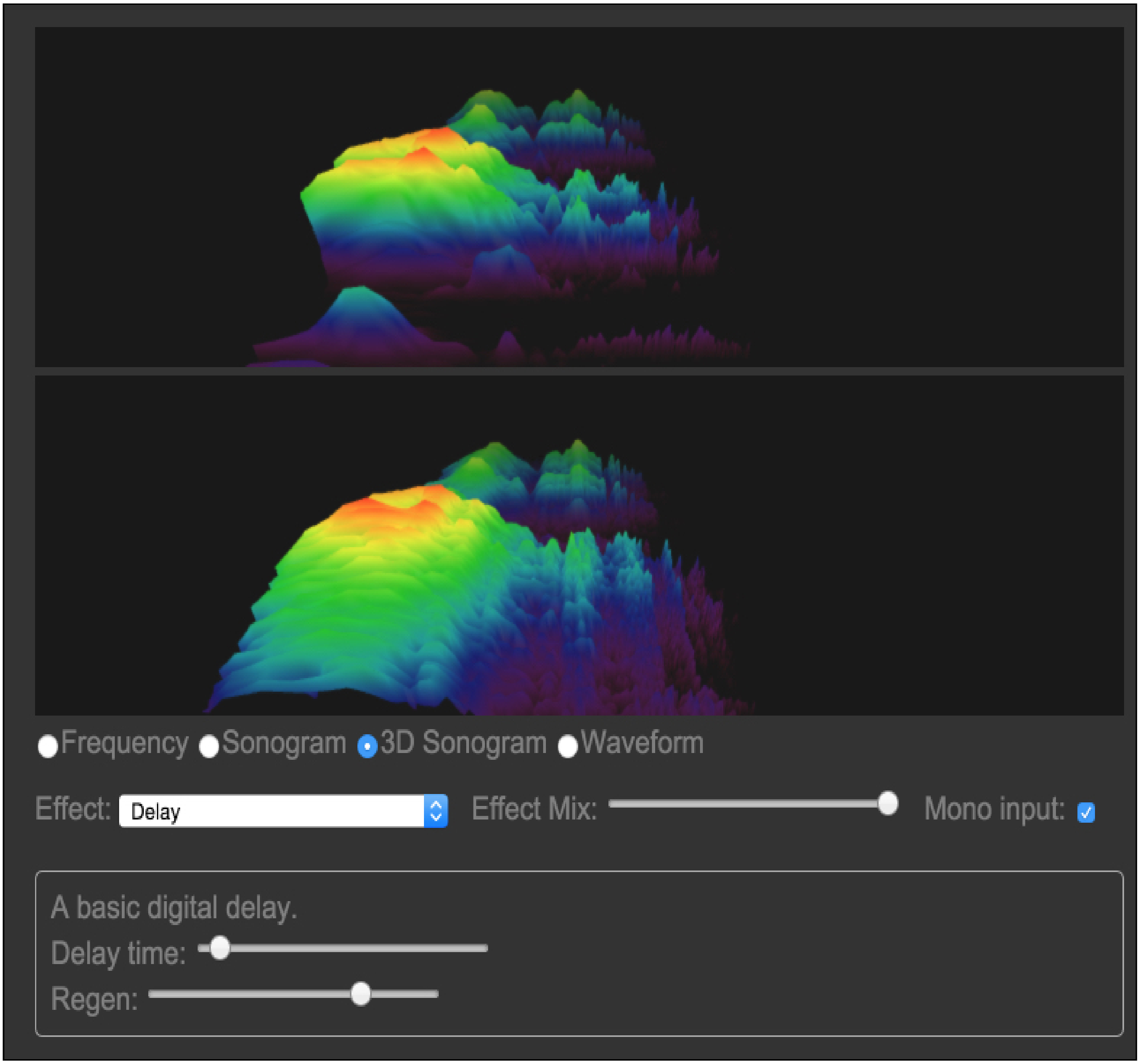
Working with the microphone

Instead of using the getUserMedia API with: `navigator.getUserMedia({video:true}, onSuccess, onError)`, it is also possible to use`{audio:true}` for the first parameter. In this case, only the microphone input will be captured. Notice that `{video:true, audio:true}` is also accepted, if you write a video conferencing system and need to capture both the audio and the video (this is often the case when writing WebRTC applications). The W3C WebRTC is another W3C specification, under development, for P2P audio/video/data Real Time Communication.

Apart from videoconferencing, microphone input will be used for music Web apps, from the WebAudio API. This API focuses on real time sound processing and music synthesis. This API will be covered in the advanced W3Cx HTML5 course.

Do try some nice WebRTC applications like Appear.in audio and video conferencing tool. Also check out the WebAudio demonstrations written by Chris Wilson, in particular the one called "Input effects".

Below is an example of real time audio processing of the microphone input using getUserMedia and WebAudio APIs



○Frequency ○Sonogram ●3D Sonogram ○Waveform

Effect: Delay

Effect Mix:

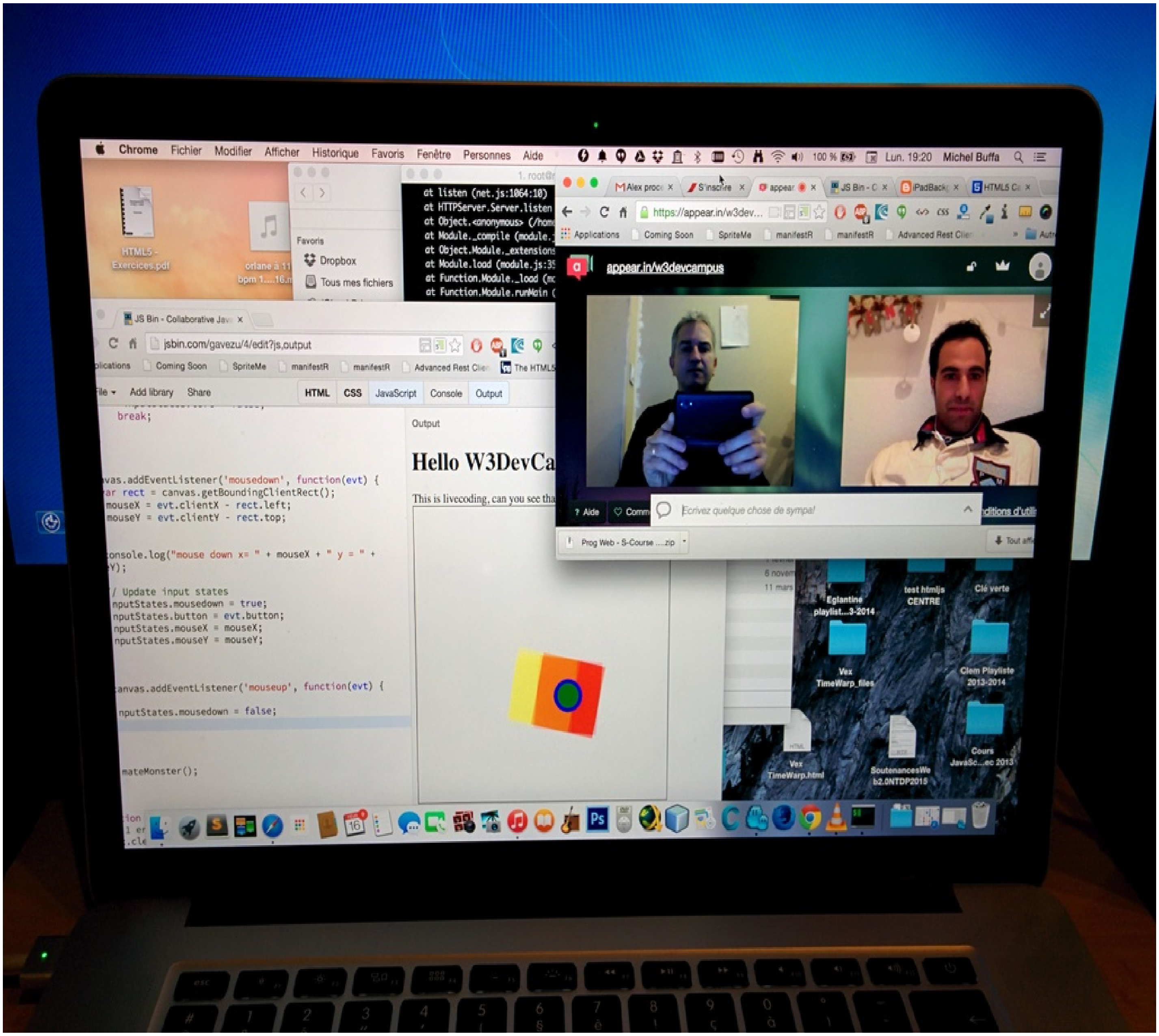
Mono input: ☑

A basic digital delay.

Delay time:

Regen:

appear.in: a free WebRTC video conferencing tool. It uses the getUserMedia API for video and audio.



```
at listen (net.js:1064:10)
at HTTPServer.Server.listen
at Object.<anonymous> (/hom
at Module._compile (module.
at Object.Module._extensions
at Module.load (module.js:35
at Function.Module._load (mo
at Function.Module.runMain (
```

```
break;


vas.addEventListener('mousedown', function(evt) {
ar rect = canvas.getBoundingClientRect();
mouseX = evt.clientX - rect.left;
mouseY = evt.clientY - rect.top;


onsole.log("mouse down x= " + mouseX + " y = " +
Y);

/ Update input states
nputStates.mousedown = true;
nputStates.button = evt.button;
nputStates.mouseX = mouseX;
nputStates.mouseY = mouseY;



anvas.addEventListener('mouseup', function(evt) {

nputStates.mousedown = false;




mateMonster();
```

Output

Hello W3DevCa

This is livecoding, can you see tha