

# 고급소프트웨어 실습 9주차 과제 보고서

20171693 조병화

## 1. websocket 과 socket.io 통신 방식에 대해 자세히 조사한 후 이에 대해 기술하라.

실시간 애플리케이션을 구축 할 때 클라이언트와 서버 간의 실시간 데이터 교환을 구현하는 대표적인 두가지 방식으로 webSockets 와 Socket.IO 가 있다. 이 방법들은 최신 웹에서 실시간 통신을 구현하는데 가장 많이 사용되는 두 가지 솔루션이다. 그렇다면 각각의 통신방식에 대해 알아보고, 공통점과 차이점을 찾아보도록 하겠다.

먼저 WebSocket 은 2010 년 chrome 4 에 처음 등장하였으며 단일 TCP connection 을 통해 전이중 통신 채널을 제공하는 웹 통신 프로토콜을 참조한다. 즉, 최소한의 오버 헤드로 클라이언트와 서버 간의 상호작용을 가능하게 한다. 따라서 실시간 통신이 필수적인 애플리케이션을 구축할 때 사용된다. WebSocket 은 채팅앱, 멀티유저 게임, 공동 편집, 위치 기반 애플리케이션 등에 사용된다.

Socket.IO 의 경우는 위의 WebSocket 등의 기술 위에서 구축 된 자바 스크립트 라이브러리이다. 현존하는 대부분의 실시간 웹기술 (websocket, flash socket 등)들을 추상화해 놓았다. 따라서 간혹 WebSocket 이 지원되지않는 환경에서도 Socket.IO 를 사용할 수 있다. 스스로 브라우저의 종류와 서버의 종류를 파악하여 실시간 웹을 구현할 수 있다. 이는 websocket 과 달리 표준기술이 아닌 한 회사의 오픈소스이다.

Websocket 은 기존의 http 의 stateless 한 통신과 달리 연결이 유지되는 성질이 있으나. 실험적인 부분들이 아직 존재하기 때문에 안전성이 더 뛰어난 Socket.io 라이브러리를 활용하기도 한다. 즉 websocket 과 socket.io 의 대표적인 차이점이라 한다면 websocket 은 양방향 통신을 위한 ‘프로토콜’ 이고, socket.io 는 양방향 통신을 위해 websocket 기술을 활용하는 ‘라이브러리’ 라는 것이다 . 즉, websocket 은 HTML 5 웹 표준 기술이며 Socket.IO 는 표준기술이 아니며 라이브러리이다.

Websocket 는 매우 빠르게 작동하며 통신할 때 적은 데이터만을 이용한다. 하지만 이벤트를 단순히 듣고, 보내는 것만 가능하다. 반면 Socket.io 는 연결 실패 시 fallback 을 통해 다른 방식으로 알아서 해당 클라이언트와의 연결을 시도한다는 차이점이 있다. 또한, 일부

클라이언트한테만 데이터를 전송하는 브로드캐스팅 이 가능하다는 차이점 또한 존재한다.

그렇다면 어떤 상황에 어떤 방식을 사용하여야 할까?

서버에 연결된 소켓들을 세심하게 관리해야 할 필요가 있는 서비스를 만들 경우 broadcasting 기능이 있는 socket.io 를 쓰는게 유지,보수의 측면에서 더 많은 이점을 가질 것이다.

반면, 데이터의 전송 량 이 많은 경우(ex. 가상화폐 거래소) 에는 빠르고 비용이 더 적게 드는 표준 websocket 을 사용하는 것이 더 바람직 할 것이다.

## 2. 실제로 websocket/socket.io 통신 방식을 사용해서 실제 채팅 프로그램을 구현하게 됐을 때의 전체적인 구현 방식 기술

이번엔 실제로 두가지 방식을 이용하여 채팅 프로그램을 구현했을 때 구현방식의 차이를 기술해 보겠다.

Websocket 을 이용해 채팅프로그램을 구현하는 방식은 다음과 같다. 우선, 서버를 생성하고, 클라이언트가 연결할 때마다 세션에 고유한 ID 를 할당한다. 클라이언트가 메시지를 보내면 [`<client-id>`]: 의 방식으로 회신한다. 이를 통해 메시지가 성공적으로 전송되었음을 확인할 수 있는 것이다. 만약 연결된 모든 클라이언트들에 메시지를 브로드캐스트하려면 어떻게 해야 할까? Websocket 는 기본적으로 브로드캐스팅을 지원하지 않지만 ‘바닐라 웹 소켓’을 이용하여 구현할 수 있다.

이번에는 socket.io 를 이용한 채팅 프로그램 구현 방식을 알아보도록 하겠다. on 함수로 connection 로 이벤트를 연결한다. Socket.io 변수로 http 객체를 전달받으면, 서버로 들어오는 socket 연결들을 listening 할 수 있다. 이 때 창을 여러개 띄우면 user connect 가 2 개 생성되는 것을 확인할 수 있다. 이제 socket.io 연결을 끊는 방식도 구현해보겠다. Io.on 함수에 disconnect 이벤트를 추가하면 접속 후 브라우저를 닫았을 때 연결이 끊어졌다는 알림이 옴을 확인할 수 있다.

출처 : <https://ichi.pro/ko/websocketgwa-socket-ioui-chaijeom-186805209650140>  
<https://www.peterkimzz.com/websocket-vs-socket-io/>