockets. “Web Sockets allow full duplex communication in an HTML5 compliant browser over a single socket” (Gupta, Bhumij and Vani, MP, 2018). In the chat application I developed, The Nodejs server and Socket.IO server run on the same port (5000). The server listens for a new connection event. When a new connection event is triggered by a new user from the client, this connection request is sent to the server. If permission is granted by the server, a two-way connection is made. This connection is called a “Handshake” (Pimentel, Victoria and Nickerson, Bradford G, 2012). When the client sends a message, the socket emits a message to the server. The server listens for this event and broadcasts this message to all active users except the user from which the message originates. The client also handles this event and prints the message on the screen for the other active users.

## Design

I decided to use Bootstrap because it has classes that make HTML elements more beautiful and helps with building a responsive website design. I also included a custom CSS file after including Bootstrap so that I have more control over my design by overwriting some of the default behaviour of Bootstrap classes. I also included jQuery and used the animate function to add effect to my welcome message on the site. After building through the basic design I had in the wireframe I looked up some existing templates to get some inspiration. I then proceeded to build my design.

## Summary

I have learnt a lot from this development experience. Top on my list is that planning is a very important step when designing the front end of an application. I lost a lot of time because I tried to design without proper planning and this led to me getting stuck several times.

## References

Gupta, Bhumij and Vani, MP, 2018. An overview of web sockets: The future of real-time communication. *Int. Res. J. Eng. Technol. IRJET*, 5(12), p. 4.

Pimentel, Victoria and Nickerson, Bradford G, 2012. Communicating and displaying real-time data with websocket. *IEEE Internet Computing*, 16(4), pp. 45-53.