**EUROPEAN UNIVERSITY OF LEFKE**

FACULTY OF ENGINEERING

Graduation Project I

Real Time Peer-to-peer, Conference Calling, Broadcasting and Media Processing

Peter Chidubem Atunalu

144013

**Abstract.** A Real Time Peer-to-peer, video conference calling, broadcasting and media processing web application. This Project is about sharing your everyday. Watch your favorite shows with your friends, without being in the same room (or even the same city!). Collaborate with your coworkers when you’re all on the road. Shop together for a birthday present for Mom, then sing her “happy birthday” with family far away. The possibilities are endless.

**Supervisor**

Asst. Prof. Dr. Zafer Erenel

December 2017

# Introduction

## Problem definition

For once, we were all together, celebrating life and love, then distance came in and set us apart form our family, friend and work. With so many social media applications like facebook, Instangram and Whatsapp most of the problem that araise from connect with those we love have been solved. But most of this application like Facebook can only connect you (using Video) with one user a given time, so have a family call with multiple family member is not possible.

Video conferencing is the technology that allows you to hold meetings with several correspondents who are located in different places while seeing them and talking to them in real time. It is different from simple video calling, which is normally one-to-one video communication (WebRTC is built for pear-to-pear communication between two users and has no built in support for conference calling).

Some time ago, expensive complex equipment and expertise is required for video calling or conferencing. Today, one can literally carry it in the pocket. You can participate in and host video conferencing sessions on your smartphone and mobile device as well as on your computer with basic hardware and adequate Internet connectivity.

Video conferencing has become more common and more accessible thanks to the advent and development of Voice Over IP (VOIP), which harnesses the underlying IP infrastructure of the Internet to make free communication possible. Packets of video data, along with packets of voice and other types of data, are carried on the Internet, thereby making voice and video communication free.

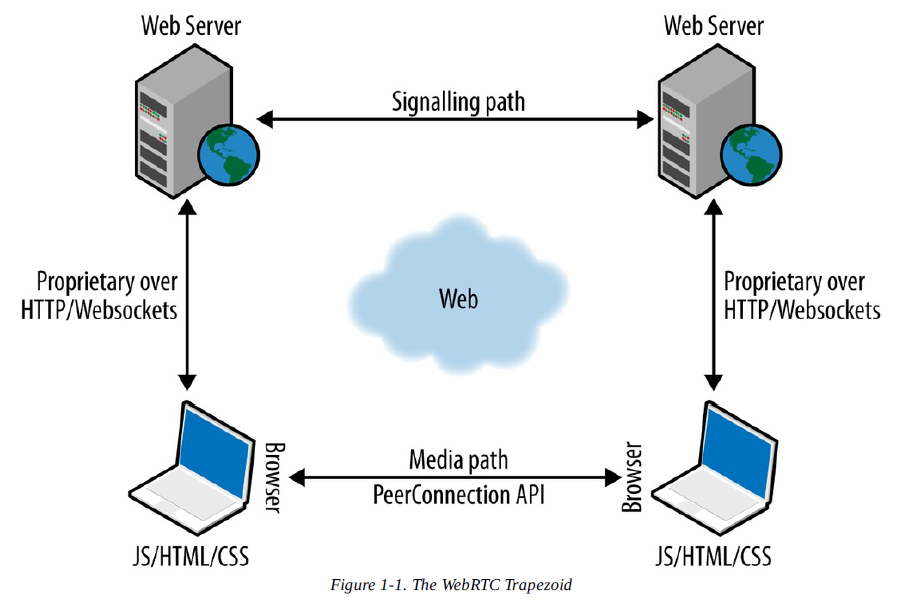
## Literature Surva

Web Real-Time Communication (WebRTC) is a new standard which allows browsers communicate in real time, using a peer-to-peer architecture. It is a secure, consent-based, audio/video (and data) peer-to-peer communication API between HTML5 browsers. WebRTC is an evolution in web applications world, as it enables, web developers to build real-time multimedia applications with no need for proprietary plug-ins for the very first time.WebRTC puts telecommunications and web development together two historically separated camps.

The World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF) are jointly defining the JavaScript APIs (Application Programming Interfaces), the standard HTML5 tags, and the underlying communication protocols for the setup and management of a reliable communication channel between any pair of next-generation web browsers. The standardization goal is to define a WebRTC API that enables a web application running on any device, through secure access to the input peripherals (such as webcams and microphones), to exchange real-time media and data with a remote party in a peer-to-peer fashion

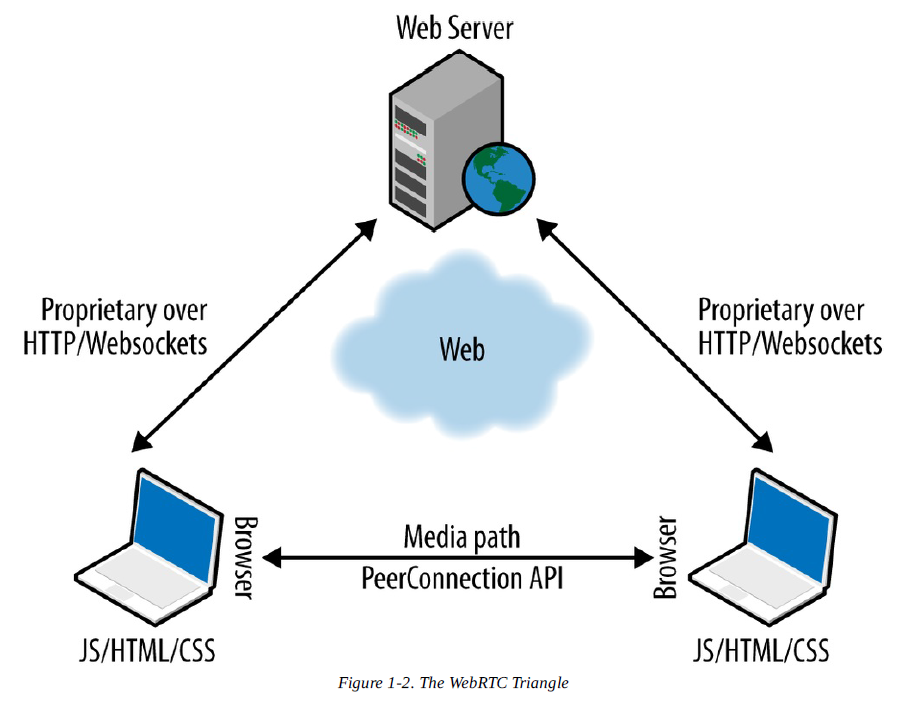
The classic web architecture semantics are based on a client-server paradigm, where browsers send an HTTP (Hypertext Transfer Protocol) request for content to the web server, which replies with a response containing the information requested. The resources provided by a server are closely associated with an entity known by a URI (Uniform Resource Identifier) or URL (Uniform Resource Locator). In the web application scenario, the server can embed some JavaScript code in the HTML page it sends back to the client. Such code can interact with browsers through standard JavaScript APIs and with users through the user interface

WebRTC extends the client-server semantics by introducing a peer-to-peer communication paradigm between browsers. The most general WebRTC architectural model draws its inspiration from SIP (Session Initiation Protocol) Trapezoid.



In the WebRTC Trapezoid model, both browsers are running a web application, that is been downloaded from different web servers. Signaling messages transported by the HTTP or WebSocket protocol via web servers that can modify, translate, or manage them as needed are used to set up and terminate communications.

The signaling between browser and server is not standardized in WebRTC, as it is considered to be part of the application (see Signaling). As for the data path, a PeerConnection allows media to flow directly between browsers without any intervening servers. The two web servers can communicate using a standard signaling protocol such as SIP or Jingle (XEP-0166). Otherwise, they can use a proprietary signaling protocol.



## Goals

The intangible benefits of video conferencing include more efficient meetings with the exchange of non-verbal communications and a stronger sense of community among business contacts, both within and between companies, as well as with customers. On a personal level, the face-to-face connection adds non-verbal communication to the exchange and allows participants to develop a stronger sense of familiarity with individuals they may never actually meet in person. till, when it came to live video broadcasting, the costs were at a level where only very well-funded organizations could afford to produce and distribute it. High production and distribution costs mean you’ll need a large audience to make ends meet. As a result, you need to produce content for the lowest common denominator; this is why you won’t see regular-season Little League Baseball being broadcast live — yet.

The introduction of mobile live video broadcasting was going to change that. If costs could be brought down to the expense of having a smartphone and an Internet connection, what would people broadcast and what would people choose to watch? Fast-forward seven years, with an increasing number of services offering easy-to-use live video broadcasting services; we see people sharing everything from cute puppies to coffee mugs, to — well — other things, like the Decorah Eagles bird’s nest live stream on Ustream that’s been viewed 325 million times. Or when viewers tuned in to watch “Twitch plays Pokémon” and spent more than a billion minutes in total watching the stream.

But the world also is witnessing an increasing number of live broadcasts of inconvenient truths, breaking news situations and angles that were never previously available. Even war is streamed live. Traditional media companies are running to keep up with an entire demographic that consists of individuals who are able to create their own news and share it globally, with the world as their audience.

Since the advent of consumer live streaming, broadcasters and publishers have had an interest in mining live video streaming platforms for content they could never have captured themselves. This is content created by people like you and me, shared around the world in real-time via new mobile consumer tech — not only in 140 characters, but as interactive video.

https://techcrunch.com/2015/09/25/live-broadcasting-is-too-important-to-be-left-to-the-broadcasters/ ***(half a page)***

# Resources

## Required software

* **NodeJs:** Node.js is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine). Node.js was developed by Ryan Dahl in 2009. Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.
* **MongoDB:** MongoDB is free and open-source distributed database at its core, so high availability, horizontal scaling, and geographic distribution are built in and easy to use
* **Angular 2+:** Angular is a platform that makes it easy to build applications with the web. Angular combines declarative templates, dependency injection, end to end tooling, and integrated best practices to solve development challenges. Angular empowers developers to build applications that live on the web, mobile, or the desktop
* **WebRTC:** WebRTC is a free, open project that provides browsers and mobile applications with Real-Time Communications (RTC) capabilities via simple APIs. The WebRTC components have been optimized to best serve this purpose.
* **Socoket.io:** Socket.io is a Javascript networking library that runs server-side on Node.js and in the browser. It abstracts away websockets and other communication schemes, depending upon browser capabilities. It also includes convenient features such as broadcasts and multicasts, which are beyond the features of plain websockets.

## Others

* **Amazon EC2:** Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.
* **Docker:** Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. By doing so, thanks to the container, the developer can rest assured that the application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code. It Eliminate the “works on my machine” statement once and for all. Gain independence across on-prem and cloud environments.
* **Amazon S3:** Amazon S3 is object storage built to store and retrieve any amount of data from anywhere – web sites and mobile apps, corporate applications, and data from IoT sensors or devices. It is designed to deliver 99.999999999% durability, and stores data for millions of applications used by market leaders in every industry.

# Modules

The modules or sections of the project. Each module should be numbered and a short description along with the duration should be listed. If multiple students will work together on this project, the developer of this module should be listed. ***Each module should take half a page.***

## Instant Texting (IT)

This module builds the basics of the project. With IT, user can keep a list of people they interact with. Users can text with anyone on their connection list or contact list as long as that person is online. You type messages to each other into a small window that shows up on both of your screens. Instant texting is a text-based conversation over the Internet between two or more people who are online at the same time. To access this type of communication technology, users must first subscribe with each other. They then select a screen name and begin chatting. Using this technology, people can, while sitting at their computer, send and receive instant messages. When the is occurring by cell phone (using the phone’s key pad) instead of a computer, the communication is generally referred to as text messaging rather than instant messaging and, as in the case of computers, may involve the transmission of pictures and videos. To equip a cell phone for text messaging, a user must request the service from his or her cellular provider. A third possibility—communicating between computers and cell phones—is becoming increasingly popular.

## Peer-to-Peer Video Calling

Peer-to-peer (P2P) is a decentralized communications model in which each party has the same capabilities and either party can initiate a communication session. Unlike the client/server model, in which the client makes a service request and the server fulfills the request, the P2P network model allows each node to function as both a client and server. This model enables connection from a client device to another client device, allowing them share their live video and audio. This i module is just like the standard video calling module in social media application like Facebook. It's your mobile phone directly connecting to your colleague’s tablet at work, or to your mom’s phone at home. With the new open web-standard, WebRTC and its RTCPeerConnection API, so that data transport from one web browser to another web browser.

## Conference Calling Module

Just as the name suggest, this enables visual connection between two or more people residing in separate locations for the purpose of Enterprise and Social communication.At its simplest, video conferencing provides transmission of static images and text between two locations. At its most sophisticated, it provides transmission of full-motion video images and high-quality audio between multiple locations.

Thus, this module will allow more than two user to be connected to each other, in a full-duplex comunicaton model (i.e a user can conneect with mutltiple user and share text, audio and video data in real time, without interupt). Because WebRTC is design for peer-to-peer communication , a work around will be required for this module to function at a desirable measure. The combination of WebRTC and WebSocket will be used.

## Live Broadcasting

Streaming is a term often used to define the display of video and audio media in real time. Basically, there are two types of streaming. Streaming requires a special service which broadcasts the audio/video information in real time. The player on the visitor's computer interprets this stream of data and visualizes it immediately. Such technology requires powerful dedicated servers because it needs a lot of resources to run correctly. You need such server if you want to broadcast live video for example.

Most of the streaming server technologies record, encode and broadcast your videos in real time making them suitable for websites of TV stations, live shows, or websites like YouTube.com or DailyMotion.com that have millions of visitors playing and uploading videos.

Using mordern technolgy I will be adding this module, which will allow users broadcast to millions of users in real time.

## Screen Sharing

Screen sharing is a tool that enables users to share their screen with other participants on their conference call. By adding a visual element to their conference call, it enhances the experience of their meetings so users can better understand the presentation or topic being explained.

Using technologt that power Google Chrome and Firefox extension, I plan on adding screen sharing to the appliation, due to the many benefits it brings like.

* Training Sessions
* Presentations & Reports
* Sales Demos
* Education & Online Courses
* Team Collaboration & Remote Work

# Risk analysis

* WebRTC is new and still under development, so it more likely that a code written today won’t work tomorrow.
* It system funtionality depends on users device and browser. So for two people to connect their device/browser has to be an updated version.
* Real-Time system need immidiate response, but singnaling server may go down for serveral reasons (like Too much traffic).

What do you think will cause the most issues, is there any chance of failure. ***(at least half a page)***

# Conclusion

## Benefits

* ***Utilise every minute:*** *There's no need to waste money travelling up and down the country for a meeting when you have iMeet. iMeet brings you video chat and screen share 24/7 so you can hold meetings face-to-face without any hassle.*
* ***Wherever, whenever:*** *With iMeet you can work wherever you like, whether you're in the office, or on-the-go, it works wherever you are whilst still delivering the same amazing experience. Best yet you can even swap between devices seamlessly; computer, smartphone or tablet, the choice is yours.*
* ***Take control:*** *As host, you're in control of who is on the main stage. Is it your department's turn to speak? Then put your cube front and centre and bring co-presenters on when you are ready. You can add multiple people to the main stage as you move through your meeting.*
* ***Take notes:*** *iMeet is the only online meeting tool that allows you to take notes in real time during the meeting. You can even link your notes to Evernote, allowing you to easily share your notes with other participants.*
* ***Improve Mobile Experience:*** *WebRTC-enabled technologies allow Mobile Network Operators to create a more compelling customer experience on mobile devices.*
* ***Humanize Your Conversation:*** *Take this point in contrast with voice communication or email correspondence. Video is moving pictures, which are worth more than a million words. By showing yourself and seeing others, you can work the charms of body language, which is so important in business and other activities involving human interaction. Also, seeing someone while talking to them completely changes the nomenclature of a conversation, be it for business or in a personal relationship.*
* ***Learn and Teach Online:*** *There are great courses being offered and great teachers teaching, but most of them may be far, very far from you. If you are a teacher or trainer, your market may be lying far from where you are. This app is a great way of acquiring and sharing knowledge beyond hurdles. While it will not be like being physically present, the interaction is adequate. You will be able to use multimedia facilities like online interactive whiteboards, and you can use online collaboration tools. Courses can take place with the teacher and each different student being in a different location, and that would constitute a real video conferencing session. Alternately, the teacher would be in one location and the class in another, with all students looking at one same screen.*

*Some time ago, video calling or conferencing was a luxury and required expensive and (then) complex equipment and expertise. Today, you literally carry it in your pocket. You can participate in and host video conferencing sessions on your smartphone and mobile device as well as on your computer with basic hardware and adequate Internet connectivity.*

## Ethics

This section is optional, are there any ethical implications of your project. These could be negative or positive.

## Future works

The ideal that lead to this project , was one that I had two year ago during a church service, where the preacher talked about checking out an event that was annouced on Facebook, and seeing so many religious activities and post on my Facebok timeline. It felt a bit off as Facebook didn’t seem to be a fitting community for such. Then I reailized that if a new community can be built for the church, with the right requirement, a kind of social network with religious body as it niche.

Two year ago, I have no knowledge on how to build a server/client web application, but today I have a very good understanding and knowledge of the required component required to build the social network I plan on building in the future.

Completion of this project means I have nearly (90%) completed the list of required knowledge I set out two years ago. I plan on making the application as integrateable as possible, so that I and other developers out there trying to build an application that connect people together in real time, will have no difficulty in doing so. Therefore this projct will continuesly be extened and improve.

# References

Numbered references. Try to keep number of reference above 10. Literature survey should contain many references. References should be like the follows:

[1]: Title, author, year

[2]: Title, author, year, website link

<https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm>

<https://www.mongodb.com/what-is-mongodb>

<https://angular.io/docs>

<https://webrtc.org/>

Socket.io, Mark Mokryn, 2012, <https://www.quora.com/What-is-Socket-io>

https://opensource.com/resources/what-docker