

Ομάδα Ασκήσεων Α3: Τεχνολογίες Βιντεοδιάσκεψης

Α. Δικτύωση πολυμέσων

Θεματική ενότητα: Α. Δικτύωση πολυμέσων

Ομάδα Ασκήσεων: Α3 – Τεχνολογίες Βιντεοδιάσκεψης

4^η Άσκηση

Άσκηση: Α3.2. – Σχεδιασμός και υλοποίηση εφαρμογής βιντεοδιάσκεψης με WebRTC

Ομάδα Ασκήσεων: Α3 – Τεχνολογίες βιντεοδιάσκεψης

Θεματική ενότητα: Α. Δικτύωση πολυμέσων

Διάρκεια: 6 ώρες

Προαπαιτούμενα:

- Το Εκπαιδευτικό Υλικό Αναφοράς «Εισαγωγή στις Τεχνολογίες Βιντεοδιάσκεψης», και κυρίως η σχετική παρουσίασης» καθώς και η αναφορά "An introduction to FSU".
- Προαιρετικά, τα σχετικά κείμενα και παρουσιάσεις για τα H.323 και SIP.

Σκοπός

Σκοπός της άσκησης είναι οι φοιτητές να σχεδιάσουν και να υλοποιήσουν μια απλή υπηρεσία βιντεοδιάσκεψης βασισμένη στο WebRTC, η οποία θα επιτρέπει σε τουλάχιστον 4 χρήστες να συμμετέχουν ταυτόχρονα σε ένα εικονικό δωμάτιο επικοινωνίας.

Το WebRTC επιτρέπει σε web εφαρμογές να μεταδίδουν ήχο, βίντεο και δεδομένα μεταξύ browsers χωρίς την ανάγκη εγκατάστασης πρόσθετων plugins

Μαθησιακοί Στόχοι Δραστηριότητας

Με την ολοκλήρωση της άσκησης, οι φοιτητές θα πρέπει να μπορούν να:

- Εξηγούν τη βασική αρχιτεκτονική του WebRTC.
- Χρησιμοποιούν τα APIs getUserMedia, RTCPeerConnection και, προαιρετικά, RTCDataChannel.
- Υλοποιούν μηχανισμό signaling μέσω WebSocket ή αντίστοιχης τεχνολογίας.
- Διαχειρίζονται πολλαπλές ταυτόχρονες συνδέσεις μεταξύ χρηστών.
- Υποστηρίζουν είσοδο και έξοδο χρηστών από ένα δωμάτιο βιντεοδιάσκεψης.
- Αξιολογούν τις διαφορές μεταξύ αρχιτεκτονικής mesh, SFU και MCU για πολυμερή επικοινωνία.

- Αναλύουν βασικά ζητήματα ποιότητας υπηρεσίας, καθυστέρησης, NAT traversal και κλιμάκωσης.

Απαιτούμενη Υποδομή – Οδηγίες

-

Σύντομη Περιγραφή – Σενάριο

Καλείστε να αναπτύξετε μια web εφαρμογή βιντεοδιάσκεψης που θα επιτρέπει σε πολλούς χρήστες να συνδέονται σε ένα κοινό δωμάτιο και να επικοινωνούν με ήχο και βίντεο σε πραγματικό χρόνο.

Η εφαρμογή πρέπει να υποστηρίζει τουλάχιστον 4 ταυτόχρονους χρήστες στο ίδιο δωμάτιο.

Η λύση μπορεί να βασιστεί αρχικά σε αρχιτεκτονική mesh, όπου κάθε χρήστης δημιουργεί ξεχωριστή WebRTC σύνδεση με κάθε άλλο χρήστη. Για 4 χρήστες αυτό είναι αποδεκτό εκπαιδευτικά, επειδή επιτρέπει στους φοιτητές να κατανοήσουν σε βάθος το signaling και τη διαχείριση πολλαπλών RTCPeerConnection αντικειμένων. Για μεγαλύτερη κλίμακα, οι φοιτητές μπορούν να συζητήσουν ή να υλοποιήσουν προαιρετικά αρχιτεκτονική SFU, καθώς οι SFU λύσεις χρησιμοποιούνται για πιο αποδοτικές πολυμερείς WebRTC συνδιασκέψεις.

Χρήση εργαλείων AI

Η χρήση εργαλείων Τεχνητής Νοημοσύνης, όπως ChatGPT, GitHub Copilot ή παρόμοιων βοηθών προγραμματισμού, **όχι μόνο επιτρέπεται αλλά επιβάλλεται**, υπό την προϋπόθεση ότι οι φοιτητές κατανοούν, ελέγχουν και τεκμηριώνουν τον κώδικα που παραδίδουν.

Τα εργαλεία AI μπορούν να χρησιμοποιηθούν για:

- κατανόηση βασικών εννοιών WebRTC,
- βοήθεια στον σχεδιασμό της αρχιτεκτονικής,
- παραγωγή αρχικού κώδικα,
- εντοπισμό και διόρθωση σφαλμάτων,
- βελτίωση τεκμηρίωσης,
- δημιουργία διαγραμμάτων ή ψευδοκώδικα,
- προτάσεις για testing και debugging.

Δεν επιτρέπεται όμως η άκριτη αντιγραφή κώδικα χωρίς κατανόηση. Η κατανόηση είναι αυτή που θα αξιολογηθεί. Κάθε φοιτητής/φοιτήτρια θα πρέπει να μπορεί να εξηγήσει προφορικά και γραπτά:

- πώς λειτουργεί το signaling,
- πώς δημιουργούνται τα RTCPeerConnection αντικείμενα,
- πώς ανταλλάσσονται offer, answer και ICE candidates,
- πώς υποστηρίζονται τουλάχιστον 4 χρήστες,
- πώς γίνεται η διαχείριση εισόδου και αποχώρησης χρηστών,
- ποιο τμήμα του κώδικα υλοποιήθηκε, τροποποιήθηκε ή προτάθηκε με βοήθεια AI.

Στην τεχνική αναφορά πρέπει να υπάρχει ξεχωριστή ενότητα με τίτλο:

Δήλωση χρήσης εργαλείων Τεχνητής Νοημοσύνης

Σε αυτή την ενότητα οι φοιτητές πρέπει να αναφέρουν:

- ποια εργαλεία AI χρησιμοποίησαν,
- για ποια μέρη της εργασίας τα χρησιμοποίησαν,
- ενδεικτικά prompts που έδωσαν,
- τι τροποποίησαν οι ίδιοι,
- ποια προβλήματα εντόπισαν ή διόρθωσαν στον παραγόμενο κώδικα.

Παράδειγμα:

Για την εργασία χρησιμοποιήθηκε το ChatGPT για την αρχική κατανόηση του WebRTC signaling και για την παραγωγή ενδεικτικού κώδικα WebSocket server. Ο παραγόμενος κώδικας τροποποιήθηκε από την ομάδα ώστε να υποστηρίζει πολλαπλά δωμάτια, αποχώρηση χρηστών και διαχείριση 4 ταυτόχρονων peer connections.

Ανάλυση Άσκησης A3.2

1. Βασικές λειτουργικές απαιτήσεις

Η εφαρμογή πρέπει να υποστηρίζει τα εξής:

1.1 Είσοδος σε δωμάτιο

Ο χρήστης πρέπει να μπορεί να:

- εισάγει ένα όνομα χρήστη,
- εισάγει ή να δημιουργεί ένα roomId,
- συνδέεται σε δωμάτιο βιντεοδιάσκεψης.

1.2 Πρόσβαση σε κάμερα και μικρόφωνο

Η εφαρμογή πρέπει να ζητά πρόσβαση στην κάμερα και το μικρόφωνο του χρήστη μέσω `navigator.mediaDevices.getUserMedia()`.

Η κλήση στο `getUserMedia()` ενεργοποιεί αίτημα άδειας από τον browser και επιστρέφει ένα `MediaStream` εφόσον ο χρήστης αποδεχθεί την πρόσβαση.

1.3 Μετάδοση ήχου και βίντεο

Κάθε χρήστης πρέπει:

- να βλέπει το δικό του τοπικό video stream,
- να βλέπει τα video streams των υπόλοιπων χρηστών,
- να ακούει τον ήχο των υπόλοιπων χρηστών.

1.4 Signaling server

Πρέπει να υλοποιηθεί signaling server, για παράδειγμα (ενδεικτικά) με:

- Node.js και WebSocket,
- Node.js και Socket.IO,
- Java Spring Boot WebSocket,
- άλλη ισοδύναμη τεχνολογία.

Ο signaling server δεν μεταδίδει το ίδιο το βίντεο ή τον ήχο. Χρησιμοποιείται για την ανταλλαγή μηνυμάτων ελέγχου, όπως:

- join-room,
- user-joined,
- offer,
- answer,
- ice-candidate,
- user-left.

Η διαδικασία signaling είναι απαραίτητη για τη δημιουργία των `RTCPeerConnection` συνδέσεων, καθώς οι peers πρέπει να ανταλλάξουν offers, answers και ICE candidates.

1.5 Υποστήριξη τουλάχιστον 4 χρηστών

Η εφαρμογή πρέπει να δοκιμαστεί με τουλάχιστον 4 χρήστες στο ίδιο δωμάτιο.

Για mesh αρχιτεκτονική, κάθε χρήστης θα πρέπει να διατηρεί σύνδεση με κάθε άλλο χρήστη.

1.6 Αποχώρηση χρήστη

Όταν ένας χρήστης αποχωρεί:

- οι υπόλοιποι χρήστες πρέπει να ενημερώνονται,
- το αντίστοιχο remote video πρέπει να αφαιρείται από το UI,
- οι σχετικές RTCPeerConnection συνδέσεις πρέπει να κλείνουν καθαρά.

2. Προτεινόμενη αρχιτεκτονική

Για την υποχρεωτική έκδοση της άσκησης προτείνεται αρχιτεκτονική mesh. Κάθε χρήστης συνδέεται απευθείας με όλους τους υπόλοιπους μέσω WebRTC. Ο signaling server χρησιμοποιείται μόνο για την αρχική διαπραγμάτευση και την ανταλλαγή ICE candidates. Μετά το signaling θα πραγματοποιείται ομότιμη επικοινωνία:

Client 1 ⇌ Client 2, Client 1 ⇌ Client 3, Client 1 ⇌ Client 4, Client 2 ⇌ Client 3, Client 2 ⇌ Client 4, Client 3 ⇌ Client 4

3. Τεχνικές απαιτήσεις

Η υλοποίηση πρέπει να περιλαμβάνει:

3.1 Frontend

Ενδεικτικές επιλογές:

HTML, CSS, JavaScript, React, Vue, Angular.

Το frontend πρέπει να περιλαμβάνει:

- φόρμα εισόδου σε δωμάτιο,
- περιοχή προβολής τοπικού video,
- περιοχή προβολής remote videos,
- κουμπί ενεργοποίησης/απενεργοποίησης μικροφώνου,
- κουμπί ενεργοποίησης/απενεργοποίησης κάμερας,
- κουμπί αποχώρησης από το δωμάτιο.

3.2 Backend / Signaling server

Ο signaling server πρέπει να:

- διαχειρίζεται δωμάτια,
- κρατά λίστα χρηστών ανά δωμάτιο,
- ενημερώνει τους υπάρχοντες χρήστες όταν μπαίνει νέος χρήστης,
- προωθεί offer, answer και ice-candidate μηνύματα,
- ενημερώνει τους υπόλοιπους χρήστες όταν κάποιος αποχωρεί.

3.3 WebRTC

Η εφαρμογή πρέπει να χρησιμοποιεί:

- `navigator.mediaDevices.getUserMedia()`,
- `RTCPeerConnection`,
- `addTrack()`,
- `ontrack`,
- `createOffer()`,
- `createAnswer()`,
- `setLocalDescription()`,
- `setRemoteDescription()`,
- `onicecandidate`,
- `addIceCandidate()`.

Συνιστάται οι φοιτητές να μελετήσουν το WebRTC perfect negotiation pattern, καθώς μειώνει προβλήματα σύγκρουσης κατά τη διαπραγμάτευση σύνδεσης μεταξύ peers.

4. Ελάχιστο σενάριο λειτουργίας

- Η εφαρμογή πρέπει να μπορεί να εκτελέσει το εξής σενάριο:
- Ο χρήστης Α ανοίγει την εφαρμογή και δημιουργεί δωμάτιο `room1`.
- Ο χρήστης Β συνδέεται στο `room1`.
- Οι χρήστες Α και Β βλέπουν και ακούν ο ένας τον άλλον.
- Ο χρήστης Γ συνδέεται στο ίδιο δωμάτιο.
- Όλοι βλέπουν και ακούν όλους.
- Ο χρήστης Δ συνδέεται στο ίδιο δωμάτιο.
- Η βιντεοδιάσκεψη λειτουργεί με 4 χρήστες.
- Ο χρήστης Γ αποχωρεί.
- Οι υπόλοιποι χρήστες συνεχίζουν να επικοινωνούν χωρίς σφάλμα.

5. Παραδοτέα

Οι φοιτητές πρέπει να παραδώσουν:

5.1 Πηγαίο κώδικα

Ο κώδικας πρέπει να περιλαμβάνει:

frontend, signaling server, οδηγίες εγκατάστασης, οδηγίες εκτέλεσης.

5.2 Σύντομη Τεχνική αναφορά

Η αναφορά πρέπει να περιλαμβάνει:

- Περιγραφή αρχιτεκτονικής.
- Διάγραμμα επικοινωνίας μεταξύ clients και signaling server.
- Περιγραφή των signaling μηνυμάτων.
- Περιγραφή της διαδικασίας δημιουργίας WebRTC σύνδεσης.

- Περιγραφή διαχείρισης πολλαπλών χρηστών.
- Προβλήματα που αντιμετωπίστηκαν και τρόποι επίλυσης.
- Περιορισμοί της mesh αρχιτεκτονικής.
- Προτάσεις βελτίωσης με χρήση SFU ή TURN server.

5.3 Σύντομη δια ζώσης παρουσίαση και Demo

Μία σύντομη παρουσίαση της Τεχνικής Αναφοράς, της αρχιτεκτονικής και τεκμηρίωση των αποφάσεων που λήφθηκαν κατά το σχεδιασμό και την υλοποίηση.

Μία σύντομη επίδειξη 3–5 λεπτών που δείχνει:

- την είσοδο 4 χρηστών στο ίδιο δωμάτιο,
- την ταυτόχρονη λειτουργία ήχου και βίντεο,
- την αποχώρηση ενός χρήστη,
- τη συνέχιση της κλήσης από τους υπόλοιπους.

6. Προαιρετικές επεκτάσεις

Για επιπλέον βαθμολογία, μπορούν να υλοποιηθούν μία ή περισσότερες από τις παρακάτω λειτουργίες:

- Text chat με RTCDataChannel.
- Screen sharing.
- Ένδειξη ενεργού ομιλητή.
- Δυνατότητα mute/unmute.
- Δυνατότητα camera on/off.
- Υποστήριξη πολλαπλών δωματίων.
- Χρήση TURN server.
- Καταγραφή στατιστικών μέσω `getStats()`.
- Προσαρμογή ποιότητας video.
- Υλοποίηση με SFU, π.χ. mediasoup, Janus ή Jitsi Videobridge.
- Authentication χρηστών.
- Responsive UI για κινητές συσκευές.

Σημειώματα σχετικά με τα δικαιώματα Πνευματικής Ιδιοκτησίας

Σημείωμα Αναφοράς

Copyright Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών 2026 Παντελής Μπαλαούρας.
«Ειδικά Θέματα – Πολυμέσα και Ασύρματη Δικτύωση» , Α. Δικτύωση Πολυμέσων, Ομάδα
ασκήσεων Α3. Τεχνολογίες Βιντεοδιάσκεψης.» Έκδοση: 1.0. Αθήνα 2026.

Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά
Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση.



Η άδεια αυτή ανήκει στις άδειες που ακολουθούν τις προδιαγραφές του Ορισμού Ανοικτής Γνώσης [2], είναι ανοικτό πολιτιστικό έργο [3] και για το λόγο αυτό αποτελεί ανοικτό περιεχόμενο [4].

[1] <http://creativecommons.org/licenses/by-sa/4.0/deed.el>

[2] <http://opendefinition.org/okd/ellinika/>

[3] <http://freedomdefined.org/Definition/EI>

[4] <http://opendefinition.org/buttons/>

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- Το Σημείωμα Αναφοράς
- Το Σημείωμα Αδειοδότησης
- Τη δήλωση διατήρησης Σημειωμάτων
- Το σημείωμα χρήσης έργων τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.