International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 539-545
Published Online August 2020 in IJEAST (http://www.ijeast.com)

VIDEO CONFERENCING USING WEBRTC

Jay Kaushal
Department of Computer Engineering
Terna Engineering College,
Dombivli, Maharashtra, India

Kshitija Khadtare
Department of Computer Engineering
Terna Engineering College,
Sea woods, Maharashtra, India

Dr. Siddharth Hariharan
Department of Computer Engineering
Terna Engineering College,
Vashi, Maharashtra, India

Kshama Gandevar
Department of Computer Engineering
Terna Engineering College,
Panvel, Maharashtra, India

Abstract— **Communication is a significant piece of life. Throughout the years, the way where individuals convey has consistently gotten better, from the more established phones to the cutting-edge cell phones. Alongside this, progress has been produced using simply voice correspondence to video correspondence. With the appearance of the web, video correspondence continuously has become a reality. We have built up a web application utilizing WebRTC that licenses voice just as video correspondence, on a coordinated premise or among numerous clients. Because of the program-based nature of the application, it is gadget free and can run on an assortment of gadgets. This makes it stage free. Moreover, the web application guarantees the security of the information being transmitted. Financially savvy enormous scope continuous video broadcasting innovation has stayed far off in spite of gigantic endeavors and specialized advancements in the course of recent decades. Presentation of Internet Protocol (IP) Multicast and Unicast design were prior endeavors to handle this issue. Be that as it may, genuine intricacies with respect to versatility and troublesome organization have hounded these endeavors. As of late distributed (P2P) based communicate has risen as a promising method since this worldview brings numerous exceptional focal points, for example, cost viability, simple arrangement, adaptability, strength. While P2P applications, for example, record download and voice-over-IP have increased a lot of prevalence, video conferencing is still in its beginning times. The objective of the following paper is to provide information on VIDEO CONFERENCING SYSTEM BASED ON Web Real Time Communication (WebRTC). WebRTC gives the chance of building up P2P associations inside web, without extra modules and outsider programming. This paper presents a versatile live video conferencing engineering planned dependent on WebRTC. Our trials show that WebRTC is an able structure hinder for versatile live video conferencing inside an internet browser.**

Keywords— WebRTC, peer-to-peer communication, testing, maintenance

I. INTRODUCTION

Video conferencing has gained significant popularity in recent years. Video conferencing applications are extremely testing since they have to at the same time bolster an enormous number of members, manage dynamic changes to member participation and adapt to the high transfer speed prerequisite. In addition, during a multiparty video gathering video gushing. Load increases with the number of users in the conference as each user could generate and distribute video streams.

1.1. AIM

The aim to make simultaneous video broadcast a common internet communication so that anyone can broadcast video content to any number of receivers has provided motivation behind this research agenda for a long time now.

1.2. SYSTEM OVERVIEW

We have made a web application that provides video communication between two and more than two users over the internet. The application needs only a browser, no other plug-in or software has to be installed, making the application platform independent. Our application allows the user to perform a video call, which is one to one, or a voice call which includes only audio or a multi-chat, in which more than two users can participate.

Video communication was first started in the form of Video-telephony. Video-telephony is an augmentation to the phone; it is a method of synchronous, two-way correspondence including both sound and video segments. Clients can both hear and see each other continuously. WebRTC disentangles the procedure of video correspondence. It is a lot of JavaScript APIs, conventions and principles by Google, drafted by W3C that permits program based correspondence and information move. It is a lot of JavaScript APIs, conventions and principles by Google, drafted by W3C that permits program based correspondence and information move. WebRTC is open source. The normalization of WebRTC is as yet going on, cutting edge executions have fundamentally been done in Mozilla Firefox and Google Chrome programs. By and by, work area programs, Google Chrome, Mozilla Firefox, Opera and versatile stages like Android support WebRTC

1.3. WEBRTC

Web Real Time Communication is a lot of open source, JavaScript APIs, gauges and conventions, drafted by W3C and created by Google. WebRTC permits program based distributed sound, video correspondence and information sharing. It doesn't rely upon outsider modules or select programming. Permitting ongoing correspondence in the program is unquestionably one of the most significant additional items to the web, since the beginning. WebRTC occupies from the customary customer server model, the consequence of which is, finished re-building of the systems administration layer in the program, and furthermore the expansion of another media stack required to empower productive, continuous handling of sound and video. The upsides of utilizing WebRTC are as per the following, it doesn't rely upon any product or module it is totally program based. It is open source and henceforth no sovereignties must be paid to Google. It makes the activity straightforward for designers and the best sound and video motors have been included by Google. The main downside of WebRTC is that it is a standard still a work in progress; the code can experience basic changes sooner rather than later. The subtleties of the WebRTC API regarding usage are as per the following. Friend Connection is utilized to make an association with a companion. It takes in information about which servers to utilize and decisions for the kind of association. Intuitive Connectivity Establishment (ICE) is a system to permit the internet browser to associate with peers, this structure finds competitors, to empower the client to interface with the friend, this is called ICE applicant Signal Channel is a strategy used to send metadata and ICE contender to the friend. In our application we have utilized Firebase as the Signal Channel. The metadata that discloses to the next companion the arrangement to expect (codecs, goals, size, video and so forth.) is called an Offer SDP (Session Description Protocol).

For a trade to happen between peers, one friend must send an offer and the other companion ought to get it and answer with an answer. The Offer is imparted through the Sign Channel. An Answer SDP is a reaction that is answering the phone. An answer can only be caused once the offer has been received.

II. LITERATURE REVIEW

There have been numerous investigates that investigated the building decisions for Internet communicate. In [2], the creators checked on the engineering decisions for supporting Internet communicate.

2.1. ROUTER-BASED ARCHITECTURES: IP MULTICAST:

In the Internet condition, the significant issue for communicate is deciding the layer of usage. There are two clashing contemplations 1) a usefulness ought to be pushed to higher layers if conceivable except if 2) execution at a lower layer can accomplish huge execution benefits that exceeds the extra multifaceted nature. In [3], the creators contended that this subsequent thought ought to win and it has since been broadly acknowledged, prompting the IP multicast model of today. Lamentably, in spite of the colossal exertion in the past the present IP multicast organization stays restricted. Three complex reasons are making this methodology less appealing. To begin with, IP multicast requires switch bolster which presents high multifaceted nature and genuine scaling limitations. Furthermore, IP multicast endeavours to adjust to the conventional partition of steering and transport that has functioned admirably in the unicast setting.

2.2. NON-ROUTER-BASED ARCHITECTURES: UNICAST:

In the new thousand years a few specialists proposed to move multicast usefulness away from switches towards end frameworks [4], [5], [6], [7]. In these methodologies, multicast-related highlights are executed at end frameworks, accepting just unicast IP administration. Moving multicast usefulness to end frameworks can possibly address a significant number of the issues related with IP multicast. Arrangement is quickened on the grounds that all bundles are transmitted as unicast parcels. Likewise, answers for supporting higher layer highlights can be fundamentally rearranged by utilizing unicast arrangements and application explicit insight. Be that as it may, moving multicast usefulness away from switches has clear execution punishments. For instance, forestalling various overlay edges from navigating the equivalent physical connection isn't totally conceivable so there will consistently be repetitive traffic on physical connections. Further, correspondence between end frameworks need the crossing of opposite end frameworks which builds inertness. Consequently, many research endeavours have concentrated on tending to these exhibition concerns.

2.3. PEER-TO-PEER ARCHITECTURES:

Each friend which additionally can be known as a client or hub conversely, goes about as both recipient and sender simultaneously. P2P expects hubs to be equipped for discussing straightforwardly with different hubs utilizing a subset of hubs in the system. Spreading media stream, hubs utilize their correspondence abilities. Associations are made between subjective hubs and a spreading over tree can be built from a total diagram that assists with interfacing all other single hub. It is trying to give multiparty video-conferencing administration utilizing customary server-based arrangements because of its high data transmission request and tough spilling quality necessity. In [8], the creator proposed an End System Multicast engineering to help video- conferencing applications, where multicast usefulness is pushed to the edge. In [9], the creator proposed to coordinate application-layer multicast with local IP multicast in P2P conferencing frameworks. In [10], the creator proposed half and half answers for utilize partners to augment the utility in P2P conferencing, where assistants help clients in handing-off video streams to recipients.

2.4. PROPOSED SYSTEM:

Overview of proposed framework Figure beneath shows a diagram of the proposed P2P sound and video calling application work. This proposed framework furnishes direct continuous association with remote friends with no server. So as to start the association, clients need to experience the flagging server by distinguish and find the remote friends.

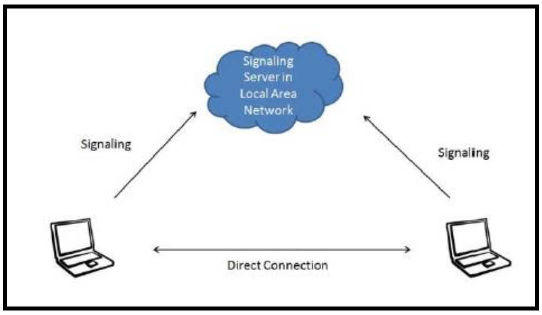

Fig 1: Overview of Proposed System

Traditional Customary P2P overlay models depend in the way that hubs have open IP addresses and are effectively tuning in on a predefined port. When setting aside the intricacies presented by firewalls and NAT, this empowers exchange of new associations straightforwardly dependent on the information on another hub's open IP address. In WebRTC, a Peer Connection can be set up between clients to permit program to program (P2P) correspondence. Be that as it may, for this to be conceivable, a flagging channel is required, and Web Sockets permit this channel to be given by the web server. A meeting between the friends is built up utilizing JavaScript Session Establishment Protocol (JSEP). JSEP gives instruments to making offers and answers in this way characterizing the substance of the necessary meeting exchange messages. In any case, how these messages are traded between two potential friends isn't determined in JSEP, however is made conceivable by Web Socket associations among programs and the Web server.

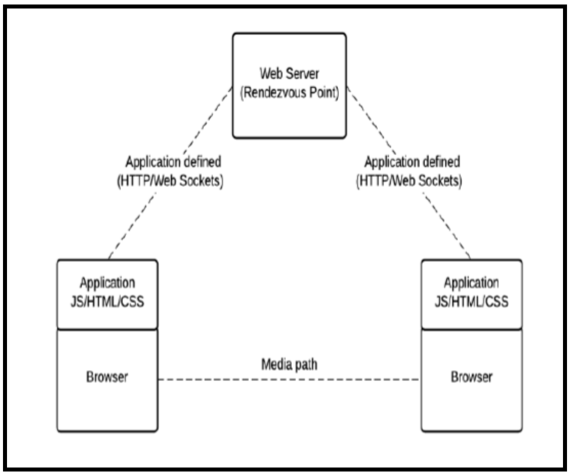

Fig 2: Overview of WebRTC architecture

A key distinction between a customary P2P application and a WebRTC web application is the powerlessness to straightforwardly set up associations with peers, in spite of knowing their open IP-address. Limitations forced by the program sandbox deny a JavaScript application to tune in for approaching rush hour gridlock from self-assertive sources. Accordingly, to have the option to trade P2P messages in WebRTC, customers need to arrange information channels utilizing a current flagging channel. On the off chance that the objective is to send lumps of parallel information, a Data Channel must be haggled between the companions. A Web application utilizing WebRTC will be stacked from a web server. By permitting the customers to interface with the webserver utilizing Web Sockets, it can hand-off messages to and fro among customers, and accordingly flexibly the necessary flagging channel. Henceforth the web server can go about as an intermediary for JSEP flagging, permitting association foundation between customers, subsequently going about as the RP in the overlay organize

2.5. WEBRTC ADVANTAGES:

In modern day economy, to embracing WebRTC for businesses as part of their enterprise to reduce costs and remain effective and competitive. Most leading Browsers support WebRTC for Windows.

- WebRTC is known as an open-source application programming interface i.e. API.
- WebRTC enables the browser of any operating system.
- a real-time voice or video connection to another WebRTC device or to a WebRTC media server can directed to a web services application

- It allows on voice and video encryption. The encryption and authentication of both voice and video is provided by Secure Real-time Transport protocol (SRTP).
- Advanced voice and video quality is provided by WebRTC.
- Reliable session establishment is supported by WebRTC.
- The negotiation of multiple media types and endpoints is supported by WebRTC.

III. METHODOLOGY

3.1. .NET FRAMEWORK:

The programming model for building applications on Windows clients, servers, and mobile or embedded devices is the Microsoft's Managed Code .NET Framework.

It is a software technology which is available with several Microsoft Windows operating systems. The fundamentals of Microsoft .Net Frame work Technology and its related programming models.

C# is a language used for professional programming. C# is designed for building extensive enterprise applications that run on the .NET Framework. The objective of C# is to provide a object-oriented, high performance, safe, modern language for .NET development. Also, it authorizes developers to build solutions for the broadest range of clients, including Web applications, Microsoft Windows Forms-based applications, and thin- and smart-client devices.

3.2. SYSTEM DEVELOPMENT LIFE CYCLE:

The process of developing information systems through analysis, design and implementation is called the System Development Life Cycle . The System Development Life

Cycle I.e. SDLC can also be known as Information Systems Development or Application Development.

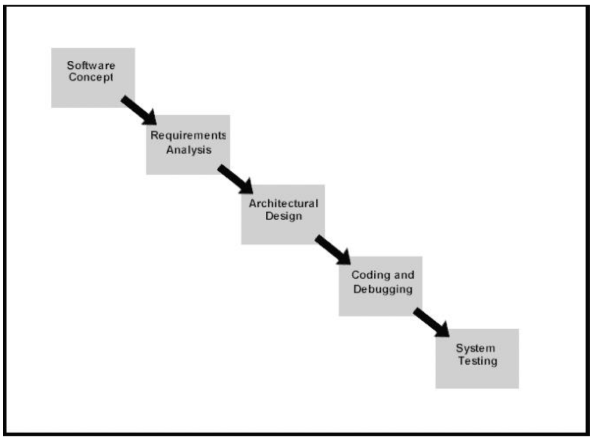

Fig 3: Development Life Cycle

Steps involved in the System Development Life Cycle are as follows:
Each phase within the overall cycle can be made up of multiple steps.

Step 1: Software Concept
The first step is to recognize a need for the new system. This will include decide whether a opportunity or business problem exists by conducting a feasibility study to determine if the proposed solution is cost effective.
Step 2: Requirements Analysis
The process of analysing the details needs of the end users, the organizational environment, and any system presently being used, developing the functional requirements of a system that can meet the needs of the users is called requirements analysis.
Step 3: Architectural Design
After the requirements are gathered, the necessary specifications for the hardware, software, data resources, and the information products that will satisfy the functional requirements of the proposed system will be determined.
Step 4: Coding and Debugging
Coding and debugging is nothing but creating the final system. This step is always done by the software developer.
Step 5: System Testing
To evaluate its actual functionality in relation to expected or intended functionality the system must be tested. During this stage converting old data into the new system data and training employees to use the new system takes place. End users will determine whether the developed system meets the intended requirements, and the extent to which the system is actually used.

Step 6: Maintenance
The system will need maintenance inescapably. Software undergoes changes once it is delivered to the customer while maintenance takes place. There are multiple reasons for changing the software. Changes could be done due to some unexpected variable within the system. In addition, the

changes in the system could personally affect the software operations. The software should be developed to fit in with the changes that could take place during the post implementation period.
Various software process models are: -

- The Iterative Model
- Prototyping Model
- The Spiral Model
- The Waterfall Model
- RAD Model

3.3. FEASIBILITY STUDY:

Any system developing life cycle consists of the first step as preliminary investigation. The feasibility study is a major part of the first step. How beneficial and practical a development of any information system is, is checked by feasibility study.

The feasibility of the development software is:

- Technical Feasibility.
- Motivational Feasibility.
- Operational Feasibility.
- Economic feasibility.

3.4. SYSTEM PLANNING AND SCHEDULING:

A bar chart that illustrates a project schedule is called a Gantt chart. They are potent to show visual timeline in project management. Gantts shows the start date and the end date for each task that must be followed in order to successfully complete a project.

It includes a plan for the first half of the year. The plan includes - project title finalization, literature survey, business case, project charter, requirement gathering and user survey, implementation and testing of basic functionality, implementation and testing of GUI, implementation and testing of machine learning functionality, synopsis and report and final presentation.

3.5. TESTING TECHNOLOGY:

System testing is a condemnatory phase of implementation. Testing of the system involves debugging of the computer programs and testing information processing method. Testing can be done with the help of text data, which attempts to stimulate all feasible conditions that may arise during processing. If structured programming approach have been adapted during coding, the testing proceeds from higher level to lower level of program module until the entire program is tested as unit. The testing methods adapted during the testing of the system were unit testing and integrated testing.

3.5.1. UNIT TESTING:

Unit testing focuses first on the phases to locate errors. This enables the tester to detect errors in coding and logical errors that is accommodated within that phase alone. The results from the interaction between modules are initially avoided.

3.5.2. INTEGRATION TESTING:

The integration testing can be known to be a systematic technique for building the program structure while at the same time to expose the errors associated with interfacing. The aim is to take unit-tested phase and build a program structure that has been discovered by designing. It also tests to find the inconsistency between the system and its aims. Subordinate stubs are replaced one at a time in the actual phase. Tests were supervised at each phase after it was integrated. On completion of each set another stub was replaced with the real phase.

3.5.3. FUNCTIONAL TESTING:

Functional testing is a approach in which all the performances of the program are tested to check whether all the functions that where proposed during the planning stage were full filled. This is also done to check if all the functions proposed are working properly.

This is further done in two phases:

• First, to see if they still work properly after they have been integrated to check if some functional similarity issues arise.

• Secondly, before the incorporation to see if all the unit components work properly.

3.5.4. PERFORMANCE TESTING:

Expected Result

• The client should connect to the server properly without any problems or difficulty.

• The connection establishment between the mobile device and the server should take least possible time.

• The mobile device should be able to receive data from the server in a steady manner.

• Details provided by the application should be correct and as per the user's need.

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 539-545
Published Online August 2020 in IJEAST (http://www.ijeast.com)

Observation

- Connection can be established easily provided that the server is on and the user has good internet access.
- The connection with the server will consume time as it uses Internet connection.
- Receiving data from the server also may take time.
- Information coming from the database will always

  be correct.

3.5.5. LOAD / STRESS TESTING:

Expected Result

- Response time should be unaltered disregarded of the no of users.
- The introduction of the new users should not the disturb the server.
- Constant use of the server by different clients should not result into the server getting slowed down.

• Response time should not be degraded if there is crowding in network.

Observation

The speed of transferal will be okay even when the newer users are getting added. The response of the server is fulfilling even with the introduction of new users.

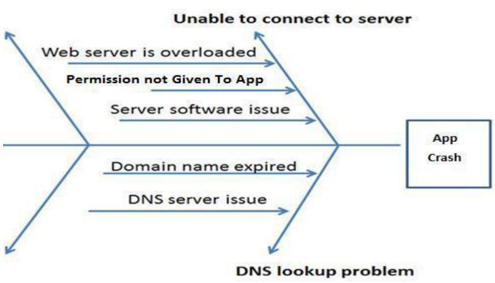

Fig 4: Cause Effect Diagram

IV. EVALUATION

System evaluation provides framework for categorizing scheme to recognize sets of similar systems. The framework amalgamates previous studies on software evaluation, productivity models, software, and total quality models. It also classifies details about software systems from the perspective of the project and the environment.

V. RESULTS AND DISCUSSION

After completing the video conference system using WebRTC, the following results has been achieved:

- Web-based video conference system was developed.
- User can make or join a room.
- Users can communicate through voice or video calls.
- Users in the same communication room can chat with each other.
- User who creates the communication session can know how attend the call.
- The system reduces physical effort by using this video conference system instead of a real life meeting.
- It provides easy and simple communicating and sharing files.
- User can attend or make a call at any place or time.

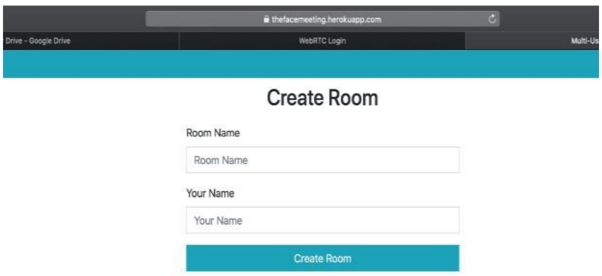

Fig: 5

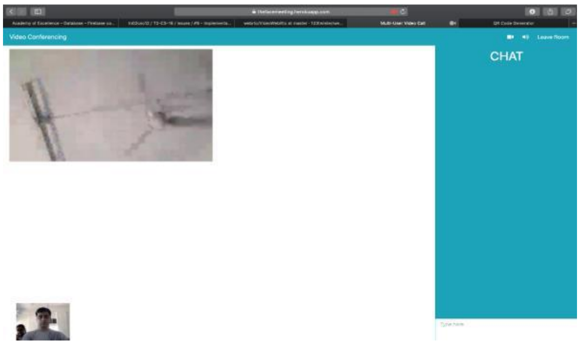

Fig 6: VIDEO CONFERENCING WINDOW

VI. SUMMARY AND CONCLUSION

A video conferencing system using WebRTC technology was created and approved. In this research we used the WebRTC technology for its ease of use and does not need any plugins or application to install it just need browser that it supports. The video conference system is implemented as a web-based browser to be used from different operating systems. The aim of this research is to reduce the effort and difficulty of mobility to communicate and to create a video conference that supports the characteristics of voice calls, video calls, share files, share desktop, record in different format and attendance for who attend. These goals have been achieved.

The system is not perfect, perfection is not something humans can accomplish but we as developers did our best to provide all the features, we need to ease communication, health care, education and other fields required in our country.

VII. REFERENCES

[1] Peer to Peer Multimedia Real-Time Communication System based on WebRTC Technology by Zinah Tareq Nayyef, Sarah Faris Amer, Zena Hussain.

- A Vector-based Trust Framework for WebRTC Calling Services by Ibrahim Tariq Javed, Khalifa Toumi, Noel Crespi, Amir Mohammadinejad
- Getting started with webRTC -HTML5 Rocks
- WebRTC -GitHub