- kumunikacja w obie strony server i przeglądarka

- najpierw handshake a potem komunikacja w obie strony

- java support for web sockets - jEE

- endpointy programistycznie bądź przez adnotacje

- text, binary data

- mozna konwertery np. Json

- mozna wypychac informacje do podłączonych

- jest kod js dla komunikacji, uzywa socket.onmessage socket.onclose itp.

- ws:// - web socket url np. Ws://http:/websocket/chat

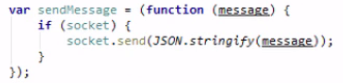
- w js też tworzenie websocket w zalezności od przegladarki

- event handlers tworzone jak wczesniej socket.onopen = function() {if(handler) {handler.onopen()}};

-- socket.onmessage = function(message) {handler.onmessage(message.data);}

-- spr czy handler jest jak w onopen

- sendMessage funkcja śle json do serwera



- po serwerze

-- message - POJO

-- endpoint



- encoder i decoder + w dekodoreye dekodowanz pryzchodyacz json

- jest tryzmana lista klientw i do nich wysyłana informacja

Dobre wsparcie dla springa – obadac

<https://spring.io/guides/gs/messaging-stomp-websocket/>

<http://www.concretepage.com/spring-4/spring-4-websocket-sockjs-stomp-tomcat-example>

<https://gerrydevstory.com/2014/03/04/spring-4-websocket-stock-ticker-example/>