**UNICONNECT: SISTEM DE VIDEOCONFERINȚĂ**

**Candidat: Petru-Lucian, MOCRIȘ**

**Coordonator științific: Ș.L. Dr. Ing. Valer BOCAN**

Sesiunea: Iunie 2024

**Cuprins**

[**1.** **INTRODUCERE** 5](#_Toc169210608)

[**1.1** **Context** 5](#_Toc169210609)

[**1.2** **Stadiul actual al problemei în România** 5](#_Toc169210610)

[**1.3** **Obiective** 6](#_Toc169210611)

[**2.** **Prezentarea aplicației UniConnect** 7](#_Toc169210612)

[**2.2** **Arhitectura sistemului** 7](#_Toc169210613)

[**2.3** **Cerințe software** 8](#_Toc169210614)

[**2.4** **Diagrama Use-Case a aplicației UniConnect** 9](#_Toc169210615)

[**3.** **Fundamentele teoretice** 11](#_Toc169210616)

[**3.1** **Limbajele de programare HTML și CSS** 11](#_Toc169210617)

[**3.2** **Limbajul de programare JavaScript** 11](#_Toc169210618)

[**3.3** **Tehnologia WebRTC** 12](#_Toc169210619)

[**4.** **Implementarea aplicației UniConnect** 15](#_Toc169210620)

[**4.1** **Back-end-ul aplicației** 15](#_Toc169210621)

[**BIBLIOGRAFIE** 17](#_Toc169210622)

**Lista figurilor**

[Figură 1 Google Meet[5] 6](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257448)

[Figură 2 Microsoft Teams[4] 6](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257449)

[Figură 3 ZOOM[3] 6](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257450)

[Figură 4 Arhitectura de sistem. Sistemul conține aplicația în sine si cele 2 servere de tip STUN. Server-ul principal este server-ul STUN 1 (Session Traversal Utilities for NAT[8]). În cazul în care server-ul STUN 1 nu comunică, atunci se trece pe server-ul STUN 2. De la server merge mai departe la ceilalți participanți. 8](#_Toc169257451)

[Figură 5 Diagrama Use-Case 9](#_Toc169257452)

[Figură 6 Web API[13] 12](#_Toc169257453)

[Figură 7 WebRTC[18] 13](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257454)

[Figură 8 Conexiunea dintre 2 persoane în WebRTC[20] 13](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257455)

[Figură 9 Arhitectura unui server WebRTC[21] 14](#_Toc169257456)

[Figură 10 Agora RTM[23] 15](#_Toc169257457)

[Figură 11 Mesajele trimise în consolă când o cameră este creată iar un utilizator a intrat. 19](#_Toc169257458)

[Figură 12 Un singur utilizator prezent în cameră 19](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257459)

[Figură 13 Mai mult de un utilizator prezent în cameră 20](file:///D:\Licenta%20final\Mocriș_Petru-Lucian_Documentație_INFOZI_Licență.docx#_Toc169257460)

**REZUMAT**

Aplicația a luat naștere din ideea de a satisface nevoia studențiilor atunci când intervine problema de distanță. Cu toții știm de existența aplicației Zoom, care din păcate poate fi folosită la capacitate maximă doar în schimbul unui abonament plătit. Cu astfel de probleme m-am confruntat și eu în trecut iar din această cauză, aplicația UniConnect este concepută să satisfacă nevoile studențiilor fără a fi nevoiți să plătească o sumă de bani.

# **INTRODUCERE**

## **Context**

Un sistem de videoconferință este o tehnologie modernă de telecomunicație, prin intermediul căreia mai multe persoane se pot conecta într-un singur loc de la distanță. *[1]*

La nivel global, cel puțin o persoană pe zi are de-a face cu o ședință, ori un training, curs sau un simplu apel în mediul online. Pentru aceste contexte există și sunt folosite diverse aplicații, care de cele mai multe ori te obligă să plătești un abonament lunar pentru a beneficia de toate implementăriile lor. Sistemele de videoconferință sunt cel mai des utilizate în educație și mediul de afaceri, dar, uneori sunt folosite și pentru a vorbi cu cei dragi care se află la distanță.

Pentru a te putea folosi de aceste aplicații ai nevoie de o cameră web, un microfon, o pereche de căști și un calculator/laptop sau un telefon/tabletă. Acest tip aplicații este foarte benefic deoarece te scutește de timp pierdut și costuri mari, fiind mai ușor și mai accesibil să vorbești sau să te vezi cu o persoană doar printr-un simplu click, decât să mergi până la aceasta, mai ales dacă se află în alt oraș sau în altă țară.

## **Stadiul actual al problemei în România**

Luând în calcul problemele actuale din acest domeniu, majoritatea problemelor au apărut în domeniul educației. Când țara noastră a fost lovită de pandemie, toate școliile și universitățiile au fost închise iar majoritatea au fost luate pe nepregătite. La început, mulți elevi sau studenți au luat asta ca pe o vacanță până s-au găsit soluții la toate problemele. Ulterior s-au găsit soluții precum aplicațiile Zoom, Microsoft Teams sau Google Meet, dar în cazul facultățiilor unde vorbim de sute de studenți, aceste aplicații nu au fost cele mai eficiente deoarece fără un abonament plătit în cazul aplicației Zoom, timpul unei întâlniri este de maxim 40 de minute cu un număr limită de 100 de persoane. Din această cauză, mulți studenți nu aveau loc sau la fiecare 40 de minute erau nevoiți să creeze o nouă înâlnire.

**Aplicația ZOOM** (Figura 1) este o platformă care oferă o interfață prietenoasă utilizatorilor și este cunoscută pentru că este ușor de folosit dar și pentru funcțiile sale robuste. Zoom dispune de funcții precum mesagerie, partajarea ecranului sau înregistrarea apelului.*[2]* Această aplicație fiind dezvoltată de o întreagă echipă.

**Aplicația Microsoft Teams** (Figura 2) este o aplicație care face parte din pachetul Microsoft 365 și este cel mai des utilizată de companii. Pe partea de funcții, Microsoft Teams dispune de gestionarea dispozitivelor, suport pentru ecran dublu, o tablă pe care se poate scrie și de detectarea proximității. *[2]*

**Aplicația Google Meet** (Figura 3) este o aplicație dezvoltată de gigantul tehnologic Google . Google Meet este o soluție de colaborare securizată care are ca și funcții sincronizarea cu calendarul Google, reprogramarea ședințelor sau creearea ușoară a unei întâlniri. *[2]*

A logo of a video camera

Description automatically generatedA blue and white logo

Description automatically generatedA group of purple squares with white text

Description automatically generated

Figură 1 Google Meet[5]

Figură 2 Microsoft Teams[4]

Figură 3 ZOOM[3]

## **Obiective**

Obiectivul aplicației UniConnect este acela de a implementa toate funcțiile necesare domeniului educațional și de a beneficia utilizatorii de ele fără a fi nevoiți să plătească.

În final, se dorește ca aplicația UniConnect să fie una stabilă și să satisfacă toate cerințele utilizatoriilor.

# **Prezentarea aplicației UniConnect**

* 1. **Cerințe de sistem**

Cerințele de sistem sunt acele specificații care trebuie îndeplinite pentru ca un program să ruleze eficient. Cerințele de sistem pot fi de tip Software sau Hardware. Cerințele de sistem se împart în două categorii: Cerințe de sistem minime și Cerințe de sistem recomandate.*[6]*

Cerințele de sistem sunt importante mai ales pentru echipa de dezvoltare deoarece pe baza acestora se pot definii cazuri de testare dar, acestea sunt importante și pentru utilizatori deoarece în funcție de acestea se știe dacă se poate folosi aplicația și dacă pe viitor sunt necesare viitoare imbunătățiri ale sistemului pentru o mai bună rulare a aplicației.

Cerințele de sistem reflectă ceea ce programul sau aplicația ar trebuie să facă și cum să se comporte.

Cerințele de sistem ale aplicației UniConnect sunt următoarele:

* Aplicația trebuie să permită utilizatorilor să creeze noi camere sau să intre în cele care sunt deja creeate;
* Aplicația trebuie să permită utilizatorilor să pornească/oprească camera sau microfonul;
* Aplicația trebuie să permită utilizatorilor să își partajeze ecranul;
* Aplicația trebuie să premită utilizatorilor să iasă din cameră oricând doresc.

Pentru a folosi aplicația UniConnect aveți nevoie de următoarele:

* Periferice: Căști, cameră web, mouse, tastatură;
* Cerințe hardware: Placă de sunet;
* Conexiune la internet;
* Sistem de operare instalat.

## **Arhitectura sistemului**

Arhitectura sistemului este o reprezentare a structurii și comportamentului unui sistem. Aceasta cuprinde componentele sistemului și ne arată cum acest sistem este folosit.*[7]*

Arhitectura sistemului pentru aplicația UniConnect este prezentată în diagrama următoare:

A diagram of a software application

Description automatically generated

Figură 4 Arhitectura de sistem. Sistemul conține aplicația în sine si cele 2 servere de tip STUN. Server-ul principal este server-ul STUN 1 (Session Traversal Utilities for NAT[8]). În cazul în care server-ul STUN 1 nu comunică, atunci se trece pe server-ul STUN 2. De la server merge mai departe la ceilalți participanți.

## **Cerințe software**

Cerințele software sunt cele care descriu în detaliu funcționalitatea unui sistem. Fiecare sistem are la rândul său și constrângeri. Cerințele software sunt foarte importante pentru echipa de testare deoarece pe baza acestora sunt create scenarii de testare. Pentru a începe dezvoltarea unui sistem, este necesar să avem un document SRS pus la punct.

Documentul SRS sau Specificația Cerințelor Software este o colecție structurată de informații referitoare la sistemul software care urmează să fie dezvoltat. Cerințele software sunt de două tipuri: Cerințe software funcționale și Cerințe software non-funcționale. Cerințele software reprezintă înțelegerea dintre client și contractor sau furnizor în legătură cu modul de funcționare alt sistemului.*[9]*

Cerințele software în urma cărora aplicația UniConnect a fost dezvoltată sunt:

* Cerințe funcționale:
* Utilizatorul trebuie să aibă acces să creeze o cameră sau să intre într-o cameră deja creată;
* Utilizatorul trebuie să aibă acces să poată partaja ecranul;
* Utilizatorul trebuie să aibă acces să își folosească microfonul sau camera web.
* Cerințe non-funcționale:
* Timp de latență minim pentru asigurarea transmisiunii audio și video;
* Sistemul trebuie să fie compatibil cu orice navigator web;
* Sistemul trebuie să fie scalabil pentru a asigura o încărcare balansată între servere;
* Sistemul trebuie sa fie scalabil orizontal pentru a suporta un număr cât mai mare de utilizatori și cât mai multe camere create simultan;
* Utilizatorul trebuie să dispună de o interfață ușor de folosit;
* În timpul vorbirii, nu trebuie să fie prezent efectul de ecou;
* Aplicația trebuie să fie capabilă să accepte mai multe tipuri de camere web (camere web cu o claritate ridicată sau camere web cu o claritate mai scăzută).

## **Diagrama Use-Case a aplicației UniConnect**

Diagrama Use-Case reprezintă cum este folosit un sistem de către unul sau mai mulți utilizatori. Aceasta nu reprezintă cum funcționează sistemul.

A diagram of a camera

Description automatically generated

Figură 5 Diagrama Use-Case

Utilizatorul poate să creeze o camera nouă sau poate intra deja într-una existentă. Pe langă acestea, utilizatorul poate sa își pornească sau să își oprească camera, să își pornească sau să își oprească microfonul și să își partajeze ecranul.

# **Fundamentele teoretice**

## **Limbajele de programare HTML și CSS**

HyperText Markup Language sau mai des folosit HTML, este un limbaj descriptiv utilizat pentru a crearea de site-uri. Este cel mai simplu limbaj dintre cele folosite pentru crearea site-urilor. *[10]*

Cascading Style Sheets sau CSS, este un limbaj folosit pentru editarea și formatarea elementelor prezente pe site. Majoritatea persoanelor confundă HTML și CSS ca fiind același lucru, dar fiecare este conceput pentru alt scop.*[10]*

Cele mai folosite limbaje în domeniul web sunt PHP, Java, JavaScript, ASP.NET, React, Angular sau Phyton iar având cunoștiințe în limbajul HTML reprezintă un avantaj în învățarea acestora. *[10]*

## **Limbajul de programare JavaScript**

JavaScript sau JS, este un limbaj bazat pe programarea orientate obiecte. Folosit în general pentru implementarea funcționalitățiilor care stau la baza unui site web, codul JS este rulat de către browser. La începturile sale, în anul 1995, JS a apărut sub numele de Mocha, ulterior LiveScript, iar în final primind numele de JavaScript. Multe persoane consider că JavaScript și Java vin din aceeași “familie”, dar ele sunt concepute pentru chestii diferite. *[11]*

Reprezentarea unei pagini web este reținută de către browsere sub forma unui arbore de obiecte, aceste obiecte fiind puse la dispoziția script-urilor de JS. Acest arbore este cunoscut sub denumirea de DOM sau Document Object Model. Pentru acest DOM, există un standard numit W3C, care oferă baza scrierii de script-uri portabile, pentru ca acestea să poată fi rulate pe toate navigatoarele de internet.*[11]*  W3C sau cunoscut ca și Consorțiul Web, are ca scop dezvoltarea de standarde pentru World Wide Web.*[12]*

În ultima perioadă, cea mai întâlnită tehnică de dezvoltare a paginilor web este AJAX sau “Asynchronous JavaScript and XML”. Tehnica constă în executarea cererilor de tip HTTP în fundal fără a fi necesară reîncărcarea în totalitate a paginii web.*[11]*

Majoritatea aplicațiilor web sunt dezvoltate folosind API-uri. API sau Application Programming Interface este un set de protocoale, sub-programe definite care ajută la programarea aplicațiilor.*[14]* API-urile trimit date în mod bidirectional.*[15]*

A diagram of a computer

Description automatically generated

Figură 6 Web API[13]

## **Tehnologia WebRTC**

WebRTC sau Web Real-Time Communications, este o tehnologie open-source care a fost promovată de Google. Această tehnologie permite comunicarea în timp real prin intermediul JavaScript API. Codec-ul folosit de WebRTC care convertește semnalele analogice în semnale digitale pentru a le trimite pe o rețea de date este VP8.*[16]*

VP8 sau Video Compression Format, este specificație folosită la codarea și decodarea video de înaltă calitate. Codec-ul VP8 este în contradicție cu codec-ul H.264 care este oferit gratuit de către Google.*[17]*

A computer network with a cloud and a cloud with people connected

Description automatically generated with medium confidence

Figură 7 WebRTC[18]

A diagram of a customer service

Description automatically generatedAgoraRTM folosit în combinație cu WebRTC poate oferi o bază solidă când vine vorba de dezvoltarea aplicațiilor de tip Real-Time Communications.*[19]*

Figură 8 Conexiunea dintre 2 persoane în WebRTC[20]

Arhitectura unui server WebRTC:

A diagram of a network

Description automatically generated

Figură 9 Arhitectura unui server WebRTC[21]

# **Implementarea aplicației UniConnect**

## **Back-end-ul aplicației**

Pentru a dezvolta partea de back-end a aplicației m-am folosit de tehnologia WebRTC împreună cu SDK-ul Agora RTM.

SDK sau Software Development Kit, este un set de instrumente care ajută la dezvoltarea software deoarece totul este predefinit deja.*[22]*

A diagram of a sign

Description automatically generated

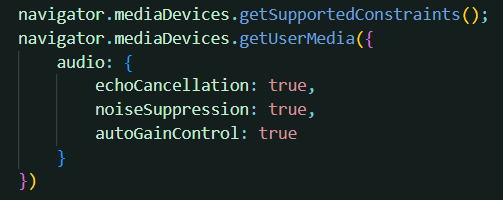
Figură 10 Agora RTM[23]

Back-end-ul aplicației este împărțit în două fișiere. Unul pentru toată funcționalitatea aplicației, numit “main.js” și unul pentru funcționalitatea paginii principale din care se pornesc apelurile numit “lobby.js”.

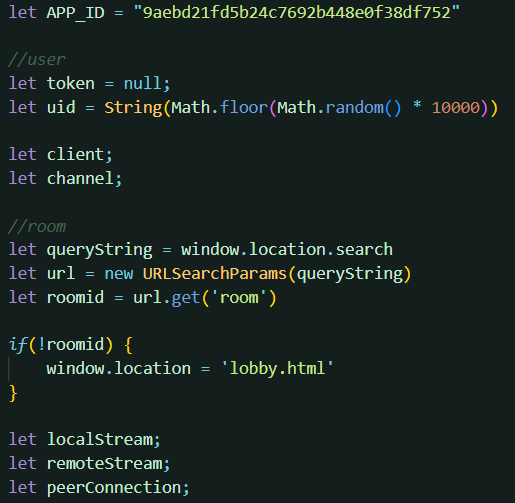
A screen shot of a computer code

Description automatically generated

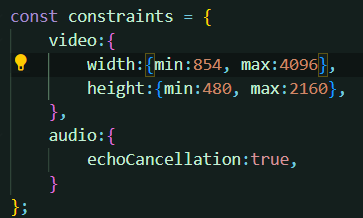
În codul de mai sus are loc partea de validare a butonului submit și se verifică dacă este trimis către server atunci cand utilizatorul dorește să creeze o cameră nouă sau să intre intr-o cameră deja creată. Dacă utilizatorul introduce în “joinForm” id-ul unei camere iar acesta este unul valid, mai departe se trimite o cerere la server pentru a-i permite utilizatorului să intre. Daca utilizatorul are permisiunea de a intra, acesta este redirecționat către pagina camerei prin linia de cod “window.location = `main.html?room=${invite}”. În fișierul main.html sunt implementate butoanele pe care utilizatorul le poate folosi și chenarul în care este arătat ce captează camera web.

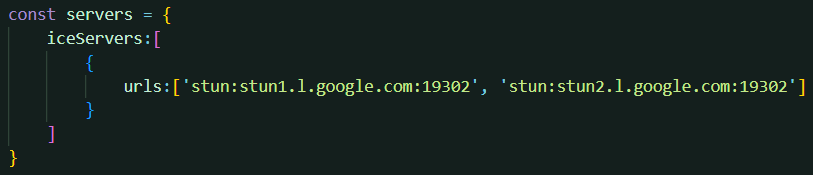


Această bucată de cod reprezintă constrângeriile pentru partea de audio, pentru a reduce efectul de ecou și pentru a anula zgomotul din fundal.



Mai sus au fost declarate variabilele globale de care ne vom folosi pe parcursul dezvoltării aplicației. Variabila „APP\_ID” reprezintă identificatorul aplicației din consola SDK-ului Agora RTM. Pentru indentificatorul utilizatorilor “uid” am folosit o formulă pentru a genera un indentificator aleatoriu de fiecare dată când un utilizator intră. Dacă același utilizator intră și reintră în conferință, acesta va avea un alt identificator. Identificatorul camerei este preluat de comanda “url.get(‘room’)”. Variabilele “localStream”, “remoteStream” și “peerConnection” sunt folosite la realizarea conexiunii dintre utilizator și ceilalți utilizatori virtuali.

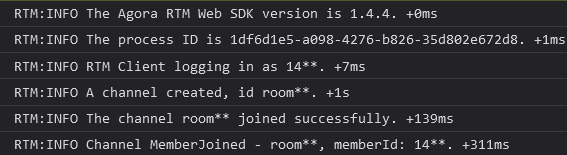
Aici avem constrângeriile pentru captarea video. Minimul acceptat pentru claritatea unei camere web este de 480p, iar maximul acceptat este o rezoluție de 4K. Dacă nu sunt respectate aceste valori, camera nu poate fi pornită.



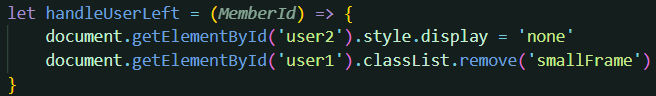
Serverele de tip ICE sau “Interactive Connectivity Establishment server”, sunt importante în stabilirea conexiunii de tip peer-to-peer. Serverele ajută la localizarea interfețelor de rețea și a IP-urilor care permit conexiunea peer-to-peer.*[24]*

Personal am folosit serverele de la Google deoarece sunt gratuite și ajută ca aplicația să fie dezvoltată cu un cost cât mai mic. Primul server este cel principal, prin care se realizează conexiunea, iar al doilea server este cel de rezervă, în cazul în care primul server pică sau nu poate stabili conexiunea.

Pentru a crea instanțele, am creat o funcție asincronă care trimite un mesaj în consola navigatorului web atunci când o persoana a intrat sau a ieșit din cameră.



Figură 11 Mesajele trimise în consolă când o cameră este creată iar un utilizator a intrat.

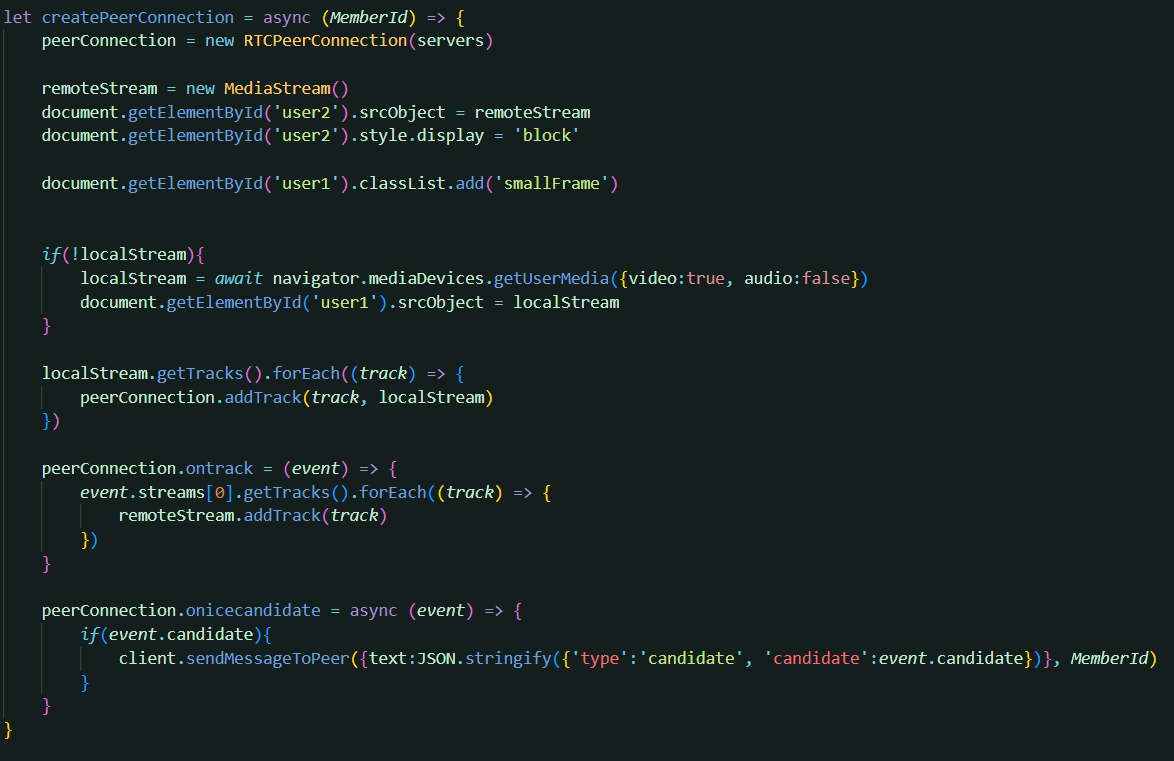
În cazul în care un utilizator nu reușește să se conexteze, va apărea un mesaj de eroare în consolă.

Dacă este un singur utilizator în cameră, acesta se va vedea pe tot ecranul. Însă, dacă este o situație în care sunt prezenți mai mulți utilizatori, utilizatorul prezent in fața monitorului se va vedea într-un chenar mic amplasat în partea de sus a paginii iar pe pe suprafața cea mare vor apărea restul utilizatorilor. Dacă utilizatorul iese, acel chenar este scos.

Figură 12 Un singur utilizator prezent în cameră



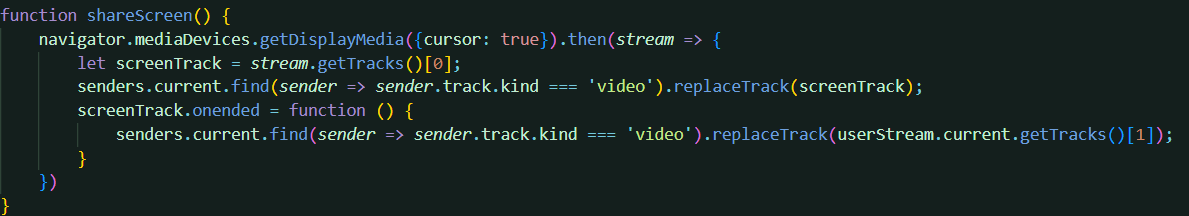
Figură 13 Mai mult de un utilizator prezent în cameră



În bucata de cod de mai sus are loc conextiunea de tip peer-to-peer. La începutul funcției a fost declarata variabila “remoteStream” care reține fluxul media al utilizatorului conectat de la distanță. Utilizatorul 1 sau cel prezent “fizic” este trimis către acel chenar mic unde se va vedea ce capturează camera web. După aceea, este verificat daca “localStream” este definit. Dacă “localStream” nu este definit, acesta poate reține doar fluxul video, fară cel audio. Ulterior, fiecare urmă din fluxul local este adăugată la “peerConnection” pentru a permite conexiunea cu celălalt utilizator. Evenimentul de tip ICECandidate este trimis între utilizatori pentru a stabili conexiunea.



Aici sunt definite funcțiile de control pentru partea audio și partea video. Totodată, aici sunt aplicate și culorile pentru fiecare buton. Fiecare funcție are butonul ei. Utilizatorul are posibilitatea de a-și opri sau porni microfonul sau camera web. Pe lângă acestea, utilizatorul poate să vadă partea video in modul “imagine-în-imagine”.



Utilizatorul are posibilitatea de a-și partaja ecranul cu ceilalți utilizatori. Funcția trebuie să solicite premisiunea utilizatorului pentru a partaja ecranul, totodată este vizibil și cursorul mouse-ului.

# **BIBLIOGRAFIE**

[1]

claudiad (2007) ‘videoconferință’, *MDA2*. https://dexonline.ro/definitie/videoconferin%C8%9B%C4%83/1252506. [Accesat în 12 iunie 2024]

[2]

https://vivo.tech/blogs/news/zoom-vs-google-meet-vs-microsoft-teams-how-to-choose. [Accesat în 13 iunie 2024]

[3]

https://teach.cbs.dk/tools-for-teaching/zoom/. [Accesat în 13 iunie 2024]

[4]

https://its.northeastern.edu/2023/10/19/make-the-switch-today-to-the-new-microsoft-teams/. [Accesat în 13 iunie 2024]

[5]

https://bignet.in/blog/2158/what-is-a-google-meet-in-english. [Accesat în 13 iunie 2024]

[6]

https://pctroubleshooting.ro/topic/16962-ce-sunt-cerintele-de-sistem/. [Accesat în 13 iunie 2024]

[7]

https://arhitectura980.wordpress.com/arhitectura-sistemului/. [Accesat în 13 iunie 2024]

[8]

https://en.wikipedia.org/wiki/STUN. [Accesat în 13 iunie 2024]

[9]

https://en.wikipedia.org/wiki/Software\_requirements\_specification. [Accesat în 13 iunie 2024]

[10]

https://www.link-academy.com/curs-introducere-html-css. [Accesat în 13 iunie 2024]

[11]

https://ro.wikipedia.org/wiki/JavaScript. [Accesat în 13 iunie 2024]

[12]

https://ro.wikipedia.org/wiki/World\_Wide\_Web\_Consortium. [Accesat în 13 iunie 2024]

[13]

https://dev.addinn-group.com/2024/05/20/api-management-a-catalyst-for-innovation-in-the-banking-world/. [Accesat în 13 iunie 2024]

[14]

https://www.google.com/search?client=firefox-b-d&q=ce+este+un+API. [Accesat în 13 iunie 2024]

*[15]* https://support.apple.com/ro-ro/guide/shortcuts-mac/apd2e30c9d45/mac. [Accesat în 13 iunie 2024]

*[16]* https://www.3cx.com/global/ro/voip-sip-webrtc/ce-este-webrtc/. [Accesat în 13 iunie 2024]

*[17]* https://www.3cx.com/global/ro/voip-sip-webrtc/vp8/. [Accesat în 13 iunie 2024]

*[18]* https://dyte.io/blog/webrtc/. [Accesat în 13 iunie 2024]

*[19]* https://docs.agora.io/en/signaling/overview/product-overview?platform=android. [Accesat în 13 iunie 2024]

*[20]* https://eytanmanor.medium.com/an-architectural-overview-for-web-rtc-a-protocol-for-implementing-video-conferencing-e2a914628d0e. [Accesat în 13 iunie 2024]

*[21]* https://www.wowza.com/blog/webrtc-server-what-it-is-and-why-you-need-one. [Accesat în 13 iunie 2024]

*[22]* https://www.adjust.com/glossary/sdk/. [Accesat în 14 iunie 2024]

*[23]*https://docs.agora.io/en/1.x/signaling/get-started/get-started-sdk?platform=android  
[Accesat în 14 iunie 2024]

*[24]*https://medium.com/@mshuecodev/what-should-i-know-about-ice-server-ebed04f54369. [Accesat în 14 iunie 2024]