RealTES: A Virtual Office Application Enhancing Productivity and Work-Life Balance in Remote Work Environments

Devu Balakrishnan[1]
Information Technology
Rajagiri School of Engineering and Technology
Kakkanad, Kerala

Aambal Grace CS[2]
Information Technology
Rajagiri School of Engineering and Technology
Kakkanad, Kerala

Harun Jamal Rasheed[3]
Information Technology
Rajagiri School of Engineering and Technology
Kakkanad, Kerala

Kuruvilla Biju Cheruvallil[4]
Information Technology
Rajagiri School of Engineering and Technology
Kakkanad, Kerala

Dr. Ranju S Kartha[5]
Asst. Professor
Information Technology Rajagiri School of Engineering and Technology
Kakkanad, Kerala

Abstract:- The rise of remote work has necessitated the devel- opment of innovative solutions that bridge the gap between physical and virtual workspaces. RealTES (Real Time Environ- mental simulation) is a virtual office application that seeks to create a simulated in-office working environment by providing customizable workspaces, private meeting rooms, and deep-focus spaces to enhance productivity. The application is designed to help users maintain work-life balance by providing tools to enhance productivity without compromising on one's health and well-being. RealTES uses a two-dimensional workspace built with the aid of a Game Engine called Unity , real-time audio video communication platform Agora.io [6], and it aims to be browser-first, so that users do not need to download additional software. Users can navigate the virtual office using keyboard or mouse inputs and can connect seamlessly with other users in the same room via audio and video channels using the WebRTC technology. A secondary application system in RealTES uses computer vision to determine a user's alertness levels and workload by analyzing facial cues and advises the user to take breaks accordingly. The application also includes integrated reminders, alerts, and timers to help users manage their deadlines. RealTES seeks to improve digital interactions by incorporating a social aspect into the communication process, creating a sense of presence within an organization. However, it is important to note that there are some limitations to this approach, such as the difficulty in conveying physical sensations beyond audio and visual stimulation and the potential for network faults that can disrupt the online application.

Keywords:- *Virtual Office Application, Remote Work, Productivity, Unity, Work-Life Balance, Computer Vision, Webrtc, Digital Inter- Actions.*

I. INTRODUCTION

The onset of the pandemic has made remote work and learning incredibly repetitive and had a negative impact on both the mental health of employees and students. The majority of companies are currently getting rid of their physical offices and switching entirely to work from home arrangements since they find this to be more lucrative financially. Incorporation of platforms such as Google meet, Zoom, Microsoft Teams into the daily workflows have proven to be monotonous and its impact on one's mental and physical wellbeing as well as isolating for the user. The proposed solution implements a close to life environment that reflects the typical day to day activities of offline offices. The objective is to build a 2- D virtual workplace or classroom where users may interact using their digital personas to forge connections with their coworkers or students that a standard Zoom Video Call cannot provide. The ability to roam around a virtual workplace or classroom and communicate with other users through real-time audio video and chat. Every space in the office functions as a private speech channel that can only be heard inside that space. Meeting spaces can be secured with locks to give higher-ranking staff privacy and to simulate an actual office environment. Each user may have a private office that they may personalize. This goes one step beyond existing systems by providing tools/aids that help boost user productivity, such as Post it notes, Whiteboards, Timers, Fatigue detection using openCV , alerts, notification and deadline reminders and more. RealTES offers a breath of fresh air in terms of bringing out what we all love about face to face conversations. The productivity suite ,an add on feature helps RealTES go one step further when compared to similar offerings as it helps address falling productivity and emphasizes on keeping users healthy and happy. RealTES has the capability to be the preferred choice for remote work or learning and believe that its true potential lies beyond the scope that it has been currently defined.

The remainder of the paper is structured as follows. Literature reviews are addressed in Section II. The proposed system methodology/architecture in Section III. This section also includes a presentation of the numerous features and functionalities included within each module. The project's results are discussed in Section IV The conclusion of this paper and its future scope are provided in Section VI.

II. LITERATURE SURVEY

The following provides an overview of the six papers that were analysed, which mostly focused on topics like WebRTC, game development, VOIP, network sockets, and facial expression recognition.

George and Stefan [1] summarized the history of Real Time Communication systems and the features that WebRTC provides such as streaming content, obtaining network information, dealing with NAT and firewall etc, alongside a general high level architecture. The protocols used such as ICE and STUN are also discussed, through its interaction over the O.S.I network layer, and the reason to use UDP instead of TCP. The paper implements a thermal WebRTC application using Scaledrone, which is a push messaging service; WebRTC, for the website server to send the browser IDs; HTML for the frontend, with the core of the system being a JavaScript API server backend.

According to a study [2] on a WebRTC-based video conferencing system's effectiveness , Baseline Experiments, Cross Traffic, Multi-Party Topology, Video Codecs, Mobile Performance, and Real Wireless Networks parameters were evaluated .The User Datagram Protocol, which is the foundation of WebRTC, aids in realising the demands for high throughput and low latency, while the GCC algorithm, which uses adaptive threshold, handles congestion control.This work develops performance evaluation benchmarks, and the overall performance was comparable to a TCP flow. In terms of cross traffic, a UDP stream with GCC and adaptive threshold enabled was given a higher priority than TCP.In terms of video codecs, the WebRTC system utilized the VP8 codecThe most recent version.However, VP9 is more effective (lower bitrate due to im- proved compression), but also requires more processing power. With the exception of iOS devices' lack of WebRTC, the performance on mobile devices and in actual wireless networks was deemed sufficient.

A different study uses Autodesk 3D Max to construct a multi-user virtual learning environment that mimics a historical museum [3] . Students walk around the simulation as virtual avatars with gestures, implemented using Avatar Studio 2.0. An interaction server is built using Java for enabling users to walk through the world and to communicate with the different avatars by using different interactive gestures. The user executes a Client browser application that is fundamentally in charge of the visualization and communicating with the server. The server knows the positions of all avatars. This way students can walk around and share emotions with interactive avatar gestures, making learning more realistic and complete. fast prototyping tool with flexible behaviors, stunning visual effects and capabilities.

The authors of the paper [5] examines the pros and cons of work in a virtual environment, along with the incorporation of multimedia elements such as audio & video chat, representation of the office as an 8-bit video game, etc.The Project is built by creation of UI in Unity Game Engine. The authors also try to build the office in a virtual environment. Organizations will be virtually able to see the employees working around in an 8-bit virtual environment. After completing the UI, in-game voice is given to the users present in the virtual environment with the help of Agora.io voice call SDK. It provides natural audio reproduction and ensures clear sound. It also includes local background music and sound effects along with voice for a more immersive user experience. It also provides an automatic echo cancellation. After completing in-game interactions, the server is hosted on Heroku. Heroku is basically a container-based cloud platform which can be used to deploy, manage and scale modern apps. This platform is distinguished, pliable and simple to use offering the users/developers the easiest path to get their apps to market.

Researchers proposed a system employed using Computer Vision to analyse blink rate, eye closure, yawning etc to effectively and quickly identify the drowsiness of a driver during driving the vehicle and alert the driver accordingly . In the work [6], pre-existing feature stores for facial landmark detection is implemented to identify the state of drowsiness and fatigue. These predefined landmarks help in shape prediction to clearly identify the various regions of the face like eyebrows, eye, mouth region etc and changes in parameters report various expressions of the person. Camera monitors and captures to extract frames. Each extracted frame is analyzed to study the pattern of facial features. There are threshold values for eye aspect ratio and mouth aspect ratio, which upon exceeding, will record as a blink and a yawn. Blink and yawn rates are analyzed to determine whether the driver is drowsy and an alert is triggered.

III. PROPOSED SYSTEM METHODOLOGY

Construct 2 [4] is a very simple commercial application for the creation and design of two dimension games similar to Unity. In Construct 2, the classic coding is replaced with dragging and dropping game objects. Games developed in Construct 2 are implemented in HTML5 and Javascript. In this game engine, every layout has its event sheet in which all the steps related to the addition of the events, the addition of actions, and other events. This eliminates the need for complicated scripting in programming languages with fixed syntax, which makes it an easy tool to learn. It allows a creator to define objects, its types, give objects behavior and effects and group them into families. Instant preview and iterative development makes it a.

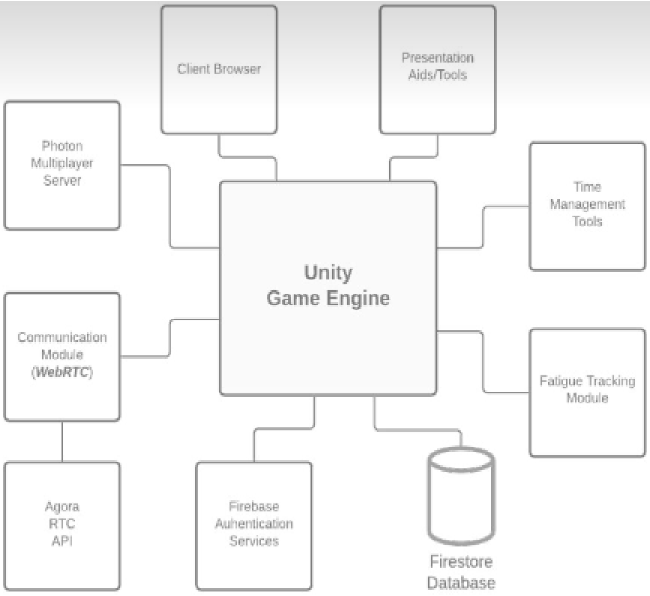


Fig 1 System Architecture

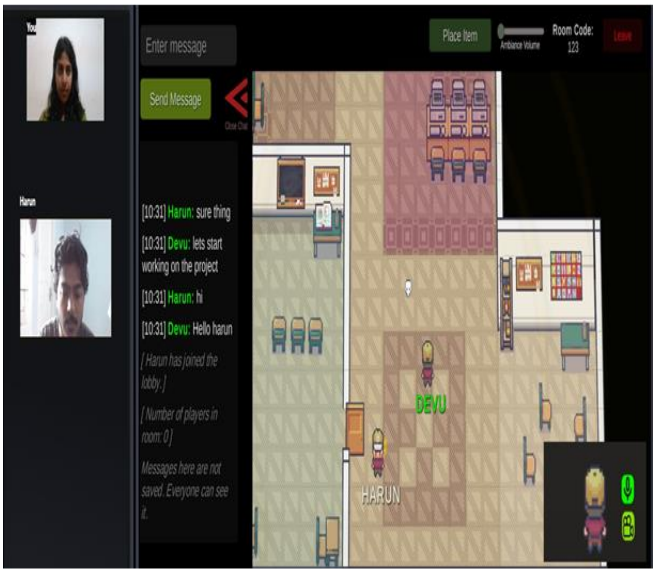
As the entire ecosystem, seen in figure 1, interacts with this single component, which is the unity game engine,it is regarded as the brains of the RealTES program. The player movement, animations, noises, and physics interactions are abstracted away from the application's internal workings by using WebGL to operate the game instance. Although you interface with the Unity engine through a C# wrapper, it is internally developed with native C/C++. Unity also has a sophisticated 2D environment renderer and permits the import of sprites for use in 2D games.Unity uses JSLib to communicate with browser scripting.

Additionally, it communicates with the Photon Multiplayer server to establish concurrency among all clients and the Agora audio- video engagement platform to allow live audio-video communication among several users. To interface with browser scripting, Unity makes use of JSLib. It serves as a Unity plugin that allows you to call javascript functions. A .jslib compiles into library files that are linked to unity during build. RealTes uses JSLib plugins to communicate with Firestore and the Agora communication module. Firestore authentication services keep a persistent copy of the user and office room data in the background.

A secondary application system in the virtual office en- vironment uses computer vision technology to track and assess the user's level of attention and effort. To gauge the user's degree of engagement and exhaustion, this system looks at facial clues such eye movement, facial expressions, and rate of yawning.

A. *Proximity Chat*

Users dynamically communicate with each other when they are standing next to each other in the virtual space as in fig 2. People outside of the range of this circle are unable to listen in to the conversation. This helps enforce a sense of space within the world, helping sell the notion of a real office space. People can have multiple conversations within different parts of the office without disrupting a single channel like other video conferencing apps. This proximity feature is enabled so as to give a feeling of virtual connection between different individuals when they come physically in contact with one another.



[10:31] Harun: sure thing

[10:31] Devu: lets start working on the project

[10:31] Harun: hi

[10:31] Devu: Hello harun

[Harun has joined the lobby.]

[Number of players in room: 0]

Messages here are not saved. Everyone can see it.

Fig 2 Proximity Chat

Proximity Chat is implemented using a complex system of calls from between the Unity app and the user's web browser. Initially Unity finds all users within a threshold distance of your player. It then calls a subscribe function for each of these users. The connect function calls a Plugin within the JSLib module which translates Unity's C# call to Javascript. This matching method is invoked on the browser which "subscribes" the user to the nearby users.

B. *Private Space*

This system allows users to have added privacy in the sense that only people within the room marked as a private space can listen in on the conversation. They can physically limit the other users from entering by locking the virtual door. When inside a Private space, the Proximity Chat feature is overridden and this system takes control. A Private Space can be a small cubicle or even a large auditorium. This feature uses a lot of complex Remote Procedure Calls (RPCs) and data structures. A stack stores the users who are within a particular private space, and these stacks are kept synchronised throughout the network using RPCs. All of these RPC's are buffered within the Master Client of the server. This buffer is shared with each new user joining the server and hence privacy is ensured.

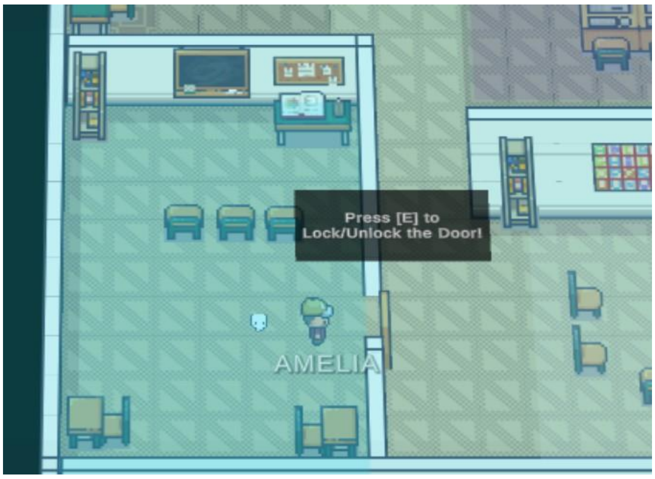


Fig 3 Private Space

C. *World Decoration and Path Finding*

World decoration: Users can place custom objects within the gamespace to personalise their office and surroundings. Players can click on these items and see who placed it, and the owner can remove them at anytime. A Firebase store call fetches data, when each item is placed, storing the metadata of the item placed. This data is fetched for each new user so that the items placed persist through all clients and all future sessions.

Path finding: The system also allows the user to traverse the gamespace using a smart pathfinding algorithm that automatically moves the player. The player takes the shortest path to the coordinate where the mouse clicked (fig 4).

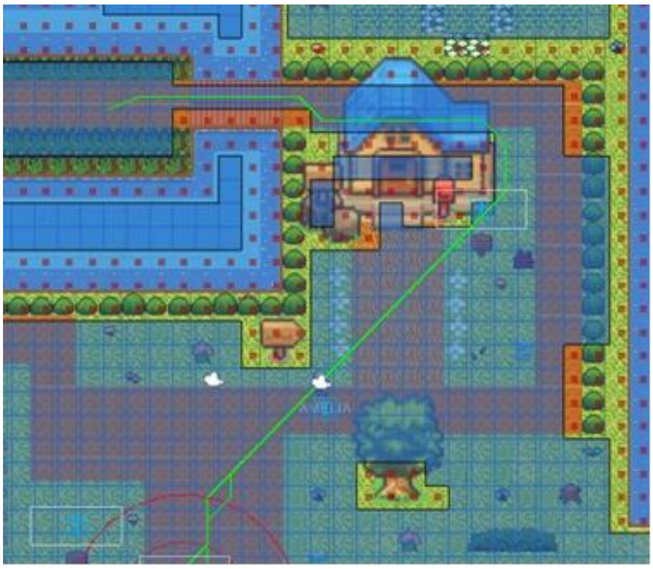


Fig 4 Path Finder Using A* Algorithm

This feature uses the A* algorithm that scans the games- pace and generates a matrix of cubes. It tests for solid colliders in the world and marks them as impassable terrain. It then uses the A* shortest path algorithm to compute the shortest path between the player's current position and the mouse position when clicked. Finally the path vector array is fed to the custom Player Movement controller script to traverse the world.

D. *Village Game Space*

A fictional map created to demonstrate the capabilities of the office space shown in fig 5. Uses tilemaps and layers to create an illusion of an isometric projection. Has three houses with their own interiors Also has a variety of different biomes, which is based on the Clipperton Island tragedy with many small easter eggs to find related to the Incident of 1711.

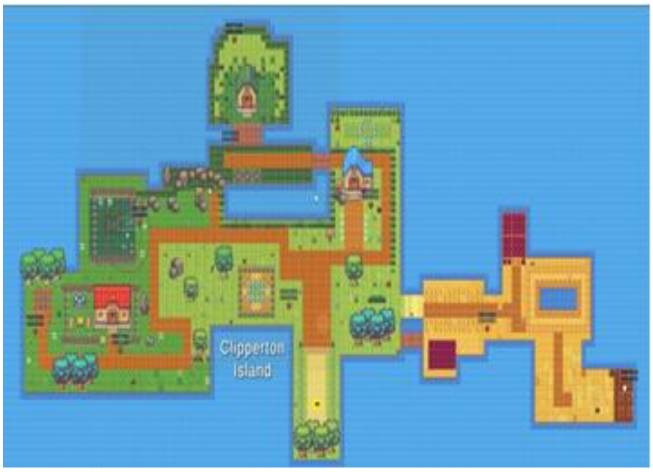


Fig 5 Village Game Space

E. *Photon Multiplayer Server*

Photon Multiplayer Server is a cutting-edge platform designed for hosting and managing multiplayer games. With its advanced features and high-performance capabilities, Photon revo- lutionizes the multiplayer gaming experience. Photon provides the following functionalities out of the bag:

- *Server Creation*
- *Load Balancing*
- *Lobby System*
- *Room Object Instantiation*
- *Player Position Synchronisation Among Users.*
- *Player Animation Synchronisation Among Users.*
- *Remote Procedure Call Wrapper For Our Functions.*
- *Server Member Hierarchy Manager.*

This module is the main server of the application, that manages the Rooms and server node instances. It ensures that all clients connected to the same room are in sync with each other. It seamlessly abstracts away the creation of rooms and the joining of rooms by users.It automatically load balances and allocates game server instances. Essentially, its the environment on top of which the game instance runs on in the network.However, Photon only provides the base logic for the game server. The game logic was built on top of this solid base using the many different methods and RPC's provided by the framework. Game Objects are synchronised among all the clients using Photon's "View" module that seamlessly synchronises the location and physical existence. However, all other attributes are manually synchronised using RP Calls and Streams.

F. Agora

Audio/Video communication between users is handled over WebRTC. WebRTC helps abstract the underlying UDP mecha- nism while also handling congestion control using the Google Con- gestion Control (GCC) algorithm. Agora is a Communications as a Service platform that provides WebRTC technology. It provides APIs for establishing RTC connections ,control communication and live broadcast services. Users communicate with each other using Agora API which is integrated into RealTES. The Agora API helps abstract the following RTC Functionalities:Client creation and initialization, track generation, publish and subscribe operations, management of remote users, screen sharing, and event listeners.

RealTES makes use of WebRTC (Real Time Communica- tion) for facilitating communication between users, the next genera- tion Agora platform (Agora SDK-NG v4.0). WebRTC is abstracted by Agora Real-Time Engagement platform for devs to integrate voice & video chat, interactive live streaming and messaging through the integration of the Agora SDK. It follows a typical Publish-Subscribe (pub-sub) model, Offers superior performance through the usage of a SD-RTN (Software Defined Real-time Network).

A JavaScript (JS) based Web-App which calls the neces- sary APIs to perform the basic operations in a typical video-chat app.HTML and CSS are used to construct user-facing elements, and the Unity-specific HTML elements are merged. There are three ways to integrate SDKs: the Agora website, NPM (Node Package Manager), and CDN (Content Delivery Network). Due to its low latency, scalability, redundancy, dependability, and bandwidth opti- mization, CDN was chosen. As the only channel for the gamespace, two tracks—Audio and Video—per user were chosen, and they are distributed. Here Users have the option of subscribing to one or both of the other users' tracks.

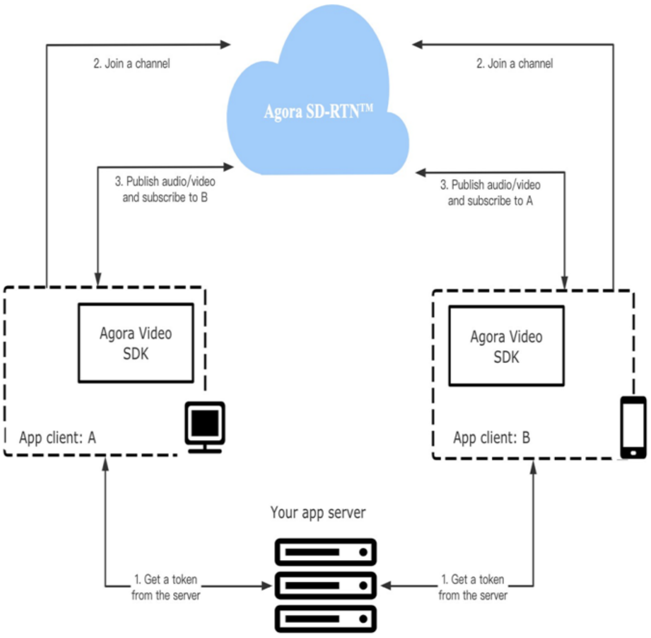

2. Join a channel
2. Join a channel
Agora SD-RTN™
3. Publish audio/video and subscribe to B
3. Publish audio/video and subscribe to A
Agora Video SDK
Agora Video SDK
App client: A
App client: B
Your app server
1. Get a token from the server
1. Get a token from the server

Fig 6 Basic Video Call Architecture in Agora

- *Proximity-based Subscription for Spatial Audio:*

A deviation from the conventional subscription flow to play on the on- user published event listener. Functions for Subscribing and Unsub- scribing are divided - subscribeWhenNear() and unsubscribeWhen- Far().The user ID or player id is sent as a parameter.Only when called by Unity via jslib is subscription or unsubscription performed.

G. Productivity Suite

- *Presentation Aids and Tools –*

Are used to help the users to communicate effectively. Presentations are one of the most unavoidable parts of an organization setup. It will make the process of idea pitching way easier. The aids offered by the platform include whiteboard and file sharing features in addition to screen sharing via RTC. Agora allows a client to publish only one video based track as a result of which a custom logic to swap between the camera and screen feeds was implemented.

- *Fatigue Tracking System –*

The independent component of the Productivity suite is implemented in Python and operates separately from the core RealTES Platform. Its main functionality involves utilizing Computer Vision techniques to monitor a user's face for signs of fatigue. The component focuses on two specific metrics: Eye Aspect Ratio (EAR) and Mouth Opening Ratio (MOR). By continuously analyzing the user's facial features, such as the position and movement of their eyes and mouth, the component can determine if the user is showing any signs of tiredness. The Eye Aspect Ratio (EAR) is a measure of how open the eyes are, while the Mouth Opening Ratio (MOR) indicates the degree to which the user's mouth is open. These metrics serve as indicators of fatigue levels [6]. The system tells the user to take breaks and rest when it notices signs of exhaustion or stress.It includes timers, notifications, and reminders to help the user remember crucial deadlines. These tools offer timely notifications to help users stay organised and complete their tasks within the allotted times. When the component detects signs of tiredness based on the EAR and MOR values, it will notify the user by sending an alert message. This notification acts as a reminder for the user to take a break. Users can run it as an executable file, that is they can utilize its functionality without having to install Python.

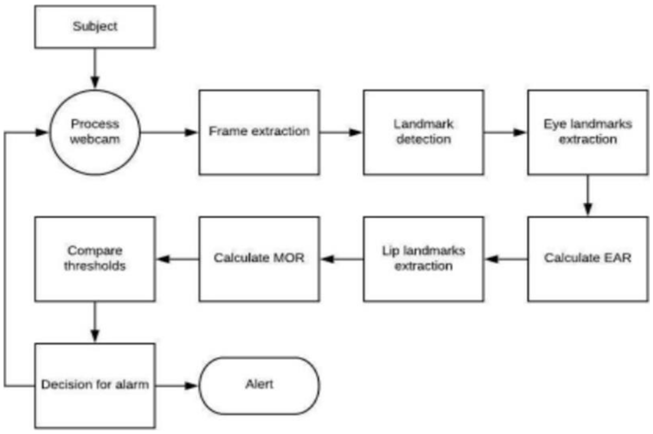


Fig 7 Fatigue Detection System

ISSN No:-2456-2165

➢ *Time Management Tools* –

These tools abstracts the set of services and tools on the platform that are managed by the Calendar API. These tools include meeting scheduling, deadline reminders and new assignment alerts. Such tools help utilize the available time in the best possible way and reduce clashes which in turn boosts productivity.

- *Unity - Photon*
- *Unity - Firestore (via jslib)*
- *Unity - Agora (via jslib)*

In system testing all the modules are integrated in order of execution. Testing is carried out over the whole system. All the events starting with joining a room, walking in the gamespace, audio and video communication etc till the user leaving the gamespace was examined. A few kinks due to network related issues like ghost players, track publishing delay etc were noticed and sufficient guards were implemented to compensate for these out of hand issues in the best way possible. The overall event flow was re-examined and the results were deemed satisfactory.

A real-time multi user chat and video calling web applica- tion was implemented with a gamified representation of a real world environment in a 2-D space which includes the following features:

- *Private Spaces*
- *Path Finding Algorithm*
- *Multiple Gamespaces*
- *Gamespace customization*
- *Spatial Audio/Video*
- *Screen Sharing*
- *Fatigue Tracking*
- *Clock, reminders and alerts.*

IV. RESULT AND DISCUSSION

Testing this project, primarily looked at how the various components in each module behaved for multiple scenarios and checked if the overall behavior was consistent with the expectations during the various phases of development.

➢ *Testing Methodologies*

Two common software testing methodologies are white- box testing and black-box testing. White-box testing involves testing the internal structures or workings of a program, using an internal perspective of the system and programming skills to design test cases. On the other hand, black-box testing treats the software as a black-box, examining functionality without any knowledge of internal implementation. Testers are only aware of what the software is supposed to do, not how it does it.The testing methods applied were: Unit testing , Integration testing and System testing.

Unit testing involves designing test cases to validate that the internal program logic is functioning properly, and the inputs provided produce valid output. All decision branches and internal code flow are validated. All individual modules are tested separately. It ensures that each module performs accurately to the specification and has well defined inputs and outputs

In Integration Testing, different modules were combined and tested to see if the modules interact properly and produce the correct output. Tests were performed after the integration of the following modules.

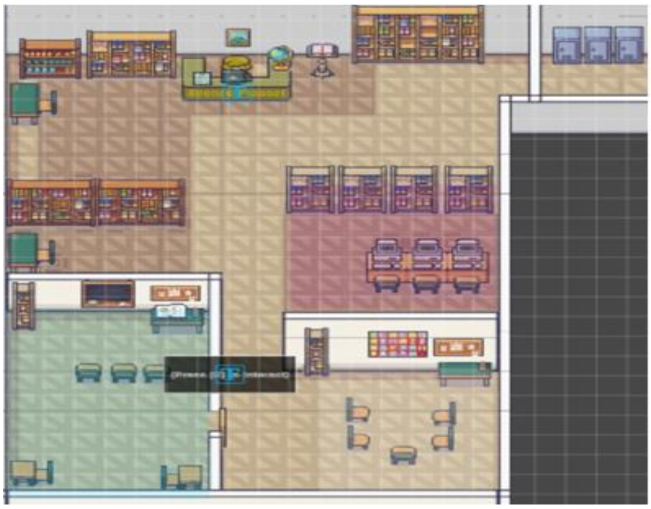


Fig 8 The Office Space

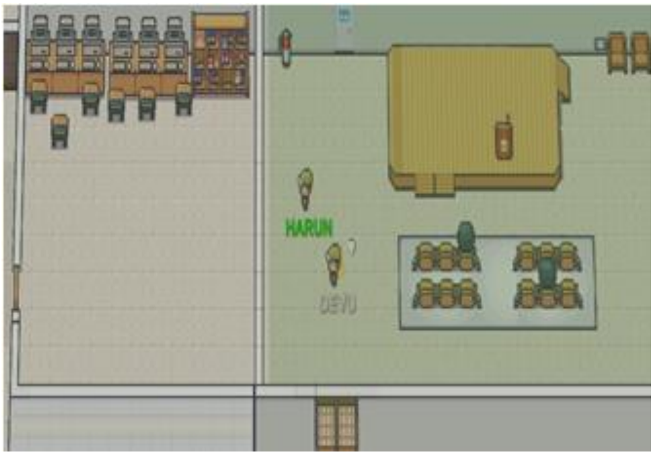
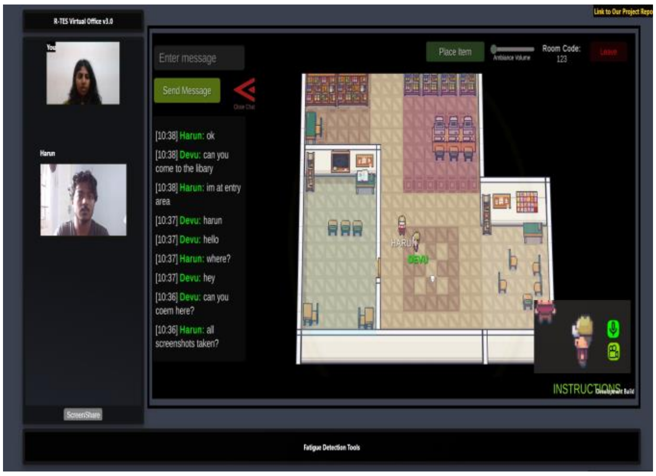


Fig 9 Conference Hall



[10:38] Harun: ok
[10:38] Devu: can you come to the libary
[10:38] Harun: im at entry area
[10:37] Devu: harun
[10:37] Devu: hello
[10:37] Harun: where?
[10:37] Devu: hey
[10:36] Devu: can you coem here?
[10:36] Harun: all screenshots taken?

Fig 10 Real-TES Application

V. CONCLUSION AND FUTURE SCOPE

The RealTES platform would likely be a popular choice for many individuals if there were to be another situation in the future where people were required to work or learn virtually. With its advanced features and capabilities, RealTES is well-equipped to provide a seamless virtual experience for users, making it an ideal solution for remote work and learning needs. Whether working remotely from home, or participating in virtual classes, RealTES offers a dependable platform that can be relied upon in these types of scenarios.The suggested method incorporates a real-world setting that mimics the regular day-to-day operations of offline organisations. It most closely resembles the core elements of a realistic human discourse and best captures the essence of a setting found in the real world. By avoiding tedious administrative activities and working with a skilled team, it helps to lower overhead costs, reduce risk, increase credibility, and promote productivity. You can then devote all of your time to the task that only you are capable of performing. These spaces give the ability of employees to collaborate with one another throughout the entire workday.It also helps users improve their productivity through the usage of time management and fatigue tracking tools.

In future work ,the following represents some of the areas that are worth exploring that could aid in the commercial use of RealTES.Allow users to create their own custom avatars instead of having to choose from the presets, allow for the gamespace to be modified, with additional rooms and areas as per the user's choosing, adding more features like post-it-notes, whiteboards etc within the gamespace, setting up a token server for Agora so that features like host and co-host can be implemented, creating an all in one executable that can run the Fatigue Tracking system on Mac/Linux devices.

REFERENCES

[1]. Suciu, George; Stefanescu, Stefan; Beceanu, Cristian; Ceaparu, Marian (2020). [IEEE 2020 Global Internet of Things Summit (GIoTS) - Dublin, Ireland (2020.6.3-2020.6.3)] 2020 Global Internet of Things Summit (GIoTS) - WebRTC role in real-time communication and video conferencing. , (), 1–6. doi:10.1109/GIOTS49054.2020.9119656

[2]. Jansen, Bart; Goodwin, Timothy; Gupta, Varun; Kuipers, Fernando; Zussman, Gil (2018).Performance Evaluation of WebRTC-based Video Conferencing. ACM SIGMETRICS Performance Evaluation Review, 45(2), 56–68

[3]. Mu, Bo; Yang, YuHui; Zhang, JianPing (2009). [IEEE 2009 International Conference on Information Technology and Computer Science (ITCS 2009) - Kiev, Ukraine (2009.07.25-2009.07.26)] 2009 International Con- ference on Information Technology and Computer Science- Implementa- tion of the Interactive Gestures of Virtual Avatar Based on a Multi-user Virtual Learning Environment. , (), 613–617.doi:10.1109/itcs.2009.134

[4]. Sag, Antonio; Orehovacki, Tihomir (2019). [IEEE 2019 42nd In- ternational Convention on Information and Communication Tech- nology, Electronics and Microelectronics (MIPRO) -Opatija, Croatia (2019.5.20 2019.5.24)] 2019 42nd International Convention on Infor- mation and Communication Technology, Electronics and Microelec- tronics (MIPRO) - Development of 2D Game with Construct 2. , (), 1647–1652.doi:10.23919/mipro.2019.8756702

[5]. C. P. Mallikarjuna Gowda, A. Srivastava, S. Chakraborty, A. Ghosh and H. Raj, "Virtual Office Prototype: A Unity-2D work Simu- lation," 2021 International Conference on Technological Advance- ments and Innovations (ICTAI), 2021, pp. 150-155, doi: 10.1109/IC- TAI53825.2021.9673456.

[6]. Garg, Hitendra (2020). [IEEE 2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC) Mathura, Uttar Pradesh, India (2020.2.28-2020.2.29)] 2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC) - Drowsiness Detection of a Driver using Conventional Computer Vision Application. , (), 50–53. doi:10.1109/PARC49193.2020.236556