**DOKUMENTACJA PROJEKTU**

Systemy udostępniania usług sieciowych

**AGH**

Wydział Informatyki, Elektroniki i Telekomunikacji

Michał Kostrzewa

Temat projektu:

Wykorzystanie technologii Kaazing WebSocket Gateway wraz ze wsparciem JMS do sterowania gry (technologia Construct2) poprzez urządzenia mobilne (Android)

**Idea projektu**

Celem projektu jest umożliwienie **n** użytkownikom sterowanie postaciami w grze uruchomionej na przeglądarce poprzez urządzenia mobilne.   
Do tego celu potrzebny będzie nam serwer Websocket Gateway odpalony na lokalnej maszynie - a dokładniej:  
apache-activemq oraz kaazing-websocket-gateway-jms  
Stworzona prosta gra w Construct2, oparta o HTML5 wraz ze wsparciem JavaScriptu, w którym to zdefiniujemy nasze połączenie do serwera.  
Na urządzeniu mobilnym zmodyfikowana aplikacja do łączenia się z serwerem i sterowanie graczem.  
Użytkownik powienien mieć możliwość połączyć się z serwerem, zasubskrybować temat, a następnie wybracz gracza do sterowania - po czym za pomocą przycisków na ekranie dotykowym - sterowanie nim.

**Wymagania funkcjonalne**

Podstawowym wymaganiem jest skonfigurowanie odpowiednio serwera - podając adres IP lokalnego komputera. Teraz możemy tworzyć skończoną ilość połączeń na ten serwer. Wykorzystywać będziemy połączenie z Subskrybcją.

Kolejnym wymaganiem jest stworzenie **Producera** do którego będziemy się łączyć oraz **Customera**, który to będzie nawiązywał połączenie. O tym dokładniej za chwilkę.  
Aby tego dokonać najpierw jednak musimy stworzyć grę przy użyciu programu Construct2 - dodać w niej graczy - oraz (najważniejsze) umożliwić wczytywanie skryptów JavaScript, dzięki którym, możemy z zewnątrz sterować naszą grą.  
<screen>

Gotową grę wraz z dodatkiem JS, należy wyeksportować, aby móc ją otworzyć w przeglądarce.  
Następnie należy odpowiednio zmodyfikować skypt JS - w którym to nawiązujemy połączenie z naszym lokalnym serwerem, o którym wspomniałem wyżej.

Oprócz nawiązania połączenia - tworzymy Producera, oraz własny temat -  **topic**, pod którym będą dostępne funkcje sterowania graczem.

Korzystać będziemy z dobrodziejstw **Construct2**, za pomocą jego API możemy wywołać funkcje zdefiniowane bezpośrdenio w programi Construct2 - w naszej prostej grze.

A więc pod naszym tematem, zakładamy listener nasłuchujący przychodzące wiadomości pod wskazany topic.  
Listener wywoła odpowiednią funkcję, która będzie rozdzielała przychodzące wiadomości - dotyczące gracza pierwszego lub drugiego wraz z jego kierunkiem poruszania.

Typem wiadomości będzie prosty TextMessage z odpowiednią formułą po której rozpoznamy nazwę funkcji do wywołania.

Cenną zaletąjest tutaj fakt, iż podany temat mogą subskrybować **n** użytkowników, co może się przydać w przyszłości podczas np. rozbudowy tej gry o kolejnych graczy.  
Możemy sobie wyobrazić następującą sytuację:  
Uruchamiamy serwer lokalnie bądź zdalnie, który uruchamiają użytkownicy chcący pograć w grę - za pomocą telefonu każdy z graczy steruje swoją postacią.

To jedna z wielu możliwości, jaką nam daje Kaazing.

Kolejnym etapem jest pobranie oraz zainstalowanie na telefonie z systemem Android aplikacji ze strony Kaazing do połączenia się za pomocą JMS z serwerem/topiciem.

Wymaganiem w tym przypadku będzie dodanie nowego okna aktywności z wykorzystaniem **Android Strudio** do sterowania graczem.

A zatem musimy dodać przyciski dół/góra/lewo/prawo oraz umożliwić wybór gracza Player1/Player2.  
Po nawiązaniu połączenia w klasie Main - należy podać adres IP naszego lokalnego hosta oraz wpisać temat.  
Po wciśnięciu przycisków - będą **wysyłane** wiadomości JMS ! - Które to będzie nasłuchiwał **Producer** w naszej grze - a konkretnie w skrypcie JS .

Producer wywoła odpowiednie metody zdefiniowane w **Construct2** + które to znowu są odpowiedzialne za poruszanie się odpowiednim graczem.

**Stos technologiczny**

Do stworzenia tego projektu potrzebujemy następujących technologi.

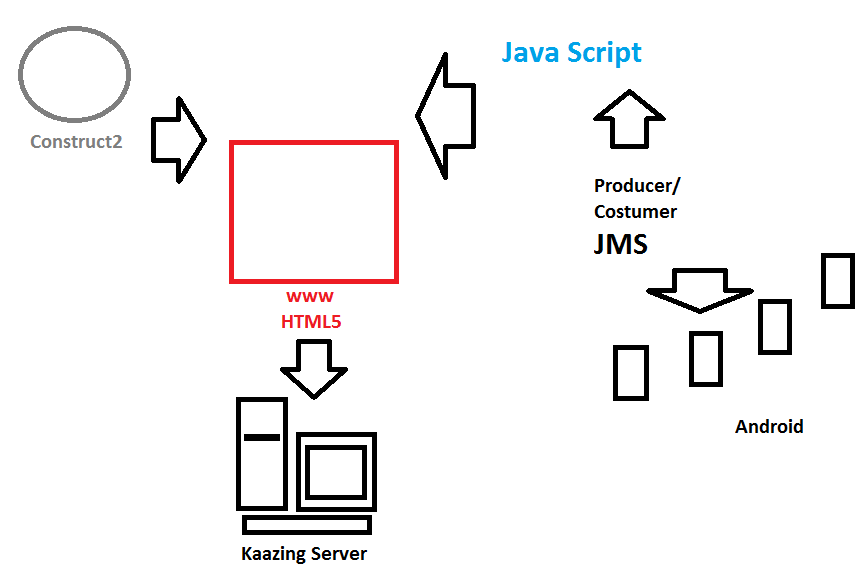
Oczywiście Kaazing - dostarczający nam serwer Websocket wraz ze wsparciem dla JMS.

JavaScript - dzięki któremu możemy stworzyć Producera JMS, listenera nasłuchującego, metody służace do odbioru wiadomości oraz możliwość połączenia się z zewnątrz z naszą grą!

Construct2 - technologia służaca do robienia gier na przeglądarki z wykorzystaniem HTML5

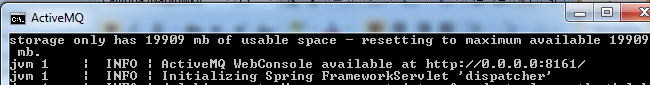
Java - a konkretnie budowanie aplikacji dla Androida

**Ogólna architektura**

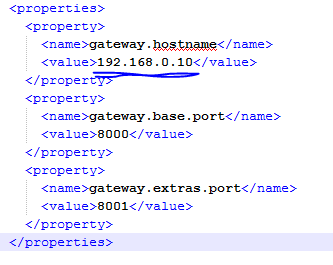


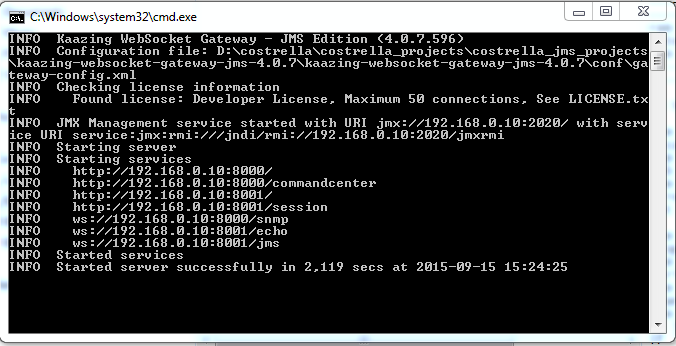
**Instrukcja instalacji i konfiguracji systemu**

Uruchamiamy :

activemq 

Kaazing WebSocket Gateway - odpowiednio skonfigurowany

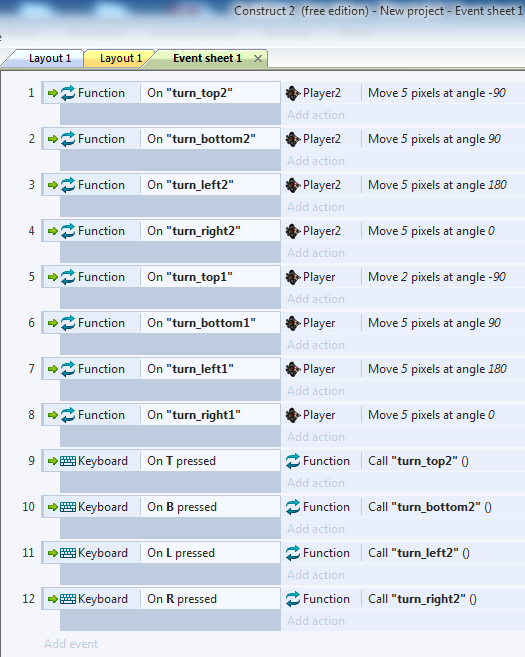




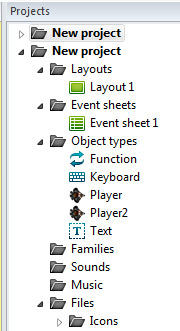
Dodajemy graczy:



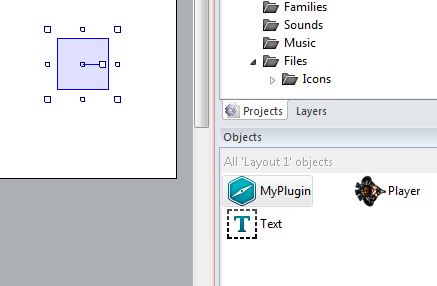
Definiujemy odpowiednie funkcje w Construct2



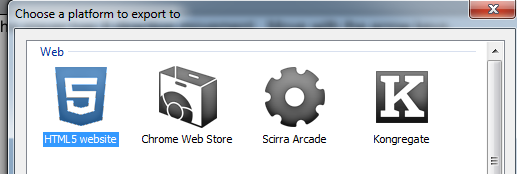
Widok projektu - ważne jakich używamy **Object type:**



Pozatym potrzebny jest też własny zdefionowany plugin - nie będe rozpisywał, co do tego jest nam potrzebne - tutorial znajduje się w internecie.



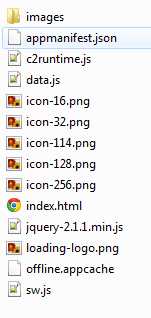
Jeśli mamy gotowy projekt w Construct2 - musimy go wyeksportować:



Wyeksportowane pliki kopiujemy do kaazing:

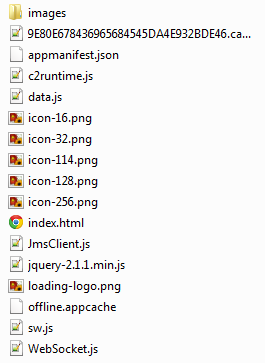
**kaazing-websocket-gateway-jms-4.0.7\web\extras\**

W moim wypadku eksport wygląda następująco:



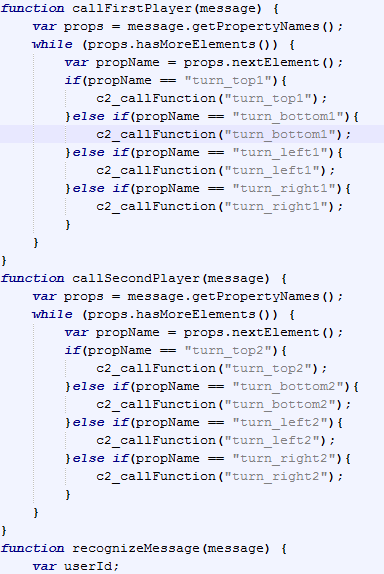
Jednak należy dodać odpowiednie biblioteki, aby można było korzystać z dobrodziejstw **JMS** - a konkretnie **JmsClient.js** oraz **WebSocket.js**

w pliku **c2runtime.js** mamy możliwość definiowania własnych funkcji, a więc końcowy stos plików:



Przechodzimy do pliku c2runtime, ważniejsze funkcję:





**c2\_callFunkction - wywołuje funkcje Construct2**