International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 1026-1036 www.ijaem.net ISSN: 2395-5252

Virtual 3d Classroom

[1]Savio Siby, [2]Shameel K

[1,2]*Student, Dept. of Computer Science & Engineering, Mar Athanasius College of Engineering, Kothamangalam- Kerala, India*

Date of Submission: 15-09-2022 Date of Acceptance: 24-09-2022

ABSTRACT: Since the COVID pandemic learning across the world has switched to online learning, but the platforms of today do not captivate the younger students to actively partake in learning with their teachers. To solve this problem, we propose an application that binds the features of presentvirtual classroom solutions like Zoom and Google Classroom into a virtual 3D classroom environment. This Virtual classroom is built using HTML5 WebGL™ as a framework with Ammo Physics for interactions, WebRTC for voice communication. To increase student immersion features like proximity chatting, character customization, animations, gestures, interactions etc. will be incorporated into the application. We believe the platform where the user actively participates is the most successful one.

KEYWORDS: Classroom, WebRTC, Peer-to-Peer, Online Class,Ammo JS, HTML

I. INTRODUCTION

Since the start of 2020 the global pandemic has affected mankind's way of life in many ways. The COVID-19 pandemic has brought about a time of change and forced us to adapt to the situation at hand. One such liveliness affected was our education system. This has made us to rethink the traditional school model and move on to an online mode of education. Furthermore, with this sudden shift away from the classroom in many parts of the globe, the adoption of online learning will continue to persist post-pandemic, and such a shift would impact the worldwide education market. The online model of education removes restrictions such as a need for physical presence and to learn in confines of one's home, allows students' choice and flexibility to learn at their pace, creates the ideal condition to continue schooling with a raging pandemic. This brings in the need for efficient, easy and interactive way to use online platforms for education.

1.1 Motivation

The present platforms such as Google Meet, Cisco WebEx, and Zoom etc. are video-conferencing applications aimed for an older audience. Younger kids usually when using such applications their attention drifts off while staring at a computer screen. What we need is a stand-alone custom-made application for the job at hand which should satisfy the needs of the student and teachers as one does in a classroom. We developed our idea from the exponential rise in kids who became addicted to the MOBA RPG called PUBG when released in India back in March 2017. Number of users in India has surpassed all other countries in a short span of time. The main reason for such growth is the inter-activeness of the game able to captivate young kids. Incorporating the same ideas, we developed a game like environment as a classroom that kids can immerse with. We believe such an application will allow kids to grasp the information taught to them at a quicker and efficient rate.

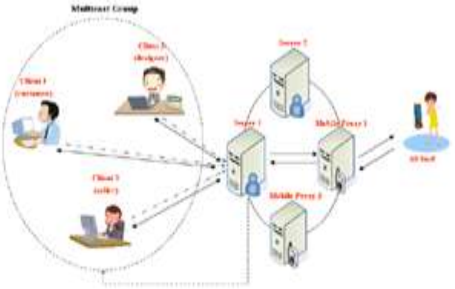
We propose a virtual 3-D environment similar to multiplayer video games built using WebRTC and Peer-to-Peer platform specifically optimized for education that functionswith reduced latency across all platforms. Initially implemented as a browser application this canmeet all their educational requirements in one place provides basic tools like screen sharing andvoice communication but also adding new features like personalized avatar, facial expression, interactionwith other users and objects, player movement and animations, proximity chatting and so on.

1.2 Major Contribution

The important contribution of the project is the design of a 3-dimensional virtual classroom that can replace existing online platforms such as Zoom and Meet. Here each individual gets represented as a 3D model with capabilities to perform interactions through real life actions. By using3D graphics we built a classroom widely accessible through web browsers where users can authenticate themselves into. Another contribution of this project is Proximity Chat, allowing participants to talk with a select few, those who are in their surroundings. Also features like streaming of studymaterials through a virtual screen are also built into the project.

II. AREA OF RESEARCH

2.1 Video Conferencing



The term “video conferencing” or “web conferencing” encompasses collaboration technologies that enable users to place calls via an application over the internet, through software downloaded locally to a computer or device, or in an on-premises environment. Web conferencing encompasses popular real-time collaboration solutions like video conferencing, screen share, document sharing, and online meetings. Web conferencing software allows participants to conduct or attend meetings via the internet. This is also known as online meeting software or sometimes simply video conferencing itself. It enables remote meetings based on VoIP, online video, instant messaging, file sharing, and screen sharing. Web conferencing is made possible by Internet technologies, particularly on TCP/IP connections. Services may allow real-time point-to-point communications as well as multicast communications from one sender to many receivers. It offers data streams of text-based messages, voice and video chat to be shared simultaneously, across geographically dispersed locations. Some examples of such software include Google Meet, Zoom, Microsoft Teams, etc.

2.2 Massively Multiplayer Online Games

A massively multiplayer online game (MMOG, or more commonly, MMO) is an online game with large numbers of players, often hundreds or thousands, on the same server. MMOs usually feature a huge, persistent open world, although there are games that differ. These games can be found for most network-capable platforms, including the personal computer, video game console, or smartphones and other mobile devices.

MMOs can enable players to cooperate and compete with each other on a large scale, and sometimes to interact meaningfully with people around the world. They include a variety of gameplay types, representing many video game genres. Most MMOGs also share other characteristics that make them different from other multiplayer online games. MMOGs host many players in a single game world, and all of those players can interact with each other at any given time.

2.3 3D Video Game Graphics

With the advent of 3D accelerated graphics, video games could expand beyond the typically sprite-based 2D graphics of older graphics technologies to describe a view frequently truer to reality and lifelike than their predecessors. Perspective projection has also been used in some earlier titles to present a 3D view from a fixed (and thus somewhat less hardware-intensive) perspective with a limited ability to move.

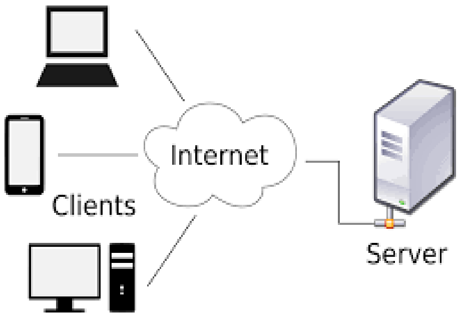
Third person refers to a graphical perspective rendered from a view that is some distance away (usually behind and slightly above) from the player’s character. This viewpoint allows players to see a more strongly characterized avatar, and is most common in action and action-adventure games. This viewpoint poses some difficulties, however, in that when the player turns or stands with his back to a wall, the camera may jerk or end up in awkward positions. Developers have tried to alleviate this issue by implementing intelligent camera systems, or by giving the player control over the camera. There are three primary types of third-person camera systems: “fixed camera systems” in which the camera positions are set during the game creation:” tracking camera systems” in which the camera simply follows the player’s character; and “interactive camera systems” that are under the player’s control.

2.4 Game Physics and Physics Engine

Computer animation physics or game physics involves the introduction of the laws of physics into a simulation or game engine, particularly in 3D computer graphics, for the purpose of making the effects appear more realistic to the observer. Typically, simulation physics is only a close approximation to actual physics, and computation is performed using discrete values. Additionally, games may intentionally deviate from actual physics for gameplay purposes; a common example is allowing the player to double jump when there is nothing to jump from, or changing the values of certain physical parameters, such as the amount of gravity present. There are several elements that form components of simulation physics including the physics engine, program code that is used to simulate Newtonian physics within the environment, and collision detection, used to solve the problem of determining when any two or more physical objects in the environment cross each other’s path.A physics engine is computer software that provides an approximate simulation of certain physical systems, such as rigid body dynamics (including collision detection), soft body dynamics, and fluid dynamics, of use in the domains of computer graphics, video games and film (CGI). Their main uses are in video games (typically as middleware), in which case the simulations are in real time. The term is sometimes used more generally to describe any software system for simulating physical phenomena, such as high-performance scientific simulation.

2.5 Web Communication

1. Client-Server Communication



A client and server networking model is a model in which computers such as servers provide the network services to the other computers such as clients to perform a user based tasks. This model is known as client-server networking model. An application program is known as a client program, running on the local machine that requests for a service from an application program known as a server program, running on the remote machine.

A client program runs only when it requests for a service from the server while the server program runs all time as it does not know when its service is required. A server provides a service for many clients not just for a single client. Therefore, we can say that client-server follows the many-to-one relationship. Many clients can use the service of one server.

Services are required frequently, and many users have a specific client-server application program. For example, the client-server application program allows the user to access the files, send e-mail, and so on. If the services are more customized, then we should have one generic application program that allows the user to access the services available on the remote computer.

2. Peer-to-Peer Communication

Peer-to-peer (P2P) computing or networking is a distributed application architecture that partitions tasks or workloads between peers. Peers are equally privileged, equipotent participants in the application. They are said to form a peer-to-peer network of nodes.

III. LITERATURE SURVEY

Today there are lot of video chat apps like Zoom, Google Meet, Webex etc. but education-oriented features are less seen in these apps. Google Classroom is one of the apps that focuses on bringing features to enhance online teaching. But we could not arrange any kind of live meeting from the app, we should use other third-party apps to create video conference. It helps in assigning works, uploading recorded class videos, e-textbooks, PowerPoints etc. Below are the detailed explanation of Zoom Platform, Google Classroom and its drawbacks.

3.1 Zoom

Zoom is a cloud-based video conferencing service. It is intended for businesses to work remotely and offers a wide range of services to accommodate any business’s needs. These include video and audio conferencing, collaboration, chat, and webinars.it supports on both desktop and mobile devices.

3.2 Google Classroom

Google Classroom is a form of learning management system (LMS) that pulls in genuinely useful tools that are free and easily accessible on many devices. You’ve probably already used many of them and that’s of key importance here, as most people have used Google Docs, Sheets and Slides, making this Classroom easy to get to grips with. The result is a streamlined offering that makes learning online a clear process for both teachers and students of all levels. When a teacher logs into the service they can see a section for each of their classes, withcourses differentiated by colors and banners. Name the class, enter a description, pick the banner image and it’s good to go. Resources can be shared to students all at the same time and they’re notified by email. So, a teacher can share and annotate a Google Doc, Sheet or Slide and keep track of when students had that distributed. Since you can pull material in from Google Drive this can also be an easy process. While a presentation can be carried out using Google Slides, the teacher is also able to create areal-world view by using Google Hangouts to see and hear the students for video interactions.

3.3 Google Meet

Google Meet (formerly known as Hangouts Meet) is a video-communication service developed by Google. It is one of two apps that constitute the replacement for Google Hangouts, the other being Google Chat. While Google Meet introduced the above features to upgrade the original Hangouts application, some standard Hangouts features were deprecated, including viewing attendees and chatsimultaneously. The number of video feeds allowed at one time was also reduced to 8 prioritizing those attendees who most recently used their microphone. Additionally, features such as the chat box were changed to overlay the video feeds, rather than resizing the latter to fit. Hangouts is scheduled to cease operationin the first half of 2021. During the 2020 COVID-19 pandemic, the use of Meet grew by a factorof 30 between January and April 2020, with 100 million users a day accessing Meet, comparedto 200 million daily uses for Zoom as of the last week of April 2020. Google suspended its usual60-minute limit for unpaid accounts.

3.4 Drawbacks of past works

- Google classroom doesn‘t have integrated features for video conferencing
- Zoom doesn’t have features on Learning Management Systems
- Network status monitoring is absent on both platforms
- Live Attendance is absent on both platforms
- Zoom, Google Meet cannot hold the attention of younger audience

IV. PROPOSED SYSTEM

In this chapter we discuss the necessary features that are needed to build the platform. The application can be divided into two phases: back-end and front-end. The front-end of a software program or website is everything with which the user interacts. From a user standpoint, the front-end is synonymous with the user interface. From a developer standpoint, it is the interface design and the programming that makes the interface function. We use WebGL to design the front-end. Back-end Development refers to the server-side development. It focuses on databases, scripting, and website architecture. It contains behind-the-scenes activities that occur when performing any action on a website. We use WebRTC for back-end.

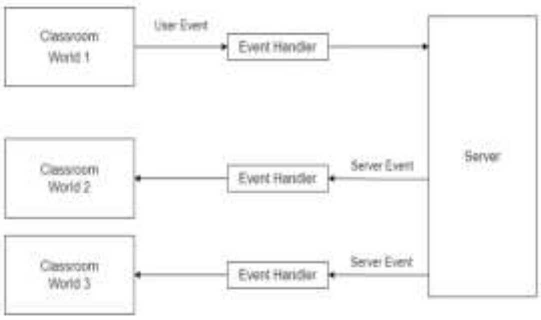
4.1 Browser Based 3D World

We made a virtual 3D world that can be accessible online, and can be run on common browsers. It work on the Object oriented model. Everything in the virtual world is treated as objects and each object has their own state, attributes.

WebGL is using for the implementation of the 3D world, in order to attain hardware acceleration of 3D graphics, WebGL is a JavaScript API for rendering interactive 2D and 3D graphics within any compatible web browser, without the use of plug-ins. WebGL is fully integrated with other web standards, allowing GPU-accelerated usage of physics and image processing and effects as part of the web page canvas.

WebGL elements can be mixed with other HTML elements and composited with other parts of the page or page background. A platform using WebGL for browsers works at the client devices. We use Three.JS to build a cross-browser JavaScript library and application programming interface(API) to create and display animated 3D computer graphics in a web browser using WebGL. Environment Physics is managed by using Ammo Physics Library, The platform we create using Ammo Physics will obeys the laws of physics of real world. In Ammo.JS this world is called a Collision World and has among its derivatives the Dynamic World. The physics world has options to set gravity and expose functions and objects for the following to be possible.

4.2 Server Based State Synchronization



The state of a system is to be synchronized through an intermediate server. A TCP connection is made to the server whenever a client joins which only stops when the user exits. An event handler is used to send and receive events between browser and clients. Browser accepts the inputs from user then send to the event handler which applies the input and applies on the classroom also sends the same to the server. Server accepts the data from a client, and change are made on server copy of the world state then it broadcasts to other devices. The events send from server is accepted by event handlers and then applied on the created world.

4.3 Peer To Peer Voice Communication

To implement communication between users, voice communication can be enabled. Peer to peer voice communication will help to reduce server load and latency. WebRTC technology is used to create a peer to peer voice connection. Voice agent creates a unique identification for each user, and thus creates a connection between the devices.

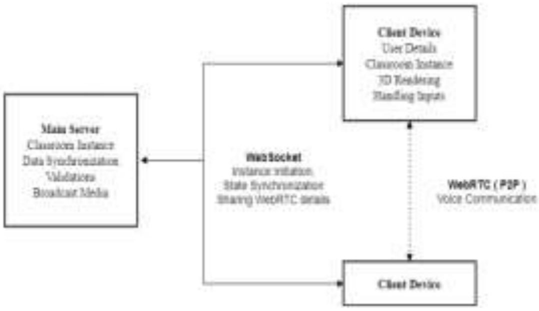
V. DESIGN

The most creative and challenging phase of the system life-cycle is the system design. The term design describes a final system and the process by which it is developed. It refers to the technical specifications that will be applied in implementing the candidate system. In system design, there is a movement from the logical to the physical aspects of the life cycle. This is the phase in which the details of the system selected in the study phase is designed. Software design is the preliminary step and is also building block of software engineering. The efficiency software is promoted through design phase. The design phase begins when the requirement specification document for the software to be developed is available.

Design is the first step to moving from the problem domain to the solution domain. Design is essentially the bridge between requirement specification and the final solution for satisfying the requirements. Design of input and outputs are important features of design specification. The input design is the links that tie the system to the world of its users. Output generally refers to the results that are generated by the system. The first step is to determine how the output is to be produced and in what format. Then input data and master files have to be designed at the next step and finally the impact of the candidate system on the user and organization are documented and evaluated by the management. After identifying the problem and the limitations of the existing system, a detailed design of the proposed system is conducted. Design is the phase that indicates the final system. It is the solution, the translation of requirements into ways of meeting them. In this phase the following elements were designed namely, data-flow, data stores, processes, procedures.

Firstly the logical design was done where the outputs, inputs, databases and the procedures was formulated in a manner that meet the project requirements. After logical design physical construction of the system is done. The Input screens, output screens, output reports and graphs are designed. After analysing the various functions involved in the system the database, labels dictionaries are designed. Care is taken for the field name to be in self-explanatory form. Unnecessary fields are avoiding so as not affecting the storage of the system. Care must be taken to design the input screen in the most user-friendly way so as to help even the novice users make entries approximately in the right place. This is being accomplished by the use of giving online help messages, which are brief and cleanly prompts users for the appropriate action. The characteristics of a well-defined system are: accessibility, decision making capability, economy, reliability, simplicity.

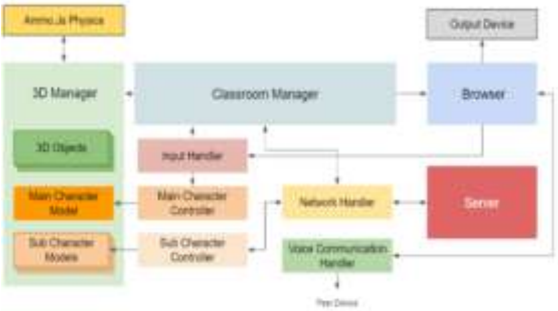
5.1 System Model



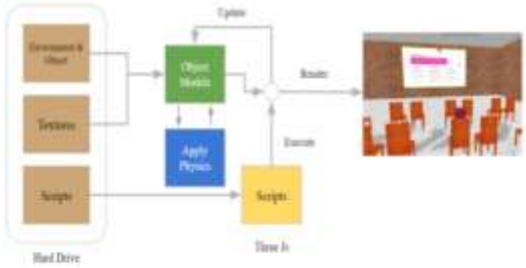
The Server is deployed in a network that is accessible from every client device. The server serves the static files for client. It contains a copy of the classroom world and its present state and it is the most authentic copy. Events from clients are accepted and validated in the server. Corresponding changes are made on the local instance copy and the data is broadcasted to other devices including the metadata of media streaming into virtual screen.

Client renders the web application using the static files received from the server. A classroom instance is created on the client device. The handling of the 3D environment and their rendering works in this instance. Browser accepts the inputs from the user and apply the same on the platform and is also sent to the server. Events that are happening in the client are sent to the server using web-sockets. A web socket connection is made between client and server initially when the client platform is set up. It is a TCP connection which is only closed when the platform is closed or in real world the browser tab is closed. The server accepts this data as events and emits the data as a broadcast.

5.2 Client Platform



The browser will create a classroom manager instance which handles the entire process that happens in the client platform. The classroom manager will initiate a 3D manager and the 3D manager will create 3D world along with physics engine using Ammo.JS physics. In the 3D world there will be 3D objects as object oriented models. There will be a 3D character for main user and also other users. For controlling these 3D characters we will use a character controllers. The main character controller is used to control the primary user it takes input from the browser and sends player details to server. The sub character controllers gives the details of other users it takes the input from the server and display the character details on the browser.

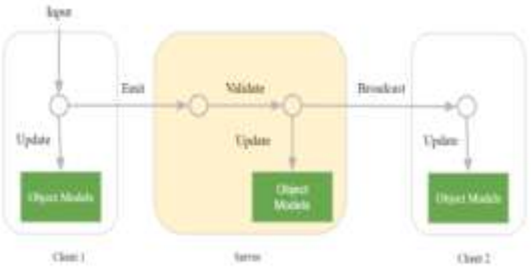


The classroom manager will create a input handler that will accept input from the browser and informs it to the main character controller which will make necessary effects on the main character 3D model like movement, rotation and various actions. The Input handler will also informs the server regarding this input events through network handler. The server will then inform the other devices regarding this event. The network handlers of other devices will accept events from the server and transfer it to other sub character controllers Sub character controller will make necessary changes on the 3D models of other characters.

Voice communication handler will accept audio input from the browser and streams each voice to the other connected devices. It also receives stream from other devices and gives it into the classroom manager. The classroom manager will adjust the volume level according to the distance between characters and then transfer the stream to the browser for audio output. This is how we enable proximity voice communication.

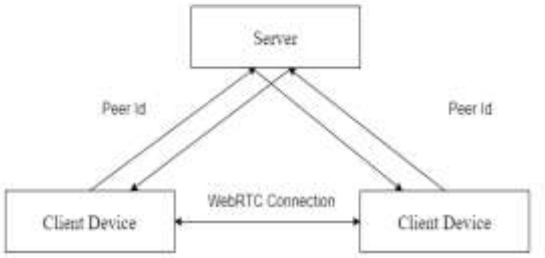
5.3 Character Movement

The Operating system maps the input system to the browser. Browser informs the classroom manager the input send from the user through mouse and keyboard. The classroom manager will map the inputs to events. A ’W’ key from keyboard is mapped for forward movement event. Also mouse movement in thex-y direction is mapped intocharacter rotation event. The ’C’ key from keyboard is mapped to character sit event. The input manager will call particular functions of main character controller for each event.

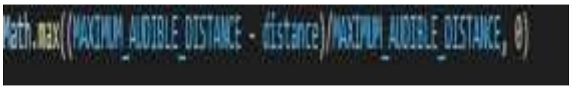


Main character controller will perform actions regarding the functions called by the user. If move forward is needed to be performed a particle velocity is applied for the object. For character rotation event the mouse movement delta X is map into rotating camera around y axis also delta Y is used for rotating camera along x axis from a third person perspective behind each user model. The input manager also inform the network manager regarding the event. The event is synchronised through the server into other devices and the other devices also handle the events like main character controller.

5.4 Voice Communication



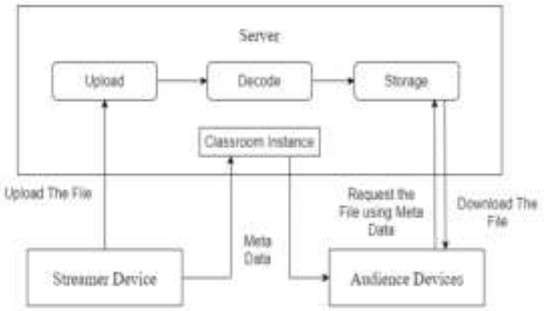
The voice communication handler in the client device creates a unique ID for all the devices. The clients informs the server its own peer ID which is further broadcasted onto other devices. When a device receives an Id it tries to create a communication channel between the devices by calling them using the peer ID. When a communication channel is established the browser will start listening to the microphone and converts the voice input as stream of bytes. Then the stream is transferred into other devices through this communication channel. Also the browsers create an HTML audio element and sets the source as audio stream received from other clients. During each animation frame the classroom manager calculate the distance between main character and each sub characters and informs the distance to the voice communication handler The voice communication handler adjust the audio level using the below function.



```
Math.max((MAXIMUM_AUDIBLE_DISTANCE - distance)/MAXIMUM_AUDIBLE_DISTANCE, 0)
```

5.5 Study Material Streaming

In order to reduce the bandwidth real time streaming is not used. The study materials are streamed through the server. The files are to be uploaded into the server for streaming. The server will decode the file and splits it into smaller and optimised pieces and store it in the server. This stored data is accessible to other clients. Also the streamer will update the metadata of current streaming media in the classroom instance at the server side. The current streaming media metadata is then broadcasted into other devices and the other devices will fetch media from the server using the meta-data received. The downloaded file is converted as text file and placed on the 3D screen to be displayed at the user side.

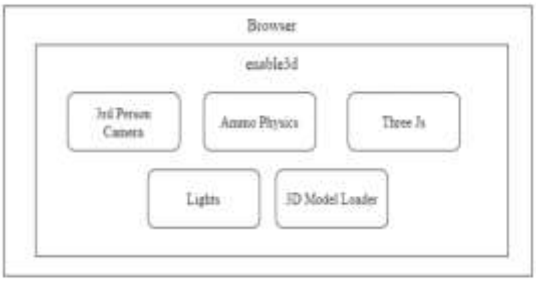


5.6 Summary

The project is designed to work on a browser so it runs on a client-server model. The Server is deployed in a network that is accessible from every client device. The server serves the static files for client. It contains a copy of the classroom world and its present state and it is the most authentic copy. Client renders the web application using the static files received from the server. A classroom instance is created on the client device. Events that are happening in the client are sent to the server using web-sockets.

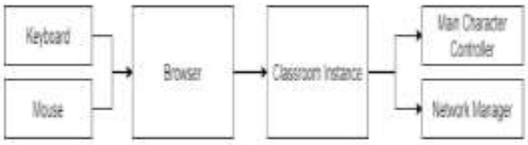
VI. IMPLEMENTATION

6.1 Client Platform



The Client platform is implemented using JavaScript on an HTML project, with CSS for styling it uses Enable-3D framework for project architecture. The Enable-3d is a standalone 3 dimensional framework on top of three-graphics. It allows to integrate the three-graphics package into your Phaser-3 games. When Enable3D is loaded the classroom environment is created using basic geometries and the wall and floor materials are loaded from the server. Inside the classroom a directional light as well as an ambient light is added in order to attain lights as well as shadows in the classroom. Using box colliders from Ammo Physics, the boundaries are set on walls. The base plan is also set on the floor using box colliders, and their gravity is set to zero. Then the chair model is loaded from the resources with its material and added to the classroom using a loop.

When the environment is ready, the character models and their animations are imported from the resources. The materials are dynamically generated. A character model and its animations are designed using 3D software like blender and stored in gITF format. Using the state manager in side character, the controller animations are applied to the model which can be triggered using different states. The 3D character model is added to the classroom and attached with a new Main Character Controller instance. A random colour is fetched from the server, and the client creates a dynamic material using the colour and applies it on the model. A virtual camera is added to the scene and attached with the character model as a third person camera which is also controlled by the Main Character Controller.

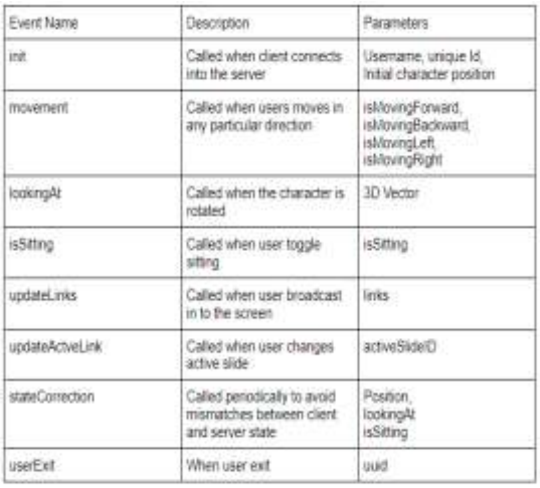


The classroom instance also starts accepting inputs from the browser on initiation. This inputs are mapped to the various events. The key 'W' is mapped to moveForward event the key 'C' is mapped to toggleSit event Mouse movement is mapped to rotateCamera event. These events are passed to both the Main Character Controller and Network Manager The server broadcasts these events to the other devices, and corresponding Network Managers transfers this to Sub Character Controllers.

Algorithm:
1. Start
2. Setup environment, lights, physics and camera and create MainScene
3. Import 3D models, materials and textures and add to MainScene
4. Request for voice communication ID
5. Input username from user
6. Handle browser input
7. Connect with server and spawn main character and main character controller
8. Add event listeners with server
9. When new user join, spawn character with character controller
10. Request animation frame from browser
11. Update object models
12. Repeat from step 10 till tab closes
13. Stop

6.2 Server Side

The Server accepts events from one device and broadcasts it into other clients. Server is implemented using Node JS with express JS framework. Socket.IO library is used to created durable web-socket connection between server and clients. Socket.IO port is opened on the server to which clients can request connections. Requests from valid domains are accepted and the connections are made. Then it waits for events sent from the clients.



Event Name	Description	Parameters
init	Called when client connects into the server	Username, unique Id, Initial character position
movement	Called when users moves in any particular direction	isMovingForward, isMovingBackward, isMovingLeft, isMovingRight
lookingAt	Called when the character is rotated	3D Vector
isSitting	Called when user toggle sitting	isSitting
updateLinks	Called when user broadcast in to the screen	links
updateActiveLink	Called when user changes active slide	activeSlideID
stateCorrection	Called periodically to avoid mismatches between client and server state	Position, lookingAt isSitting
userExit	When user exit	uuid

Using a switch statement, these events are passed into certain functions. The server has acopy of the classroom state. Which is regularly updated using the events. This state is passed tothe new clients when they join. Also clients periodically make requests to ensure the server andclients are in same state.

6.3 Voice Communication

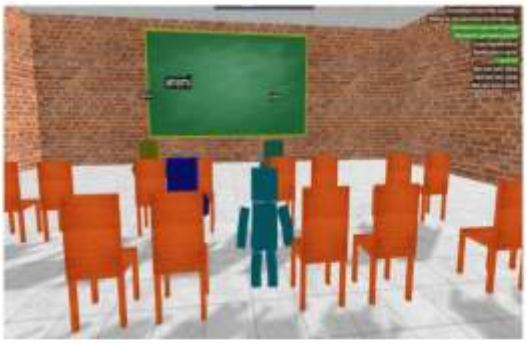
Peer JS library is used to handle WebRTC connections. This library will help to create a unique ID. Also using this unique ID it will help to create channels between clients. This is peer to peer channel hence no intermediate server is present. The peer JS unique ID is passed to the server which broadcasts it to other users. Peer JS updates the channel or create new route when intermediate nodes are broken or the network is changed.

Algorithm:
1. Start
2. Connect to server and fetch Peer ID
3. Wait for user to join
4. When new user joins fetch Peer ID of new user
5. Make WebRTC call using Peer ID, send stream from microphone to the user
6. Output the stream from user into a HTML Audio Element
7. When other user calls answer the call and send stream from microphone to the user
8. Output the stream from the user into a HTML Audio Element
9. Stop

6.4 Summary

The project is implemented in the browser. For that we have a client and a server side. The Client platform is implemented using JavaScript on an HTML project, with CSS for styling it uses Enable-3D framework for project architecture. The Server accepts events from one device and broadcasts it into other clients. Server is implemented using Node JS with express JS framework. Socket.IO library is used to created durable web-socket connection between server and clients. For voice communications we use WebRTC connections to create a Peer ID and stream the audio to HTML Audio Element.

VII. RESULTS

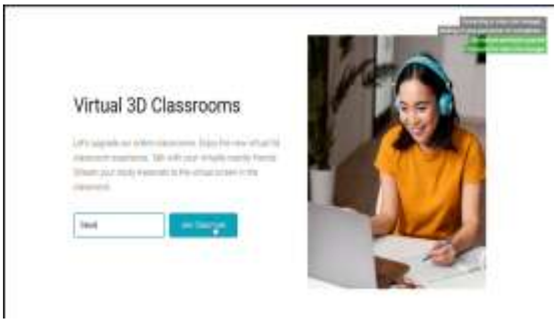


We have successfully developed, implemented and tested the 3D Virtual Classroom for conducting online classes. Following are the results we are able to concur. A browser that is able to run a Virtual 3D Classroom is created. Static files are generated andthe project is deployed into a digital ocean server droplet using pm2. A public domain is attached tothe current domain and the Users can use the platform through the link. The browser fetch the staticfiles from the server which consists HTML file, JavaScript bundle and media resources(Texturesand 3D Models). The web application files only has a total size of 12.7 MB so storage is not aproblem. A user with an average internet speed can download the project and start the applicationwithin 6-20 seconds.

The browser generates 3D frames till the tab gets closed. A logger is attached with the project for understanding the process and troubleshooting the bugs. Browsers give permission only to the microphone when HTTPS is enabled. So the data is secure using CloudFlare Proxy Using proximity chat function we can conduct group or private discussions.

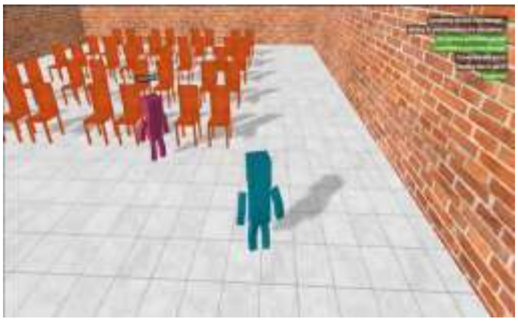
Below shows the first page the user sees when they enter the URL. Here we have option to authenticate the user.

The users enter the details and click ’Join Classroom’ to get spawned into the environment. Now after entering into the classroom the user can walk to a chair and perform the action to ’Sit’. The users could now see the study materials streamed to them on the board. Pressing the ’O’ key makes the materials appear in a zoomed in version for better clarity that encompass the entire screen.

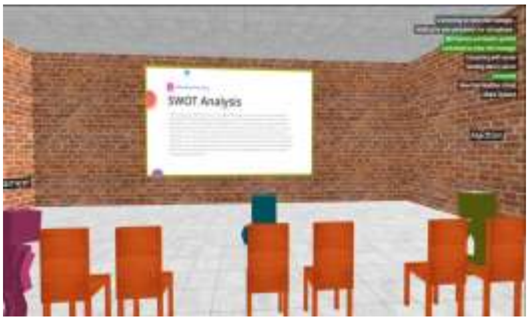


Virtual 3D Classrooms

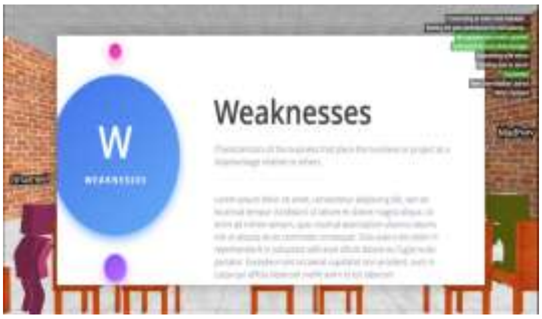
Login Page



Character Spawn



Character Attending Class



Enlarged View of Presentations

VIII. PERFORMANCE EVALUATION

8.1 Evaluation Metrics

Evaluation for media streaming was done on two main parameters that are critical to quality.

These were:-

1. Packet Loss

2. Bandwidth
3. Frames per Second

1. Packet Loss

Packet loss represents gaps in data transmitted over the network namely the internet in this case. WebRTC uses the User Datagram Protocol in the network layer. Video packets that are lost when being transmitted cannot be reconstructed at the receiver side. UDP being a connection less protocol has higher chance of losing packets compared to any other connection oriented protocol. For normal video streams simply retransmitting the packets works fine. But in a video conference system, the video being live retransmission will not work. As a result a high packet loss results in frozen video experience. Packet loss generally translates to a reduction in frame rate.

2. Bandwidth

There are basically two types of bandwidth upload bandwidth and download bandwidth. Upload bandwidth is required to publish the video of the user and download bandwidth is required to see the video stream of the other participants. When either of those become insufficient the video stream will be halted and only audio will be transmitted.

3. Frames per Second

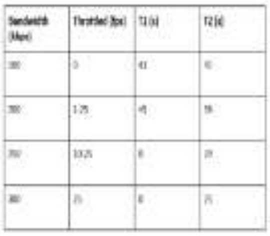
FPS is used to measure frame rate – the number of consecutive full-screen images that are displayed each second. It is a common specification used in video capture and playback and is also used to measure video game performance. In object detection mechanism fps shows the model‘s efficiency in detecting objects in multiple frames per second.

8.2 Performance Analysis

1. Game Rendering Analysis

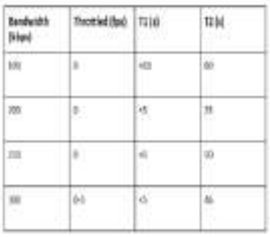
A comparison between WebRTC and Zoom was done to evaluate the above metrics. The measurement is done by first restricting the bandwidth at rates ranging from 100 kbps to 300 kbps. The framerates of both applications is then measured. The time T1 is the time taken by the video to resume streaming normally, at more than 1 frame per second after the bandwidth restriction is removed. The time measured by T2 is the time taken to get back to the full resolution after the bandwidth restriction is removed.

WebGL uses the latest technology introduced after HTML 5 Hence older browsers may not support the application. Also it may not be compatible with out of date hardwares. A research is made to estimate percentage of global internet users who can use this platform through their common browser. It is found that more than 90% of the global users can use this platform using their default browsers and without any extra additional support.



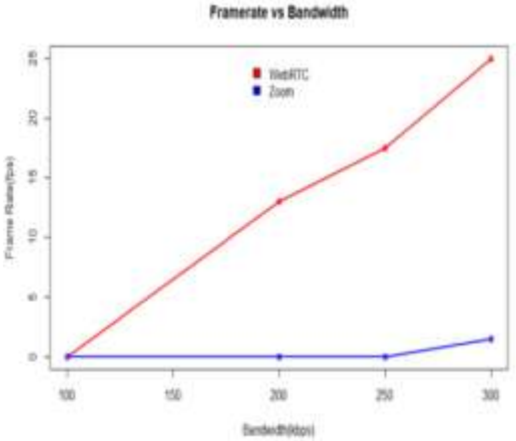
Bandwidth (kbps)	Throttled (fps)	T1 (s)	T2 (s)
100	0	43	45
200	1.75	45	58
250	13.25	8	29
300	25	8	25

WebRTC



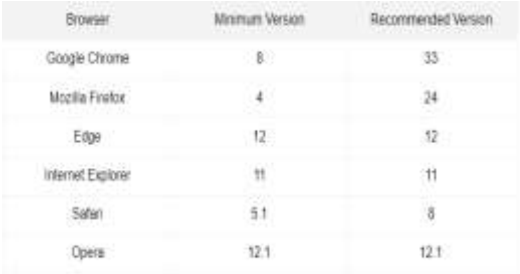
Bandwidth (kbps)	Throttled (fps)	T1 (s)	T2 (s)
100	0	403	60
200	0	45	38
250	0	40	90
300	0-1	<5	46

Zoom



Framerate vs Bandwidth

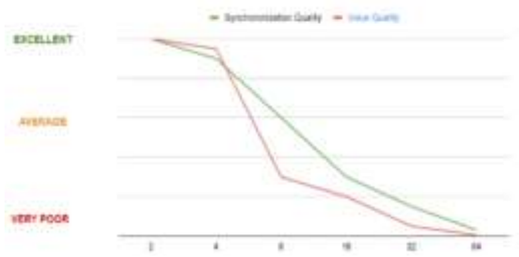
2. Browser Support



Browser	Minimum Version	Recommended Version
Google Chrome	8	33
Mozilla Firefox	4	24
Edge	12	12
Internet Explorer	11	11
Safari	5.1	8
Opera	12.1	12.1

3. Synchronization and Voice Communication Quality

The bandwidth usage will increase as number of users' increases. Thus when the number of users increases the synchronization and communication quality will also decrease. It is found that there is large quality drop after 8 users join. Latency is found when we synchronize with client devices. This leads to audio jittering. When number of users is 32, packet loss is very high and every changes are not reflected on the platform, and also audio communication is fully down. However rendering quality and system performance remain stable even though the number of users increases. The media streaming quality is not affected since it is directly fetched from server and it is a one-time process.



8.3 Summary

From the above results it can be deduced that the application using WebRTC can transmit media even at bandwidths lower than 300 kbps where Zoom had struggled. Though Zoom has a lower recovery time to transmit video when the bandwidth restriction at 100 kbps is removed, with all other restrictions at different bandwidths WebRTC shows a better recovery time. Also we can see that the application is able to run on wide range of browsers available in the internet and is compatible with most devices. As the number of users logged into the classroom increases there is found to be slight delay in synchronization. This problem also leads to audio problems, but has no effect on study material streaming.

IX. CONCLUSION

In this project, we try to introduce a new way of conducting online classes that far outshines the present one. The idea is t make learning sessions more interactive and immersive for the students. We were successful in creating this 3D environment as a browser application that is accessible by everyone. The application is deployed and connected to a public domain. It is found that 90% of the global users can directly access the platform through their internet browsers on laptop or desktop. Each user is spawned as a 3D model which acts as his/her virtual avatar in the classroom. This enables them to perform actions through animations within the system. Proximity chat function allows voice communication among users in 10 unit radius within the classroom environment. The application can be extended for mobile users after mapping touch inputs to actions. The scope of the project is not tied to classrooms itself it can be extended to other virtual gatherings and functions where users login remotely.

REFERENCES

[1] K. Elissa, M. Bacco, M. Catena, T. De Cola, A. Gotta and N. Tonellotto, ”Performance Analysis of WebRTC-Based Video Streaming Over Power Constrained Platforms,” 2018 IEEE Global Communications Conference (GLOBECOM), 2018

[2] ”Performance evaluation of WebRTC-based online consultation platform E. Alperay TARIM, H. Cumhur TEK˙IN

[3] Nayyef, ZinahAmer, Sarah Hussain, Zena. (2019). Peer to Peer Multimedia Real-Time Communication System based on WebRTC Technology. International Journal for the History of Engineering Technology. 2.9. 125-130.

[4] R. R. Chodorek, A. Chodorek, G. Rzym and K.Wajda, ”A Comparison of QoS Parameters of WebRTC Videoconference with Conference Bridge Placed in Private and Public Cloud,” 2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2017