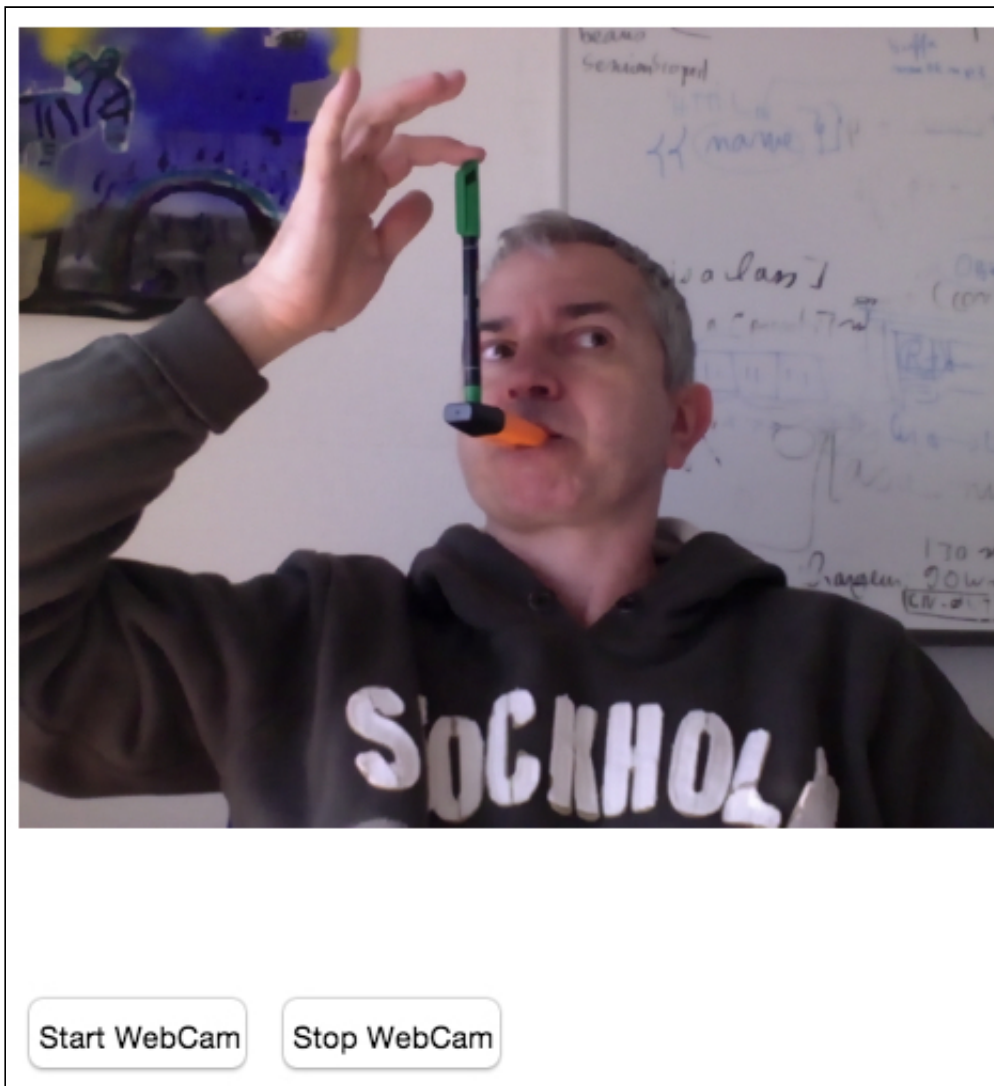


More on getUserMedia

Let's see some more examples of what we can do with the getUserMedia API: start/stop the webcam, take a screenshot from the current video stream from the webcam, and apply CSS effects in real time. And at the end, we give links to some cool examples available on the Web.

HOW TO STOP/RELEASE THE WEBCAM



[Online version at JS Bin](#)

In order to stop the webcam and make the hardware "unlock it", you need to call

the `stop()` method of the video stream.

Modified version of the previous example:

```
<html>
<head>
  <meta charset="utf-8">
  <title>JS Bin</title>
  <script>
    navigator.getUserMedia = ( navigator.getUserMedia ||
                                navigator.webkitGetUserMedia ||
                                navigator.mozGetUserMedia ||
                                navigator.msGetUserMedia);
    var webcamStream;

12.  function startWebCam() {
      if (navigator.getUserMedia) {
        navigator.getUserMedia (
          // constraints
          {
            video: true,
            audio: false
          },

22.  // successCallback
      function(localMediaStream) {
        var video = document.querySelector('video');
        video.src = window.URL.createObjectURL(localMediaStream);
        webcamStream = localMediaStream;
      },

      // errorCallback
      function(err) {
        console.log("The following error occurred: " + err);
      }

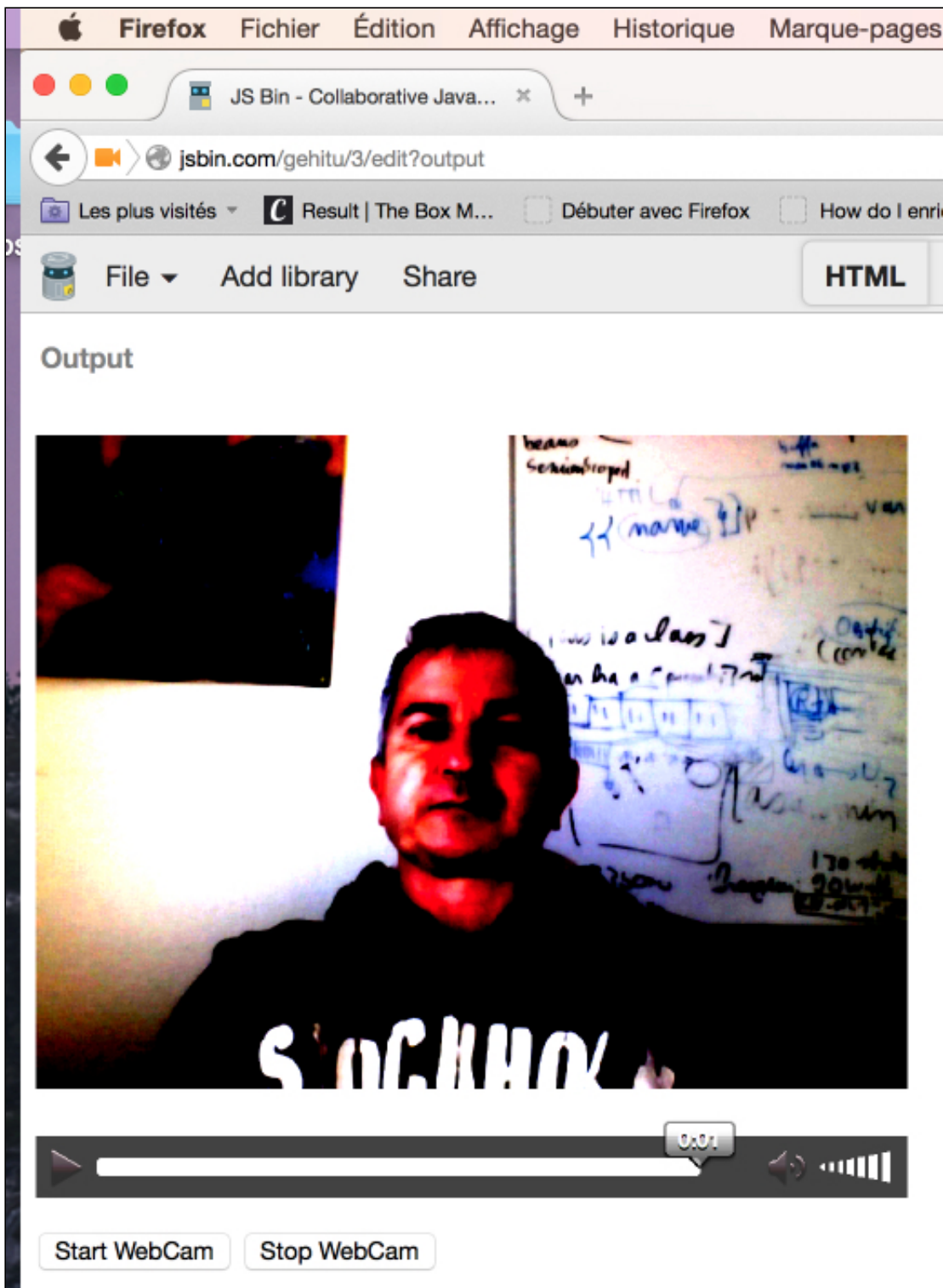
32. );
```

```
    } else {  
      console.log("getUserMedia not supported");  
    }  
  }  
  function stopWebcam() {  
    webcamStream.stop();  
  }  
</script>  
</head>  
<body >  
  <video width=200 height=200 id="video" controls autoplay></video>  
  <button onclick="startWebcam();">Start Webcam</button>  
  <button onclick="stopWebcam();">Stop Webcam</button>  
</body>  
</html>
```

OTHER EXAMPLES THAT MIX IN WHAT WE'VE SEEN IN PREVIOUS CHAPTERS, BUT THIS TIME WITH A LIVE VIDEO STREAM

Applying CSS effects on a video element with a live webcam

Try this example that shows how to use the getUserMedia API - note the CSS effects (click on the video to cycle from one effect to another). Works in Chrome/Firefox/Opera: [Online version at JS Bin](#)



Taking a snapshot from the live webcam stream

The trick is to copy and paste the current image from the video stream into a `<canvas>` element:

[Online version at JS Bin](#)

We will get into more details about this example next week when we will look at

the<canvas> element. For the time being, just play with the example.

Firefox
Fichier
Édition
Affichage
Historique
Marque-pages
Outils
Fenêtre
Aide

JS Bin
JS Bin - Collaborative Java...

jsbin.com/gehitu/6/edit?js,output
Rechercher

Les plus visités
Result | The Box M...
Débuter avec Firefox
How do I enrich m...
New Super Mario ...
Re: RESTful API fo...

File
Add library
Share
HTML
CSS
JavaScript
Console
Output
Account
Blog

```

    audio: false
  },

  // successCallback
  function(localMediaStream) {
    video = document.querySelector('video');
    video.src =
window.URL.createObjectURL(localMediaStream);
    webcamStream = localMediaStream;
  },

  // errorCallback
  function(err) {
    console.log("The following error occurred: " +
err);
  }
};
} else {
  console.log("getUserMedia not supported");
}
}

function stopWebcam() {
  webcamStream.stop();
}

//-----
// TAKE A SNAPSHOT CODE
//-----
var canvas, ctx;

function init() {
  // Get the canvas and obtain a context for
  // drawing in it
  canvas = document.getElementById("myCanvas");
  ctx = canvas.getContext('2d');
}

function snapshot() {
  // Draws current image from the video element into
  the canvas
  ctx.drawImage(video, 0,0, canvas.width,
canvas.height);
}

```

Output
Run with JS
Auto-run JS

Take a snapshot of the current video stream

See CSS and JavaScript tabs. Click on the Start WebCam button.

Start WebCam
Stop WebCam
Take Snapshot

Screenshots :

Bin
just

IMPRESSIVE DEMONSTRATIONS AVAILABLE ON THE WEB

- Exploding video: look at yourself and click on the video to make it explode! [Working demonstration](#) and [explanations here](#).
- [Live CSS filter Photo booth](#)
- A MUST TRY: [Paul Neave's WebGL Camera Effects](#)
- [Paint in real time with webcam output](#), choose webcam in the right menu, then click and drag rectangles. Share the URL and you have a multi-participant paint program with webcam painting! By Michel Buffa.