



HTML5-based Video Conferencing For Synchronous Groupware Applications

DOI:

[10.1109/CSCWD.2017.8066671](https://doi.org/10.1109/CSCWD.2017.8066671)

Document Version

Accepted author manuscript

[Link to publication record in Manchester Research Explorer](#)

Citation for published version (APA):

Hussain, A., Wang, W., & Xu, D. (2017). HTML5-based Video Conferencing For Synchronous Groupware Applications. In *2017 IEEE 21st International Conference on Computer Supported Cooperative Work in Design* <https://doi.org/10.1109/CSCWD.2017.8066671>

Published in:

2017 IEEE 21st International Conference on Computer Supported Cooperative Work in Design

Citing this paper

Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights

Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy

If you believe that this document breaches copyright please refer to the University of Manchester's Takedown Procedures [<http://man.ac.uk/04Y6Bo>] or contact openresearch@manchester.ac.uk providing relevant details, so we can investigate your claim.



Comparing WebRTC Video Conferencing with Skype In Synchronous Groupware Applications

Asif Hussain, Weigang Wang, Dong-Ling Xu

Alliance Manchester Business School

The University of Manchester

Manchester, United Kingdom

asif.hussain@alumni.manchester.ac.uk, weigang.wang@mbs.ac.uk, ling.xu@mbs.ac.uk

Abstract — A study was carried out to compare an integrated HTML5-based real-time video conferencing solution against a separately installed video conferencing solution in the same web-based meeting support application. To carry out this comparison, WebRTC and Skype have been integrated into an existing groupware application. Our study shows that comparing to a separately installed, best of breed video conferencing solution; the web browser-based solution would be an adequate and more convenient one. Our work indicates that the cost to implement WebRTC is not large in comparison to the number of benefits that can be gained. The quality of video conferencing has been continuously improving, and the implementation cost is getting lower as better WebRTC libraries emerge and evolve.

Keywords—Groupware; HTML5; WebRTC; Real-time; Social Computing; P2P; Video conferencing; Synchronous;

I. INTRODUCTION

Web 2.0 [2] is enabling web technologies such as AJAX, Java, and Flash to allow the creation of rich web-based groupware applications [1]. However, many of such applications have relied on third party plug-ins or externally installed solutions to provide A/V communication support.

HTML5 looks to address these problems by re-standardising the foundation that all web applications are built on, that is HTML. A number of changes have been introduced to make it easy to handle multimedia and graphical content on the web without having to resort to proprietary plug-ins and API's [3].

Over the past three years, HTML5 has increased in prominence across the Web. A limited number of groupware applications have already implemented some features of HTML5, namely for its networking and cross-platform capabilities [4]. There is however still a lack of browser-based applications that have realised the full benefits of HTML5. Features such as embedded audio/video support, local storage, and improved networking capabilities would appear to be immediately beneficial to groupware applications [5]. However, it remains to be seen whether groupware application developers will consider these significant enough to change their existing applications or develop new ones. One area of particular interest is HTML5-based video conferencing, referred to as WebRTC.

This paper looks to address the question of whether WebRTC can provide a viable alternative to plug-in/external based approaches for video conferencing in a groupware application. The approach taken was to improve upon an existing browser-based groupware application (PowerMeeting) using WebRTC.

PowerMeeting is a synchronous groupware application developed by Dr Weigang Wang at the Manchester Business School [6]. It has several groupware tools such as brainstorming, multi-criteria decision analysis and a shared whiteboard. PowerMeeting was already utilising Web 2.0 in many aspects, mainly with its implementation of AJAX that allowed workspaces to be updated dynamically without requiring a full page refresh. There was no implementation of HTML5 in PowerMeeting previously, and therefore it became a good candidate to improve using the new technology.

For this study, we compared WebRTC video conferencing functionality against a Skype-based solution to determine whether an integrated approach could provide an experience as good as or better than the off the shelf solution on certain aspects.

The rest of the paper is organised as follows: Section II discusses related work. Section III presents our integration methods. The comparison case studies have been provided in section IV. While the results and discussions are in Section V and VI respectively. The paper ends in section VII with conclusions and future work.

II. RELATED WORK

As HTML5 has introduced several new features to the way we view web pages, the current work in this area has been limited to a select number of features. Arguably the greatest benefit HTML5 brings to groupware is the improvement to networking capabilities with the introduction of WebSockets [5, 7, 8, 9].

In the study towards cross-platform collaboration, Wenzel et al. redeveloped a Java-based groupware application, Tele-Board, by introducing HTML5 rendering capabilities using the canvas API as well as the networking approach of WebSockets [10]. The study showed that HTML5's canvas element performed better than SVG-based rendering, in particular when there was a large amount of data and high level of interaction required on screen. The study also concluded that HTML5 web workers had an

advantage over traditional simulated multi-threaded approaches for applications that required high computational power.

Katayama et al. focused on using HTML5 to increase synchronisation times in their study [11]. The experiment looked at how the HTML5 Canvas API can be used to annotate existing PDF documents and the amount of synchronisation delay that occurs between users when annotations are being made. The existing JavaScript application was re-designed by implementing the HTML5 Canvas API into a new collaborative web application used to annotate PDF documents in real-time. The application also implemented HTML5 storage capabilities for client side storage of PDF documents as well as WebSockets for networking capabilities. The study found that the synchronisation delay increased depending on the number of annotations in the pdf documents. This was not considered a major issue as the average number of annotations made to a document would not cause a significant delay in rendering. The application therefore, demonstrated several advantages of HTML5 in a synchronous groupware application as well as concluding that HTML5 Canvas API is well suited for the on-screen synchronisation requirements of a collaborative groupware application.

In the paper, we examine the real-time audio and video capabilities offered by HTML5 that are now available in a project known as WebRTC. We enhanced a browser-based groupware application by implementing WebRTC and compared this with a Skype-based approach. Unlike many other video conferencing solutions that focus on the things happening in the video, the video conferencing in PowerMeeting has been mainly designed to provide fine-grained coordination support on joint activities in a shared workspace. Specifically, video conferencing in PowerMeeting is mainly used for describing or referring to specific shared visual artefacts in the shared workspace, providing grounding clues or explanations to their manipulating actions, correcting a misunderstanding, or notifying participants to move to the next step in a group process by the session chair.

III. INTEGRATION METHODS

PowerMeeting is a web-based synchronous groupware [6]. It allows users to work together in a shared workspace using tools that include text messaging, brainstorming, voting, a multi-criteria decision analysis tool, and a whiteboard. It is a real-time groupware application developed using Google Web Toolkit (GWT). Its server side is developed using AJAX technology (GWT RPC) in Java; While its client side is built using GWT, which allows the client side code developed in Java and then translated into HTML and CSS files. This makes it a browser-based groupware, accessible using a URL as with any AJAX-based web app, without a need to download and install [12].

The application already included a Skype integration which allowed Skype conferences to be called in a groupware session. The Skype integration is based its URI API; whose syntax is `Skype:participant1[;participant2;...participant9]?call&video=true`. The specific URL in a groupware is composed dynamically using information collected at user registration. The invocation of the

URL is attached to a conference call button, to be called by the session chair of a groupware session. In order to remove the reliance on Skype, we decided to implement WebRTC into the application, an HTML5 area expected to see wide adoption [13].

WebRTC is a set of APIs drafted by the World Wide Web consortium that provides real-time communication abilities such as voice calling, video chat and P2P file sharing to web browsers and mobile applications [14]. The feature of WebRTC that we were interested in examining was video conferencing. A number of libraries already existed that wrapped the WebRTC API into a more simplistic API. One of these libraries is SimpleWebRTC.

SimpleWebRTC is a WebRTC library that enables developers to utilise the video and audio chat of WebRTC with a few lines of HTML and JavaScript [15]. We implemented SimpleWebRTC into PowerMeeting to allow users to start a video chat during a collaborative working session. The session chair (facilitator) can initiate the video chat during a session, while participants can join and leave the video chat as they wish.

The implementation required adding a video and div element into the HTML. The video element would playback the output from the user's own webcam whilst the div element would hold the video elements of each participant in the conference. A few lines of JavaScript were required to initiate a WebRTC object and reference the video elements. The video conference could then be started by calling the `joinRoom` function of the SimpleWebRTC library. The `joinRoom` function took a room name as a parameter. For simplicity in our implementation, we passed the PowerMeeting session name as the room name.

As the main purpose of video conferencing in PowerMeeting is to facilitate the group activities in a shared workspace, the HTML video and div elements were embedded in a dialog box. The dialog box would appear when the user joined in the video conference (i.e. when the Join button is clicked, see Fig. 1). The dialog box could then be moved around, so as to avoid covering up the visual artefacts in the shared workspace. To avoid disrupting the focus in the shared workspace, we intentionally displayed the video element small by default, although it could be enlarged by toggling the Small/Large button.

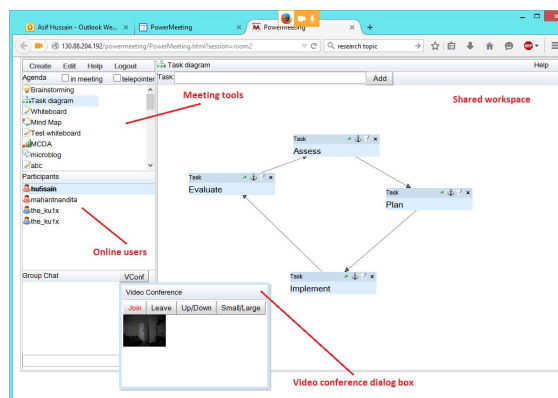


Fig. 1. PowerMeeting with WebRTC video conferencing

As GWT is largely based on web applications that are implemented in Java. The JavaScript implementation of SimpleWebRTC into PowerMeeting required making use of GWT's JavaScript Notation Interface (JSNI). The total number of man hours to implement the new features into PowerMeeting was approximately 35.

IV. EXPERIMENT

With the HTML5-based enhancements to PowerMeeting, we were able to design an experiment to gain user feedback on the enhancements. The experiment would allow us to determine whether integrated video conferencing in the browser can offer experience as good as or better than an externally installed one. To make this comparison, we set up an experiment for participants to carry out a set of predefined group activities using the PowerMeeting web application. Participants were asked to complete the following activities in PowerMeeting:

- 1) Draw a simple diagram using PowerMeeting whiteboard.
- 2) Carry out a brainstorming session for a chosen topic

This was a multiple step activity consisting of brainstorming, categorisation, voting, and reporting. A session chair would lead the process, by using video conferencing to coordinate the group activities, e.g. to ask if all the participants had completed their actions in each step and then forward the group to the next step until all the steps had been completed.

The study utilised convenience sampling by pre-selecting 22 participants from a number of IT organisations with an even split between male and female. All participants had previous experience of using Skype for personal means as well as some form of groupware experience within their professions. However, none of the participants had previously worked together or had any previous experience of PowerMeeting, and no prior training was given.

Participants worked in groups of 3 to 6 people with one of the participants per group acting as the facilitator (i.e. session chair) in order to co-ordinate the group activity. Each group was given a specific time and date with details of the session to join.

When all participants within a particular group had joined the session at their allotted time, each participant would be required to carry out the two tasks using one of the following methods:

- PowerMeeting with Skype integration
- PowerMeeting with WebRTC implementation

We used an A/B test approach to direct some groups of participants to the Skype video conferencing version while others would see the WebRTC implementation. This was achieved by creating specific session names for the two groups. Session names that had an odd number of characters in the name were presented with the WebRTC incorporated version while session names that had an even number were presented with Skype conferencing. All the participants are asked to use Google

Chrome web browser to access the PowerMeeting system, as this browser provides the best support for WebRTC at the moment.

For the first task, the facilitator would be given a simple diagram that should take no longer than 3 minutes to draw on the whiteboard i.e. an outline of a house. Participants would then be given instructions on what to draw by the facilitator using the chosen video chat method. The facilitator was required to describe the diagram to the team members and ensure that all team members were synchronised before giving the next instruction. Once the diagram had been completed it would be shared amongst the team before moving on to the next task.

The second task required team members to carry out a decision-making process on a topic chosen by the facilitator (in this case, 'what are ways to make the planet greener?'). The facilitator would initiate the brainstorming tool within PowerMeeting and set the topic. The team members would then input their ideas and once the facilitator felt enough ideas had been generated would proceed to categorisation. Ideas would then be consolidated and categorised, with similar ideas being placed into a single category. With all ideas placed into categories, the voting screen would be displayed, and each team member voted on the best ideas. Once all votes had been submitted, the facilitator proceeded to the voting results presented as a ranked list in order to prompt further discussion from the team and then conclude the session.

A total of six groups participated in the experiment. One of those groups was also asked to use both the Skype and WebRTC versions of PowerMeeting. The group repeated each of the activities using both methods with a different set of diagrams and brainstorming question in each. The feedback provided by this group would enable us to see a direct comparison between the WebRTC video conferencing and existing Skype based solution. A separate questionnaire was designed for this particular group. In the interests of time, not all groups were asked to complete the tasks using both video conferencing methods. Each session lasted between 15 to 20 minutes.

All participants were asked to complete a post-experiment survey that consisted of 20 questions made up of both qualitative and quantitative questions on usability and the communication and coordination role of the video conference in a groupware session. The questionnaire was designed to determine whether WebRTC could provide a collaborative environment as good as a Skype but also to establish whether the facilitation and coordination aspects of a groupware session could be improved with the addition of a WebRTC video conferencing feature.

V. RESULTS

The results showed that most people agreed that the HTML5-based WebRTC implementation for real-time communication provided a good method for collaboration with team members and helped them to complete groupware activities. Ease of use and simplicity was also agreed upon by participants for the WebRTC solution. The results, however, were also very similar

for the Skype solution, with no clear winner. Participants agreed that their preference would be to use a fully integrated solution for real-time communication to avoid having to install additional software on their machine.

The results below show a number of different areas where answers are slightly more favourable towards the WebRTC solution. (See Tables I-II).

TABLE I.

The video conferencing facility made it easier for you to provide guidance to the participants		
<i>Response</i>	<i>WebRTC</i>	<i>Skype</i>
Strongly disagree	0%	0%
Disagree	0%	9.1%
Neither agree nor disagree	10%	27.3%
Agree	60%	36.4%
Strongly agree	30%	27.3%

TABLE II.

The use of video conferencing helped to facilitate the session more effectively		
<i>Response</i>	<i>WebRTC</i>	<i>Skype</i>
Strongly disagree	0%	10%
Disagree	0%	0%
Neither agree nor disagree	0%	20%
Agree	80%	60%
Strongly agree	20%	10%

Although participants did agree that they would prefer a non-plug-in based solution, there was a consensus that the Skype solution allowed the group to work more efficiently overall and collaboration with group members was easier (See Tables III-IV). Participants reported more stability issues with the WebRTC implementation. The video quality of the WebRTC solution was also reported as not being as good as the Skype offering (See Fig 2 and 3).

TABLE III.

The video conferencing facility allowed the group to work more efficiently		
<i>Response</i>	<i>WebRTC</i>	<i>Skype</i>
Strongly disagree	0%	0%
Disagree	0%	0%
Neither agree nor disagree	9.1%	0%
Agree	81.8%	63.6%
Strongly agree	9.1%	36.4%

TABLE IV.

The video conferencing facility made it easier for you to collaborate with group members		
<i>Response</i>	<i>WebRTC</i>	<i>Skype</i>
Strongly disagree	0%	0%
Disagree	10%	0%
Neither agree nor disagree	0%	0%
Agree	60%	63.6%
Strongly agree	30%	36.4%

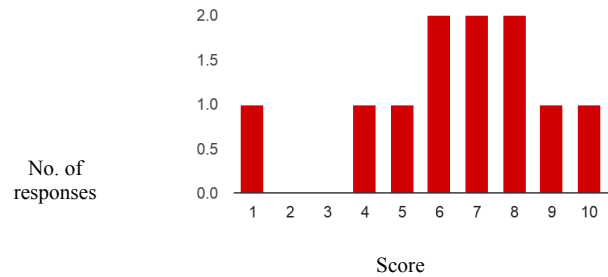


Fig. 2. WebRTC - How would you rate the video quality of video conferencing in PowerMeeting?

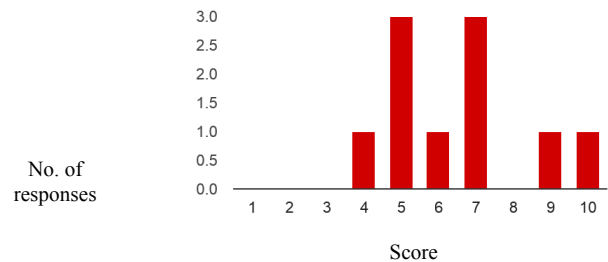


Fig. 3. Skype - How would you rate the video quality of video conferencing in PowerMeeting?

Regarding the stability of the integrated WebRTC version of PowerMeeting, below are some of the feedback comments received from the participants regarding issues that participants faced when using the WebRTC implementation:

- "My camera was frozen after leaving the chat and re-joining."
- "Froze once so had to restart the application."
- "The video feedback screens were too small."
- "Video would freeze frequently."

Other results from the survey revealed that for participants of the WebRTC solution there was agreement (81.8%) on the importance of not requiring external plug-ins in order to start video conferencing. The same question was not answered as

convincingly by participants that used Skype with just over 54% of participants in agreement on the importance of a non-plug-in based approach.

For the group that carried out the experiment in both the WebRTC implementation and Skype, the results were general in favour of the WebRTC solution.

All participants stated that the WebRTC solution was:

- Easier to set-up
- Made it easier to complete the groupware activities
- Made it easier to facilitate the session

The majority of participants in that particular group also stated the following in favour of the WebRTC implementation:

- Provided a good method of communicating with other group members
- Provided a convenient integrated support and good experience

However, the majority of participants felt that Skype provided better video and audio quality and the overall experience of Skype was rated slightly higher than that of the WebRTC implementation.

VI. DISCUSSION

To summarise the results of the survey, participants liked the idea of not having to rely on proprietary external solutions in order to be able to utilise real-time communication features of the web. Participants also appeared to suggest that the WebRTC implementation allowed better facilitation of the groupware session.

However, in the current early state of WebRTC, participants still felt that the external application based Skype method offered an overall more stable experience than the WebRTC solution with better video quality and allowing group members to communicate more effectively. The results were overall slightly in favour of the Skype version.

Part of the reason why Skype was seen as the overall more stable experience for real-time communication is due to Skype's current dominance in the video conferencing market and the number of years in which it has constantly been refined. Skype is also well maintained by the development team at Microsoft. Most participants were also already familiar with Skype, and this would have been expected considering it has over 100 million active users per month [16].

The other factor in why participants rated the Skype solution well was due to the relative infancy of WebRTC itself. WebRTC was first introduced by Google in May 2011, and there has since been constant progress by the open source community to refine it. However, WebRTC is still yet in working draft state by the W3C and therefore cannot yet be considered a finished product [14].

As work on WebRTC continues to progress, we expect the number of WebRTC libraries to increase in number. This will lead to libraries that offer a richer and better user experience of WebRTC than those currently available. Browser support for WebRTC is also likely to increase. Wider Web browser support is very important for our groupware applications, as the users are distributed around the world using various web browsers.

Apart from being a plug-in or installation free alternative to Skype, the WebRTC implementation also holds the advantage of being integrated into the application rather than being shown in an external window. Therefore, group activities can be conducted while the video conferencing element remains in the corner of the application. The video size is also adjustable in order to allow the user to control whether the window is small enough to minimise any distraction from the groupware activities.

One of the issues participants faced when using the WebRTC solution was with the quality of connection. Video chat sessions using WebRTC would occasionally freeze and sometimes require restarting. The cause of the issues was most likely due to the choice of servers we used in order to deploy WebRTC. WebRTC requires three types of servers [17]:

- 1) Signalling – to allow clients to exchange metadata in order to coordinate communication
- 2) STUN (Session Traversal Utilities for NAT) – to allow each client to discover its public IP address
- TURN (Traversal Using Relays around NAT) – to relay traffic when a direct connection fails

For our implementation of WebRTC, we chose to host our own signalling server by utilising SignalMaster [18]. However, for the STUN and TURN servers, we used Google's publically available STUN and TURN test servers. As the STUN and TURN servers from Google are for testing purposes only, and available to everyone to use, there was an expectation that our WebRTC solution's connectivity may not be as reliable as deploying our own servers. Our solution, therefore, suffered from connectivity issues reported by the participants.

As WebRTC matures there will inevitably be more support available for implementing it in the form of richer libraries and better documentation. The quality and speed of WebRTC will also be an area for improvement with a greater range of STUN and TURN server providers to emerge. This will enable WebRTC to compete better with plug-in based solution as well as desktop applications. The purpose of this study was to determine how well the upcoming web-based WebRTC technology can provide a viable alternative to current solutions that are considered best of breed, in this case Skype. Whilst we found results to be slightly more favourable towards Skype, this by no means suggests that WebRTC is destined to fail. For developers looking to implement WebRTC, it may be worth waiting for further developments in the area before looking to deploy a WebRTC solution for general use. In the interim period, however, we would recommend the SimpleWebRTC library due to its simple API and well documented usage guide.

VII. CONCLUSIONS AND FUTURE WORK

HTML5 has enabled well-known companies to bring significant improvements to their widely used web applications. Facebook, YouTube and Twitter are only a handful that have utilised HTML5 to improve the end user's experience [19].

One of the less prominent types of applications on the web today are real-time groupware applications. In order to support the real-time communication requirements, groupware applications have relied on external plug-ins in order to deliver real-time communication as well as other rich features.

Our study examined how HTML5-based WebRTC can offer a user experience as good as external or plug-in based approaches and whether the advantage of not requiring plug-ins may encourage the adoption of groupware applications. In particular, we wanted to perform a comparison between an integrated web-based approach against current best of breed video conferencing in terms of which can best support group decision support systems. We achieved this by implementing WebRTC into an existing groupware application: PowerMeeting, and compared this with PowerMeeting's existing Skype-based solution.

We found that whilst users felt that WebRTC was capable of delivering a solution that could be used without any major issues, the quality and reliability of the Skype solution provided a more stable experience of groupware activities overall. With that said the results were only slightly in favour of our Skype-based implementation and the issues reported were mainly due to our implementation not running on production servers. Our study, therefore, shows that contrary to current widely used, off the shelf solutions being seen as best of breed for video conferencing in group decision support system and online meeting applications, an integrated approach would be an adequate and more convenient one. Ultimately, developers can now begin to consider WebRTC to deliver good working alternatives to plug-in based approaches, however, due to WebRTC still being in Working Draft state by W3C, the features and quality of the solution may not yet be as good as the more established plug-in based solutions.

The study suggests that there are three main directions for further work. Firstly, as we mentioned, our implementation of WebRTC utilised our own signalling server but used Google's test STUN and TURN servers. This may have been the cause to the connectivity issues participants faced and therefore to fully realise the benefits of WebRTC, the solution could be improved by deploying and utilising our own servers.

Secondly, as WebRTC is yet in its early stages of being utilised on the web, there was an expectation of the technology to not be as fully functional as a final product. It would therefore be interesting to see how WebRTC compares when it has been refined and achieved Web Standard status by W3C. A fairer comparison could then be made between WebRTC and the more established current plug-in based solutions.

Finally, it would be beneficial to know whether integrated web-based video conferencing can provide a significantly good coordination mechanism for facilitators and participants when collaborating together. This is an area where we have already planned some further work to explore such coordination support.

REFERENCES

- [1] Ellis, C.A., S.J. Gibbs, and G. Rein, Groupware: some issues and experiences. *Communications of the ACM*, 1991. 34(1): p. 39-58.
- [2] Web 2.0 - Wikipedia. 2015 29/01/2015]; Available from: https://en.wikipedia.org/wiki/Web_2.0.
- [3] HTML5 - Wikipedia 2015 15/02/2015]; Available from: <https://en.wikipedia.org/wiki/HTML5>.
- [4] Marco, F.A., V.M. Penichet, and J.A. Gallud, Drag & Share: A Shared Workspace for Distributed Synchronous Collaboration, in *Distributed User Interfaces*. 2011, Springer. p. 125-132.
- [5] Pimentel, V. and B.G. Nickerson, Communicating and displaying real-time data with WebSocket. *Internet Computing, IEEE*, 2012. 16(4): p. 45-53.
- [6] PowerMeeting. 24/02/2015]; Available from: <https://sites.google.com/site/meetinginbrowsers/>.
- [7] WebSocket - Wikipedia. 05/02/2015]; Available from: <https://en.wikipedia.org/wiki/WebSocket>.
- [8] Marco, F.A., V.M. Penichet, and J.A. Gallud, Collaborative e-Learning through Drag & Share in Synchronous Shared Workspaces. *Journal of Universal Computer Science*, 2013. 19(7): p. 894-911.
- [9] Gutwin, C.A., M. Lippold, and T. Graham. Real-time groupware in the browser: testing the performance of web-based networking. in *Proceedings of the ACM 2011 conference on Computer supported cooperative work*. 2011. ACM.
- [10] Wenzel, M., et al., Towards Cross-Platform Collaboration-Transferring Real-Time Groupware To The Browser.
- [11] Katayama, S.-y., et al. A Fast Synchronization Mechanism for Collaborative Web Applications based on HTML5. in *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, 2013 14th ACIS International Conference on. 2013. IEEE.
- [12] Wang, W. Powermeeting: gwt-based synchronous groupware. in *Proceedings of the nineteenth ACM conference on Hypertext and hypermedia*. 2008. ACM.
- [13] Singh, V., A.A. Lozano, and J. Ott. Performance analysis of receive-side real-time congestion control for WebRTC. in *Packet Video Workshop (PV)*, 2013 20th International. 2013. IEEE.
- [14] W3C WebRTC 1.0 24/02/2015]; Available from: <http://www.w3.org/TR/webrtc/>.
- [15] SimpleWebRTC from &yet. 19/02/2015]; Available from: <http://simplewebrtc.com/>.
- [16] Skype - Wikipedia 25/02/2015]; Available from: <https://en.wikipedia.org/wiki/Skype>.
- [17] WebRTC in the real world: STUN, TURN and signalling - HTML5 Rocks 26/02/2015]; Available from: <http://www.html5rocks.com/en/tutorials/webrtc/infrastructure/>.
- [18] SignalMaster - GitHub 26/02/2015]; Available from: <https://github.com/andyet/signalmaster>.
- [19] YouTube Now Defaults to HTML5 Player Over Flash - The Next Web 27/02/2015]; Available from: <http://thenextweb.com/google/2015/01/27/youtube-will-now-default-html5-players-better-support-devices/>