**User Guide**

**jWebSocket**

**Audio and Video transmition using jWebSocket Framework**

**1.0**

**Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 24/04/2012 | 1.0 | Document Creation | Alexander Rojas Hernández |

**1. Characteristics of the solution**

The transmition of audio and video using jWebSocket framework guaranty a real time flow communication under a WebSocket Protocol.

The systems that currently uses Streaming technology are based in a Client-server (C2S) structure using Real Time Protocol(RTP) as a protocol communication, which provides a good images and sound synchronization. However the use of RTP for this technology has disadvantages: It´s not guaranty a continue received of the packets send from the server. It´s can´t avoid the disorder received of the packets. In spite of the RTP is supported by UDP to make a fast send of the information, does not guaranty high levels of precision, disponibility and security.

The demostrative application for audio and video transmition realized for Web using Java Media Framework(JMF) and a native application for the Smartphone’s using PhoneGap, both applications uses jWebSocket Sever that allows the received the codified packets from the client, the broadcasting to all clients that are connected . With the native application it can access to the Smartphone video, audio and image controllers. The packets are processed in the client and send it to the server for the broadcast; the other client received the packets and decodified to visualize it. In the stationary application the process is similar. These applications guaranty a real time using WebSocket Protocol and a bidirectional work with high precision, besides guaranty the security and disponibility of the information.

The aspects mentioned before: real time and bidirectional communication supports a new Web Communication Paradigm. It´s possible using jWebSocket framework, a new technology oriented to the Web application development over the WebSocket communication protocol.

**2. Mean features**

The stationary application developed for the audio and video transmition using the jWebSocket framework presents the following functionalities:

* Capture of video from the webcam
* Visualize the content using a java applet
* Encode the capture data
* Send the data to the server
* Broadcast the data in the server
* Received and decode the data

The native application developed for the audio and video transmition using the jWebSocket framework presents the following functionalities:

* Capture the image, audio and video controllers from the Smartphone.
* Encode the capture data
* Send the data to the server
* Broadcast the data in the server
* Received and decode the data
* Visualize the data

**3. Problems to be solved**

|  |  |
| --- | --- |
| **Problem** | **Contributions of the solution** |
| The audio and video transmition does not realize in real time using WebSocket protocol. | The application allows the audio and video transmition in real time using the WebSocket protocol, could be adapted to another clients. |
| The disponibility of the information is affected in the transmition when used a RTP | The application allows that doesn´t lose packets in the transmition, the server always controls the client connection. |

**4. Glossary of Terms**

**Streaming:** Technology that allows transfer encode packets over the network, processing in a regular and continue flow. After the packets are decodes and visualized in the client.

**Real Time in the Web:** Technologies and practices that allows to the users received information as soon that are published by the authors instead to checks the source continuously.

**Websocket:** The WebSocket protocol defines the procedures to update the connection using HTTP to a connection using WebSocket full-bidirectional with TCP. The client send a HTTP GET request to establish a WebSocket communication to the server. After the communication stays active until it´s closed, allowing interchange messages between the client and the server.

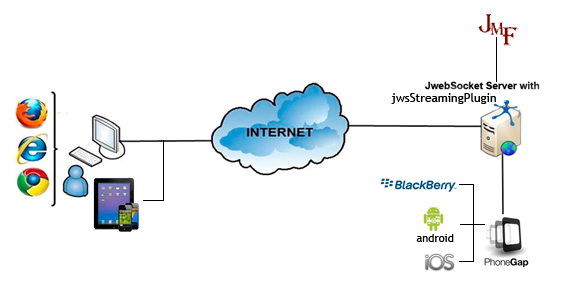
**jWebSocket:** New technology oriented to the applications development based in WebSocket protocol that gives high levels of speed, scalability, security and real time work, main element to the network.

**JMF:** Java Media Framework is a framework that allows the work with the audio and video functionalities from pc(e.g Webcams), which add to java applications and applets.

**PhoneGap:** PhoneGap is an Open Source framework to create native applications for Smartphone’s, using HMTL, CSS and JavaScript, its supported by different Operative Systems like iOs, Android, BlackBerry, Palm webOS, y Symbian WRT (Nokia).

**5. Solution Model**

Ejemplo:

****

**7. Requirements for use**

The application has the advantage to work over different Operative System(for the moment, only test in Windows), it´s needs the following elements:

* OpenJDK 1.7 or higher installed.
* OpenJRE 1.5 or higher installed in the client side (Stationary Application).
* Specify the enviorment variables for jmf library and jWebSocket server. (Stationary Application).
* jWebSocket Server executed.
* Broswer that support WebSocket (Stationary Application).
* Webcam to the capture proccess
* Smartphone (Android 2.3).

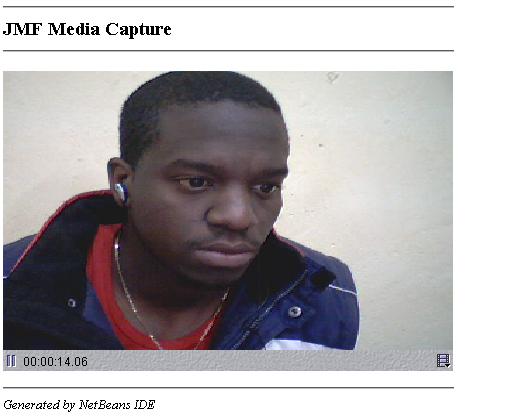
**8. Solution roles**

**System User:** Is able to start the capture process and the transmition to the server.

**9. System functionality**

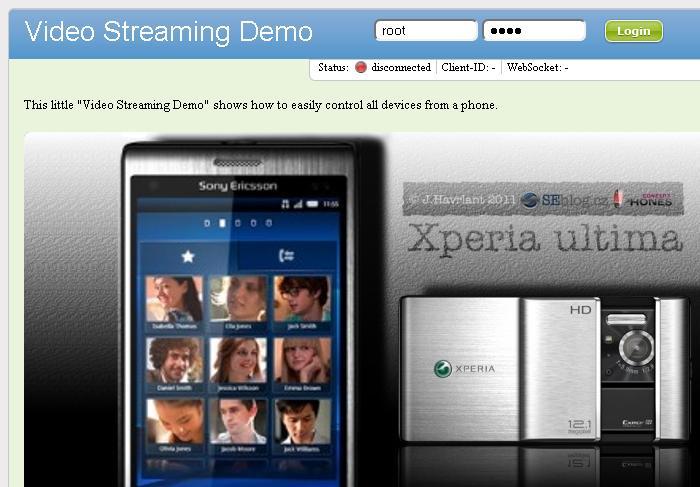
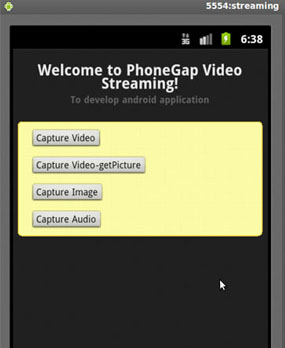
**Main view for the stationary application**

**1**

****

1. Visualization in a java applet the capture using Java Media Framework.

1. **Main view for the native application**

****

**5**

**4**

**3**

**2**

**1**

1. Access to the video capture controller in the Smartphone.
2. Search a video stored in the SmartPhone.
3. Access to the image capture controller in the Smartphone.
4. Access to the audio capture controller in the Smartphone.
5. Player to show the content send from the server.

**9. Solution Configurations**

Para poner en funcionamiento la aplicación estacionaria, se le debe de especificar en las variables de entorno la utilización de la librería jmf.jar.

For the stationary application configure in the enviorment variables for jmf and jWebSocket server.

|  |
| --- |
| *$JMFHOME/C:/ARCHIV~1/JMF21~1.1E/lib/jmf.jar*  JWEBSOCKET\_HOME/C:/jWebSocket/ |

The configuration can be realized for the system users

**10.Solution Rules**

**R1:** If the pc where is jWebSocket server doesn´t started correctly the capture process can´t be initialized. The clients has to be configured the jvm in the broswers with supports for Websocket.

**R2**: The dispositive for the audio and video capture (Stationary application).

**R3:** The native application must be present the Wifi connection