



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Διαδικτυακή συνεργατική πλατφόρμα για προηγμένες ιατρικές διαβουλεύσεις

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Γεώργιος Κ. Ρασσιάς

Επιβλέπων: Παναγιώτης Δ. Τσανάκας
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2015



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Διαδικτυακή συνεργατική πλατφόρμα για προηγμένες ιατρικές διαβουλεύσεις

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Γεώργιος Κ. Ρασσιάς

Επιβλέπων: Παναγιώτης Δ. Τσανάκας

Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 21^η Ιουλίου 2015.

.....
Παναγιώτης Τσανάκας
Καθηγητής Ε.Μ.Π.

.....
Δημήτριος Κουτσούρης
Καθηγητής Ε.Μ.Π.

.....
Ηλίας Μαγκλογιάννης
Επίκουρος καθηγητής ΠΑ.ΠΕΙ.

Αθήνα, Ιούλιος 2015

.....

Γεώργιος Κ. Ρασσιάς

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Γεώργιος Κ. Ρασσιάς, 2015.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Η σταδιακή μεταστροφή που παρατηρείται στη σύγχρονη ιατρική επιστήμη, για τη συγκρότηση ιατρών σε πολυεπιστημονικές ομάδες που διαχειρίζονται από κοινού ιατρικά περιστατικά, εγείρει την ανάγκη για τη δημιουργία εργαλείων που θα επιτρέπουν τη συνεργασία σε πραγματικό χρόνο (online) γεωγραφικά κατακεντρωμένου ιατρικού προσωπικού. Η παρούσα διπλωματική εργασία εισάγει μια πλατφόρμα ιστού με χαρακτηριστικά αποδοτικής διαχείρισης και ανταλλαγής ιατρικών δεδομένων για την παροχή πραγματικού χρόνου, συνεργατικών ιατρικών υπηρεσιών. Η παρουσιαζόμενη εργασία αξιοποιεί τελευταίες διαδικτυακές τεχνολογίες και διεπαφές προγραμματισμού εφαρμογών (APIs) για να υποστηρίξει την επεξεργασία των ιατρικών δεδομένων από την πλευρά του ιατρού χρήστη. Επιπρόσθετα, για να αντιμετωπιστούν τα συνήθη σημεία συμφόρησης εύρους ζώνης και τα ζητήματα κλιμακωσιμότητας που παρουσιάζει ο κεντροποιημένος διαμοιρασμός δεδομένων, εισάγεται ένας έμμεσος μηχανισμός κλήσης απομακρυσμένης διαδικασίας (RPC) μέσω συγχρονισμού αντικειμένων πάνω από διαύλους του WebRTC.

Λέξεις κλειδιά

τηλεϊατρική, συνεργατική ροή εργασίας, συνεργατική διάγνωση, ιατρική απεικόνιση, μορφή DICOM, HTML5, WebGL, WebRTC, εφαρμογή μονής σελίδας

Abstract

The gradual shift in modern medical practice, from working alone clinical doctors to MDTs (MultiDisciplinary Teams), raises the need of online real-time collaboration among geographically distributed medical personnel. This diploma thesis presents a web-based platform, featuring an efficient medical data management and exchange, for hosting real-time collaborative services. The presented work leverages state-of-the-art features of the web (technologies and APIs) to support client-side medical data processing. Moreover, to address the typical bandwidth bottleneck and known scalability issues of centralized data sharing, an indirect RPC (Remote Process Call) scheme is introduced through object synchronization over the WebRTC paradigm.

Keywords

telemedicine, collaborative workflow, collaborative diagnosis, medical imaging, DICOM format, HTML5, WebGL, WebRTC, Single Page Application

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τους αγαπητούς κυρίους Παναγιώτη Τσανάκα και Ηλία Μαγκλογιάννη για τις πολύτιμες συμβουλές και την εμπιστοσύνη που έδειξαν προς το πρόσωπό μου αναθέτοντάς μου την παρούσα διπλωματική εργασία. Ακόμα, τις θερμές μου ευχαριστίες στο συνεργάτη και πάνω από όλα φίλο, Χρήστο Ανδρίκο που με την εμπειρία, την καθοδήγηση και την υπομονή του συνέβαλε τα μέγιστα στην εκπόνησή της. Τέλος, ευχαριστίες απευθύνω και στο φίλο Αριστοτέλη Κατσανά για την πολυποίκιλη συνεισφορά του όλο αυτόν τον καιρό.

Περιεχόμενα

1	Εισαγωγή	1
1.1	Προσέγγιση της ιατρικής συνεργατικότητας	1
1.2	Σκοπός της παρούσας εργασίας.....	2
1.3	Προτεινόμενη λύση.....	3
1.4	Οργάνωση κειμένου	4
2	Σχετικά πρότυπα, εφαρμογές και βιβλιογραφία.....	5
2.1	Πρότυπα ανταλλαγής ιατρικών δεδομένων	5
2.1.1	Το πρότυπο HL7	5
2.1.2	EMR/EHR	6
2.1.3	Το πρότυπο DICOM και τα PACS.....	6
2.1.3.1	Εισαγωγή στο DICOM	7
2.1.3.2	Βασικές έννοιες	7
2.1.3.3	Το λεξικό DICOM.....	9
2.1.3.4	Η ιεράρχηση πληροφοριών κατά DICOM.....	10
2.1.3.5	Ιατρικές εικόνες στο DICOM	12
2.1.3.6	DICOM BMPs.....	12
2.1.3.7	Συμπιεσμένες DICOM εικόνες.....	15
2.1.3.8	Web Access to DICOM Objects (WADO)	19
2.2	Ιατρική απεικόνιση.....	20
2.3	Εφαρμογές απεικόνισης ιατρικών εικόνων	21
2.3.1	OsiriX.....	21
2.3.2	RadiAnt.....	21
2.3.3	LEADTOOLS HTML5 Zero-footprint Medical Viewer.....	21
2.4	Βιβλιογραφία.....	22
3	Τεχνολογικό Υπόβαθρο	25
3.1	Τεχνολογίες Πληροφορικής και Επικοινωνιών	25
3.1.1	Φυλλομετρητές ιστού (Web browsers)	25
3.1.2	Ανάπτυξη web applications	27
3.1.3	Συμβατότητα web browsers.....	29
3.1.4	Single Page Applications and Web frameworks	31
3.1.5	Γραφικά για εφαρμογές ιστού.....	34
3.1.6	Δισδιάστατα γραφικά.....	35
3.1.7	Τρισδιάστατα γραφικά	35
3.1.7.1	Προ-WebGL	36
3.1.7.2	WebGL	36

3.1.8	Ταυτοχρονισμός στη JavaScript.....	40
3.1.8.1	Ο βρόχος συμβάντων (event loop) της JavaScript.....	40
3.1.8.2	Run-to-completion.....	41
3.1.8.3	Web Workers.....	41
3.1.9	WebRTC.....	43
3.1.9.1	Σύντομη ιστορική προσέγγιση.....	43
3.1.9.2	Το WebRTC ως τεχνολογία.....	44
4	Προτεινόμενο σύστημα.....	47
4.1	Τεχνικές Προδιαγραφές.....	47
4.1.1	Λειτουργικές απαιτήσεις.....	47
4.1.2	Σύντομη περιγραφή των χαρακτηριστικών της λύσης.....	48
4.2	Αρχιτεκτονική πλατφόρμας.....	49
4.3	Αρχιτεκτονική εφαρμογής πελάτη.....	51
4.4	Θέματα υλοποίησης.....	55
4.4.1	Διατήρηση και αναπαράσταση κατάστασης εφαρμογής πελάτη.....	55
4.4.2	Άνοιγμα αρχείων DICOM, parsing και εξαγωγή εικόνας.....	58
4.4.3	Ιατρική απεικόνιση και επεξεργασία.....	63
4.4.4	Επικοινωνία μέσω WebRTC.....	66
5	Το σύστημα στην πράξη.....	71
5.1	Παρουσίαση της διεπαφής χρήστη.....	71
5.1.1	SPA Layout.....	71
5.1.2	Παρεχόμενη λειτουργικότητα διαχείρισης ιατρικής εικόνας.....	72
5.2	Σενάρια χρήσης και αξιολόγηση.....	76
6	Επίλογος.....	79
6.1	Σύνοψη και συμπεράσματα.....	79
6.2	Μελλοντικοί στόχοι.....	79
7	Βιβλιογραφία.....	81

Εικόνες

Εικόνα 1. Από πραγματικές οντότητες στα DICOM IODs. Κάθε IOD είναι μια συλλογή γνωρισμάτων	8
Εικόνα 2. DICOM υπηρεσίες	9
Εικόνα 3. Τα 4 επίπεδα της ιεραρχίας πληροφοριών στο DICOM. Κάθε επίπεδο προσδιορίζεται αμφιμονοσήμαντα από ένα στοιχείο-κλειδί	11
Εικόνα 4. Μεγεθύνοντας τα εικονοστοιχεία ιατρικής εικόνας	13
Εικόνα 5. Αποθήκευση δειγμάτων ανά εικονοστοιχείο έγχρωμης (RGB) εικόνας	14
Εικόνα 6. Αποτελέσματα υπερβολικής απωλεστικής συμπίεσης σε εικόνες και γραμματοσειρές. Στα αριστερά, η συμπίεση JPEG παράγει τα λεγόμενα blocking artifacts. Στα δεξιά, η συμπίεση JPEG2000 παράγει έντονη θολότητα.....	18
Εικόνα 7. Το WADO ως υπηρεσία	20
Εικόνα 8. Αξονική τομογραφία αποκαλύπτει ρήξη ανευρύσματος της κοιλιακής αορτής	20
Εικόνα 9. Μεριδίο αγοράς browsers - Μάιος 2015 (εικόνα από www.w3counter.com [26])	25
Εικόνα 10. Ιστορικό μεριδίου χρήσης browsers (εικόνα από www.w3counter.com [26])	25
Εικόνα 11. Τάση της αγοράς browsers (στοιχεία από www.w3schools.com [27])	26
Εικόνα 12. Επίδοση κλήσεων του Canvas API μεταξύ των κυριότερων browsers στην πλατφόρμα των Windows.....	27
Εικόνα 13. Γραφική απεικόνιση απλού HTML αρχείου (εικόνα από [29])	28
Εικόνα 14. Άποψη ιστοσελίδας με χρήση και χωρίς χρήση CSS	28
Εικόνα 15. Οργάνωση της βάσης δεδομένων, του εξυπηρετητή και του πελάτη	32
Εικόνα 16. Συνήθεις αλληλεπιδράσεις των MVC συστατικών.....	33
Εικόνα 17. Screenshot της εφαρμογής "100,000 stars" από τη Google.....	37
Εικόνα 18. Η διασωλήνωση απεικόνισης (rendering pipeline) του WebGL.....	38
Εικόνα 19. Βρόχος συμβάντων της JavaScript.....	41
Εικόνα 20. Web worker και κύριο νήμα εκτέλεσης.....	42
Εικόνα 21. Κοινή τοπολογία βασισμένη σε WebRTC επικοινωνία	45
Εικόνα 22. Συνήθεις τοπολογίες διασύνδεσης	48
Εικόνα 23. Γενική άποψη της αρχιτεκτονικής.....	50
Εικόνα 24. Ακολουθιακό διάγραμμα διαδικασίας εγκαθίδρυσης κλήσης.....	51
Εικόνα 25. Η αρχιτεκτονική της SPA	52
Εικόνα 26. Η αντιστοιχία της εσωτερικής οργάνωσης του SH με αυτή του DICOM ..	53
Εικόνα 27. Ακολουθιακό διάγραμμα αλληλεπίδρασης μεταξύ των μονάδων της SPA	54
Εικόνα 28. Ο SH υλοποιημένος με Backbone models και collections.....	57
Εικόνα 29. Διάγραμμα των Backbone views που εμπλέκονται στην απεικόνιση των ιατρικών εικόνων	58
Εικόνα 30. Ροή εργασίας για την προβολή αρχείων DICOM	60
Εικόνα 31. Σχηματική αναπαράσταση της ακολουθίας μονοχρωματικών μετασχηματισμών κατά DICOM.....	60
Εικόνα 32. Παράδειγμα "windowing".....	62
Εικόνα 33. Αλληλεπίδραση μεταξύ κύριου νήματος εκτέλεσης και Web Worker	63
Εικόνα 34. Διαδικασία εφαρμογής φίλτρων μέσω του Canvas API	64
Εικόνα 35. Υβριδική υλοποίηση του DIV.....	65

Εικόνα 36. Σύγκριση επιδόσεων μεταξύ WebGL-based και Canvas-based υλοποίησης. Το διάγραμμα απεικονίζει το συνολικό χρόνο φόρτωσης (από το άνοιγμα μέχρι την εμφάνιση των εικόνων).	66
Εικόνα 37. Κλιμάκωση των καναλιών επικοινωνίας όπως παρέχεται από το WebRTC. Άμεση αμφίδρομη σύνδεση μεταξύ δύο χρηστών, αμφίδρομες συνδέσεις σε πλέγμα για περισσότερους.	67
Εικόνα 38. Ο browser ζητά πάντα την άδεια του χρήστη προκειμένου να αποκτήσει πρόσβαση στην κάμερα και το μικρόφωνο.	68
Εικόνα 39. Διαδικασία απομακρυσμένου συγχρονισμού του SH πάνω από RTCDatChannel	70
Εικόνα 40. Γενική διάταξη της SPA	72
Εικόνα 41. Υποστήριξη για άνοιγμα τοπικών αρχείων, καταλόγων και απομακρυσμένων (από WADO server)	72
Εικόνα 42. Πλοήγηση μέσω thumbnails σε επίπεδο σειράς.....	73
Εικόνα 43. Βασική εργαλειοθήκη	74
Εικόνα 44. Λειτουργία αυξομείωσης φωτεινότητας και αντίθεσης εικόνας.....	74
Εικόνα 45. Μετρήσεις ROI	75
Εικόνα 46. Modal view με τα περιεχόμενα του αρχείου DICOM.....	75
Εικόνα 47. Απεικόνιση μηχανισμού εγκαθίδρυσης συνεδριών	76
Εικόνα 48. Παράδειγμα JSON αντικειμένου που περιέχει πληροφορία για την απεικόνιση ιατρικής σειράς.....	77
Εικόνα 49. Εκτίμηση επίδοσης του RTCDatChannel: Ποσοστό μείωσης χρόνου μεταφοράς DICOM αρχείων σε σχέση με κεντρικοποιημένες αρχιτεκτονικές	77
Εικόνα 50. Αποτελέσματα SUS ερωτηματολογίου για την ευχρηστία της εφαρμογής πελάτη - SPA.....	78

Πίνακες

Πίνακας 1. Μερικές γραμμές από το λεξικό DICOM.....	10
Πίνακας 2. Μερικά από τα πιο συνήθη γνωρίσματα εικόνας που περιέχονται στο λεξικό του DICOM.....	15
Πίνακας 3. Τυπικά μεγέθη ιατρικών εικόνων και μελετών	16
Πίνακας 4. Υποστηριζόμενα Transfer Syntaxes	59

1 Εισαγωγή

Η σύγχρονη τάση στην παροχή υπηρεσιών υγείας είναι η χρησιμοποίηση ιατρικών πολυεπιστημονικών ομάδων (MDT - MultiDisciplinary Teams) για τη βελτίωση του ποσοστού επιτυχίας της θεραπευτικής αγωγής και την ισχυροποίηση της προληπτικής ιατρικής, αξιοποιώντας παράλληλα ιατρικούς πόρους, υλικούς και άυλους, αποδοτικότερα. Τα μέλη των εν λόγω ομάδων είναι αμοιβαίως υπεύθυνα τόσο για το σχεδιασμό στρατηγικών όσο και την εκτέλεσή τους σε κάθε στάδιο της διαδικασίας. Η κλασική σχέση (μοναδικού) ιατρού-ασθενή σταδιακά αλλάζει μορφή. Καθότι απαιτούνται δύσκολες και εξειδικευμένες δράσεις για την αντιμετώπιση περίπλοκων ιατρικών περιστατικών, οι ιατροί σήμερα αναγνωρίζουν πως η συμμετοχή περισσότερων ειδικοτήτων στη φροντίδα των ασθενών αυξάνει τις πιθανότητες επιτυχούς ανάρρωσης των τελευταίων. Αυτή η ολιστική προσέγγιση από την πλευρά των επιστημόνων υγείας πλέον βρίσκει εφαρμογή στο ευρύ φάσμα των παρόχων ιατροφαρμακευτικής περίθαλψης (από κλινικές και μεγάλες νοσοκομειακές εγκαταστάσεις μέχρι ιδιωτικά ιατρεία). Ως φυσικό ακόλουθο, η συμμετοχή ατόμων διαφορετικών ειδικοτήτων σε αυτές τις ομάδες προϋποθέτει την άμεση αλληλεπίδραση μεταξύ τους στα πλαίσια των συναντήσεων πολυεπιστημονικών ομάδων (MDTM – MultiDisciplinary Team Meetings).

1.1 Προσέγγιση της ιατρικής συνεργατικότητας

Έχει αποδειχθεί ότι η ποιοτική και απρόσκοπτη επικοινωνία και αλληλεπίδραση είναι ουσιώδης για τη βελτίωση της ποιότητας της συνεργασίας μεταξύ των μελών μιας πολυεπιστημονικής ομάδας, ενώ υποδομές όπως συστήματα τηλεδιάσκεψης ενισχύουν περαιτέρω την αποδοτικότητά της [1]. Ένας σημαντικός όγκος εργασίας στον τομέα της συνεργατικότητας βοηθούμενης από υπολογιστή (computer-aided collaboration) όσον αφορά τις κατανομημένες συνεδρίες πολυεπιστημονικών ομάδων έχει επικεντρωθεί στη διερεύνηση του κατά πόσο είναι εφικτή μια τέτοια συνεδρία με χρήση video-conferencing τεχνικών, αντιπαραβάλλοντάς τη με μια κλασική πρόσωπο με πρόσωπο επικοινωνία [2] [3] [4] [5]. Ως αποτέλεσμα αυτής της δουλειάς, τεκμηριώθηκε η διευκόλυνση που παρέχουν τα οπτικοακουστικά μέσα στην πολυεπιστημονική διαχείριση ιατρικών υποθέσεων. Συγκεκριμένα, οι τεχνικές αυτές βρέθηκε πως επηρεάζουν τον τρόπο με τον οποίο διαβουλεύονται οι ιατροί καθώς και το πώς εκλαμβάνουν το συσχετισμό κύρους και εξουσίας αφού έχουν χαρακτήρα πιο επίσημο και οργανωμένο σε σχέση με τις κανονικές [2]. Ιδιαίτερη αξία έχει η δυνατότητα παρουσίασης ιατρικών ευρημάτων σε αυτές [3]. Οι βιντεοδιασκέψεις, παρ' όλα αυτά ίσως είναι λιγότερο αποδοτικές και διαρκούν περισσότερο χρόνο. Δυσκολίες σχετιζόμενες με το συντονισμό και τη συναισθηση σε αυτού του είδους τα περιβάλλοντα οφείλονται σε τεχνικούς όπως και κοινωνικο-συναισθηματικούς και οργανωτικούς παράγοντες [3] [4].

Έχει υπάρξει ένα μεγάλο εύρος από συστήματα συνεργατικότητας που παρέχουν απομακρυσμένη επίγνωση κατάστασης και διαμοιραζόμενα ιατρικά ευρήματα σε μια ποικιλία από περιβάλλοντα. Η αρχική ενασχόληση με το λεγόμενα «media spaces»¹

¹ Ως media space ορίζεται ο ηλεκτρονικός «χώρος» στον οποίο ομάδες ανθρώπων μπορούν να εργαστούν συλλογικά ακόμα και όταν δε βρίσκονται στον ίδιο χώρο και χρόνο. Σε έναν τέτοιο χώρο, οι άνθρωποι μπορούν να δημιουργήσουν πραγματικού χρόνου οπτικά και ακουστικά περιβάλλοντα τα οποία εκτείνονται σε γεωγραφικά απομακρυσμένες περιοχές. Επίσης, δύνανται

είχε ως σκοπό την παροχή επίγνωσης σε καταναλωμένες ομάδες εργασίας διαμέσου βίντεο, ήχου και διαμοιραζόμενων υπολογιστικών πόρων [6]. Έμφαση δίνεται στην ανάγκη οι συμμετέχοντες που αλληλεπιδρούν να είναι σε θέση να γνωρίζουν τι κάνουν οι υπόλοιποι όπως και να έχουν πρόσβαση στα ευρήματα μέσα στον κοινό χώρο εργασίας. Επιπρόσθετα, οι απαιτήσεις αυτές πρέπει «ρητά και σκόπιμα» να υποστηρίζονται από ένα τέτοιο συνεργατικό σύστημα [7]. Μέσα στον τομέα της υγείας υπάρχουν λοιπόν συστήματα που προσφέρουν χαρακτηριστικά κοινωνικής και χωρικής συναίσθησης για κλινικούς ιατρούς όσον αφορά την τρέχουσα εργασία και την τοποθεσία των συναδέλφων τους μέσα σε ιατρικούς χώρους [8]. Επίσης, έχει γίνει επίδειξη ενός διαδραστικού τηλεκατευθυνόμενου συστήματος εστιασμένου στην υποστήριξη διευθέτησης χώρου και αλληλεπίδρασης με ιατρικές εικόνες.

Η υποστήριξη μιας γεωγραφικά καταναλωμένης ιατρικής ομάδας απαιτεί τις κατάλληλες συνεργατικές τεχνολογίες προσαρμοσμένες πλήρως στο εργασιακό περιβάλλον για να επιτρέψει την ομάδα να επιτύχει τους στόχους της. Έχει γίνει όλο και περισσότερο κατανοητό το γεγονός ότι η επιτυχία μιας οποιασδήποτε τέτοιου είδους τεχνολογίας επαφίεται στην πλήρη κατανόηση της αλληλεπίδρασης ανθρώπου-μηχανής που η χρήση της τεχνολογίας αυτής συνεπάγεται [9], και μια κοινωνικοτεχνική προσέγγιση στη σχεδιαστική διαδικασία [10] [11].

1.2 Σκοπός της παρούσας εργασίας

Παρότι είναι κοινώς αποδεκτό ότι η διαχείριση των ασθενών από πολυεπιστημονικές ομάδες γενικά οδηγεί σε καλύτερα αποτελέσματα από την πλευρά των ασθενών, υπάρχουν κάποιοι παράγοντες που λειτουργούν ως τροχοπέδη στην εκμετάλλευση της πλήρους δυναμικής που αναπτύσσεται από τη νέα αυτή τάση. Χαρακτηριστικά αναφέρονται οι χρονικοί περιορισμοί ως απόρροια του διαφορετικού προγράμματος των ιατρών και οι ανεπαρκείς υποδομές και εγκαταστάσεις που δυσχεραίνουν την επικοινωνία μεταξύ τους (π.χ. έλλειψη εξειδικευμένου ηλεκτρονικού εξοπλισμού για υποστήριξη τηλεδιάσκεψης). Επιπρόσθετα, η πραγματοποίηση συναντήσεων σε μικρά νοσοκομεία ή κλινικές μπορεί ακόμα και να αποθαρρυνθεί καθώς οι ειδικοί που απαιτούνται δε βρίσκονται σε αυτά, ενώ επίσης δεν πρέπει να παραλείπεται το γεγονός ότι ορισμένα περιστατικά επιβάλλουν διαδοχικές συνεδρίες για τη βεβαίωση πως αποφάσεις που ελήφθησαν αρχικά υλοποιούνται στην πορεία καταλλήλως.

Καθώς η παροχή υπηρεσιών υγείας συνεχίζει να γίνεται όλο και πιο κεντροποιημένη και εξειδικευμένη, σύγχρονες τεχνολογίες υπολογιστών και δικτύων άρχισαν να υιοθετούνται από τους φορείς υγείας για την υποστήριξη των πολυεπιστημονικών ομάδων και την αντιμετώπιση ορισμένων εκ των προαναφερθέντων δυσκολιών. Η διάχυσή τους όμως κρίνεται περιορισμένη – συναντώνται κυρίως σε μεγάλες νοσοκομειακές εγκαταστάσεις και ιδιωτικές κλινικές, και ως επί το πλείστον πρόκειται για ιδιόκτητες λύσεις που συνεπάγονται και την απόκτηση του αντίστοιχου υλισμικού/λογισμικού. Επίσης, οι δυνατότητες απομακρυσμένης συνεργατικότητας που παρέχονται συχνά περιορίζονται στην απλή μετάδοση βίντεο και ήχου. Έρευνες ωστόσο έχουν αποδείξει ότι

να ελέγχουν την εγγραφή, την πρόσβαση και την αναπαραγωγή εικόνων και ήχων από τα εν λόγω περιβάλλοντα.

εκτός από την άμεση επικοινωνία και διαβούλευση μεταξύ των ιατρών, κομβική σημασία για μία εποικοδομητική διάσκεψη έχει η αλληλεπίδραση με ιατρικές εικόνες, σημειώσεις και αποτελέσματα εξετάσεων (βλ. 1.1).

Η ερευνητική κοινότητα έχει επίσης να επιδείξει προτάσεις και καινοτομίες στο συγκεκριμένο τομέα. Ωστόσο, εκτός ορισμένων περιπτώσεων, οι πλατφόρμες που εισάγουν, όπως και οι αντίστοιχες εμπορικές, ακολουθούν το κλασικό κεντροποιημένο παράδειγμα (client-server paradigm), μια λογική σχεδιαστική επιλογή δεδομένου ότι απευθύνονται αυστηρά στο προσωπικό ενός οργανισμού (π.χ. κεντρικού νοσοκομείου) με σκοπό τη μεταξύ του διασύνδεσή ή την επικοινωνία με συναδέλφους κοντινών αστικών ιατρικών κέντρων. Εντούτοις, στην πράξη η συνεργασία έχει αποδειχθεί ότι είναι μια εγγενώς περίπλοκη διαδικασία και επιβάλλει μια ποικιλία από προκλήσεις και απαιτήσεις. Οι κυριότερες από αυτές σχετίζονται με την ευελιξία, την επεκτασιμότητα και την ανθεκτικότητα, χαρακτηριστικών που δεν επιτυγχάνονται απόλυτα με τις κεντροποιημένες αρχιτεκτονικές. Η ασφάλεια και η ιδιωτικότητα είναι επίσης σημαντικά ζητήματα καθώς τα ιατρικά δεδομένα είναι ευαίσθητης φύσεως και θα πρέπει να είναι διαθέσιμα μόνο σε εξουσιοδοτημένους συμμετέχοντες στο συνεργατικό περιβάλλον σύμφωνα με συγκεκριμένες πολιτικές και κανόνες.

1.3 Προτεινόμενη λύση

Στο σημείο αυτό, το διομότιμο παράδειγμα (peer-to-peer paradigm) φαίνεται να αποτελεί την πιο εποικοδομητική υπολογιστική προσέγγιση με σκοπό την κάλυψη των απαιτήσεων της ιατρικής κοινότητας για παροχή συνεργατικών υπηρεσιών. Η ποιότητα των παρεχόμενων ιατροφαρμακευτικών υπηρεσιών βασίζονται στη συνεργασία μεταξύ οντοτήτων (δηλαδή ιατρικού και παραϊατρικού προσωπικού, κτλ.) που συνιστούν ένα μεγάλης κλίμακας καταναμημένο σύστημα, το οποίο ταυτίζεται ακριβώς με την ίδια την έννοια της διομότιμης δικτύωσης. Η επίβλεψη της συνεργασίας πληθώρας ομάδων με δυναμική συμπεριφορά δείχνει να είναι αδύνατη για τις σημερινές κεντρικές οντότητες. Τα διομότιμα συστήματα παρέχουν αυτοοργανούμενα χαρακτηριστικά που μπορούν να συνεισφέρουν στην εξέλιξη μεγάλων, καταναμημένων και δυναμικών συστημάτων. Επιπρόσθετα, αυτού του είδους τα συστήματα παρουσιάζουν ανθεκτικότητα (fault tolerance) και ευρωστία (robustness), αφού μπορούν να ανταπεξέλθουν καλύτερα σε αποτυχίες. Ενσωματώνουν μηχανισμούς που «απορροφούν» ανωμαλίες εισαγόμενες από μεμονωμένες ομότιμες οντότητες (peers) και κρατούν πολλαπλά αντίγραφα που εγγυώνται ότι η ζητούμενη πληροφορία θα είναι πάντα διαθέσιμη. Επίσης, κατά τη διάρκεια εγκαθίδρυσης μιας τηλεσυνεδρίας, είναι ιδιαίτερα σημαντικό για τους επαγγελματίες υγείας να εξοικονομούν όσο περισσότερο χρόνο γίνεται και η πρόσβασή τους σε ιατρικά δεδομένα και πληροφορία να γίνεται με συνεπή και αξιόπιστο τρόπο. Οι χαμηλοί χρόνοι απόκρισης με εγγυημένα αποτελέσματα που προσφέρουν τα peer-to-peer δίκτυα μπορούν να συνεισφέρουν προς την κατεύθυνση αυτή. Από τα παραπάνω γίνεται προφανές πως τα ιδιαίτερα γνωρίσματα των δικτύων αυτών είναι άκρως ωφέλιμα για τον τομέα της υγείας αφού οι αλληλεπιδράσεις μεταξύ των πολλαπλών ενδιαφερομένων είναι ευθυγραμμισμένες με τη peer-to-peer λογική. Δοθέντος των στοιχείων αυτών, αποτελούν και την πιο επικερδή λύση σήμερα για τη δόμηση μιας πλατφόρμας που θα υποστηρίξει τη συνεργασία και τη ροή πληροφορίας μεταξύ ετερογενών ιατρικών ομάδων.

Σκοπός λοιπόν της παρουσιαζόμενης εργασίας είναι ο σχεδιασμός και η υλοποίηση μιας open source συνεργατικής πλατφόρμας βασισμένης σε τελευταίες τεχνολογίες ιστού (βλ. HTML5, CSS3, JavaScript) και διεπαφές προγραμματισμού εφαρμογών – APIs (βλ. WebRTC) με σκοπό την παροχή φιλικών προς το χρήστη απομακρυσμένων ιατρικών συνεδριών πραγματικού χρόνου. Προκειμένου να αξιοποιηθεί η ευρεία διάδοση των μοντέρνων «έξυπνων» συσκευών που συνεχώς αποκτούν περισσότερη υπολογιστική ισχύ και ακολουθώντας τη λογική του BYOD (Bring Your Own Device), επιλέχθηκε ο browser ως το περιβάλλον στο οποίο θα τρέχει η εφαρμογή του πελάτη. Παράλληλα, δε θυσιάζεται τίποτα από άποψη παρεχόμενης λειτουργικότητας, αφού καλύπτονται πλήρως οι απαιτήσεις του ιατρικού προσωπικού για δυνατότητα απεικόνισης ιατρικής εικόνας και ανταλλαγής ιατρικών αναφορών.

1.4 Οργάνωση κειμένου

Στο Κεφάλαιο 2 παρατίθεται μια σύντομη εισαγωγή στα προτεινόμενα πρότυπα του χώρου της υγείας, μερικά εκ των οποίων έχουν άμεση σχέση με το αντικείμενο της διπλωματικής, παρουσιάζονται κάποιες εφαρμογές ιατρικής απεικόνισης και εξετάζεται η σχετική βιβλιογραφία και έρευνα που έχει γίνει στο χώρο της ιατρικής συνεργατικότητας. Στο Κεφάλαιο 3 επιχειρείται μια προσέγγιση των ποικίλων τεχνολογιών ιστού που χρησιμοποιήθηκαν για την εξέλιξη της πλατφόρμας, ενώ παράλληλα τεκμηριώνεται η σημασία τους και ο λόγος για τον οποίο επιλέχθηκαν. Επίσης, παρουσιάζονται κάποιες κεντρικές έννοιες του προτύπου DICOM και της ιατρικής απεικόνισης που βρίσκονται στο επίκεντρο της υλοποίησης. Το Κεφάλαιο 4 περιέχει την ανάλυση και το σχεδιασμό του ευρύτερου πλαισίου της παρεχόμενης υπηρεσίας, ενώ γίνεται η παρουσίαση της αρχιτεκτονικής τόσο της γενικότερης πλατφόρμας όσο και της εφαρμογής πελάτη. Το Κεφάλαιο 5 ασχολείται με πρακτικά ζητήματα, όπως η διεπαφή χρήστη, ενώ παράλληλα θίγονται σημαντικές λεπτομέρειες και προκλήσεις κατά την υλοποίηση. Επίσης, αν και η τελική εφαρμογή έχει καινοτόμες πτυχές και δεν υπάρχει η δυνατότητα άμεσης σύγκρισης με κάποια αντίστοιχη, επιχειρείται μια σύντομη αξιολόγηση που τεκμηριώνει την αξία της. Στο Κεφάλαιο 6 παρατίθεται μια αποτίμηση της συνολικής εργασίας καθώς επίσης και κάποιοι προβληματισμοί και ιδέες που μπορούν να οδηγήσουν σε μελλοντικές επεκτάσεις της λειτουργικότητας. Τέλος, το Κεφάλαιο 7 αποτελείται από τη βιβλιογραφία που χρησιμοποιήθηκε κατά τη συγγραφή.

2 Σχετικά πρότυπα, εφαρμογές και βιβλιογραφία

2.1 Πρότυπα ανταλλαγής ιατρικών δεδομένων

Σήμερα, οι περισσότερες πληροφορίες που αφορούν έναν ασθενή είναι αποθηκευμένες σε διαφορετικά συστήματα διασκορπισμένα στην ιατρική κοινότητα – γραφεία ιατρών, διαγνωστικές ή ακτινολογικές κλινικές, νοσοκομεία, πολλά εκ των οποίων δεν υποστηρίζουν εγγενώς τη διαλειτουργικότητα. Ένας ιδιώτης ιατρός μπορεί να δυσκολευτεί να αποκτήσει ολοκληρωμένες πληροφορίες σχετικά με κάποιον ασθενή του που νοσηλεύεται σε νοσοκομείο ή να επαναλάβει ελέγχους και εξετάσεις λόγω ελλειψών πληροφοριών για την πρότερη κατάσταση του ασθενούς. Επιπρόσθετα στις ποιοτικές, προκλήσεις επιβάλλονται και από τις γραφειοκρατικές διαδικασίες, οι οποίες κοστίζουν από άποψη χρόνου αλλά και χρημάτων. Ενδεικτικό είναι το παράδειγμα ενός ασθενούς που αναγκάζεται προσωπικώς να παραδώσει αντίγραφο του ιατρικού του φακέλου σε κάποιον άλλο ιατρό και ο τελευταίος, αν χρειαστεί κάποια δεύτερη γνώμη, να το αποστείλει με φαξ ή αλληλογραφία σε συνάδελφό του. Ακόμα περισσότερες δυσκολίες ανακύπτουν αν στην παραπάνω διαδικασία εμπλέκονται και ιατρικές εικόνες (π.χ. ακτινογραφίες). Ο τομέας, λοιπόν, των επαγγελματιών υγείας, προκειμένου να επιλύσει αυτά τα θέματα, έχει ερευνήσει μεθόδους επίτευξης διαλειτουργικότητας για χρόνια. Γεγονός είναι πως αυτές οι προσεγγίσεις απέφεραν αποτελέσματα, κυρίως με τη μορφή προτασόμενων προτύπων. Τα κυριότερα από αυτά θα εξεταστούν στη συνέχεια ενώ κάποια από αυτά θα χρησιμοποιηθούν και στα πλαίσια της παρούσας διπλωματικής εργασίας.

2.1.1 Το πρότυπο HL7

Το πρότυπο HL7 το οποίο έχει αναπτυχθεί από τον ομώνυμο οργανισμό (www.hl7.org) είναι το πλέον ώριμο και ευρέως χρησιμοποιημένο πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων στο χώρο της υγείας. Στη δημιουργία του οδήγησε η έρευνα τόσο από την ακαδημαϊκή κοινότητα όσο και από την βιομηχανία και τις εταιρίες συμβούλων. Η κυριότητά του ανήκει στο μη κερδοσκοπικό οργανισμό Health Level 7 και έχει αναγνωριστεί από πολλά εθνικά ιδρύματα προτυποποίησης όπως ο ANSI (Η.Π.Α.) και ο DIN (Γερμανία).

Το HL7 είναι ένα πρωτόκολλο επικοινωνίας που μπορεί να εφαρμοστεί τόσο σε νοσοκομεία και εργαστήρια όσο και σε μονάδες οργάνωσης, διοίκησης και διαχείρισης των υπηρεσιών υγείας κάθε χώρας. Αυτό το πετυχαίνει διότι εξασφαλίζει την ηλεκτρονική επικοινωνία ετερογενών πληροφοριακών συστημάτων ανταλλάσσοντας δεδομένα (μέσω HL7 μηνυμάτων). Τα πληροφοριακά αυτά συστήματα μπορεί να υποστηρίζουν διαφορετικές λειτουργικές μονάδες ενός οργανισμού υγείας ή ακόμη και να ανήκουν σε διαφορετικούς οργανισμούς υγείας. Το πρότυπο HL7 λοιπόν είναι ένας κοινά αποδεκτός από όλους τους κατασκευαστές κώδικας επικοινωνίας.

Με τη χρήση του προτύπου, για παράδειγμα, μπορεί ένας αναλυτής σε εργαστήριο νοσοκομείου να δέχεται απευθείας εντολές εξετάσεων από τα κλινικά τμήματα και να επιστρέφει τις απαντήσεις των εξετάσεων που διενεργεί στα τμήματα που τις παρήγγειλαν αυτόματα. Έτσι υποβοηθείται σημαντικά ένας κλινικός γιατρός αφού απαλ-

λάσσεται από το φόρτο της χειρωνακτικής διαχείρισης τεράστιου όγκου ιατρικής πληροφορίας, που απορροφά σημαντικό χρόνο και τον αποσπά από τον πρωταρχικό σκοπό του, τη διάγνωση και θεραπεία του ασθενή του.

Το πρότυπο HL7 δεν αφορά αποκλειστικά τη διαβίβαση πληροφορίας μεταξύ εργαστηρίου και κλινικής. Είναι έτσι δομημένο που εκτός από κλινικά και εργαστηριακά δεδομένα εμπεριέχει και όλες τις υπαρκτές πληροφορίες σε κάθε μονάδα υγείας δηλαδή ασφαλιστικά και οικονομικά στοιχεία, προμήθειες και διαχείριση υλικών, φαρμάκων και εργαλείων, αναλώσιμων και πάγιου εξοπλισμού.

Το μόνο που απαιτείται είναι η φυσική διασύνδεση των συστημάτων και το κάθε τμήμα μπορεί να έχει τα στοιχεία που του είναι απαραίτητα για τη λειτουργία του. Έτσι διεκπεραιώνεται αυτόματα το υπόλοιπο πλην του κλινικού έργο και αποφεύγεται εντελώς η γραφειοκρατία εφόσον μία και μοναδική εγγραφή για κάθε ασθενή μπορεί να διανέμεται εύκολα και κατάλληλα σε κάθε τμήμα, κλινικό, εργαστηριακό ή διοικητικό ανάλογα με τις ανάγκες του τμήματος. Το ίδιο εύκολη είναι και η διαδικασία της ενημέρωσης της κάθε εγγραφής αφού αρκεί να γίνει αυτή η διαδικασία από ένα μόνο τμήμα. Γι' αυτό και το πρότυπο HL7 επικρατεί αφού αντιμετωπίζει ένα νοσοκομείο ή ένα ευρύτερο σύστημα, σαν ενιαία λειτουργική οντότητα, όπως πράγματι είναι.

Το πρότυπο HL7 μπορεί να εγκατασταθεί και να λειτουργήσει στα ήδη υπάρχοντα πληροφοριακά συστήματα και στον ήδη υπάρχοντα ιατροτεχνολογικό εξοπλισμό. Δεν απαιτεί καμία αλλαγή και διασυνδέει τα συστήματα και τα μηχανήματα κάθε κατασκευαστή. Ό,τι είναι ήδη εγκατεστημένο σε ένα νοσοκομείο ή μια μονάδα υγείας, από πλευράς τεχνολογικού εξοπλισμού, κάθε είδους, με την χρήση του προτύπου HL7 συνδέεται και με τον ολόκληρο το υπόλοιπο εξοπλισμό.

2.1.2 EMR/EHR

Ως EHR (Electronic Health Record), ή EMR (Electronic Medical Record), ορίζεται μια συστηματική συλλογή από ηλεκτρονικές πληροφορίες υγείας σχετιζόμενες με ένα συγκεκριμένο ασθενή ή πληθυσμό. Είναι μια καταχώρηση σε ψηφιακή μορφή που θεωρητικά είναι σε θέση να διαμοιραστεί μεταξύ διαφορετικών φορέων υγείας. Σε μερικές περιπτώσεις, αυτός ο διαμοιρασμός υλοποιείται διαμέσου δικτυακά διασυνδεδεμένων πληροφοριακών συστημάτων σε επιχειρησιακό επίπεδο. Τα EHRs μπορεί να περιλαμβάνουν ένα μεγάλο εύρος από δεδομένα, συμπεριλαμβανομένων δημογραφικών στοιχείων, ιατρικού ιστορικού, φαρμακευτικής αγωγής και αλλεργιών, εργαστηριακών εξετάσεων, ιατρικών εικόνων, ζωτικών δεικτών, προσωπικών στοιχείων όπως ηλικία και βάρος και πληροφοριών χρέωσης παρεχόμενων ιατρικών υπηρεσιών.

Το όλο σύστημα είναι σχεδιασμένο να αναπαριστά δεδομένα που περιγράφουν με ακρίβεια την κατάσταση του ασθενούς την εκάστοτε χρονική στιγμή. Επιτρέπει την προβολή ολόκληρου του ιστορικού του ασθενούς χωρίς την ανάγκη για εντοπισμό προηγούμενων καταχωρήσεων και εξασφαλίζει πως τα εν λόγω δεδομένα είναι πάντα ενημερωμένα και έγκυρα. Μειώνει δε την πιθανότητα των διπλοτύπων, καθώς υπάρχει μία και μοναδική καταχώρηση ανά ασθενή. Για τον τελευταίο αυτό λόγο καθιστά επίσης την εξαγωγή χρήσιμης ιατρικής πληροφορίας περισσότερο αποδοτική.

2.1.3 Το πρότυπο DICOM και τα PACS

Το πρότυπο DICOM (Digital Imaging and Communications in Medicine) είναι υπεύθυνο για το χειρισμό, την αποθήκευση, την εκτύπωση και τη διάδοση ιατρικών

εικόνων. Περιλαμβάνει έναν ορισμό τύπου αρχείων και ένα δικτυακό πρωτόκολλο επικοινωνίας. Το τελευταίο αποτελεί στην ουσία ένα πρωτόκολλο επιπέδου εφαρμογής που χρησιμοποιεί το TCP/IP για την επικοινωνία μεταξύ συστημάτων. Τα δε αρχεία DICOM μπορεί να αποσταλούν μεταξύ οντοτήτων ικανών να λαμβάνουν εικόνες και δεδομένα ασθενών σε μορφή DICOM. Η κυριότητά του ανήκει στον οργανισμό NEMA (National Electrical Manufacturers Association) και εξελίσσεται από τη DICOM Standards Committee, της οποίας τα μέλη είναι επίσης μέλη της NEMA.

Συνοπτικά, το DICOM επιτρέπει την ενοποίηση σαρωτών, εξυπηρετητών, σταθμών εργασίας, εκτυπωτών και δικτυακού εξοπλισμού διαφορετικών κατασκευαστών σε ένα σύστημα γνωστό ως PACS (Picture Archiving and Communication System). Οι διαφορετικές συσκευές αποστέλλονται με δηλώσεις συμμόρφωσης με το πρότυπο στις οποίες ρητώς αναφέρουν ποιες κλάσεις του υποστηρίζουν. Το DICOM πλέον έχει ευρέως υιοθετηθεί από μεγάλες νοσοκομειακές εγκαταστάσεις και κλινικές ενώ αρχίζει να βρίσκει εφαρμογή και σε μικρότερους πάροχους υπηρεσιών υγείας (π.χ. ιδιωτικά γραφεία ιατρών).

Το σύστημα PACS παρέχει οικονομική αποθήκευση και πρακτική πρόσβαση σε εικόνες διαφορετικής προέλευσης (modality). Εικόνες και αναφορές μεταδίδονται σε ηλεκτρονική μορφή μέσω των PACS με ό,τι ευκολίες αυτό συνεπάγεται (αποφυγή διαχείρισης των αντίστοιχων αναλογικών). Η καθολικά αποδεκτή μορφή αποθήκευσης και μετάδοσης εικόνας είναι η DICOM, όπως αυτή παρουσιάστηκε ανωτέρω. Δεδομένα που δεν είναι εικόνες, όπως σαρωμένα έγγραφα, δύναται να ενσωματωθούν στην υπάρχουσα υποδομή χρησιμοποιώντας κάποια γνωστή μορφή αποθήκευσης (π.χ. PDF) αφού πρώτα ενθυλακωθούν σε πακέτα DICOM.

Ένα PACS αποτελείται από τέσσερα κύρια μέρη: την προέλευση της ιατρικής εικόνας (π.χ. ακτινογραφία, αξονική τομογραφία, μαγνητική τομογραφία), ένα ασφαλές δίκτυο για τη μετάδοση των δεδομένων των ασθενών, τους σταθμούς εργασίας για τη διερμηνεία, την παρουσίαση και την αξιολόγηση των εικόνων και το σύστημα αρχειοθέτησης για την αποθήκευση και ανάκτηση των εικόνων και των αναφορών. Σε συνδυασμό με τη διαθέσιμες αλλά και αναδυόμενες τεχνολογίες ιστού, τα PACS έχουν τη δυνατότητα να ανταποκριθούν σε έγκαιρη και αποδοτική πρόσβαση σε εικόνες, ερμηνείες και σχετιζόμενα ιατρικά δεδομένα. Συμπερασματικά, τα PACS καταργούν τους φυσικούς και χρονικούς περιορισμούς που σχετίζονται με την παραδοσιακή, βασισμένη σε φιλμ, ανάκτηση, διανομή και απεικόνιση εικόνας.

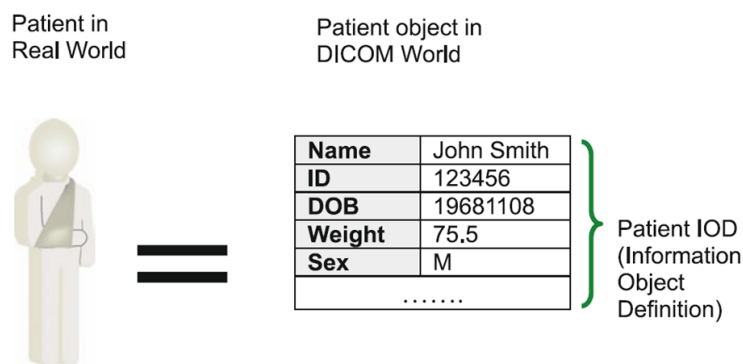
2.1.3.1 Εισαγωγή στο DICOM

Το πρότυπο DICOM είναι αχανές ως προς την έκτασή του, γεγονός που εν πολλοίς οφείλεται στο διπτό ρόλο που καλείται να επιτελέσει – ορισμός τύπου αρχείου και περιγραφή πρωτοκόλλου επικοινωνίας (βλ. 2.1.3). Για τους σκοπούς αυτής της εργασίας δεν κρίνεται σκόπιμη η αναλυτική προσέγγισή του, παρά προτιμάται η παράθεση μόνο κάποιων συγκεκριμένων κεντρικών εννοιών που συνδέονται άμεσα με τη λογική της παρουσιαζόμενης υλοποίησης. Για κάποιον που επιζητεί να εντρυφήσει στις λεπτομέρειες, στην ιστοσελίδα <http://dicom.nema.org/> είναι διαθέσιμα και τα είκοσι μέρη στα οποία υποδιαιρείται το πρότυπο.

2.1.3.2 Βασικές έννοιες

Για να εισάγει λογική και τάξη στο πολυσύνθετο ιατρικό περιβάλλον, το πρότυπο DICOM χρησιμοποιεί τη δική του τεχνική ορολογία, βασισμένη στο μοντέλο του για

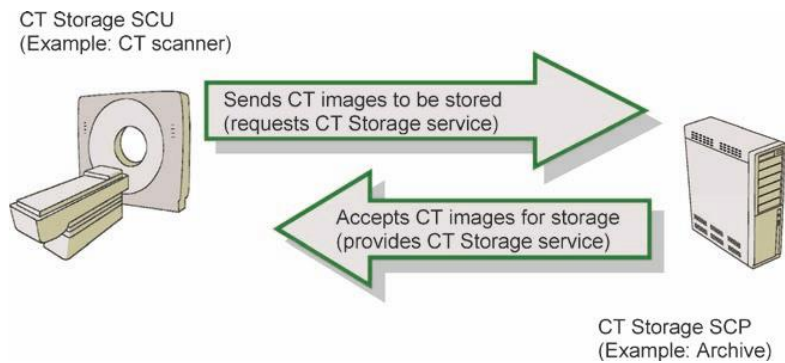
τον πραγματικό κόσμο, το γνωστό και ως *μοντέλο DICOM πληροφορίας (DICOM information model)*. Σύμφωνα με αυτό, όλα τα πραγματικά δεδομένα – ασθενείς, ιατρικές μελέτες, εικόνες, συσκευές, κτλ. – αντιμετωπίζονται από το πρότυπο ως αντικείμενα με αντίστοιχες ιδιότητες ή γνωρίσματα². Οι ορισμοί τέτοιων αντικειμένων και γνωρισμάτων είναι προτυποποιημένοι σύμφωνα με τα DICOM IODs (*Information Object Definitions*). Μπορεί κάποιος να προσεγγίσει τα IODs ως συλλογές γνωρισμάτων που περιγράφουν ένα συγκεκριμένο αντικείμενο. Το IOD *Ασθενής*, για παράδειγμα, μπορεί να περιγραφεί με βάση το όνομα, τον αριθμό του ιατρικού φακέλου, την ηλικία, το φύλο, το βάρος, κ.ό.κ. – όσα γνωρίσματα απαιτούνται για να σκιαγραφηθεί όλη η κλινικώς σχετική πληροφορία που τον αφορά. Υπό μια ευρεία έννοια, ένας ασθενής (όπως και κάθε άλλο DICOM αντικείμενο) είναι το σύνολο των χαρακτηριστικών/γνωρισμάτων από τα οποία αποτελείται, όπως απεικονίζεται στην *Εικόνα 1*. Για το σκοπό αυτό, το DICOM διατηρεί μια λίστα από όλα τα κοινώς αποδεκτά γνωρίσματα (συνολικά περισσότερα από 2000) που λειτουργεί ως λεξικό και διασφαλίζει τη συνέπεια της ονοματολογίας και επεξεργασίας τους. Όλα τους μορφοποιούνται σύμφωνα με 27 διαφορετικούς *τύπους αναπαράστασης τιμών (value representation – VR types)*, ανάλογα με τον εάν πρόκειται για ημερομηνίες, χρονικές στιγμές, ονόματα, αριθμούς, κτλ.



Εικόνα 1. Από πραγματικές οντότητες στα DICOM IODs. Κάθε IOD είναι μια συλλογή γνωρισμάτων

Μόλις τα πραγματικά δεδομένα μοντελοποιηθούν ως δεδομένα γνωρισμάτων DICOM, μπορούν να μεταδοθούν και να επεξεργαστούν μεταξύ διαφόρων συσκευών και λογισμικών (*Application Entities – AEs*, όπως ονομάζονται κατά το πρότυπο). Το DICOM αναπαριστά αυτού του είδους τις διαδικασίες χρησιμοποιώντας ένα μοντέλο προσανατολισμένο σε υπηρεσίες (*service-rendering model*): οι εφαρμογές DICOM παρέχουν υπηρεσίες η μία στην άλλη. Επειδή κάθε υπηρεσία συνήθως περιέχει κάποιου είδους ανταλλαγή δεδομένων (τυπικά πάνω από δίκτυο υπολογιστών), απορρέει με φυσικό τρόπο ο συσχετισμός συγκεκριμένων ειδών υπηρεσιών με τα δεδομένα (IODs) που αυτά επεξεργάζονται. Το DICOM ονομάζει αυτούς τους συσχετισμούς *Service-Object Pairs (SOPs)* και τους ομαδοποιεί σε *SOP Classes*. Για παράδειγμα, αποθηκεύοντας μια αξονική τομογραφία από έναν ψηφιακό αξονικό τομογράφο σε ένα PACS αντιστοιχεί στο λεγόμενο CT Storage SOP, όπως απεικονίζεται στην *Εικόνα 2*. Στο συγκεκριμένο παράδειγμα, η αξονική τομογραφία (ως εικόνα) αναπαριστά ένα DICOM IOD.

² Παρόμοια λογική εφαρμόζουν και αντικειμενοστραφείς γλώσσες προγραμματισμού όσον αφορά τη μοντελοποίηση των δεδομένων.



Εικόνα 2. DICOM υπηρεσίες

2.1.3.3 Το λεξικό DICOM

Στην ουσία, το λεξικό DICOM (DICOM Data Dictionary) περιέχει όλα τα συνήθη αντικείμενα δεδομένων (γνωρίσματα) που χρησιμοποιούνται στην «ψηφιακή» ιατρική, μορφοποιημένα σύμφωνα με κάποιο από τα 27 διαθέσιμα για το σκοπό αυτό VRs. Για την οργάνωση και ταξινόμηση της πληθώρας αυτής γνωρισμάτων, τα τελευταία διαχωρίζονται πρώτα σε αριθμημένες συλλογές με βάση γενικές ομοιότητες. Οι συλλογές αυτές τώρα αποτελούνται από τα μεμονωμένα στοιχεία/γνωρίσματα. Έτσι, σε κάθε αντικείμενο (γνωρίσμα ή στοιχείο) αντιστοιχεί ένα μοναδικό ζεύγος τιμών που έχει τη μορφή (group, element) (γνωστό και ως αναγνωριστικό-tag).

Τόσο τα σύνολα όσο και τα στοιχεία αριθμούνται με δεκαεξαδικούς αριθμούς. Ο Πίνακας 1 περιέχει ένα απόσπασμα του λεξικού. Όπως φαίνεται, η πρώτη στήλη περιλαμβάνει τη δεκαεξαδική τιμή του αναγνωριστικού. Η δεύτερη αντιστοιχεί στην ονομασία του εκάστοτε στοιχείου και ίσως έχει τη μεγαλύτερη σημασία, καθώς επεξηγεί ποια δεδομένα από τον πραγματικό κόσμο πρέπει να αναπαρασταθούν από το συγκεκριμένο στοιχείο. Ξεκάθαρα, τα αναγνωριστικά προσδιορίζουν αμφιμονοσήμαντα τα ονόματα και μπορούν να χρησιμοποιηθούν εναλλάξ για την αναφορά σε κάποιο στοιχείο, ωστόσο τα πρώτα είναι πιο «συμπαγή», σταθερού και μικρού μεγέθους και πιο κοντά στη λογική του υπολογιστή (λόγω και δεκαεξαδικής μορφής). Όλες οι εφαρμογές DICOM (συμπεριλαμβανομένων και των αρχείων ιατρικής εικόνας) αναφέρονται στα στοιχεία αυτά κάνοντας χρήση των εν λόγω αναγνωριστικών. Η στήλη VR καθορίζει τη μορφή του κάθε στοιχείου, όπως παρουσιάστηκε και ανωτέρω. Για παράδειγμα, το στοιχείο «Patient's Birth Date» με tag (0010, 0030), πρέπει να έχει μορφή DA, δηλαδή μια συμβολοσειρά οχτώ χαρακτήρων της μορφής YYYYMMDD (όπου οι χαρακτήρες «YYYY» αντιστοιχούν στο έτος, οι «MM» στο μήνα και, τέλος, οι «DD» στην ημέρα). Η πολλαπλότητα τιμής στοιχείου, γνωστής και ως VM (Value Multiplicity), καθορίζει το εάν το σχετικό στοιχείο δύναται να περιλαμβάνει μία μόνο τιμή ή πολλαπλές. Για παράδειγμα, το στοιχείο «Other Patient's Names» με tag (0010,1001) μπορεί προφανώς να περιλαμβάνει περισσότερα του ενός ονόματα, για αυτό και η πολλαπλότητα σημειώνεται ως «1 - n» (όπου n οποιοσδήποτε ακέραιος αριθμός). Τέλος, η στήλη «RET» του λεξικού επισημαίνει τα αποσυρθέντα στοιχεία, εκείνα δηλαδή που περιλαμβάνονταν σε προηγούμενες εκδόσεις του προτύπου και θα πάψουν να υποστηρίζονται από μελλοντικές επανεκδόσεις του. Τα στοιχεία αυτά δεν μπορούν να οριστούν ξανά και ο ρόλος τους έχει ανατεθεί σε νέα, καλύτερα σχεδιασμένα στοιχεία.

(Group, Element) tag	Attribute (data element) name	VR	VM	Retired status
(0008,0001)	Length to End			RET
(0008,0005)	Specific Character Set	CS	1-n	
...				
(0010,0010)	Patient Name	PN	1	
(0010,0020)	Patient ID	LO	1	
(0010,0021)	Issuer of Patient ID	LO	1	
(0010,0030)	Patient's Birth Date	DA	1	
(0010,0032)	Patient's Birth Time	TM	1	
(0010,0040)	Patient's Sex	CS	1	
...				
(0010,1000)	Other Patient IDs	LO	1-n	
(0010,1001)	Other Patient Names	PN	1-n	
...				
(FFFE,E00D)	Item Delimitation Item		1	
(FFFE,E0DD)	Sequence Delimitation Item		1	

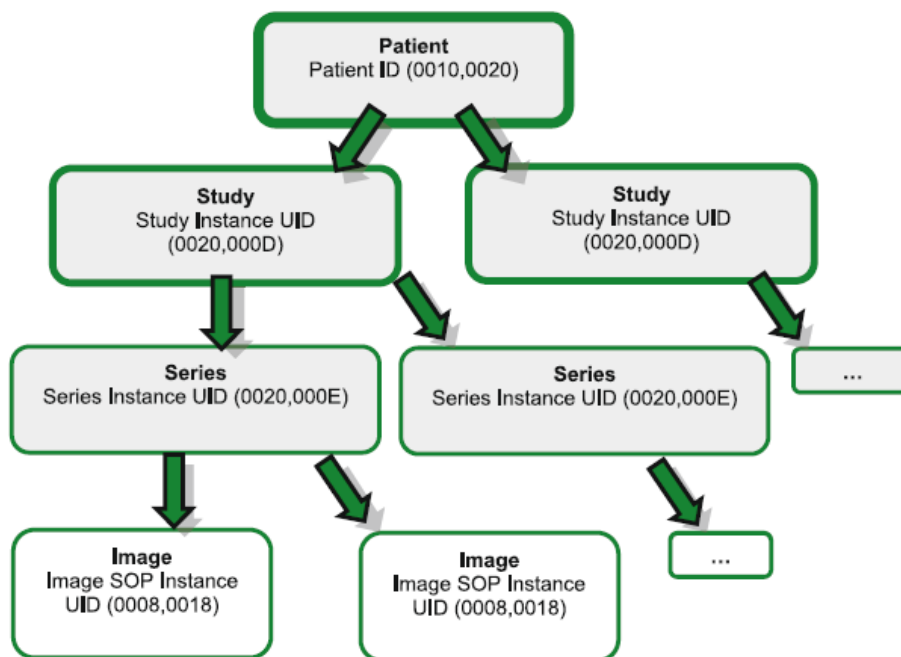
Πίνακας 1. Μερικές γραμμές από το λεξικό DICOM

Το λεξικό DICOM αποτελεί στην ουσία το αποτέλεσμα της προσπάθειας του προτύπου να αποδομήσει όλη την περίπλοκη πληροφορία του πραγματικού κόσμου στα μικρότερα δυνατά «ατομικά» κομμάτια (data elements), κωδικοποιημένα με έναν από τους 27 διαφορετικούς τύπους VR. Ένα, λοιπόν, αντικείμενο DICOM (DICOM object) δεν είναι τίποτα άλλο από μια συλλογή τέτοιων στοιχείων – δεν υπάρχει ξεχωριστή DICOM επικεφαλίδα (“DICOM header”) και DICOM εικόνα (“DICOM image”), όπως πολλοί θέλουν να πιστεύουν. Για παράδειγμα, ας θεωρηθεί μια ψηφιακή ιατρική εικόνα, η οποία περιλαμβάνει χαρακτηριστικά όπως πλάτος, ύψος, χρωματική παλέτα (color palette), ημερομηνία δημιουργίας, τιμές εικονοστοιχείων κτλ. Όλα αυτά τα χαρακτηριστικά μπορούν να βρεθούν μέσα στο λεξικό και θα «μεταφραστούν» σε DICOM data elements, κάθε ένα με το δικό του tag και τιμή. Η ακολουθία αυτών των μεταφρασμένων στοιχείων που περιγράφουν πλήρως την εικόνα στην ολότητά της αποτελεί το αντικείμενο DICOM της εικόνας.

2.1.3.4 Η ιεράρχηση πληροφοριών κατά DICOM

Αναφέρθηκε ήδη η παρουσία DICOM λεξικού δεδομένων το οποίο περιέχει πάνω από 2000 λήμματα/γνωρίσματα και διαδραματίζει κομβικό ρόλο στην αντιστοίχιση δεδομένων πραγματικού κόσμου με το πρότυπο. Εντούτοις, ο αριθμός αυτός είναι σημαντικά μεγάλος για να αφηθεί στην περιγραφή οντοτήτων με τυχαίο και αδόμητο τρόπο. Για το σκοπό αυτό, το DICOM ακολουθεί την ιεραρχία *Ασθενής-Μελέτη-Σειρά-Εικόνα (Patient-Study-Series-Image hierarchy*, βλ. *Εικόνα 3*):

- Ένας ασθενής μπορεί να έχει πραγματοποιήσει πολλαπλές μελέτες.
- Κάθε μελέτη μπορεί να περιέχει πολλαπλές σειρές εικόνων.
- Κάθε σειρά έχει μία ή περισσότερες εικόνες.



Εικόνα 3. Τα 4 επίπεδα της ιεραρχίας πληροφοριών στο DICOM. Κάθε επίπεδο προσδιορίζεται αμφιμονοσήμαντα από ένα στοιχείο-κλειδί

Αυτή η ιεράρχηση αντανακλά με φυσικό τρόπο τι πραγματικά συμβαίνει στον πραγματικό κόσμο όταν ο ασθενής χρειάζεται να πραγματοποιήσει ιατρικές εξετάσεις. Για παράδειγμα, επισκέπτεται ένα νοσοκομείο ή μια ακτινολογική κλινική όπου μπορεί να υποβληθεί σε ένα σύνολο μελετών/εξετάσεων (π.χ. μαγνητική και αξονική τομογραφία, υπερηχογράφημα, κτλ.), στις οποίες αργότερα δύνανται να προστεθούν και άλλες, αν κριθεί σκόπιμο. Κάθε μελέτη μπορεί να έχει πολλαπλές σειρές εικόνων (π.χ. με και χωρίς σκιαγραφικό) κάθε μία εκ των οποίων περιέχει μία (σε περίπτωση π.χ. υπερηχογραφήματος) ή περισσότερες (σε περίπτωση αξονικής ή μαγνητικής) ιατρικές εικόνες. Τώρα, αν χρειαστεί να ανακτηθούν ή να ταξινομηθούν εικόνες του συγκεκριμένου ασθενούς, αρκεί να χρησιμοποιηθούν χαρακτηριστικά γνωρίσματα του ασθενούς αυτού, καθώς επίσης και των μελετών, σειρών και εικόνων που τον αφορούν.

Για να υλοποιήσει αυτήν την ιεραρχία, το DICOM αποδίδει ένα γνώρισμα-κλειδί σε κάθε επίπεδο της. Για το επίπεδο του ασθενούς, αυτό είναι το Patient ID (όλοι οι ασθενείς πρέπει να έχουν μοναδικά αναγνωριστικά – IDs). Παρόμοια, και τα επόμενα επίπεδα διαθέτουν μοναδικά Study Instance UID, Series Instance UID και SOP Instance UID, αντίστοιχα. Όπως είναι λογικό, τα πεδία των γνωρισμάτων αυτών είναι υποχρεωτικό να υπάρχουν σε κάθε έγκυρο αρχείο DICOM.

Όλες οι εντολές DICOM και τα περισσότερα DICOM γνωρίσματα συνδέονται πάντα με αυτή την ιεράρχηση των τεσσάρων επιπέδων. Για το λόγο αυτό, τα τέσσερα γνωρίσματα-κλειδιά διαδραματίζουν έναν καίριο ρόλο: όπως ακριβώς τα ονόματά τους υπονοούν (UID – Unique Identifiers), αναγνωρίζουν με μοναδικό τρόπο τα δεδομένα τους. Αν, λοιπόν, δύο ιατρικές μελέτες έχουν την ίδια τιμή του πεδίου Study Instance UID, τότε αναμένεται να είναι πανομοιότυπες. Αυτό συμπεριλαμβάνει να έχουν επίσης όμοια πεδία Series Instance UID και SOP Instance UID στα αντίστοιχα ιεραρχικά επίπεδα. Παρομοίως, αν δύο ασθενείς έχουν την ίδια τιμή στο Patient ID, αναμένεται να είναι το ίδιο άτομο.

2.1.3.5 Ιατρικές εικόνες στο DICOM

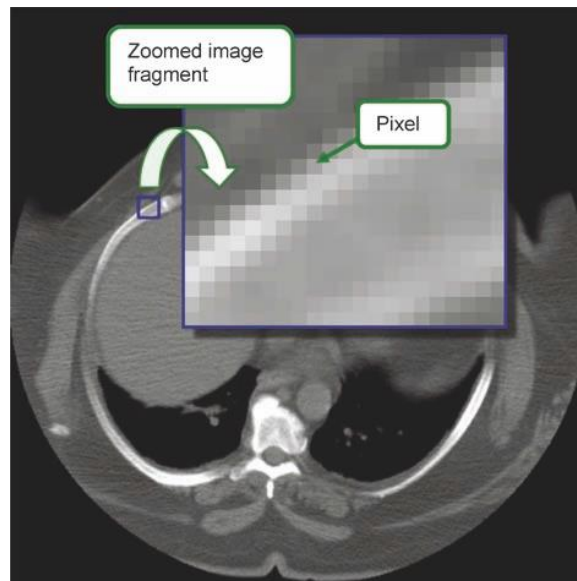
Το πρότυπο DICOM δημιουργήθηκε πρωταρχικά για την ψηφιακή αποθήκευση ιατρικών εικόνων. Οι ψηφιακές εικόνες, γενικά, κατέχουν μερικές γνωστές ιδιότητες, όπως διαστάσεις (πλάτος και ύψος) εκφρασμένες σε εικονοστοιχεία (pixels), βάθος χρώματος σε bits ανά εικονοστοιχείο, όλες εκ των οποίων βρίσκονται στο λεξικό του προτύπου κωδικοποιημένες με κάποιο VR. Το πιο ενδιαφέρον ωστόσο γνώρισμα εικόνας (image attribute) είναι η ίδια η εικόνα – η ακολουθία εικονοστοιχείων δηλαδή που την αποτελούν – αποθηκευμένη στο λεγόμενο «Pixel Data» attribute.

Το DICOM υποστηρίζει ένα μεγάλο εύρος από μορφές εικόνας. Ομαδοποιούνται σε δύο κατηγορίες:

- Ορισμένες κατά DICOM (DICOM-specific), δηλαδή μορφές που χρησιμοποιούνται αποκλειστικά από το πρότυπο. Τυπικά αποτελούν τις πιο παλιές, εισηγμένες από την αρχή της εποχής των υπολογιστών, προτού εξελιχθούν καλύτερες και πιο αποδοτικές μορφές. Ομοιάζουν σε υλοποιήσεις BMPs [12] με διάφορους τρόπους «συσκευασίας» της πληροφορίας των εικονοστοιχείων.
- Ανεξάρτητες προτυποποιημένες μορφές αποδεκτές από το DICOM. Σε αυτές περιλαμβάνονται διαδοσμένες μορφές όπως JPEG, RLE (Run-Length Encoding), ZIP καθώς και τα λιγότερα γνωστά JPEG2000 και JPEG-LS. Όλα αυτά τα πρότυπα σχετίζονται με διάφορες τεχνικές συμπίεσης, τόσο αναστρέψιμες όσο και μη, γεγονός που τις καθιστά ιδιαίτερα χρήσιμες στην ιατρική απεικόνιση (η μείωση του όγκου των δεδομένων της εικόνας είναι σημαντική). Αυτή η αρθρωτή προσέγγιση, κατά την οποία το κύριο πρότυπο (DICOM) προβλέπει τη χρήση άλλων προτύπων (π.χ. JPEG) για συγκεκριμένες εργασίες (όπως η κωδικοποίηση εικόνας), είναι ιδιαίτερα βολική, συνεπής και έχει πρακτική αξία.

2.1.3.6 DICOM BMPs

Μια ψηφιακή εικόνα, ως γνωστόν, μπορεί να μοντελοποιηθεί ως ένας ορθογώνιος πίνακας εικονοστοιχείων – μικροσκοπικών κουκκίδων διαφορετικού χρώματος – που σχηματίζουν την πραγματική εικόνα. Για παράδειγμα, μία συνήθης εικόνα αξονικής τομογραφίας έχει πλάτος και ύψος 512 εικονοστοιχεία, που σημαίνει ότι περιέχει συνολικά $512 \times 512 = 262,144$ εικονοστοιχεία. Αν οι τιμές των εικονοστοιχείων αυτών γραφτούν γραμμή-γραμμή ξεκινώντας από την πάνω αριστερή γωνία της εικόνας σε έναν πίνακα, μπορούν ύστερα να αποθηκευτούν σε ένα αρχείο. Διαισθητικά, αυτό το αρχείο είναι η ακατέργαστη BMP εικόνα (*Εικόνα 4*).



Εικόνα 4. Μεγεθύνοντας τα εικονοστοιχεία ιατρικής εικόνας

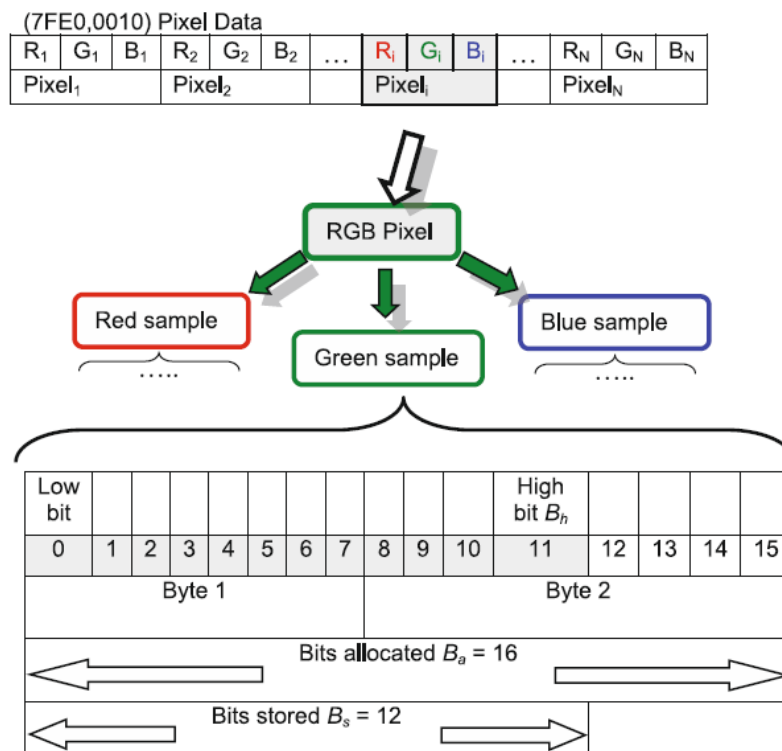
Απαραίτητα χαρακτηριστικά για την επιτυχή ανάκτηση της πληροφορίας της εικόνας είναι τα κάτωθι:

- Σειρές και στήλες (Rows and Columns): στην ουσία είναι η DICOM ονομασία του πλάτους και ύψους της εικόνας. Το γινόμενο τους (γνωστό και ως χωρική ανάλυση εικόνας – spatial image resolution) ισούται με το συνολικό αριθμό των εικονοστοιχείων που την αποτελούν.
- Δείγματα ανά εικονοστοιχείο (Samples per Pixel): κάθε εικονοστοιχείο μπορεί να αποτελεί μια μίξη διαφόρων τιμών δειγμάτων. Το πιο παραστατικό παράδειγμα είναι ένα έγχρωμο εικονοστοιχείο, το οποίο αποτελείται από τρία ανεξάρτητα χρωματικά δείγματα: κόκκινο, πράσινο και μπλε (γνωστά ως RGB χρωματικός χώρος). Ενώ η «δύναμη» του κάθε δείγματος συνεισφέρει στη φωτεινότητα του εκάστοτε εικονοστοιχείου, η μίξη τους δημιουργεί το χρώμα. Έτσι, αναμειγνύοντας ίση ποσότητα κόκκινου και πράσινου παράγει απόχρωση του κίτρινου τη στιγμή που η ανάμειξη ίσης ποσότητας και των τριών χρωμάτων παράγει απόχρωση του γκρι. Οι ασπρόμαυρες εικόνες, ωστόσο, συνήθως αποθηκεύονται με ένα μονοχρωματικό δείγμα ανά εικονοστοιχείο, αντιστοιχώντας στη φωτεινότητα της απόχρωσης του γκρι που το χαρακτηρίζει. Όταν λοιπόν για μια τέτοια DICOM εικόνα χρησιμοποιούνται δείγματα μεγέθους δύο bytes, παρέχεται η δυνατότητα για απεικόνιση $2^{2 \times 8} = 65,536$ διαφορετικών αποχρώσεων του γκρι. Σε κάθε περίπτωση, η επιλογή του τρόπου δειγματοληψίας παραμένει σταθερή για κάθε εικονοστοιχείο μιας συγκεκριμένης εικόνας – εξαρτάται μόνο από την τεχνική που χρησιμοποιήθηκε κατά την απόκτησή της.
- Αριθμός αποθηκευμένων bits δείγματος εικονοστοιχείου, B_s : “Bits Stored” κατά το DICOM. Σε μια ασπρόμαυρη εικόνα, τα εικονοστοιχεία της οποίας χρησιμοποιούν οκτώ bits, ισχύει $B_s = 8$ και ο αριθμός των διαφορετικών γκρι αποχρώσεων της ισούται με $2^{B_s} = 2^8 = 256$. Αν αυξηθεί ο αριθμός των αποθηκευμένων bits σε $B_s = 10$, τότε θα υποστηρίξει $2^{10} = 1024$ αποχρώσεις του γκρι, κ.ό.κ. Όπως γίνεται αντιληπτό, ο αριθμός των αποθηκευμένων bits είναι αυτός που ορίζει το χρωματικό βάθος της εικόνας (δηλαδή τόσο του αριθμού των χρωμάτων σε μια έγχρωμη εικόνα, όσο και του αριθμού των διαβαθμίσεων του γκρι σε μια ασπρόμαυρη). Υπό μια σκοπιά, το συγκεκριμένο γνώρισμα είναι παρόμοιο με τη χωρική ανάλυση, αλλά στο πεδίο της φωτεινότητας. Είναι δε ξεχωριστής σημασίας για την ιατρικό χώρο όπου απαιτείται ο εντοπισμός ακόμα και

απειροελάχιστων διαφοροποιήσεων στο χρώμα και τη φωτεινότητα των εικονοστοιχείων μιας εικόνας.

- Αριθμός δεσμευμένων bits ανά δείγμα εικονοστοιχείου, B_s : ουσιαστικά είναι η τιμή του B_s στρογγυλεμένη προς τα πάνω στην κοντινότερη δύναμη του 2 (έτσι ώστε η τιμή του δείγματος να «χωρά» σε ολόκληρα bytes). Ορίζει την ποσότητα υπολογιστικής μνήμης που απαιτείται για την αποθήκευση ενός χρωματικού δείγματος. Προφανώς, $B_s \leq B_a$.
- Υψηλό bit, B_h : το γνώρισμα που ορίζει το πώς τα αποθηκευμένα bits είναι ευθυγραμμισμένα μέσα στα bits που έχουν δεσμευτεί. Η τιμή του αντιστοιχεί στον αύξων αριθμό του τελευταίου bit που χρησιμοποιείται (το πρώτο bit θεωρείται πως είναι το bit 0).

Η *Εικόνα 5* συνοψίζει τη δομή ενός έγχρωμου DICOM εικονοστοιχείου. Το τελευταίο αποτελείται από τρία δείγματα (κόκκινο, πράσινο και μπλε) και κάθε δείγμα στο συγκεκριμένο παράδειγμα έχει $B_s = 12$ bits. Επειδή όλα τα δεδομένα αποθηκεύονται από τους υπολογιστές σε bytes (όπου 1 byte = 8 bits), τα 12 bits στρογγυλοποιούνται σε $B_a = 16$, με $B_h = 11^3$. Η ίδια 16-bits αποθήκευση χρησιμοποιείται και για τα υπόλοιπα δείγματα (κόκκινο και μπλε στη συγκεκριμένη περίπτωση) και ολόκληρη η εικόνα «γράφεται» ως μια ακολουθία από τα δείγματα των εικονοστοιχείων της. Με τον ίδιο ακριβώς σειριακό τρόπο γίνεται η αποθήκευση και ασπρόμαυρων εικόνων, μόνο που αντί για τρία δείγματα ανά εικονοστοιχείο, στην περίπτωση αυτή συναντάται ένα.



Εικόνα 5. Αποθήκευση δειγμάτων ανά εικονοστοιχείο έγχρωμης (RGB) εικόνας

³ Αυτή η λογική όμως επιφέρει σπατάλη αποθηκευτικού χώρου κατά ποσοστό $4/16 = 25\%$ στο συγκεκριμένο παράδειγμα. Για άλλου τύπου εικόνας (π.χ. με $B_s = 7$), το ποσοστό αυτό μπορεί να ανέλθει ακόμα και στο $7/16 = 43,75\%$! Αυτός είναι ένας από τους λόγους (μαζί και με κάποιους άλλους που θα εξεταστούν στην επόμενη υποενότητα) που εισήχθη η υποστήριξη για ενθυλάκωση συμπιεσμένης εικόνας από το DICOM.

Το DICOM χειρίζεται την πληροφορία της εικόνας όπως οποιαδήποτε άλλη ενθυλακωμένη πληροφορία και όλα τα σημαντικά γνωρίσματα της κωδικοποιούνται σε κάποιας μορφής VR. Στη γλώσσα του DICOM, η ύπαρξή τους είναι υποχρεωτική. Ο Πίνακας 2 παρέχει μια συγκεντρωτική (αλλά όχι εξαντλητική) απογραφή μερικών σημαντικών γνωρισμάτων που περιλαμβάνονται στο λεξικό του DICOM (συμπεριλαμβανομένων των ήδη παρουσιασμένων). Κάποιος με υπόβαθρο στην ψηφιακή απεικόνιση δεν μπορεί παρά να εκτιμήσει την πληρότητα του προτύπου. Για παράδειγμα, όπως υποδηλώνει το στοιχείο με κωδικό (0028,0008), κάποιος μπορεί να αποθηκεύσει μια ακολουθία από καρέ (frames) – στην ουσία βίντεο ψηφιακής μορφής – μέσα σε ένα μόνο αρχείο εικόνας. Επίσης, μέσω του πεδίου (0028,0030) προσδιορίζονται οι φυσικές διαστάσεις των εικονοστοιχείων, κάτι που επιτρέπει την πραγματοποίηση μετρήσεων σε φυσικά μεγέθη (μήκος σε mm, επιφάνεια σε mm²) πάνω στην ιατρική εικόνα. Γνωρίζοντας τώρα τις διαστάσεις αυτές σε συνδυασμό με την απόσταση μεταξύ των εικόνων που αποτελούν μια σειρά (βλ. 102.1.3.4) μέσω του πεδίου (0018,0088), είναι δυνατή η επίτευξη πολύπλοκων τρισδιάστατων ανακατασκευών, αφού διατηρούνται τα ακριβή μεγέθη των αντικειμένων.

Tag	Name	VR	VM
(0028,0002)	Samples per Pixel	US	1
(0028,0004)	Photometric Interpretation	CS	1
(0028,0008)	Number of Frames	IS	1
(0028,0010)	Rows	US	1
(0028,0011)	Columns	US	1
(0028,0030)	Pixel Spacing	DS	2
(0028,0100)	Bits Allocated B_a	US	1
(0028,0101)	Bits Stored B_s	US	1
(0028,0102)	High Bit B_h	US	1
(0028,0103)	Pixel Representation	US	1
(7FE0,0010)	Pixel Data	OW/OB	1

Πίνακας 2. Μερικά από τα πιο συνήθη γνωρίσματα εικόνας που περιέχονται στο λεξικό του DICOM

2.1.3.7 Συμπιεσμένες DICOM εικόνες

Θεωρώντας πάλι το προηγούμενο παράδειγμα της εικόνας αξονικής τομογραφίας, έχει αξία να υπολογιστεί η ποσότητα μνήμης που απαιτείται για την αποθήκευσή της. Καταρχήν, αποτελείται από 262,144 εικονοστοιχεία (όπως έχει υπολογιστεί ήδη). Πρόκειται για ασπρόμαυρη εικόνα, οπότε κάθε εικονοστοιχείο έχει ένα χρωματικό δείγμα. Αν τώρα υποθεθεί (και στην πράξη όντως ισχύει) πως το δείγμα αυτό περιγράφεται με οποιοδήποτε πλήθος bits μεταξύ 8 και 16, τότε συμπεραίνεται πως τα απαιτούμενα bytes ανά δείγμα και, κατά συνέπεια, εικονοστοιχείο είναι 2. Οπότε, για την αποθήκευση ολόκληρης της εικόνας απαιτούνται συνολικά $512 \times 512 \times 2 = 524,288$ bytes \approx 0,5 MB. Όμως, όπως δείχθηκε, οι εικόνες αξονικής τομογραφίας παράγονται στα πλαίσια κάποιας μελέτης και κάποιας σειράς, κάθε μία εκ των οποίων μπορεί να περιέχει μερικές δεκάδες έως και εκατοντάδες ιατρικές εικόνες. Αν δε συνυπολογίσει κανείς τη

συνεχή βελτίωση των ακτινολογικών και ραδιολογικών μονάδων με αποτέλεσμα την αύξηση της ανάλυσης και του βάθους των παραγόμενων εικόνων, η αποθήκευση και η αποστολή τους – ειδικά παλαιότερα που το υλισμικό ήταν πιο ακριβό και οι ταχύτητες των δικτύων περιοριστικές, αποτελούσε πρόβλημα για οποιοδήποτε οργανισμό επιχειρούσε να επενδύσει σε ένα λειτουργικό και κλιμακώσιμο τηλεακτινολογικό (teleradiological) εξοπλισμό (βλ. Πίνακας 3).

Για να αντιμετωπίσει τους σκοπέλους αυτούς, το DICOM υιοθέτησε την υποστήριξη για συμπίεση εικόνας σχεδόν από τις απαρχές του. Η συμπίεση αυτή εφαρμόζεται στοχευμένα στα δεδομένα εικονοστοιχείων του πεδίου "Pixel Data" και με έξυπνο τρόπο τα αναδιατάσσει σε μια πολύ πιο σύντομη μορφή. Κατά συνέπεια, επιτυγχάνεται και μια παράλληλη σημαντική μείωση του μεγέθους ολόκληρου του αρχείου DICOM – αν αναλογιστεί κανείς πως το μεγαλύτερο κομμάτι του καταλαμβάνεται από τα δεδομένα της εικόνας, που με τη σειρά του εξοικονομεί αποθηκευτικό χώρο και ελαττώνει το χρόνο διαμοιρασμού του πάνω από το δίκτυο. Το ίδιο το πρότυπο, όπως προαναφέρθηκε, δεν υλοποιεί κάποιο δικό του μηχανισμό συμπίεσης, αλλά χρησιμοποιεί κάποιες δοκιμασμένες και δημοφιλείς λύσεις, όπως RLE, JPEG, JPEG2000, JPEG-LS και ZIP. Όλοι οι αλγόριθμοι στους οποίους είναι βασισμένες, έχουν αναπτυχθεί ξεχωριστά και έχουν καταλήξει να αποτελούν ξεχωριστά ISO (International Organization for Standardization) πρότυπα. Το DICOM επέλεξε απλώς να τα υιοθετήσει.

Image modality	Typical image matrix (height, width, bytes per pixel)	Image size, kilobytes (KB)	Typical number of images in a study ^a	Typical study size, megabytes (MB)
NM	128 × 128 × 1	16	100	1.5
MR	256 × 256 × 2	128	200	25
CT	512 × 512 × 2	512	500	250
Ultrasound	600 × 800 × 3	1400	500	680
CR	2140 × 1760 × 2	7356	4	30
Color 3D reconstructions ^b	1024 × 1024 × 3	3000	20	60
Digital mammography	Up to 6400 × 4800 × 2	60,000	4	240

Πίνακας 3. Τυπικά μεγέθη ιατρικών εικόνων και μελετών

Η εφαρμογή της συμπίεσης εικόνας μπορεί να διαδραματίσει καταλυτικό ρόλο στην εμφάνισή της αλλά και στη συνολική επίδοση των PACS. Κάθε αλγόριθμος συμπίεσης αναζητά να βρει και να απομακρύνει πλεονάζουσα και επαναληπτική πληροφορία στην προσπάθειά του να μειώσει το μέγεθος των δεδομένων. Η αποδοτικότητά του προσεγγίζεται με το λεγόμενο λόγο συμπίεσης:

$$R_{comp} = \frac{\text{αρχικό μέγεθος δεδομένων}}{\text{συμπιεσμένο μέγεθος δεδομένων}}$$

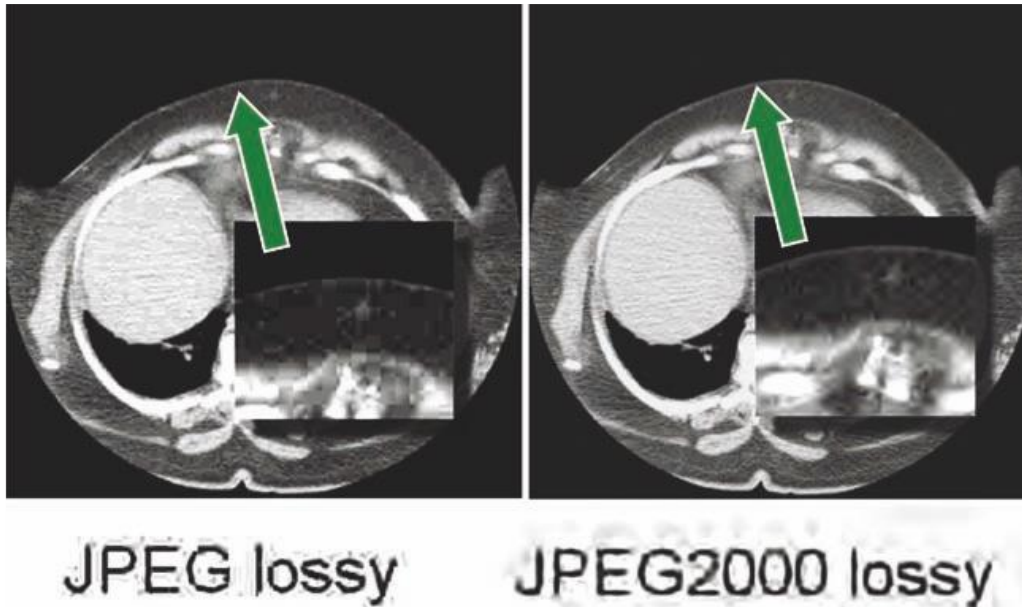
Όσο μεγαλύτερος ο λόγος, τόσο πιο αποτελεσματική συμπίεση επιτυγχάνεται. Κάθε αλγόριθμος προσφέρει τη δική του στρατηγική μεγιστοποίησης του R_{comp} , αλλά

εννοιολογικά και πρακτικά, όλες οι διαφορετικές τεχνικές που προσφέρουν ομαδοποιούνται σε μη-απωλεστικές (lossless) και σε απωλεστικές (lossy).

Η μη-απωλεστική συμπίεση δεν τροποποιεί την εικόνα. Όσες φορές και αν συμπίεστεί και στη συνέχεια αποσυμπιεστεί, πάντοτε τα δεδομένα της θα είναι ίδια – δεν παρατηρείται απώλεια πληροφορίας. Αυτό επιτυγχάνεται με έξυπνη ανασύνταξη και μετονομασία των bytes των εικονοστοιχείων. Σε γενικές γραμμές, ένας τυπικός αλγόριθμος μη-απωλεστικής συμπίεσης θα προσπαθήσει να εντοπίσει την επαναληπτικότητα των συχνότερων τιμών και να τις αντικαταστήσει με συντομότερα σύμβολα (π.χ. αν πολλαπλά εικονοστοιχεία μιας ασπρόμαυρης εικόνας έχουν την τιμή 1000, αυτή μπορεί να αντικατασταθεί με το σύμβολο "α"). Ωστόσο, οι μετασχηματισμοί αυτοί μπορούν να χρησιμοποιηθούν μόνο μέχρι ενός σημείου και για μια μέση ιατρική εικόνα, ο λόγος R_{comp} βρίσκεται κάπου μεταξύ δύο και τέσσερα – νούμερο αποδεκτό αλλά όχι εντυπωσιακό αν συγκριθεί με το αντίστοιχο μιας απωλεστικής προσέγγισης.

Η δεύτερη εναλλακτική πετυχαίνει σημαντικά μεγαλύτερο βαθμό συμπίεσης, αλλά με το κόστος – όπως φανερώνει και το όνομα – της απώλειας αρχικής πληροφορίας. Η θυσία αυτή γίνεται έτσι ώστε να εισαχθεί τεχνηέντως πλεονασμός δεδομένων τον οποίο και θα εκμεταλλευτεί περαιτέρω ο αλγόριθμος μη-απωλεστικής συμπίεσης. Με λίγα λόγια, αν η μη-απωλεστική συμπίεση αξιοποιεί τα εικονοστοιχεία με ίσες τιμές, η απωλεστική επεκτείνει αυτή την αξιοποίηση και σε εικονοστοιχεία σχεδόν ίσων τιμών με ένα δύσκολο αντιληπτό περιθώριο λάθους. Με τον τρόπο αυτό, οι σημερινοί απωλεστικοί αλγόριθμοι μπορεί να επιτύχουν λόγο συμπίεσης ίσο ακόμα και με 100, αν και σε μια μέση περίπτωση περιορίζονται μεταξύ του 10 και του 20. Η τιμή του λόγου στην απωλεστική συμπίεση εξαρτάται από την τιμή που ορίζεται ως επιτρεπτό σφάλμα. Προφανώς, και το σφάλμα αυτό δεν μπορεί να αυξάνεται επ' αόριστον προκειμένου να πετυχαίνεται συνεχώς καλύτερος βαθμός συμπίεσης. Από ένα σημείο και μετά, στη συμπιεσμένη εικόνα αρχίζουν και εισάγονται ορατές αλλοιώσεις (artifacts), όπως απεικονίζεται και στην *Εικόνα 6*. Η ισορροπία μεταξύ υψηλού R_{comp} και ορατής παραμόρφωσης της εικόνας έχει γίνει πλέον μια «μορφή τέχνης» από μόνη της. Σημαντικοί παράγοντες για τη διαμόρφωσή της είναι οι εξής:

- Η απωλεστική συμπίεση μπορεί να οδηγήσει σε νομικές διαφορές. Αν πρόκειται να χρησιμοποιηθούν εικόνες με τέτοιου είδους συμπίεση, το DICOM και ο FDA (Food and Drug Administration) απαιτούν την υποχρεωτική επισήμανσή τους.
- Ζητήματα διάγνωσης με τη βοήθεια υπολογιστή (Computer-Aided Diagnosis – CAD). Καθώς οι τεχνικές CAD γίνονται δημοφιλέστερες, οι υπολογιστές και το λογισμικό τους εμπλέκονται όλο και περισσότερο στην ανάλυση ιατρικής εικόνας. Σφάλματα αόρατα στο ανθρώπινο μάτι μπορεί να είναι ορατά για το λογισμικό, υπονομεύοντας την αποτελεσματικότητά του.



Εικόνα 6. Αποτελέσματα υπερβολικής απωλεστικής συμπίεσης σε εικόνες και γραμματοσειρές. Στα αριστερά, η συμπίεση JPEG παράγει τα λεγόμενα blocking artifacts. Στα δεξιά, η συμπίεση JPEG2000 παράγει έντονη θολότητα.

Στην *Εικόνα 6* αντιπαρατίθενται τα δύο πιο δημοφιλή πρότυπα συμπίεσης στο χώρο της ιατρικής απεικόνισης, το JPEG και το JPEG2000. Το πρώτο χρησιμοποιεί αλγορίθμους βασισμένους στο μετασχηματισμό DCT (Discrete Cosine Transform) ενώ το δεύτερο και μεταγενέστερο αλγορίθμους βασισμένους σε wavelets. Ο DCT εκφράζει μια πεπερασμένη ακολουθία σημείων δεδομένων (data points) μέσω ενός αθροίσματος συνημιτονοειδών συναρτήσεων διαφορετικών συχνοτήτων. Στην ουσία, είναι παρόμοιος με το DFT (Discrete Fourier Transform), αλλά χρησιμοποιεί μόνο πραγματικούς αριθμούς. Το wavelet είναι μια μαθηματική συνάρτηση που χρησιμοποιείται στη διαίρεση μιας δοθείσας άλλης συνάρτησης ή σήματος σε διαφορετικά κλιμακούμενα συστατικά (scale components). Συνήθως κάποιος μπορεί να αναθέσει ένα εύρος συχνοτήτων σε κάθε κλιμακούμενο συστατικό, το οποίο κατόπιν μπορεί να μελετηθεί με μια ανάλυση που ταιριάζει στην κλίμακά του. Ένας wavelet μετασχηματισμός είναι η αναπαράσταση μιας συνάρτησης από wavelets.

Το JPEG κάνει χρήση του 8×8 DCT, ενώ το JPEG2000 του DWT (Discrete Wavelet Transformation). Ο τελευταίος παρέχει όχι μόνο καλύτερη ενεργειακή συμπίκνωση (οπότε και υψηλότερο κέρδος συμπίεσης), αλλά και κλιμακωσιμότητα της ανάλυσης – αφού οι συντελεστές του wavelet μπορούν να διαχωριστούν σε διαφορετικές αναλύσεις, είναι εφικτό να εξαχθεί μιας χαμηλότερης ανάλυσης εικόνα χρησιμοποιώντας μόνο τους αναγκαίους συντελεστές. Το JPEG ακόμα, κατατέμνει την εικόνα σε blocks των 8×8 ή 16×16 στο πεδίο του χώρου και στη συνέχεια εφαρμόζει τους υπόλοιπους μετασχηματισμούς. Εφόσον ακολουθεί αυτή τη λογική της ανεξάρτητης κωδικοποίησης των blocks, από ένα σημείο και μετά γίνονται ορατά τα blocking artifacts της παραπάνω εικόνας. Σε αντίθεση, το JPEG2000 πραγματοποιεί την κατάτμηση στο πεδίο των wavelet και σε συνδυασμό με τον αντίστοιχο μετασχηματισμό, δεν εμφανίζει παρόμοια προβλήματα. Επίσης, ενώ με το JPEG είναι δυνατή η προβολή μιας εικόνας σε συγκεκριμένη μόνο ανάλυση ορισμένη κατά τη κωδικοποίηση, το JPEG2000 επιτρέπει τη μερική αποσυμπίεση των wavelets που περιέχει, προκειμένου ο χρήστης να προβάλλει, αν επιθυμεί την εικόνα σε χαμηλότερη ανάλυση. Τέλος, ένα ακόμα σημαντικό χαρακτηριστικό που

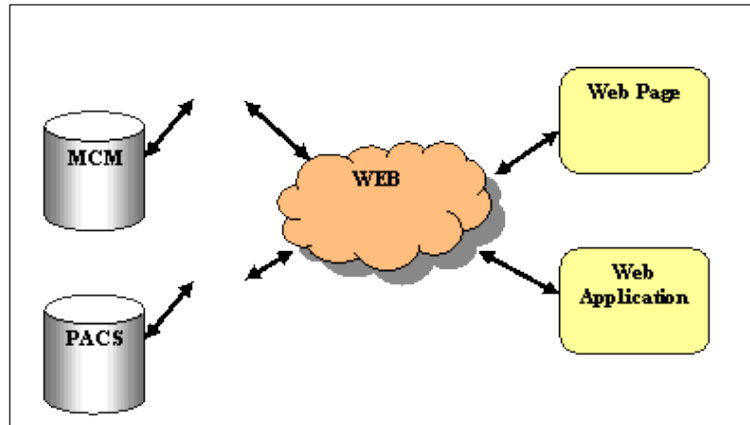
προσδίδει η χρήση των wavelets στο JPEG2000 είναι οι ROI (Region of Interest) δυνατότητες, δηλαδή να μπορεί να προβάλλει κάποιος μια συγκεκριμένη περιοχή της εικόνας σε υψηλότερη ποιότητα από ότι την υπόλοιπη, με συνέπεια τη μείωση της απαιτούμενης μνήμης αλλά και του χρόνου πρόσβασης. Από όλα τα παραπάνω, γίνεται εύκολα κατανοητό το πλεονέκτημα που προσδίδει η χρήση JPEG2000 κωδικοποίησης για τις ιατρικές εικόνες, τόσο κατά την αποθήκευσή τους, όσο και κατά τη μεταφορά τους πάνω από το δίκτυο.

2.1.3.8 Web Access to DICOM Objects (WADO)

Σήμερα, τα περισσότερα DICOM αντικείμενα αρχειοθετούνται σε ειδικές αποθήκες (repositories), όπως PACS και MCM⁴, για συγκεκριμένες χρονικές περιόδους. Όταν ένα ιατρικό πληροφοριακό σύστημα επιθυμεί να συνδεθεί σε μια αποθήκη και να ανακτήσει DICOM πληροφορία, πρέπει να είναι «εξοικειωμένο» με το συγκεκριμένο, συνήθως κλειστό (proprietary) πρωτόκολλο που η αποθήκη χρησιμοποιεί. Το γεγονός αυτό συνεπάγεται την επένδυση σημαντικών προσπάθειών κατά την εξέλιξη εφαρμογών με δυνατότητα απομακρυσμένης σύνδεσης σε τέτοιου είδους αποθήκες δεδομένων. Επιπρόσθετα, οι εν λόγω εφαρμογές γενικά υπολείπονται σε ευελιξία, καθότι δεν μπορούν εύκολα να επεκταθούν για την υποστήριξη περισσότερων αποθηκών.

Για να αντιμετωπιστούν αυτοί οι σκόπελοι και να καταστεί δυνατή η πρόσβαση σε DICOM πληροφορία από οπουδήποτε στον ιστό, δημιουργήθηκε το πρότυπο WADO (για την ακρίβεια αποτελεί μέρος του ιδίου του DICOM, καθώς περιγράφεται στην παράγραφο PS 3.18-2004 των προδιαγραφών του). Το τελευταίο ορίζει μια υπηρεσία ιστού (web-based service) για την ανάκτηση και παρουσίαση αποθηκευμένων αντικειμένων DICOM, όπως εικόνες και αναφορές ιατρικής απεικόνισης (medical imaging reports) και έχει ως σκοπό τη διανομή αποτελεσμάτων και εικόνων σε επαγγελματίες υγείας. Στην ουσία, παρέχει έναν απλό μηχανισμό ώστε η πρόσβαση αυτή να καθίσταται δυνατή από σελίδες HTML ή αρχεία XML διαμέσου HTTP/HTTPS, κάνοντας παράλληλη χρήση των μοναδικών αναγνωριστικών που παρέχει το DICOM. Τα δεδομένα μπορούν να ανακτηθούν είτε σε κάποια μορφή έτοιμη προς παρουσίαση (π.χ. σε μορφή εικόνας JPEG ή PNG) είτε στη φυσική τους. Για παράδειγμα, ένας ιατρός-χρήστης δύναται να εκτελέσει απομακρυσμένα αναζήτηση σε μια αποθήκη που συμμορφώνεται με το WADO για ένα συγκεκριμένο ασθενή (μέσω του πεδίου Patient ID) και κατόπιν να πραγματοποιήσει λήψη μιας ολόκληρης μελέτης (μέσω του Series Instance UID) στην οποία έχει υποβληθεί στο παρελθόν.

⁴ Το MCM (Medical Content Manager) είναι ένα σύστημα της IBM για αποθήκευση ιατρικού περιεχομένου που απαιτεί μακροχρόνια διαχείριση και αρχειοθέτηση [60].



Εικόνα 7. Το WADO ως υπηρεσία

2.2 Ιατρική απεικόνιση

Η ιατρική απεικόνιση είναι η τεχνική και η διαδικασία δημιουργίας οπτικής αναπαράστασης του εσωτερικού του ανθρώπινου σώματος για κλινική ανάλυση και ιατρική παρέμβαση. Η ιατρική απεικόνιση αναζητά να αποκαλύψει εσωτερικές δομές κάτω από το δέρμα και τα κόκκαλα, με σκοπό να συμβάλει στη διάγνωση και τη θεραπεία παθήσεων. Ακόμα, εγκαθιδρύει μια βάση δεδομένων με στοιχεία ανθρώπινης ανατομίας και φυσιολογίας επιτρέποντας έτσι την πιο εύκολη αναγνώριση ανωμαλιών. Παρότι η απεικόνιση οργάνων και ιστών που έχουν αφαιρεθεί μπορεί να γίνει για ιατρικούς λόγους, τέτοιες πρακτικές θεωρούνται μέρος του κλάδου της παθολογίας.



Εικόνα 8. Αξονική τομογραφία αποκαλύπτει ρήξη ανευρύσματος της κοιλιακής αορτής

Ως επιστημονικός κλάδος και εξεταζόμενος κάτω από ένα ευρύτερο πρίσμα, η ιατρική απεικόνιση αποτελεί μέρος της βιολογικής απεικόνισης και ενσωματώνει την ακτινολογία (radiology), η οποία χρησιμοποιεί τεχνολογίες απεικόνισης όπως ακτινογραφία (X-ray radiography), μαγνητική τομογραφία (magnetic resonance imaging), υπέρηχος (medical ultrasonography ή ultrasound), ελαστογραφία (elastography), θερμογραφία (thermography), τομογραφία εκπομπής ποζιτρονίων (positron emission tomography) κ.ά.

2.3 Εφαρμογές απεικόνισης ιατρικών εικόνων

Μέχρι και το 2010, έχουν πραγματοποιηθεί πέντε δισεκατομμύρια μελέτες ιατρικής απεικόνισης [13]. Εξαιτίας λοιπόν αυτής της διάδοσης της ιατρικής εικόνας τα τελευταία χρόνια αλλά και της προτυποποίησης αυτής (βλ. 2.1.3), έχει εμφανιστεί μια ευρεία γκάμα desktop και web εφαρμογών που επιτρέπουν την προβολή και επεξεργασία της. Για τους σκοπούς αυτής της διπλωματικής εργασίας αναφέρονται ενδεικτικά μερικές από τις πιο γνωστές:

2.3.1 OsiriX

Η εφαρμογή OsiriX είναι ανοικτό λογισμικό επεξεργασίας εικόνων παραγόμενων από εξοπλισμό ιατρικής απεικόνισης (MRI, CT, PET, PET-CT, SPECT-CT, Ultrasound, κτλ.) για την πλατφόρμα Macintosh. Εξελίσσεται και συντηρείται από την Pixmeo, μια εταιρία με βάση τη Γενεύη. Είναι πλήρως συμβατό με το πρότυπο αρχείων DICOM (επέκτασης .dcm) αλλά και το πρωτόκολλο επικοινωνίας του, οπότε μπορεί να λειτουργήσει παράλληλα και ως σταθμός εργασίας DICOM PACS. Έχει σχεδιαστεί συγκεκριμένα για πλοήγηση και απεικόνιση πολυδιάστατων εικόνων πολλαπλών modalities (2D, 3D, 4D και 5D) χρησιμοποιώντας όλες τις σύγχρονες πρακτικές, όπως πολυεπίπεδη ανασύνθεση (multiplanar reconstruction), απεικόνιση επιφάνειας (surface rendering), απεικόνιση όγκου (volume rendering), κτλ. Τέλος, να σημειωθεί ότι είναι γραμμένο σε Objective-C, η αρχιτεκτονική του υποστηρίζει την επέκταση των δυνατοτήτων του σύμφωνα με τις ανάγκες του χρήστη μέσω plug-ins, ενώ είναι από τους ελάχιστους viewers που είναι εγκεκριμένος από τον αμερικανικό οργανισμό FDA (Food and Drug Administration).

2.3.2 RadiAnt

Το RadiAnt είναι μια standalone εφαρμογή για την πλατφόρμα των Windows η οποία διατίθεται δωρεάν (freeware) αλλά, σε αντίθεση με το OsiriX, ο κώδικάς της δεν είναι ανοικτός. Είναι και αυτή συμβατή με το πρότυπο DICOM κάτι που σημαίνει πως μπορεί να ανοίξει και να προβάλλει οποιοδήποτε αρχείο .dcm, εντούτοις δεν υποστηρίζει αλληλεπίδραση με κάποιο PACS. Οι δυνατότητες που παρέχει στο χρήστη είναι βασικές και δε φτάνουν σε κανένα βαθμό το εύρος του OsiriX (έλλειψη προηγμένων τεχνικών rendering, φίλτρων, image fusion), αλλά η ταχύτητά του είναι αξιοσημείωτη μιας και εκμεταλλεύεται πλήρως τις σύγχρονες πολυύρηνες αρχιτεκτονικές, η διεπαφή της υποδειγματική και ο χρόνος εξοικείωσης μαζί της μικρός.

2.3.3 LEADTOOLS HTML5 Zero-footprint Medical Viewer

Το προϊόν αυτό είναι ένα web application αποτελούμενο από μια ισχυρή συλλογή από βιβλιοθήκες σε JavaScript και RESTful Web Services που παρέχουν τη δυνατότητα για προβολή, διαχείριση και επεξεργασία DICOM εικόνας από PACS. Αυτό σημαίνει πως δεν υποστηρίζει σε αντίθεση με τις προηγούμενες εφαρμογές, την πρόσβαση σε αρχεία DICOM από εξωτερικές συσκευές αποθήκευσης, αλλά το γεγονός ότι είναι γραμμένο

για να εκτελείται σε περιβάλλον browser, το καθιστά προσβάσιμο ανεξαρτήτως πλατφόρμας (platform-independent) χωρίς να θυσιάζει τίποτα από άποψη λειτουργικότητας και δυνατοτήτων. Ακόμα σημειώνεται πως διατίθεται έναντι πληρωμής.

2.4 Βιβλιογραφία

Από ερευνητική άποψη, η σημασία της συνεργατικότητας μεταξύ συμμετεχόντων σε συνεδρίες πολυεπιστημονικών ομάδων αλλά και γενικότερα ιατρικού προσωπικού που βρίσκεται σε απομακρυσμένες περιοχές έχει επιβεβαιωθεί και αναλυθεί διεξοδικά. Προσπάθειες για εκμετάλλευση της σύγχρονης τεχνολογίας υπολογιστών και δικτύων στην υπηρεσία του σκοπού αυτού έχουν ήδη γίνει και θα συνεχίσουν να γίνονται.

Διάφορες ερευνητικές ομάδες έχουν εμπλακεί ενεργά στον τομέα. Το CREW [14] είναι ένα τυπικό παράδειγμα συνεργατικού συστήματος υλοποιημένου στο πανεπιστήμιο του Michigan. Το συγκεκριμένο βασισμένο σε υπολογιστές περιβάλλον είναι σε θέση να δώσει πολλαπλή πρόσβαση σε διαμοιρασμένα αρχεία και παρέχει στην κοινότητά του πληροφορίες σχετιζόμενες με άλλα μέλη της, όπως τις δραστηριότητες, την τοποθεσία, τη διαθεσιμότητα και τα ενδιαφέροντά τους. Επιπρόσθετα, υποστηρίζει συνεργατικά εργαλεία όπως post σημειώσεων και αποστολή e-mail. Όσον αφορά την ιατρική ερευνητική κοινότητα και αυτή έχει να επιδείξει αντίστοιχο αξιοσημείωτο έργο. Το InterMed [15] είναι ένα σύστημα που στοχεύει στην εξέλιξη ενός συμπαγούς πλαισίου εργασίας (framework) για υποστήριξη ιατρικής συνεργατικότητας χρησιμοποιώντας το Internet για τη διασύνδεσή του. Το σχέδιο TeleMed [16] επιτρέπει πραγματικού χρόνου αλλά και εκτός σύνδεσης συνεδρίες. Πολλαπλοί ιατροί σε απομακρυσμένες περιοχές είναι σε θέση να προβάλλουν, να επεξεργαστούν και να σχολιάσουν δεδομένα ασθενών σε πραγματικό χρόνο. Η συγκεκριμένη υλοποίηση εστιάζει στην ασφάλεια και σε εμπιστευτικές υπηρεσίες, στην πρόσβαση σε online γνώση, στο διαμοιρασμό κλινικών δεδομένων, όπως επίσης σε συνεργατικές πολιτικές, διαδικασίες και κατευθυντήριες γραμμές. Το σχέδιο ARTEMIS αποσκοπεί σε «προηγμένες συνεργατικές δραστηριότητες των παρόχων υπηρεσιών υγείας προς προώθηση της παροχής ολοκληρωμένης και πραγματικού χρόνου φροντίδας» [17]. Ένα σύνολο από προγραμματιστές, ιατρούς και ιατρικούς ερευνητές χρησιμοποιούν ένα συνδυασμό από πρωτότυπες και έτοιμες εμπορικές τεχνολογίες για να εξελίξουν ένα ανοικτό συνεργατικό περιβάλλον υγείας. Στο ARTEMIS, δίκτυα κοινοτικής περίθαλψης αποτελούμενα από πρωτοβάθμιους και εξειδικευμένους φορείς συνεργάζονται για να καλύψουν τις ανάγκες της κοινότητας και να βελτιώσουν την ποιότητα των παρεχόμενων υπηρεσιών υγείας. Η επικοινωνιακή υποδομή επιτρέπει σε ιατρούς να συμβουλευονται απομακρυσμένους εξειδικευμένους συναδέλφους τους, ανταλλάσσοντας ακτινογραφίες, υπερήχους, ηχητικές σημειώσεις και άλλες πληροφορίες πολυμέσων. Ένα άλλο ερευνητικό σχέδιο που επικεντρώνεται στον ιατρικό ερευνητικό τομέα είναι το ιατρικό γραφείο εικονικής πραγματικότητας [18], στο οποίο ένα ολοκληρωμένο περιβάλλον ενθαρρύνει τους χρήστες να αναλάβουν ενεργό ρόλο στη διαχείριση της υγείας τους. Η συγκεκριμένη πρόταση παρέχει πρόσβαση σε ψηφιακές ιατρικές βιβλιοθήκες, χρυσούς οδηγούς και περιφερειακούς ιατρικούς πόρους. Το WebOnColl [19] έχει σχεδιαστεί στο πλαίσιο του περιφερειακού δικτύου ιατροφαρμακευτικής περίθαλψης της Κρήτης και χρησιμοποιεί την υπάρχουσα υποδομή του για να παράσχει ολοκληρωμένες υπηρεσίες για εικονικούς χώρους εργασίας, σχολιασμούς, e-mail και online συνεργασία. Οι χώροι εργασίας αυτοί υποστηρί-

ζουν συνεργατικές έννοιες όπως προσωπικές ιστοσελίδες, πίνακες ανακοινώσεων, λίστες συζητήσεων και ιατρικούς φακέλους. Οι σχολιασμοί παρέχουν στους χρήστες ένα φυσικό τρόπο αλληλεπίδρασης με πολυμεσικό περιεχόμενο ενώ η αποστολή e-mail είναι πλέον ένας από τους πιο διαδεδομένους τρόπους επικοινωνίας.

Μια νεότερη αξιοσημείωτη προσέγγιση στο ζήτημα [20], βασιζόμενη σε μοντέρνες πρακτικές, υιοθετεί μια υβριδική αρχιτεκτονική: peer-to-peer ανταλλαγή ηλεκτρονικών φακέλων υγείας (Electronic Health Record - EHR), ιατρικών αρχείων DICOM, ήχου και βίντεο μεταξύ του απομακρυσμένου ιατρικού προσωπικού και client-server επικοινωνία για τη διευθέτηση ζητημάτων ασφαλείας, διαχείρισης χρηστών και άντλησης των ιατρικών δεδομένων από ένα RDBMS. Συνδυάζοντας τα οφέλη των δύο αυτών διαφορετικών στρατηγικών, οι ιατροί διαβουλεύονται ενώ παράλληλα εργάζονται πάνω σε εικόνες και αναφορές σε πραγματικό χρόνο. Εντούτοις, το σύστημα που περιγράφεται λειτουργεί μέσα στα πλαίσια ενός συγκεκριμένου ιδρύματος και οι πληροφορίες είναι αποθηκευμένες στον εξυπηρετητή που βρίσκεται στον ίδιο χώρο. Το δίκτυο VIRGO [21] υλοποιήθηκε για να υποστηρίξει ιατρούς, ασθενείς και ακαδημαϊκά ινστιτούτα στην ανακάλυψη και εξαγωγή πολύπλοκων ιατροφαρμακευτικών πληροφοριών. Υιοθετεί τη σημασιολογική προσέγγιση για αναζήτηση πληροφορίας πάνω από DHTs (Distributed Hash Table) και εστιάζει κυρίως στο διαμοιρασμό δεδομένων μεταξύ ανεξάρτητων ιδρυμάτων. Στο ίδιο μήκος κύματος, οι συγγραφείς του [22] χειρίζονται την ολοκλήρωση των δεδομένων και ζητήματα διαλειτουργικότητας βασισμένα στο οντολογικό μοντέλο για αναζήτηση δεδομένων. Με αυτόν τον τρόπο, ένας peer μπορεί να έχει πρόσβαση σε ένα μεγάλο σύνολο πληροφοριών και μεταδεδομένων. Οι συγγραφείς του [23] παρουσιάζουν μια αρχιτεκτονική για διαχείριση ροών εργασίας στον ιστό βασισμένη επίσης στη peer-to-peer λογική. Βρίσκει εφαρμογή σε ποικίλες περιστάσεις όπου επαγγελματίες από διαφορετικούς τομείς της υγείας μπορούν να συνεργαστούν. Μια ιατρική συμβουλευτική διαδικασία εξετάζεται ως αντικείμενο μελέτης. Σε αυτή την περίπτωση, δραστηριότητες σχετιζόμενες με ιατρικές συσκέψεις κατανέμονται στους ομότιμους συμμετέχοντες. Η στρατηγική αυτή προσφέρει αποδοτικότητα χρόνου, ταχύτερη θεραπεία και μείωση του κινδύνου απώλειας ιατρικών αρχείων. Μια κεντρική οντότητα διαδραματίζει κομβικό ρόλο στην ανάθεση εργασιών στους peers μιας και δεν υπάρχουν μηχανισμοί για κατανομημένη αποθήκευση και ανάκτηση δεδομένων. Στο [24], παρουσιάζεται μια προσέγγιση βασισμένη σε πλέγμα (grid-based) για τηλείατρικούς σκοπούς. Σύμφωνα με αυτή, ένα δίκτυο δεδομένων σχηματίζεται μεταξύ νοσοκομείων, κινητών ιατρικών μονάδων και περιφερειακών κλινικών. Οι οντότητες που συμμετέχουν, σχηματίζουν ομάδες με γεωγραφικά κριτήρια με σκοπό τη μείωση του χρόνου αντίδρασης σε ιατρικά συμβάντα.

Τέλος, στο [25] ερευνητές εισάγουν την έννοια της συνεργατικής πλατφόρμας ενεργού περιεχομένου (active content collaboration platform), η οποία υποστηρίζει αυτοματοποιημένη, οδηγούμενη από συμβάντα (event-driven) συνεργατικότητα, αξιοποιώντας τη δύναμη του cloud για εκτέλεση υπολογιστικά απαιτητικών εργασιών, όπως η ανασύνθεση τρισδιάστατης εικόνας. Ωστόσο, απουσιάζει η αμεσότητα της επικοινωνίας μεταξύ των χρηστών λόγω έλλειψης δυνατοτήτων τηλεδιασκέψεων μέσω βίντεο και δυνατότητας αλληλεπίδρασης με τα ιατρικά δεδομένα σε πραγματικό χρόνο.

Όλη η έρευνα και η εργασία που παρουσιάστηκε ανωτέρω, με εξαίρεση κάποιες περιπτώσεις, δείχνει πως η κοινότητα των επιστημόνων στρέφεται σταδιακά σε peer-to-peer λύσεις, αφού φαίνεται οι τελευταίες να ταιριάζουν καλύτερα στην εξυπηρέτηση των αναγκών τους. Ωστόσο, ο μεγαλύτερος όγκος επικεντρώνεται κυρίως στην απο-

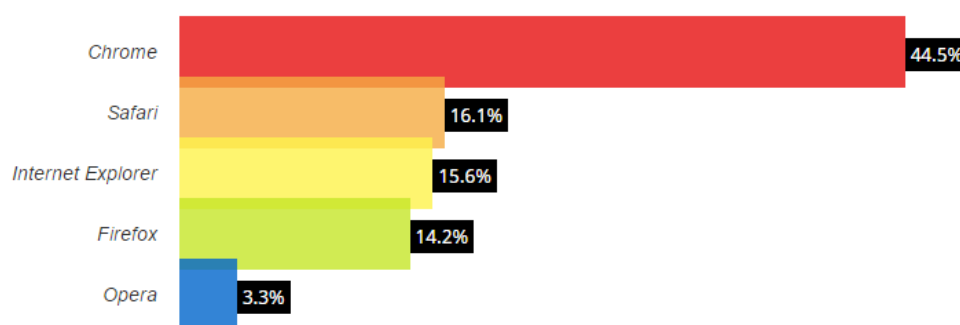
θήκευση και ανάκτηση ιατρικής πληροφορίας ενώ όσον αφορά την υποστήριξη συνεργατικότητας, αυτή περιορίζεται σε στενά γεωγραφικά πλαίσια ή ακόμα σε συγκεκριμένους οργανισμούς (π.χ. νοσοκομεία). Πουθενά επίσης δε φαίνεται να αναδεικνύεται η σημασία της ιατρικής εικόνας στη διάγνωση και τη λήψη ιατρικών αποφάσεων. Αυτό το κενό ευελπιστεί να καλύψει η παρουσιαζόμενη εφαρμογή, χρησιμοποιώντας ένα σύνολο καινοτόμων εργαλείων και προσεγγίσεων.

3 Τεχνολογικό Υπόβαθρο

3.1 Τεχνολογίες Πληροφορικής και Επικοινωνιών

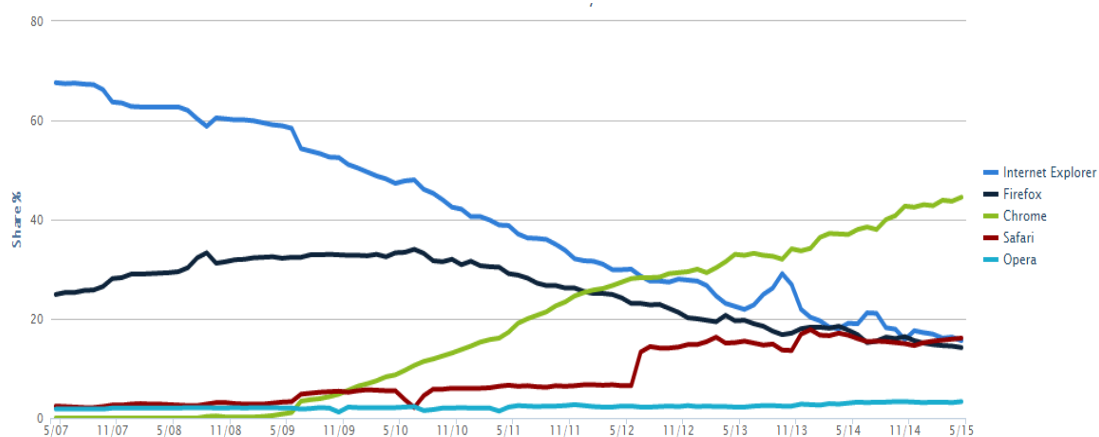
3.1.1 Φυλλομετρητές ιστού (Web browsers)

Σύμφωνα με στατιστικά χρήσης γνωστών browsers που αντλήθηκαν από τον ιστότοπο www.w3counter.com και αφορούν την πιο πρόσφατη καταχώρησή τους τη στιγμή της συγγραφής (Μάιος 2015), οι πιο διαδεδομένοι browsers είναι κατά σειρά ο Chrome, ο Firefox, ο Safari, ο Internet Explorer, ο Firefox και ο Opera. Τα ποσοστά τους παρουσιάζονται στην κάτωθι εικόνα (Εικόνα 9).



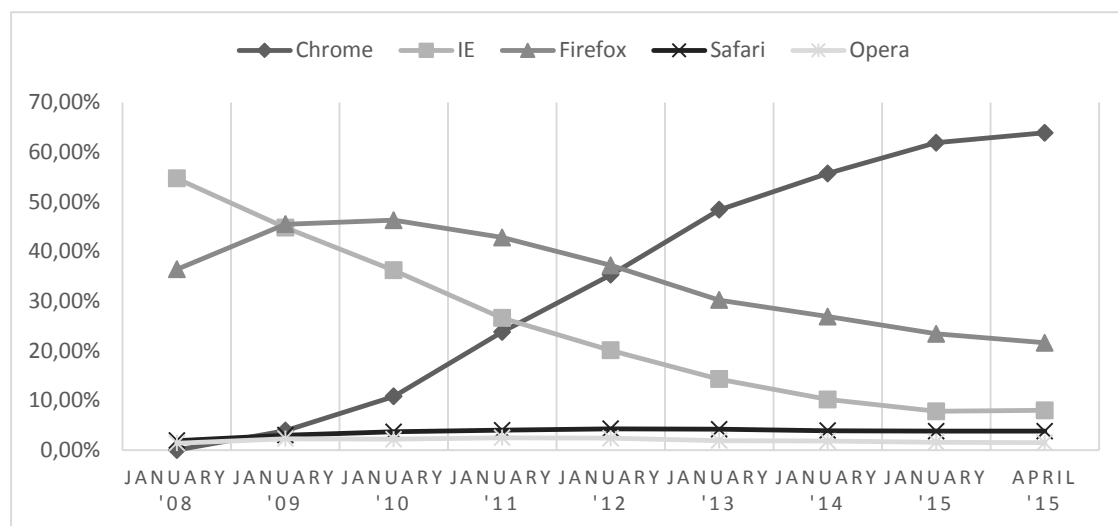
Εικόνα 9. Μεριδίο αγοράς browsers - Μάιος 2015 (εικόνα από www.w3counter.com [26])

Κάνοντας μια μικρή αναδρομή των τελευταίων οκτώ ετών (Εικόνα 10), παρατηρείται ότι στα μέσα του 2007 δύο στους τρεις χρήστες επέλεξαν το browser της Microsoft για την πλοήγησή τους στο Internet. Ορόσημο αποτέλεσε η είσοδος της Google στην εν λόγω αγορά με τον Chrome (Σεπτέμβριος 2008). Από τότε και σταδιακά το μερίδιο του IE άρχισε να συρρικνώνεται ενώ το αντίστοιχο του Chrome να αυξάνει διαρκώς κερδίζοντας χρήστες από όλους τους ανταγωνιστές, ώσπου έφτασε σήμερα να είναι ο αδιαφιλονίκητος κυρίαρχος της αγοράς.



Εικόνα 10. Ιστορικό μεριδίου χρήσης browsers (εικόνα από www.w3counter.com [26])

Για του λόγου το αληθές, παρατίθενται στοιχεία που αντλήθηκαν από μια δεύτερη ιστοσελίδα, ιδιαίτερα δημοφιλή στο χώρο των προγραμματιστών (www.w3schools.com) στην *Εικόνα 11*. Τα στατιστικά που παρουσιάζονται σε αυτήν μπορεί να μην ταυτίζονται απόλυτα με τα προηγούμενα (με μεγαλύτερη διαφορά την εναλλαγή θέσεων μεταξύ Firefox και Internet Explorer), εντούτοις ο Chrome είναι και εδώ στην κορυφή των προτιμήσεων με διαφορά. Αυτή η διαφοροποίηση να σημειωθεί ότι παρατηρείται διότι οι μετρήσεις από το www.w3schools.com αντιστοιχούν στους χρήστες που πραγματοποιούν πρόσβαση στον ιστότοπό τους – δηλαδή προγραμματιστές ως επί το πλείστον.



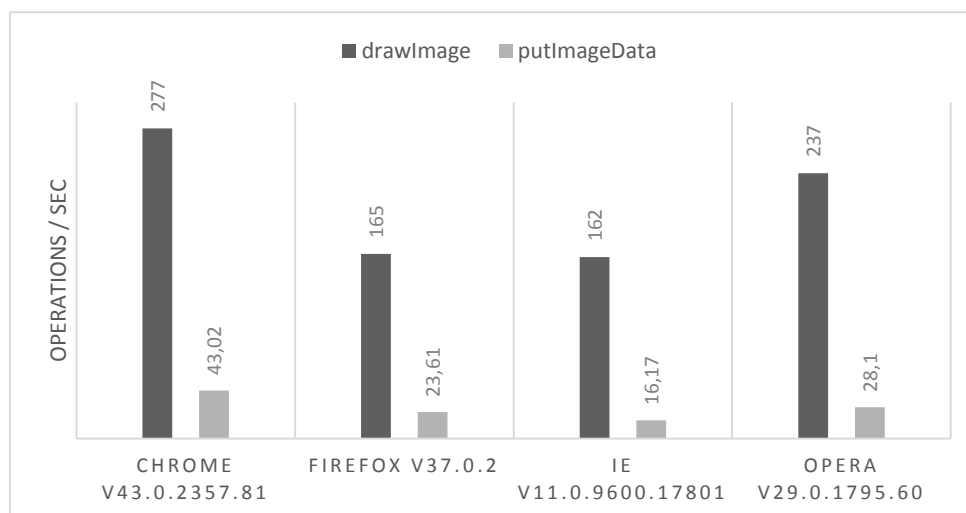
Εικόνα 11. Τάση της αγοράς browsers (στοιχεία από www.w3schools.com [27])

Λαμβάνοντας υπόψιν τα παραπάνω, αλλά και λόγω της προσωπικής προτίμησης του υποφαινόμενου προς το φυλλομετρητή της Google, επιλέχθηκε ο Chrome ως περιβάλλον εξέλιξης της εφαρμογής (εγκατεστημένος πάνω σε λειτουργικό σύστημα Windows). Αυτό βέβαια δεν αποκλείει τη δυνατότητα εκτέλεσής της και στους υπόλοιπους – μικρές αλλαγές ίσως απαιτούνται σε κάποιες περιπτώσεις κυρίως για λόγους συμβατότητας.

Ένας άλλος λόγος, εξίσου σημαντικός, για την επιλογή αυτή είναι η απόδοση κατά την εκτέλεση ορισμένων απαιτητικών εργασιών. Ενδεικτικά, στην *Εικόνα 12* παρουσιάζονται τα αποτελέσματα που εξήχθησαν από την ιστοσελίδα www.jspperf.com⁵, όταν μετρήθηκε η επίδοση τεσσάρων από τους πέντε browsers κατά την εκτέλεση κλήσεων στο Canvas API (απουσιάζει ο Safari λόγω έλλειψης συμβατότητας με το λειτουργικό σύστημα της Microsoft). Συγκεκριμένα, η έκδοση του test case που χρησιμοποιήθηκε βρίσκεται αναρτημένη στο site από την 3^η Απριλίου 2015 και περιλαμβάνει την τοποθέτηση των pixel μιας εικόνας 364x126 στο στοιχείο <canvas> της HTML με τη χρήση δύο διαφορετικών συναρτήσεων που παρέχει για το λόγο αυτό το Canvas API των browsers [28]. Ο Chrome αποδείχτηκε ο ταχύτερος για την εργασία αυτή, χωρίς αυτό να σημαίνει πως γενικά είναι ο ταχύτερος μεταξύ των browsers. Εξάλλου,

⁵ Το site www.jspperf.com παρέχει έναν εύκολο και διαισθητικό τρόπο για τη δημιουργία και το διαμοιρασμό test cases, συγκρίνοντας την επίδοση κομματιών κώδικα σε JavaScript εκτελώντας benchmarks.

οι επιδόσεις όλων μπορεί να διαφέρουν σημαντικά από έκδοση σε έκδοση⁶. Σε άλλες εργασίες (π.χ. σύνθεση μιας μεγάλης συμβολοσειράς από πολλές μικρότερες) μπορεί να υστερεί σε σχέση με τους άλλους, αλλά λόγω της φύσης της αναπτυσσόμενης εφαρμογής, που απαιτεί σε μεγάλο βαθμό αλληλεπίδραση με εικόνα, η χρήση του παρέχει αναμφισβήτητα οφέλη.



Εικόνα 12. Επίδοση κλήσεων του Canvas API μεταξύ των κυριότερων browsers στην πλατφόρμα των Windows

3.1.2 Ανάπτυξη web applications

Μία web-based εφαρμογή είναι στην ουσία μία ιστοσελίδα που στοχεύει στην παροχή πλουσιότερης και πιο περίπλοκης λειτουργικότητας από την απλή προβολή κειμένου και εικόνων. Κατά τη βάση του, μια ιστοσελίδα μπορεί να υλοποιηθεί με χρήση απλώς HTML (Hyper Text Markup Language) [29]. Η γλώσσα HTML γράφεται με τη μορφή στοιχείων (HTML elements) αποτελούμενα από tags, περικλειόμενα από τους χαρακτήρες '<', '>', όπως φαίνεται και στην *Εικόνα 13*. Ο κώδικας μετατρέπεται από το browser σε μια δενδρική δομή αποτελούμενη από JavaScript αντικείμενα-κόμβους, ή όπως αλλιώς ονομάζεται DOM (Document Object Model). Ο σκοπός της μετατροπής αυτής είναι η παροχή μιας προγραμματιστικής διεπαφής για την εκτέλεση σεναρίων (scripting), όπως αφαίρεση, πρόσθεση, αντικατάσταση και τροποποίηση του «ενεργού» HTML εγγράφου χρησιμοποιώντας JavaScript [30]. Στα HTML στοιχεία μπορούν να αποδοθούν γνωρίσματα με τη μορφή ζευγών ονόματος-τιμής (key-value pairs), ώστε να καταστεί δυνατή η παραμετροποίησή τους. Στοιχεία που αντιστοιχούν σε κόμβους στο DOM μπορούν επίσης να δημιουργηθούν δυναμικά χρησιμοποιώντας μεθόδους που παρέχει η JavaScript, όπως η `createElement()`. Το νόημα όλων αυτών είναι πως JavaScript και HTML είναι στενά συνδεδεμένες, υπό την έννοια ότι ο προγραμματισμός σε HTML μπορεί να επιτευχθεί με χρήση JavaScript.

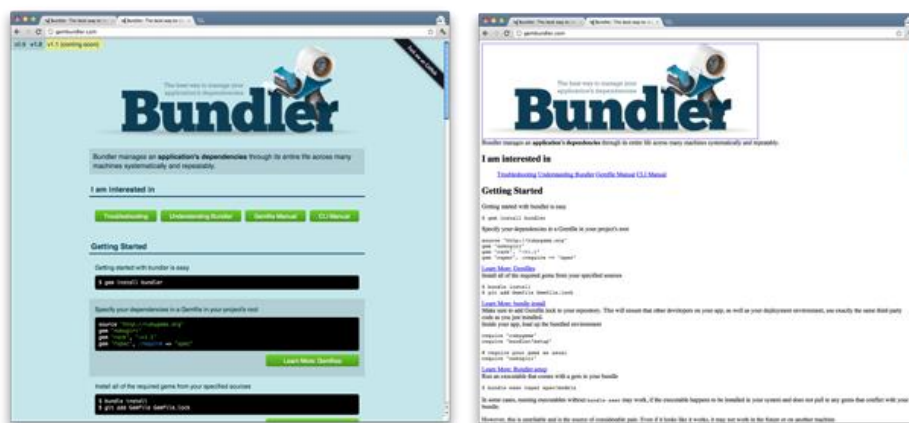
⁶ Για την εκτέλεση του συγκεκριμένου test case χρησιμοποιήθηκαν οι τελευταίες σταθερές εκδόσεις τη στιγμή της συγγραφής, σε σύστημα με Inter Core 2 Duo CPU E7500 @ 2.93GHz, 8GB RAM, ενσωματωμένο υποσύστημα γραφικών Intel Q45/Q43 Express Chipset και λειτουργικό Windows 8.1 Pro 64-bit.

```
<!DOCTYPE html>
<html>
<!-- created 2010-01-01 -->
<head>
<title>sample</title>
</head>
<body>
<p>Voluptatem accusantium
totam rem aperiam.</p>
</body>
</html>
```

HTML

Εικόνα 13. Γραφική απεικόνιση απλού HTML αρχείου (εικόνα από [29])

Η δημιουργία ιστοσελίδων μόνο με HTML ωστόσο κρίνεται ανεπαρκής για τα σημερινά δεδομένα του δικτυακού προγραμματισμού. Οι σύγχρονοι ιστότοποι μπορούν να εννοηθούν ως ένας συνδυασμός δομής (structure), ύφους (style) και διαδραστικότητας (interactivity) [31]. Υπεύθυνες για τα τρία αυτά διαφορετικά στοιχεία είναι οι τεχνολογίες της HTML, CSS και JavaScript, αντίστοιχα, οι οποίες από κοινού συνεισφέρουν στη διανομή πλούσιου περιεχομένου σελίδων ιστού. Η γλώσσα CSS (Cascading Style Sheet) δημιουργήθηκε και προτάθηκε από την κοινοπραξία W3C (World Wide Web Consortium) [32], τον κύριο διεθνή οργανισμό προτύπων για το Web, το 1996. Γενικά χρησιμοποιείται για να παραμετροποιήσει την εμφάνιση του HTML περιεχομένου, προσθέτοντας γνώρισμα όπως στυλ γραμματοσειράς, χρώμα, διαστάσεις κτλ. (βλ. *Εικόνα 14*). Παλιότερα, κάτι τέτοιο ήταν εφικτό και με χρήση μόνο HTML, αλλά ήταν μια επίπονη προγραμματιστική διαδικασία καθώς για κάθε στοιχείο έπρεπε να γραφτούν όλα τα γνώρισμα «με το χέρι». Με την εισαγωγή της CSS, σε ένα ή περισσότερα στοιχεία μπορεί να ανατεθεί μια κλάση (class attribute) και ο κώδικας που είναι υπεύθυνος για τους κανόνες εμφάνισης της κλάσης να συνταχτεί μία φορά και η αντιστοίχιση με τα στοιχεία να γίνεται αυτόματα από το browser.



Εικόνα 14. Άποψη ιστοσελίδας με χρήση και χωρίς χρήση CSS

Επιπρόσθετα, όπως η μορφοποίηση που παρέχει η CSS μπορεί να συμπεριληφθεί στο ίδιο αρχείο HTML με τη χρήση του tag `<style>`, υπάρχει μέριμνα και για τη JavaScript με το αντίστοιχο tag `<script>`. Η τελευταία χρησιμοποιείται για την προσθήκη λειτουργικότητας σε μια ιστοσελίδα. Διαχειρίζεται διαδραστικά στοιχεία – όπως

οι φόρμες κειμένου και τα κουμπιά, εκτελεί υπολογισμούς, αναλαμβάνει την επικοινωνία με τον εξυπηρετητή και γενικά ο,τιδήποτε μπορεί να απαιτεί ένας προγραμματιστής από μια κλασική γλώσσα προγραμματισμού. Προηγμένα χαρακτηριστικά, όπως οι τεχνολογίες WebGL και HTML5 Canvas, είναι προσβάσιμα μέσω εξειδικευμένων JavaScript APIs. Όταν μια ιστοσελίδα περιέχει JavaScript – και σχεδόν όλες σήμερα περιέχουν, ο κώδικας λαμβάνεται από τον εξυπηρετητή στο μηχάνημα του χρήστη (client), όπου μεταφράζεται και εκτελείται από το browser. Με τη σημερινή αυξανόμενη υπολογιστική ισχύ των συσκευών αλλά και τη διαρκή βελτίωση των μηχανών επιφορτισμένων με την εκτέλεση JavaScript (JavaScript engines⁷) που αποτελούν τον πυρήνα των σύγχρονων browsers, πολύπλοκος και ογκώδης κώδικας εκτελείται τοπικά βασιζόμενος στην ταχύτητα του εγκατεστημένου υλισμικού και όχι του δικτύου.

Από άποψη οργάνωσης, είναι δυνατό να συμπεριληφθεί σε ένα μόνο HTML αρχείο ο κώδικας όλων των παραπάνω γλωσσών (με τη χρήση των κατάλληλων tags), αλλά γενικά κάτι τέτοιο αποθαρρύνεται γιατί απουσιάζει η αυτονομία των επιμέρους συστατικών στοιχείων (modularity). Επίσης, ο παραγόμενος κώδικας είναι ιδιαίτερα δύσκολο να ελεγχθεί και να αποσφαλματωθεί. Αντί αυτού, προτιμάται η λύση των πολλών διαφορετικών αρχείων, όπου το καθένα αυστηρά θα περιέχει κώδικα μιας γλώσσας. Η σπονδυλωτή (modular) προσέγγιση αυτή έχει τα οφέλη της ευκολότερης επεκτασιμότητας, αναγνωσιμότητας και συντηρησιμότητας. Για το σκοπό αυτό έχουν επίσης αναπτυχθεί ειδικά εργαλεία και μέθοδοι που διευκολύνουν ακόμα περισσότερο το έργο του προγραμματιστή – κάποια από αυτά χρησιμοποιήθηκαν κατά την εξέλιξη της παρουσιαζόμενης εφαρμογής και θα αναλυθούν στη συνέχεια.

3.1.3 Συμβατότητα web browsers

Ο προγραμματισμός για την πλατφόρμα των browsers, παρ' όλα αυτά είναι απαιτητικός από την άποψη ότι διάφορες πλευρές της λειτουργικότητας που παρέχουν οι τεχνολογίες του διαδικτύου – HTML, CSS, JavaScript, δεν έχουν υλοποιηθεί ομοίωμα από όλους τους browser που εξετάστηκαν στην αρχή του κεφαλαίου. Αυτό απορρέει από το γεγονός ότι οι τελευταίοι εξελίσσονται και συντηρούνται από διαφορετικές εταιρίες με διαφορετικούς επιχειρησιακούς στόχους και στρατηγικές υλοποίησης. Κατά τα τέλη της δεκαετίας του '90 αυτό αποδείχτηκε καταστροφικό καθώς οι χρήστες έφτασαν στο σημείο να έχουν να επιλέξουν μεταξύ browsers με διαφορετικά χαρακτηριστικά χωρίς κανένα συντονισμό μεταξύ των ατόμων που τους παρείχαν. Ωστόσο, με τη δημιουργία του World Wide Web Consortium, τα πράγματα άλλαξαν προς το καλύτερο. Η οργάνωση άρχισε να κινείται προς την κατεύθυνση της προτυποποίησης των γλωσσών και των τεχνολογιών που χρησιμοποιούνται για την παροχή διαδικτυακού περιεχομένου. Αυτό συνέβη σε αρκετά στάδια και αρκετές επανεκδόσεις των προτύπων απαιτήθηκαν ώστε να φτάσει σήμερα η εικόνα του διαδικτύου να μοιάζει πιο ενοποιημένη από ποτέ.

Εντούτοις, εξακολουθούν να υπάρχουν ακόμα ορισμένες μικρές διαφοροποιήσεις, το οποίο σε κάθε περίπτωση κρίνεται δικαιολογημένο αφού διαφορετικές εταιρίες συντηρούν διαφορετικούς browsers. Για παράδειγμα, το WebGL υποστηρίζεται από το Chrome από το Φεβρουάριο του 2011, ενώ εισήχθη στον Internet Explorer μόλις τον

⁷ Χαρακτηριστικό παράδειγμα αποτελεί η V8 JavaScript engine που εξελίχθηκε από τη Google για τη χρησιμοποίησή της στο Chrome. Από την πρώτη της έκδοση το 2008, έχει δεχθεί μεγάλες βελτιώσεις και επεκτάσεις με συνέπεια σήμερα να αποτελεί μια υψηλών επιδόσεων standalone μηχανή που μπορεί να ενσωματωθεί σε ανεξάρτητα project, όπως για server-side JavaScript στο Node.js [56].

Οκτώβριο του 2013 με την 11^η επανέκδοσή του. Επίσης, όλα τα χαρακτηριστικά που προσφέρει η CSS δεν είναι καθολικώς υποστηριζόμενα ή τουλάχιστον η υποστήριξη σε ορισμένους browsers υπολείπεται έναντι άλλων. Το ευτύχημα είναι πως τα πιο συνηθισμένα και ευρέως χρησιμοποιούμενα στοιχεία τείνουν να υποστηρίζονται από όλους. Ο ιστότοπος www.caniuse.com παρέχει στους ενδιαφερόμενους, προγραμματιστές και μη, εκτενή καταγραφή των υποστηριζόμενων χαρακτηριστικών ανά έκδοση⁸ και οικογένεια browser. Ακόμα, ένα ζήτημα που όλοι όσοι εξελίσσουν λογισμικό για browser πρέπει να γνωρίζουν είναι πως οι διερμηνείς JavaScript μπορεί να διαφέρουν στην υλοποίησή τους.

Ως αναμενόμενο, όλες αυτές οι αποκλίσεις αποτέλεσαν πηγή προβλημάτων κατά τη διάρκεια συγγραφής της εφαρμογής και αρκετός χρόνος αφιερώθηκε στον έλεγχο και την επανεγγραφή συγκεκριμένων κομματιών κώδικα που δεν εκτελούνταν επιτυχώς σε διαφορετικούς browsers. Υπάρχει μια μέθοδος γνωστή ως "browser sniffing" που μπορεί να εφαρμοστεί σε τέτοιες περιπτώσεις, κατά την οποία εξακριβώνεται ο τύπος και η έκδοση του browser και ο κώδικας που ακολουθεί διαμορφώνεται ανάλογα. Ωστόσο, ο W3C αποδοκιμάζει την πρακτική αυτή, καθώς υπάρχουν πάρα πολλές διαφορετικές εκδοχές για τις οποίες ο προγραμματιστής πρέπει να ανταποκριθεί ξεχωριστά, οδηγούμενος έτσι να συντάξει πολλαπλάσιες γραμμές κώδικα για την εκτέλεση εργασιών κατά τα άλλα τετριμμένων. Αντί γι' αυτή, προτείνει τη λεγόμενη ανίχνευση χαρακτηριστικών, η οποία επιτρέπει στον προγραμματιστή να «ρωτήσει» το εκάστοτε περιβάλλον εκτέλεσης – το browser δηλαδή – αν υποστηρίζει κάποιο συγκεκριμένο χαρακτηριστικό το οποίο σκοπεύει να χρησιμοποιήσει. Η λύση αυτή είναι σίγουρα πιο κομψή και ελκυστική, όπως επίσης και λιγότερο επιρρεπής σε λάθη.

Προσεγγίζοντας το ζήτημα από τη σκοπιά του κώδικα, η ανίχνευση του τύπου και της έκδοσης του browser γίνεται μέσω του JavaScript αντικειμένου navigator και συγκεκριμένα του πεδίου που παρέχει, userAgent. Η πρόσβαση σε αυτό επιστρέφει μια συμβολοσειρά, όπως:

```
Mozilla/5.0 (Windows NT 6.3; WOW64; rv:38.0) Gecko/20100101  
Firefox/38.0
```

Η συμβολοσειρά πρέπει να αναλυθεί γραμματικά (ανακτώντας με τον τρόπο αυτό την πληροφορία για το browser ως Firefox v38), γεγονός που εγκυμονεί κινδύνους για λάθη, ανάλογα με τη μέθοδο που επιλέγεται για την ανάλυση και τις υποθέσεις που γίνονται για τη μορφή της συμβολοσειράς. Αντί αυτού, ανιχνεύοντας ένα συγκεκριμένο χαρακτηριστικό απαιτεί κώδικα της μορφής:

```
var c = document.createElement('canvas');  
if (c.getContext('webgl') || c.getContext('experimental-  
webgl')) {  
    // run some code that relies on WebGL  
} else {  
    // do something else, perhaps with Canvas  
}
```

⁸ Και η έλευση μιας επανέκδοσης του ίδιου browser πολλές φορές μπορεί να προξενήσει προβλήματα ασυμβατότητας, καθώς ορισμένα χαρακτηριστικά μπορεί να πάψουν να υποστηρίζονται πλέον ή μπορεί να συνεχίσουν να υποστηρίζονται με αλλαγές στην υλοποίησή τους που είναι πιθανόν να οδηγήσουν υπάρχουσες εφαρμογές που βασίζονται σε αυτά σε απρόβλεπτη συμπεριφορά.

Ελέγχοντας για χαρακτηριστικά με τον παραπάνω τρόπο, ο προγραμματιστής απαλλάσσεται από την ανάγκη να γνωρίζει ποια έκδοση, ποιου browser παρέχει τις δυνατότητες που επιθυμεί να αξιοποιήσει με τον κώδικά του. Για του λόγου το αληθές, όλες οι δημοφιλείς βιβλιοθήκες που έχουν αναπτυχθεί σε JavaScript (jQuery, THREE.js, κτλ.) ακολουθούν τη λογική αυτή για να εξασφαλίσουν πως η λειτουργικότητα την οποία παρέχουν είναι ανεξάρτητη από το browser στον οποίο εκτελούνται.

3.1.4 Single Page Applications and Web frameworks

Στο σημείο αυτό έχει ιδιαίτερη αξία να εξεταστεί η πρόοδος που έχει γίνει στο χώρο του δικτυακού προγραμματισμού (web development) για την περαιτέρω στοιχειοθέτηση της επιλογής του browser ως περιβάλλον εκτέλεσης της παρουσιαζόμενης εφαρμογής.

Η ιστορία του web development ξεκινά από την απαρχή του Internet, όπου η HTML χρησιμοποιείτο για το «σερβίρισμα» στατικού περιεχομένου, γεγονός που σηματοδότησε την εποχή των στατικών σελίδων. Κάποια στιγμή εμφανίστηκε το λεγόμενο CGI (Common Gateway Interface) και μαζί του η εποχή του δυναμικού περιεχομένου. Το CGI χρησιμοποιήθηκε για τη δημιουργία δυναμικού περιεχομένου σε απλές ιστοσελίδες αλλά και διαδικτυακές εφαρμογές (web applications). Ακολούθησε η περίοδος κατά την οποία δυναμικοί ιστότοποι, των οποίων οι σελίδες δημιουργούνταν δυναμικά, ανάλογα με το ευρύτερο πλαίσιο (context), από κάποιο backend σύστημα γραμμένο σε PHP/ASP/JSP/ColdFusion, κτλ., και απεικονίζονταν από τους πελάτες (thin clients). Σύντομα έκαναν την εμφάνισή τους τεχνολογίες όπως Flash και Silverlight, που στόχευαν να κλείσουν την «ψαλίδα» μεταξύ desktop και web εφαρμογών, κυρίως από άποψη λειτουργικότητας και διαδραστικότητας.

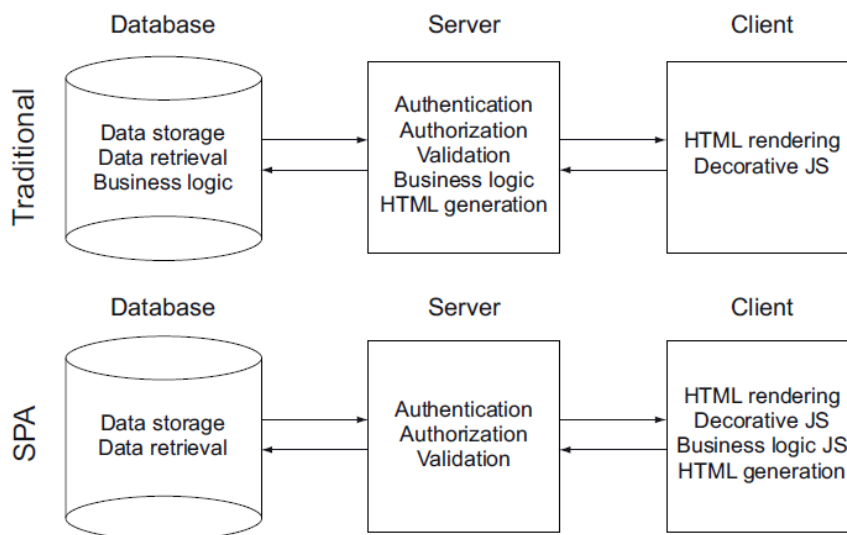
Η μεγάλη αλλαγή, ωστόσο, επιτεύχθηκε με την είσοδο των AJAX (Asynchronous JavaScript and XML) τεχνικών, οι οποίες επέτρεψαν για πρώτη φορά τη μετατόπιση του ελέγχου από την πλευρά του εξυπηρετητή στην πλευρά του πελάτη. Οι τεχνικές αυτές συνδυάστηκαν με πλαίσια εργασίας στον εξυπηρετητή (server-side frameworks) που επέτρεπαν το «διασκορπισμό» της λογικής παρουσίασης (presentation logic) μεταξύ αυτού και του πελάτη. Σηματοδότησαν δε την αρχή μιας διαφορετικής προσέγγισης στο δικτυακό προγραμματισμό, κατά την οποία ο πελάτης μπορεί να επιλεκτικά να αιτείται δεδομένα από το backend, να τα διαχειρίζεται από την πλευρά του όταν τα λάβει και τέλος να τα απεικονίσει όπως απαιτείται στην οθόνη του χρήστη.

Η έλευση της HTML5 ύστερα στόχευσε σε σημαντικές βελτιώσεις στους browsers και στις δυνατότητές τους μέσα από την αναθεώρηση και την εκλέπτυνση της δομής της ίδιας της HTML, του συνοδευόμενου JavaScript API και της CSS. Όλα αυτά άφηναν υποσχέσεις για ένα πιο ώριμο και αποδοτικό περιβάλλον εξέλιξης εφαρμογών πελάτη. Η διαδικτυακή κοινότητα, που εν τω μεταξύ για την παροχή πλούσιου και διαδραστικού περιεχομένου είχε στραφεί προς τις RIAs (Rich Internet Application) – εφαρμογές οι οποίες έχουν πολλά από τα χαρακτηριστικά των native αλλά συνήθως η εκτέλεσή τους απαιτεί την εγκατάσταση κάποιας επέκτασης στο browser, αντιλήφθηκε αυτή τη νέα δυναμική που αναπτυσσόταν και πλέον από το 2012, σύμφωνα με τις τάσεις που καταγράφει η Google [33] [34], άρχισε σταδιακά να εγκαταλείπει τα frameworks που στηρίζονταν σε plug-ins και third-party εργαλεία. Τη μεταστροφή αυτή βοήθησε και η αντικατάσταση των τελευταίων με πληθώρα νέων που αξιοποιούσαν τα βελτιωμένα APIs που προσέφεραν πλέον οι browsers και την επεξεργαστική ισχύ των προσωπικών

υπολογιστών και διευκόλυναν την ανάπτυξη και συντήρηση μεγάλης έκτασης εφαρμογών.

Τα προγράμματα-πελάτες που παλαιότερα ονομάζονταν thin clients (από την άποψη «αρμοδιοτήτων» και υπολογιστικού φόρτου που αναλάμβαναν) τώρα μεταλλάσσονται σε thick clients. Ενώ συνηθιζόταν η απαραίτητη για την παρουσίαση περιεχομένου λογική να διαχειρίζεται από το backend/εξυπηρετητή και οι πελάτες απλώς να προσθέτουν στην εμφάνιση κάποιες στοιχειώδεις δυνατότητες αλληλεπίδρασης, τώρα το σενάριο αλλάζει, με τους browsers και τη JavaScript να είναι παραπάνω από ικανοί για να αναλάβουν υπολογιστικά πολύπλοκες εργασίες. Σύμφωνα, λοιπόν, με τη νέα τάση στο χώρο, η λογική παρουσίασης μετατοπίζεται εκεί που για πολλούς έχει νόημα, δηλαδή στο frontend/πελάτη.

Συγκεκριμένα, ήρθε πλέον στο προσκήνιο η έννοια της SPA. Η μονοσέλιδη εφαρμογή (single-page application - SPA), είναι μια web εφαρμογή ή ένας ιστότοπος που «χωρά» σε μία μόνο ιστοσελίδα με το σκοπό να παρέχει μία πιο άμεση και ομαλή εμπειρία χρήσης συγκρίσιμη με μίας αντίστοιχης εφαρμογής επιφάνειας εργασίας (desktop application). Στο περιβάλλον της SPA, όλος ο απαραίτητος κώδικας – HTML, JavaScript και CSS – ανακτάται με μία μοναδική φόρτωση σελίδας ή οι κατάλληλοι πόροι (resources) φορτώνονται δυναμικά και προστίθενται στη σελίδα, συνήθως ως αποτέλεσμα της αλληλεπίδρασης με το χρήστη. Η σελίδα δεν επαναφορτώνεται σε κανένα σημείο της διαδικασίας, ούτε ο έλεγχος μεταφέρεται σε άλλη σελίδα, αν και τελευταίες τεχνολογίες παρέχουν την αίσθηση της πλοηγησιμότητας σε διαφορετικές λογικές σελίδες μέσα στην εφαρμογή. Η αλληλεπίδραση με αυτή συχνά περιέχει την έννοια της δυναμικής επικοινωνίας με κάποιον web server στο παρασκήνιο (Εικόνα 15).

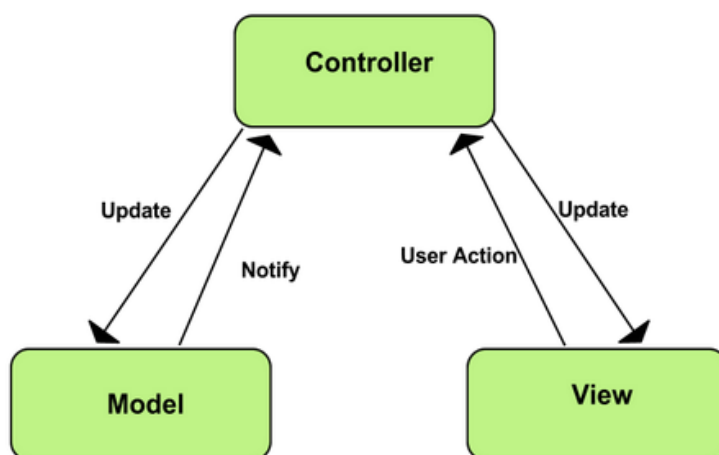


Εικόνα 15. Οργάνωση της βάσης δεδομένων, του εξυπηρετητή και του πελάτη

Καθώς η ποσότητα και η πολυπλοκότητα του κώδικα που πλέον πρέπει να συσταχθεί για τα προγράμματα-πελάτες αυξήθηκε σημαντικά, δεν είναι πλέον δυνατή η εμφώλευσή του στα <script> tags της HTML σελίδας. Η λύση και πάλι δόθηκε εφαρμόζοντας πρακτικές και σχεδιαστικές τεχνολογίες (design patterns) που χρησιμοποιούνταν μέχρι πρότινος για την οργάνωση του κώδικα στον εξυπηρετητή. Η πιο γνωστή, πάνω στην οποία βασίστηκε επίσης και η δόμηση των περισσότερων JavaScript frame-

works που συναντώνται σήμερα, είναι η λεγόμενη Model-View-Controller (MVC pattern). Η τεχνοτροπία αυτή διαχωρίζει τα επιμέρους συστατικά μιας εφαρμογής σε τρία μέρη (Εικόνα 16):

- Μοντέλα (Models) που αναπαριστούν τις σχετικές με την εφαρμογή πληροφορίες και δεδομένα, όπως συγκεκριμένες κλάσεις-δοχεία (container classes) δεδομένων. Τα μοντέλα μπορούν να ενημερώσουν τυχόν παρατηρητές όταν η κατάσταση τους αλλάζει (π.χ. όταν κάποια πληροφορία που κρατούν ενημερώνεται ή διαγράφεται).
- Όψεις (Views) που τυπικά θεωρούνται ως η διεπαφή του χρήστη με την εφαρμογή (π.χ. ο κώδικας HTML και CSS). Πρέπει να γνωρίζουν για την ύπαρξη των Μοντέλων έτσι ώστε να τα παρατηρούν, αλλά δεν επικοινωνούν κατευθείαν μαζί τους.
- Ελεγκτές (Controllers) που υλοποιούν τη λογική παρουσίασης (presentation logic) της εφαρμογής. Αυτοί είναι που παίρνουν τις αποφάσεις και ο συνδεδεμένος κρίκος μεταξύ Μοντέλων και Όψεων.



Εικόνα 16. Συνήθεις αλληλεπιδράσεις των MVC συστατικών

Η πληθώρα των MVC frameworks που υπάρχουν διαθέσιμα σήμερα, ανάλογα με τους στόχους τους, δεν υλοποιούν με τον ίδιο τρόπο το παραπάνω προγραμματιστικό μοτίβο αλλά συνήθως κάποια παραλλαγή του, με σκοπό πάντα όμως την επιβολή της σχεδιαστικής αρχής γνωστής ως SoC (Separation of Concerns). Παρατηρείται, για παράδειγμα, η συγχώνευση του ρόλου του ελεγκτή και της όψης σε μία οντότητα με αυξημένες αρμοδιότητες και λειτουργικότητα. Για αυτό το λόγο, οι υλοποιήσεις που στηρίζονται σε τέτοιου είδους πιο «χαλαρές» προσεγγίσεις συναντώνται στη βιβλιογραφία και ως MV*.

Αποτέλεσμα μιας τέτοια προσέγγισης αποτελεί και το Backbone.js, το framework πάνω στο οποίο στηρίχθηκε η εξέλιξη της συγκεκριμένης εφαρμογής. Τη στιγμή της συγγραφής βρισκόταν μέσα στις πρώτες επιλογές από προγραμματιστές που επιθυμούσαν τα οφέλη που απορρέουν από την εύκολη και διαισθητική οργάνωση και διαχείριση του κώδικά τους χωρίς να ξοδέψουν χρόνο στην εξοικείωση με νέες έννοιες ή προγραμματιστικά στιλ που άλλα αντίστοιχα frameworks επιβάλλουν (π.χ. Angular.js, Ember.js, κτλ.). Αναπτύχθηκε από τον Jeremy Ashkenas και σχεδιάστηκε για την εξέλιξη

SPAs και τη διατήρηση των διαφόρων μερών τους συγχρονισμένους με κάποιο REST API. Αποτελεί μια δοκιμασμένη λύση που έχει χρησιμοποιηθεί από μεγάλες εταιρίες για την κατασκευή πολύπλοκων εφαρμογών, όπως Walmart, SoundCloud και LinkedIn.

Σε συμμόρφωση με το MV* μοτίβο, το Backbone παρέχει τους εξής μηχανισμούς: συμβάντα (events), μοντέλα (models), συλλογές (collections) και όψεις (views). Τα μοντέλα αναπαριστούν την πληροφορία σε δομές-κλάσεις, οι συλλογές αποτελούν λίστες από όμοιου είδους μοντέλα και οι όψεις διαχειρίζονται την τρόπο εμφάνισης της HTML σελίδας. Οι όψεις, στην ουσία, παράγουν τον κώδικα HTML (μέσω δημιουργίας συμβολοσειρών ή τη χρησιμοποίηση ειδικού σκοπού προτύπων – HTML templates) που εισάγεται στο DOM και μπορεί να οπτικοποιηθεί στην οθόνη του χρήστη. Τέλος, τα συμβάντα είναι αντικείμενα που χρησιμοποιούνται για την παρακολούθηση των αλλαγών κατάστασης σε όλα τα μοντέλα, συλλογές και όψεις. Να σημειωθεί πως από το δημιουργό έχει προβλεφθεί η εύκολη παραμετροποίηση όλων αυτών των στοιχείων/αντικειμένων μέσω μηχανισμών κληρονομικότητας (inheritance).

Η ενσωμάτωση του Backbone σε οποιοδήποτε σελίδα προϋποθέτει την εισαγωγή και δύο άλλων πολύ γνωστών βιβλιοθηκών/frameworks, του jQuery και Underscore.js. Η βιβλιοθήκη jQuery είναι γρήγορη, σχετικά μικρή και πλούσια σε χαρακτηριστικά. Παρέχει απλότητα σε συνήθειες για τον προγραμματιστή εργασίες όπως διάσχιση και χειρισμό του DOM, διαχείριση συμβάντων (event handling), animations και Ajax επικοινωνία, μέσω ενός εύχρηστου και λειτουργικού API ανεξαρτήτως browser. Με ένα συνδυασμό ευελιξίας και επεκτασιμότητας, η jQuery έχει επηρεάσει βαθιά τον προγραμματισμό σε JavaScript, γι' αυτό και σήμερα χρησιμοποιείται σχεδόν από όλους τους προγραμματιστές που ασχολούνται με το frontend εφαρμογών. Η Underscore, τώρα, αποτελεί μια συλλογή από χρήσιμες και πρακτικές μεθόδους που παρέχει αυξημένη λειτουργικότητα χωρίς την επέκταση των κλασικών JavaScript αντικειμένων που ενσωματώνουν οι σύγχρονοι browsers.

3.1.5 Γραφικά για εφαρμογές ιστού

Η ανάπτυξη μιας διαδραστικής εφαρμογής βασισμένης σε browser με σκοπό την απεικόνιση δεδομένων ιατρικής εικόνας, προϋποθέτει την εξέταση των διαθέσιμων εργαλείων για τη δημιουργία δισδιάστατων και τρισδιάστατων γραφικών μέσα στο browser. Γενικά, στο παρελθόν διάφορες μέθοδοι παρέχονταν για το σκοπό αυτό από τους δημιουργούς τους αλλά γεγονός είναι πως διέφεραν από υλοποίηση σε υλοποίηση. Μόλις πρόσφατα, αρχικά με την εισαγωγή της HTML5 και αργότερα του WebGL, αναδείχθηκαν κάποια πρότυπα που έτυχαν ευρείας συμμόρφωσης.

Η πέμπτη έκδοση της HTML, μεταξύ των άλλων, εισήγαγε και μια σειρά από σημαντικά χαρακτηριστικά που σχεδιάστηκαν για τη διευκόλυνση του χειρισμού πολυμέσων και γραφικού περιεχομένου στον ιστό χωρίς να απαιτείται η χρήση «κλειστών» επεκτάσεων (proprietary plug-ins) και APIs. Ένα από αυτά τα νέα χαρακτηριστικά είναι το στοιχείο `<canvas>`, ένα «προγραμματιζόμενο» στοιχείο γραφικής απεικόνισης που στην ουσία αποτελεί ένα χαμηλού επιπέδου διαδικαστικό μοντέλο το οποίο ενημερώνει ένα bitmap. Αρχικά παρουσιάστηκε από την Apple [35]. Το συστατικό αυτό συνιστά τη βάση για τις πιο περίπλοκες τεχνικές απεικόνισης δισδιάστατου και τρισδιάστατου περιεχομένου στους σύγχρονους browsers.

3.1.6 Δισδιάστατα γραφικά

Παραδοσιακά, η απεικόνιση δισδιάστατων γραφικών αποτελούσε αναπόσπαστο κομμάτι των περισσότερων browsers. Εκτός από την απλή εμφάνιση αρχείων εικόνας (JPEG, PNG, BMP, κτλ.), υπήρχε μια ποικιλία από προσεγγίσεις και μεθόδους που επέτρεπαν την παραγωγή και το χειρισμό δισδιάστατου γενικά περιεχομένου. Η CSS, όπως παρουσιάστηκε παραπάνω, χρησιμοποιείται για να αλλάξει τον τρόπο εμφάνισης ενός HTML στοιχείου, αλλά δεν κρίνεται πραγματικά κατάλληλη για τη δημιουργία περίπλοκων γραφικών λόγω απουσίας εντολών σχεδίασης. Το κενό αυτό προσπάθησε να καλύψει ήδη από το 1999 ο οργανισμός W3C όταν ξεκίνησε να προδιαγράφει ένα νέο πρότυπο εικόνων, το SVG (Scalable Vector Graphics) [36]. Παραστατικά, το SVG μπορεί να θεωρηθεί ως το αντίστοιχο της HTML για τα γραφικά του υπολογιστή. Είναι μια γλώσσα σήμανσης εγγράφου (markup language) για την περιγραφή όλων των πτυχών μιας εικόνας, από τη γεωμετρία των σχημάτων, στο στυλ του κειμένου, στην κίνηση και την παρουσίαση πολυμέσων, όπως βίντεο και ήχος. Είναι πλήρως διαδραστική, και περιλαμβάνει επίσης τη δική της έννοια του προγραμματιζόμενου DOM. Υποστηρίζει ένα μεγάλο εύρος από οπτικά χαρακτηριστικά όπως χρωματική διαβάθμιση (gradient), αδιαφάνεια (opacity), φίλτρα, κ.ά. Η χρήση εικόνων SVG επιτρέπει πλήρως κλιμακούμενα (scalable), ομαλά, επαναχρησιμοποιούμενα γραφικά για μια ποικιλία περιπτώσεων (παιχνίδια, επιστημονικές απεικονίσεις, κτλ.). Υποστηρίζεται δε από τους περισσότερους σημερινούς browsers, κινητές συσκευές και set-top boxes. Μπορούν να παραχθούν από γλώσσες για client-side ή server-side προγραμματισμό, αλλά συνήθως προτιμάται εξειδικευμένο λογισμικό που έχει εξελιχθεί για την απλοποίηση της όλης διαδικασίας.

Τελικά, η πιο πρόσφατη εναλλακτική ήρθε με την άφιξη της HTML5. Το Canvas είναι ένα προγραμματιζόμενο γραφικό στοιχείο, υποστηριζόμενο από όλους τους μεγάλους browsers, που μπορεί να ελεγχθεί με JavaScript, και αποτελεί σήμερα τη πιο δημοφιλή επιλογή για απεικόνιση δισδιάστατου υλικού στο διαδίκτυο. Στην ουσία, πρόκειται για μια ορθογώνια περιοχή της HTML σελίδας που μπορεί να ζωγραφιστεί με χρήση ενσωματωμένων μεθόδων για το σκοπό αυτό. Είναι ένα αρκετά χαμηλού επιπέδου μοντέλο χωρίς να διαθέτει αντίληψη περί γράφων σκηνής (scene graphs) ή παρουσία DOM και για το λόγο αυτό είναι ιδιαίτερα αποδοτικό. Προγραμματιστές, λοιπόν, είναι σε θέση να εξελίσσουν παιχνίδια ή ακόμη και εφαρμογές πλήρους χαρακτηριστικών χρησιμοποιώντας το Canvas API μόνο ή σε συνδυασμό με SVG. Να σημειωθεί πως πριν την έλευση του Canvas, αντίστοιχη λειτουργικότητα παρείχαν στους browsers κάποιες επεκτάσεις που απαιτούσαν ξεχωριστή εγκατάσταση. Οι πιο γνωστές εκ των οποίων είναι ο Flash Player της Adobe (συνεχίζει να υποστηρίζεται από κάποιες συσκευές) και τα Java applets (θεωρούνται πλέον παρωχημένα).

3.1.7 Τρισδιάστατα γραφικά

Τα τρισδιάστατα γραφικά συνήθως ορίζονται από ένα χώρο καρτεσιανών συντεταγμένων στον οποίο βρίσκονται τοποθετημένα τρισδιάστατα αντικείμενα, καθώς επίσης και ένα αντικείμενο που λειτουργεί ως κάμερα μέσα από την οποία η όλη σκηνή προβάλλεται με τη βοήθεια ενός πίνακα προβολής (projection matrix). Τα τρισδιάστατα γραφικά είναι γενικά περισσότερο απαιτητικά από άποψη απαιτούμενων υπολογιστικών πόρων σε σχέση με τα δισδιάστατα λόγω των πιο περίπλοκων μαθηματικών υπολογισμών που περιλαμβάνουν. Επιπρόσθετα, η απεικόνιση μιας τρισδιάστατης σκηνής (3D

rendering) σημαίνει πως η εικόνα πρέπει να ανανεώνεται με ρυθμό 60 καρέ το δευτερόλεπτο, στόχος που αποτελεί μεγάλη πρόκληση για τους προγραμματιστές.

3.1.7.1 Προ-WebGL

Γενικά μιλώντας, η ιστορία των τρισδιάστατων γραφικών στο διαδίκτυο μπορεί να διαχωριστεί στην εποχή πριν και μετά την προτυποποίηση του WebGL. Πριν την έλευση του τελευταίου, ο κανονικός τρόπος για την απόδοση τρισδιάστατου περιεχομένου στο browser εξαρτιόταν άμεσα από ειδικές για το σκοπό αυτό επεκτάσεις που ο χρήστης καλείτο να «κατεβάσει» και να εγκαταστήσει στο μηχάνημά του. Ο Flash Player, που ήδη έχει αναφερθεί για τη χρήση του στα δισδιάστατα γραφικά, ήταν μια από τις κυρίαρχες επεκτάσεις και στο χώρο αυτό. Εντούτοις, η ύπαρξη των επεκτάσεων αυτών δεν ήταν πανάκεια. Χαρακτηριστικό είναι το γεγονός ότι η κάθε μία τους έπρεπε να έχει διαφορετικές υλοποιήσεις για να μπορεί να εκτελείται πάνω από διαφορετικές πλατφόρμες – Windows, Linux, MacOS. Επιπρόσθετα, η Apple από κάποια στιγμή και μετά, αρνήθηκε να υποστηρίξει αυτή τη νέα τάση με τις επεκτάσεις και, αντί αυτού, να επενδύσει στα ανερχόμενα πρότυπα της HTML5. Η εικόνα αυτή, λοιπόν, στην κοινότητα των browsers έδειχνε πως απαιτούσε μια πιο γενική και κοινώς αποδεκτή στο ζήτημα αυτό.

Διάφορες άλλες επεκτάσεις δημιουργήθηκαν, εν τω μεταξύ, προς την κατεύθυνση της εισαγωγής τρισδιάστατου περιεχομένου στο διαδίκτυο. Άλλα χαρακτηριστικά παραδείγματα αποτελούν τα Java applets, είδος μικροεφαρμογών γραμμένων σε Java, που εκτελούνται στο μηχάνημα του χρήστη μέσα σε εικονική μηχανή (απαιτούσαν προφανώς να είναι εγκατεστημένη και κάποια έκδοση της Java). Η Microsoft δημιούργησε τη δική της «κλειστή» υλοποίηση, γνωστή ως Silverlight, ενώ η Google ξεκίνησε την εξέλιξη της δικής της εκδοχής, γνωστής ως O3D⁹.

3.1.7.2 WebGL

Παράλληλα με την προτυποποίηση της HTML5, το WebGL έγινε ευρύτερα γνωστό και υποστηριζόμενο, βοηθώντας στον περιορισμό της πληθώρας των επεκτάσεων που χρησιμοποιούνταν από τους προγραμματιστές μέχρι τότε και των ασυμβατοτήτων που η χρήση τους εισήγε. Γενικά, το WebGL επιτρέπει στους προγραμματιστές εισάγουν διαδραστικά τρισδιάστατα γραφικά στο browser σε πραγματικό χρόνο. Μπορεί να εφαρμοστεί σε διαδραστικά μουσικά βίντεο, παιχνίδια, οπτικοποίηση δεδομένων, μορφές τέχνης, τρισδιάστατα περιβάλλοντα σχεδίασης, τρισδιάστατη μοντελοποίηση αντικειμένων, σχεδιασμό μαθηματικών συναρτήσεων, δημιουργία φυσικών προσομοιώσεων, κ.ά. Ήδη χρησιμοποιείται μεταξύ άλλων σε εφαρμογές όπως το Google Maps, το Autocad Cloud της Autodesk και το Citadel demo της Epic Games. Η Google επίσης κυκλοφόρησε στον ιστότοπο <http://stars.chromeexperiments.com> ένα επιστημονικό project που αφορά μία προσομοίωση γαλαξιακής οπτικοποίησης, το οποίο ονομάζουν "100,000 stars". Πρόκειται για μια εφαρμογή ιστού αναπτυγμένη από το τμήμα έρευνας της εταιρίας – Google Labs, που επιτρέπει στο χρήστη την εξερεύνηση της αστρικής «γειτονιάς» του πλανήτη Γη και αποτελεί ένα πρώτης τάξης παράδειγμα των δυνατοτήτων που φέρνει το νέο αυτό API (*Εικόνα 17*).

⁹ Ενώ αρχικά η υλοποίησή της βασιζόταν σε μια plug-in based αρχιτεκτονική και χρησιμοποιούσε τη C, το Μάη του 2010, η Google ανακοίνωσε πως το όλο project αλλάζει προσανατολισμό και από plug-in θα γίνει port σε βιβλιοθήκη της JavaScript που θα τρέχει πάνω από WebGL [57].



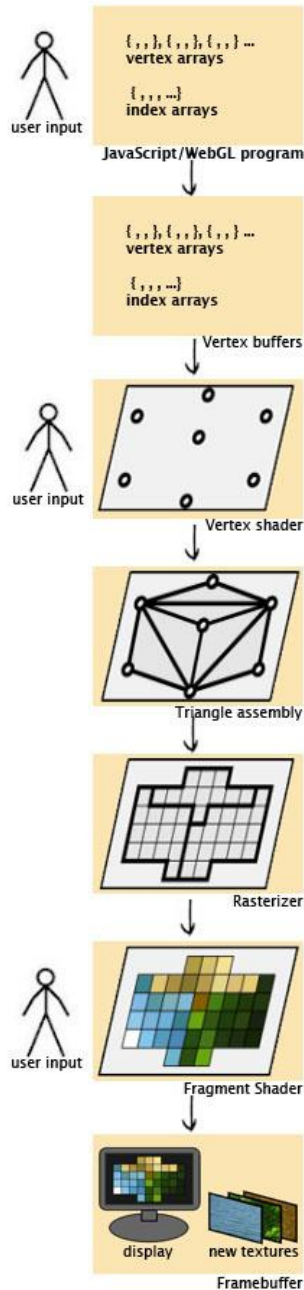
Εικόνα 17. Screenshot της εφαρμογής "100,000 stars" από τη Google

Το WebGL ελέγχεται αποκλειστικά μέσω JavaScript και «τρέχει» σε ένα συγκεκριμένο πλαίσιο (context) του HTML `<canvas>` στοιχείου, που δίνει πρόσβαση σε τρισδιάστατη επιτάχυνση υλικού. Προτυποποιήθηκε από το Khronos group [37], ενεργά μέλη του οποίου μεταξύ άλλων είναι όλοι οι μεγάλοι παίκτες στην αγορά των browsers, εξαιρουμένης της Microsoft. Είναι βασισμένο πάνω στο OpenGL ES 2.0, την έκδοση του ευρέως χρησιμοποιούμενου προτύπου για παραγωγή τρισδιάστατων γραφικών σε εφαρμογές προσωπικών υπολογιστών OpenGL, προσανατολισμένου σε ενσωματωμένα συστήματα [38]. Η επιλογή αυτή έγινε για λόγους συνοχής και συμβατότητας μεταξύ διαφορετικών πλατφόρμων και συσκευών – ως API θα πρέπει να είναι προσβάσιμο από browsers εγκατεστημένους τόσο σε φορητές συσκευές, όπως smartphones και tablets, όσο και προσωπικούς υπολογιστές ανεξαρτήτως λειτουργικού συστήματος [39].

Γεγονός είναι πως το WebGL είναι αρκετά πιο πολύπλοκο από τις υπόλοιπες τεχνολογίες των browsers επειδή σχεδιάστηκε για την απευθείας «συνεργασία» με την κάρτα γραφικών (GPU). Ως αποτέλεσμα, είναι ένα χαμηλού επιπέδου (low-level) API, η χρήση του οποίου όμως, ως αντιστάθμισμα, επιτυγχάνει υψηλές επιδόσεις κατά τον τρισδιάστατο σχεδιασμό (rendering).

Ο προγραμματισμός με το WebGL στοχεύει συνήθως στην απεικόνιση κάποιου είδους σκηνής που απαρτίζεται από μοντέλα. Αυτή εμπεριέχει πολλαπλές διαδοχικές κλήσεις σχεδίασης (draw calls), κάθε μία από τις οποίες εκτελείται στην κάρτα γραφικών μέσω μιας διαδικασίας γνωστής ως διασωλήνωση απεικόνισης (rendering pipeline, βλ. *Εικόνα 18*). Δομικό στοιχείο όλων των μοντέλων που σχεδιάζονται είναι το τρίγωνο, οπότε η μέθοδος σχεδιασμού περιέχει τη δημιουργία πληροφορίας για το πού αυτά τα τρίγωνα θα σχεδιαστούν και το πώς θα φαίνονται (χρώμα, υφή, σκιά, κτλ.). Η πληροφορία αυτή τότε παρέχεται στη GPU, που την επεξεργάζεται και έπειτα επιστρέφει μια όψη της σκηνής.

Η διαδικασία ξεκινά με τη δημιουργία πινάκων κορυφών (vertex arrays), που περιέχουν γνωρίσματα που περιγράφουν κορυφές όπως η τοποθεσία τους, το χρώμα, η υφή, η επίδραση του φωτός (vertex normal) πάνω τους, κ.ά. Οι πίνακες αυτοί κατασκευάζονται σε JavaScript και, λόγω αυξημένης πολυπλοκότητας, συνήθως χρησιμοποιείται κάποια βιβλιοθήκη που παρέχει έτοιμους για διάφορα γεωμετρικά σχήματα.



Εικόνα 18. Η διασωλήνωση απεικόνισης (rendering pipeline) του WebGL

Τα δεδομένα τότε των vertex arrays αντιγράφονται σε ειδικούς για το σκοπό αυτό vertex buffers της GPU. Όταν κατατίθεται μια εργασία σχεδίασης, στη GPU πρέπει επίσης να παρασχεθεί ένας επιπλέον πίνακας (index array) τα στοιχεία του οποίου δεικτοδοτούν τα στοιχεία του πίνακα κορυφών. Χρησιμοποιείται για το μετέπειτα έλεγχο της συγκέντρωσης των κορυφών ώστε να σχηματίσουν τρίγωνα.

Η GPU ξεκινά διαβάζοντας κάθε κορυφή από το vertex buffer και περνώντας την σε ένα vertex shader. Vertex shader ονομάζεται ένα πρόγραμμα – γραμμένο σε GLSL για να εκτελείται παράλληλα από τις σύγχρονες GPUs [40] – το οποίο δέχεται κάθε φορά σαν όρισμα τα γνωρίσματα μιας κορυφής που πρόκειται να επεξεργαστεί. Κατ’

ελάχιστο, ο vertex shader υπολογίζει την προβολή της κορυφής στο επίπεδο της οθόνης αλλά μπορεί να παράγει και άλλες πληροφορίες όπως χρώμα ή συντεταγμένες υφής (texture coordinates) για κάθε κορυφή.

Η GPU ύστερα «συνδέει» τις προβολές των κορυφών που έχουν προκύψει από το vertex shader για να σχηματίσει τρίγωνα. Το καταφέρνει αυτό παίρνοντας τις κορυφές με τη σειρά που υποδεικνύει ο index array και ομαδοποιώντας τις σε σύνολα των τριών έκαστο.

Αμέσως μετά ο rasterizer παίρνει κάθε τρίγωνο, το ψαλιδίζει, απορρίπτει τα μέρη που θα μείνουν εκτός οθόνης και «σπάει» τα υπόλοιπα ορατά σε τεμάχια μεγέθους pixel (pixel-sized fragments). Στην υπόλοιπη – χρωματική κυρίως – πληροφορία που προκύπτει από την επεξεργασία του vertex shader εφαρμόζεται παρεμβολή (interpolation) καθ' όλη την έκταση της επιφάνειας του κάθε τριγώνου, αναθέτοντας μία ομαλή χρωματική διαβάθμιση σε κάθε fragment. Για παράδειγμα, αν ο vertex shader εκχωρήσει μία χρωματική τιμή σε κάθε κορυφή, ο rasterizer θα πραγματοποιήσει κατάλληλη μίξη των χρωμάτων των γειτονικών κορυφών, ώστε η επιφάνεια να έχει όσο γίνεται περισσότερο απαλές χρωματικές διακυμάνσεις.

Τα παραγόμενα pixel-sized fragments τροφοδοτούνται κατόπιν σε ένα ακόμα πρόγραμμα που ονομάζεται fragment shader. Ο fragment shader γράφεται και αυτός από το προγραμματιστή σε GLSL, εκτελείται παράλληλα από τις GPUs και εξάγει τιμές χρώματος και βάθους για κάθε pixel, οι οποίες γράφονται στη συνέχεια στο framebuffer. Συνήθεις λειτουργίες που εκτελεί είναι η αντιστοίχιση υφής (texture mapping) και προσομοίωση φωτισμού. Αφού ο shader εκτελείται ανεξάρτητα για κάθε pixel του Canvas, μπορεί να χρησιμοποιηθεί για να παράγει τα πιο πολύπλοκα και περίτεχνα οπτικά εφέ, ωστόσο πρέπει πάντα να λαμβάνεται υπόψιν πως είναι και το ευαίσθητο κομμάτι της διασωλήνωσης από άποψη επίδοσης.

Ο framebuffer είναι ο τελικός προορισμός της διαδικασίας σχεδίασης που εκτελεί η GPU. Ο framebuffer δεν αποτελεί απλώς μια απλή δισδιάστατη εικόνα: εκτός από έναν ή περισσότερους επιμέρους χρωματικούς buffers μπορεί επιπλέον να έχει και κάποιον depth buffer ή/και stencil buffer. Και οι δύο τελευταίοι αυτοί buffers χρησιμοποιούνται για να «φιλτράρουν» τα fragments πριν αυτά αποδοθούν στο framebuffer. Ο έλεγχος βάθους απορρίπτει fragments αντικειμένων που βρίσκονται πίσω από ήδη σχεδιασμένα και ο έλεγχος με στένσιλ χρησιμοποιεί σχήματα «ζωγραφισμένα» στο stencil buffer για να περιορίσει το μέρος της εικόνας που τελικά θα αποδοθεί. Τα fragments εκείνα που δεν απορρίφθηκαν από το φιλτράρισμα του τελευταίου αυτού σταδίου αποδίδονται τελικώς στην οθόνη του χρήστη.

Η εξοικείωση με το WebGL API, όπως γίνεται κατανοητό, είναι χρονοβόρα και η απευθείας χρήση του μια επίπονη διαδικασία, ιδιαίτερα για προγραμματιστές χωρίς μεγάλη εμπειρία από τρισδιάστατη απεικόνιση. Για το σκοπό αυτό, έχουν αναπτυχθεί πολλές αξιόλογες βιβλιοθήκες που προσθέτουν μια αφαιρετική στρώση πάνω από τις χαμηλού επιπέδου κλήσεις του WebGL, επιταχύνοντας και απλοποιώντας την ανάπτυξη εφαρμογών. Η πιο διαδεδομένη από αυτές, η οποία επιλέχθηκε και για τη συγκεκριμένη υλοποίηση ονομάζεται Three.js [41]. Αποτελεί μια ευέλικτη και «ελαφριά» τρισδιάστατη μηχανή με μικρή πολυπλοκότητα, που μπορεί να χρησιμοποιήσει για την απεικόνιση όλες τις τεχνολογίες που προαναφέρθηκαν, δηλαδή <canvas>, <svg> και WebGL. Επίσης παρέχει μια σειρά από έτοιμα μοντέλα, εφέ και υλοποιήσεις vertex και fragment shaders που μειώνουν δραστικά την ποσότητα του κώδικα που απαιτείται από τον προγραμματιστή, ενώ η μεγάλη κοινότητα των χρηστών του αποτελεί σημαντική πηγή γνώσεων και συμβουλών.

3.1.8 Ταυτοχρονισμός στη JavaScript

Γεγονός είναι πως υπάρχει μια σειρά από εμπόδια που αποτρέπουν τη μεταφορά όλων των εφαρμογών από τον εξυπηρετητή στο browser του πελάτη. Μερικά από αυτά είναι η ασυμβατότητα κάποιων χαρακτηριστικών μεταξύ των διαφορετικών browsers, η έλλειψη στατικού συστήματος τύπων (η JavaScript αποτελεί παράδειγμα κατεξοχήν δυναμικής γλώσσας), η προσβασιμότητα και η επίδοση. Ευτυχώς, το τελευταίο σταδιακά παύει να αποτελεί πρόβλημα καθώς, όπως ήδη παρουσιάστηκε, οι μηχανές της JavaScript γίνονται όλο και πιο γρήγορες και αποτελεσματικές στην εκτέλεση του κώδικα.

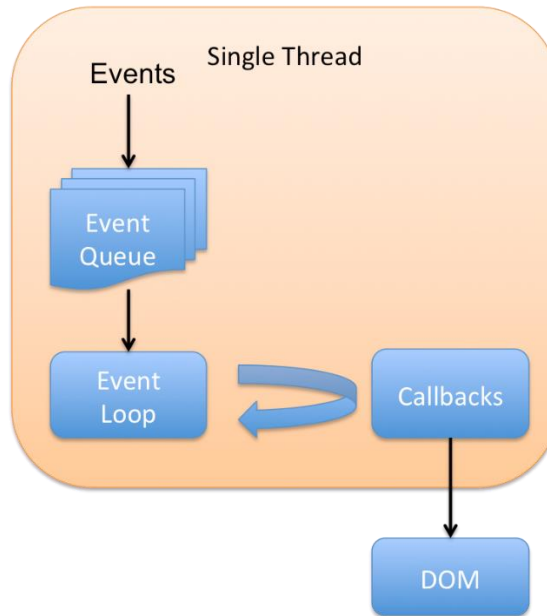
Ένα στοιχείο που παρ' όλα αυτά δρα αρνητικά ως προς τη δυναμική της έχει να κάνει με την ίδια τη φύση της γλώσσας. Το περιβάλλον εκτέλεσης της JavaScript είναι μονονηματικό (single-threaded), κάτι που σημαίνει πως πολλαπλά scripts δεν μπορούν να εκτελούνται ταυτόχρονα. Ως παράδειγμα, μπορεί να αναλογιστεί κάποιος μια HTML σελίδα που χρειάζεται να διαχειριστεί συμβάντα που προκύπτουν από την αλληλεπίδραση με το χρήστη (UI events), να αποκτήσει και να επεξεργαστεί δυνητικά μεγάλο όγκο δεδομένων από κάποιο backend API και να πραγματοποιήσει αλλαγές στο DOM – ενέργειες αρκετά συνηθισμένες για οποιαδήποτε εφαρμογή ιστού. Δυστυχώς, όλα αυτά παραδοσιακά δεν ήταν δυνατό να περατωθούν παράλληλα από τους browsers. Το εύλογο ερώτημα που προκύπτει λοιπόν έχει να κάνει με το πώς παρ' όλα αυτά οι καλογραμμένες εφαρμογές κατάφεραν να αποκρύπτουν όλη αυτήν την πολυπλοκότητα από τη χρήστη και συνάμα να παραμένουν αποκρίσιμες.

3.1.8.1 Ο βρόχος συμβάντων (event loop) της JavaScript

Στη JavaScript, σχεδόν όλες οι λειτουργίες εισόδου/εξόδου (I/O) εκτελούνται ασύγχρονα, χωρίς μπλοκάρισμα (non-blocking execution). Σε αυτές περιλαμβάνονται HTTP αιτήσεις, ενέργειες πάνω σε βάσεις δεδομένων, εγγραφές και αναγνώσεις από το σκληρό δίσκο. Το μοναδικό νήμα ζητά από το περιβάλλον εκτέλεσης να πραγματοποιήσει μια ενέργεια, παρέχοντάς του μια συνάρτηση επανάκλησης (callback function) και συνεχίζει με την εκτέλεση άλλων εργασιών. Όταν η ενέργεια που ζητήθηκε προηγουμένως ολοκληρωθεί, ένα μήνυμα εισάγεται σε μια ουρά μαζί με ένα παρεχόμενο callback. Κάποια στιγμή στο μέλλον, το μήνυμα αυτό θα αφαιρεθεί από την ουρά και η συνάρτηση επανάκλησης θα εκτελεστεί. Αυτό το διαδραστικό πλήρως ασύγχρονο μοντέλο μπορεί να είναι γνώριμο στους προγραμματιστές που αναπτύσσουν λογισμικό διεπαφής χρήστη – εκεί που συμβάντα όπως το πάτημα ενός κουμπιού ή η κύλιση του παραθύρου μπορούν να προκύψουν οποιαδήποτε στιγμή και πρέπει να «εξυπηρετηθούν», αλλά διαφέρει σημαντικά από το σύγχρονο μοντέλο αίτησης-απόκρισης που συναντάται σε τυπικές υλοποιήσεις εφαρμογών εξυπηρετητή.

Αυτή η απεμπλοκή του καλούντος από την απάντηση που αυτός αναμένει, επιτρέπει στο περιβάλλον εκτέλεσης της JavaScript να ασχοληθεί με άλλες διεργασίες ενώ «περιμένει» την ασύγχρονη ενέργεια να διεκπεραιωθεί και να έρθει η στιγμή να εκκινήσει το callback. Αυτή η ουρά – αόρατη στον προγραμματιστή – στην οποία τα μηνύματα¹⁰ αποθηκεύονται προσωρινά μαζί με τα αντίστοιχα εγγεγραμμένα callbacks, ονομάζεται βρόχος συμβάντων (event loop, βλ. επίσης *Εικόνα 19*).

¹⁰ Τα μηνύματα αυτά εισάγονται στην ουρά ως απόκριση σε εξωτερικά συμβάντα που ο προγραμματιστής επιθυμεί να διαχειριστεί ιδιαίτερος μέσω των παρεχόμενων callbacks, όπως το πάτημα κάποιου κουμπιού του ποντικιού, του πληκτρολογίου ή της HTML σελίδας, ή την απάντηση από κάποιον εξυπηρετητή. Αν όμως, για παράδειγμα, ο χρήστης πατήσει κάποιο πλήκτρο



Εικόνα 19. Βρόχος συμβάντων της JavaScript

3.1.8.2 Run-to-completion

Η πρακτική που ακολουθεί το περιβάλλον της JavaScript είναι η πλήρης επεξεργασία κάθε μηνύματος προτού συνεχίσει με το επόμενο. Αυτό προσφέρει κάποιες ελκυστικές ιδιότητες κατά το σχεδιασμό της λογικής των προγραμμάτων, συμπεριλαμβανομένης της εγγύησης της εγκυρότητας των δεδομένων κατά τη διάρκεια της εκτέλεσης συναρτήσεων. Παρατηρείται δηλαδή διαφορά με το μοντέλο της C, κατά το οποίο, όταν μία συνάρτηση ή κομμάτι κώδικα εκτελείται μέσα σε κάποιο νήμα, το σύστημα έχει τη δυνατότητα να διακόψει την εκτέλεσή τους και να μεταφέρει τον έλεγχο σε κάποιο άλλο νήμα.

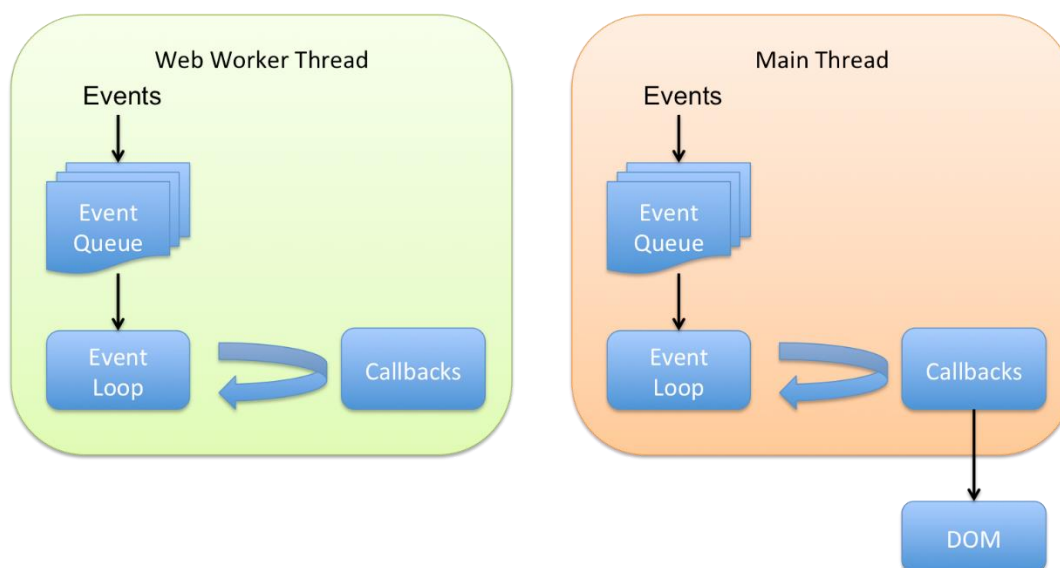
Βέβαια, υπάρχουν και μειονεκτήματα στην προσέγγιση αυτή. Το σπουδαιότερο εκ των οποίων έχει να κάνει με το εάν ένα μήνυμα χρειαστεί σημαντικό χρονικό διάστημα για την επεξεργασία του, όλη η εφαρμογή καθίσταται ανίκανη να διαχειριστεί οποιαδήποτε αλληλεπίδραση με το χρήστη. Δεν είναι σπάνια, ακόμα και σήμερα, η εμφάνιση ειδοποίησης με τη μορφή ξεχωριστού παραθύρου από το browser, σύμφωνα με την οποία «η εκτέλεση κάποιου script καθιστά την εφαρμογή μη αποκρίσιμη». Οι προγραμματιστές για να υπερκεράσουν τους περιορισμούς αυτούς προσπαθούσαν να μειώσουν όσο γίνεται το χρόνο επεξεργασίας που απαιτούσαν ή, αν αυτό δεν ήταν δυνατό, να «μοιράσουν» το φόρτο σε μικρότερα μηνύματα, τα οποία τοποθετούσαν εκ νέου στο βρόχο συμβάντων χρησιμοποιώντας ειδικά API calls, όπως `setTimeout()` [42] και `setInterval()` [43].

3.1.8.3 Web Workers

Μεγάλη πρόοδο επιτεύχθηκε στο συγκεκριμένο τομέα με την έλευση των Web Workers. Σύμφωνα με τις επίσημες προδιαγραφές, οι browsers πλέον καλούνται να υλοποιήσουν ένα API για την παρασκηνιακή εκτέλεση scripts από τις εφαρμογές ιστού. Με τον τρόπο αυτό, υπολογιστικά απαιτητικές εργασίες μπορούν να μεταφερθούν σε

και δεν έχει καταχωρηθεί κάποια συνάρτηση-callback για το χειρισμό της ενέργειας αυτής, στην ουρά δε θα εισαχθεί κάποιο μήνυμα.

δικό τους ανεξάρτητο νήμα εκτέλεσης με το δικό του ανεξάρτητο βρόχο συμβάντων, χωρίς να υπάρχει το ενδεχόμενο να «μπλοκάρουν» τη διεπαφή της εκάστοτε εφαρμογής με το χρήστη, η οποία «τρέχει» στο κύριο νήμα (Εικόνα 20).



Εικόνα 20. Web worker και κύριο νήμα εκτέλεσης

Η επικοινωνία μεταξύ του κυρίου νήματος (main or UI thread) και των Web workers λαμβάνει χώρα μέσω ενός μοντέλου συμβάντων (event model) και της μεθόδου `postMessage()`. Ανάλογα με τον τύπο και την έκδοση του browser καθώς και τον τρόπο κλήσης της μεθόδου, η τελευταία μπορεί να λειτουργήσει με δύο τρόπους. Οι περισσότεροι browsers υλοποιούν το λεγόμενο structured cloning αλγόριθμο, ο οποίος επιτρέπει το πέρασμα πολύπλοκων τύπων δεδομένων από και προς τους workers, όπως `File`, `Blob`, `ArrayBuffer` και JSON αντικείμενα. Ωστόσο, τα αντικείμενα αυτά αντιγράφονται πριν το πέρασμά τους (κάτι αντίστοιχο με το πέρασμα κατά τιμή στη C), γεγονός που εισάγει καθυστέρηση στην όλη διαδικασία. Για το λόγο αυτό, αναπτύχθηκε μια δεύτερη τεχνική που αξιοποιεί τα λεγόμενα transferrable objects. Χρησιμοποιώντας την, δεδομένα μεταφέρονται από το περιβάλλον του κυρίου νήματος σε αυτό του παρασκηνακού χωρίς να αντιγράφεται τίποτα με σημαντικό αντίκτυπο στην επίδοση. Ομοιάζει με το πέρασμα κατ' αναφορά της C, με μια μικρή αλλά σημαντική διαφορά: όταν ένα αντικείμενο επιχειρηθεί να «περαστεί» από το κύριο νήμα σε κάποιο worker, το κύριο χάνει την αναφορά που κρατούσε στο αντικείμενο αυτό, και το ανάποδο. Αυτό συμβαίνει λόγω της σχεδιαστικής επιλογής των δημιουργών του προτύπου να διαχωρίσουν πλήρως το χώρο μνήμης των διαφόρων νημάτων μιας εφαρμογής έτσι ώστε να εξασφαλίσουν την αποφυγή φαινομένων, όπως data races και deadlocks.

Να σημειωθεί ακόμα πως από το περιβάλλον ενός Web Worker είναι αδύνατη η χρησιμοποίηση όλων των χαρακτηριστικών και APIs της JavaScript. Ο κύριος περιορισμός έχει να κάνει με την αδυναμία αλληλεπίδρασης με το DOM, μιας και το JavaScript αντικείμενο που είναι υπεύθυνο για το ρόλο αυτό (document object) δε μοιράζεται με τα νήματα που τρέχουν στο παρασκήνιο. Για τον ίδιο λόγο δεν υπάρχει πρόσβαση και σε τεχνολογίες που εξετάστηκαν παραπάνω (WebGL και Canvas). Παρ' όλα αυτά, οι workers είναι σε θέση να επικοινωνήσουν με το δίκτυο, μέσω του XMLHttpRequest API [44].

Αν αναλογιστεί κανείς την πρόοδο που έχει γίνει σήμερα στο κομμάτι του υλισμικού και ειδικά των πολυπύρηνων αρχιτεκτονικών πάνω στις οποίες είναι βασισμένη η πλειονότητα των συσκευών στις οποίες έχουν πρόσβαση οι χρήστες, ένα πρότυπο σαν και αυτό αποτελούσε το επόμενο εύλογο βήμα. Το αντίκτυπο λοιπόν που είχε η εισαγωγή των Web Workers στην ανάπτυξη εφαρμογών ιστού, αντικατοπτρίζεται στην πληθώρα σεναρίων χρήσης που ήδη καλύπτουν:

- Δημιουργία, γραμματική ανάλυση και επεξεργασία αρχείων PDF.
- Συμπίεση/αποσυμπίεση εικόνων σε προσαρμοσμένη μορφή αρχείου (JPEG, BMP, PNG).
- Υπολογισμός A* μονοπατιού σε περιβάλλον διαδραστικών παιχνιδιών.
- Αναζήτηση λέξης σε βάση δεδομένων για ορθογραφικό έλεγχο σε επεξεργαστή κειμένου.
- Φόρτωση και γραμματική ανάλυση πόρων μέσα σε παιχνίδι.
- Πραγματοποίηση πολύπλοκων υπολογισμών στο παρασκήνιο και απομακρυσμένη καταγραφή των αποτελεσμάτων.

Η κοινότητα όμως των κατασκευαστών browsers δε μένει εκεί αλλά προσπαθεί συνεχώς να εκμεταλλευτεί την αυξημένη δυνατότητα παραλληλίας που τα μηχανήματα σήμερα προσφέρουν με σκοπό πάντα την παροχή συνεχώς βελτιωμένης εμπειρίας χρήσης. Προς τη κατεύθυνση αυτή, λοιπόν, έχουν προτυποποιηθεί τελευταία και δύο καινούρια παρόμοια APIs: Shared Workers [45] και Service Workers [46]. Το πρώτο από αυτά αφορά την πρόσβαση σε νήματα παρασκήνιου από διαφορετικά παράθυρα του browser, `<iframe>` tags ή και άλλα ακόμα νήματα, ενώ το δεύτερο εισάγει νέου τύπου workers με σκοπό την παροχή πλούσιας εμπειρίας χρήσης ακόμα και σε κατάσταση εκτός σύνδεσης, δυνατότητας περιοδικού συγχρονισμού με το backend στο παρασκήνιο, την υποστήριξη push notifications από τον εξυπηρετητή, και άλλων προηγμένων χαρακτηριστικών που συνεχώς εμπλουτίζονται.

3.1.9 WebRTC

3.1.9.1 Σύντομη ιστορική προσέγγιση

Μία από τις τελευταίες μεγάλες προκλήσεις για το διαδίκτυο είναι η παροχή της δυνατότητας για ανθρώπινη επικοινωνία μέσω ήχου και βίντεο, κάτι που επικράτησε να λέγεται επικοινωνία πραγματικού χρόνου (Real-Time Communication - RTC). Οι οραματιστές της κοινότητας θεωρούσαν ανέκαθεν πως οι RTC δυνατότητες θα πρέπει να είναι εξίσου φυσικές για τις εφαρμογές ιστού, όπως είναι και η εισαγωγή κειμένου σε φόρμες. Χωρίς αυτές αισθάνονταν περιορισμό ως προς την ικανότητά τους να καινοτομούν και να εξελίσσουν νέους τρόπους για την αλληλεπίδραση των χρηστών.

Εντούτοις, ιστορικά η επικοινωνία πραγματικού χρόνου επιτυγχανόταν μέσα σε επιχειρησιακά περιβάλλοντα, καθώς, λόγω αυξημένης πολυπλοκότητας, απαιτούσε την απόκτηση ακριβών και αδειοδοτημένων τεχνολογιών βίντεο και ήχου ή την εξέλιξή τους από την ίδια την επιχείρηση. Επιπρόσθετα, η ενοποίηση των RTC τεχνολογιών με το ήδη υπάρχον περιεχόμενο, δεδομένα και υπηρεσίες υπήρξε δύσκολο και χρονοβόρο έργο, ειδικά στον ιστό.

Το Gmail video chat έγινε δημοφιλές το 2008 και το 2011 η Google εισήγαγε την εφαρμογή Hangouts, η οποία χρησιμοποιούσε την ίδια υπηρεσία με το Gmail, γνωστή ως Google Talk. Στη συνέχεια, η ίδια εταιρία αγόρασε τη GIPS. Η τελευταία είχε εξελίξει

μια σειρά από συστατικά (component) απαραίτητα για RTC, όπως κωδικοποιητές και τεχνικές ακύρωσης ηχούς (echo cancellation techniques), τα οποία κατόπιν κατέστησε «ανοικτά» στο κοινό (open source). Συνεργάστηκε επίσης με τις αρμόδιες επιτροπές IETF [47] και W3C [32] για να επιτύχει την αποδοχή της βιομηχανίας. Το Μάη του 2011, η Ericsson κατασκεύασε την πρώτη υλοποίηση του WebRTC (Web Real-Time Communication).

Το WebRTC περιέχει σήμερα ανοικτά πρότυπα για πραγματικού χρόνου επικοινωνία μέσω βίντεο, ήχου και δεδομένων στο browser, χωρίς την ανάγκη επεκτάσεων. Η ανάγκη για ένα τέτοιο σύνολο προτύπων είναι πράγματι υπαρκτή:

- Πολλές υπηρεσίες ιστού (web services) ήδη χρησιμοποιούν RTC τεχνικές, αλλά απαιτούν τη λήψη και εγκατάσταση native εφαρμογών ή επεκτάσεων (αν αναφερόμαστε στο browser). Παραδείγματα τέτοιων δημοφιλών εφαρμογών αποτελούν το Skype, το Facebook (το οποίο χρησιμοποιούσε την υποδομή του Skype μέχρι πρότινος που αποφάσισε να υιοθετήσει το WebRTC) και το Google Hangouts (που χρησιμοποιεί τη Google Talk επέκταση).
- Η λήψη, εγκατάσταση και ενημέρωση των native εφαρμογών και των επεκτάσεων μπορεί να είναι περίπλοκη, επιρρεπής σε σφάλματα και ενοχλητική ακόμα για τον απλό χρήστη.
- Ειδικά οι επεκτάσεις μπορεί να είναι απαιτητικές για την ανάπτυξη, την αποσφαλμάτωση, την επιδιόρθωση, τον έλεγχο και τη συντήρησή τους – μπορεί ακόμα να απαιτείται η αδειοδότηση και η ενοποίηση με πολύπλοκη και ακριβή τεχνολογία. Επίσης, συχνά είναι δύσκολο ακόμα και να πειστεί αρχικά ο χρήστης να τις εγκαταστήσει και να τις χρησιμοποιήσει.

Σύμφωνα με τις κατευθυντήριες αρχές που συνοδεύουν το WebRTC project, τα APIs του πρέπει να είναι ανοικτού κώδικα, ελεύθερα, προτυποποιημένα, ενσωματωμένα στους browsers και σαφώς πιο αποδοτικά από τις ήδη υπάρχουσες τεχνολογίες.

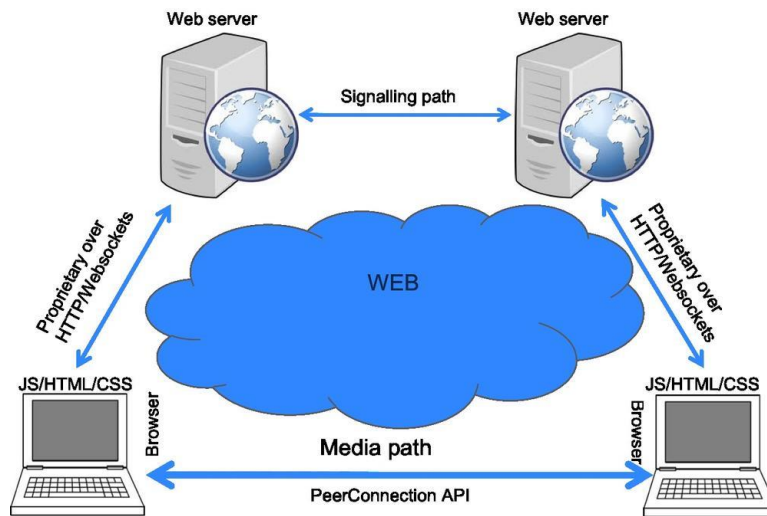
3.1.9.2 Το WebRTC ως τεχνολογία

Με τον όρο WebRTC καλείται ένα σύνολο από προδιαγραφές, πρωτόκολλα και JavaScript APIs, ο συνδυασμός των οποίων επιτρέπει τον peer-to-peer διαμοιρασμό ήχου, βίντεο και δεδομένων μεταξύ browsers (peers). Αντί να βασίζεται σε εξωτερικές επεκτάσεις (third-party plug-ins) ή κλειστό (proprietary) λογισμικό, το WebRTC μετατρέπει την επικοινωνία πραγματικού χρόνου σε ένα καθιερωμένο χαρακτηριστικό στο οποίο μπορεί να έχει πρόσβαση οποιαδήποτε εφαρμογή ιστού με τη μορφή τριών ακόμα απλών και διαισθητικών JavaScript APIs.

Το πρώτο από αυτά ονομάζεται MediaStream (ευρύτερα γνωστό και ως getUserMedia) και αναπαριστά συγχρονισμένες ροές πολυμέσων που μπορούν να μεταδοθούν μεταξύ των peers. Μια τέτοια ροή (stream) μπορεί να αποτελείται από ένα ή περισσότερα κομμάτια, αν και συνήθως αποτελείται από ένα για το βίντεο και ένα για τον ήχο. Μια ροή μπορεί να μεταδώσει είτε «ζωντανά» πολυμέσα (για ηχητικές κλήσεις ή βιντεοδιασκέψεις) ή αποθηκευμένα (π.χ. ταινίες ή μουσική που βρίσκονται στον τοπικό σκληρό δίσκο). Το δεύτερο, και πιο σημαντικό ίσως, είναι το RTCPeerConnection, υπεύθυνο για τη σταθερή και αποδοτική μετάδοση των ροών δεδομένων μεταξύ των peers. Χάρη σε αυτό, οι προγραμματιστές κατά την ανάπτυξη των εφαρμογών δε χρειάζεται να ασχοληθούν με χαμηλού επιπέδου ζητήματα, όπως διαχείριση χαμένων

πακέτων και διαθέσιμου bandwidth, ακύρωση ηχούς, «καθαρισμό» ήχου και βίντεο, κ.ά. Τέλος, εκτός από ήχο και βίντεο, το WebRTC, όπως προαναφέρθηκε, υποστηρίζει επίσης μετάδοση και άλλων τύπων δεδομένων σε πραγματικό χρόνο. Το σκοπό αυτό εξυπηρετεί το RTCDataChannel API. Επιτρέπει την αποστολή τυχαίων δεδομένων (arbitrary data) με χαμηλή καθυστέρηση (low latency) και υψηλή διεκπεραιωτικότητα. Μπορεί να χρησιμοποιηθεί μεταξύ άλλων για ανάπτυξη παιχνιδιών, εφαρμογών απομακρυσμένης επιφάνειας εργασίας, δωμάτια συνομιλιών, μεταφορά αρχείων μέχρι και επικοινωνία αποκεντρωποιημένων δικτύων.

Το WebRTC αξιοποιεί το RTCPeerConnection API για την ανταλλαγή ροών δεδομένων μεταξύ browsers/peers, αλλά όπως είναι εύλογο, απαιτεί ένα μηχανισμό για να συντονίσει την εγκαθίδρυση των απομακρυσμένων συνδέσεων μέσω της αποστολής μηνυμάτων ελέγχου – μια διαδικασία γνωστή ως σηματοδότηση (signaling). Συγκεκριμένες μέθοδοι και πρωτόκολλα δεν παρέχονται για το σκοπό αυτό από το WebRTC. Οι κατασκευαστές λογισμικού αφήνονται ελεύθεροι να επιλέξουν από μια ευρεία γκάμα έτοιμων μηχανισμών και πρωτοκόλλων ανταλλαγής μηνυμάτων (SIP [48], XMPP [49], WebSockets [50]) ανάλογα με τις προσωπικές τους προτιμήσεις. Η ανταλλαγή των απαραίτητων πληροφοριών μέσω σηματοδότησης πρέπει να έχει ολοκληρωθεί επιτυχώς για να καταστεί δυνατή η μετέπειτα ανταλλαγή οποιασδήποτε μορφής δεδομένων μεταξύ των browsers/peers (Εικόνα 21).



Εικόνα 21. Κοινή τοπολογία βασισμένη σε WebRTC επικοινωνία

4 Προτεινόμενο σύστημα

4.1 Τεχνικές Προδιαγραφές

4.1.1 Λειτουργικές απαιτήσεις

Σκοπός της διπλωματικής εργασίας είναι ο σχεδιασμός και η υλοποίηση μιας δικτυακής συνεργατικής πλατφόρμας βασισμένη σε browser, μέσα από την οποία επαγγελματίες υγείας θα μπορούν να εκκινούν εικονικές συνεδρίες και να εργάζονται από κοινού πάνω σε ιατρικές εικόνες. Αυτό επιτυγχάνεται κάνοντας χρήση των δυνατοτήτων επιτάχυνσης γραφικών (Canvas, WebGL) και των προηγμένων τεχνικών διασύνδεσης (WebRTC) που υποστηρίζει η πλειοψηφία των σύγχρονων browsers. Γενικά, η συγκεκριμένη υλοποίηση στοχεύει στην παροχή λειτουργικότητας και εμπειρίας χρήσης που παλιότερα θα ήταν εφικτή μόνο μέσα από desktop εφαρμογές και proprietary τεχνολογίες (βλ. 3.1.4).

Καταρχήν, η client εφαρμογή πρέπει να μπορεί να λειτουργεί σαν ένας πλήρης λειτουργιών standalone DICOM viewer. Οι κυριότερες απαιτήσεις που πρέπει να εκπληρώνονται για το σκοπό αυτό είναι οι ακόλουθες:

- Φόρτωση και προβολή οποιουδήποτε αρχείου εικόνας DICOM από το σκληρό του δίσκο, εξωτερικό χώρο αποθήκευσης ή σύστημα PACS.
- «Έξυπνη» ομαδοποίηση πολλαπλών εικόνων ανάλογα με ασθενή, μελέτη/εξέταση, υποστήριξη thumbnails και δυνατότητας πλοήγησης μεταξύ αυτών με διαισθητικό τρόπο.
- Βασικά εργαλεία διαχείρισης της ιατρικής εικόνας, όπως μετακίνηση, μεγέθυνση/σμίκρυνση, αλλαγή προσανατολισμού, κτλ.
- Βασικά ιατρικά φίλτρα για επεξεργασία εικόνας.
- Εργαλεία μέτρησης σημείων ενδιαφέροντος πάνω στην εικόνα (μήκους, επιφάνειας, γωνίας, κτλ.).
- Εμφάνιση στοιχείων ασθενούς και πληροφοριών εξέτασης πάνω από την εικόνα για καλύτερη εποπτεία.
- Εμφάνιση όλων των στοιχείων/δεδομένων αποθηκευμένων στο αρχείο DICOM.

Όσον αφορά το κομμάτι της συνεργατικότητας, η εφαρμογή πρέπει να υποστηρίζει ένα εύρος από λειτουργίες διασύνδεσης χρηστών:

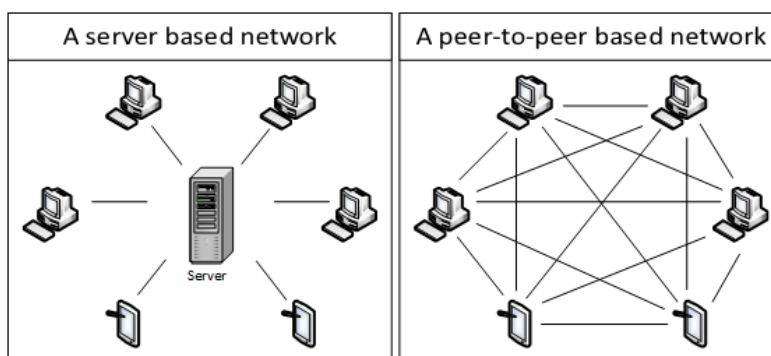
- Υποστήριξη επαφών με εμφάνιση των συνδεδεμένων χρηστών.
- Τηλεδιάσκεψεις με εικόνα, ήχο και γραπτά μηνύματα.
- Αυτόματος διαμοιρασμός ιατρικών εικόνων κατά τη διάρκεια της τηλεδιάσκεψης.
- Δυνατότητα πλήρους αλληλεπίδρασης με την ιατρική εικόνα όλων των συμμετεχόντων στην τηλεδιάσκεψη.
- Κοινή ροή εργασίας (workflow) – διάδοση των αλλαγών σε πραγματικό χρόνο.
- Ανταλλαγή αυθαίρετων δεδομένων (αρχεία).
- Υποστήριξη πολλαπλών συμμετεχόντων στις τηλεσυνεδρίες.

4.1.2 Σύντομη περιγραφή των χαρακτηριστικών της λύσης

Η ανταλλαγή ιατρικών δεδομένων μεταξύ εξυπηρετητών και πελατών είναι ιδιαίτερα απαιτητική ως προς τη χρησιμοποίηση του bandwidth, έτσι κεντροποιημένες αρχιτεκτονικές μπορεί να εισάγουν συμφόρηση (bottleneck) [20]. Επιπρόσθετα, όπως παρουσιάστηκε και στο προηγούμενο κεφάλαιο, η πλειοψηφία των συστημάτων που αναφέρονται στη σχετική βιβλιογραφία, απαιτούν την ύπαρξη ειδικού εξοπλισμού ή την εγκατάσταση ξεχωριστών εργαλείων/προγραμμάτων, ενώ δεν υποστηρίζουν τη συνεργατική δημιουργία ροών εργασίας πάνω σε ιατρικές εικόνες.

Σκοπός, λοιπόν, αυτής της διπλωματικής εργασίας είναι ο σχεδιασμός και η υλοποίηση μιας διαφανούς και εύρωστης SPA που χειρίζεται τη χρησιμοποίηση δικτυακών πόρων έξυπνα έτσι ώστε να παρέχει πλούσιες MDTM υπηρεσίες πραγματικού χρόνου. Ένα από τα κυριότερα καινοτόμα χαρακτηριστικά της προτεινόμενης πλατφόρμας είναι η εκμετάλλευση της νεο-εισαχθείσας WebRTC τεχνολογίας, σε συνδυασμό με τη χρήση σύγχρονων browser APIs. Η πλατφόρμα επίσης, εισάγει έναν έμμεσο μηχανισμό κλήσης απομακρυσμένης διαδικασίας (RPC – Remote Procedure Call) που επιτρέπει στο συμμετέχον στις συνεδρίες ιατρικό προσωπικό να συνεργαστεί σε πραγματικό χρόνο ακολουθώντας εξελιγμένες «οδούς» εργασίας μέσω μιας εργαλειοθήκης διαχείρισης ιατρικής εικόνας.

Η σχεδιαστική δόμηση της παρουσιαζόμενης πλατφόρμας πάνω στην peer-to-peer λογική δικτύωσης οδηγεί στην εξάλειψη εισαγόμενων από ενδιάμεσους κόμβους overheads και ζητημάτων δυνατότητας κλιμάκωσης (βλ. *Εικόνα 22*). Η ανάγκη για ύπαρξη κάποιου server βέβαια παραμένει – κυρίως για λόγους σηματοδοσίας (ώσπου να εγκαθιδρυθεί η σύνδεση μεταξύ των απομακρυσμένων κόμβων, όπως θα δούμε στη συνέχεια), αλλά το χρησιμοποιούμενο bandwidth είναι σημαντικό μικρότερο, κάτι που συνεπάγεται και μειωμένα κόστη.



Εικόνα 22. Συνήθεις τοπολογίες διασύνδεσης

Η ακολουθούμενη αρχιτεκτονική εκμεταλλεύεται επίσης πλήρως την επεξεργαστική ισχύ των σύγχρονων συσκευών. Σταθμοί εργασίας και προσωπικοί υπολογιστές μέχρι κινητά τηλέφωνα και tablets διαθέτουν πολλαπλούς πυρήνες επεξεργασίας, ισχυρά υποσυστήματα γραφικών, επαρκή ποσότητα μνήμης και δυνατότητα σύνδεσης σε συνεχώς αυξανόμενης ταχύτητας δίκτυα. Μπορεί οι τελευταίες τάσεις στο χώρο της πληροφορικής να δίνουν ιδιαίτερη βαρύτητα στη μεταφορά των υπολογισμών στο υπολογιστικό νέφος, εντούτοις οι σημερινές συσκευές είναι παραπάνω από ικανές για την αποδοτική εκτέλεση εργασιών που παλιότερα θα κρίνονταν απαγορευτικές. Με

αυτή την εκμετάλλευση του λεγόμενου client-side computing, βελτιώνεται και η εμπειρία χρήσης καθώς μειώνεται στο ελάχιστο δυνατό η επικοινωνία με το δίκτυο.

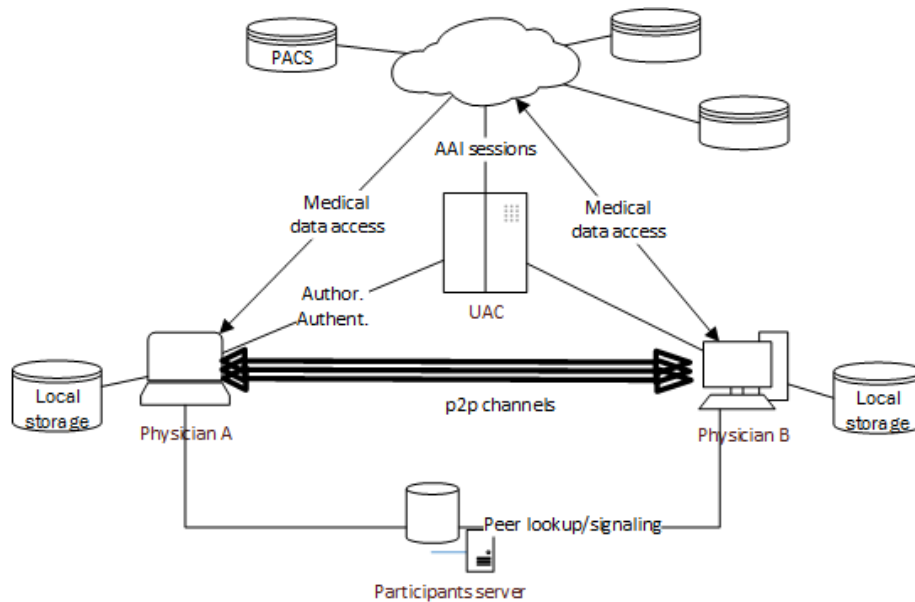
Τέλος, προκειμένου η εφαρμογή να αποκτήσει τη μεγαλύτερη δυνατή προσβασιμότητα από την ιατρική κοινότητα και να «συμμορφώνεται» με την αρχή BYOD, απαιτείται η υλοποίησή της να είναι διαφανής ως προς το υποκείμενο υλικό. Στη συγκεκριμένη περίπτωση, αυτό επιτυγχάνεται με τη χρησιμοποίηση της πιο ευρέως διαδεδομένης πλατφόρμας που οι περισσότεροι έχουν εγκατεστημένη στη συσκευή τους, ανεξαρτήτως τύπου, λειτουργικού συστήματος, κτλ. – το φυλλομετρητή ιστού (browser).

4.2 Αρχιτεκτονική πλατφόρμας

Σύμφωνα και με την *Εικόνα 23*, το σύστημα για τη λειτουργία του βασίζεται σε δύο ουσιώδη υποσυστήματα, το UAC (User Access Control) και το PAS (Participant Server). Το πρώτο είναι υπεύθυνο για την πιστοποίηση και εξουσιοδότηση των χρηστών, η οποία επιτυγχάνεται αλληλεπιδρώντας με μια υποδομή ταυτοποίησης και εξουσιοδότησης (Authentication and Authorization Infrastructure – AAI)¹¹. Το δεύτερο συγκρατεί όλη την πληροφορία σχετικά με τα προφίλ των χρηστών, την κατάστασή τους (π.χ. αν είναι διαθέσιμοι για συνεδρία) και, συγχρόνως, διαδραματίζει το ρόλο του εξυπηρετητή σηματοδότησης (signaling server), συντονίζοντας τα αρχικά στάδια της επικοινωνίας μεταξύ των peers/χρηστών. Οι τελευταίοι έχουν πρόσβαση στην εφαρμογή ιστού από οποιαδήποτε συσκευή διαθέτει κάποιο browser και μπορούν να τη χρησιμοποιήσουν ελεύθερα για την προβολή και επεξεργασία ιατρικών εικόνων που βρίσκεται σε κάποιο τοπικό αποθηκευτικό χώρο (CD/DVD, flash disk, σκληρό δίσκο), όπως επίσης και για την εκκίνηση κάποιας συνεδρίας – αξιοποιώντας τους απευθείας διαύλους επικοινωνίας που παρέχει το WebRTC. Ωστόσο, για απομακρυσμένη πρόσβαση σε κάποιο PACS απαιτείται πρώτα η εξακρίβωση της ταυτότητας του ενδιαφερομένου μέσω του UAC.

Λόγω της φύσης της εφαρμογής – που συνεπάγεται ανταλλαγή ευαίσθητων και προσωπικών ιατρικών δεδομένων, εκτός των πολιτικών ταυτοποίησης των χρηστών, αδήριτη είναι η ανάγκη για εγκαθίδρυση κρυπτογραφημένων καναλιών μεταξύ όλων των εμπλεκόμενων μερών. Υπό αυτό το πρίσμα, η επικοινωνία τόσο με το UAC και το PAS, όσο και με τα PACS γίνεται αξιοποιώντας το πρωτόκολλο HTTPS (HTTP Secure), ενώ για το peer-to-peer κομμάτι μεριμνά το ίδιο το WebRTC, το οποίο έχει ενσωματωμένους κρυπτογραφικούς μηχανισμούς. Συγκεκριμένα, η μεταφορά βίντεο και ήχου γίνεται αξιοποιώντας το πρωτόκολλο SRTP (Secure Real-Time Protocol) τη στιγμή που οποιαδήποτε άλλα δεδομένα πάνω από τα data channels (όπως ιατρικές εικόνες) κωδικοποιούνται μέσω DTLS.

¹¹ Η AAI επιτρέπει σε διαφορετικούς οργανισμούς να συνεργάζονται στην εκχώρηση δικαιωμάτων πρόσβασης για εφαρμογές που έχουν διδρυματικό χαρακτήρα. Μέσω της υποδομής, οι χρήστες μιας ομοσπονδίας μπορούν να λάβουν υπηρεσίες με ασφάλεια και εμπιστευτικότητα των προσωπικών τους δεδομένων χρησιμοποιώντας απλά τον ιδρυματικό τους λογαριασμό. Συνήθως, τα μέλη μιας ομοσπονδίας χωρίζονται σε δύο κατηγορίες: τους παρόχους ταυτότητας (Identity Providers – IdP) και τους παρόχους υπηρεσιών (Service Providers - SP). Η προτεινόμενη πλατφόρμα αποτελεί στην ουσία μια παρεχόμενη υπηρεσία που απευθύνεται στην ευρεία κοινότητα των επαγγελματιών υγείας. Η πρόσβαση, λοιπόν, σε αυτή μπορεί να γίνει από ένα κεντρικό σημείο χωρίς να απαιτείται η υλοποίηση ξεχωριστών μεθόδων ταυτοποίησης των χρηστών επί τούτου.

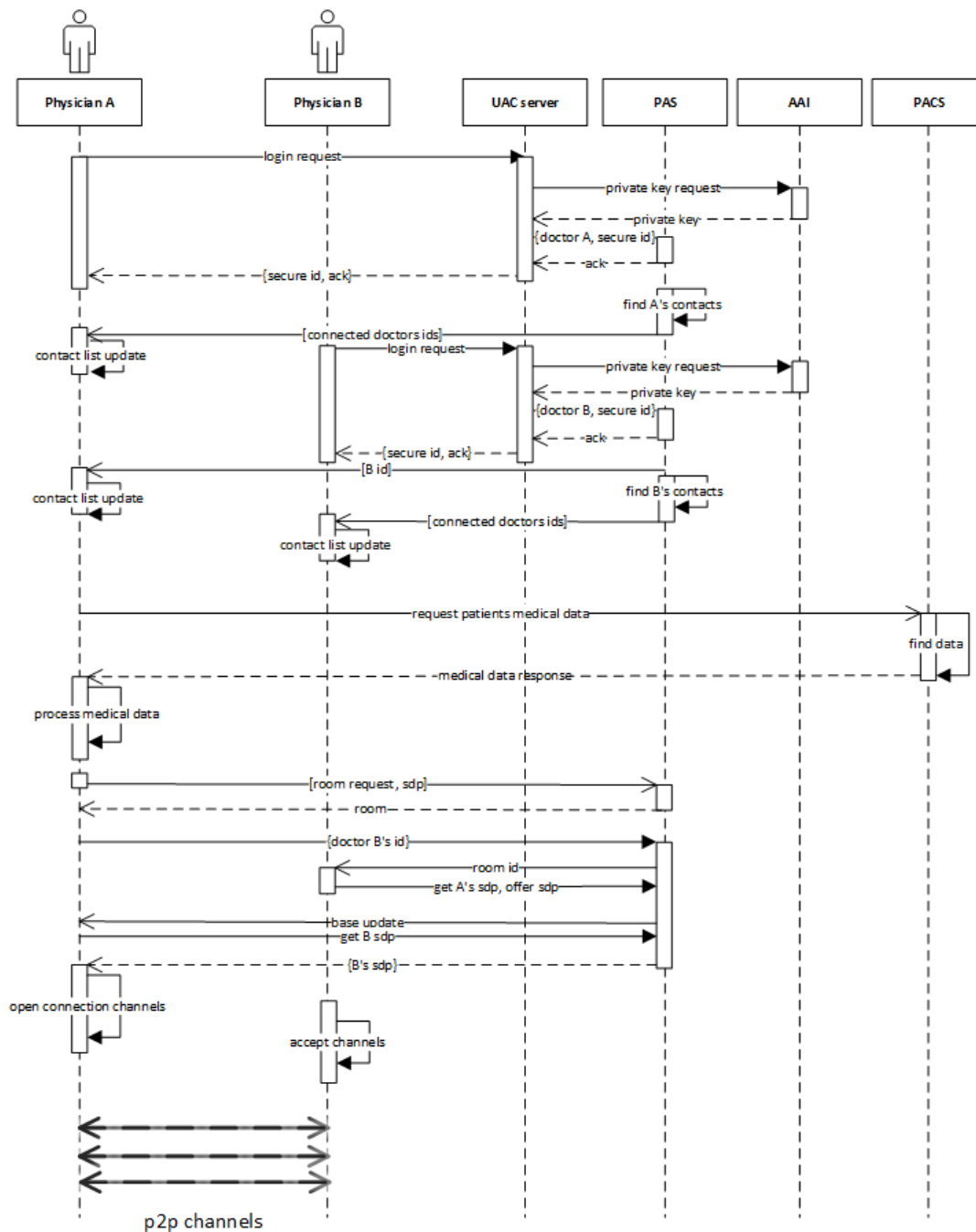


Εικόνα 23. Γενική άποψη της αρχιτεκτονικής

Η εγκαθίδρυση των peer-to-peer καναλιών επικοινωνίας απεικονίζεται στην *Εικόνα 24* και επιτυγχάνεται με τα ακόλουθα βήματα:

- Προκειμένου ένας χρήστης να συνδεθεί με την υπηρεσία, εισάγει το URL της εφαρμογής.
- Ο UAC, δεδομένου ότι ο χρήστης δεν φέρει κάποια ψηφιακή υπογραφή κατά την αίτησή του (request) – δηλαδή δεν είναι ταυτοποιημένος, ανακατευθύνεται στην υποδομή ταυτοποίησης και εξουσιοδότησης (AAI) και αποκτά μια ψηφιακή υπογραφή που βεβαιώνει την ιδιότητά του ως ιατρός.
- Κατόπιν, ενημερώνεται ο UAC για την αντιστοίχιση της ψηφιακής υπογραφής με το συγκεκριμένο άτομο και με τη σειρά του ενημερώνει τον PAS για την είσοδό του στο σύστημα και τη διαθεσιμότητά του.
- Ο χρήστης στη συνέχεια αποκτά πρόσβαση στην κύρια εφαρμογή.
- Ασύγχρονα ενημερώνεται η λίστα των επαφών του από τον PAS.
- Με την ίδια ακριβώς ακολουθία βημάτων έστω ότι συνδέεται και νέος χρήστης/γιατρός που ανήκει στη λίστα επαφών του πρώτου.
- Ασύγχρονα ενημερώνονται οι λίστες επαφών και των δύο.
- Έστω πως στη συνέχεια ο πρώτος χρήστης επιθυμεί να αποκτήσει πρόσβαση σε κάποιο PACS για το οποίο πληροί τα κριτήρια πρόσβασης. Του δίνεται η δυνατότητα να εκτελέσει αναζήτηση και να πραγματοποιήσει λήψη της ιατρικής εικόνας πάνω από ασφαλές δίαυλο.
- Επεξεργάζεται την εικόνα.
- Ζητά να εκκινήσει μια νέα συνεδρία από το PAS, ο οποίος του αναθέτει για το σκοπό αυτό ένα εικονικό «δωμάτιο».
- Προωθεί στον PAS το αναγνωριστικό της επαφής με την οποία επιζητά να συνεργαστεί και ο τελευταίος προωθεί το αίτημα στο δεύτερο ιατρό.
- Με διαμεσολαβητή (relay) τον PAS, ανταλλάσσονται τα offers¹² των χρηστών και εγκαθίσταται peer-to-peer επικοινωνία μεταξύ τους. Τα δεδομένα συγχρονίζονται χωρίς την ανάγκη κάποιου server πλέον.

¹² Πρόκειται για ειδικά JavaScript αντικείμενα τα οποία συμμορφώνονται με το πρωτόκολλο SDP που υιοθετεί το WebRTC και χρησιμεύουν στην εγκαθίδρυση συνεδριών με πολυμέσα.



Εικόνα 24. Ακολουθιακό διάγραμμα διαδικασίας εγκαθίδρυσης κλήσης

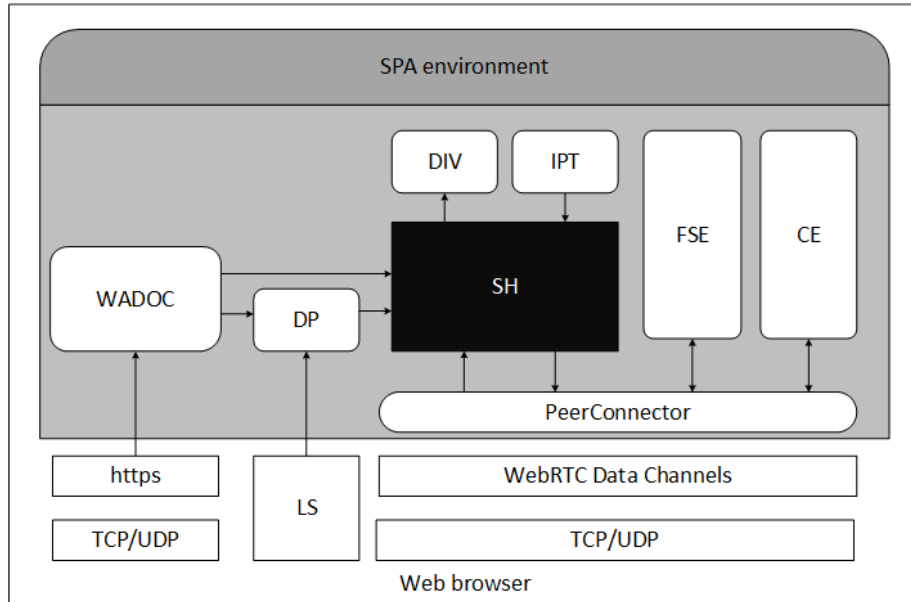
4.3 Αρχιτεκτονική εφαρμογής πελάτη

Η δόμηση της SPA έγινε γύρω από επτά πλήρως αυτόνομες μονάδες (Εικόνα 25):

- Η μονάδα WADOC (WADO Connector) είναι υπεύθυνη για την αναζήτηση, την αίτηση και την αποδοχή αποθηκευμένων αντικειμένων DICOM, δηλαδή ιατρικών εικόνων και αναφορών, από απομακρυσμένες αποθήκες δεδομένων (repositories) μέσω ενός απλού REST API.
- Η μονάδα DP (DICOM Parser) είναι επιφορτισμένη με το ρόλο της ανάλυσης (parsing) των αρχείων DICOM και της εξαγωγής τόσο της ιατρικής εικόνας όσο

και των υπόλοιπων δεδομένων που περιέχονται σε αυτά. Τα αρχεία μπορούν να προέρχονται, σύμφωνα και με τις προδιαγραφές, από κάποια μορφή τοπικού αποθηκευτικού χώρου (LS – Local Storage) – από το σκληρό δίσκο του χρήστη, USB stick, CD/DVD, αλλά και από κάποιο απομακρυσμένο PACS, μέσω του WADOC. Λόγω της νευραλγικότητας της συγκεκριμένης μονάδας για την αποκρισιμότητα και τη συνολική επίδοση της εφαρμογής, σχεδιάστηκε με τέτοιο τρόπο ώστε να εκμεταλλεύεται τις σύγχρονες πολυπύρηνες αρχιτεκτονικές.

- Η μονάδα DIV (DICOM Image Viewer) είναι η μονάδα που αναλαμβάνει την ακριβή και ταχεία εμφάνιση της ιατρικής εικόνας, η οποία ανακτήθηκε σε πρότερο στάδιο είτε από τη DP είτε από τη WADOC.
- Η μονάδα IPT (Image Processing Toolbox) παρέχει όλη την απαραίτητη λειτουργικότητα για την αλληλεπίδραση του χρήστη με την ιατρική εικόνα. Η αλληλεπίδραση αυτή συνεπάγεται αλλαγές της DIV. Στην ουσία αποτελεί την εργαλειοθήκη της SPA για τη διαχείριση και επεξεργασία εικόνας.
- Η μονάδα CE (Conference Engine) αποδίδει τις ροές βίντεο (απομακρυσμένες και μη) των συμμετεχόντων σε μια συνεδρία στην οθόνη.
- Η μονάδα FSE (File Sharing Engine) επιτρέπει την ανταλλαγή (μη DICOM) αρχείων.
- Η μονάδα PeerConnector είναι η μονάδα εκείνη που αξιοποιεί το WebRTC API για να διαχειριστεί με έξυπνο και αποδοτικό τρόπο τις ανάγκες επικοινωνίας της εφαρμογής. Αυτή, σε άμεση συνεργασία με τον PAS, εκτελεί τις διαπραγματεύσεις μεταξύ των ενδιαφερόμενων μερών για τον τρόπο και το είδος των καναλιών επικοινωνίας που πρόκειται να εγκαθιδρυθούν και στη συνέχεια φροντίζει να μεταδίδει τα ασύγχρονα μηνύματα που προκύπτουν από και προς τους συνδεδεμένους peers.

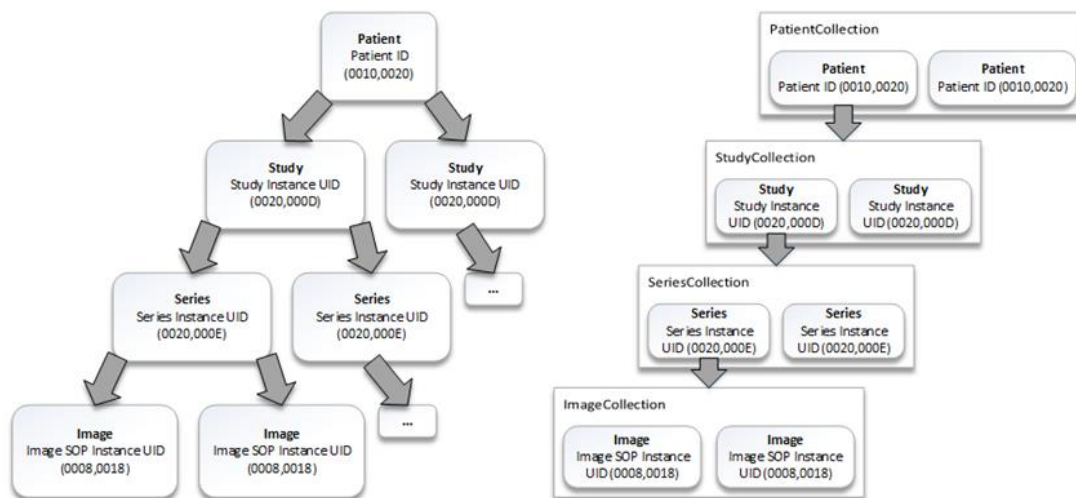


Εικόνα 25. Η αρχιτεκτονική της SPA

Το συνεργατικό περιβάλλον πραγματικού χρόνου λοιπόν πρέπει να διευθετεί θέματα αλληλεπίδρασης χρηστών, επικοινωνίας με το δίκτυο και επιπλέον, το συγχρονισμό και τη συνοχή των δεδομένων μεταξύ των ομότιμων χρηστών του. Το ρόλο αυτό αναλαμβάνει να εκτελέσει το SH (Symmetric Hear) – η καρδιά της όλης υλοποίησης γύρω από την οποία δομούνται και οι υπόλοιπες μονάδες.

Ουσιαστικά, το SH είναι ένα σύνολο δυναμικών δομών δεδομένων που εναρμονίζεται πλήρως με την ιεραρχία πληροφοριών του προτύπου DICOM (βλ. 2.1.3.4) και προσανατολίζονται στη διατήρηση της κατάστασης της εφαρμογής πελάτη (Εικόνα 26). Τα αντικείμενα DICOM, είτε προέρχονται από τη μονάδα DP, τη WADOC ή τη PeerConnector, αντιστοιχίζονται σε μοντέλα. Μοντέλα που ανήκουν στο ίδιο επίπεδο της ιεραρχίας, οργανώνονται σε συλλογές. Κάθε μοντέλο μιας ιεραρχίας κρατά μια αναφορά σε μια συλλογή μοντέλων του αμέσως κατώτερου επιπέδου της ιεραρχίας. Με τον τρόπο αυτό καθίσταται γρήγορη και ταυτόχρονα διαισθητική η διαδικασία εύρεσης, ανάκτησης και εισαγωγής νέας πληροφορίας.

Για παράδειγμα, έστω ότι ένας χρήστης «φορτώνει» ένα αρχείο DICOM από το σκληρό του δίσκο. Αφού εξαχθεί όλη η απαραίτητη πληροφορία από τη DP, θα πρέπει με κάποιο τρόπο να εισαχθεί στη δένδρική δομή. Έτσι, αν υποθέσουμε πως το αρχείο περιείχε μια μόνο ιατρική εικόνα (single frame), δημιουργείται ένα νέο μοντέλο Image με το δικό του μοναδικό αναγνωριστικό που αντιστοιχεί στο πεδίο Image SOP Instance. Ανάλογα τώρα με τα υπόλοιπα πεδία – Patient ID, Study Instance UID και Series Instance UID, εκτελείται αναζήτηση στη δομή για το πού πρέπει να τοποθετηθεί. Στην περίπτωση που δεν υπάρχει καταχώρηση για το συγκεκριμένο ασθενή (δεν υπάρχει μοντέλο ασθενή με το συγκεκριμένο Patient ID στη συλλογή ασθενών), δημιουργείται ένα νέο μοντέλο και γι' αυτόν. Στη συνέχεια, νέα μοντέλα κατασκευάζονται και για τη συγκεκριμένη μελέτη και σειρά στην οποία ανήκει η εικόνα και τοποθετούνται στα κατάλληλα επίπεδα. Αν ο ασθενής υπάρχει στη δομή, εκτελείται αναζήτηση στο υποδέντρο του για το αν υπάρχει καταχώρηση για τη μελέτη στην οποία ανήκει η εικόνα, κ.ό.κ. Στην περίπτωση τώρα που στο παρελθόν έχει φορτωθεί ήδη τουλάχιστον μία εικόνα που ανήκει στο συγκεκριμένο ασθενή, στη συγκεκριμένη μελέτη και στη συγκεκριμένη σειρά, υπάρχουν οι αντίστοιχες καταχωρήσεις στη δομή, οπότε, μετά από αναζήτηση επίπεδο-επίπεδο, το μοντέλο της νέας εικόνας θα προστεθεί απλά στην ήδη υπάρχουσα συλλογή.



Εικόνα 26. Η αντιστοιχία της εσωτερικής οργάνωσης του SH με αυτή του DICOM

Η συνολική αρχιτεκτονική της εφαρμογής πελάτη, αλλά και η αρχιτεκτονική των επιμέρους μονάδων, όπως αυτές παρουσιάστηκαν, υπακούει στις αρχές του προγραμματισμού οδηγούμενου από συμβάντα (event-driven programming). Η ροή του προγράμματος καθορίζεται από συμβάντα προερχόμενα από ενημερώσεις των δομών του SH, κάθε μία εκ των οποίων φέρει ξεχωριστό νόημα και διαχειρίζεται από την αντίστοιχη μονάδα (Εικόνα 27).

4.4 Θέματα υλοποίησης

Στο σημείο αυτό παρατίθεται μια σύντομη παρουσίαση των σπουδαιότερων χαρακτηριστικών της υλοποίησης και του τρόπου με τον οποίο αξιοποιήθηκαν τελευταίες τεχνολογίες ιστού προκειμένου να επιτευχθεί καινοτομία από την παρουσιαζόμενη εφαρμογή στο πεδίο της απομακρυσμένης ιατρικής συνεργατικότητας.

4.4.1 Διατήρηση και αναπαράσταση κατάστασης εφαρμογής πελάτη

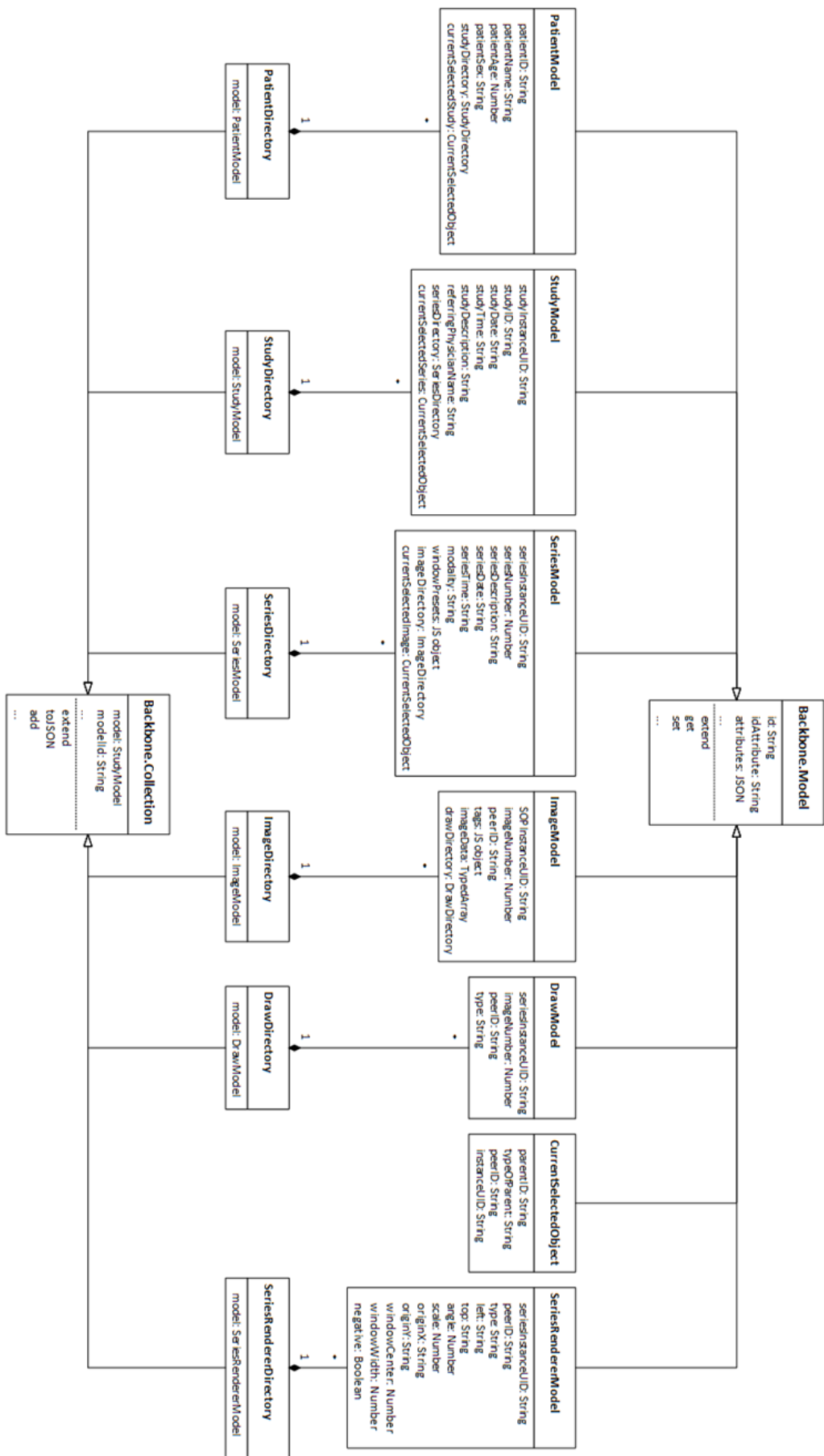
Στην υποενότητα 4.3 δόθηκε μια πρώτη διαισθητική προσέγγιση στο ζητούμενο της διατήρησης κατάστασης της εφαρμογής πελάτη και επεξηγήθηκε η πηγή της έμπνευσής της (ιεραρχία κατά DICOM). Αυτή η δένδρική δομή προέκυψε εντελώς φυσικά, καθώς κύρια σχεδιαστική απαίτηση της SPA είναι η δυνατότητα υποστήριξης προβολής πολλαπλών εικόνων προερχόμενων από διαφορετικές σειρές, μελέτες ή ακόμα και ασθενείς. Η επεξεργασία δε που θα υφίσταται οποιαδήποτε από αυτές, πρέπει επίσης να μπορεί να ανακτηθεί για όσο διάστημα τουλάχιστον η εφαρμογή παραμένει ανοικτή, καθώς επίσης και να είναι σε θέση να εφαρμοστεί απομακρυσμένα, σε πραγματικό χρόνο, από χρήστες που συμμετέχουν σε μια συνεδρία.

Για τη διατήρηση λοιπόν της κατάστασης σε JavaScript περιβάλλον, χρησιμοποιήθηκαν κάποια προγραμματιστικά εργαλεία που παρέχει ακριβώς για αυτό το σκοπό το Backbone.js – το framework πάνω στο οποίο βασίστηκε η εξέλιξη ολόκληρης της εφαρμογής. Τα Backbone μοντέλα χρησιμοποιούνται για τη διατήρηση του λεγόμενου data και business logic, τις φορτώσεις και τις αποθηκεύσεις από τον εξυπηρετητή και την εκπομπή συμβάντων όταν τα δεδομένα που περιέχουν υποστούν οποιαδήποτε αλλαγή. Οι Backbone συλλογές συνεισφέρουν στην οργάνωση και την ενιαία διαχείριση ομάδων από μοντέλα ίδιου είδους, παρέχοντας βοηθητικές λειτουργίες που δρουν στο σύνολο, απλουστεύοντας έτσι τους υπολογισμούς. Εκτός από τα δικά τους συμβάντα, μεσολαβούν επίσης σε όλες τις εκπομπές των μοντέλων που περιέχουν, επιτρέποντας έτσι την τοποθέτηση listeners σε ένα μόνο μέρος (πάνω στη συλλογή) αντί για κάθε μοντέλο ξεχωριστά.

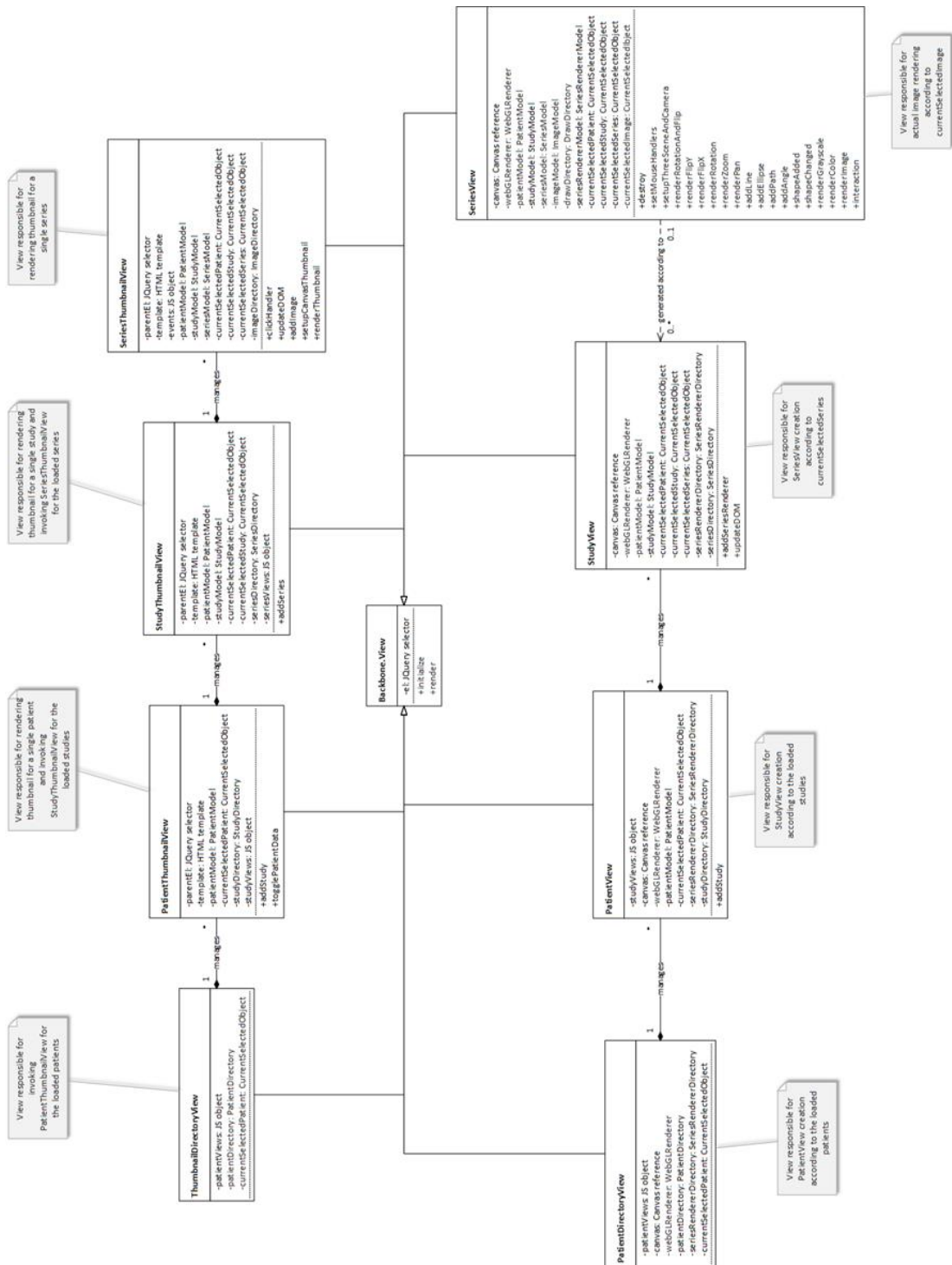
Το UML διάγραμμα της *Εικόνα 28* αποτελεί στην ουσία την εσωτερική δόμηση του Symmetric Hear που προέκυψε από τη μεταφορά όλης της ανωτέρω λογικής (που αναπαρίσταται και στην *Εικόνα 26*) σε όρους και αντικείμενα που «κατανοεί» η JavaScript και διευκολύνουν περαιτέρω την εργασία του προγραμματιστή. Όλη η πληροφορία ενθυλακώνεται μέσα σε μοντέλα, είτε πρόκειται για τα στοιχεία του ασθενή, της μελέτης ή της σειράς, είτε πρόκειται για τα ίδια τα δεδομένα της ιατρικής εικόνας. Σε ξεχωριστά μοντέλα αποθηκεύεται επίσης η πληροφορία για τους μετασχηματισμούς που έχουν εφαρμοστεί στις εικόνες σε επίπεδο σειράς (μεγέθυνση ή αλλαγή του μεγέθους παραθύρου), αλλά και για τυχόν μετρήσεις που έχουν γίνει πάνω σε συγκεκριμένες εικόνες. Ακολουθώντας τις καλύτερες πρακτικές που απορρέουν από τη χρήση της βιβλιοθήκης Backbone.js, τα όμοια μοντέλα κατόπιν οργανώνονται σε συλλογές.

Τα παραπάνω μπορεί να αρκούν για την απαραίτητη για οποιαδήποτε εφαρμογή αναπαράσταση και διατήρηση της εσωτερικής κατάστασης, αλλά στην τελική πρέπει κάποια πληροφορία να απεικονιστεί και στην οθόνη του χρήστη. Το σκοπό αυτό επιτελεί το έτερο βασικό συστατικό της βιβλιοθήκης, οι όψεις. Οι όψεις μπορούν να θεωρηθούν σαν ατομικά «κομμάτια» της διεπαφής χρήστη. Ενώ τα μοντέλα και οι συλλογές αντικειμένων είναι γενικά ανεξάρτητες μονάδες (δεν είναι ενήμερες για την ύπαρξη όψεων), οι όψεις «αφουγκράζονται» τυχόν αλλαγές τους και αντιδρούν κατάλληλα.

Συχνά χρησιμοποιούνται για την απόδοση (render) στην οθόνη δεδομένων προερχόμενων από ένα ή και περισσότερα μοντέλα, αλλά μπορεί επίσης να είναι και «κομμάτια» της διεπαφής χωρίς καθόλου δεδομένα. Επειδή δε το Backbone.js ανήκει στην οικογένεια των MV* frameworks και δεν έχει κάποια ξεχωριστή κλάση που να υλοποιεί τη λειτουργικότητα του controller, οι όψεις αναλαμβάνουν αυτό το ρόλο όταν χρειαστεί, οπότε, υπό προϋποθέσεις, όψεις μπορεί να καταλήγουν να μην έχουν καμία σχέση με πραγματικό UI rendering. Έτσι και στη παρουσιαζόμενη SPA, κάποιες όψεις είναι όντως υπεύθυνες για την απεικόνιση ιατρικών εικόνων και δεδομένων (είτε πρόκειται για thumbnails είτε πρόκειται για το κύριο παράθυρο απεικόνισης) και κάποιες άλλες είναι υπεύθυνες για την αφαίρεση και τη δημιουργία νέων, ανάλογα με την αλληλεπίδραση του χρήστη και τα διάφορα events που εκπέμπονται (*Εικόνα 29*).



Εικόνα 28. Ο SH υλοποιημένος με Backbone models και collections



Εικόνα 29. Διάγραμμα των Backbone views που εμπλέκονται στην απεικόνιση των ιατρικών εικόνων

4.4.2 Άνοιγμα αρχείων DICOM, parsing και εξαγωγή εικόνας

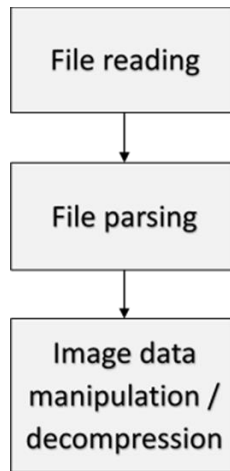
Κατά την εξέλιξη της εφαρμογής πελάτη δόθηκε ιδιαίτερη έμφαση στην υποστήριξη των περισσότερων κοινών αρχείων DICOM. Αυτό συνεπάγεται πως ο parser που αναπτύχθηκε ως αυτόνομο module και ενσωματώθηκε στη SPA, αναγνωρίζει και σαρώνει επιτυχώς τα βασικά transfer syntaxes – τρόπους κωδικοποίησης των δεδομένων

ενός έγκυρου αρχείου DICOM όπως παρουσιάζονται στο DICOM PS3.5 – που απεικονίζονται στον Πίνακα 4. Υποστηρίζονται επίσης οι πιο συνήθεις τιμές Photometric Interpretation – του πεδίου δηλαδή εκείνου που είναι υπεύθυνο για τον τρόπο διερμηνείας των δεδομένων της ιατρικής εικόνας (MONOCHROME1, MONOCHROME2, RGB), ενώ ταυτόχρονα έχει προβλεφθεί η περίπτωση αρχείων που περιέχουν πολλαπλά frames.

Όνομα Transfer Syntax	Transfer Syntax UID	Σημασία
<i>Διάταξη bytes για διαφορετικές υπολογιστικές αρχιτεκτονικές. Οι εικόνες είναι ασυμπίεστες:</i>		
Implicit VR Little Endian	1.2.840.10008.1.2	Προεπιλεγμένο DICOM Transfer Syntax
Explicit VR Little Endian	1.2.840.10008.1.2.1	Ίδιο με το άνωθεν, αλλά με ενσωματωμένους τύπους VR
Explicit VR Big Endian	1.2.840.10008.1.2.2	Αντεστραμμένη διάταξη bytes με ενσωματωμένους τύπους VR
<i>Μορφές συμπιεσμένης εικόνας:</i>		
DICOM JPEG2000 Lossy Compression	1.2.840.10008.1.2.4.91	Υλοποιεί τον αλγόριθμο συμπίεσης JPEG2000. Χρησιμοποιεί Explicit VR Little Endian

Πίνακας 4. Υποστηριζόμενα Transfer Syntaxes

Ο parser, ως ανεξάρτητη οντότητα, δέχεται ως όρισμα ένα αντικείμενο `ArrayBuffer`. Το τελευταίο χρησιμοποιείται για να αναπαραστήσει ένα γενικό, προκαθορισμένου μεγέθους buffer δυαδικών δεδομένων. Στην προκειμένη περίπτωση, ο `ArrayBuffer` περιέχει τα ακατέργαστα (raw) δεδομένα ενός DICOM αρχείου. Δημιουργείται κάθε φορά που ο χρήστης επιλέγει να ανοίξει ένα τέτοιο αρχείο από το τοπικό του μηχάνημα, μέσω της μεθόδου `readAsArrayBuffer` του `FileReader` API (ή επιστρέφεται μετά από XHR επικοινωνία με κάποιο WADO-compliant εξυπηρετητή). Ο parser τώρα αφού σαρώσει τον `ArrayBuffer`, επιστρέφει ένα αντικείμενο JavaScript, αποτελούμενο από key-value pairs, που είναι ό,τι πιο κοντινό σε μοντελοποίηση των αντικειμένων DICOM μπορεί να προσφέρει η γλώσσα. Ένα από αυτά τα ζεύγη αντιστοιχεί και στα δεδομένα της εικόνας, τα οποία αποθηκεύονται σε μια άλλη πρακτική δομή που παρέχει η JavaScript, γνωστή ως `TypedArray`. Τα δεδομένα αυτά όμως δεν είναι πάντα έτοιμα για απεικόνιση μέσω του `Canvas` ή του `WebGL` API – μπορεί να είναι συμπιεσμένα με κάποιο απωλεστικό ή μη απωλεστικό αλγόριθμο συμπίεσης ή ακόμα, όπως θα παρουσιαστεί στη συνέχεια, μπορεί να απαιτούν περαιτέρω επεξεργασία. Η συνολική ροή εργασίας (workflow) για την πλήρη ανάκτηση της εικόνας αλλά και των υπολοίπων δεδομένων που συνθέτουν ένα αρχείο DICOM απεικονίζεται στην *Εικόνα 30*.

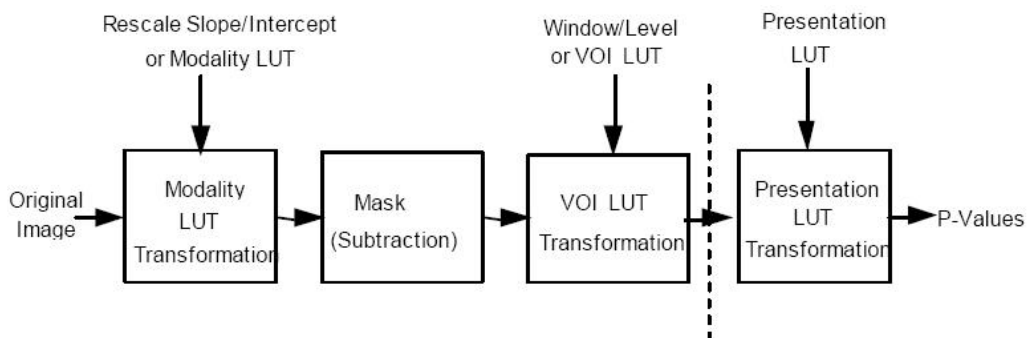


Εικόνα 30. Ροή εργασίας για την προβολή αρχείων DICOM

Το τελευταίο στάδιο που αναφέρθηκε αφορά στην αποσυμπίεση και τον κατάλληλο χειρισμό της εικόνας προκειμένου αυτή να είναι σε θέση να προβληθεί στην οθόνη του χρήστη. Καταρχήν, όπως παρουσιάστηκε στην υποενότητα 2.1.3.7, τα δεδομένα των εικονοστοιχείων μπορεί να είναι συμπιεσμένα για λόγους εξοικονόμησης χώρου και bandwidth κατά τη μεταφορά τους. Η εφαρμογή υποστηρίζει αυτή τη στιγμή την αποκωδικοποίηση συμπιεσμένων εικόνων με τον αλγόριθμο JPEG2000 (ίσως ο πιο διαδεδομένος σήμερα) αλλά σχεδιάζεται η μελλοντική επέκταση της υποστήριξης και σε άλλα πρότυπα συμπίεσης. Η βιβλιοθήκη που χρησιμοποιείται είναι η j2k.js. Αποτελεί τη μεταφορά σε JavaScript της άλλης γνωστής βιβλιοθήκης ανοικτού κώδικα OpenJPEG μέσω του μεταγλωττιστή Emscripten [51].

Ανεξάρτητα από την ανάγκη για αποσυμπίεση (άλλωστε δεν απαιτείται πάντοτε), οι μονοχρωματικές (grayscale) εικόνες απαιτούν επιπλέον μετασχηματισμούς προκειμένου να είναι σε θέση να αναπαρασταθούν ορθά. Το ίδιο το πρότυπο DICOM περιγράφει τη διαδικασία αυτή στο κεφάλαιο PS 3.4 κάτω από τον τίτλο Grayscale Transformations και τη διαχωρίζει στα τέσσερα ξεχωριστά στάδια της *Εικόνα 31*.

Η ακολουθία μονοχρωματικών μετασχηματισμών ορίζει τη μετατροπή της αποθηκευμένης DICOM εικόνας (ακατέργαστα δεδομένα εικονοστοιχείων) σε απεικονίσιμη. Η ακολουθία αυτή (όπως υποδηλώνει και το όνομά της) δεν αφορά έγχρωμες ιατρικές εικόνες – αυτές είναι έτοιμες προς απεικόνιση εφόσον αποσυμπίστούν. Από τους τέσσερις μετασχηματισμούς που αναφέρονται, η πλειονότητα των DICOM αρχείων περιέχουν πληροφορία (και άρα απαιτούν) για τον πρώτο και τον τρίτο. Η εφαρμογή, λοιπόν, τουλάχιστον στο τωρινό στάδιο της εξέλιξής της, υποστηρίζει ακριβώς αυτούς τους δύο.



Εικόνα 31. Σχηματική αναπαράσταση της ακολουθίας μονοχρωματικών μετασχηματισμών κατά DICOM

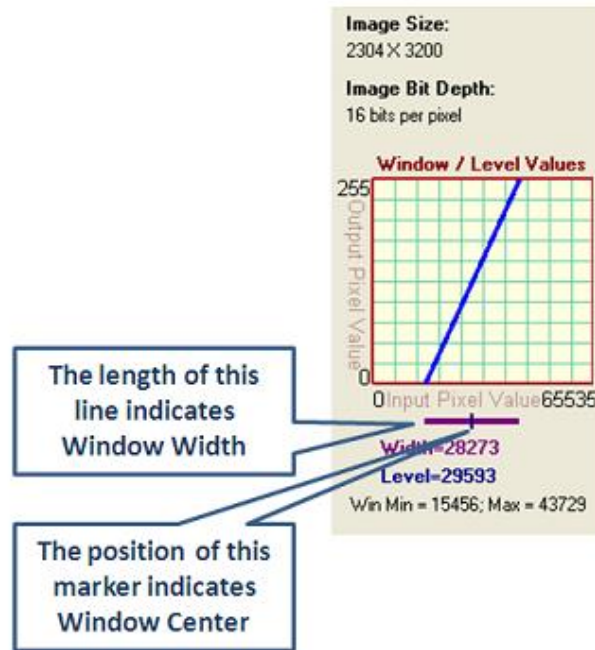
Ο πρώτος, γνωστός ως Modality LUT Transformation, είναι υπεύθυνος για τη μετατροπή των εξαρτώμενων από τον κατασκευαστή τιμών των εικονοστοιχείων σε τιμές που έχουν νόημα για την εκάστοτε μονάδα (modality) και είναι ανεξάρτητες από τον κατασκευαστή (π.χ. αριθμός Hounsfield για CT μονάδες). Οι τιμές αυτές μπορεί να αναπαριστούν φυσικά μεγέθη ή να είναι αδιάστατες. Συνήθως πρόκειται για γραμμικό μετασχηματισμό που περιγράφεται από τα πεδία του DICOM, Rescale Slope και Rescale Intercept.

Ο έτερος μετασχηματισμός που έχει ιδιαίτερο νόημα είναι ο τρίτος κατά σειρά, γνωστός ως VOI (Value Of Interest) LUT Transformation. Είναι υπεύθυνος για τη μετατροπή των εξαρτώμενων από τη συσκευή τιμών των εικονοστοιχείων (που παρήχθησαν από το Modality LUT Transformation) σε τιμές που έχουν πραγματικό νόημα για το χρήστη ή την εφαρμογή. Στη γραμμική του μορφή, περιγράφεται από τα πεδία Window Center (γνωστό και ως Window Level) και Window Width. Για να προσεγγιστεί το νόημά του αρκεί να αναλογιστεί κανείς πώς η φωτεινότητα και η αντίθεση μιας εικόνας μπορούν να προσαρμοστούν ώστε να τονιστούν ορισμένα σημεία ενδιαφέροντος. Η ενέργεια αυτή ονομάζεται "windowing" και αυτό που στην ουσία επιτυγχάνεται είναι η αναπροσαρμογή των προβαλλόμενων αποχρώσεων του γκρι μέσω επηρεασμού της χαμηλότερης και της υψηλότερης τιμής του παραθύρου, του εύρους δηλαδή των εικονοστοιχείων που μπορούν να εμφανιστούν. Το πεδίο Window Center υποδηλώνει την κεντρική τιμή μεταξύ αυτής της υψηλότερης και χαμηλότερης τιμής. Όσο μεγαλύτερη είναι, τόσο πιο σκούρα είναι η εικόνα, και το ανάποδο¹³. Το πεδίο Window Width είναι η διαφορά μεταξύ της μέγιστης τιμής που πρέπει να απεικονιστεί και της ελάχιστης. Έτσι, όσο μεγαλύτερη είναι διαφορά, τόσο μεγαλύτερη είναι και η αντίθεση της εικόνας.

Για παράδειγμα, έστω μια 16-bit μονοχρωματική εικόνα (που περιέχει δηλαδή $2^{16} = 65536$ αποχρώσεις του γκρι) και ο χρήστης επιθυμεί να επικεντρωθεί στις τιμές εικονοστοιχείων μεταξύ 15456 και 43729. Σε αυτή την περίπτωση, το Window Center αποκτά την κεντρική τιμή $\left\lfloor \frac{(15456 + 43729)}{2} \right\rfloor = 29593$ και το Window Width γίνεται ίσο με τη διαφορά, δηλαδή 28273 (βλ. *Εικόνα 32*). Όσα εικονοστοιχεία έχουν τιμή μεγαλύτερη από 43729 εμφανίζονται ως φωτεινά, ενώ όσα έχουν τιμή μικρότερη από 15456, εμφανίζονται ως σκούρα (μαύρα). Οι τιμές των εικονοστοιχείων που βρίσκονται στο ενδιάμεσο διάστημα αντιστοιχίζονται με γραμμικό τρόπο (εκτός και αν ορίζεται διαφορετικά) στο διάστημα [0-255].

Ο μετασχηματισμός αυτός έχει ιδιαίτερη ιατρική σημασία διότι οι ιατροί, πειραματιζόμενοι με διαφορετικές τιμές παραθύρων, είναι σε θέση να αποκαλύψουν λεπτομέρειες που διαφορετικά θα περνούσαν απαρατήρητες λόγω της μειωμένης δυνατότητας των οθονών σήμερα να προβάλλουν πάνω από 256 τόνους του γκρι. Οι περισσότερες δε εφαρμογές που υποστηρίζουν την προβολή εικόνων DICOM, παρέχουν επίσης τη δυνατότητα για έλεγχο αυτού ακριβώς του μετασχηματισμού στα εικονοστοιχεία της εμφανιζόμενης εικόνας.

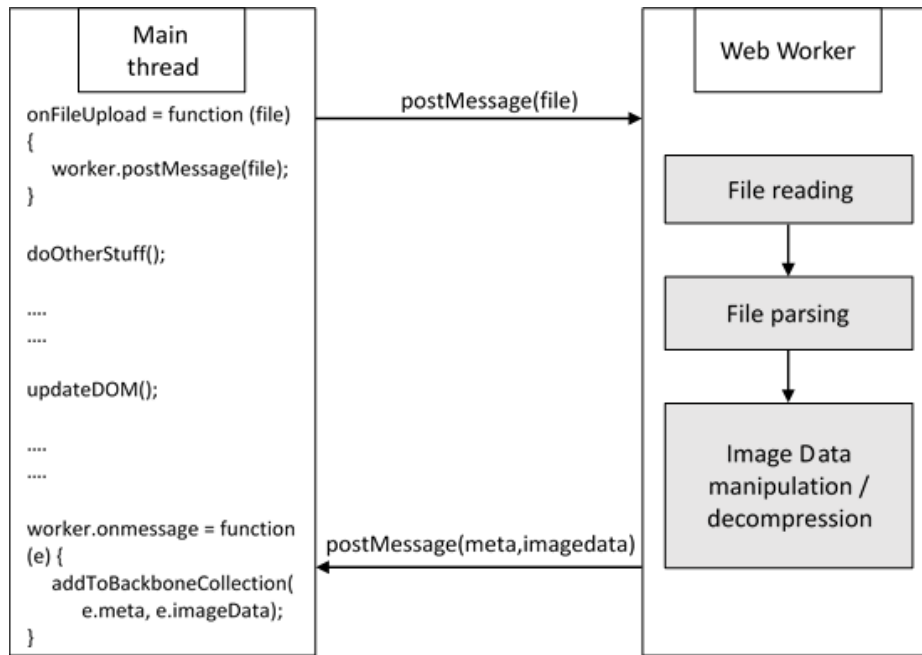
¹³ Οι περισσότερες συμβατικές οθόνες σήμερα υποστηρίζουν το πολύ μέχρι 8 bits ανά χρώμα και άρα μπορούν να απεικονίσουν το πολύ $2^8 = 256$ γκρι αποχρώσεις, με την τιμή 0 να αντιστοιχεί συνήθως στο σκούρο (μηδενική ένταση χρώματος) και την 255 στο φωτεινό (μέγιστη ένταση/φωτεινότητα). Όσο αυξάνεται λοιπόν η κεντρική τιμή του παραθύρου (Window Center) για δεδομένη τιμή του πλάτους (Window Width), τόσο περισσότερα pixels «κόβονται» – μένουν δηλαδή έξω από το παράθυρο και εμφανίζονται ως σκούρα.



Εικόνα 32. Παράδειγμα "windowing"

Στο σημείο αυτό γίνεται εύκολα κατανοητό πως η συνολική ροή εργασίας που απαιτείται για κάθε ένα αρχείο που φορτώνεται στη SPA είναι αρκετά χρονοβόρα και υπολογιστικά απαιτητική, ειδικά αν αναλογιστεί κάποιος πως στις απαιτήσεις μιας τέτοιας εφαρμογής συμπεριλαμβάνεται η δυνατότητα για άνοιγμα και προβολή πολλαπλών αρχείων ή ακόμα και καταλόγων. Η εξ' ολοκλήρου ανάθεση της εκτέλεσής της στο περιβάλλον του κυρίου νήματος θα καθιστούσε ολόκληρη την εφαρμογή μη αποκρίσιμη για τουλάχιστον μερικά δευτερόλεπτα. Η λύση στο πρόβλημα αυτό δόθηκε με τη χρησιμοποίηση των Web Workers. Κάθε φορά που ο χρήστης επιλέγει να ανοίξει κάποιο αρχείο από κάποιο τοπικό φάκελο, το κύριο νήμα που είναι υπεύθυνο για την αλληλεπίδραση με το UI αποκτά πρόσβαση σε ένα File object που αντιστοιχεί στο αρχείο αυτό και κατόπιν το αποστέλλει μέσω της `postMessage()` κλήσης στο περιβάλλον ενός worker που έχει εκκινήσει με αποκλειστικό σκοπό την εκτέλεση του παραπάνω workflow. Σε περίπτωση ανοίγματος πολλαπλών αρχείων, πραγματοποιείται αντίστοιχος αριθμός κλήσεων/αποστολών, τα workflow των οποίων σειριοποιούνται, αλλά με μηδενικές επιπτώσεις στην αντιλαμβανόμενη εμπειρία χρήσης. Όταν η επεξεργασία λάβει τέλος, ο worker περνά την εξαχθείσα πληροφορία με τον ίδιο τρόπο στο κύριο νήμα, το οποίο είναι ελεύθερο να τη διαχειριστεί (Εικόνα 33).

Προκειμένου να γίνει ακόμα καλύτερη και αποδοτική εκμετάλλευση των σύγχρονων πολυπύρηνων αρχιτεκτονικών, ο Chrome προχωρά ένα βήμα παραπέρα και δίνει τη δυνατότητα στον προγραμματιστή να γνωρίζει τον αριθμό των διαθέσιμων πυρήνων του μηχανήματος στο οποίο είναι εγκατεστημένος (μέσω του `navigator.hardwareConcurrency`). Η συγκεκριμένη SPA αξιοποιεί το γεγονός αυτό για να εκκινεί αντίστοιχο αριθμό workers, με αποτέλεσμα την επίτευξη αξιοσημείωτων επιδόσεων σχεδόν συγκρίσιμων με αντίστοιχων native εφαρμογών. Προς αυτή την κατεύθυνση συνέβαλλαν και επιπλέον βελτιστοποιήσεις που αφαίρεσαν περαιτέρω υπολογιστικό βάρος από το browser και θα σχολιαστούν στην επόμενη ενότητα.



Εικόνα 33. Αλληλεπίδραση μεταξύ κύριου νήματος εκτέλεσης και Web Worker

4.4.3 Ιατρική απεικόνιση και επεξεργασία

Από τα πρώτα στάδια του σχεδιασμού και της εξέλιξης της SPA, ήταν προδιαγεγραμμένο πως για το rendering της ιατρικής εικόνας στη σελίδα θα χρησιμοποιείτο το στοιχείο `<canvas>` της HTML5. Αυτό άλλωστε χρησιμοποιούσε και η πλειονότητα των υπόλοιπων γνωστών DICOM web viewers. Αποφασίστηκε, λοιπόν, η υλοποίηση να βασιστεί στο Canvas 2D API, αφενός λόγω της απλότητάς του σε σχέση με την εναλλακτική του WebGL, αφετέρου διότι ο βασικός στόχος, τουλάχιστον στην αρχή, ήταν η δισδιάστατη απεικόνιση ιατρικών εικόνων και θεωρήθηκε έτσι πως η χρήση ενός τρισδιάστατου API για το σκοπό αυτό θα ήταν υπερβολή.

Το Canvas 2D API παρέχει δύο μεθόδους για την αναπαράσταση εικόνων, τη `drawImage()` και τη `putImageData()`. Η πρώτη δέχεται ως όρισμα μια αναφορά σε ένα αντικείμενο `HTMLImageElement`¹⁴ ή σε ένα διαφορετικό `<canvas>` instance και στην ουσία αποτυπώνει τα περιεχόμενά τους στο συγκεκριμένο `<canvas>`. Η δεύτερη δέχεται σαν κύριο όρισμα ένα αντικείμενο `ImageData`¹⁵, το οποίο και αποδίδει στην οθόνη. Ενώ θα περίμενε κανείς πως λόγω της φύσης του `ImageData` (που βρίσκεται πιο κοντά στη λογική του Canvas), η `putImageData()` θα ήταν πιο αποδοτική από τη `drawImage()`, στην πράξη συμβαίνει το αντίθετο. Μάλιστα, η διαφορά στους περισσότερους browsers είναι χαώδης υπέρ της `drawImage()` (βλ. και *Εικόνα 12*). Όσον αφορά όμως τον τύπο των δεδομένων που η SPA παράγει και χρησιμοποιεί, όπως

¹⁴ Το `HTMLImageElement` «κληρονομεί» από το `HTMLElement`, που είναι στην ουσία η αναπαράσταση ενός HTML element μέσω JavaScript. Έτσι, το `HTMLImageElement` παρέχει επιπρόσθετα ειδικές ιδιότητες και μεθόδους για τη διαχείριση της δόμησης και της παρουσίασης των `` elements.

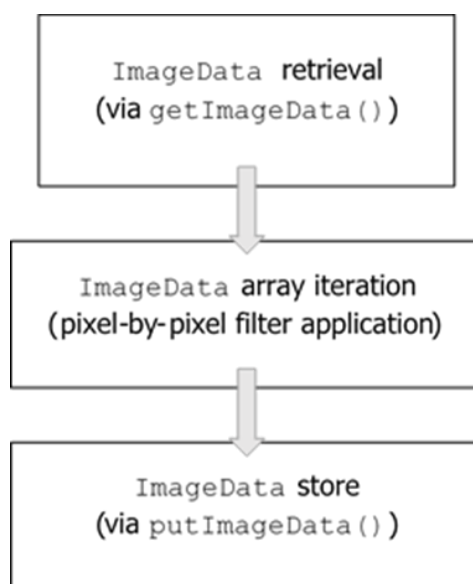
¹⁵ Το `ImageData` αναπαριστά τις τιμές των υποκειμένων εικονοστοιχείων ενός `<canvas>` element. Μπορεί να προσεγγιστεί ως ένας JavaScript `TypedArray` με επιπλέον ιδιότητες που λειτουργεί ως bitmap.

έχει ήδη αναλυθεί, πρόκειται για πίνακες από τιμές εικονοστοιχείων, οι οποίοι παράγονται μετά από τη διαδικασία του parsing. Στη φιλοσοφία της εφαρμογής, δηλαδή, ταιριάζει περισσότερο η `putImageData()`.

Μπροστά σε αυτό το δεύτερο μεγάλο σχεδιαστικό δίλημμα, η υλοποίηση επιλέχθηκε να βασιστεί στη `drawImage()`. Για το σκοπό αυτό όμως τα δεδομένα των εικόνων κάθε φορά έπρεπε να μετασχηματιστούν σε μια μορφή συμβατή με το `` element και κατά συνέπεια με το `HTMLImageElement` (για την ακρίβεια χρειάστηκε να κωδικοποιηθούν σε Base64 συμβολοσειρά), γεγονός που απέφερε κάποιο overhead μεταξύ της αρχικής φόρτωσης ενός DICOM αρχείου και της τελικής απεικόνισης στην οθόνη. Ωστόσο, με τον τρόπο αυτό εξασφαλίστηκε πως οι μετέπειτα εναλλαγές μεταξύ των εικόνων (π.χ. κατά τη διάρκεια που ο χρήστης επιλέγει να πλοηγηθεί μέσα σε μια σειρά) θα είναι όσο γίνεται πιο ομαλές και στην ουσία μη αντιληπτές από το χρήστη. Στην αντίθετη περίπτωση, μπορεί να εξοικονομείτο αρχικός χρόνος, αλλά στη συνέχεια θα παρατηρούνταν καθυστερήσεις που δε συνάδουν με μια αξιοπρεπή εμπειρία χρήσης.

Στον προγραμματισμό πάνω στο Canvas API συνεισέφερε η `fabric.js`, μια ισχυρή HTML5 βιβλιοθήκη, η οποία παρέχει μια διαδραστική μοντελοποίηση αντικειμένων πάνω από το στοιχείο `<canvas>`. Διευκόλυε ιδιαίτερα το έργο της διαχείρισης εικόνων, γεωμετρικών σχημάτων και κειμένων με έναν ενιαίο και συμπαγή τρόπο. Η αντιμετώπιση κάθε αναπαριστώμενου αντικειμένου ως ξεχωριστού μοντέλου με ειδικά γνωρίσματα (σε αντιστοιχία περίπου με αυτή των μοντέλων του `Backbone.js`) που μπορεί να εκπέμπει συμβάντα σε περιπτώσεις αλληλεπίδρασης με το χρήστη (όπως `mouse click` ή `mouse hover`) απλοποίησε επίσης τη διαδικασία εξάπλωσης (`propagation`) των αλλαγών και στους απομακρυσμένους χρήστες, όπως θα αναλυθεί και στην επόμενη υποενότητα.

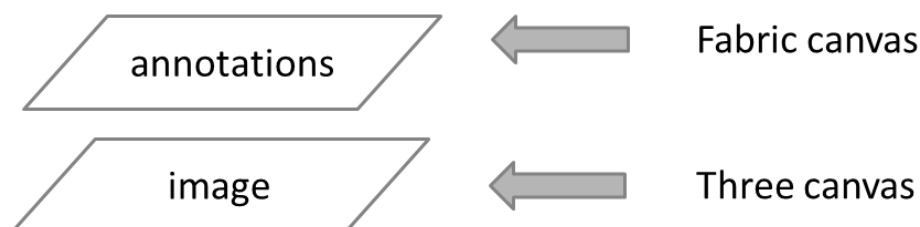
Ενώ οι επιδόσεις κρίνονταν παραπάνω από ικανοποιητικές για σχετικά απλές ενέργειες, όπως η μετακίνηση, η μεγέθυνση/σμίκρυνση και η εναλλαγή των εικόνων, ο σχεδιασμός γραμμών, ελλείψεων και μονοπατιών και η απόδοση κειμένου – εργασίες στις οποίες είναι γνωστό πως το Canvas 2D API τα καταφέρνει περίφημα, τα πράγματα αντιστράφηκαν κατά την εφαρμογή φίλτρων ή τη ρύθμιση του παραθύρου στην εικόνα. Ο μόνος τρόπος για χαμηλού επιπέδου χειρισμό των εικονοστοιχείων (`low-level pixel manipulation`) είναι αυτός που απεικονίζεται κάτωθι (*Εικόνα 34*).



Εικόνα 34. Διαδικασία εφαρμογής φίλτρων μέσω του Canvas API

Στο σημείο αυτό αξίζει να αναφερθεί πως όλοι οι σύγχρονοι browsers προσφέρουν σημαντικές βελτιστοποιήσεις στο κομμάτι του δισδιάστατου rendering μέσω του Canvas αξιοποιώντας (όπως και το WebGL) την επεξεργαστική ισχύ των καρτών γραφικών. Η διαφορά όμως έγκειται στην έλλειψη δυνατότητας εκ μέρους του προγραμματιστή να επέμβει συντάσσοντας κώδικα που να εκτελείται απευθείας σε αυτές. Αντίθετα, κατά την αρχική κλήση για rendering (είτε μέσω της `drawImage()` είτε μέσω της `putImageData()`), αυτό που γίνεται στην ουσία είναι το «πέρασμα» των δεδομένων της εικόνας από την κεντρική μνήμη του υπολογιστή στη μνήμη της κάρτας γραφικών (διαδικασία χρονοβόρα). Από το σημείο αυτό και μετά, ο προγραμματιστής περιορίζεται από το Canvas API και έτσι αν θέλει να επέμβει σε επίπεδο εικονοστοιχείου, χρειάζεται μέσω της ειδικής κλήσης `getImageData()` να ανακαλέσει όλη την πληροφορία της εικόνας στη μνήμη RAM και να εκτελέσει κώδικα JavaScript για την σειριακή πρόσβαση στις τιμές κάθε μεμονωμένου εικονοστοιχείου, τη διεξαγωγή υπολογισμών και την επανεγγραφή των νέων τιμών που προκύπτουν. Στη συνέχεια, για να γίνουν ορατά τα αποτελέσματα, πρέπει να ακολουθήσει η κλήση στην `putImageData()` που θα «τοποθετήσει» το bitmap πάλι πίσω στην κάρτα γραφικών. Οι παρατηρούμενοι δε χρόνοι για όλες αυτές τις ενέργειες είναι ανάλογοι με το μέγεθος της εικόνας προς επεξεργασία – μεγαλύτερη ανάλυση εικόνας συνεπάγεται μεγαλύτερο αριθμό εικονοστοιχείων άρα και περισσότερο χώρο στη μνήμη που για να αντιγραφεί απαιτεί και περισσότερο χρόνο.

Στο σημείο αυτό, προκειμένου να παρακαμφθούν αυτοί οι εγγενείς περιορισμοί που απορρέουν από τη χρησιμοποίηση του Canvas 2D API, αποφασίστηκε η στροφή προς το WebGL, τουλάχιστον όσον αφορά το κομμάτι της ιατρικής εικόνας. Ως αποτέλεσμα, υιοθετήθηκε τελικώς μια υβριδική φιλοσοφία όσον αφορά το DIV (DICOM Image Viewer): δύο ξεχωριστά `<canvas>` elements το ένα στοιβαγμένο πάνω στο άλλο – το πάνω για την εμφάνιση κειμένου και επισημειώσεων των ιατρών μέσω του Canvas 2D API και το κάτω για το πραγματικό rendering και post-processing της ιατρικής εικόνας μέσω του WebGL API (Εικόνα 35).



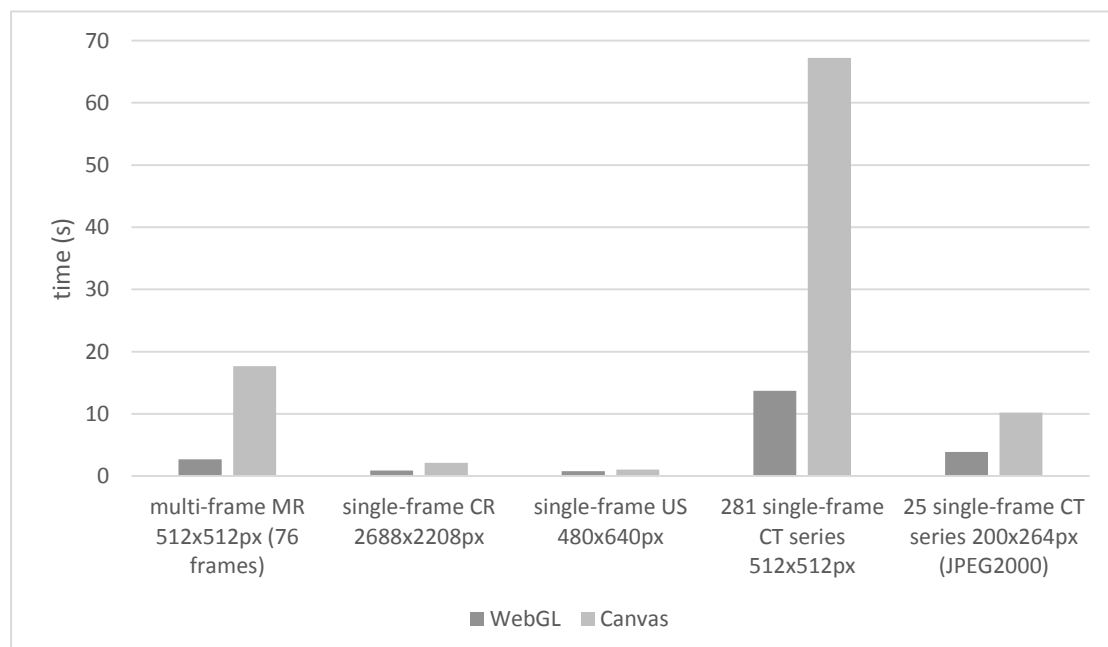
Εικόνα 35. Υβριδική υλοποίηση του DIV

Η βιβλιοθήκη που χρησιμοποιήθηκε για να προσφέρει δόμηση στον κώδικα και ένα αφαιρετικό επίπεδο πάνω από το πολύπλοκο API του WebGL, όπως επισημάνθηκε ήδη, είναι η THREE.js. Είναι γεγονός πως οι δημιουργοί της δεν είχαν ψηλά στις προτεραιότητές τους την υποστήριξη προβολής και επεξεργασίας δισδιάστατης εικόνας – όπως άλλωστε και οποιασδήποτε άλλης βιβλιοθήκης WebGL. Παρ’ όλα αυτά, χρησιμοποιώντας κάποια πολύ βασικά εργαλεία που παρέχει (π.χ. αντικείμενο `DataTexture`) και μετά από τη σύνταξη επιπλέον custom κώδικα σε GLSL, τα αποτελέσματα ήταν άκρως ενθαρρυντικά.

Η δύναμη του τρισδιάστατου αυτού API – που για το συγκεκριμένο σκοπό βέβαια δεν αξιοποιήθηκε ως τέτοιο, πηγάζει από τη δυνατότητα εκτέλεσης κώδικα κατευθείαν στη GPU. Έτσι, αν για παράδειγμα χρειαστεί να αλλάξει δυναμικά το μέγεθος του παρθύρου της ήδη προβαλλόμενης εικόνας (αν δηλαδή ο χρήστης μεταβάλλει την τιμή

του Window Center και/ή του Window Width, ή επιλέξει την εφαρμογή κάποιου φίλτρου), δεν εμπλέκεται καθόλου η JavaScript (που θα απαιτούσε την αντιγραφή των δεδομένων της εικόνας πίσω στην κύρια μνήμη), αλλά αντίθετα παρέχονται κάποιες συγκεκριμένοι παράμετροι στο GLSL κώδικα των shaders. Ο τελευταίος εκτελείται παράλληλα πάνω σε κάθε εικονοστοιχείο μέσα στα πολλαπλά shader units των σύγχρονων GPUs και οι αλλαγές στην τελική εικόνα παρατηρούνται σχεδόν ακαριαία.

Δεδομένης της παρεχόμενης ευελιξίας που προκύπτει από τη δυνατότητα φόρτωσης ακατέργαστων δεδομένων εικόνας κατευθείαν σε buffers της κάρτας γραφικών, πλέον δεν είναι απαραίτητη η επιπλέον Base64 κωδικοποίηση της εικόνας που απαιτείτο όσο χρησιμοποιούνταν η μέθοδος `drawImage()` του Canvas API – ο `TypedArray` που δημιουργείται κατά το διαδικασία του parsing παρέχεται απευθείας στη GPU. Ο απόλυτος έλεγχος δε για το πού και πώς θα παρουσιαστούν τα εικονοστοιχεία πάνω στο `<canvas>` μέσω κώδικα GLSL, οδήγησε στη μεταφορά όλων των απαιτούμενων υπολογισμών για τους μονοχρωματικούς μετασχηματισμούς από περιβάλλον JavaScript (Web Worker για την ακρίβεια) σε περιβάλλον GPU. Τα οφέλη από τη διπλή αυτή αναπροσαρμογή του κώδικα είναι αδιαμφισβήτητα, όπως απεικονίζεται και στην *Εικόνα 36*.



Εικόνα 36. Σύγκριση επιδόσεων μεταξύ WebGL-based και Canvas-based υλοποίησης. Το διάγραμμα απεικονίζει το συνολικό χρόνο φόρτωσης (από το άνοιγμα μέχρι την εμφάνιση των εικόνων).

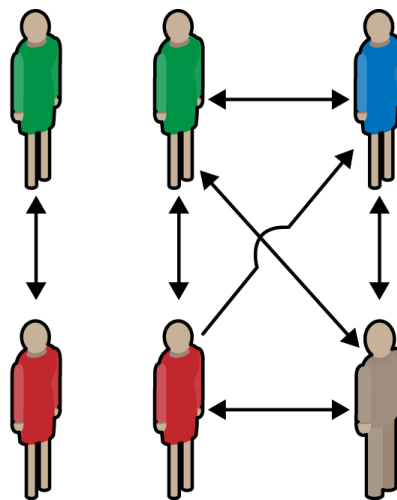
4.4.4 Επικοινωνία μέσω WebRTC

Το σπουδαιότερο ίσως από όλα τα καινοτόμα χαρακτηριστικά της παρουσιαζόμενης SPA είναι η έννοια της συνεργατικότητας σε πραγματικό χρόνο πάνω σε ιατρικές εικόνες. Οι χρήστες που επιθυμούν να εκκινήσουν μια απομακρυσμένη «εικονική» συνεδρία, όπως ήδη αναφέρθηκε, δεν έχουν παρά να αποστείλουν με κάποιον άλλο μηχανισμό επικοινωνίας το μοναδικό αναγνωριστικό που αντιστοιχεί στο στιγμιότυπο της εφαρμογής τους και οι υπόλοιποι ενδιαφερόμενοι να κάνουν "Join" στο εικονικό «δωμάτιο» που δημιουργείται. Οι περισσότερες πολύπλοκες εργασίες που απαιτούνται για την τελική εγκαθίδρυση των διαύλων επικοινωνίας μεταξύ των συμμετεχόντων, τακτοποιούνται από το WebRTC και έτσι ο προγραμματιστής δε χρειάζεται να ασχοληθεί με χαμηλού επιπέδου ζητήματα που θα έπρεπε να αντιμετωπίσει σε άλλη περίπτωση – αν

για παράδειγμα επιχειρούσε να αναπτύξει μια αντίστοιχη standalone/native εφαρμογή. Έτσι, θέματα που αφορούν την πρόσβαση σε τοπικούς πόρους (όπως η κάμερα και το μικρόφωνο του χρήστη), τη μέθοδο κωδικοποίησης των ροών βίντεο και ήχου, τα πρωτόκολλα που αναλαμβάνουν τη μεταφορά των δεδομένων και τυχόν απώλειες που μπορεί να προκύψουν, καθώς επίσης και άλλες βελτιστοποιήσεις που εκ πρώτης όψης δεν είναι ορατές, έχουν προβλεφθεί και υλοποιηθεί από τους περισσότερους τουλάχιστον browser vendors. Εντούτοις, όπως κάθε νέα τεχνολογία, προτού καθιερωθεί, έτσι και το WebRTC έχει τις δικές του ατέλειες, ορισμένες εκ των οποίων θα επισημανθούν στη συνέχεια, καθότι επηρέασαν την εξέλιξη της εφαρμογής.

Αυτή τη στιγμή, λόγω έλλειψης ξεχωριστού, προτυποποιημένου μηχανισμού σηματοδοσίας μεταξύ των ενδιαφερόμενων μερών, χρησιμοποιήθηκε για το σκοπό αυτό η υποδομή νέφους (cloud) της Firebase, η οποία, εκτός των άλλων, παρέχει ελεύθερα κανάλια σηματοδοσίας για τα πρώτα στάδια εξέλιξης αντίστοιχων εφαρμογών. Το μεγαλύτερο ενδιαφέρον, ωστόσο δεν έγκειται στον ίδιο το μηχανισμό – άλλωστε υπάρχουν αρκετές εναλλακτικές, με μικρότερη ή μεγαλύτερη πολυπλοκότητα η καθεμιά – αλλά στη δόμηση της εφαρμογής και των δεδομένων της ώστε να εκπληρώνεται ο απαιτούμενος συγχρονισμός και επικοινωνία μεταξύ των διασυνδεδεμένων χρηστών.

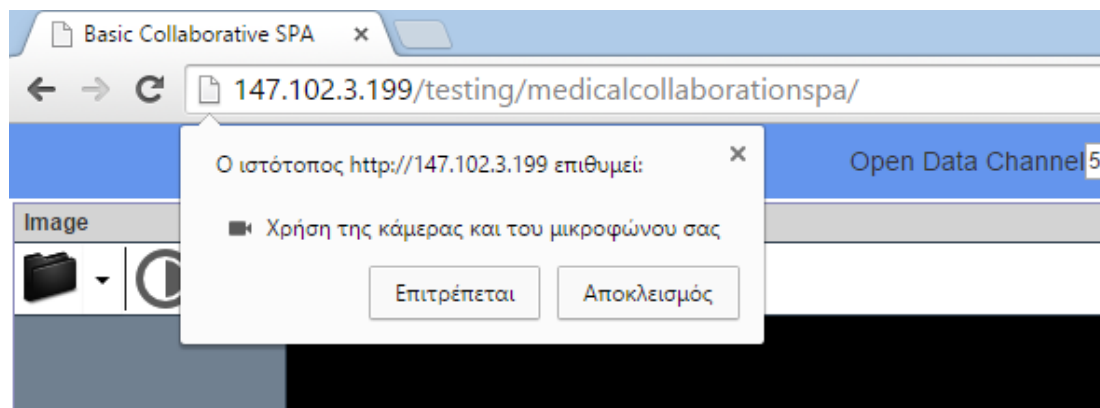
Στην *Εικόνα 37* αναπαρίσταται η λογική των συνδέσεων που ακολουθεί το ίδιο το WebRTC. Ανεξαρτήτως του αριθμού των συμμετεχόντων σε μια κλήση/συνεδρία, εγκαθιδρύονται άμεσες συνδέσεις όλων με όλους. Για οποιαδήποτε, δηλαδή, κλήση μεταξύ N χρηστών, ο καθένας τους βρίσκεται στο άκρο $N - 1$ συνδέσεων με τους υπολοίπους. Με τον τρόπο αυτό επιτυγχάνεται στην ουσία η ad-hoc δημιουργία μικρών peer-to-peer δικτύων, όπου είναι δυνατή η ανταλλαγή ενός μεγάλου εύρους τύπου δεδομένων απευθείας μεταξύ των κόμβων/χρηστών.



Εικόνα 37. Κλιμάκωση των καναλιών επικοινωνίας όπως παρέχεται από το WebRTC. Άμεση αμφίδρομη σύνδεση μεταξύ δύο χρηστών, αμφίδρομες συνδέσεις σε πλέγμα για περισσότερους.

Για πολλές εφαρμογές, οι άμεσες αυτές συνδέσεις είναι το μοναδικό που χρειάζονται. Μέσω του RTCPeerConnection API, που είναι υπεύθυνο για τη διαχείριση του πλήρους κύκλου ζωής των peer-to-peer καναλιών και, μεταξύ άλλων, τη μετάδοση των ροών βίντεο και ήχου, μπορούν να προστεθούν χαρακτηριστικά βιντεοκλήσεων σε οποιαδήποτε ιστοσελίδα. Πάνω σε αυτό βασίζεται εν μέρει και η συγκεκριμένη SPA ώστε να παρέχει την απαραίτητη αμεσότητα στην επικοινωνία μεταξύ του απομακρυσμένου ιατρικού προσωπικού. Κάθε φορά που ο χρήστης επιλέγει την εκκίνηση μιας τηλεσυνεδρίας, έχει τη δυνατότητα προσθήκης σε αυτή βίντεο ή/και ήχου (η ρητή άδειά του

στο browser είναι πάντα αναγκαία για λόγους ασφαλείας, όπως απεικονίζεται χαρακτηριστικά στην *Εικόνα 38*). Από τη στιγμή που ο πρώτος προσκεκλημένος εισέλθει στην εικονική συνεδρία, ξεκινά η άμεση – χωρίς ενδιάμεσους – ανταλλαγή ροών βίντεο και ήχου, οι οποίες αποδίδονται στην οθόνη μέσα από συνηθισμένα HTML `<video>` elements (βλ. και υποενότητα 5.1.1).



Εικόνα 38. Ο browser ζητά πάντα την άδεια του χρήστη προκειμένου να αποκτήσει πρόσβαση στην κάμερα και το μικρόφωνο.

Η δύναμη, ωστόσο, της εν λόγω εφαρμογής εδράζεται στη χρησιμοποίηση του RTCDataChannel, του ετέρου API που παρέχει το WebRTC. Παλιότερα, η αποστολή δεδομένων μεταξύ browsers για επικοινωνία, παιχνίδια ή μεταφορά αρχείων αποτελούσε ιδιαίτερη πρόκληση. Απαιτούσε την παρουσία κάποιου εξυπηρετητή που θα διαμεσολαβούσε στην αποστολή αυτή και πιθανώς, για την αντιμετώπιση θεμάτων κλιμακωσιμότητας, την εμπλοκή πολλαπλών data centers, κάτι που μεταφράζεται σε αυξημένα έξοδα. Επίσης, το σενάριο αυτό περιλαμβάνει την εν δυνάμει εμφάνιση σημαντικών καθυστερήσεων (high latency) αλλά και προβλημάτων σχετιζόμενων με την ασφάλεια των δεδομένων. Όλα αυτά τα ζητήματα όμως μπορούν να αντιμετωπιστούν πλέον αποτελεσματικά από το νεοεμφανιζόμενο αυτό δίκτυο επικοινωνίας που αναλαμβάνει την άμεση μεταφορά των δεδομένων από τον ένα ομότιμο (peer) στον άλλο, χωρίς διαμεσολαβητές. Από αυτό ακριβώς το χαρακτηριστικό επωφελείται και η SPA για να επιτύχει τον απαραίτητο απομακρυσμένο συγχρονισμό μεταξύ των διασυνδεδεμένων χρηστών της και μάλιστα σε πραγματικό χρόνο.

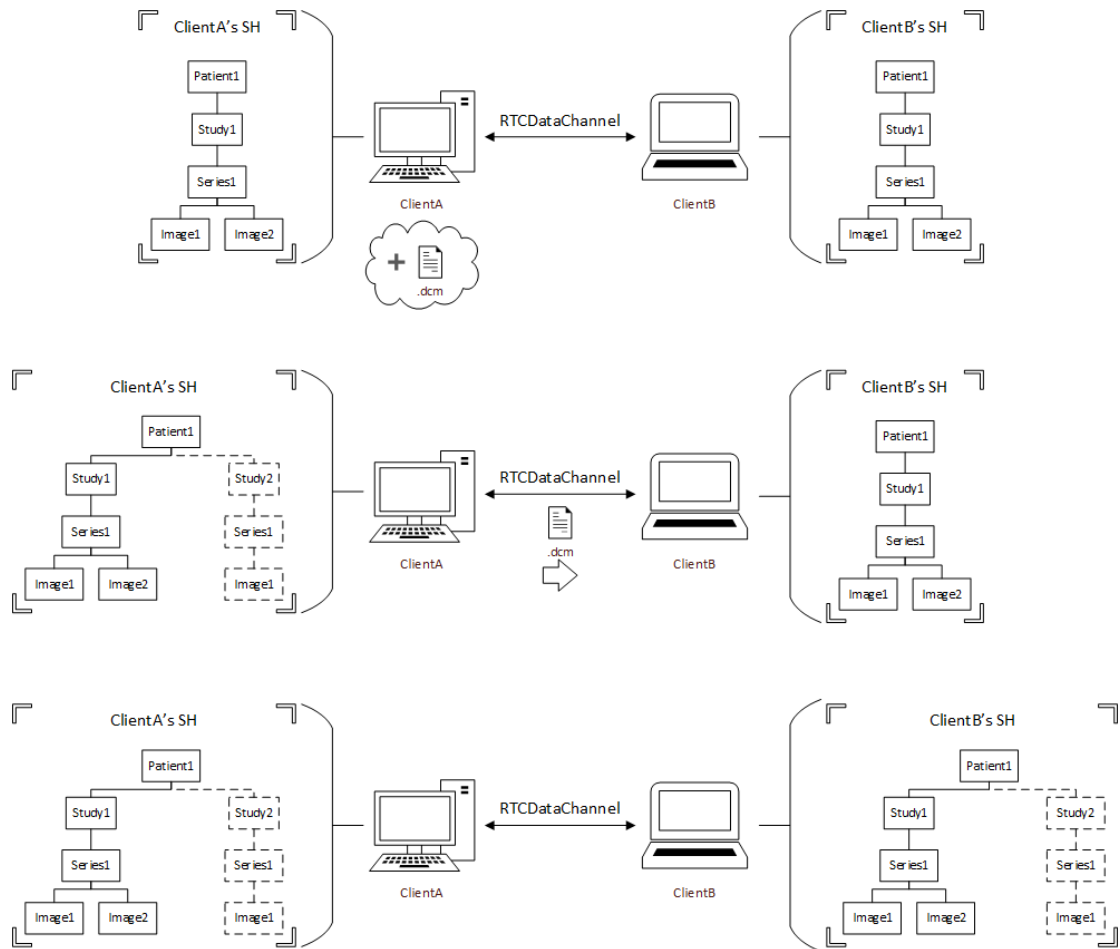
Για να γίνει αντιληπτός ο τρόπος με τον οποίο το παραπάνω γίνεται εφικτό, αξίζει να επισημανθούν ξανά κάποιες από τις βασικές μονάδες που δομούν την εφαρμογή. Η μονάδα Symmetric Hear, όπως έχει ήδη αναλυθεί, είναι υπεύθυνη για τη διατήρηση της κατάστασης σε επίπεδο εφαρμογής. Από κοινού με τη PeerConnector (που για τη SPA είναι η μονάδα εκείνη που διαχειρίζεται τη WebRTC επικοινωνία με συμπαγή και διάφανο τρόπο) είναι αυτές που επιτυγχάνουν την καταναμημένη αναπαράσταση και διαχείριση της κατάστασης του όλου συστήματος. Καθώς όλοι οι ομότιμοι διατηρούν ο καθένας τη δική του SH μονάδα, η τελευταία επαρκεί για να αποτελέσει παράλληλα και τον πυρήνα του μηχανισμού για το συγχρονισμό. Ο ρόλος της PeerConnector είναι να «αφουγκράζεται» και να αναμεταδίδει τυχόν αλλαγές της SH πάνω από τα RTCDataChannels, και το αντίθετο να γίνεται δηλαδή παραλήπτης οποιασδήποτε πληροφορίας φτάνει από το δίκτυο, να τη μοντελοποιεί ανάλογα με τη σημασιολογία της και να ενημερώνει τη SH. Με τον τρόπο αυτό συμβάντα πυροδοτούνται συμμετρικά ανάμεσα στις ομότιμες οντότητες, υλοποιώντας έναν καινοτόμο έμμεσο μηχανισμό RPC.

Η λογική αυτή περικλείεται παραστατικά στην *Εικόνα 39*. Έστω μια συνεδρία στην οποία συμμετέχουν δύο απομακρυσμένοι χρήστες. Θεωρείται πως έχουν φορτω-

θεί ήδη δύο εικόνες που αντιστοιχούν στο ίδιο ασθενή (δεν ενδιαφέρει ποιος τις φόρτωσε πραγματικά καθώς η εφαρμογή τις αντιμετωπίζει με ενιαίο τρόπο) και οι σωροί τους έχουν συγχρονιστεί πλήρως. Κάποια χρονική στιγμή, ο πρώτος (ClientA), επιλέγει να ανοίξει ένα τοπικό αρχείο DICOM. Η ενέργειά του αυτή θα πυροδοτήσει ένα συγκεκριμένο event που θα οδηγήσει αφενός στην εμπλοκή του parser προκειμένου να εξαχθεί η ενθυλακωμένη πληροφορία και κατόπιν να εισαχθεί στον τοπικό σωρό, αφετέρου στην αποστολή του αρχείου πάνω από το εγκαθιδρυμένο RTCDDataChannel προκειμένου να το παραλάβει ο ClientB και να το αναλύσει ο ίδιος με τη σειρά του. Με την ίδια λογική, η οποία βέβαια δεν απεικονίζεται, γίνεται η αμοιβαία εφαρμογή κάποιου φίλτρου σε μια συγκεκριμένη σειρά ή λήψη κάποιας μέτρησης πάνω σε μια συγκεκριμένη εικόνα. Αν θεωρήσουμε ως δράστη αυτή τη φορά τον ClientB, η επιλογή για την εφαρμογή αρνητικού φίλτρου μέσω του παρεχόμενου UI της εφαρμογής, πυροδοτεί σε τοπικό επίπεδο ένα διαφορετικό τώρα συμβάν που έχει ως αποτέλεσμα την εσωτερική αλλαγή κάποιου μοντέλου που κρατά (persist) την πληροφορία για τον τρόπο εμφάνισης της αντίστοιχης σειράς. Η αλλαγή τώρα αυτή, με τη σειρά της, πυροδοτεί ένα δεύτερο event παραλήπτης του οποίου είναι τόσο η μονάδα DIV όσο και η μονάδα PeerConnector. Η μεν πρώτη θα αναλάβει να ενημερώσει το <canvas>, η δε δεύτερη θα αναλάβει την αποστολή του ολόκληρου του μοντέλου στον ClientA. Η μονάδα PeerConnector του τελευταίου, όταν παραλάβει το μοντέλο (σε μορφή JSON), θα αναδομήσει την πληροφορία που μεταφέρει και θα ενημερώσει για τις διαφορές το αντίστοιχο τοπικό μοντέλο που περιέχεται στο σωρό του. Σε πλήρη συμμετρία με τον ομότιμό του, η αλλαγή που θα προκύψει, θα οδηγήσει σε αλλαγή του τρόπου εμφάνισης του δικού του <canvas>.

Η παραπάνω λογική κλιμακώνει όμοια και για περισσότερους εμπλεκόμενους. Γενικά, μέσω του μηχανισμού που περιεγράφηκε, οι ιατροί που εμπλέκονται σε μια συνεδρία μέσω της εφαρμογής, είναι σε θέση να φορτώσουν οποιοδήποτε αρχείο DICOM και να αρχίσουν να εργάζονται πάνω στην ιατρική εικόνα, γνωρίζοντας πως στον ελάχιστο δυνατό χρόνο¹⁶, οι υπόλοιποι όχι μόνο θα λάβουν την εν λόγω εικόνα αλλά και όσες αλλαγές έχει πραγματοποιήσει αυτός στο ενδιάμεσο. Στο μεταξύ, εφόσον η εικόνα έχει ληφθεί και προβληθεί, οι απομακρυσμένοι ιατροί είναι ελεύθεροι να συμμετάσχουν απρόσκοπτα στη διαχείριση και την επεξεργασία της. Και όλα αυτά, χωρίς να θυσιάζονται τα τεκμηριωμένα οφέλη που απορρέουν από τη χρησιμοποίηση οπτικοακουστικών μέσων κατά την πολυεπιστημονική διαχείριση ιατρικών υποθέσεων.

¹⁶ Η παρατήρηση αυτή αιτιολογείται λόγω της απουσίας ενδιάμεσων κατά την επικοινωνία. Ποιος γρηγορότερος τρόπος υπάρχει για να φτάσει κάποιος στον προορισμό του εκτός από τον προφανή (άμεσο); Στην πράξη βέβαια, για συνεδρίες άνω των δύο συμμετεχόντων, υπάρχει μια ακόμα βελτιστοποίηση που μπορεί να γίνει. Αποτελεί μελλοντικό στόχο και θα παρουσιαστεί σε επόμενη ενότητα.



Εικόνα 39. Διαδικασία απομακρυσμένου συγχρονισμού του SH πάνω από RTCDatChannel

5 Το σύστημα στην πράξη

Στο κεφάλαιο αυτό εξετάζονται πρακτικά ζητήματα που σχετίζονται με την εφαρμογή πελάτη. Αρχικά, γίνεται η παρουσίαση της διεπαφής χρήστη σε συνδυασμό με μέρος της παρεχόμενης λειτουργικότητας. Στη συνέχεια, αναφέρονται κάποια εξειδικευμένα θέματα υλοποίησης απαραίτητα για την κατανόηση του τρόπου δόμησης και λειτουργίας της εφαρμογής πελάτη και, τέλος, παρατίθεται μια συνολική εκτίμηση και αξιολόγηση της συνολικής υπηρεσίας.

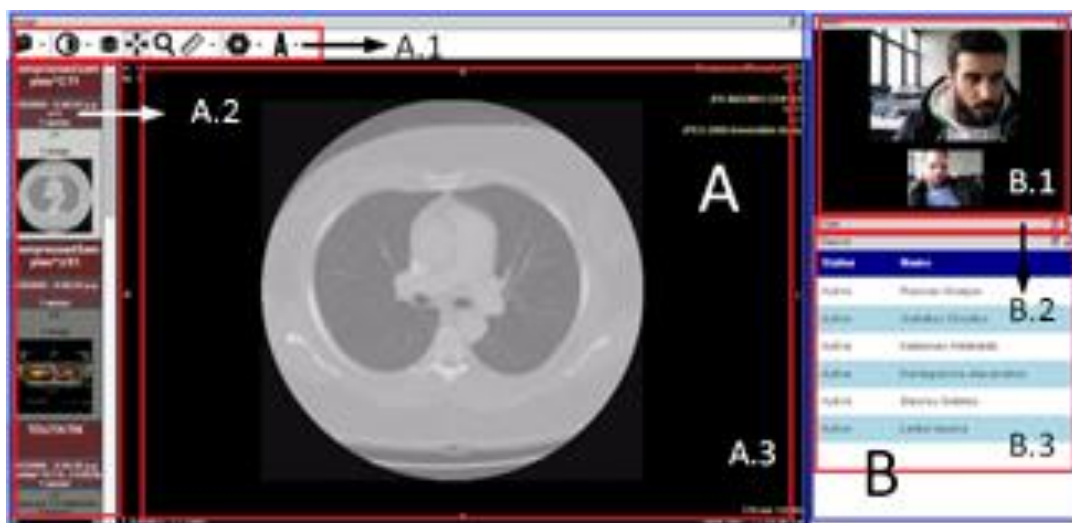
5.1 Παρουσίαση της διεπαφής χρήστη

Μία από τις πιο σπουδαίες προκλήσεις κατά την παράδοση και χρησιμοποίηση λύσεων με γνώμονα τα δεδομένα (data-driven solutions) για κάθε τύπο ανθρωπίνων διεργασιών είναι η παροχή εργαλείων οπτικοποίησης που είναι διαισθητικά και εύκολα στη χρήση τους από τους ειδικούς [52]. Η περιήγηση μεταξύ πολλαπλών πηγών DICOM, η συγκέντρωση και η απεικόνιση των απαιτούμενων στοιχείων για τη διαμόρφωση έγκυρων αποφάσεων και η επαναχρησιμοποίηση των ανακτημένων αποτελεσμάτων για την αντιμετώπιση προκλήσεων σε διαφορετικά πλαίσια είναι υψίστης σημασίας. Για να διευκολυνθεί όσο το δυνατόν περισσότερο η προβολή και αλληλεπίδραση με τις ιατρικές εικόνες με την ταυτόχρονη προαγωγή συνεργατικών χαρακτηριστικών, δημιουργήθηκε μια διεπαφή χρήστη αποτελούμενη από μια σειρά δομικών στοιχείων (components) για την υποστήριξη επεξεργασίας ιατρικής εικόνας σε πραγματικό χρόνο. Ο χώρος εργασίας που παρέχεται, εμπνεύστηκε από τις σχεδιαστικές κατευθυντήριες γραμμές που ακολουθούν καταξιωμένα λογισμικά ιατρικής απεικόνισης, όπως το RadiAnt.

5.1.1 SPA Layout

Η *Εικόνα 40* απεικονίζει μια γενική άποψη της διάταξης που υιοθετήθηκε από την εφαρμογή πελάτη επιδεικνύοντας όλα τα προαναφερθέντα χαρακτηριστικά. Το πλαίσιο A περιλαμβάνει το panel ιατρικής απεικόνισης, τον πυρήνα της SPA που φιλοξενεί όλη την ουσιαστική και απαραίτητη λειτουργικότητα για εποικοδομητικά MDTMs. Η γραμμή εργαλείων που φαίνεται στο άνω μέρος της (A.1) επιτρέπει την πρόσβαση σε τοπικά και απομακρυσμένα αρχεία DICOM καθώς επίσης και την αλληλεπίδραση του χρήστη με τις φορτωμένες ιατρικές εικόνες. Η στήλη που διακρίνεται στα αριστερά (A.2) φιλοξενεί μια προεπισκόπηση όλων των σειρών που έχουν φορτωθεί, ομαδοποιημένες κατά ασθενή και μελέτη στην οποία ανήκουν. Μέσω των προεπισκοπήσεων (thumbnails), ο χρήστης μπορεί να πλοηγηθεί στη σειρά που επιθυμεί και να εμφανίσει τις εικόνες που αυτή περιέχει μία-μία. Η πραγματική απεικόνιση λαμβάνει χώρα στο διπλανό κύριο διαδραστικό panel A.3 (στην ουσία ένα HTML5 Canvas element) και ο χρήστης μπορεί να την ελέγξει μέσω του ποντικιού και της παρεχόμενης εργαλειοθήκης. Το πλαίσιο B περιέχει το "social panel". Αποτελείται από τρία μικρότερα μέρη, τα οποία φιλοξενούν τις ροές βίντεο (B.1), τις γραπτές συνομιλίες (B.2) ή υποστήριξη των οποίων αναμένεται να προστεθεί άμεσα και τις επαφές (B.3) των συμμετεχόντων σε μια τηλεσυνεδρία. Μελλοντικά, όπως αναφέρεται και στη συνέχεια, θα επιδιωχθεί η παροχή λειτουργικότητας του panel επαφών όπως αυτής άλλων αντίστοιχων native και web εφαρμογών

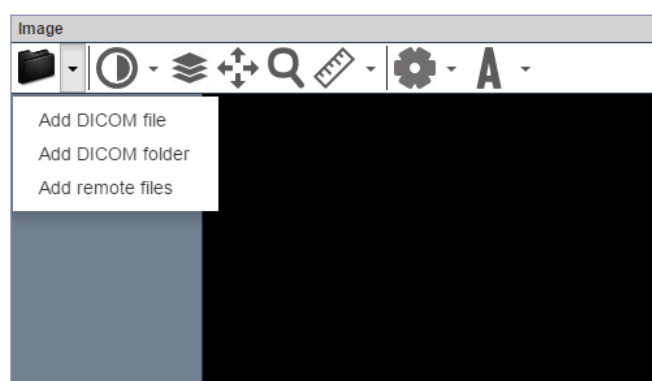
(όπως για παράδειγμα το Skype), με την οποία ο χρήστης θα μπορεί να ενημερώνεται για τη διαθεσιμότητα των άλλων και θα αυτοματοποιείται περαιτέρω η διαδικασία εγκαθίδρυσης κλήσεων μεταξύ των ενδιαφερομένων.



Εικόνα 40. Γενική διάταξη της SPA

5.1.2 Παρεχόμενη λειτουργικότητα διαχείρισης ιατρικής εικόνας

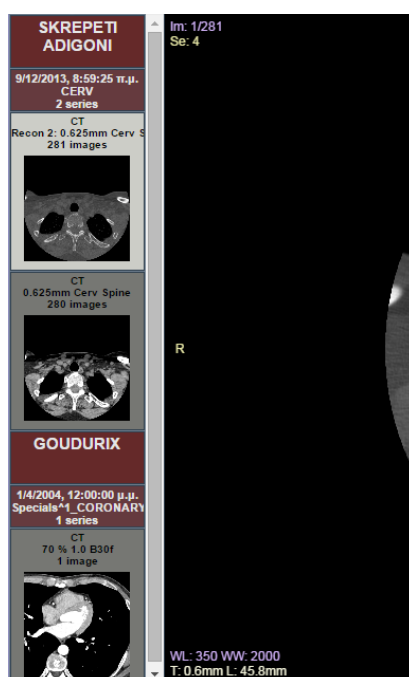
Μετά την επιτυχή φόρτωση της σελίδας, όπως είναι λογικό, αναμένεται από το χρήστη να εισάγει κάποια εικόνα προτού γίνει διαθέσιμη η υπόλοιπη λειτουργικότητα (Εικόνα 41). Του δίνεται η δυνατότητα για φόρτωση μεμονωμένων τοπικών αρχείων, ολόκληρων τοπικών καταλόγων (directories) που αντιστοιχούν σε σειρές (το χαρακτηριστικό αυτό λειτουργεί αυτή τη στιγμή μόνο στο Chrome), αλλά και αρχείων από απομακρυσμένους WADO εξυπηρετητές. Συγκεκριμένα, τη στιγμή της συγγραφής υποστηρίζεται σε δοκιμαστικό επίπεδο η επικοινωνία με τον Orthanc¹⁷.



Εικόνα 41. Υποστήριξη για άνοιγμα τοπικών αρχείων, καταλόγων και απομακρυσμένων (από WADO server)

¹⁷ Πρόκειται για ένα σχετικά απλό αλλά ισχυρό από άποψη δυνατοτήτων open-source και standalone DICOM εξυπηρετητή. Σχεδιάστηκε για να βελτιώσει τη ροή DICOM στα νοσοκομεία και για να υποστηρίξει την έρευνα γύρω από την αυτοματοποιημένη ανάλυση ιατρικών εικόνων. Το χαρακτηριστικό που τον κάνει μοναδικό είναι η παροχή ενός RESTful API, γεγονός που επιτρέπει την αλληλεπίδραση με αυτόν χρησιμοποιώντας οποιαδήποτε προγραμματιστική γλώσσα και πλατφόρμα [59].

Η εφαρμογή αντιμετωπίζει τα ανοιγμένα αρχεία με τον ίδιο τρόπο ανεξαρτήτως προελεύσεως. Επίσης, δεν υπάρχει κάποιο ανώτατο όριο όσον αφορά το υποστηριζόμενο αριθμό τους. Ο μόνος περιορισμός προκύπτει από τη μνήμη του εκάστοτε μηχανήματος στο οποίο εκτελείται η εφαρμογή. Οι εικόνες και η πληροφορία που τις συνοδεύει αποθηκεύονται εσωτερικά στο Symmetric Hear και ο χρήστης μπορεί να πλοηγηθεί σε αυτές μέσω thumbnails (Εικόνα 42). Η σειρά που απεικονίζεται στο κύριο panel (Εικόνα 40-Α.3) επισημαίνεται με πιο ανοιχτό χρώμα στη μπάρα προεπισκόπησης. Η ομαδοποίηση που ακολουθείται σε αυτή είναι η εξής: πρώτα παρουσιάζονται σε πλαίσιο τα στοιχεία του ασθενή (ονοματεπώνυμο), ακολουθούν τα χαρακτηριστικά γνωρίσματα της μελέτης (ημερομηνία εκτέλεσης, τύπος, κτλ.) και αμέσως μετά οι «φορτωμένες» σειρές που εκτός από τα thumbnails περιέχουν πληροφορίες όπως σύντομη περιγραφή τους, αριθμό περιεχόμενων εικόνων και τύπο μονάδας που τις παρήγαγε.



Εικόνα 42. Πλοήγηση μέσω thumbnails σε επίπεδο σειράς

Στην Εικόνα 43 απεικονίζονται τα βασικά εργαλεία που παρέχονται από την εφαρμογή για την αλληλεπίδραση με την ιατρική εικόνα και τις συνοδευόμενες πληροφορίες. Κατά σειρά είναι τα εξής:

- Ρύθμιση παραθύρου εικόνας (image window)
- Πλοήγηση μεταξύ εικόνων
- Μετακίνηση εικόνας
- Μεγέθυνση/σμίκρυνση εικόνας
- Εργαλεία ROI (Regions of Interest):
 - Μήκος
 - Επιφάνεια
 - Γωνία
 - Μονοπάτι
- Αλλαγή προσανατολισμού εικόνας
- Διαχείριση DICOM δεδομένων

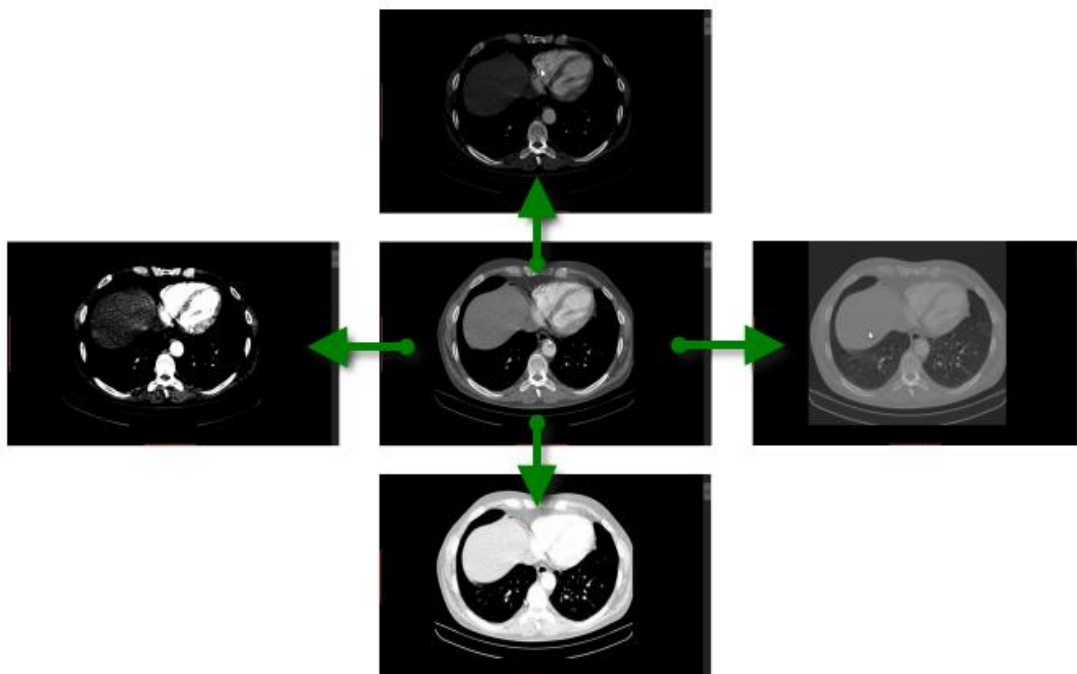


Εικόνα 43. Βασική εργαλειοθήκη

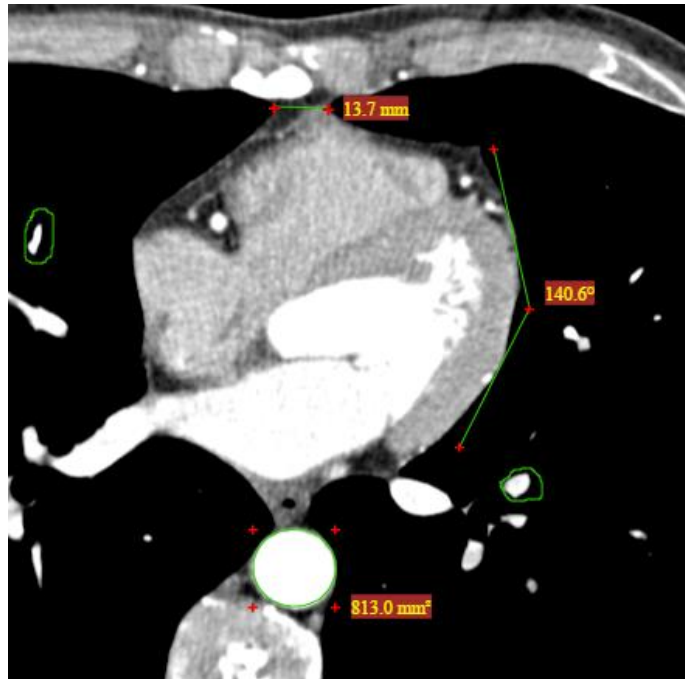
Τα πέντε πρώτα απεικονιζόμενα κουμπιά ελέγχουν τη λειτουργία click and drag του ποντικιού (stateful buttons). Έτσι, για παράδειγμα, αν πατηθεί το πρώτο (το οποίο στην ουσία ελέγχει τη φωτεινότητα και την αντίθεση της εμφανιζόμενης εικόνας), τότε το drag του ποντικιού:

- προς τα πάνω μειώνει τη φωτεινότητα,
- προς τα κάτω αυξάνει τη φωτεινότητα,
- αριστερά αυξάνει την αντίθεση,
- δεξιά μειώνει την αντίθεση,

όπως απεικονίζεται και στην *Εικόνα 44*. Αντίστοιχα, drag προς τα πάνω μεγεθύνει την εικόνα, τη μετακινεί προς τα πάνω ή προβάλλει την προηγούμενή της μέσα στη συγκεκριμένη σειρά, ενώ προς τα κάτω έχει ως αποτέλεσμα τη σμίκρυνσή της, τη μετακίνησή της προς τα κάτω ή την αντικατάστασή της με την επόμενη στη σειρά, αντίστοιχα. Παρόμοια λογική ακολουθεί και η λειτουργία μετρήσεων, με προεπιλεγμένο τη μέτρηση μήκους. Drag του ποντικιού πάνω από μια περιοχή της εικόνας σηματοδοτεί τη μέτρηση του μήκους της περιοχής αυτής σε πραγματικές διαστάσεις, συνήθως σε χιλιοστά (για το σκοπό αυτό αξιοποιείται ειδική πληροφορία που υπάρχει στο DICOM αρχείο της εικόνας), τη μέτρηση της επιφάνειάς της σε τετραγωνικά χιλιοστά ή τη μέτρηση κάποιας γωνίας σε μοίρες. Επίσης, δίνεται η δυνατότητα στο χρήστη για το σχηματισμό μονοπατιού πάνω στην εικόνα, για παράδειγμα στην περίπτωση που θέλει να επισημάνει απλά μια συγκεκριμένη περιοχή ενδιαφέροντος (*Εικόνα 45*).



Εικόνα 44. Λειτουργία αυξομείωσης φωτεινότητας και αντίθεσης εικόνας



Εικόνα 45. Μετρήσεις ROI

Το επόμενο πλήκτρο, μαζί με το dropdown menu που το συνοδεύει, παρέχει τη δυνατότητα για στροφή της εικόνας κατά 90, 180 και 270 μοίρες, ενώ το τελευταίο παρέχει τη δυνατότητα για απόκρυψη των ευαίσθητων στοιχείων του ασθενή, όπως το όνομά του, καθώς επίσης και των βοηθητικών σχολίων (annotations) που εμφανίζονται πάνω από την εικόνα. Τέλος, ο χρήστης μπορεί να εμφανίσει με τη μορφή modal view το περιεχόμενο του αρχείου DICOM που περιέχει την εκάστοτε προβαλλόμενη εικόνα (Εικόνα 46).

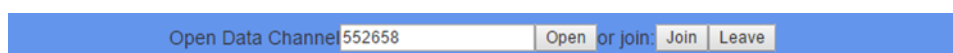
DICOM tags

×

Tag ID	VR	Length	Description	Value
(0002,0001)	OB	2	FileMetaInformationVersion	0 / 1
(0002,0002)	UI	26	MediaStorageSOPClassUID	1.2.840.10008.5.1.4.1.1.2
(0002,0003)	UI	46	MediaStorageSOPInstanceUID	1.3.12.2.1107.5.99.2.5562.4.0.575506410962905
(0002,0010)	UI	18	TransferSyntaxUID	1.2.840.10008.1.2
(0002,0012)	UI	22	ImplementationClassUID	1.3.6.1.4.1.19291.2.1
(0002,0013)	SH	10	ImplementationVersionName	OSIRIX001
(0002,0016)	AE	6	SourceApplicationEntityTitle	OSIRIX
(0008,0005)	CS	10	SpecificCharacterSet	ISO_IR 100
(0008,0008)	CS	34	ImageType	ORIGINAL / PRIMARY / AXIAL / CT_SOM5 SPI
(0008,0016)	UI	26	SOPClassUID	1.2.840.10008.5.1.4.1.1.2
(0008,0018)	UI	46	SOPInstanceUID	1.3.12.2.1107.5.99.2.5562.4.0.575506410962905
(0008,0020)	DA	8	StudyDate	20040401
(0008,0021)	DA	8	SeriesDate	20040401
(0008,0022)	DA	8	AcquisitionDate	20040420
(0008,0023)	DA	8	ImageDate	20040420
(0008,0030)	TM	14	StudyTime	120000.000000
(0008,0031)	TM	14	SeriesTime	120000.000000
(0008,0032)	TM	14	AcquisitionTime	133211.784922
(0008,0033)	TM	14	ImageTime	133211.784922
(0008,0060)	CS	2	Modality	CT
(0008,0070)	LO	8	Manufacturer	SIEMENS

Εικόνα 46. Modal view με τα περιεχόμενα του αρχείου DICOM

Όσον αφορά το κομμάτι της επικοινωνίας και της εγκαθίδρυσης απομακρυσμένων συνεδριών, η εφαρμογή μέχρι τη στιγμή της συγγραφής της διπλωματικής δεν υποστηρίζει απευθείας κλήσεις μεταξύ των χρηστών, μέσω δηλαδή του social panel. Αυτό συμβαίνει γιατί όλες οι διαθέσιμες WebRTC βιβλιοθήκες παρέχουν σχετικά περιορισμένες δυνατότητες για σηματοδότηση, χρησιμοποιώντας γενικού σκοπού APIs τρίτων (π.χ. Firebase). Η εξέλιξη ενός αντίστοιχου API ειδικής κατασκευής (custom) που θα εκπληρώνει τις αρχικές προδιαγραφές της συνολικής υπηρεσίας και θα υλοποιείται από την οντότητα PAS ανήκει στα άμεσα μελλοντικά πλάνα. Με βάση τώρα την υπάρχουσα κατάσταση της εφαρμογής, όπως απεικονίζεται και στην *Εικόνα 47*, στο πάνω μέρος της σελίδας υπάρχει μια κεντρική φόρμα που για κάθε instance της εφαρμογής που εκκινεί, περιέχει έναν αυτόματα δημιουργημένο τυχαίο αριθμό. Ο χρήστης, λοιπόν, που επιδιώκει να ξεκινήσει μια συνεδρία, δεν έχει παρά να πατήσει το κουμπί "Open" και ύστερα να αποστείλει τον αριθμό αυτό μέσω e-mail, SMS, ή οποιασδήποτε άλλης υπηρεσίας αρέσκει να χρησιμοποιεί στα άτομα που επιθυμεί. Όταν αυτά τον λάβουν, τον επικολλούν στη δική τους φόρμα και πατούν το κουμπί "Join". Τότε ξεκινά ουσιαστικά η λειτουργία εγκαθίδρυσης κλήσης και από το σημείο αυτό και μετά, η ροή εργασιών των συμμετεχόντων συγχρονίζει αυτόματα μεταξύ τους, ενώ παράλληλα, εφόσον το επιτρέψουν, ανταλλάσσουν βίντεο και ήχο.



Εικόνα 47. Απεικόνιση μηχανισμού εγκαθίδρυσης συνεδριών

5.2 Σενάρια χρήσης και αξιολόγηση

Μια συνηθισμένη εικόνα αξονικής τομογραφίας (CT image) έχει ανάλυση 512×512 εικονοστοιχεία για το κάθε ένα εκ των οποίων δεσμεύονται 16 bits. Το αντίστοιχο αρχείο DICOM λοιπόν μπορεί να θεωρηθεί ότι περιέχει περίπου 0.5 MB δεδομένων ($512 \times 512 \times 16$ bits) – το μέγεθος σε bytes της υπόλοιπης πληροφορίας είναι αμελητέα σε σύγκριση με την ίδια την εικόνα. Μια μέση σειρά αξονικών τομογραφιών (CT series) αποτελείται από 300 εικόνες, συνολικού μεγέθους 150 MB. Η μέγιστη μη απωλεστική συμπίεση που μπορεί να εφαρμοστεί στα δεδομένα είναι 3:1, μειώνοντας έτσι το μέγεθος στα 50 MB [53]. Θεωρώντας ένα απλό σενάριο, έστω δύο ιατροί συνδεδεμένοι απευθείας μέσω ενός RTCDataChannel (όπως στην *Εικόνα 39*). Με βάση την υπάρχουσα υλοποίηση, η ιατρική σειρά θα αποσταλεί πάνω από το δίκτυο μία μόνο φορά, τη στιγμή που ένας από τους δυο τους θα τη φορτώσει. Περαιτέρω ενέργειες πάνω στην εικόνα, προερχόμενες από οποιονδήποτε ιατρό, διαδίδονται μέσω του έμμεσου μηχανισμού RPC που αναλύθηκε προηγουμένως. Αντί, λοιπόν, να προκαλείται επιβάρυνση και κορεσμός του δικτύου λόγω συνεχούς αποστολής και λήψης δεδομένων εικόνας (προσέγγιση που ακολουθούν δημοφιλείς εφαρμογές screen-sharing) ως αποτέλεσμα και των πιο μικρών αλλαγών, ανταλλάσσονται αντικείμενα JSON (βλ. *Εικόνα 48*) τα οποία πρακτικά «χωρούν» σε IP πακέτα, μειώνοντας έτσι τον απαιτούμενο χρόνο επικοινωνίας και βελτιώνοντας τις δυνατότητες κλιμακωσιμότητας της εφαρμογής. Ενδεικτικά, μια ροή εργασίας αποτελούμενη από τρεις ενέργειες (π.χ. μεγέθυνση, μετακίνηση και εφαρμογή φίλτρου), έχει ως αποτέλεσμα την αποστολή τριών αντικειμένων JSON με μέσο μέγεθος 512 bytes. Σε κλασικά περιβάλλοντα διαμοιρασμού επιφάνειας εργασίας, οι παραπάνω αλληλεπιδράσεις θα απαιτούσαν την αποστολή τουλάχιστον μερικών δεκάδων KB δεδομένων εικόνας με ταυτόχρονη ορατή πτώση στην ποιότητά της. Η προσέγγιση αυτή απαιτεί σημαντικά λιγότερο bandwidth και επιφέρει

σημαντικά λιγότερες καθυστερήσεις καθώς οι συμμετέχοντες συνδέονται απευθείας μεταξύ τους. Η σχεδόν ταυτόχρονη δε εφαρμογή των ροών εργασίας τεκμηριώνει τη δυνατότητα για παροχή πραγματικού χρόνου υπηρεσιών από την εφαρμογή.

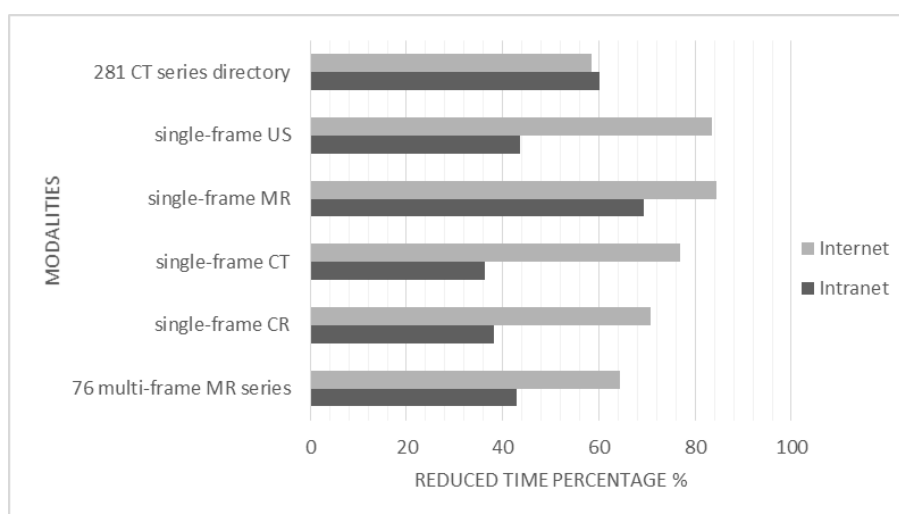
```

“{
  “peerID”:“ho4zpfu2iyhkt9”,
  “seriesInstanceUID”:“1.3.12.2.1107.5.99.2.5562.4.0.575080411334689”,
  “left”:458,
  “top”:317,
  “angle”:270,
  “scale”:1,
  “originX”:“center”,
  “originY”:“center”,
  “flipX”:false,
  “flipY”:false,
  “windowCenter”:40,
  “windowWidth”:400,
  “type”:“seriesRendererModel”,
  “negative”:false
}”

```

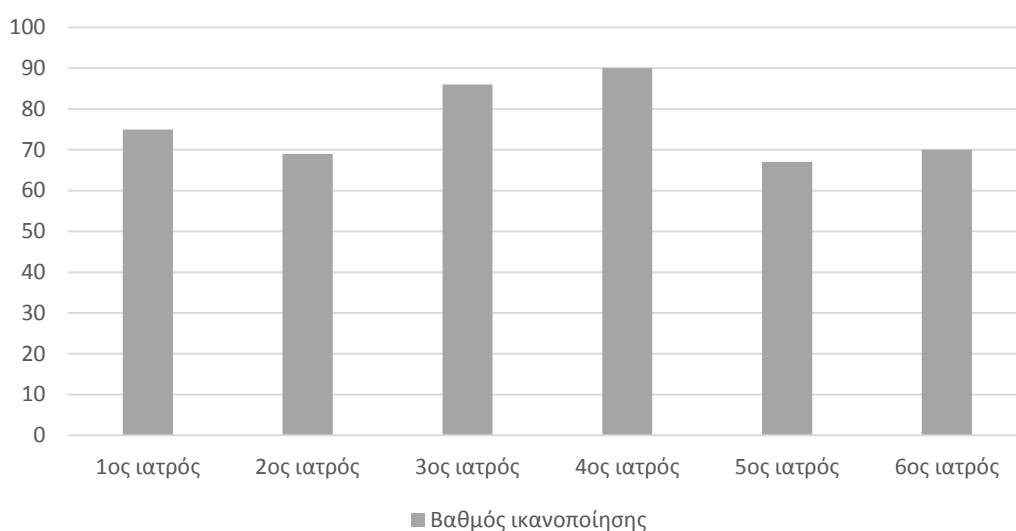
Εικόνα 48. Παράδειγμα JSON αντικειμένου που περιέχει πληροφορία για την απεικόνιση ιατρικής σειράς

Όπως μάλλον έχει γίνει ήδη κατανοητό στο σημείο αυτό, το πιο απαιτητικό κομμάτι, το οποίο αποτελεί και σημείο συμφόρησης για την εισαγόμενη πλατφόρμα, είναι ο αναπόφευκτος διαμοιρασμός των ιατρικών εικόνων. Όπως εισαγωγικά τεκμηριώθηκε, ο χρόνος του ιατρικού προσωπικού είναι πολύτιμος και η παρούσα λύση ευελπιστεί να συνεισφέρει στην όσο το δυνατόν πιο αποδοτική αξιοποίησή του. Για να το επιτύχει, εκμεταλλεύεται τους peer-to-peer διαύλους του WebRTC. Στην Εικόνα 49 παρουσιάζονται μετρήσεις που έγιναν σε διαφορετικά περιβάλλοντα (Internet και Intranet) χρησιμοποιώντας ένα σύνολο από ενδεικτικά αρχεία και καταλόγους DICOM προκειμένου να εγκαθιδρυθεί το επιχείρημα πως η peer-to-peer προσέγγιση εξασφαλίζει και στην πράξη καλύτερες επιδόσεις από μια αντίστοιχη, βασισμένη σε κεντρικό εξυπηρετητή εναλλακτική.



Εικόνα 49. Εκτίμηση επίδοσης του RTCDataChannel: Ποσοστό μείωσης χρόνου μεταφοράς DICOM αρχείων σε σχέση με κεντροποιημένες αρχιτεκτονικές

Δημιουργήθηκε επίσης ένα ειδικό SUS (System Usability Scale) ερωτηματολόγιο για την εκτίμηση της ευχρηστίας της εφαρμογής από τους ίδιους τους χρήστες για τους οποίους προορίζεται. Συμπληρώθηκε από έξι ιατρούς διαφορετικών ειδικοτήτων όλοι εκ των οποίων είχαν προηγούμενη εμπειρία από προγράμματα ιατρικής απεικόνισης. Ο μέσος βαθμός ικανοποίησης ξεπέρασε το 76%, ενώ τα αποτελέσματα ανά ιατρό ξεχωριστά παρατίθενται στην *Εικόνα 50*. Όσον αφορά την πρακτικότητα της SPA, τρεις στους έξι απάντησαν ότι τη βρίσκουν «εξαιρετικά πρακτική», δύο απάντησαν «αρκετά πρακτική» και ένας «πρακτική», γεγονός που τοποθετεί τη χρησιμότητά της κατά πολύ άνω του μετρίου. Σε ερώτηση σχετικά με τη φύση της εφαρμογής και κατά το πόσο προτιμάται σε σχέση με μια αντίστοιχη desktop εκδοχή, οι περισσότεροι δεν έδειξαν να επηρεάζονται εφόσον είναι λειτουργική και αξιόπιστη. Περαιτέρω σχόλια και κριτικές είχαν ως αντικείμενο τη διεπαφή του χρήστη, την απουσία κάποιων προηγμένων χαρακτηριστικών που συναντώνται σε καθιερωμένες standalone λύσεις, καθώς επίσης και θέματα κλιμακωσιμότητας στην περίπτωση πολλών συμμετεχόντων σε μια συνεδρία (άνω των τεσσάρων). Όλες οι παρατηρήσεις κρίνονται απολύτως λογικές και προβλέψιμες καθώς τα «μέτωπα» που δημιουργήθηκαν κατά τη διάρκεια της εξέλιξης της εφαρμογής ήταν τόσα που ήταν αδύνατο να αντιμετωπιστούν όλα με την ίδια προσοχή και με βέλτιστο τρόπο – κάποιοι συμβιβασμοί έπρεπε να γίνουν. Σε κάθε περίπτωση, αποτελούν προτεραιότητα για μελλοντικές βελτιώσεις.



Εικόνα 50. Αποτελέσματα SUS ερωτηματολογίου για την ευχρηστία της εφαρμογής πελάτη - SPA

6 Επίλογος

6.1 Σύνοψη και συμπεράσματα

Η προτεινόμενη συνεργατική πλατφόρμα προωθεί την απομακρυσμένη συνεργασία στο χώρο των επαγγελματιών υγείας, καθώς επίσης την εμπλουτίζει μέσω της ασφαλούς και απρόσκοπτης ανταλλαγής ιατρικών δεδομένων και της παράλληλης υποστήριξης για εφαρμογή εξελιγμένων ροών εργασίας. Η σημερινή παρατηρούμενη ροπή προς τη BYOD (Bring Your Own Device) λογική αποτέλεσε έναν από τους κυριότερους λόγους που οδήγησε στην εξέλιξη του πλαισίου εργασίας του πελάτη στον browser, αποφεύγοντας ταυτόχρονα προβλήματα συμβατότητας και διαλειτουργικότητας. Η εφαρμογή, λοιπόν, που παρουσιάστηκε, δεν απαιτεί καμία επιπλέον εγκατάσταση – είτε επέκτασης είτε εξωτερικού λογισμικού, ούτε ενημέρωση από την πλευρά του χρήστη. Στην ουσία αποτελεί μια τυπική εφαρμογή ιστού, στην οποία όλη η απαιτούμενη επεξεργασία λαμβάνει χώρα στην πλευρά του πελάτη (client-side) – αποφεύγεται έτσι η άσκημη επικοινωνία πάνω από το δίκτυο τη στιγμή που αξιοποιείται στο μέγιστο η επεξεργαστική ικανότητα των σύγχρονων συσκευών. Το τελευταίο αυτό στοιχείο σε συνδυασμό με τη peer-to-peer δικτύωση που χρησιμοποιείται για τη διασύνδεση των χρηστών, καθιστά το εργαλείο αρκετά κλιμακώσιμο και αποδοτικό προκειμένου να υποστηρίξει απομακρυσμένα MDTMs.

Η τελική διαγνωστική αξία της παρουσιαζόμενης συνεργατικής πλατφόρμας είναι ένα καθαρά υποκειμενικό ζήτημα που επαφίεται σε διαφορετικές περιστάσεις και συμμετέχοντες [54]. Εντούτοις, η παρούσα υλοποίηση αναμένεται να καλύψει σε ένα σημαντικό βαθμό τις σύγχρονες απαιτήσεις των χρηστών και να παράσχει χρήσιμα, λειτουργικά και ασφαλή χαρακτηριστικά για τα περισσότερα ιατρικά σενάρια χρήσης.

6.2 Μελλοντικοί στόχοι

Κατά τη σύγκριση των τελικών χαρακτηριστικών που παρέχει η πλατφόρμα σε σχέση με τις αρχικές φιλόδοξες προδιαγραφές που ετέθησαν, δείχνει να έχει εκπληρώσει ικανοποιητικά τις περισσότερες. Και όλα αυτά ως αποτέλεσμα ενός έργου που ξεκίνησε αρχικά με στόχο την εννοιολογική απόδειξη της δυνατότητας των σύγχρονων τεχνολογιών ιστού να υποστηρίξουν κάτι τόσο απαιτητικό όσο η διεξαγωγή online MDTMs. Αφού πλέον το στάδιο αυτό έχει περάσει και τα θεμέλια πάνω στα οποία εδραιώθηκε το εργαλείο – πλαίσιο εργασίας στο browser, επικοινωνία στηριγμένη στο WebRTC και απεικόνιση στο WebGL – αποδεικνύονται σταθερά¹⁸, εμφανίζονται νέοι δρόμοι και προοπτικές για περαιτέρω βελτιώσεις.

Πρωταρχικά, αυτό που χρίζει μελλοντικής προσοχής είναι η διεύρυνση της υποστήριξης του DICOM προτύπου. Το τελευταίο είναι ένα πρότυπο με εκτενείς προεκτάσεις και προδιαγραφές, η πλήρης κάλυψη των οποίων, όπως είναι φυσικό, κρίθηκε αδύνατη στα πλαίσια αυτής της διπλωματικής εργασίας. Επίσης δεν είναι λίγα τα παραδείγματα κατά τα οποία δυστυχώς δεν ακολουθούνται πιστά ούτε από τους ίδιους τους

¹⁸ Σήμερα, σχεδόν κάθε συσκευή, ανεξαρτήτως μεγέθους, έρχεται με κάποια έκδοση browser προεγκατεστημένη. Επιπρόσθετα, μπορεί ακόμα να μην υποστηρίζουν και οι πέντε μεγάλοι browser το WebRTC, αλλά υπάρχουν σημάδια ότι εν τέλει θα το κάνουν (Internet Explorer της Microsoft και Safari της Apple). Όταν δε, θα προτυποποιηθεί και επίσημα, θα εκλείψουν σιγά σιγά και τυχόν προβλήματα διαλειτουργικότητας. Όσον αφορά το WebGL, οποιαδήποτε συσκευή με σύγχρονη κάρτα γραφικών μπορεί να το υποστηρίξει απρόσκοπτα.

κατασκευαστές και παρόχους ιατρικού εξοπλισμού. Ιδιαίτερη έμφαση αξίζει να δοθεί και στην επικοινωνία με σύγχρονους WADO εξυπηρετητές που επιτρέπουν την πρόσβαση σε ιατρικές πηγές αποθηκευμένες σε απομακρυσμένα PACS.

Σημαντική προτεραιότητα (συνοδευόμενη και από πιθανό ερευνητικό ενδιαφέρον) αποτελεί και η βελτιστοποίηση της ανταλλαγής ιατρικών εικόνων DICOM πάνω από τα RTCDatChannels του WebRTC σε περιπτώσεις πολλαπλών συμμετεχόντων σε τηλεδιάσκεψη. Αντί να εξαντλείται το διαθέσιμο εύρος ζώνης του κάθε χρήστη όταν προσπαθεί να στείλει ένα μεγάλο αρχείο σε όλους τους υπόλοιπους συνδεδεμένους peers, θα επιχειρηθεί να εφαρμοστεί κάποια προσέγγιση παρόμοια με του Bittorrent¹⁹, στοχεύοντας στην επίτευξη ακόμα ποιοτικότερης εμπειρίας χρήσης (μειώνεται ο χρόνος αναμονής για τη λήψη και κατά συνέπεια την εμφάνιση των ιατρικών δεδομένων στην οθόνη του χρήστη).

Προς αυτή την κατεύθυνση (τη βελτίωση της εμπειρίας χρήσης), σαφείς βελτιώσεις μπορεί να δεχθεί ο σχεδιασμός του UI της εφαρμογής, σύμφωνα πάντα και με υποδείξεις/παρατηρήσεις των τελικών χρηστών. Προκειμένου, επίσης, η εφαρμογή να προσεγγίσει σε λειτουργικότητα άλλους standalone DICOM viewer, θα μπορούσε να δεχθεί προσθήκες για την υποστήριξη χαρακτηριστικών όπως MPR (MultiPlanar Reconstruction), 3D reconstruction, image fusion, κτλ.

Κλείνοντας, ο απώτερος – ερευνητικός κυρίως – στόχος που ανέκυψε κατά την εξέλιξη ακόμα της πλατφόρμας, είναι η ενσωμάτωση CAD δυνατοτήτων με σκοπό την περαιτέρω υποστήριξη της διαγνωστικής διαδικασίας. Έχει δειχθεί πως η ανάλυση περιοχών ενδιαφέροντος (ROIs) και η κατάταξή τους σε σημασιολογικά οργανωμένες ταξινομίες μπορεί να αυξήσει τις πιθανότητες ανακάλυψης καινοτόμου γνώσης μέσω της δημιουργίας νέων υποθέσεων [55]. Με αυτόν τον τρόπο, οι πόροι που θα γίνονται διαθέσιμοι θα είναι σε θέση να διευκολύνουν ιατρικό προσωπικό και ερευνητές διαφορετικών ειδικοτήσεων να εντοπίσουν συσχετισμούς μεταξύ του δικού τους πεδίου ενδιαφέροντος και εκείνων τα οποία έχουν ήδη μελετηθεί και ταξινομηθεί.

¹⁹ Το BitTorrent είναι ένα πρωτόκολλο μεταφοράς δεδομένων μέσω του διαδικτύου. Προσφέρει μια αξιόπιστη λύση για τη μεταφορά πολύ μεγάλων αρχείων, όπως για παράδειγμα τα αρχεία ήχου ή βίντεο, μετατρέποντας τον υπολογιστή κάθε χρήστη σε σημείο αναδιανομής. Με βάση αυτό το πρωτόκολλο, ο διανομέας ή κάτοχος του αρχείου, αντί να το διανέμει σε κάθε αιτητή ξεχωριστά, το αποστέλλει σε έναν, ο οποίος με τη σειρά του το αναδιανέμει σε άλλους αιτητές. Οι αιτητές διαμοιράζονται μεταξύ τους κομμάτια του αρχείου, τα οποία είναι μικρότερα σε μέγεθος από το αρχικό αρχείο, μέχρις ότου όλοι τους να ολοκληρώσουν τη λήψη του. Με αυτήν την τεχνική, καθίσταται δυνατή η μεταφορά τεράστιου όγκου δεδομένων μέσω του διαδικτύου αξιοποιώντας peer-to-peer διασύνδεση και χωρίς να είναι απαραίτητη η χρήση κεντρικών εξυπηρετητών. Επιπλέον, από την στιγμή που κάθε αιτητής κατεβάζει κομμάτια αρχείων από πολλούς διαμοιραστές ταυτόχρονα, είναι δυνατόν να επιτευχθούν υψηλές ταχύτητες λήψης.

7 Βιβλιογραφία

- [1] F. M. Boyle, E. Robinson, P. Heinrich και S. M. Dunn, «Cancer: communicating in the team game,» *ANZ Journal of Surgery*, τόμ. 74, αρ. 6, pp. 477-481, June 2004.
- [2] G. Delaney, S. Jacob, R. Iedema, M. Winters και M. Barton, «Comparison of face-to-face and videoconferenced multidisciplinary clinical meetings,» *Australasian Radiology*, τόμ. 48, αρ. 4, pp. 487-492, 2004.
- [3] B. Kane και L. Saturnino, «Multidisciplinary Medical Team Meetings: An Analysis of Collaborative Working with Special Attention to Timing and Teleconferencing,» *Computer Supported Cooperative Work (CSCW)*, τόμ. 15, αρ. 5-6, pp. 501-535, 2006.
- [4] B. Kane και S. Luz, «Probing the use and value of video for multi-disciplinary medical teams in teleconference,» σε *19th IEEE International Symposium on*, 2006.
- [5] I. Kunkler, G. Fielding, M. Macnab, S. Swann, J. Brebner, R. Prescott, R. Maclean, U. Chetty, A. Bowman, G. Neades, M. Dixon, M. Smith, A. Walls, J. Cairns, R. Lee, A. Lee και T. Gardner, «Group dynamics in telemedicine-delivered and standard multidisciplinary team meetings: results from the TELEMAM randomised trial,» *Journal of Telemedicine and Telecare*, τόμ. 12, pp. 55-58, 2006.
- [6] S. A. Bly, S. R. Harrison και S. Irwin, «Media spaces: bringing people together in a video, audio, and computing environment,» *Communications of the ACM*, τόμ. 36, αρ. 1, pp. 28-46, 1993.
- [7] T. Robertson, «The Public Availability of Actions and Artefacts,» *Computer Supported Cooperative Work (CSCW)*, τόμ. 11, αρ. 3-4, pp. 299-316, 2002.
- [8] J. E. Bardram, T. R. Hansen και M. Soegaard, «AwareMedia: a shared interactive display supporting social, temporal, and spatial awareness in surgery,» σε *CSCW*, Alberta, 2006.
- [9] R. L. Bashshur και C. E. Lathan, «Human Factors in Telemedicine,» *Telemedicine Journal*, τόμ. 5, αρ. 2, pp. 127-128, 2004.
- [10] C. LeRouge, M. Garfield και A. Hevner, «Quality attributes in telemedicine video conferencing,» σε *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, 2002.
- [11] S. M. Kaplan και G. Fitzpatrick, «Designing support for remote intensive-care telehealth using the locales framework,» σε *DIS '97 Proceedings of the 2nd conference on Designing interactive systems: processes, practices, methods, and techniques*, Amsterdam, 1997.
- [12] «Bitmap - Wikipedia, the free encyclopedia,» [Ηλεκτρονικό]. Available: <http://en.wikipedia.org/wiki/Bitmap>. [Πρόσβαση 10 6 2015].

- [13] C. Roobottom, G. Mitchell και G. Morgan-Hughes, «Radiation-reduction strategies in cardiac computed tomographic angiography,» *Clinical Radiology*, τόμ. 65, αρ. 11, pp. 859-867, 2010.
- [14] A. Arbor, «Collaboratory for Research on Electronic Work,» [Ηλεκτρονικό]. Available: <http://www.crew.umich.edu/CREW.home.html>.
- [15] InterMed Collaboratory, [Ηλεκτρονικό]. Available: <http://smi-web.stanford.edu/projects/>.
- [16] D. G. Kilman και D. W. Forslund, «An international collaboratory based on virtual patient records,» *Communications of the ACM*, τόμ. 40, αρ. 8, pp. 110-117, 1997.
- [17] V. Jagannathan, Y. V. Reddy, K. Srinivas, R. Karinthe, R. Shank, S. Reddy, G. Almasi, T. Davis, R. Raman, S. Qiu, S. Friedman, B. Merkin και M. Kilkenny, «An Overview of the CERC ARTEMIS Project,» σε *Proceedings of the Annual Symposium on Computer Application in Medical Care*, 1995.
- [18] «Virtual telemedicine Office,» [Ηλεκτρονικό]. Available: <http://www.telemedical.com>.
- [19] C. Chronaki, D. Katehakis, X. Zabulis, M. Tsiknakis και S. Orphanoudakis, «WebOnCOLL: Medical Collaboration in Regional Healthcare Networks,» *IEEE Transactions on Information Technology in Biomedicine*, τόμ. 1, αρ. 4, pp. 257-269, 1997.
- [20] I. Maglogiannis, C. Delakouridis και L. Kazatzopoulos, «Enabling Collaborative Medical Diagnosis Over the Internet via Peer-to-Peer Distribution of Electronic Health Records,» *Journal of Medical Systems*, τόμ. 30, αρ. 2, pp. 107-116, 2006.
- [21] L. Huang, «Semantic P2P Network for Healthcare,» σε *Fifth International Joint Conference on INC, IMS and IDC, 2009. NCM '09*, 2009.
- [22] M. Barhamgi, P.-A. C. Champin, D. Benslimane και A. M. Ouksel, «Composing Data-Providing Web Services in P2P-Based Collaboration Environments,» σε *Advanced Information Systems Engineering*, Springer Berlin Heidelberg, 2007, pp. 531-545.
- [23] G. J. Fakas και B. Karakostas, «A peer to peer (P2P) architecture for dynamic workflow management,» *Information and Software Technology*, τόμ. 46, αρ. 6, pp. 423-431, 2004.
- [24] S. Kailasam, S. Kumar και J. Dharanipragada, «Arogyasree: An Enhanced Grid-based Approach to Mobile Telemedicine,» *Int. J. Telemedicine Appl.*, τόμ. 2010, αρ. 1687-6415, pp. 2:1-2:11, 2010.
- [25] H. B. John, J. In-Yong, K. Ki-Hyun, L. Do-kwang, S. Rho και C.-s. Jeong, «Cloud-based active content collaboration platform using multimedia processing,» *EURASIP Journal on Wireless Communications and Networking*, τόμ. 2013, αρ. 1, pp. 1-13, 2013.
- [26] «W3Counter: Global Web Stats,» [Ηλεκτρονικό]. Available: <http://www.w3counter.com/globalstats.php>. [Πρόσβαση 3 6 2015].

- [27] «Browser Statistics,» [Ηλεκτρονικό]. Available: http://www.w3schools.com/browsers/browsers_stats.asp. [Πρόσβαση 3 June 2015].
- [28] «Canvas drawImage(canvasEl) vs putImageData,» [Ηλεκτρονικό]. Available: <http://jsperf.com/canvas-drawimage-vs-putimagedata/72>. [Πρόσβαση 3 6 2015].
- [29] «HTML - Wikipedia, the free encyclopedia,» [Ηλεκτρονικό]. Available: <http://en.wikipedia.org/wiki/HTML>. [Πρόσβαση 4 6 2015].
- [30] C. Lindley, DOM Enlightenment, O'Reilly Media, 2013.
- [31] B. Hunt, «How HTML, CSS and JavaScript work together in making web pages - guide to using XHTML, HTML, Javascript and CSS together,» [Ηλεκτρονικό]. Available: <http://webdesignfromscratch.com/html-css/how-html-css-js-work-together/>. [Πρόσβαση 4 6 2015].
- [32] «World Wide Web Consortium (W3C),» [Ηλεκτρονικό]. Available: <http://www.w3.org/>. [Πρόσβαση 8 6 2015].
- [33] «Google Trends - Ενδιαφέρον για Αναζήτηση Ιστού: adobe flex, microsoft silverlight, java applet, html5, gwt - Παγκοσμίως, 2004 - παρόν,» [Ηλεκτρονικό]. Available: <http://www.google.com/trends/explore#q=adobe++flex,microsoft+silverlight,java++applet,HTML5,gwt>. [Πρόσβαση 5 6 2015].
- [34] «Google Trends - Ενδιαφέρον για Αναζήτηση Ιστού: adobe flex, extjs, gwt, vaadin - Παγκοσμίως, 2004 - παρόν,» [Ηλεκτρονικό]. Available: <http://www.google.com/trends/explore#q=adobe++flex,extjs,gwt,vaadin>. [Πρόσβαση 5 6 2015].
- [35] «Canvas element - Wikipedia, the free encyclopedia,» [Ηλεκτρονικό]. Available: http://en.wikipedia.org/wiki/Canvas_element. [Πρόσβαση 6 6 2015].
- [36] «Scalable Vector Graphics - Wikipedia, the free encyclopedia,» [Ηλεκτρονικό]. Available: http://en.wikipedia.org/wiki/Scalable_Vector_Graphics. [Πρόσβαση 6 6 2015].
- [37] «The Khronos Group Inc.,» [Ηλεκτρονικό]. Available: <https://www.khronos.org/>. [Πρόσβαση 6 6 2015].
- [38] «OpenGL - The Industry Standard for High Performance Graphics,» [Ηλεκτρονικό]. Available: <https://www.opengl.org/>. [Πρόσβαση 6 6 2015].
- [39] T. Parisi, Programming 3D Applications with HTML5 and WebGL, O'Reilly Media, 2014.
- [40] «OpenGL Shading Language - Wikipedia, the free encyclopedia,» [Ηλεκτρονικό]. Available: http://en.wikipedia.org/wiki/OpenGL_Shading_Language. [Πρόσβαση 7 6 2015].
- [41] «three.js - Javascript 3D library,» [Ηλεκτρονικό]. Available: <http://threejs.org/>. [Πρόσβαση 7 6 2015].

- [42] «WindowTimers.setTimeout() 0- Web API Interfaces | MDN,» [Ηλεκτρονικό]. Available: <https://developer.mozilla.org/en-US/docs/Web/API/WindowTimers/setTimeout>. [Πρόσβαση 7 6 2015].
- [43] «WindowTimers.setInterval() - Web API Interfaces | MDN,» [Ηλεκτρονικό]. Available: <https://developer.mozilla.org/en-US/docs/Web/API/WindowTimers/setInterval>. [Πρόσβαση 7 6 2015].
- [44] «XMLHttpRequest - Wikipedia, the free encyclopedia,» [Ηλεκτρονικό]. Available: <http://en.wikipedia.org/wiki/XMLHttpRequest>. [Πρόσβαση 7 6 2015].
- [45] «SharedWorker - Web API Interfaces | MDN,» Mozilla Foundation, [Ηλεκτρονικό]. Available: <https://developer.mozilla.org/en-US/docs/Web/API/SharedWorker>. [Πρόσβαση 7 6 2015].
- [46] «ServiceWorker API - Web API Interfaces | MDN,» Mozilla Foundation, [Ηλεκτρονικό]. Available: https://developer.mozilla.org/en-US/docs/Web/API/ServiceWorker_API. [Πρόσβαση 7 6 2015].
- [47] «Internet Engineering Task Force (IETF),» [Ηλεκτρονικό]. Available: <https://www.ietf.org/>. [Πρόσβαση 8 6 2015].
- [48] «SIP: Session Initiation Protocol,» IETF, [Ηλεκτρονικό]. Available: <https://www.ietf.org/rfc/rfc3261.txt>. [Πρόσβαση 8 6 2015].
- [49] «The XMPP Standards Foundation,» [Ηλεκτρονικό]. Available: <http://xmpp.org/>. [Πρόσβαση 8 6 2015].
- [50] «WebSockets | MDN,» Mozilla, [Ηλεκτρονικό]. Available: <https://developer.mozilla.org/en-US/docs/WebSockets>. [Πρόσβαση 8 6 2015].
- [51] «kripken/j2k.js - GitHub,» GitHub, Inc., [Ηλεκτρονικό]. Available: <https://github.com/kripken/j2k.js>. [Πρόσβαση 20 6 2015].
- [52] M. R. Kamdar, D. Zeginis, A. Hasnain, S. Decker και H. F. Deus, «ReVealD: A user-driven domain-specific interactive search platform for biomedical research,» *Journal of Biomedical Informatics*, τόμ. 47, pp. 112-130, 2014.
- [53] N. H. Strickland, «Multidetector CT: what do we do with all the images generated?,» *The British Journal of Radiology*, τόμ. 77, αρ. suppl_1, pp. S14-S19, 2004.
- [54] P. Måseide, «The deep play of medicine: Discursive and collaborative processing of evidence in medical problem solving,» *Communication & Medicine*, τόμ. 3, αρ. ` , pp. 43-54, 2006.
- [55] M. Saleem, M. R. Kamdar, A. Iqbal, S. Sampath, H. F. Deus και A.-C. Ngonga Ngomo, «Big linked cancer data: Integrating linked TCGA and PubMed,» *Web Semantics: Science, Services and Agents on the World Wide Web*, Τόμ. %1 από %227-28, αρ. 0, pp. 34-41, 2014.

- [56] «Node.js,» [Ηλεκτρονικό]. Available: <https://nodejs.org/>. [Πρόσβαση 6 6 2015].
- [57] «Chromium Blog: The future of O3D,» [Ηλεκτρονικό]. Available: <http://blog.chromium.org/2010/05/future-of-o3d.html>. [Πρόσβαση 6 6 2015].
- [58] C. Vogt, M. Werner και T. Schmidt, «Leveraging WebRTC for P2P content distribution in web browsers,» σε *21st IEEE International Conference on Network Protocols (ICNP)*, 2013.
- [59] «Orthanc - DICOM server,» [Ηλεκτρονικό]. Available: <http://www.orthanc-server.com/>. [Πρόσβαση 18 6 2015].
- [60] «Medical Content Manager (MCM),» IBM, [Ηλεκτρονικό]. Available: <http://www.research.ibm.com/haifa/projects/software/mcm/index.html>. [Πρόσβαση 30 6 2015].