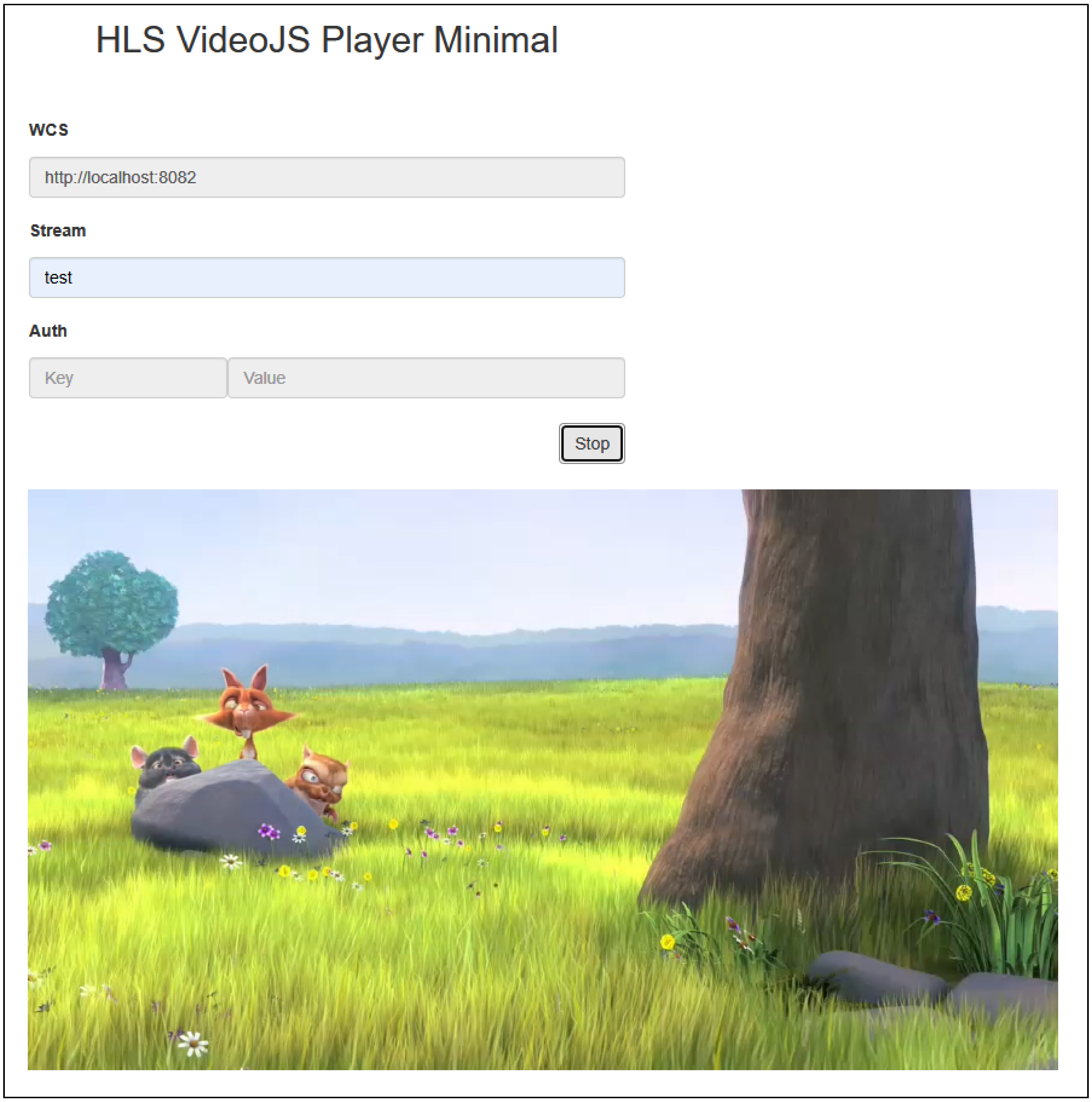
HLS VideoJS Player

Example of stream convertion to HLS and playing it in browser using VideoJS

The player shows how to convert stream published on WCS server to HLS and play it in browser. HLS stream cut starts automatically when strea is requested by HLS URL, for example http://localhost:8082/test/test.m3u8on the screenshot below


HLS VideoJS Player Minimal

WCS

http://localhost:8082

Stream

test

Auth

Key Value

The code of the example

The source code can be accessed on server by the following path:

/usr/local/FlashphonerWebCallServer/client2/examples/demo/streaming/hls-player

hls-player.css - player page styles file
video-js.css - HLS player styles file
hls-player.html - player page
hls-player.js - player launch script
player-page.html - common player page elements for three HLS playback examples
video.js - player script (http://videojs.com/, Apache License Version 2.0)
videojs-hls.min.js - player script (minimized)

The example can be tested using the fiollowing URL:

https://host:8888/client2/examples/demo/streaming/hls-player/hls-player.html

Where host is WCS server address

Analyzing the code

To analize the code get hls-player.js file version with hash ecbadc3, which is available here and can be downloaded in build 2.0.212.

1. A server HLS URL detection

getHLSUrl() code

```
function initPage() {
    $("#header").text("HLS VideoJS Player Minimal");
    $("#urlServer").val(getHLSUrl());
    ...
}
```

2. Player initialization

videojs() code

A div element for stream playback is passed to player

```
function initPage() {
    ...
    var remoteVideo = document.getElementById('remoteVideo');
    remoteVideo.className = "video-js vjs-default-skin";
    player = videojs(remoteVideo);
}
```

3. Stream name detection (the stream should be published to server)

encodeURIComponent() code

```
function playBtnClick() {
    if (validateForm()) {
        var streamName = $('#playStream').val();
        streamName = encodeURIComponent(streamName);
        ...
    }
}
```

4. HLS stream URL forming and player launching

player.play() code

If authentication key and token are set, they will be included to stream URL

```
function playBtnClick() {
    if (validateForm()) {
        ...
        var videoSrc = $("#urlServer").val() + '/' + streamName + '/' + streamName + '.m3u8';
        var key = $('#key').val();
        var token = $("#token").val();
        if (key.length > 0 && token.length > 0) {
            videoSrc += "?" + key + "=" + token;
        }
        player.src({
            src: videoSrc,
            type: "application/vnd.apple.mpegurl"
        });
        console.log("Play with VideoJs");
        player.play();
        onStarted();
    }
}
```

5. Playback stopping

player.dispose() code

This method removes the div container tag where player was initialized from the page

```
function stopBtnClick() {
    if (player != null) {
        console.log("Stop VideoJS player");
        player.pause();
    }
    onStopped();
}
```

6. New div contaioner tag creation after previous player was removed

code

```
function createRemoteVideo(parent) {
    remoteVideo = document.createElement("video");
    remoteVideo.id = "remoteVideo";
    remoteVideo.width=852;
    remoteVideo.height=480;
    remoteVideo.controls="controls";
    remoteVideo.autoplay="autoplay";
    remoteVideo.type="application/vnd.apple.mpegurl";
    remoteVideo.className = "video-js vjs-default-skin";
    parent.appendChild(remoteVideo);
    player = videojs(remoteVideo);
}
```