**Final presentation – WebRTC**

**Opening**

How many of you have used Skype or some other video chat application before? Ok thanks, that’s pretty much all of you. Today, I’m going to show you a different type of technology for video chat, and it’s called WebRTC.

**WebRTC Introduction**

WebRTC stands for Web Real-Time Communication. It enables you to communicate with others via voice calling, video chat or even P2P file sharing – directly through a web browser.

Now, you might be asking yourself: „That sounds great, but I already have Skype for that, and it works. Why would I need anything else?“ Well, then you’re absolutely right, because Skype and many other applications do excactly that.

But as you will see, WebRTC has a few really amazing unique features.

**WebRTC call setup**

Let’s take a look at what happens when you communicate with someone over a traditional chat application. We have two people, and one person calls the other, who accepts the call. Now they are connected, and all data traffic goes over the web server.

And this is where WebRTC really stands out. With WebRTC, you only need a web server to set a connection up. Afterwards, all data goes directly from one user to another, without a server involved. This is called peer-to-peer.

**WebRTC + / -**

This offers many advantages: There is significantly less network latency, because peers are connected directly, over the shortest possible network path. But what I think the most amazing feature of WebRTC is that it can be used directly from a web browser, without any plugins necessary. What this means is that with one HTML page and some JavaScript knowledge, all of you could start developing your own video chat application today. All you need is to call the right API functions, and you’re good to go.

Of course, WebRTC is not a perfect technology, there are some downsides to it: It is still in development, which means that some API functions might change in the future. And the probably biggest disadvantage is browser compatibility. WebRTC works great in Google Chrome and Firefox, pretty ok in Opera, a little bit in Microsoft Edge and not at all in Safari. Unfortunately, the folks at Apple are currently not interested in WebRTC.

**Prototype**

What I want to show is that WebRTC can be used for really helpful applications and that’s why I built a remote support application. For those of you who are not familiar with this concept, I’m going to briefly show it works.

A common problem in large factories is that machines are eventually going to fail. And as you all of know, if there’s no production, there’s nothing to sell, which is why it is so important that repairs can be conducted as quickly as possible. But the problem is, the workers might not know how to fix the problem, because the machine was built by someone else, probably another company. Now, in the past this meant that the workers either had flip through a manual with several hundred pages, or an expert had to come look at the problem on site, although the problem might just be really small and could be solved very quickly. This is where remote support applications can really help: The worker can call an expert directly over, say, a tablet. The expert can be located thousands of miles away and he can still assist the worker as if he were standing right next to him and guide him towards the solution.

**Demo time**

And how he can do that I’m going to show you myself now, because it’s demo time. What you can see here is the web page of the application I developed. It’s running on a simple node.js server that handles http rquests and WebSocket connections.

Now we can pause the video and help him by drawing some things on the video

**Closing**

Just for a quick recap, WebRTC can be used for free through web browsers and anyone can develop telecommunication applications for it. You can try it yourselves, there’s some great demos on the internet, and I promise you you’ll first your first own application running within one or two hours.

Now, I want to thank you for your attention, if you are interested in this topic, all the source code is publicly available on my Github account and you can come talk to me about it anytime you want and, of course, and now I’m more than happy to answer all your questions. Thank you.