# OOR Ćwiczenie 6 – Web Socket

Autor: Kornel Ostrowski

Grupa: 125NCI\_B

Link do projektu: <https://github.com/ostrowskik/WebSocket_oor_cw6>

Język programowania: JavaScript

**Zadania do zrealizowania:**

1. Za pomocą wybranego przez siebie języka programowania zademonstruj działanie technologii Web Socket.
2. Bądź gotowy do prezentacji możliwości i modyfikacji stworzonej aplikacji.

**Przykładowe pytania**

* Przedstaw idęę i zasadę działania technologii (protokołu) Web Socket.

Ideą WebSocket jest uzyskanie połączenia dwukierunkowego o niskich opóźnieniach oraz „pozbycie się” dodatkowych pluginów w przeglądarkach – np. w grach lub komunikatorach przeglądarkowych. Obiekt WebSocket reprezentuje gniazdko w przeglądarce, które jest połączone z serwerem, gdzięki czemu pzez adres IP i port można utrzymać obustronny kanał do serwera, a tym samym prowadzić komunikację dwustronną.

* Opisz metody i zdarzenia związane z Web Socket.

Socket.readyState – atrybut tylko do odczytu, czy połączenie jest ustanowione. 0 – nie. 1. – ustanowione i komunikacja jest możliwa. 2 – połączenie w trakcie zamykania. 3 – połącznie zamknięte lub nie może być otwarte

Socket.BufferendAmount – ilość bajtów w kolejce ustawionych przez metodę send()

Socket.onOpen – kiedy połączenie jest nawiązane

Socket.onMessage – kiedy klient otrzymuje dane z serwera

Socket.onerror – kiedy błąd komunikacji

Socket.close – kiedy połączenie jest zamykane

Socket.send() – wysyła dane używając połączenia

Socket.close() – zamyka istniejące połączenie

* Porównaj Web Socket z protokołem HTTP.

WebSocket jest dwukierunkowy, http jednokierunkowy. WebSocket pozwala na utrzymywanie połączenia pzez cały czas, http jest tylko protokołem pytanie-odpowiedź, po którym połączenie jest zamykane. WebSocket i http oferują szyfrowanie. WebSocket nie jest wspierany przez starsze przelgądarki.

# Realizacja zadań

Do realizacji zadania użyto darmowego serwera echo WebSocket ws://echo.websocket.org/

Aplikacja pozwala na: połączenie z serwerem, odłączenie od serwera oraz wysłanie dowolnej wiadomości na serwer. Po wysłaniu dowolnej wiadomości na serwer ten odpowiada tym samym, co otrzymał. Aby otrzymać wiadomość odpowiedzi z serwera i aby wysyłana wiadomość doszła na serwera, użytkownik musi być połączony z serwerem.